Begleitschreiben zum Abschlussbericht

Der vorliegende Abschlussbericht zur Sicherheitsanalyse der elektronischen Patientenakte (ePA) für alle wurde von der gematik in Auftrag gegeben, um die Sicherheit und Integrität dieser kritischen Infrastruktur zusätzlich von einer unabhängigen Stelle prüfen zu lassen. Die Analyse wurde mit dem Ziel durchgeführt, mögliche Schwachstellen zu identifizieren und Verbesserungspotenziale aufzuzeigen, die zur Erhöhung der Sicherheit und Effizienz der ePA für alle beitragen können.

Die elektronische Patientenakte wird ein wesentlicher Bestandteil der digitalen Gesundheitsversorgung in Deutschland sein. Behandelnde und Versicherte haben künftig wichtige, medizinisch relevante Informationen auf einem Blick verfügbar. Die Anwendung ermöglicht nicht nur mehr Transparenz, sondern auch mehr Effizienz im Gesundheitswesen, indem Doppeluntersuchungen vermieden und Behandlungsprozesse beschleunigt werden können.

Im Rahmen der Sicherheitsanalyse wurden verschiedene Testaktivitäten durchgeführt, die sowohl technische als auch organisatorische Aspekte der ePA umfassen. Die Ergebnisse zeigen, dass die Grundarchitektur der ePA Schutz gegen viele potenzielle Bedrohungen bietet. In kleinerem Umfang wurden spezifische Schwachstellen ermittelt, die behoben werden, um die Sicherheit weiter zu erhöhen.

Einige Ergebnisse der internen Schwachstellenbewertung führen zur Aufnahme neuer Anforderungen in die Spezifikationen. Andere Feststellungen können durch bereits festgelegte Spezifikationsinhalte gelöst werden, die nicht Teil der Prüfung waren oder werden bereits durch die Anforderungen des Bundesamtes für Sicherheit in der Informationstechnik (BSI) abgedeckt. Wenige Punkte liegen außerhalb der Regelungshoheit der gematik und können daher nur zur Kenntnis genommen werden. Darüber hinaus werden vereinzelte Aspekte über die individuellen Notfallkonzepte der Betreiber abgebildet.

Die gematik hat bereits Maßnahmen ergriffen, um die in der Analyse identifizierten Schwachstellen zu beheben. Die Erkenntnisse aus der Analyse sollen dazu beitragen, die neue ePA weiter zu optimieren. Die kontinuierliche Weiterentwicklung der Architektur der ePA für alle ist ein zentrales Anliegen der gematik. Dazu nutzen wir unter anderem Rückmeldungen von Expertinnen und Experten, Anbietern sowie Anwenderinnen und Anwendern. Der offene Dialog und die Zusammenarbeit mit verschiedenen Akteuren ist wichtig, um die Nutzererfahrung kontinuierlich verbessern zu können.

Wir danken allen Beteiligten für ihre wertvolle Arbeit und ihr Engagement bei der Durchführung dieser Analyse und freuen uns darauf, die ePA gemeinsam mit unseren Partnern weiterzuentwickeln.

Fraunhofer SIT

FRAUNHOFER-INSTITUT FÜR SICHERE INFORMATIONSTECHNOLOGIE

Sicherheitsanalyse des Gesamtsystems ePA für alle

Version 4.0

IN ZUSAMMENARBEIT MIT

ATHENE
Nationales Forschungszentrum für angewandte Cybersicherheit

Abschlussbericht Sicherheitsanalyse des Gesamtsystems ePA für alle

Fraunhofer-Institut für Sichere Informationstechnologie, SIT
in Darmstadt

Projektpartner: gematik GmbH

IN ZUSAMMENARBEIT MIT

ATHENE
Nationales Forschungszentrum
für angewandte Cybersicherheit

Versionshistorie

Version	Erstellungsdatum	Änderungen
0.1	29.05.2024	Initial
1.0	24.06.2024	Abschluss des Zwischenberichts für die Projektphase 1
2.0	19.07.2024	Abschluss des Zwischenberichts für die Projektphase 2
3.0	26.07.2024	Draft-Version Abschlussbericht
4.0	09.08.2024	Abschlussbericht

Inhaltsverzeichnis

Abbildungsverzeichnis

Abkürzungsverzeichnis

ASLR Address Space Layout Randomization

ATV Anbietertypvorschrift

BSI Bundesamt für Sicherheit in der Informationstechnik

CFI Control-Flow Integrity

CFG Control Flow Guard

CRA Cyber Resilience Act

CSRF Cross-Site-Request-Forgery

CSP Content Security Policy

CVSS Common Vulnerability Scoring System

DDoS Distributed Denial-of-Service

DoS Denial of Service

DPA Differential Power Analysis

eGK elektronische Gesundheitskarte

ePA elektronische Patientenakte

GPT Generative pre-trained transformers

HBA Heilberufsausweis

HSM Hardware-Sicherheitsmodul

IDP Identity Provider

IDS Intrusion Detection System

IPS Intrusion Prevention System

KIS Krankenhausinformationssystem

KVNR Krankenversichertennummer

LE Leistungserbringer

LEI Leistungserbringerinstitution

LL Low-Level

LLVM Low Level Virtual Machine

PVS Praxisverwaltungssystem

PTV Produkttypvorschriften

RAG Retrieval-Augmented Generation

RCE Remote Code Execution

SBOM Software Bill of Materials

SMC-B Security Module Card Typ B

SST Schnittstellenspezifikation

TI Telematikinfrastruktur

TL Top-Level

VAU vertrauenswürdige Ausführungsumgebung

WAF Web Application Firewall

XSS Cross-Site-Scripting

1 Management Summary

Die elektronische Patientenakte (ePA) ist ein wesentlicher Bestandteil der digitalen Transformation im Gesundheitswesen, der die Speicherung, Verarbeitung und den Austausch sensibler Gesundheitsdaten zwischen Patienten, Ärzten und anderen medizinischen Dienstleistern ermöglicht. Mit dem Ziel, die Gesundheitsversorgung effizienter zu gestalten, führt die Implementierung der ePA jedoch auch neue Risiken und Herausforderungen in Bezug auf die Datensicherheit und den Datenschutz ein. Dieser Bericht zielt darauf ab, eine umfassende Sicherheitsanalyse für die ePA zu präsentieren, die potenzielle Bedrohungen identifiziert und Maßnahmen zur Abwehr dieser Bedrohungen vorschlägt.

Für die Sicherheitsanalyse werden zwei verschiedene Methoden verwendet. Zum einen werden Angriffsbäume modelliert und nach möglichen Angriffswegen gesucht, zum anderen werden aus den Sicherheitsanforderungen mithilfe einer KI Cluster gebildet. Die Cluster helfen, Sicherheitslücken und Inkonsistenzen zu identifizieren.

Insgesamt wurden 21 Schwachstellen im Konzept identifiziert. Die Schwachstellen werden je nach Schweregrad als gering, mittel oder hoch eingestuft. Vier der 21 Schwachstellen werden als hoch, 6 der 21 Schwachstellen als mittel und 11 der 21 Schwachstellen als gering eingestuft.

Die als hoch eingestuften Schwachstellen beziehen sich auf folgende Themen:

- Anbieter des Aktensystems haben eine zu große Zeitspanne (72 h), um Schwachstellen an Wochenenden und Feiertagen zu bewerten.
- Es fehlt eine klare Rollentrennung der Mitarbeiter beim Umgang mit den Backups der Masterkeys zur Ableitung der Datenpersistierungsschlüssel.
- Fehlende Maßnahmen zur Rollentrennung von Mitarbeitern der Betreiber, um Angriffe auf die Verfügbarkeit der Akte zu verhindern.
- Fehlende Maßnahmen für einen sicheren Entwicklungsprozess bei den Herstellern des Aktensystems.

Für jede identifizierte Schwachstelle hat der Auftragnehmer Handlungsempfehlungen definiert. Die Schwachstellen und deren Handlungsempfehlungen sollten von der gematik geprüft werden. Die gematik sollte nicht tragbare Risiken durch die Anpassung bzw. Erweiterung der Sicherheitsanforderungen minimieren.

Neben den Schwachstellen wurden durch die Clusteranalyse insgesamt sieben Inkonsistenzen in den Dokumenten der gematik gefunden. Da es sich hierbei nicht um Schwachstellen handelt, wurden die Inkonsistenzen nicht bewertet. Es wird jedoch empfohlen, dass die gematik die Inkonsistenzen prüft und ggf. die Dokumentation anpasst.

2 Fazit

In dieser Untersuchung wurden die dokumentierten Anforderungen an die epa4all hinsichtlich Vollständigkeit, Widerspruchsfreiheit und Angemessenheit geprüft. Zuerst wurden mögliche Angriffe auf die epa4all mittels Angriffsbäumen modelliert. Anhand dieser Angriffsbäume wurden die relevanten Angriffspfade mit den dokumentierten Anforderungen abgeglichen, um das Restrisiko eines möglichen Angriffs einzuschätzen. Für die Pfade, bei denen das Restrisiko trotz der vorhandenen Anforderungen (und damit Gegenmaßnahmen gegen Angriffe) weiterhin signifikant war, wurden Empfehlungen für neue oder veränderte Anforderungen erstellt, um das Sicherheitsniveau der epa4all weiter zu erhöhen.

In Summe ergibt sich das Bild einer angemessenen Systemarchitektur, die jedoch mit besseren technischen und organisatorischen Maßnahmen gegen Innentäter abgesichert werden muss, insbesondere zur Sicherstellung der Verfügbarkeit. Hier sollte eine strikte Trennung von Rollen und Verantwortlichkeiten eingeführt werden.

Der unterspezifizierte Entwicklungsprozess für die einzelnen Komponenten legt weiteren Verbesserungsbedarf offen, um Supply-Chain-Angriffe zu vermeiden. Entsprechende Prozesse existieren und werden z. B. mit dem Cyber Resilience Act (CRA) der EU als sektorübergreifende Mindestanforderungen für alle Produkte mit digitalen Elementen eingeführt. Unabhängig von einer formalen Anwendbarkeit des CRA auf die epa4all, welche gesondert zu prüfen wäre, sollte mindestens ein vergleichbares Sicherheitsniveau erreicht werden.

Die umfangreichen Zugriffsberechtigungen der Leistungserbringer, die prinzipiell Zugriff auf Akten erhalten können, solange kein Widerspruch des Betroffenen vorliegt (opt-out), stellen eine Herausforderung für das Gesamtsystem dar. Es wird empfohlen, Anforderungen an die Primärsysteme der Leistungserbringer zu stellen. Ein unzureichend gesichertes Primärsystem genügt, um einen Datenverlust herbeizuführen, auch wenn die betroffenen Personen nie bei dem entsprechenden Leistungserbringer tatsächlich behandelt wurden.

Ähnliche Herausforderungen ergeben sich aus der föderierten Struktur z. B. bei der fehlenden Vorgabe an Anbieter von Aktensysteme, regelmäßige Penetrationstests durchzuführen, oder an definierte Anforderungen, welche Sicherheitsprüfungen beim Einreichen von Widersprüchen mindestens durchzuführen sind.

3 Projektüberblick

Das Projekt Sicherheitsanalyse des Gesamtsystems "ePA für alle" wurde von der gematik initiiert. Ziel des Projektes ist eine konzeptionelle Überprüfung, ob das Gesamtsystem "ePA für alle" auf Basis der zur Verfügung gestellten Dokumente und Spezifikationen in sich konsistent ist und keine formalen Sicherheitslücken aufweist. Das Projekt wurde in zwei Phasen unterteilt. Stufe 1 ist eine strukturelle Prüfung des Gesamtsystems 'ePA für alle'. Sie kann als High-Level-Prüfung verstanden werden. In Stufe 1 werden die infrage kommenden Angreifer modelliert und mithilfe von Angriffsbäumen mögliche Angriffsvektoren dargestellt. In Stufe 2 werden alle Sicherheitsanforderungen der gematik durch eine KI zu Clustern zusammengefasst, um die Prüfung weiter zu detaillieren.

Das Projekt erstreckte sich über folgenden Zeitraum:

Thema	Datum
Kick-off	08.05.2024
Vorbereitungen und Abstimmung	13.05.2024 bis 17.05.2024
Arbeiten an Stufe 1	17.05.2024 bis 24.06.2024
Abgabe Zwischenbericht Stufe 1	24.06.2024
Arbeiten an Stufe 2	24.06.2024 bis 19.07.2024
Abgabe Zwischenbericht Stufe 2	19.07.2024
Draft-Version Gesamtbericht	19.07.2024 bis 26.07.2024
Abgabe Draft-Version Gesamtbericht	26.07.2024
Review Draft-Version Gesamtbericht druch die gematik	29.07.2024 bis 04.08.2024
Finalisierung Gesamtbericht	05.08.2024 bis 11.08.2024

4 Methodik und Vorgehen

4.1 Methodik der Angriffsbäume

Die Angriffsbäume werden in zwei verschiedenen Ebenen dargestellt. Zunächst folgt das Top-Level (TL). Auf dieser Ebene werden die in Abschnitt 5.3.3 beschriebenen Angriffsziele modelliert. Die zweite Ebene stellen die Low-Level-Bäume (LL) dar. Die Low-Level-Bäume brechen die Angriffsziele der Top-Level-Bäume weiter herunter. Auf diese Weise ist die Darstellung der Bäume übersichtlicher und Low-Level-Bäume können so einfach für mehrere Top-Level-Bäume wiederverwendet werden.

Die Angriffsbäume verwenden UND- und ODER-Verknüpfungen. Bei einer ODER-Verknüpfung muss mindestens ein Blatt von einem Angreifer erfüllt werden, damit das übergeordnete Ziel (Knoten) erfüllt ist. Bei einer UND-Verknüpfung müssen immer alle Blätter erfüllt werden, damit das übergeordnete Ziel (Knoten) als erfüllt anerkannt wird. Im nächsten Schritt werden die Gegenmaßnahmen für die Blätter modelliert. Die Gegenmaßnahmen bestehen dabei aus den Sicherheitsanforderungen, die in den Dokumenten gefunden wurden.

Für jeden Angriffsbaum und jeden Angreifer wird eine Erfolgswahrscheinlichkeit berechnet. Dazu wird jedes Blatt (unterster Knoten) mit einer Wahrscheinlichkeit von 0,0 bis 1 (1 = 100% Erfolgswahrscheinlichkeit) versehen. Die Wahrscheinlichkeiten der Blätter werden in Abhängigkeit der Verknüpfung über alle Knoten hin bis zum obersten Angriffsziel (root) berechnet. Folgende Formeln werden dafür verwendet:

UND-Verknüpfung	$x \cdot y$
ODER-Verknüpfung	$x + y - x \cdot y$
Gegenmaßnahme	$x \cdot (1 - y)$

Die Bewertung der errechneten Werte erfolgt anhand folgender Skala:

Unmöglich	0,0
Niedrig	0,01 – 0,39
Mittel	0,4 – 0,69
Hoch	0,7 – 0,89
Sehr Hoch	0,9 - 1,0

Die IDs der Angreifer in den Angriffsbäumen können der Tabelle im Abschnitt 5.2.1 entnommen werden. Für Angreifer, welche nicht explizit in einem Angriffsbaum erwähnt werden, wird eine Erfolgswahrscheinlichkeit von 0 angenommen.

Bei der Berechnung der Angreifer werden die vier Angreifer der Kategorie Außentäter Hacker, Cyberkriminelle, Cracker und Hacktivists unter dem Begriff Außentäter zusammengefasst. Die ID für diese Gruppe lautet: "ar". Der Grund dafür ist, dass die Fähigkeiten der jeweiligen Angreifer sich nicht stark unterscheidet.

Im letzten Schritt wird geprüft, für welche Blätter keine Gegenmaßnahmen vorhanden sind. Falls es Blätter ohne Gegenmaßnahmen gibt, kann dies eine Indikation für eine Verbesserung im Konzept sein.

4.2 Eingesetzte Software für die Angriffsbäume

Die Modellierung der Angriffsbäume wurde mit der Open-Source-Software "ADTool"[1] umgesetzt. Die Software wurde im Verlauf des Projekts angepasst, damit es den Anforderungen zur Berechnung unterschiedlicher Angreifer gerecht wird. Die erweiterte Version von ADTool wird zusammen mit diesem Projektbericht bereitgestellt. Zur besseren Visualisierung wurden die Angriffsbäume mithilfe von Graphviz gezeichnet.

4.3 Methodik der Clustergenerierung

Für die Sicherheitsanalyse werden auf Ebene der Komponenten der elektronischen Patientenakte (ePA) zwei KI-gestützte Methoden eingesetzt. Zum einen wird eine KI-gestützte Clustergenerierung genutzt, um die hohe Anzahl von Anforderungen effizient zu gruppieren und zu kategorisieren, zum anderen kommt die speziell entwickelte gematik-GPT, eine fortschrittliche Retrieval-Augmented Generation (RAG) Pipeline, zum Einsatz, um präzise und kontextbezogene Informationen aus dem umfangreichen Spezifikationsdatensatz der gematik zu extrahieren und zu analysieren.

4.3.1 Überblick

Die Clustergenerierung und -analyse ermöglicht eine strukturierte Herangehensweise, um die komplexen Anforderungen effektiv zu erfassen und zu analysieren. Dieser Ansatz unterteilt die Anforderungen in logische Gruppen und erleichtert so eine systematische Untersuchung potenzieller Sicherheitsrisiken und Inkonsistenzen im System.

4.3.2 Clusterstruktur

Die Clusterstruktur wurde nach dem folgenden Schema aufgebaut:

[Komponente] / [Zielgruppe] / [Cluster] / [Subcluster]

Die Kategorien der ersten und zweiten Ebene ([Komponente]/[Zielgruppe]) wurden manuell erstellt und basieren auf den Anbietertypvorschriften (ATV), Produkttypvorschriften (PTV) und Schnittstellenspezifikationen (SST) der Komponenten aus dem Prüfumfang.

4.3.3 Mapping der Spezifikationsdokumente

Das Mapping der relevanten Spezifikationsdokumente auf die entsprechenden Komponenten und Zielgruppen erfolgt, wie in Tabelle 4.1 dargestellt.

4.3.4 Clusterbildung

Die eigentliche Clusterbildung auf der dritten und vierten Ebene erfolgte mittels KI-Algorithmen. Diese analysierten die Anforderungen und gruppierten sie nach inhaltlichen Gemeinsamkeiten und funktionalen Aspekten. Die resultierenden Cluster umfassen verschiedene Sicherheitsaspekte des ePA-Systems, wie beispielsweise Zugriffssteuerung, Datenintegrität und Verschlüsselung. Die Cluster wurden manuell nachbearbeitet, um fehlerhafte Zuordnungen der KI zu erkennen und zu bereinigen.

Beispiel der Clusterisierung

Um die Anwendung der Clusterisierung zu veranschaulichen, wird folgendes Beispiel betrachtet:

1 https://satoss.uni.lu/members/piotr/adtool/

Komponente/Zielgruppe	Quelle
Aktensystem/Hersteller Aktensystem/Anbieter	gemProdT_Aktensystem_ePA_PTV_3.0.1-0 gemAnbT_Aktensystem_ePA_ATV_3.0.1
Sektoraler IDP-Dienst/Hersteller Sektoraler IDP-Dienst/Anbieter	gemProdT_IDP-Sek_PTV_2.2.0-0 gemAnbT_IDP-Sek_KTR_ATV_1.2.0
Signaturdienst/Hersteller	gemProdT_SigD_PTV_1.3.1-0
Signaturdienst/Anbieter	gemAnbT_SigD_ATV_1.0.8
IDP-Dienst/Hersteller	gemProdT_IDP-Dienst_PTV_2.7.0-0
IDP-Dienst/Anbieter	gemAnbT_IDP-Dienst_ATV_1.0.7
Zentrales Netz der TI/Hersteller, Anbieter	gemProdT_ZentrNetz_PTV_1.6.7-0
Schnittstelle ePA-Ombudsstelle/Hersteller	gemSST_CS_ePA_Ombudsstelle_V_1.0.1
Schnittstelle ePA-Primärsysteme/Hersteller	gemSST_PS_ePA_V_1.0.1
Schnittstelle ePA-Clientsysteme/Hersteller	gemSST_CS_ePA_KTR_V_1.0.1

Tabelle 4.1 Mapping der Spezifikationsdokumente auf Komponenten und Zielgruppen

Signaturdienst / Hersteller / TLS-Verbindungen

Dieses Cluster repräsentiert Anforderungen, die sich auf die TLS-Verbindungen des Signaturdienstes beziehen und für Hersteller relevant sind. Es umfasst folgende Anforderungen:

- A_17322: TLS-Verbindungen nur zulässige Ciphersuiten und TLS-Versionen (ECC-Migration)
- A_17775: TLS-Verbindungen Reihenfolge Ciphersuiten (ECC-Migration)
- GS-A_4384-03: TLS-Verbindungen
- GS-A_5542: TLS-Verbindungen (fatal Alert bei Abbrüchen)
- A_21275-01: TLS-Verbindungen, zulässige Hashfunktionen bei Signaturen im TLS-Handshake
- GS-A_4662: Bedingungen für TLS-Handshake
- GS-A_4663: Zertifikats-Prüfparameter für den TLS-Handshake
- GS-A_5077: FQDN-Prüfung beim TLS-Handshake
- GS-A_5580-01: TLS-Klient für Betriebsunterstützende Dienste
- A_17527-01: Signaturdienst - Aufruf der Remote Operationen nur über geschützte Verbindung
- A_18464: TLS-Verbindungen, nicht Version 1.1
- A_18467: TLS-Verbindungen, Version 1.3
- GS-A_4385: TLS-Verbindungen, Version 1.2

Durch die Analyse dieses Clusters können spezifische Sicherheitsanforderungen für TLS-Verbindungen im Signaturdienst identifiziert und überprüft werden. Es ermöglicht eine umfassende Betrachtung verschiedener Aspekte der TLS-Konfiguration, einschließlich erlaubter Versionen, Ciphersuiten, Handshake-Bedingungen und Zertifikatsprüfungen.

Die Gruppierung dieser Anforderungen erleichtert es, Konsistenz und Vollständigkeit der TLS-Sicherheitsmaßnahmen zu überprüfen. Bei einer gleichzeitigen Betrachtung aller Anforderungen ohne Gruppierung in Cluster wäre es kaum möglich, durch rein manuelle Betrachtung einen Überblick herzustellen. Gleichzeitig ermöglicht die Gruppierung den Vergleich mit ähnlichen Clustern in anderen Komponenten, um eine einheitliche Umsetzung von TLS-Sicherheitsstandards im gesamten ePA-System sicherzustellen.

4.3.5 Fazit

Die Clusterisierung bietet mehrere Vorteile für die Sicherheitsanalyse des ePA-Systems:

1. **Systematische Untersuchung:** Die Clusterstruktur ermöglicht eine methodische Analyse aller Sicherheitsaspekte des ePA-Systems. Durch die Gruppierung verwandter Themen können gezielt und effizient spezifische Bereiche untersucht werden.
2. **Identifikation von Inkonsistenzen und Lücken:** Durch den Vergleich ähnlicher Cluster über verschiedene Komponenten und Zielgruppen hinweg können Widersprüche, Inkonsistenzen und fehlende Sicherheitsmaßnahmen leichter erkannt werden. Dies ergänzt die Schwachstellenanalyse mittels Angriffsbäumen um eine strukturelle Perspektive.
3. **Komponentenübergreifende Analyse:** Die Clusterisierung erleichtert die Identifikation von Querbeziehungen und Abhängigkeiten zwischen verschiedenen Systemkomponenten. Dies ist besonders wertvoll für die Analyse von Schnittstellen und übergreifenden Sicherheitskonzepten.
4. **Skalierbare Detailtiefe:** Die hierarchische Struktur erlaubt Analysen auf verschiedenen Detailebenen, von der Gesamtübersicht bis zu spezifischen Sicherheitsaspekten einzelner Komponenten. Dies unterstützt sowohl die strategische Planung als auch die technische Implementierung von Sicherheitsmaßnahmen.

Der analytische Ansatz nutzt die Vorteile der Clusterisierung, um die ganzheitliche Sicherheitsanalyse des ePA-Systems zu unterstützen.

Die durch die Clusteranalyse gefundenen Inkonsistenzen und fehlenden Anforderungen werden detailliert im Kapitel 7 erläutert. Dort werden die identifizierten Probleme systematisch aufgeführt. Zudem werden konkrete Empfehlungen zur Verbesserung des Gesamtkonzepts des ePA-Systems gegeben.

4.4 Einsatz von gematik-GPT

4.4.1 Überblick

Zur effizienten Informationsextraktion aus den umfangreichen Spezifikationsdokumenten der gematik wurde im Rahmen dieser Sicherheitsanalyse eine spezielle Retrieval-Augmented Generation (RAG) Pipeline aufgesetzt, die als "gematik-GPT" bezeichnet wird. Diese Pipeline ermöglicht es, relevante Informationen aus dem großen Datenbestand der gematik-Spezifikationen zielgerichtet zu extrahieren und zu verarbeiten.

gematik-GPT besteht aus drei Hauptkomponenten:

- **Data Warehouse**: Hier werden sämtliche relevanten gematik-Spezifikationsdokumente zur ePA gespeichert und indiziert. Dies umfasst technische Spezifikationen, Sicherheitsrichtlinien, Datenschutzkonzepte und Implementierungsleitfäden.
- **Vector Retrieval**: Bei einer Anfrage werden die relevantesten Dokumentenabschnitte mittels semantischer Ähnlichkeitssuche identifiziert.
- **Response Generation**: Ein großes Sprachmodell generiert, basierend auf den extrahierten Informationen, eine präzise Antwort. Dabei werden die Quelldokumente stets referenziert, um Nachvollziehbarkeit zu gewährleisten und eine manuelle Kontrolle der Antworten des gematik-GPT zu erlauben.

4.4.2 Funktionsweise anhand eines Beispiels

Die folgende Beispielfrage dient der Veranschaulichung der Funktionsweise von gematik-GPT: "Gibt es Anforderungen für parallele Zugriffe auf ein Aktenkonto?"

- Schritt 1: **Subquestion-Generierung**
  Zunächst wird die Eingabefrage an ein LLM mit einem speziellen Prompt übergeben, um mögliche Teilfragen zu generieren:
  - "Welche Anforderungen gibt es für gleichzeitige Zugriffe auf ein Aktenkonto?"
  - "Wie wird die Konsistenz bei parallelen Zugriffen auf ein Aktenkonto sichergestellt?"
  - "Gibt es Beschränkungen für die Anzahl gleichzeitiger Zugriffe auf ein Aktenkonto?"
- Schritt 2: **Vektorbasiertes Retrieval**
  Für jede Teilfrage werden nun mittels Vektor Retrieval die relevantesten Textpassagen aus dem Daten-Warehouse extrahiert. Beispielsweise könnte für die erste Teilfrage folgender Textauszug gefunden werden:
  "A_24696: Die VAU MUSS auch bei parallelen Zugriffen auf dasselbe Aktenkonto durch mehrere Nutzer immer einen konsistenten Zustand des Aktenkontos gewährleisten."
- Schritt 3: **Antwortgenerierung**
  Basierend auf den gefundenen relevanten Textpassagen generiert das LLM nun eine zusammenfassende Antwort:
  "Ja, es gibt spezifische Anforderungen für parallele Zugriffe auf ein Aktenkonto im ePA-System. Die Spezifikationen fordern explizit, dass bei parallelen Zugriffen durch mehrere Nutzer immer ein konsistenter Zustand des Aktenkontos gewährleistet werden muss. Quelle: gemSpec_Aktensystem_ePAfueralle_V1.1.0.pdf, Seite 77"

Es sei darauf verwiesen, dass die genannten Abfragen lediglich der Durchsuchung des Datenbestandes dienen. gematik-GPT wird nicht eingesetzt, um die identifizierten Anforderungen hinsichtlich ihrer Vollständigkeit, Angemessenheit oder Widerspruchsfreiheit zu bewerten. Die Bewertung erfolgt im Anschluss durch menschliche Experten.

4.4.3 Vorteile und Herausforderungen

Der Einsatz von gematik-GPT bietet mehrere Vorteile für die Sicherheitsanalyse:

- Schneller Zugriff auf relevante Informationen aus einem großen Dokumentenbestand
- Kontextbezogene Informationsextraktion
- Flexible Beantwortung komplexer Fragen
- Genaue Quellangaben

Allerdings müssen auch einige Herausforderungen berücksichtigt werden:

- Sensitivität gegenüber Frageformulierungen kann zu unterschiedlichen Antworten führen
- Mögliche Fehlinterpretationen komplexer technischer Zusammenhänge können dazu führen, dass für die Frage irrelevante Dokumente zurückgegeben werden

4.4.4 Fazit

gematik-GPT stellt ein vielversprechendes Werkzeug dar, um die Komplexität der ePA-Spezifikationen besser zu bewältigen und konsistente Informationen bereitzustellen. Gleichzeitig erfordert der Einsatz einer solchen KI-gestützten Lösung besondere Sorgfalt und kontinuierliche Überprüfung. Die Technologie sollte als Unterstützungswerkzeug betrachtet werden, das die menschliche Expertise ergänzt, aber nicht ersetzt.

Obwohl gematik-GPT als Werkzeug für die vorliegende Sicherheitsanalyse entwickelt wurde, besteht über das Projekt hinausgehendes Einsatzpotenzial für die Technologie. Hierauf kann bei Bedarf in Folgeprojekten aufgebaut werden. Eine Bereitstellung von gematik-GPT als Abfragewerkzeug ist denkbar, erfordert jedoch für eine gute Benutzbarkeit zusätzliche Engineering-Aufwände.

5 Bedrohungsmodellierung

Im Rahmen der ersten Stufe erfolgt eine Modellierung der Bedrohungen für die elektronische Patientenakte (ePA). Zentral ist dabei die Identifikation und der Schutz kritischer Daten und Systemkomponenten, die potenziellen Bedrohungen ausgesetzt sind. Im ersten Schritt werden die schutzbedürftigen Assets detailliert aufgelistet, welche sowohl Patientendaten als auch die Systeminfrastruktur umfassen. Anschließend erfolgt die Modellierung der potenziellen Angreifer, welche Interesse am Zugriff oder der Manipulation dieser Daten haben könnten, einschließlich Cyberkrimineller, unzufriedener Angestellter oder externer Hacker. Abschließend wird die Struktur der möglichen Angriffe mithilfe von Angriffsbäumen visualisiert und analysiert.

5.1 Schutzbedürftige Werte (Assets)

Als Assets werden schutzbedürftige Werte bezeichnet. Diese Werte spielen für die beteiligten Unternehmen und die Versicherten eine entscheidende Rolle. Angreifer werden daher versuchen, diese Werte anzugreifen. Zu Beginn der Sicherheitsanalyse werden die Werte der elektronischen Patientenakte identifiziert, um so ein besseres Verständnis über die Angreifer zu gewinnen.

5.1.1 Überblick

Bezeichner	ID	Referenz
Das Aktensystem	ast_epa_record	Abschnitt 5.1.2
Befugnisse	ast_epa_entitlements	Abschnitt 5.1.3
Widersprüche	ast_epa_consents	Abschnitt 5.1.4
Geräte	ast_epa_devices	Abschnitt 5.1.5
Digitale Identität	ast_epa_identity	Abschnitt 5.1.6
XDS Daten	ast_epa_data	Abschnitt 5.1.7
Metadaten	ast_epa_metadata	Abschnitt 5.1.8
Private Schlüssel	ast_epa_private_keys	Abschnitt 5.1.9
Masterkeys	ast_epa_masterkeys	Abschnitt 5.1.10

5.1.2 Das Aktensystem

Das zentrale Asset der elektronischen Patientenakte ist die Akte selbst. Sie steht im Mittelpunkt der Angreifer. Sie werden versuchen, die Verfügbarkeit, die Integrität und die Vertraulichkeit der Akte zu verletzen. Die folgenden Assets entsprechen teilweise einer Teilmenge der elektronischen Patientenakte.

5.1.3 Befugnisse

Befugnisse (Entitlements) sind ein relevantes Asset für die elektronische Patientenakte. Die Befugnisse entscheiden darüber, wer auf Daten innerhalb der elektronischen Patientenakte zugreifen oder nicht zugreifen darf. Angreifer könnten versuchen, die Befugnisse zu ihren Gunsten zu manipulieren.

5.1.4 Widersprüche

Versicherte können der Nutzung der elektronischen Patientenakte widersprechen, indem sie bei dem Kostenträger einen Widerspruch einlegen. Sobald ein Widerspruch eingelegt wurde, wird die gesamte Akte des Versicherten gelöscht. Angreifer könnten die Widersprüche missbräuchlich für sich nutzen, um gezielt Patientenakten zu löschen.

5.1.5 Geräte

Versicherte haben nur mit registrierten Geräten Zugriff auf die elektronische Patientenakte. Die Geräte bzw. deren Device-Token werden von dem Device Management verwaltet. Für potenzielle Angreifer sind die Geräte daher ein wichtiges Zwischenziel, um Zugriff auf die elektronischen Patientenakte zu erlangen.

5.1.6 Digitale Identität

Die Schutzziele Vertraulichkeit und Integrität der elektronischen Patientenakte werden durch die Authentizität der Nutzer des Verfahrens gewährleistet. Jeder Nutzer des Verfahrens besitzt eine digitale Identität, die über verschiedene Dienste der TI gewährleistet wird. Angreifer könnten versuchen, eine digitale Identität zu stehlen, um sich als andere Person auszugeben oder die Berechtigungen der anderen Person auszunutzen.

5.1.7 XDS Daten

Die Daten der Versicherten werden in den Systemen der Betreiber abgespeichert. Diese Daten sind ein zentrales Asset der elektronischen Patientenakte. Für Angreifer stellen die Daten damit auch eins der wichtigsten Angriffsziele dar. Angreifer könnten versuchen, die Daten zu lesen, zu löschen, zu stehlen oder zu manipulieren.

5.1.8 Metadaten

Metadaten der elektronischen Patientenakte entstehen automatisch bei der Verarbeitung von Daten. Die Metadaten sind für die Betreiber, Hersteller und Nutzer des Verfahrens nicht interessant. Für Angreifer allerdings enthalten Metadaten wichtige Informationen. Metadaten können Rückschlüsse auf Versicherte und deren Gesundheitsverlauf ermöglichen.

5.1.9 Private Schlüssel

Private Schlüssel sind geheime Informationen für kryptografische Verfahren. Angreifer könnten versuchen, in Besitz der privaten Schlüssel zu kommen. Sie könnten damit ggf. Daten signieren, verschlüsseln und entschlüsseln.

5.1.10 Masterkeys

Die Masterkeys sind wie die privaten Schlüssel geheime Informationen. Sie werden genutzt, um private Schlüssel abzuleiten. Falls Angreifer in Besitz der Masterkeys kommen, könnten sie in der Lage sein, ihre eigenen Schlüssel abzuleiten und damit die Daten der Versicherten zu entschlüsseln.

5.2 Angreifertypen

5.2.1 Überblick

In diesem Kapitel wird dargestellt, welche Typen von Angreifern potenziell auf das ePA System einwirken können. Anschließend werden die einzelnen Angreifertypen anhand verschiedener Metriken, wie finanzieller Ausstattung oder technischer Expertise, kategorisiert. Diese Kategorisierung erlaubt, die realistisch anzunehmenden Möglichkeiten der einzelnen Angreifertypen abzuschätzen und mögliche Sicherheitsmaßnahmen in einem wirtschaftlichen Kontext qualitativ zu bewerten.

Nicht explizit betrachtet werden Kooperationen zwischen Angreifern. Es steht der Angreifer, der ein eigenes übergeordnetes Ziel (z. B. finanziellen Vorteil) verfolgt, im Vordergrund.

Bedient dieser sich weiterer Personen als Erfüllungsgehilfen, sind diese trotzdem durch seine Zielsetzungen (und Inhibitoren) sowie seine Möglichkeiten zur Beauftragung (i. d. R. Finanzielle Möglichkeiten) gebunden. Ein Erfüllungsgehilfe, der aufgrund eigener Motivation beiträgt, wäre ein eigener Angreifertyp, der individuell zu modellierende Ziele verfolgt. Komplexe Kooperationen außerhalb von Auftragsverhältnissen, z. B. zwischen Regierungsorganisationen und Cyberkriminellen[2], werden aus Komplexitätsgründen an dieser Stelle nicht betrachtet.

Nicht explizit erwähnt werden mögliche Unterauftragnehmer von Betreibern oder Herstellern (z. B. IT-Dienstleister) einzelner Komponenten der Telematikinfrastruktur. Es wird davon ausgegangen, dass die Subauftragnehmer über dieselben Fähigkeiten, Berechtigungen und Wissensstände verfügen wie der Anbieter oder der Hersteller selbst.

Für jeden Angreifer wird eine Relevanz ermittelt, um die Wahrscheinlichkeit eines Einwirkens dieses Angreifers abzuschätzen. Die Relevanz ist dabei ein kombiniertes Maß für die Eintrittswahrscheinlichkeit einer schadhaften Aktion und einer a-priori-Schätzung der Erfolgswahrscheinlichkeit auf Basis der Möglichkeiten des jeweiligen Angreifers. Die Einbeziehung der a-priori-Erfolgswahrscheinlichkeit beruht darauf, dass ein Angreifer einen erkennbar aussichtslosen, aber Ressourcen beanspruchenden Angriff nicht durchführen wird. Die präzisen Angriffswege werden in einem späteren Teil der Analyse beleuchtet. In diesem Stadium wird eine a-priori-Überabschätzung über alle Angriffswege auf Basis von Erfahrungswerten vorgenommen. Wurde während der Analyse der konkreten Angriffswege festgestellt, dass die a-priori-Schätzungen nicht passend sind, wurden sie nachträglich angepasst.

Bezeichner	ID	Referenz
Regierungsorganisation	att_state	Abschnitt 5.2.2
Hacker	att_hacker	Abschnitt 5.2.3
Cyberkriminelle	att_criminal	Abschnitt 5.2.4
Cracker	att_cracker	Abschnitt 5.2.5
Hacktivists	att_hacktivists	Abschnitt 5.2.6
Hersteller des Aktensystems	att_hm	Abschnitt 5.2.7
Betreiber des Aktensystems	att_bm	Abschnitt 5.2.8
Betreiber des sektoraler IDP	att_sp	Abschnitt 5.2.9
Betreiber des IDP	att_ip	Abschnitt 5.2.10
Betreiber des SigD	att_sd	Abschnitt 5.2.11
Mitarbeiter der gematik	att_gk	Abschnitt 5.2.12
Hersteller des Primärsystems	att_hs	Abschnitt 5.2.13
Versicherte Person	att_vn	Abschnitt 5.2.14
Vertreter der versicherten Person	att_vr	Abschnitt 5.2.15
Leistungserbringer	att_lr	Abschnitt 5.2.16
Kostenträger	att_kr	Abschnitt 5.2.17
Ombudsstelle	att_oe	Abschnitt 5.2.18
E-Rezept-Fachdienst	att_et	Abschnitt 5.2.19

2 Beispiele hierfür sind bekannt in Bezug auf "Hochrisikoländer"

5.2.2 Regierungsorganisation

Titel	Regierungsorganisation	
ID	att_state	
Kategorie	Außentäter	
Ziele	Spionage, Cyberkrieg	
Möglichkeiten	Zugang zu Systemen	Nein
	Insider	Nein
	Technische Expertise	Hoch
	Finanzielle Möglichkeiten	Hoch
Relevanz	Hoch	

Regierungsorganisationen sind daran interessiert, vertrauliche Informationen über Bürger zu sammeln, die Infrastruktur anderer Staaten anzugreifen oder nachrichtendienstliche Informationen zu gewinnen. Sie verfügen über erhebliche finanzielle und technische Ressourcen sowie hoch qualifiziertes Fachpersonal. Ihre Ziele sind politisch und nicht wirtschaftlich motiviert, weshalb die Angriffe keinen finanziellen Gewinn erzielen müssen.

In Bezug auf das ePA-System wurde nach Absprache mit der gematik festgelegt, dass Angriffe durch Regierungsorganisationen nicht relevant sind.

5.2.3 Hacker

Titel	Hacker	
ID	att_hacker	
Kategorie	Außentäter	
Ziele	Kompromittierung von Anwendungen und Daten, »Hacking for Fame«, Plattform zur Durchführung von »Hacks« auch auf angeschlossene Systeme, finanzieller Zugewinn	
Möglichkeiten	Zugang zu Systemen	Nein
	Insider	Nein
	Technische Expertise	Hoch
	Finanzielle Möglichkeiten	Mittel
Relevanz	Hoch	

Hacker sind daran interessiert, Schwachstellen in Anwendungen zu finden, um ihre Fähigkeiten zu verbessern und sich einen Namen zu machen. Sie sind technisch versiert und verfügen über erhebliche technische Mittel, aber ihre finanziellen Möglichkeiten sind eher begrenzt und entsprechen denen von Privatpersonen oder kleinen Gruppen. Der finanzielle Gewinn steht für Hacker nicht im Vordergrund, sondern die technische Herausforderung und das Ansehen in der Community.

Angriffe auf sicherheitsrelevante Anwendungen wie das ePA-System stellen eine besondere Motivation für Hacker dar, da hier höhere Sicherheitsanforderungen bestehen. Der damit verbundene potenzielle Prestigegewinn und finanzielle Anreize können zusätzliche Motivationen sein.

Insgesamt wird die Relevanz von Hackern für das vorliegende Szenario als hoch eingeschätzt.

5.2.4 Cyberkriminelle

Titel	Cyberkriminelle	
ID	att_criminal	
Kategorie	Außentäter	
Status	Entwurf	
Ziele	Finanzieller Profit	
Möglichkeiten	Zugang zu Systemen	Nein
	Insider	Nein
	Technische Expertise	Mittel
	Finanzielle Möglichkeiten	Mittel
Relevanz	Mittel	

Cyberkriminelle sind auf finanziellen Gewinn aus und versuchen, durch Angriffe auf IT-Systeme mit minimalem Aufwand maximalen Profit zu erzielen. Typischerweise verkaufen sie gestohlene Daten oder erpressen Opfer, beispielsweise durch Ransomware-Angriffe. Auch Betrug, wie etwa Kreditkartenbetrug, ist gängig.

Cyberkriminelle haben im Allgemeinen weniger technisches Wissen als Hacker. Sie verfolgen zwei Hauptansätze:

Angriff in der Breite Diese Angreifer nutzen leicht ausnutzbare Schwachstellen, wie ungepatchte Software oder Phishing, und greifen möglichst viele Ziele gleichzeitig an, um die Erfolgsquote zu maximieren.

Zielgerichteter Angriff Hier fokussieren sich die Angreifer auf wertvolle Ziele und investieren erheblichen Aufwand, solange der erwartete finanzielle Gewinn hoch genug ist.

Für das ePA-System sind Breitenangriffe weniger relevant, da grundlegende Sicherheitsmaßnahmen diese oft abwehren können. Zielgerichtete Angriffe, die spezifisch auf das ePA-System abzielen, sind jedoch denkbar, insbesondere durch Phishing, um Nutzer zur Bestätigung betrügerischer Transaktionen zu verleiten.

Insgesamt wird die Relevanz von Cyberkriminellen für das ePA-System als mittel eingeschätzt.

5.2.5 Cracker

Titel	Cracker	
ID	att_cracker	
Kategorie	Außentäter	
Ziele	Zugriff auf die technische Infrastruktur wie Netzwerk- und Rechenkapazitäten	
Möglichkeiten	Zugang zu Systemen	Nein
	Insider	Nein
	Technische Expertise	Mittel
	Finanzielle Möglichkeiten	Niedrig
Relevanz	Niedrig	

Cracker sind daran interessiert, Server und Dienste für illegale Aktivitäten wie den Betrieb von Botnetzen oder das Mining von Kryptowährungen zu missbrauchen. Die Art der angegriffenen Systeme ist ihnen egal, Hauptsache, sie können die Rechen- und Netzwerkkapazitäten ausnutzen.

Ihre Angriffe sind meist automatisiert und zielen auf bekannte Schwachstellen in Standardsoftware ab. Bei sicherheitsrelevanten Anwendungen wie dem ePA-System ist die Erfolgswahrscheinlichkeit solcher Angriffe gering. Cracker werden daher als von niedriger Relevanz eingestuft, da es für ihre Zwecke deutlich einfachere Ziele gibt.

5.2.6 Hacktivisten

Titel	Hacktivisten	
ID	att_hacktivists	
Kategorie	Außentäter	
Ziele	Politischer Aktivismus, Zufügen von Reputationsschäden gegen politische Institutionen und relevante Unternehmen, Exfiltration und Veröffentlichung vertraulicher Daten	
Möglichkeiten	Zugang zu Systemen	Nein
	Insider	Nein
	Technische Expertise	Mittel
	Finanzielle Möglichkeiten	Niedrig
Relevanz	Mittel	

Hacktivisten verfolgen politische Ziele und wollen durch öffentlichkeitswirksame Aktionen, wie das Defacement von Webseiten oder das Veröffentlichen erbeuteter Daten, Aufmerksamkeit erregen. Ihre Ziele sind auf politisch motivierte Einrichtungen oder Unternehmen beschränkt.

Hacktivisten haben meist geringe finanzielle Mittel, können sich jedoch in losen Gruppen organisieren und technisches Wissen aufbauen. Ihre Angriffe auf das ePA-System würden zwar negative Auswirkungen auf die medizinische Versorgung haben und entsprechend bei der Mehrheit der Bürger auf wenig Verständnis stoßen. Dennoch ist die ePA ein politisch relevantes Thema und nicht alle Aktivisten stehen einer elektronischen Patientenakte im Allgemeinen oder der konkreten Umsetzung im Speziellen positiv gegenüber. Daher kann das eigene politische Interesse genügen, um Angriffe durchzuführen.

Der Aufwand für solche Angriffe ist hoch. Dennoch ist nicht auszuschließen, dass sie nach Sicherheitslücken suchen, um das System zu diskreditieren, oder bei entsprechender Gelegenheit (z. B. Zugriff über kompromittierte Drittsysteme) auch Schaden verursachen.

In Summe wird die Relevanz von Hacktivisten als mittel eingeschätzt.

5.2.7 Hersteller des Aktensystems

Titel	Hersteller des Aktensystems	
ID	att_hm	
Kategorie	Innentäter	
Ziele	Finanzieller Profit, Schädigung des eigenen Arbeitgebers	
Möglichkeiten	Zugang zu Systemen	Ja
	Insider	Ja
	Technische Expertise	Mittel
	Finanzielle Möglichkeiten	Niedrig
Relevanz	Hoch	

Die Hersteller des Aktensystems sind Mitarbeiter von Unternehmen, welche an der Entwicklung des Aktensystems oder des Frontends beteiligt sind. Die Mitarbeiter haben vollen Zugriff auf den Quellcode der Applikationen. Auf diese Weise könnten gezielt Backdoors oder Schwachstellen eingebaut werden. Zusätzlich besitzen sie das Wissen über den archi-

tektonischen Aufbau der Applikationen und können so deutlich gezielter Schwachstellen identifizieren. Für Angreifer der Kategorie Außentäter kommen die Hersteller des Aktensystems als Angriffsziel infrage. Außentäter könnten versuchen, die Software direkt im Entwicklungsprozess zu manipulieren (Supply-Chain-Angriff).

Die Motivation eines Innentäters bei dem Hersteller des Aktensystems können finanzieller Profit oder Schädigung des eigenen Arbeitgebers sein. Die Fähigkeiten der Innentäter schwanken stark, abhängig von der jeweiligen Person im Unternehmen.

Die Relevanz der Hersteller des Aktensystems wird als hoch eingeschätzt.

5.2.8 Betreiber des Aktensystems

Titel	Betreiber des Aktensystems	
ID	att_hm	
Kategorie	Innentäter	
Ziele	Finanzieller Profit, Schädigung des eigenen Arbeitgebers, Schädigung von Kunden	
Möglichkeiten	Zugang zu Systemen	Ja
	Insider	Ja
	Technische Expertise	Mittel
	Finanzielle Möglichkeiten	Niedrig
Relevanz	Hoch	

Die Betreiber des Aktensystems sind Mitarbeiter von Unternehmen, welche für den Betrieb der elektronischen Patientenakte verantwortlich sind. Sie erhalten vom Hersteller des Aktensystems das VAU-Image und speichern alle Daten der Patientenakte auf den IT-Systemen. Die Mitarbeiter des Betreibers haben physischen Zugriff und Remotezugriff auf die hochsensiblen IT-Systeme. Die abgespeicherten Patientendaten sind verschlüsselt, sodass kein direkter Zugriff auf die Daten durch die Mitarbeiter erfolgen kann. Allerdings könnten einzelne Innentäter versuchen, ihre Befugnisse und ihr Insiderwissen auszunutzen. Für Außentäter stellt der Betreiber des Aktensystems ebenfalls ein wichtiges Angriffsziel dar.

Ein Innentäter beim Betreiber des Aktensystems kann verschiedene Ziele verfolgen. Am wahrscheinlichsten ist der finanzielle Profit sowohl als intrinsische Motivation als auch im Rahmen einer möglichen Bestechung durch Außentäter. Weitere Motive können Schädigung des eigenen Arbeitgebers und die Schädigung von versicherten Personen sein. Die Fähigkeiten der Innentäter schwanken stark, abhängig von der jeweiligen Person im Unternehmen.

Die Relevanz der Betreiber des Aktensystems wird als hoch eingeschätzt.

5.2.9 Betreiber des sektoraler IDP

Titel	Betreiber des sektoraler IDP	
ID	att_sp	
Kategorie	Innentäter	
Ziele	Finanzieller Profit	
Möglichkeiten	Zugang zu Systemen	Ja
	Insider	Ja
	Technische Expertise	Mittel
	Finanzielle Möglichkeiten	Niedrig
Relevanz	Mittel	

Der sektorale IDP dient der Authentifizierung von externen Nutzern, z. B. Versicherten. Mitarbeiter dieses Dienstes könnten versuchen, den Authentifizierung-Prozess zu manipulieren, um so auf Daten von Versicherten zuzugreifen. Die Ziele der Innentäter sind primär finanziell getrieben. Das Wissen und die Fähigkeiten der Innentäter reduziert sich auf den sektoralen IDP. Der sektorale IDP stellt aufgrund seiner Rolle ein Angriffsziel für weitere Außentäter dar.

Die Relevanz der Betreiber des sektoralen IDP wird als mittelgroß eingeschätzt.

5.2.10 Betreiber des IDP

Titel	Betreiber des IDP	
ID	att_ip	
Kategorie	Innentäter	
Ziele	Finanzieller Profit	
Möglichkeiten	Zugang zu Systemen	Ja
	Insider	Ja
	Technische Expertise	Mittel
	Finanzielle Möglichkeiten	Niedrig
Relevanz	Mittel	

Der IDP dient der Authentifizierung von Nutzern der Telematikinfrastruktur. Mitarbeiter des IDP könnten versuchen, ihre privilegierten Rechte auszunutzen und so Identitäten von Teilnehmern der Telematikinfrastruktur zu stehlen. Die Ziele der Mitarbeiter beschränken sich auf finanziellen Profit. Die Mitarbeiter des IDP besitzen Insiderwissen, welches sich allerdings auf den IDP begrenzt. Auch der Betreiber des IDP stellt ein interessantes Angriffsziel für Außentäter dar.

Die Relevanz der Betreiber des IDP wird als mittelgroß eingeschätzt.

5.2.11 Betreiber des Signaturdienstes

Titel	Betreiber des Signaturdienst	
ID	att_sd	
Kategorie	Innentäter	
Ziele	Finanzieller Profit	
Möglichkeiten	Zugang zu Systemen	Ja
	Insider	Ja
	Technische Expertise	Mittel
	Finanzielle Möglichkeiten	Niedrig
Relevanz	Mittel	

Der Signaturdienst erstellt Signaturen für die versicherte Person, mit welcher diese wiederum Befugnisse in der elektronischen Patientenakte signieren. Mitarbeiter des Signaturdienstes könnten versuchen, die Signaturen der Versicherten zu stehlen, um so Befugnisse in den Patientenakten zu ändern. Die Ziele der Innentäter des Signaturdienstes sind primär finanzieller Profit. Die Fähigkeiten der Mitarbeiter beschränken sich allein auf technisches Wissen rund um den Signaturdienst.

Die Relevanz der Betreiber des Signaturdienstes wird als mittelgroß eingeschätzt.

5.2.12 Mitarbeiter der gematik

Titel	Mitarbeiter der gematik	
ID	att_gk	
Kategorie	Innentäter	
Ziele	Finanzieller Profit, Schädigung des eigenen Arbeitgebers	
Möglichkeiten	Zugang zu Systemen	Nein
	Insider	Ja
	Technische Expertise	Mittel
	Finanzielle Möglichkeiten	Niedrig
Relevanz	Niedrig	

Die Mitarbeiter der gematik entwickeln das Konzept für die Telematikinfrastruktur und die elektronische Patientenakte. Einzelne Mitarbeiter könnten versuchen, absichtlich Fehler und Schwachstellen in das Konzept zu implementieren, um diese später ausnutzen zu können. Ziele dieser Mitarbeiter könnten vor allem finanzieller Profit und die Schädigung des eigenen Arbeitgebers sein.

Die Relevanz der gematik wird als niedrig eingeschätzt, da die Erfolgswahrscheinlichkeit der Manipulation des Konzeptes relativ gering ist. Hierbei ist zu beachten, dass die Spezifikation öffentlich verfügbar ist, wodurch Prüfungen durch jedermann (auch Kritiker der elektronischen Patientenakte) prinzipiell möglich sind. Auch Projekte wie die vorliegende Studie könnten solche bewussten Sicherheitslücken unter Umständen identifizieren.

5.2.13 Hersteller des Primärsystems

Titel	Hersteller des Primärsystems	
ID	att_hs	
Kategorie	Innentäter	
Ziele	Finanzieller Profit, Schädigung des eigenen Arbeitgebers	
Möglichkeiten	Zugang zu Systemen	Nein
	Insider	Ja
	Technische Expertise	Mittel
	Finanzielle Möglichkeiten	Niedrig
Relevanz	Mittel	

Die Hersteller des Primarsystems sind Mitarbeiter von Unternehmen, welche Software für die Leistungserbringer entwickeln. Dazu zählen etwa KIS-Systeme (Krankenhäuser) oder PVS (Arztpraxen). Die Primärsysteme besitzen Module, um mit der elektronischen Patientenakte zu kommunizieren. Leistungserbringer können ihre Daten so direkt aus der elektronischen Patientenakte abrufen und Befunde einstellen. Ein Entwickler des Unternehmens könnte absichtliche Backdoors oder andere Schwachstellen in das Primärsystem einbauen, um Schaden an der elektronischen Patientenakte anzurichten. Die Ziele des Entwicklers können finanzieller Art sein oder die Schädigung des eigenen Arbeitgebers.

Fähigkeiten eines Entwicklers sind als mittelmäßig einzuordnen. Die Entwickler kennen zwar ihr jeweiliges System im Detail, besitzen jedoch nicht immer hinreichendes Wissen in IT-Sicherheit, um eine gut verschleierte Backdoor umzusetzen.

Die Relevanz der Hersteller des Primärsystems wird als mittelgroß eingeschätzt.

5.2.14 Versicherte Person

Titel	Versicherte Person	
ID	att_vn	
Kategorie	Nutzer des Verfahrens	
Ziele	Finanzieller Profit, negative Gesinnung	
Möglichkeiten	Zugang zu Systemen	Nein
	Insider	Nein
	Technische Expertise	Niedrig
	Finanzielle Möglichkeiten	Niedrig
Relevanz	Niedrig	

Die versicherte Person hat berechtigen Zugriff auf die eigenen Patientendaten für das Frontend. Ein Angriff auf die eigenen Daten kann daher ausgeschlossen werden. Eine versicherte Person könnte jedoch versuchen, den berechtigen Zugang zu missbrauchen, um auf Daten anderer versicherter Personen zuzugreifen. Die versicherte Person können darüber hinaus versuchen, dem Betreiber des Aktensystems bzw. der Krankenkasse Schaden zuzufügen. Motive der versicherten Person könnten dabei neben finanziellen Profiten Neugier und negative Gesinnung gegenüber einer Krankenkasse oder einer Person sein. Die Fähigkeiten der versicherten Person sind gering und beschränken sich auf den legitimen Zugriff.

Die Relevanz der versicherten Person wird als niedrig eingeschätzt.

5.2.15 Vertreter der versicherten Person

Titel	Vertreter der versicherten Person	
ID	att_vr	
Kategorie	Nutzer des Verfahrens	
Ziele	Finanzieller Profit, negative Gesinnung	
Möglichkeiten	Zugang zu Systemen	Nein
	Insider	Nein
	Technische Expertise	Niedrig
	Finanzielle Möglichkeiten	Niedrig
Relevanz	Niedrig	

Der Vertreter der versicherten Person übernimmt die gesetzliche Vertretung und hat daher dieselben Berechtigungen wie die versicherte Person selbst. Die Ziele und die Fähigkeiten sind identisch zur versicherten Person.

Die Relevanz der versicherten Person wird als niedrig eingeschätzt.

5.2.16 Leistungserbringer

Titel	Leistungserbringer	
ID	att_lr	
Kategorie	Nutzer des Verfahrens	
Ziele	Finanzieller Profit	
Möglichkeiten	Zugang zu Systemen	Ja
	Insider	Nein
	Technische Expertise	Niedrig
	Finanzielle Möglichkeiten	Mittel
Relevanz	Mittel	

Die Gruppe der Leistungserbringer besteht aus allen Parteien, die Behandlungen für den Patienten erbringen. Darunter fallen beispielsweise niedergelassene Ärzte, Krankenhäuser oder Zahnärzte. Die Leistungserbringer haben über das Primärsystem direkten Zugriff auf die elektronische Patientenakte. Sie könnten ihre Berechtigungen missbrauchen, um auf die Patientenakten zuzugreifen. Das wahrscheinlichste Motiv für Leistungserbringer sind finanzielle Vorteile. Über besondere Fähigkeiten verfügen die Leistungserbringer nicht. Da Leistungserbringer auf eine Vielzahl an Patientenakten zugreifen können, sind sie ein interessantes Angriffsziel für Außentäter.

Die Relevanz der Leistungserbringer wird als mittelgroß eingeschätzt.

5.2.17 Kostenträger

Die Kostenträger sind die Krankenkassen und meist auch Betreiber des Aktensystems. Siehe daher Abschnitt 5.2.8.

5.2.18 Ombudsstelle

Titel	Ombudsstelle	
ID	att_oe	
Kategorie	Nutzer des Verfahrens	
Ziele	Finanzieller Profit, negative Gesinnung	
Möglichkeiten	Zugang zu Systemen	Ja
	Insider	Nein
	Technische Expertise	Niedrig
	Finanzielle Möglichkeiten	Niedrig
Relevanz	Mittel	

Die Ombudsstelle ist neutraler Ansprechpartner für die versicherte Person oder den Vertreter. Die Mitarbeiter haben keinen Zugriff auf die Patientendaten. Sie können allerdings Befugnisse in die Patientenakte aufnehmen. Dadurch haben sie Zugriff auf Metadaten, anhand welcher sie Rückschlüsse zur Behandlung der versicherten Person ziehen können. Ebenfalls können sie die Patientenakte durch Befugnisausschlüsse unbrauchbar machen. Die Ziele von Mitarbeitern der Ombudsstelle sind Neugier und negative Gesinnungen gegenüber versicherten Personen. Neben dem legitimen Zugriff auf Teile der Patientenakte haben die Mitarbeiter der Ombudsstelle keine weiteren nennenswerten Kenntnisse und Fähigkeiten.

Die Relevanz der Ombudsstelle wird als mittelgroß eingeschätzt.

5.2.19 E-Rezept-Fachdienst

Titel	E-Rezept-Fachdienst	
ID	att_et	
Kategorie	Innentäter	
Ziele	Finanzieller Profit, negative Gesinnung, Neugier	
Möglichkeiten	Zugang zu Systemen	Ja
	Insider	Ja
	Technische Expertise	Mittel
	Finanzielle Möglichkeiten	Niedrig
Relevanz	Mittel	

Mitarbeiter des E-Rezept-Fachdienstes sind für die Verarbeitung von E-Rezepten verantwortlich. Durch die Verarbeitung der Daten in einer verschlüsselten Umgebung haben die Mitarbeitenden keinen direkten Zugriff auf die Daten. Sie könnten allerdings versuchen, die Verschlüsselung zu umgehen. Bei Zugriff auf die Daten könnten diese manipuliert oder ausgelesen werden. Die Ziele von Mitarbeitern des E-Rezept-Fachdienstes können finanzieller Profit, negative Gesinnung oder Neugier sein. Die Mitarbeiter verfügen über technische Expertise von den internen IT-Systemen.

Die Relevanz des E-Rezept-Fachdienstes wird als mittelgroß eingeschätzt.

5.3 Angriffsszenarien

5.3.1 Überblick

In den folgenden Abschnitten werden verschiedene Angriffsszenarien erläutert. Jedes Angriffsszenario wird in einem eigenen Angriffsbaum dargestellt. Dabei ist jeweils das Angriffsziel die Wurzel des Baums. Die inneren Knoten stellen Zwischenziele dar, welche ein Angreifer auf dem Weg zum Gesamtziel verfolgen kann. Das Gesamtziel wird also in graduell immer kleinere Teilziele heruntergebrochen. Der Prozess endet bei den Blattknoten, die als Angriffsschritte konkret genug sind, um bzgl. ihrer Realisierbarkeit eingeschätzt zu werden.

Einige Teilziele überschneiden sich zwischen Angriffsbäumen. So kann es bspw. für mehrere Ziele erforderlich sein, ein Phishing durchzuführen. Um die Lesbarkeit zu erhöhen und eine Wiederholung dieser Teilziele und ihrer jeweiligen Umsetzungsmöglichkeiten (Angriffspfade) zu vermeiden, werden diese Teilbäume gesondert in eigenen Abschnitten diskutiert. In den größeren Bäumen, welche diese Teilbäume verwenden, wird jeweils nur die Wurzel des Teilbaums dargestellt.

5.3.2 Quantifizierung des Risikos

Das Risiko eines Angriffs ist definiert als das Produkt der Wahrscheinlichkeit und des Schadens. Die Wahrscheinlichkeit lässt sich wiederum aufspalten in Eintritts- und Erfolgswahrscheinlichkeit eines Angriffs.

Die Eintrittswahrscheinlichkeit lässt sich auf Basis historischer Daten oder von Vergleichsdaten ähnlicher Systeme abschätzen. In diesem Projekt wird jedoch vereinfachend die Annahme getroffen, dass ein Angriff, für den es einen motivierten und befähigten Angreifer gibt, auch ausgeführt wird. Ein Angreifer ist motiviert, wenn er ein Angriffsziel im Kontext der ePA besitzt. Befähigt ist er, wenn es keine grundlegenden Anforderungen an den Angriff gibt (z. B. legitimer administrativer Zugang zu einem bestimmten Server), die er nicht erfüllt. In diesem Fall wird die Eintrittswahrscheinlichkeit auf 100% gesetzt.

Die Erfolgswahrscheinlichkeit wird mittels Kategorien (unmöglich, niedrig, mittel, hoch, sehr hoch) abgeschätzt. Eine präzisere Abschätzung ist weder zielführend noch mit den vorliegenden Daten realistisch zu quantifizieren. Je nach Angriffsart ist eine direkte Abschätzung der Erfolgswahrscheinlichkeit möglich, z. B. beim Erraten eines kryptografischen Schlüssels einer bestimmten Länge. Ist dies nicht möglich, wird die inverse Komplexität des Angriffs herangezogen. Je komplizierter ein Angriff durchzuführen ist, desto größer ist die Wahrscheinlichkeit, dass er scheitert, und desto geringer ist die Erfolgswahrscheinlichkeit.

5.3.3 Sicherheitsziele der gematik

Die Untersuchung der folgenden Sicherheitsziele wurden von der gematik als Teil der Leistungsbeschreibung vorgegeben.

- Dritte, ohne berechtigten Zugriff, dürfen zu keiner Zeit die Möglichkeit haben, auf Daten in einer ePA zuzugreifen.
- Ein Versicherter (oder sein Vertreter) darf nicht auf die ePA-Inhalte eines anderen Versicherten zugreifen können, für die er nicht befugt ist.
- Leistungserbringer dürfen nur auf die Akten und Aktenbestandteile zugreifen können, für die sie befugt sind und die Dauer der Befugnis bisher nicht abgelaufen ist.
- Einzelne Innentäter als Mitarbeiter des Betreibers dürfen nicht auf medizinische Daten einer Akte zugreifen können.
- Falls ein Angreifer eine serverseitige Schwachstelle im Aktensystem (ausgenommen der VAU) ausnutzen könnte, soll der Angreifer trotzdem keinen Zugriff auf klartextmedizinische Daten erhalten können.
- Die potenzielle Beeinträchtigung der Sicherheit eines Aktenkontos darf die Sicherheit eines anderen Aktenkontos nicht beeinträchtigen.

Die Sicherheitsziele lassen sich in die folgenden generischen Angriffsziele einteilen. Dabei können sich zwei Sicherheitsziele auf dasselbe Angriffsziel beziehen, das lediglich von unterschiedlichen Angreifern angestrebt wird. Die Ziele orientieren sich an den grundlegenden Zielen der Vertraulichkeit, Integrität und Verfügbarkeit informationstechnischer Systeme.

- Unbefugtes Lesen der Akte
- Unbefugtes Manipulieren der Akte
- Unbefugtes Löschen der Akte
- Denial-of-Service-Angriff auf Akte
- Aufdecken von Behandlungen und Krankheiten

Das letztgenannte Angriffsziel stellt eine Spezialisierung des Eingriffs in die Vertraulichkeit der Akte dar. So kann bspw. bereits die Durchführung einer bestimmten Untersuchung oder das Vorhandensein eines Rezept Rückschlüsse auf schützenswerte medizinische Umstände einer Person zulassen, auch ohne Einblick in die sonstige Akte.

5.4 Angriffsbäume

5.4.1 Übersicht der Angriffsbäume und Erfolgswahrscheinlichkeiten

In der nachfolgenden Tabelle sind die Top-Level-Bäume und die Erfolgswahrscheinlichkeiten der jeweiligen Angreifer aufgelistet:

Baum	Erfolgswahrscheinlichkeiten
TL1: Unbefugtes Lesen der Akte	Lr=0,10; HS=0,13; Ar=0,73; IP=0,10; SD=0,10
TL2: Unbefugtes Manipulieren der Akte	Lr=0,10; HS=0,13; Ar=0,72; IP=0,10; SD=0,10
TL3: Unbefugtes Löschen der Akte	Ar=0,76; IP=0,10; SD=0,10; Oe=1,00; Hm=0,00
TL4: Denial of Service der Akte	sP=1,00; Ar=0,87; Oe=1,00; Bm=1,00
TL5: Aufdecken von Behandlungen/ Krankheiten/Versichertendaten	Lr=0,10; Oe=1,00; Kr=0,00; Hm=0,00; or=0,00

5.4.2 TL1: Unbefugtes Lesen der Akte

Abb. 5.1:
TL1: Unbefugtes Lesen der Akte

Dieses Top-Level-Ziel wird von Angreifern verfolgt, um lesenden Zugriff auf die Patientenakte zu erhalten. Die Manipulation der Akte spielt bei diesem Angriff keine Rolle.

LL1: Aufbrechen der VAU Die Beschreibung befindet sich in Abschnitt 5.4.7.

LL2: Zugriff über das Primärsystem des LE erlangen Die Beschreibung befindet sich in Abschnitt 5.4.8.

LL3: Zugriff auf das Frontend des Versicherten erlangen Die Beschreibung befindet sich in Abschnitt 5.4.9.

LL10: Berechtigungen manipulieren Die Beschreibung befindet sich in Abschnitt 5.4.16.

LL9: Manipulieren des VAU-Images Die Beschreibung befindet sich in Abschnitt 5.4.15.

LL6: Ausnutzen einer abgelaufenen Berechtigung Die Beschreibung befindet sich in Abschnitt 5.4.12.

LL12: Entschlüsseln der Daten außerhalb der VAU Die Beschreibung befindet sich in Abschnitt 5.4.18.

5.4.3 TL2: Unbefugtes Manipulieren der Akte

Abb. 5.2: TL2: Unbefugtes Manipulieren der Akte

Im Gegensatz zu TL1 wird in diesem Angriffsbaum betrachtet, wie ein Angreifer die Akte manipulieren könnte.

LL1: Aufbrechen der VAU Die Beschreibung befindet sich in Abschnitt 5.4.7.

LL2: Zugriff über das Primärsystem des LE erlangen Die Beschreibung befindet sich in Abschnitt 5.4.8.

LL3: Zugriff auf das Frontend eines Versicherten erlangen Die Beschreibung befindet sich in Abschnitt 5.4.9.

LL9: Manipulieren des VAU-Images Die Beschreibung befindet sich in Abschnitt 5.4.15.

LL10: Berechtigungen manipulieren Die Beschreibung befindet sich in Abschnitt 5.4.16.

LL11: Manipulation Daten außerhalb der VAU Die Beschreibung befindet sich in Abschnitt 5.4.17.

5.4.4 TL3: Unbefugtes Löschen der Akte

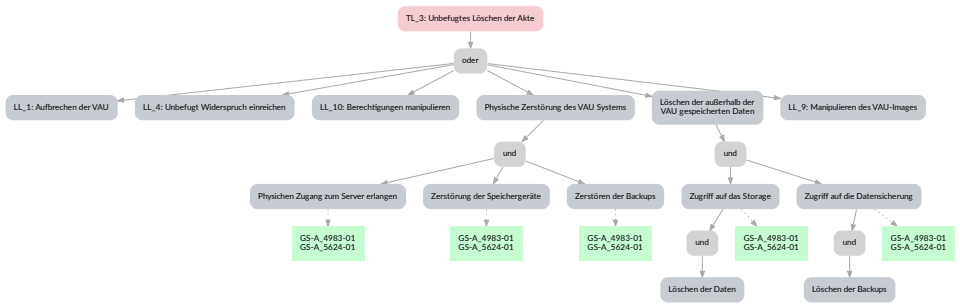

Abb. 5.3: TL3: Unbefugtes Löschen der Akte

Das Löschen der Akte könnte generell auch als eine Manipulation betrachtet werden. Jedoch ergeben sich für den Angreifer neben der inhaltlichen Löschung der Akte weitere, umfassendere Ansätze zur Vernichtung von Akten, sodass in diesem Angriffsbau auch diese Ansätze betrachtet werden.

LL1: Aufbrechen der VAU Die Beschreibung befindet sich in Abschnitt 5.4.7.

LL4: Unbefugt Widerspruch einreichen Die Beschreibung befindet sich in Abschnitt 5.4.10.

LL9: Manipulieren des VAU-Images Die Beschreibung befindet sich in Abschnitt 5.4.15.

LL10: Berechtigungen manipulieren Die Beschreibung befindet sich in Abschnitt 5.4.16.

Löschen der Backups Das Löschen von Backups ist aus Sicht des Angreifers notwendig, um eine dauerhafte Löschung der Daten zu erreichen.

Löschen der Daten Das Löschen von außerhalb der VAU gespeicherten Daten stellt den ersten Schritt zur Erreichung des Teilziels dar.

Löschen der außerhalb der VAU gespeicherten Daten In diesem Schritt werden die verschlüsselten und außerhalb der VAU gespeicherten Daten der elektronischen Patientenakte gelöscht.

Physichen Zugang zum Server erlangen Der physische Zugang zum Server ist notwendig, um die Speichergeräte physisch zu zerstören.

Physische Zerstörung des VAU Systems Die physische Zerstörung des VAU Systems setzt den physischen Zugriff zum Server voraus und erfordert die Zerstörung von Speichergerät und Backup, um eine Wiederherstellung der Daten zu verhindern.

Zerstören der Backups Das Löschen von Backups ist aus Sicht des Angreifers notwendig, um eine dauerhafte Löschung der Daten zu erreichen und eine Wiederherstellung zu verhindern.

Zerstörung der Speichergeräte Durch das Zerstören der Speichergeräte ist ein Zugriff auf die dort gespeicherten Akten nicht mehr möglich, was aus Sicht des Angriffsziels als äquivalent zu Löschung betrachtet wird.

Zugriff auf das Storage Für das Löschen von Daten wird der Zugriff zu dem Datenspeicher benötigt.

Zugriff auf die Datensicherung Für das Löschen von Backups wird der Zugriff auf die Datensicherung benötigt.

5.4.5 TL4: Denial of Service der Akte

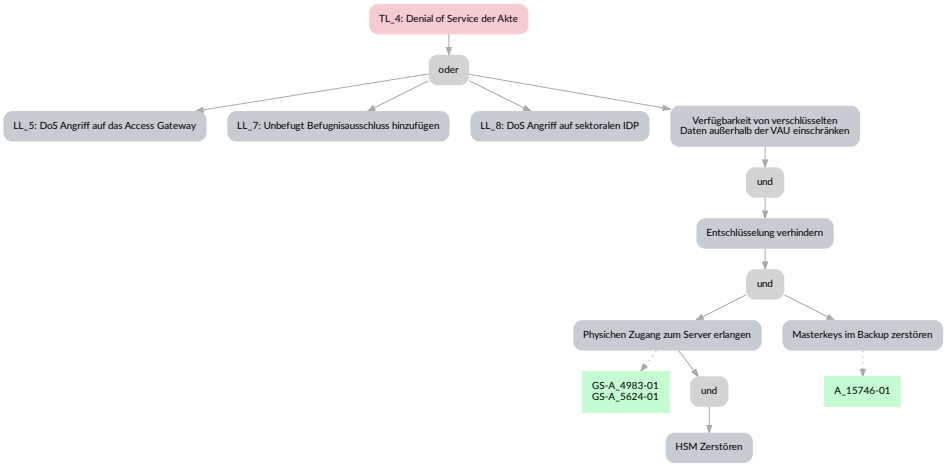

Abb. 5.4:
TL4: Denial of Service der Akte

Bei diesem Top-Level-Ziel möchte der Angreifer einen Zugriff auf die Akte verhindern oder stören.

Entschlüsselung verhindern Kann die Entschlüsselung der Daten in der VAU verhindert werden, wird auch die Nutzung der Akte verhindert.

LL5: DoS-Angriff auf das Access Gateway Die Beschreibung befindet sich in Abschnitt 5.4.11.

LL7: Unbefugt Befugnisausschluss hinzufügen Die Beschreibung befindet sich in Abschnitt 5.4.13.

LL8: DoS-Angriff auf sektoralen IDP Die Beschreibung befindet sich in Abschnitt 5.4.14.

Masterkeys im Backup zerstören Um ein Wiederherstellen der Masterkeys aus Backups zu verhindern, muss der Angreifer auch die Backups zerstören.

Physichen Zugang zum Server erlangen Der physische Zugang zum Server ist die Voraussetzung für die Zerstörung des HSM.

HSM Zerstören Durch das Zerstören des HSM ist kein Zugriff mehr auf die dort gespeicherten Schlüssel möglich, sodass eine Entschlüsselung der Daten der VAU nicht mehr möglich ist.

Verfügbarkeit von verschlüsselten Daten außerhalb der VAU einschränken Für die Entschlüsselung der Daten außerhalb der VAU werden Schlüssel benötigt, deren Verfügbarkeit angegriffen werden kann.

5.4.6 TL5: Aufdecken von Behandlungen/Krankheiten/Versichertendaten

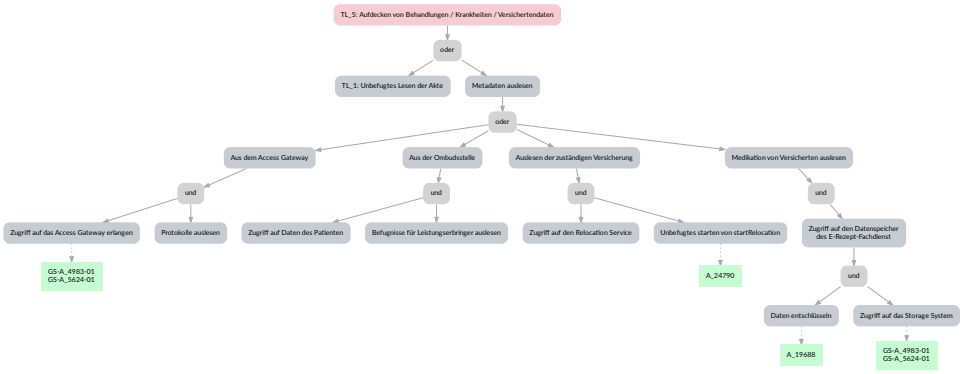

Abb. 5.5:
TL5: Aufdecken von Behandlungen/Krankheiten/Versichertendaten

Dieses Top-Level-Ziel wird vom Angreifer verfolgt, um Informationen über Behandlungen, Krankheiten oder Versichertendaten allgemein aufzudecken. Dies kann zum einen durch den unbefugten Zugriff auf die Akte erfolgen (siehe Abschnitt 5.4.7). Zusätzlich wird dies aber auch für indirekte Angriffsarten berücksichtigt, bei denen die Auswertung oder Korrelation von Metadaten verwendet wird.

TL1: Unbefugtes lesen der Akte Falls Angreifer direkten Zugriff auf die elektronische Patientenakte haben, so können sie alle Informationen aus der Akte extrahieren. Die Beschreibung dazu befindet sich in Abschnitt 5.4.2.

Metadaten auslesen Angreifer können auch an Informationen über Metadaten gelangen, welche bei der Verarbeitung von Daten entstehen können. Durch Korrelation mehrerer Daten können sensible Informationen abgeleitet werden.

Aus dem Access Gateway Jeder Versicherte greift über das Access Gateway auf die elektronische Patientenakte zu. Gateways protokollieren üblicherweise alle eingehenden Verbindungen. Angreifer könnten daher versuchen, die Protokolle des Access Gateways auszulesen.

Zugriff auf das Access Gateway erlangen Bevor die Protokolle auf dem Access Gateway ausgelesen werden können, benötigten die Angreifer Zugriff auf das Access Gateway. In den Low-Level-Bäumen LL5 (5.4.11) und LL8 (5.4.14) wird bereits beschrieben, wie Angreifer dazu vorgehen können.

Protokolle auslesen Die Protokolle enthalten möglicherweise Verbindungsdaten, in welchen personenbezogene Daten oder IDs von Versicherten enthalten sind. Angreifern könnten nun Informationen wie Uhrzeit des Zugriffs, Häufigkeit des Zugriffs, Eigenschaften des Endgeräts des Versicherten auslesen.[3]

Aus der Ombudsstelle Die Ombudsstelle kann Befugnisse für Mandanten verwalten. Angreifer könnten diesen Weg für das Extrahieren von Metadaten nutzen.

Zugriff auf Daten des Patieten Zunächst benötigt ein Außentäter Zugriff auf die Daten der Ombudsstelle. In Abschnitt 5.4.13 werden dazu mögliche Wege beschrieben,

3 Auswertung der User-Agent-Informationen

Innentätern der Ombudsstelle stehen diese Informationen bereits über legitime Wege zur Verfügung.

Befugnisse für Leistungserbringer auslesen Angreifer könnten nun anhand der Befugnisse ablesen, welche Ärzte oder Institutionen ein Versicherter aufgesucht hat. Aus diesen Daten können wiederum Informationen zum Gesundheitszustand eines Versicherten abgelesen werden.

Auslesen der zuständigen Versicherung Angreifer könnten die zuständige Versicherung eines Versicherten auslesen. Die gewonnenen Informationen könnten sie für zukünftige Angriffe nutzen.

Unbefugtes Starten von startRelocation Über den Dienst startRelocation könnte ein Innentäter einer fremden Versicherung herausfinden, bei welcher Versicherung jemand versichert ist.

Zugriff auf den Relocation Service Bevor der Dienst startRelocation ausgeführt werden kann, benötigt ein Angreifer Zugriff auf den Dienst.

Medikation von Versicherten auslesen Angreifer könnten versuchen, die Medikation über den E-Rezept-Fachdienst auszulesen. Anhand der Medikation können Krankheiten und der allgemeine Gesundheitszustand des Versicherten abgeleitet werden.

Zugriff auf den Datenspeicher des E-Rezept-Fachdienst Dieses Zwischenziel ist für die Angreifer notwendig, um Zugriff auf die Daten und die Medikation zu bekommen.

Daten entschlüsseln Bevor die Daten ausgelesen werden können, müssen sie entschlüsselt werden. Die dazu nötigen Schritte ähneln den Schritten in Abschnitt 5.4.18.

Zugriff auf das Storage System Angreifer benötigen Zugriff auf das Storage System, in welchem die Daten abgelegt werden. Innentäter des E-Rezept-Fachdienstes können einen legitimen Zugang besitzen.

5.4.7 LL1: Aufbrechen der VAU

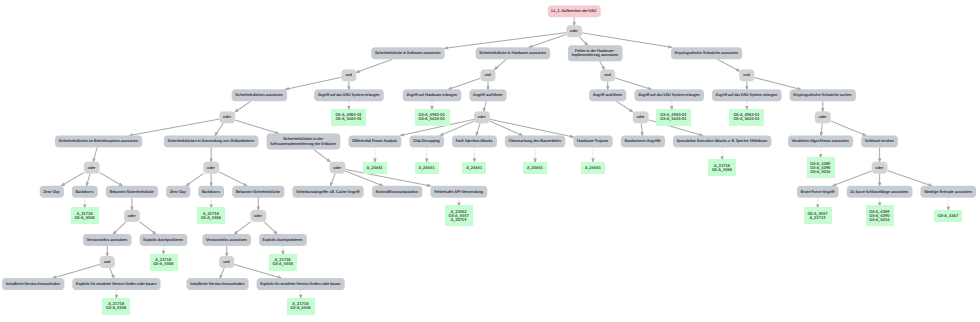

Abb. 5.6:
LL1: Aufbrechen der VAU

Dieser Angriff verfolgt das Ziel, die vertrauenswürdige Ausführungsumgebung (VAU) aufzubrechen, um Zugriff auf die Services und die unverschlüsselten Daten während der Verarbeitung zu erlangen.

Sicherheitslücke in Software ausnutzen Das Ausnutzen von Sicherheitslücken ist eine Möglichkeit, um bestehende Sicherheitsfunktionen zu umgehen.

Zugriff auf das VAU System erlangen Damit Sicherheitslücken in der Software ausgenutzt werden können, muss ein Angreifer zunächst Zugriff auf den Server haben. In diesem Fall handelt es sich um einen softwareseitigen Zugriff (z. B. Remotezugriff über SSH).

Sicherheitslücke ausnutzen Neben dem Zugriff muss die Sicherheitslücke selbst ausgenutzt werden können.

Sicherheitslücke im Betriebssystem ausnutzen Viele Sicherheitslücken befinden sich im Betriebssystem. Da das Betriebssystem die Ressourcen des Computers verwaltet, können Sicherheitslücken schnell zu einer Ausweitung der Benutzerrechte führen (Privilege Escalation).

Zero-Day Der Angreifer könnte eine Zero-Day-Sicherheitslücke nutzen, um Zugriff auf die VAU zu erlangen. Da es für Zero-Day-Sicherheitslücken keine Sicherheitsupdates gibt, sind sie besonders gefährlich. Die Zero-Day-Sicherheitslücken können sowohl im Betriebssystem als auch in Anwendungen von Drittanbietern existieren.

Backdoor Eine Backdoor-Sicherheitslücke kann ebenfalls durch einen Angreifer ausgenutzt werden. Backdoors sind dem Opfer unbekannte Funktionen, um Zugriff auf das System zu erlangen. Auch die Backdoors können sowohl im Betriebssystem als auch in einer Anwendung von Drittanbietern existieren.

Bekannte Sicherheitslücke Bekannte Sicherheitslücken, welche vom Opfer nicht behoben wurden, können bei Zugriff auf das System verhältnismäßig einfach ausgenutzt werden. Bekannte Sicherheitslücken können sich im Betriebssystem oder in der Anwendung von Drittanbietern befinden.

Versionsinfos ausnutzen Um eine bekannte Sicherheitslücke aufzuspüren, muss der Angreifer zunächst die Version der eingesetzten Software bzw. des Betriebssystems herausfinden.

Installierte Version herausfinden In der Praxis können die Softwareversionen relativ leicht ausgelesen werden. Um den Prozess möglichst effizient zu gestalten, können Angreifer Schwachstellenscanner nutzen.

Exploits für veraltete Versionen finden oder bauen Wenn der Angreifer die Softwareversion herausgefunden hat, kann er entweder einen bereits entwickelten Exploit nutzen oder selbst Exploit entwickeln.

Exploits durchprobieren Falls die Softwareversion dem Angreifer nicht bekannt ist, kann er versuchen, eine Vielzahl an Exploits durchzuprobieren.

Sicherheitslücken in Anwendung von Drittanbietern Der Angreifer könnte versuchen, nicht die primäre Software bzw. das Betriebssystem anzugreifen, sondern eine von Drittanbietern entwickelte Software, welche auf dem System installiert ist.

Sicherheitslücken in der Softwareimplementierung der Enklaven Angreifer können Sicherheitslücken in der Software, welche in der VAU ausgeführt wird, ausnutzen.

Seitenkanalangriffe z. B. Cache-Angriff Eine Möglichkeit, die Software innerhalb der VAU anzugreifen, sind Seitenkanalangriffe. Bei einem Cache-Angriff wird die Cache-Nutzung der Enklave analysiert. Dem Angreifer könnte es dabei gelingen, Rückschlüsse auf verwendete kryptografische Schlüssel zu ziehen.

Kontrollflussmanipulation Der Kontrollfluss der Software innerhalb der VAU könnte womöglich von außerhalb der VAU beeinflusst werden. Der Angreifer könnte bestimmte Funktionen innerhalb der VAU auslösen und so an Informationen gelangen.

Fehlerhafte API-Verwendung Die unsachgemäße Verwendung von APIs an der Schnittstelle zwischen VAU und dem unsicheren Bereich des IT-Systems kann zu Fehlern in der VAU führen.

Sicherheitslücke in Hardware ausnutzen Angreifer können neben Softwareangriffen auch durch Angriffe auf die Hardware an Informationen innerhalb der VAU gelangen.

Zugriff auf Hardware erlangen Damit Hardwareangriffe durchgeführt werden können, benötigt der Angreifer zunächst Zugriff auf die Hardware. Er muss also Zugang zum Rechenzentrum des Betreibers erlangen.

Angriff ausführen Hat ein Angreifer Zugang erhalten, kann er verschiedene Angriffe ausführen.

Differential Power Analysis Ein Angreifer kann die Differential Power Analysis (DPA) nutzen, um kryptografische Schlüssel zu extrahieren. Der Angreifer sammelt dazu Messungen zur Leistungsaufnahme des Servers und beobachtet das Verhalten, wenn ein kryptografischer Schlüssel verarbeitet wird. Nachdem einige Datensätze gesammelt wurden, kann durch eine Analyse auf Basis von Korrelationstechniken möglicherweise ein Schlüssel extrahiert werden.

Chip Decapping Bei dem Chip Decapping wird der Siliziumchip der CPU freigelegt. Mithilfe von Reverse Engineering könnte ein Angreifer versuchen, an Informationen zu kryptografischen Schlüsseln zu gelangen. Da der Prozessor bei diesem Angriff physisch angegriffen und analysiert wird, ist dieser Angriff während des Betriebs nur schwer umzusetzen.

Fault Injection Attacks Ein Angreifer könnte mit Fault Injection Attacks versuchen, die Hardware von außen anzugreifen. Bei Fault Injection Attacks konfrontiert der Angreifer die Hardware mit z. B. Spannungsstörungen oder Taktstörungen. Diese Störfaktoren können die Hardware beeinflussen, sodass Fehlfunktionen entstehen.

Überwachung des Busverkehrs Eine weitere Möglichkeit für Angreifer besteht in der Überwachung des Busverkehrs in Servern. Dabei werden die Datenbusse zwischen Prozessor und Speicher mitgelesen. Bei diesem Angriff können eventuell Informationen bei der Verarbeitung in der VAU mitgelesen werden.

Hardware-Trojaner Ein Angreifer könnte einen Hardware-Trojaner innerhalb der Hardware platzieren. Dieser könnte so entwickelt werden, dass alle Daten, die innerhalb der VAU verarbeitet werden, extrahiert werden.

Fehler in der Hardware-Implementierung ausnutzen Sicherheitsarchitekturen wie Intel SGX oder ARM Trustzone können Fehler in der Implementierung auf Hardware-Ebene aufweisen. Damit ein Angreifer diese Form von Angriffen ausnutzen kann, ist ein Zugriff auf die Hardware nicht erforderlich.

Rowhammer-Angriffe Bei diesem Angriff wird durch wiederholtes und schnelles Zugreifen auf eine Speicherreihe (Row) elektrischer Stress erzeugt, der dazu führen kann, dass benachbarte Speicherzellen ihren Inhalt verlieren oder verändern. Dies kann zu unvorhersehbaren und unerwünschten Änderungen des Speichers führen.

Speculative Execution Attacks z. B. Spectre Meltdown Um die Sicherheitseigenschaften der Hardware zum Umgehen kann ein Angreifer Speculative Execution wie z. B. Spectre oder Meltdown ausführen.

Kryptografische Schwäche ausnutzen Die Daten der VAU im Hauptspeicher des Servers sind verschlüsselt. Hier könnte ein Angreifer ansetzen und die Daten entschlüsseln. Die Verarbeitung der Daten innerhalb der VAU kann dann von einem Angreifer nachvollzogen werden.

Veralteten Algorithmus ausnutzen Veraltete Algorithmen sind eine Gefahr und können durch Angreifer ausgenutzt werden. Die Entschlüsselung der Daten ist dann verhältnismäßig einfach.

Schlüssel erraten Es besteht die Option, den Schlüssel für die Entschlüsselung zu erraten. Dem Angreifer stehen dazu verschiedene Möglichkeiten zur Verfügung.

Brute-Force-Angriff Mit einem Brute-Force-Angriff kann der Angreifer versuchen, den Schlüssel zu erraten. Bei dieser Form des Angriffs handelt es sich allerdings um einen sehr zeitintensiven Angriff.

Zu kurze Schlüssellänge ausnutzen Ein Angreifer kann die Daten bei zu kurzen Schlüssellängen entschlüsseln. Mögliche Verfahren sind das bereits genannte Brute-Force-Verfahren und Rainbow-Tables.

Niedrige Entropie ausnutzen Die Daten können bei zu niedriger Entropie z. B. durch vorhersehbare Zufallszahlen entschlüsselt werden.

5.4.8 LL2: Zugriff über das Primärsystem des LE erlangen

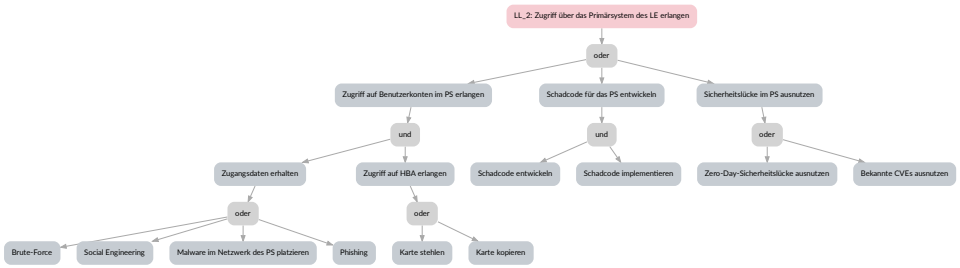

Abb. 5.7:
LL2: Zugriff über das Primärsystem des LE erlangen

Das Low-Level-Ziel Zugriff über das Primärsystem des LE zu erlangen, kann von Angreifern verfolgt werden, da die Leistungserbringer (LE) über berechtigten Zugriff auf Patientenakten verfügen. Anders als versicherte Personen oder deren Vertreter haben die Leistungserbringer Zugriff auf eine große Anzahl an Patientenakten.

Zugriff auf Benutzerkonten im PS erlangen Ein Zwischenziel für den Zugriff auf die elektronische Patientenakte über die Primärsysteme der Leistungserbringer ist der Zugriff auf ein berechtigtes Benutzerkonto. Angreifer können so etwa die Berechtigungen von einem Arzt ausnutzen. Dieses Zwischenziel ist besonders für Außentäter erstrebenswert.

Zugriff auf HBA erlangen Für einen legitimen Zugriff auf das Primärsystem über ein Benutzerkonto benötigt ein Angreifer Zugriff auf die Security Module Card Typ B (SMC-B) oder den elektronischen Heilberufsausweis (HBA).

Karte stehlen Ein Angreifer könnte die Karte des Opfers stehlen.

Karte kopieren Ergänzend dazu könnte der Angreifer versuchen, die digitalen Informationen der Karte auszulesen und diese auf eine neue Karte zu schreiben. Die Karte des Opfers würde dabei kopiert werden.

Brute Force Brute-Force-Angriffe sind eine Möglichkeit, um Zugriff auf Benutzerkonten im Primärsystem zu erlangen. Je kürzer und simpler das Passwort eines Nutzers, desto schneller führt der Angriff zum Erfolg.

Phishing Ein sehr verbreiteter und zugleich sehr erfolgreicher Angriff ist das Phishing.

Social Engineering Social-Engineering-Angriffe sind ebenfalls eine sehr effektive Methode, um Benutzerdaten zu stehlen.

Malware im Netzwerk des PS platzieren Angreifer könnten in der IT-Umgebung des Leistungserbringers Malware platzieren und so die Benutzerdaten auslesen.

Schadcode für das PS entwickeln Angreifer, in diesem Fall hauptsächlich Innentäter des Herstellers des Primärsystems, könnten Schadsoftware oder Backdoors in das Modul, welches Kommunikation mit der elektronischen Patientenakte übernimmt, implementieren.

Schadecode entwickeln Der Angreifer entwickelt den schadhaften Quellcode.

Schadecode implementieren In einem weiteren Schritt sorgt der Angreifer dafür, dass der Quellcode implementiert wird.

Sicherheitslücke im PS ausnutzen Angreifer könnten über Sicherheitslücken in dem Primärsystem Zugriff auf die Daten in der Patientenakte erhalten.

Zero-Day-Sicherheitslücke ausnutzen Angreifer könnten Zero-Day-Sicherheitslücken im Primärsystem ausnutzen.

Bekannte CVEs ausnutzen Angreifer könnten bekannte Sicherheitslücken im Primärsystem ausnutzen.

5.4.9 LL3: Zugriff auf das Frontend des Versicherten erlangen

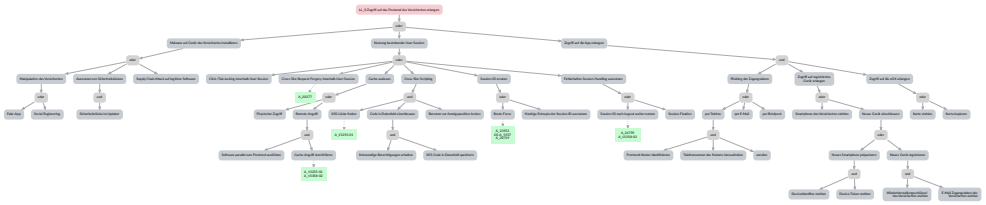

Abb. 5.8:
LL3: Zugriff auf das Frontend des Versicherten erlangen

Ziel des Angreifers in diesem Teilbaum ist es, Zugriff auf das Frontend des Versicherten zu erlangen. Dies kann über eine unautorisierte Nutzung der App erfolgen oder durch Angriffe auf Schnittstellen innerhalb des Smartphones, auf die mittels Malware zugegriffen wird.

Ausnutzen von Sicherheitslücken (Smartphone) Das Ausnutzen von Sicherheitslücken in Smartphones und Apps bezieht sich auf das Vorgehen, bei dem Angreifer Schwachstellen in der Software findet oder Schutzmaßnahmen umgeht, um unberechtigten Zugriff auf das Smartphone zu erlangen. Diese Schwachstellen können aus Fehlern im Betriebssystem, in den Apps oder in der Firmware resultieren.

Benutzer zur Anzeigeposition locken (Cross-Site-Scripting) Der Angreifer muss in der Lage sein, bösartigen Code in eine Web-Ansicht (WebView) einzuschleusen, der dann ausgeführt wird, wenn ein Benutzer die Seite besucht. Dazu muss der Angreifer den Benutzer dazu verleiten, z. B. einen manipulierten Link zu öffnen.

Brute-Force (Session-ID raten) Für einen Brute-Force-Angriff zum Raten einer Session-ID ist es notwendig, dass serverseitig die Session-IDs nicht ausreichend lang sind. Der Angreifer benötigt die Fähigkeit, automatisierte Anfragen an den Server zu senden, um systematisch alle Session-ID-Kombinationen auszuprobieren. Zudem muss der Server so konfiguriert sein, dass er mehrere fehlerhafte Anmeldeversuche toleriert, ohne Schutzmechanismen wie Account-Sperrung oder Verzögerungen bei der Anmeldung zu aktivieren.

Cache auslesen (Nutzung bestehender User Session) Im Cache von Web-Ansichten (WebViews) befinden sich in der Regel auch Cookies mit den Token der aktuellen User Session. Erlangt ein Angreifer Zugriff auf den Cache kann er die Token auslesen und für die Authentifizierung von eigenen Anfragen nutzen.

Cache-Angriff durchführen (Remote Angriff auf Session-ID) Ein Angreifer kann berechtigte Komponenten missbrauchen, um Zugriff auf den Cache zu erlangen. Durch das

Ausnutzen dieser Schwachstellen kann der Angreifer die im Cache gespeicherten sensiblen Daten wie Session-IDs extrahieren und somit die Sitzung des Benutzers übernehmen.

Click-/Tap-Jacking innerhalb User-Session Clickjacking innerhalb einer App, oft auch als „UI Redressing" bezeichnet, erfolgt, wenn ein Angreifer eine manipulierte Benutzeroberfläche über das tatsächliche Userinterface der angegriffenen App legt. Benutzer können dadurch getäuscht werden, auf Elemente zu tippen, die sie für einen Teil der legitimen App halten, während sie tatsächlich andere Aktionen ausführen.

Code in Datenfeld einschleusen (Cross-Site-Scripting) Um einen Cross-Site-Scripting-Angriff durchzuführen, muss der Code zunächst in Daten eingebettet werden, die von Programmcode verwendet werden, der den eingebetteten Code auf unsichere Weise verarbeitet.

Cross-Site-Scripting Cross-Site-Scripting (XSS) innerhalb einer App tritt auf, wenn Angreifer es schaffen, bösartigen Skriptcode in die von der App geladenen Inhalte einzuschleusen, typischerweise durch manipulierte Eingaben, die von der App nicht sicher verarbeitet werden. Sobald der schädliche Code ausgeführt wird, kann er dazu verwendet werden, sensible Daten des Benutzers zu stehlen, wie z. B. Cookies, Session-Token oder persönliche Informationen. Überdies kann XSS dazu benutzt werden, um im Namen des Benutzers Aktionen durchzuführen, ohne dass der Benutzer dies bemerkt.

Cross-Site-Request-Forgery innerhalb User-Session Bei Cross-Site-Request-Forgery (CSRF) bringt der Angreifer den Nutzer dazu, ungewollte Aktionen in einer Anwendung auszuführen, während der Benutzer authentifiziert ist. Dies geschieht typischerweise durch das Senden von Anfragen an eine App über eine andere Website oder einen anderen Kontext, die der Benutzer besucht, wobei die Anwendung fälschlicherweise annimmt, dass die Anforderungen legitim sind. Durch die bestehende Sitzung werden die Aktionen autorisiert.

DeviceIdentifier stehlen Der DeviceIdentifier wird für die Registrierung eines Geräts benötigt. Kann der Angreifer den DeviceIdentifier eines registrierten Geräts auslesen, so kann ein Gerät unter der Kontrolle des Angreifers mit diesem DeviceIdentifier verwendet werden. Zudem wird auch das Device-Token benötigt.

Device-Token stehlen Das DeviceToken stellt den zweiten Teil der Geräteregistrierung dar. Kann es zusammen mit dem DeviceIdentifier von einem registrierten Gerät ausgelesen werden, so kann ein Gerät unter der Kontrolle des Angreifers mit diesem DeviceToken verwendet werden.

E-Mail-Zugangsdaten des Versicherten stehlen Hat ein Angreifer Zugang zu dem E-Mail-Postfach des Opfers, so kann ein neues Gerät im Namen des Opfers registriert werden, sofern ebenfalls der Wiederherstellungsschlüssel des Opfers entwendet wurde.

Fake-App Mit einer Fake-App kann ein Opfer getäuscht werden, eine nicht authentische Version der App für den Zugang zum Frontend des Versicherten zu installieren. Gibt der Nutzer seine Zugangsdaten in die Fake-App ein, kann sie damit Zugriff auf das Frontend des Versicherten erhalten und Daten Mitlesen oder Manipulieren.

Fehlerhaftes Session-Handling ausnutzen Das Ausnutzen von fehlerhaftem Session-Handling in Apps bezieht sich auf Sicherheitslücken, die entstehen, wenn eine Anwendung Sessions nicht sicher verwaltet. Angreifer können diese Schwachstellen ausnutzen, um

Session-IDs zu übernehmen oder zu erraten, was es ihnen ermöglicht, sich als ein anderer Benutzer auszugeben. Typische Sicherheitsprobleme umfassen unzureichend zufällige oder vorhersagbare Session-IDs oder fehlerhafte Prozesse bei der Session Initiierung (Session Fixation).

Frontend-Nutzer identifizieren (Phishing) Für Phishing-Angriffe muss die Zuordnung zwischen Account und realer Identität durch den Angreifer hergestellt werden.

Malware auf Gerät des Versicherten installieren Dieser Teilbaum beschreibt Möglichkeiten, wie Angreifer Malware auf Geräten des Versicherten installieren könnten. Dies umfasst die Möglichkeiten, den Versicherten selbst dazu zu bringen, die Malware zu installieren, Sicherheitslücken auf dem Endgerät des Versicherten auszunutzen oder durch Supply-Chain-Angriffe auf legitime Software des Versicherten die Malware auf das Endgerät des Versicherten zu bringen.

Manipulation des Versicherten Als Manipulation des Versicherten werden Angriffe verstanden, die den Versicherten über die Konsequenzen der selbst durchgeführten Handlungen täuschen sollen.

Neues Gerät einschleusen Gelingt es einem Angreifer ein neues Endgerät am Frontend für einen legitimen Nutzer zu registrieren, so kann über dieses neue Endgerät im Kontext des Opfers auf das Frontend zugegriffen werden. Dazu kann entweder das bestehende Endgerät geklont oder ein neues Gerät für den Nutzer über den vorgesehenen Prozess registriert werden.

Neues Gerät registrieren Für die Nutzung mehrere Geräte oder beim Wechsel des Gerätes ist ein entsprechender Prozess für legitime Nutzer vorgesehen, mit dem neue Geräte registriert werden können. Sind dem Angreifer die für diesen Prozess notwendigen Informationen bekannt, so kann auch ein Angreifer im Namen des Opfers ein neues Gerät registrieren. Dieses kann dann für den Zugriff auf das Frontend verwendet werden.

Neues Smartphone präparieren Alternativ zur Registrierung eines neuen Smartphones, kann ein Angreifer auch versuchen, das bestehende Smartphone eines Opfers zu klonen, indem gegenüber dem Frontend die genutzten IDs des Smartphones des Opfers verwendet werden.

Niedrige Entropie der Session-ID ausnutzen Ist die Entropie einer Session-ID zu niedrig, kann ein Angreifer Session-IDs vorhersagen bzw. erraten. Dadurch kann auf Sitzungen unberechtigt zugegriffen werden.

Notwendige Berechtigungen erhalten Um Daten für eine Cross-Site-Scripting zu präparieren werden auch die Berechtigungen benötigt dieses Datum zu verändern. Da es sich um Daten handeln muss, die nutzerübergreifend verwendet werden, um andere Nutzer damit angreifen zu können, muss der Angreifer die dafür notwendige Berechtigung erlangen.

Nutzung bestehender User Session Hat ein Versicherter auf seinem Endgerät eine Sitzung zum Frontend aufgebaut, so kann ein Angreifer versuchen, diese Sitzung für seinen Angriff zu missbrauchen.

Phishing der Zugangsdaten Für den Zugriff auf Zugangsdaten ist ein üblicher Angriff die Nutzung von Phishing-Kampanien, mit denen Opfer dazu gebracht werden sollen, Zugangsdaten unbeabsichtigt an Angreifer weiterzugeben.

Physischer Zugriff (Cache auslesen) Erlangt ein Angreifer physischen Zugriff auf ein Gerät eines Versicherten, kann er versuchen, den Cache auf dem Gerät auszulesen, um Zugriff auf Sitzungsinformationen zu erhalten.

Remote-Angriff (Cache auslesen) Neben dem direkten Zugriff kann der Angreifer auch versuchen, indirekt auf dem Cache Daten auszulesen.

Session Fixation Session Fixation ist ein Angriff, bei dem der Angreifer eine gültige Session-ID auf dem Gerät eines Opfers anlegt, bevor das Opfer sich authentifiziert. Nachdem das Opfer die ihm zugewiesene Session-ID verwendet und sich eingeloggt hat, kann der Angreifer diese Session-ID nutzen, um Zugriff auf das Konto des Opfers zu erhalten, da er die Session-ID bereits kennt.

Session-ID erraten Beim Erraten von Session-IDs versucht der Angreifer, die Session-ID eines anderen Benutzers durch systematisches Ausprobieren oder durch die Nutzung von Algorithmen, die Schwachstellen in der Generierung der Session-ID ausnutzen, zu erraten. Wenn es dem Angreifer gelingt, eine gültige Session-ID zu erraten, kann er Zugriff auf die Benutzersitzung erlangen und in deren Namen Aktionen durchführen.

Session-ID nach Logout weiter nutzen Ein üblicher Fehler im Session-Management besteht in einer fehlenden Invalidierung der Sitzung, wenn dies beispielsweise nur auf Clientseite umgesetzt wird, aber der Server die Session-ID weiterhin akzeptiert.

Sicherheitslücke im Updater Wird Software als Teil eines Aktualisierungsprozesses installiert, so kann ein Angreifer versuchen, statt der originären Programmpakete diese während des Downloads zu manipulieren oder den Abruf auf andere Quellen umzuleiten. Auf diese Weise kann Malware auf dem Endgerät des Opfers installiert werden, die für die weiteren Ziele des Angreifers genutzt werden kann.

Smartphone des Versicherten stehlen Durch die Bindung des Endgeräts eines Versicherten an dessen digitale Identität, kann ein Angreifer versuchen, das Smartphone zu entwenden, um darüber Zugang zum Frontend zu erhalten.

Social Engineering Social Engineering ist eine Angriffsmethode, bei der Angreifer psychologische Manipulationstechniken verwenden, um Personen dazu zu bringen, vertrauliche Informationen preiszugeben oder Sicherheitsprotokolle zu umgehen. Dabei täuschen sie oft eine vertrauenswürdige Identität vor oder erschaffen Dringlichkeitssituationen, um Opfer zu unüberlegten Handlungen wie dem Klicken auf schädliche Links oder dem Weitergeben von Passwörtern zu bewegen.

Software parallel zum Frontend ausführen Der Angreifer muss Zugriff auf das gleiche System wie das Frontend haben, um einen Cache-Angriff durchzuführen. Das kann durch Ausführung von Code innerhalb derselben Ausführungsumgebung vom Frontend ermöglicht werden.

Supply-Chain-Attacke auf legitime Software Eine Supply-Chain-Attacke auf legitime Software zur Installation von Malware auf einem Smartphone erfolgt, wenn ein

Angreifer in die Entwicklungskette einer Software eingreift, um Schadcode in eine ansonsten vertrauenswürdige Anwendung oder ein Systemupdate einzuschleusen. Sobald das kompromittierte Produkt oder Update von Nutzern installiert wird, aktiviert sich die Malware, was dem Angreifer unberechtigten Zugriff auf das Gerät oder sensible Daten ermöglicht.

Telefonnummer des Nutzers herausfinden Für gezielte Phishing-Angriffe benötigt der Angreifer die Telefonnummer des Opfers, um es anzurufen, um dann Zugangsdaten vom Opfer zu erhalten.

Wiederherstellungsschlüssel des Versicherten stehlen Der Wiederherstellungsschlüssel wird benötigt, um neue Geräte für einen Nutzer zu registrieren. Erhält ein Angreifer Kenntnis eines Wiederherstellungsschlüssels und hat er Zugriff auf das registrierte E-Mailpostfach des Opfers, so kann er ein neues Gerät für die Identität des Opfers registrieren.

XSS-Code in Datenfeld speichern Für sog. Stored-Cross-Site-Scripting ist es notwendig, dass der XSS-Code zunächst in einem Datenfeld gespeichert wird. Werden jedoch Daten aus Nutzereingaben direkt in Serverantworten eingebettet, so besteht auch die Möglichkeit von sog. Reflected-Cross-Site-Scripting.

XSS-Lücke finden Für den Angriff über Coss-Site Scripting ist es zunächst notwendig, eine Schwachstelle in der Verarbeitung von Eingabedaten zu finden. Die Schwäche kann dabei entweder in einer verwendeten Drittanbieterbibliothek zur Verarbeitung der Daten vorhanden sein, oder in eigenem fehlerhaftem Code begründet sein.

Zugriff auf die App erlangen Ein generischer Angriffspunkt ist die Nutzung der Zugriffsberechtigung innerhalb der App. Gelingt es einem Angreifer den authentifizierten Kanal zwischen App und Frontend unautorisiert zu nutzen, so kann er unbemerkt Aktionen über die App ausführen.

Zugriff auf die eGK erlangen Ein Angreifer benötigt zusätzlich Zugriff auf die elektronische Gesundheitskarte (eGK) des Versicherten.

Karte stehlen Ein Angreifer könnte die Karte des Opfers stehlen.

Karte kopieren Ergänzend dazu könnte der Angreifer versuchen, die digitalen Informationen der Karte auszulesen und diese auf eine neue Karte zu schreiben. Die Karte des Opfers würde dabei kopiert werden.

Zugriff auf registriertes Gerät erlangen Wegen der Gerätebindung benötigt ein Angreifer neben dem Zugriff auf Zugangsdaten des Versicherten ebenfalls Zugriff auf das Smartphone des Opfers.

Anrufen (Phishing) Bei einem Phishing-Anruf gibt sich der Anrufer als Vertreter einer vertrauenswürdigen Organisation oder als Behörde aus, um das Vertrauen des Opfers zu gewinnen. Der Anrufer stellt oft dringende oder bedrohliche Szenarien dar, wie z. B. ein Problem mit einem Bankkonto oder einen angeblichen Betrugsfall, und fordert das Opfer auf, vertrauliche Informationen preiszugeben, um das Problem zu "lösen".

Per Briefpost (Phishing) Phishing per Briefpost erfolgt durch das Versenden von physischen Briefen, die scheinbar von legitimen Unternehmen oder Institutionen stammen, um Empfänger zu täuschen. Diese Briefe enthalten oft gefälschte Aufforderungen zur Überprüfung oder Aktualisierung persönlicher Informationen, zur Teilnahme an einer angeblichen dringenden Aktion oder zur Zahlung von Gebühren. Die Empfänger werden meist dazu aufgefordert, eine angegebene Telefonnummer anzurufen oder ihre persönlichen Daten über eine Webseite einzugeben.

Per E-Mail (Phishing) Phishing per E-Mail ist eine gängige Betrugsmethode, bei der Angreifer gefälschte E-Mails versenden, die so aussehen, als kämen sie von einer vertrauenswürdigen Quelle. Diese E-Mails enthalten oft Links oder Anhänge, die, wenn angeklickt oder geöffnet, Malware installieren können oder die Opfer auf gefälschte Websites leiten, wo sie zur Eingabe persönlicher oder finanzieller Informationen aufgefordert werden.

Per Telefon (Phishing) Phishing per Telefonanruf, auch bekannt als Vishing (Voice Phishing), ist eine Form des Social Engineering, bei der Angreifer sich als vertrauenswürdige Personen oder Institutionen ausgeben, um vertrauliche Informationen zu erlangen. Während des Anrufs versucht der Angreifer, das Opfer zu überzeugen, sensible Daten preiszugeben.

5.4.10 LL4: Unbefugt Widerspruch einreichen

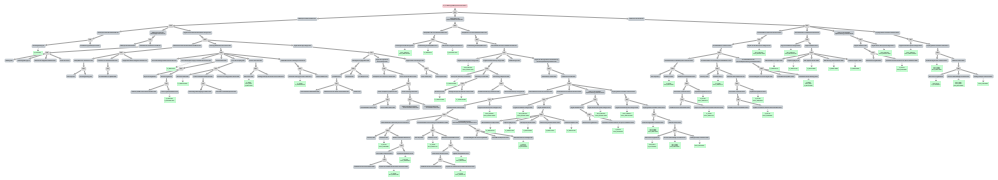

Abb. 5.9:
LL4: Unbefugt Widerspruch einreichen

Angreifer könnten versuchen, einen Widerspruch gegen die Nutzung der elektronischen Patientenakte einzureichen. Die Patientenakte wird in diesem Fall durch den Kostenträger automatisch gelöscht.

LL11: Manipulation der Daten außerhalb der VAU Angreifer könnten den Widerspruch durch die Manipulation der Daten außerhalb der VAU hinterlegen. Siehe dazu Abschnitt 5.4.17.

LL1: Aufbrechen der VAU Durch die Verarbeitung der Daten innerhalb der VAU könnten Angreifer einen Widerspruch einfügen. Siehe dazu Abschnitt 5.4.7.

Widerspruch direkt einreichen Angreifer könnten auch versuchen, den Widerspruch über legitime Wege einzureichen. Dazu missbrauchen sie Berechtigungen oder stehlen Zugangsdaten.

LL3: Zugriff auf Frontend des Versicherten erlangen Über das Frontend des Versicherten könnten Angreifer ebenfalls einen Widerspruch unbefugter Weise hinterlegen.

Widerspruch über die Ombudsstelle Die Ombudsstelle ist ebenfalls in der Lage, einen Widerspruch für Mandanten zu hinterlegen. Dies könnten Angreifer ausnutzen.

Social Engineering Durch Social Engineering könnte ein Angreifer versuchen, über die Ombudsstelle einen Widerspruch für einen bestimmten Mandanten zu hinterlegen.

Widerspruch über den Kostenträger einreichen Jeder Kostenträger wird einen Weg etablieren müssen, um den Widerspruch für die Versicherten zu ermöglichen. Auch hier könnten Angreifer ansetzen.

Credentials des Versicherten stehlen Um den Widerspruch einreichen zu können, müssen die Angreifer zunächst die Zugangsdaten stehlen.

Brute-Force Mit Brute-Force-Angriffen können die Zugangsdaten der Versicherten erraten werden.

Malware auf Endgeräte installieren Angreifer könnten Malware auf dem Endgerät des Versicherten installieren, um die Zugangsdaten auszulesen oder aufzuzeichnen.

Phishing Phishing ist eine sehr effektive Methode, um die Zugangsdaten der Versicherten zu stehlen.

Social Engineering Social-Engineering-Angriffe könnten Versicherte zur Herausgabe der Zugangsdaten verleiten.

Widerspruch einreichen Mit den gestohlenen Zugangsdaten kann der Widerspruch eingereicht werden.

5.4.11 LL5: DoS-Angriff auf das Access Gateway

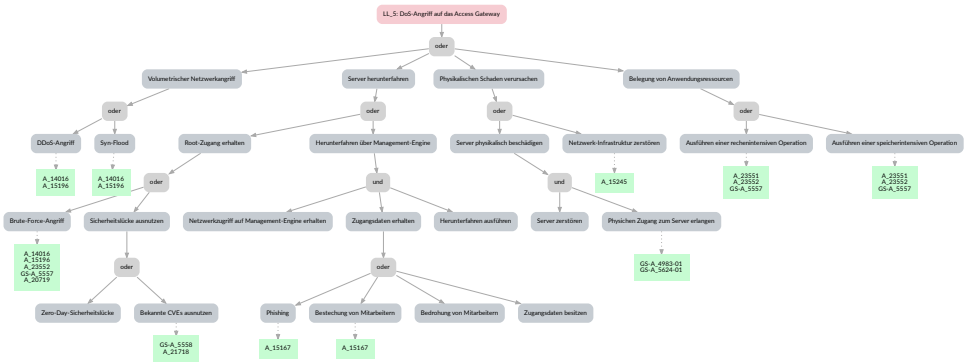

Abb. 5.10:
LL5: DoS-Angriff auf das Access Gateway

Das Angriffsziel DoS-Angriff auf das Access Gateway dient dazu, die Verfügbarkeit der elektronischen Patientenakte einzuschränken. Durch den Angriff auf das Access Gateway wird allerdings nur die Verfügbarkeit der Akte für die Versicherten eingeschränkt. Die Leistungserbringer können wie gewohnt auf Daten zugreifen.

Volumetrischer Netzwerkangriff Volumetrische Netzwerkangriffe können dazu genutzt werden, die Bandbreite und die Ressource des Access Gateways zu erschöpfen, was dann in einem Denial of Service (DoS) resultiert.

DDoS-Angriff Der bekannteste Angriff aus der Kategorie der volumetrischen Netzwerkangriffe ist der DDoS-Angriff. Bei einem DDoS-Angriff werden sehr viele Anfragen von unterschiedlichen Clients gleichzeitig zu einem Server geschickt. Dies erschöpft die Bandbreite und der Server ist nicht mehr verfügbar.

Syn-Flood Neben dem DDoS-Angriff könnten Angreifer auch einen Syn-Flood-Angriff nutzen. Bei einem Syn-Flood-Angriff initiiert der Angreifer den TCP-Handshake-Prozess, schließt ihn aber nicht ab, wodurch der Server in einem Zustand verbleibt, in dem er auf Antworten wartet und seine Ressourcen blockiert.

Server herunterfahren Die Verfügbarkeit des Access Gateways kann auch beeinträchtigt werden, indem der Server von Angreifern heruntergefahren wird.

Root-Zugang erhalten Zum Herunterfahren des Servers werden privilegierte Rechte benötigt.

Brute-Force-Angriff Die Angreifer könnten einen Brute-Force-Angriff auf die SSH-Schnittstelle durchführen, um in Besitz von privilegierten Benutzerkennungen zu bekommen.

Sicherheitslücke ausnutzen Alternativ könnten die Angreifer auch über Softwareschwachstellen auf dem Access Gateway in Besitz von privilegierten Benutzerkennungen kommen.

Zero-Day-Sicherheitslücke Die Angreifer könnten dafür eine Zero-Day-Sicherheitslücke nutzen. Besonders gefährlich sind Zero-Day-Sicherheitslücken, die eine Remote Code Execution (RCE) auf dem Access Gateway erlauben. Angreifer könnten dann nur mit einigen Befehlen auf der Kommandozeile den Server herunterfahren.

Bekannte CVEs ausnutzen Das Gleiche gilt auch für bereits bekannte Sicherheitslücken. Auch in diesem Fall könnten Angreifer über die Schwachstelle die Verfügbarkeit durch Herunterfahren des Access Gateways beeinträchtigen.

Herunterfahren über Management-Engine Die meisten Serversysteme verfügen über eine Management-Engine zur Remote-Steuerung. Auch über diesen Weg könnten Angreifer den Server herunterfahren. Damit dieses Zwischenziel erreicht werden kann, müssen weitere Bedingungen erfüllt werden.

Netzwerkzugriff auf Management-Engine erhalten Ein Angreifer benötigt Netzwerkzugriff auf die Management-Engine. Er muss sich im selben Netzwerk befinden oder Zugriff auf ein PC in diesem Netzwerk haben.

Zugangsdaten erhalten Als weiteres Zwischenziel muss der Angreifer die Zugangsdaten für die Management-Engine erhalten.

Bestechung von Mitarbeitern Der Angreifer könnte die Mitarbeiter bestechen, um an die Zugangsdaten zu gelangen.

Bedrohung von Mitarbeitern Der Angreifer könnte die Mitarbeiter bedrohen, um an die Zugangsdaten zu gelangen.

Phishing Angreifer können durch Phishing-Techniken versuchen, die Zugangsdaten zu stehlen.

Zugangsdaten besitzen Wenn der Angreifer ein Innentäter, also ein Mitarbeiter des Betreibers des Aktensystems ist, besitzt dieser womöglich bereits die Zugangsdaten.

Herunterfahren ausführen Wenn die Angreifer Netzwerkzugriff auf die Management-Engine erhalten haben und im Besitz der Zugangsdaten sind, können sie den Server herunterfahren.

Physikalischen Schaden verursachen Eine weitere Alternative zur Beeinträchtigung der Verfügbarkeit ist die physikalische Zerstörung von IT-Komponenten.

Server physikalisch beschädigen Wenn der Server im Rechenzentrum physikalisch beschädigt wird, ist die Verfügbarkeit der elektronischen Patientenakte beeinträchtigt.

Physichen Zugang zum Server erlangen Um den Server physikalisch zu beschädigen, benötigen die Angreifer zunächst Zugang zum Rechenzentrum. Innentäter können bereits ein Berechtigen Zugang zum Rechenzentrum haben.

Server zerstören Ein Angreifer kann mit entsprechendem Werkzeug den Server nachhaltig zerstören. Falls das Access Gateway aus Performance- oder Redundanz-Gründen im Cluster betrieben wird, müssen alle Cluster-Knoten zerstört werden, damit die Beeinträchtigung sichergestellt wird.

Netzwerk-Infrastruktur zerstören Angreifer könnten die Netzwerk-Infrastruktur im Rechenzentrum des Betreibers angreifen. Dazu würde allerdings ein direkter Zugang zum Rechenzentrum benötigt. Um dies zu umgehen, könnten Angreifer die WAN-Verbindungen angreifen und so das gesamte Betreiberunternehmen vom Internet ausschließen.

Belegung von Anwendungsressourcen Neben den volumetrischen Angriffen kann der Server auch durch logische Angriffe in der Verfügbarkeit angegriffen werden.

Ausführen einer rechenintensiven Operation Angreifer könnten absichtlich rechenintensive Operationen über das Frontend durchführen. Die Applikation könnte so auf logischer Ebene überfordert werden, was wiederum in einem Denial of Service resultiert. Angreifer könnten auch gezielt nach Fehlern in der Software suchen, um rechenintensive Operationen anzustoßen.

Ausführen einer speicherintensiven Operation Die Angreifer können neben rechenintensiven Operationen auch speicherintensive Operationen ausführen, um den Speicher zu füllen und den Server so zu überlasten oder durch vollgelaufene Festplatten zum Absturz zu bringen.

5.4.12 LL6: Ausnutzen einer abgelaufenen Berechtigung

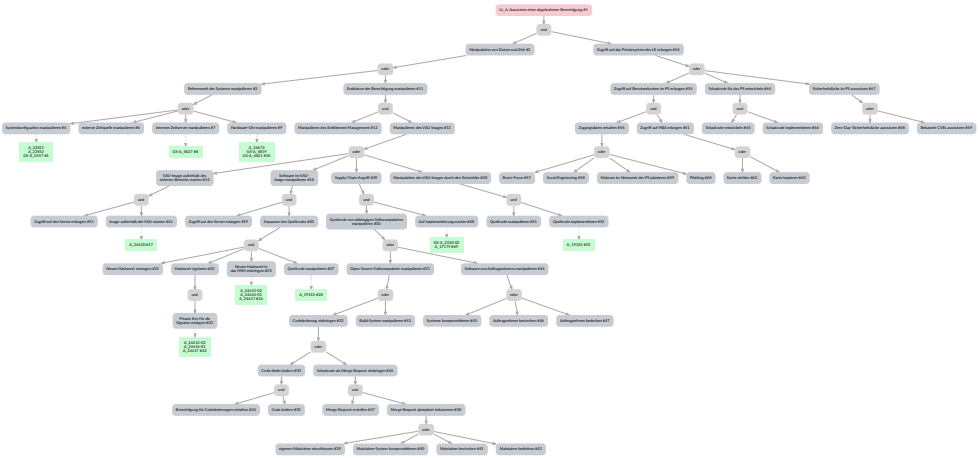

Abb. 5.11:
LL6: Ausnutzen einer abgelaufenen Berechtigung

Dieses Low-Level-Angriffsziel verfolgt den Ansatz, eine vom Leistungserbringer erstellte Berechtigung über die vorgegebenen 90 Tage hinaus in der Patientenakte zu behalten.

LL2: Zugriff über das Primärsystem des LE erlangen Zunächst benötigt der Angreifer Zugriff auf die Patientenakte über ein Primärsystem. Die Beschreibung dazu befindet sich in Abschnitt 5.4.8.

Manipulation von Datum und Zeit Zusätzlich muss ein Angreifer das Datum und die Zeit manipulieren. Dazu hat ein Angreifer zwei Optionen. Entweder manipuliert er Datum und Zeit beim Schreiben in der Akte oder er ändert nachträglich Datum und Zeit auf dem Serversystem.

Enddatum der Berechtigung manipulieren Ein Angreifer könnte über das Entitelment Management in der VAU das Enddatum der Berechtigung manipulieren.

LL9: Manipulieren des VAU-Images Zunächst muss der Angreifer in der Lage sein, das VAU Image zu manipulieren. Die Beschreibung dazu befindet sich in Abschnitt 5.4.15

Manipulieren des Entitelment Management Anschließend könnte der Angreifer den Entitelment-Management-Service manipulieren, sodass die vorgegebenen 90 Tage überschrieben werden.

Referenzzeit der Systeme manipulieren Es besteht die Möglichkeit, die Berechtigung zu verlängern, ohne dabei das VAU Image zu manipulieren. Dazu wird die Referenzzeit der Systeme angegriffen. Die Anwendungen greifen auf die Referenzzeit zurück und berechnen so, an welchem Tag die 90-Tage-Frist abgelaufen ist. Liefert die Referenzzeit falsche Werte, so stimmt die Berechnung nicht mehr.

Systemkonfiguation manipulieren Ein Angreifer könnte das System des Betreibers so manipulieren, dass ein unbekannter NTP-Server kontaktiert wird. Dieser NTP-Server befindet sich unter der Kontrolle des Angreifers und dieser kann das Datum und die Zeit beliebig ändern.

Externe Zeitquelle manipulieren Der Angreifer kann eine externe Zeitquelle (außerhalb des Netzes der TI) angreifen und Datum und Zeit verändern. Voraussetzung dafür ist, dass der Server eine Zeitquelle außerhalb des Netzwerkes der TI verwendet.

Internen Zeitserver manipulieren Verwendet der Server die NTP-Server, die für die TI vorgesehen sind, könnte ein Angreifer versuchen, diese NTP-Server anzugreifen.

Hardware-Uhr manipulieren Ein Angreifer könnte versuchen, die Hardware-Uhr des Servers zu manipulieren.

5.4.13 LL7: Unbefugt Befugnisausschluss hinzufügen

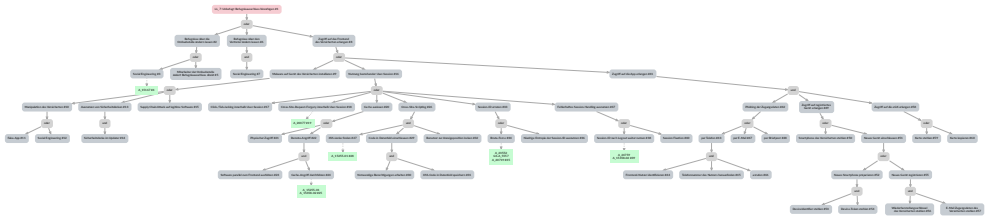

Abb. 5.12:
LL7: Unbefugt Befugnisausschluss hinzufügen

Angreifer könnten versuchen, die Patientenakte für Leistungserbringer unzugänglich zu machen, in dem sie Befugnisausschlüsse in die Akte einfügen. Leistungserbringer können dann nicht mehr auf die Patientenakte zugreifen, was einem Denial of Service der Patientenakte gleich kommt.

LL3: Zugriff auf das Frontend des Versicherten erlangen Der direkte Weg für einen Befugnisausschluss führt über das Frontend des Versicherten. Angreifer können so Befugnisausschlüsse direkt in die Akte einfügen. Die Beschreibung dazu befindet sich in Abschnitt 5.4.9

Befugnisse über die Ombudsstelle ändern lassen Die Befugnisse könnte ein Angreifer indirekt über die Ombudsstelle ändern lassen.

Social Engineering Für die Änderung über die Ombudsstelle könnten die Angreifer Social-Engineering-Taktiken verwenden.

Mitarbeiter der Ombudsstelle ändert Befugnisausschluss direkt Es besteht die Möglichkeit, dass ein Innentäter der Ombudsstelle ohne Absprache mit den Mandanten Befugnisausschlüsse in die Patientenakte einfügt.

Befugnisse über den Vertreter ändern lassen Ein Angreifer könnte die Befugnisse zusätzlich zur Ombudsstelle auch über einen gesetzlichen Vertreter ändern lassen. Der Nachteil dieser Variante ist allerdings, dass die Vertreter nicht auf viele Patientenakten gleichzeitig Zugriff haben.

Social Engineering Ein Angreifer würde auch für die Änderung der Befugnisse über den gesetzlichen Vertreter auf Social-Engineering-Taktiken setzen.

5.4.14 LL8: DoS-Angriff auf sektoralen IDP

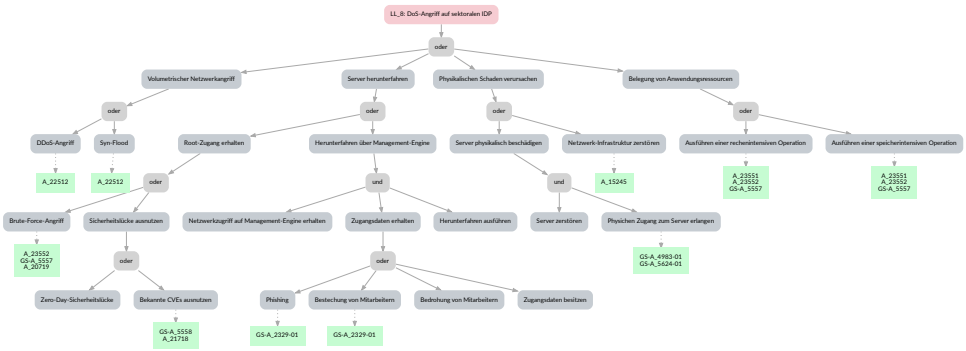

Abb. 5.13:
LL8: DoS-Angriff auf sektoralen IDP

Angreifer könnten das Ziel verfolgen, die Verfügbarkeit des sektoralen IDP einzuschränken. Die Versicherten könnten sich in diesem Fall nicht mehr authentisieren und der Zugriff auf die Patientenakte ist so nicht mehr möglich. Da der sektorale-IDP-Dienst nicht zentral bereitgestellt wird, sondern auf dem Prinzip der Föderation basiert, müssten alle sektoralen IDPs zeitgleich angegriffen werden, um die Verfügbarkeit der Patientenakte für alle Versicherten zu beeinträchtigen.

Volumetrischer Netzwerkangriff Volumetrische Netzwerkangriffe können dazu genutzt werden, die Bandbreite und die Ressource des sektoralen IDPs zu erschöpfen, was dann in einem Denial of Service (DoS) resultiert.

DDoS-Angriff Der bekannteste Angriff aus der Kategorie der volumetrischen Netzwerkangriffe ist der DDoS-Angriff. Bei einem DDoS-Angriff werden sehr viele Anfragen von unterschiedlichen Clients gleichzeitig zu einem Server geschickt. Dies erschöpft die Bandbreite und der Server ist nicht mehr verfügbar.

Syn-Flood Neben dem DDoS-Angriff könnten Angreifer auch einen Syn-Flood-Angriff nutzen. Bei einem Syn-Flood-Angriff initiiert der Angreifer den TCP-Handshake-Prozess, schließt ihn aber nicht ab, wodurch der Server in einem Zustand verbleibt, in dem er auf Antworten wartet und seine Ressourcen blockiert.

Server herunterfahren Die Verfügbarkeit des sektoralen IDPs kann auch beeinträchtigt werden, indem der Server von Angreifern heruntergefahren wird.

Root-Zugang erhalten Zum Herunterfahren des Servers werden privilegierte Rechte benötigt.

Brute-Force-Angriff Die Angreifer könnten einen Brute-Force-Angriff auf die SSH-Schnittstelle durchführen, um in Besitz von privilegierten Benutzerkennungen zu bekommen.

Sicherheitslücke ausnutzen Alternativ könnten die Angreifer auch über Softwareschwachstellen auf dem sektoralen IDP in Besitz von privilegierten Benutzerkennungen kommen.

Zero-Day-Sicherheitslücke Die Angreifer könnten dafür eine Zero-Day-Sicherheitslücke nutzen. Besonders gefährlich sind Zero-Day-Sicherheitslücken, die eine Remote Code

Execution (RCE) auf dem sektoralen IDP erlauben. Angreifer könnten dann nur mit einigen Befehlen auf der Kommandozeile den Server herunterfahren.

Bekannte CVEs ausnutzen Das Gleiche gilt auch für bereits bekannte Sicherheitslücken. Auch in diesem Fall könnten Angreifer über die Schwachstelle die Verfügbarkeit durch Herunterfahren des sektoralen IDPs beeinträchtigen.

Herunterfahren über Management-Engine Die meisten Serversysteme verfügen über eine Management-Engine zur Remote-Steuerung. Auch über diesen Weg könnten Angreifer den Server herunterfahren. Damit dieses Zwischenziel erreicht werden kann, müssen weitere Bedingungen erfüllt werden.

Netzwerkzugriff auf Management-Engine erhalten Ein Angreifer benötigt Netzwerkzugriff auf die Management-Engine. Er muss sich im selben Netzwerk befinden oder Zugriff auf ein PC in diesem Netzwerk haben.

Zugangsdaten erhalten Als weiteres Zwischenziel muss der Angreifer die Zugangsdaten für die Management-Engine erhalten.

Bestechung von Mitarbeitern Der Angreifer könnte die Mitarbeiter bestechen, um an die Zugangsdaten zu gelangen.

Bedrohung von Mitarbeitern Der Angreifer könnte die Mitarbeiter bedrohen, um an die Zugangsdaten zu gelangen.

Phishing Angreifer können durch Phishing-Techniken versuchen, die Zugangsdaten zu stehlen.

Zugangsdaten besitzen Wenn der Angreifer ein Innentäter, also ein Mitarbeiter des Betreibers des Aktensystems ist, besitzt dieser womöglich bereits die Zugangsdaten.

Herunterfahren ausführen Wenn die Angreifer Netzwerkzugriff auf die Management-Engine erhalten haben und im Besitz der Zugangsdaten sind, können sie den Server herunterfahren.

Physikalischen Schaden verursachen Eine weitere Alternative zur Beeinträchtigung der Verfügbarkeit ist die physikalische Zerstörung von IT-Komponenten.

Server physikalisch beschädigen Wenn der Server im Rechenzentrum physikalisch beschädigt wird, ist die Verfügbarkeit der elektronischen Patientenakte beeinträchtigt.

Physichen Zugang zum Server erlangen Um den Server physikalisch zu beschädigen, benötigen die Angreifer zunächst Zugang zum Rechenzentrum. Innentäter können bereits ein Berechtigen Zugang zum Rechenzentrum haben.

Server zerstören Ein Angreifer kann mit entsprechendem Werkzeug den Server nachhaltig zerstören. Falls die sektoralen IDPs aus Performance- oder Redundanz-Gründen im Cluster betrieben wird, müssen alle Cluster-Knoten zerstört werden, damit die Beeinträchtigung sichergestellt wird.

Netzwerk-Infrastruktur zerstören Angreifer könnten die Netzwerk-Infrastruktur im Rechenzentrum des Betreibers angreifen. Dazu würde allerdings ein direkter Zugang zum Rechenzentrum benötigt. Um dies zu umgehen, könnten Angreifer die WAN-Verbindungen angreifen und so das gesamte Betreiberunternehmen vom Internet ausschließen.

Belegung von Anwendungsressourcen Neben den volumetrischen Angriffen kann der Server auch durch logische Angriffe in der Verfügbarkeit angegriffen werden.

Ausführen einer rechenintensiven Operation Angreifer könnten absichtlich rechenintensive Operationen über das Frontend durchführen. Die Applikation könnte so auf logischer Ebene überfordert werden, was wiederum in einem Denial of Service resultiert. Angreifer könnten auch gezielt nach Fehlern in der Software suchen, um rechenintensive Operationen anzustoßen.

Ausführen einer speicherintensiven Operation Die Angreifer können neben rechenintensiven Operationen auch speicherintensive Operationen ausführen, um den Speicher zu füllen und den Server so zu überlasten oder durch vollgelaufene Festplatten zum Absturz zu bringen.

5.4.15 LL9: Manipulieren des VAU-Images

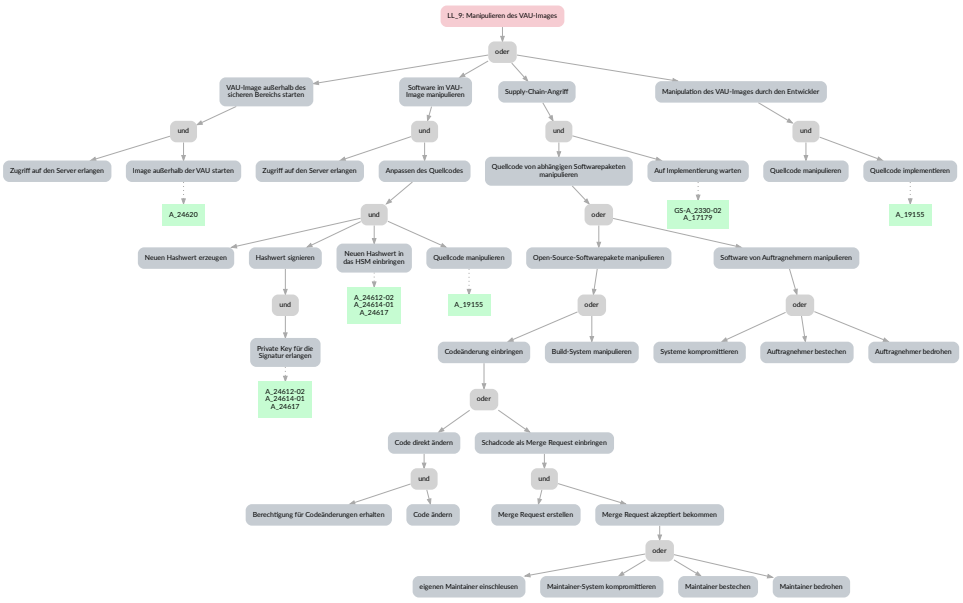

Abb. 5.14:
LL9: Manipulieren des VAU-Images

Die Aktenverwaltung der epa4all wird innerhalb einer geschützten Umgebung, der VAU, ausgeführt. Statt die Schutzmaßnahmen der VAU anzugreifen, kann ein Angreifer versuchen, die innerhalb der VAU ausgeführte Software so zu manipulieren, dass sie Daten unverschlüsselt nach außen weitergibt oder die Daten im Interesse des Angreifers verändert oder löscht. Im Extremfall kann ein Angreifer sogar versuchen, die Software in der VAU mit einer Remote Shell auszustatten, damit er auch nachträglich beliebige Befehle im Kontext der VAU ausführen kann und deren Schutzfunktion somit vollständig umgeht.

In diesem Zusammenhang besonders relevant sind Angriffe auf den Entwicklungsprozess an beliebigen Stellen innerhalb der Supply Chain. Hierdurch kann ein Angreifer versuchen, den legitimen Code zu verändern, welcher nachfolgend für die Ausführung signiert wird. Aus operativer Sicht ist der Angriff nicht erkennbar, da bereits die ursprüngliche Quelle der Software manipuliert ist.

Anpassen des Quellcodes Angreifer können versuchen, die in der VAU ausgeführte Software zu manipulieren, indem sie den Quellcode der dort laufenden Software verändern und die geänderte Softwareversion installieren. Da die Software in der VAU Zugriff auf die Daten der Patientenakte im Klartext besitzt, können auf diese Art Daten ausgelesen oder manipuliert werden.

Auf Implementierung warten Als Teil eines Supply-Chain-Angriffs müssen Angreifer nach Manipulation einer externen Bibliothek darauf warten, dass die geänderte Version in das VAU-Image übernommen wird. Dies kann z. B. im Rahmen eines Updates der im Code verwendeten Bibliotheken geschehen. Wird der Mirror, von dem benötigte Bibliotheken bezogen werden, manipuliert, kann die Implementierung auch während eines erneuten Kompiliervorgangs erfolgen, auch wenn eigentlich gar nichts an den Abhängigkeiten geändert wird.

Auftragnehmer bedrohen Ein Mitarbeiter eines Auftragnehmers, der Software entwickelt, die im ePA-Umfeld eingesetzt wird, kann bedroht werden, um die Integration von Schadcode zu erzwingen. Dieser Ansatz bringt jedoch ein Risiko der Erkennung mit sich, da der bedrohte Mitarbeiter keine intrinsische Motivation zur Kooperation mit dem Angreifer besitzt.

Auftragnehmer bestechen Ein Mitarbeiter eines Auftragnehmers, der Software entwickelt, die im ePA-Umfeld eingesetzt wird, kann bestochen werden, um die Integration von Schadcode zu erreichen.

Berechtigung für Codeänderungen erhalten Der Quellcode einer Software unterliegt - angemessene Entwicklungsprozesse vorausgesetzt - einer Versionskontrolle mit einem Prozess zur Integration von Änderungen. Mitarbeiter, die keine Projektverantwortlichen oder Maintainer sind, können im Regelfall nicht eigenmächtig beliebige Änderungen am Code durchführen. Stattdessen muss in einem angemessenen Secure Development Lifecycle ein Prozess für Änderungen eingehalten werden. Für einen Angreifer ist es relevant, die Berechtigung zum Umgehen des Änderungsprozesses und damit für direkte Codeänderungen zu erhalten.

Build-System manipulieren Statt den Quellcode der Anwendung im Versionskontrollsystem zu manipulieren, kann auch das Build-System so manipuliert werden, dass eine im Quellcode nicht sichtbare Backdoor während des Kompiliervorgangs eingefügt wird. So bleibt die Backdoor auch bei einer Untersuchung des Quellcodes unerkannt.

Code direkt ändern Verfügt der Angreifer über die notwendigen Berechtigungen, kann er den Quellcode der Anwendung, wie er für die Erzeugung der später installierten Binärdateien verwendet wird, direkt ändern. Dies kann z. B. über eine Änderung am Master-Branch des Code-Repositories erfolgen, ohne hierfür einen Merge Request zu stellen.

Code ändern Ein Angreifer kann den Code der in der VAU ausgeführten Software so ändern, dass eine Backdoor oder ein Schadcode integriert wird, z. B. zur Manipulation von Daten. Bei diesem Knoten handelt es sich um die unmittelbare Veränderung der Codedateien.

Codeänderung einbringen Hierbei handelt es sich um das Zwischenziel, den Code der in der VAU ausgeführten Software zu ändern. Dies kann entweder direkt erfolgen, oder der Angreifer kann einen Merge Request mit seiner Codeänderung stellen und diesen akzeptiert bekommen.

eigenen Maintainer einschleusen Vorgeschlagene Codeänderungen müssen, wenn ein angemessener Entwicklungsprozess verwendet wird, von einem Produktverantwortlichen freigegeben werden, bevor sie in den Master-Branch übernommen werden. Für den Angreifer ist es nützlich, eine Person seines Vertrauens einzuschleusen, die solche Berechtigungen besitzt und die Codeänderungen des Angreifers entsprechend freigeben kann.

Hashwert signieren Das VAU-Image wird nach der Erstellung signiert und kann nur ausgeführt werden, wenn die Signatur korrekt ist. Ändert ein Angreifer das Image, muss er den Hash der neuen Image-Datei berechnen und diesen mit dem korrekten Schlüssel signieren, damit seine manipulierte Software weiterhin als gültig angesehen wird.

Image außerhalb der VAU starten Ein Angreifer kann versuchen, die zur Ausführung in der VAU vorgesehene Software außerhalb der VAU zu starten. Gelingt dies, stellt die Software weiterhin die benötigte Funktionalität bereit (d. h. der Angriff ist nicht unmittelbar ersichtlich), aber die Sicherheitsfunktionen der VAU entfallen. Entsprechend sind spätere Angriffe auf die Ausführungsumgebung erheblich leichter. Dieser Knoten beschreibt den tatsächlichen Startvorgang, wenn bereits das VAU-Image vorliegt und der Angreifer über die notwendigen Berechtigungen auf dem Zielsystem verfügt.

Maintainer bedrohen Damit eine vorgeschlagene Codeänderung vom Projektverantwortlichen akzeptiert wird, kann der Angreifer versuchen, diesen zu bedrohen. Hierbei besteht jedoch das Risiko einer Erkennung, da der bedrohte Maintainer keine intrinsische Motivation zur Kooperation besitzt.

Maintainer bestechen Damit eine vorgeschlagene Codeänderung vom Projektverantwortlichen akzeptiert wird, kann der Angreifer versuchen, diesen zu bestechen.

Maintainer-System kompromittieren Vorgeschlagene Codeänderungen müssen vom Projektverantwortlichen angenommen werden. Um dies auch ohne Kooperation des Maintainers zu erreichen, kann sein System kompromittiert werden, um seine Berechtigungen unbemerkt zu missbrauchen.

Manipulation des VAU-Images durch den Entwickler Ein legitimer Entwickler, der als Teil seiner normalen Arbeitsaufgabe Änderungen am Code der in der VAU ausgeführten Software vornimmt, kann versuchen, Schadcode zusätzlich zu seinen normalen Arbeitsergebnissen einzuschleusen.

Merge Request akzeptiert bekommen Vorgeschlagene Codeänderungen müssen - einen angemessenen Entwicklungsprozess vorausgesetzt - von einem Produktverantwortlichen angenommen werden. Dieser Knoten beschreibt die Möglichkeiten, die sich einem Angreifer bieten, um eine solche Annahme zu erreichen.

Merge Request erstellen Um eine Codeänderung in ein Code-Repository einbringen zu können, auf welches der Angreifer keine unmittelbaren Schreibrechte besitzt, muss er eine sogenannten Merge Request erstellen, d. h. einen Antrag auf eine Codeänderung, der von einem Maintainer geprüft und ggf. freigegeben wird. Erst durch die Freigabe des Maintainers wird die vorgeschlagene Codeänderung tatsächlich angewendet.

Neuen Hashwert erzeugen Die in der VAU ausgeführte Software ist signiert. Dabei wird zuerst ein Hash über die Software berechnet und anschließend dieser Hashwert

signiert. Wird die Software verändert, stimmen Hashwert und Software nicht mehr überein. Folglich muss der Angreifer einen neuen Hashwert berechnen.

Neuen Hashwert in das HSM einbringen Die in der VAU ausgeführte Software ist signiert. Dabei wird zuerst ein Hash über die Software berechnet und anschließend dieser Hashwert signiert. Damit die VAU eine veränderte Software akzeptiert, muss der neue Hashwert mit zugehöriger Signatur in das HSM eingebracht werden.

Open-Source-Softwarepakete manipulieren Eine Möglichkeit, Schadcode oder Backdoors in Code einzuschleusen, besteht in der Manipulation von Drittanbieterbibliotheken, die von der Zielsoftware direkt oder indirekt verwendet werden. Der indirekte Ansatz (d. h. die Manipulation einer Bibliothek, die von einer anderen Bibliothek benutzt wird, die wiederum in der Zielsoftware eingesetzt wird) ist mitunter schwer zu entdecken.

Private Key für die Signatur erlangen Um manipulierte Software in der VAU zur Ausführung zu bringen, muss die Software (präziser: Der auf der Software berechnete Hashwert) signiert werden. Um die Signatur durchzuführen, ist der private Schlüssel des legitimen Herausgebers notwendig.

Quellcode implementieren Dieser Knoten repräsentiert die unmittelbare Manipulation des Quellcodes der in der VAU ausgeführten Software durch einen berechtigten Entwickler, der als Teil seiner normalen Arbeitsaufgabe Entwicklungstätigkeiten am Code durchführt.

Quellcode manipulieren Dieser Knoten repräsentiert die unmittelbare Manipulation des Quellcodes der in der VAU ausgeführten Software durch einen externen Angreifer.

Quellcode von abhängigen Softwarepaketen manipulieren Dieser Knoten stellt die Möglichkeiten dar, wie ein Supply-Chain-Angriff ausgeführt werden kann, indem nicht die Software der Hersteller, sondern verwendete Drittanbieterkomponenten manipuliert werden.

Schadcode als Merge Request einbringen Besitzt ein Angreifer keinen direkten Schreibzugriff auf ein Code-Repository, muss er seine Code-Änderungen als Änderungsantrag (Merge Request) einbringen. Hierfür sind verschiedene Einzelschritte erforderlich, die unterhalb dieses Knoten gesammelt werden.

Software im VAU Image manipulieren Dieser Knoten sammelt alle notwendigen Angriffsschritte, um die Software zu manipulieren, die in der VAU ausgeführt wird.

Software von Auftragnehmern manipulieren Ein Angreifer kann versuchen, die in der VAU ausgeführte Software indirekt zu manipulieren, indem er (transitiv) verwendete Komponenten verändert. In diesem Knoten werden die Schritte zur Veränderung von kundenspezifischer Software, die bei einem Zulieferer in Auftrag gegeben wird (und somit keine Open-Source-Software ist), gesammelt.

Supply-Chain-Angriff Bei einem Supply-Chain-Angriff wird nicht die Zielsoftware direkt manipuliert, indem z. B. Schadcode oder Backdoors eingeschleust werden, sondern wird eine direkt oder indirekt verwendete Komponente manipuliert. Der Angreifer muss dabei die Sicherheitsmaßnahmen des betreffenden Zulieferers überwinden statt der Maßnahmen, die vom Entwickler der Hauptsoftware ergriffen werden. Verfügt ein Zulieferer innerhalb der Lieferkette über geringere Schutzmaßnahmen als der Entwickler der Hauptsoftware, kann dieses Vorgehen einen Angriff erheblich vereinfachen.

Systeme kompromittieren Um Schadcode entlang der Lieferkette einzuschleusen, kann ein Angreifer versuchen, die Systeme eines Zulieferers zu kompromittieren und z. B. Schreibzugriff auf das Code-Repository zu erhalten. Wird der eingeschleuste Schadcode nicht entdeckt, wird er Teil des offiziellen Produkts, entlang der Lieferkette weitergegeben und schließlich in das System der Hersteller integriert. Maßnahmen wie die Prüfung von Signaturen verwendeter Komponenten zur Sicherstellung der Authentizität und der Integrität aufseiten des Verwenders sind nicht ausreichend, um solche Angriffe zu verhindern.

VAU Image außerhalb des sicheren Bereichs starten Ein Angreifer kann versuchen, die Software, die eigentlich für den Betrieb in der VAU gedacht ist, außerhalb der VAU zu starten. Damit bietet die Software weiterhin ihre Dienste gegenüber der Außenwelt an, jedoch ohne die Schutzmaßnahmen der VAU. Dieser Knoten beinhaltet alle einzelnen Aktivitäten, die erforderlich sind, um dieses Ziel zu erreichen.

Zugriff auf den Server erlangen Dieser Knoten beschreibt Tätigkeiten, die erforderlich sind, um logischen Zugriff auf den Server zu erlangen, z. B. mittels Remote-Shell.

5.4.16 LL10: Berechtigungen manipulieren

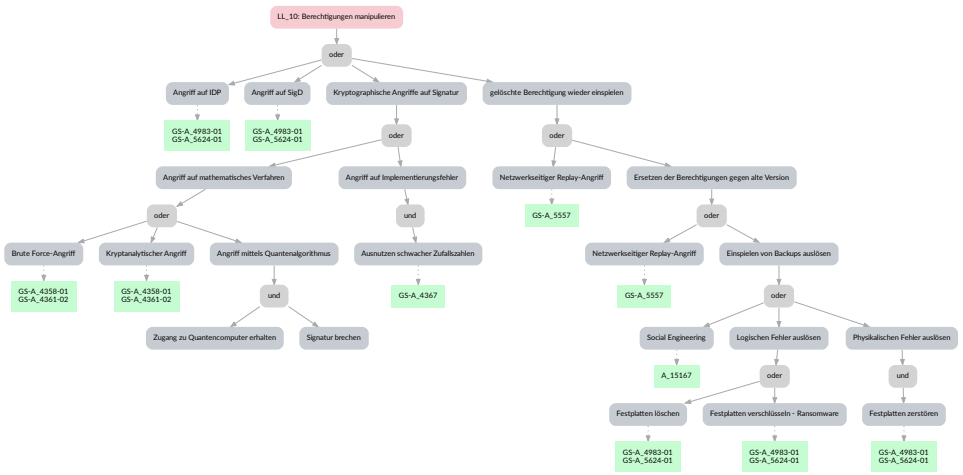

Abb. 5.15: LL10: Berechtigungen manipulieren

Jeder Benutzer in der epa4all besitzt festgelegte Berechtigungen. Ein Angreifer kann versuchen, sich erweiterte Berechtigungen zu verschaffen. Dabei wird zwischen den Berechtigungen unterschieden, die nie existiert haben, und solchen, die zwischenzeitlich entzogen wurden, und die ein Angreifer wiederherstellen möchte. Letzteres ist z. B. relevant, wenn ein Versicherter einen Vertreter aus dem System entfernt hat.

Dieser Angriffsbaum bezieht sich auf Operationen innerhalb der epa4all, nicht aber z. B. auf Berechtigungen auf Ebene des Betriebssystems auf den Servern der epa4all. Es wird nicht davon ausgegangen, dass es neben den Administratoren weitere legitime Benutzer mit Zugriff auf die Systeme gibt, die ihre Berechtigungen erweitern können, um zu Administratoren zu werden. Ein Angriff auf Ebene des Betriebssystems ohne vorbestehenden Zugang wird gesondert im Kontext des jeweiligen Angriffsziels (z. B. unbefugtes Lesen einer Akte) betrachtet.

Angriff auf IDP Dieser Knoten modelliert Angriffe auf den IDP, um die Berechtigungen eines Nutzers zu manipulieren.

Angriff auf Implementierungsfehler Ein Angreifer kann einen Implementierungsfehler in der Berechtigungsprüfung ausnutzen, um Zugriff auf Funktionen zu erhalten, die ihm normalerweise nicht zur Verfügung stehen würden. Dieser Knoten modelliert die Möglichkeiten, einen solchen Angriff durchzuführen. Die Ausführung von Schadcode wird hierbei nicht betrachtet, da diese im Kontext der Manipulation der Software in der VAU (Angriffsbaum LL_9) gesondert analysiert wird.

Angriff auf mathematisches Verfahren Die Berechtigungen eines Nutzers werden in Entitlements hinterlegt, die außerhalb der VAU gespeichert werden, und kryptografisch mittels Signatur geschützt. Verändert ein Angreifer die Berechtigungen im Entitlement, muss er eine neue, dazu passende Signatur fälschen. Dies kann bspw. durch die Ausnutzung mathematischer Schwächen in einem veralteten oder unsicheren Signaturalgorithmus geschehen.

Angriff auf SigD Dieser Knoten umfasst Angriffe auf den Signaturdienst, die verwendet werden, um sich höhere Berechtigungen bescheinigen zu lassen, als sie eigentlich bestehen.

Angriff mittels Quantenalgorithmus Die Sicherheit von Signaturverfahren beruht im Regelfall auf der Komplexität eines dem Verfahren zugrundeliegenden mathematischen Problems, z. B. der Faktorisierung eines Produkts zweier großer Primzahlen. Für einige dieser Probleme sind effiziente Quantenalgorithmen bekannt, d. h. ein hinreichend leistungsfähiger Quantencomputer kann diese Probleme in realistischer Zeit lösen, auch wenn die Schlüssellänge erhöht wird.

Ausnutzen schwacher Zufallszahlen Für viele kryptografische Verfahren sind Zufallszahlen erforderlich. Schwache, d. h. mit realistischem Aufwand vorhersagbare, Zufallszahlen können von einem Angreifer im Kontext der epa4All genutzt werden, um die Signatur zu fälschen, welche die Berechtigungen eines Nutzers (Entitlements) vor Manipulation schützt. So kann der Angreifer evtl. den für die Erstellung der Signaturen verwendeten privaten Schlüssel erraten, wenn dieser auf unsicheren Zufallszahlen beruht, und damit beliebige Signaturen erstellen.

Brute Force-Angriff Der Angreifer kann versuchen, die für kryptografische Verfahren wie bspw. die Signatur der Berechtigungen des aktuellen Benutzers verwendeten Schlüssel zu brechen, indem systematisch alle möglichen Schlüssel durchprobiert werden. Die Erfolgswahrscheinlichkeit eines solchen Angriffs hängt primär von der Schlüssellänge ab. Gelingt der Angriff, kennt der Angreifer den Schlüssel und kann beliebige Signaturen erstellen, z. B. auch für ein Entitlement mit erweiterten Berechtigungen.

Einspielen von Backups auslösen Dieser Angriff geht davon aus, dass einem Benutzer Berechtigungen entzogen wurden, z. B. als Vertreter eines Versicherten. Durch Einspielen eines Backups kann der Angreifer versuchen, das System auf einen älteren Stand zurückzusetzen, in dem er noch über die Berechtigung verfügte. Dadurch würde das Löschen rückgängig gemacht. Für diesen Angriff ist das Alter des Backups relevant. Grundsätzlich kann der Angriff jedoch nicht verhindert werden, da nicht nach jeder Datenänderung ein neues Backup angelegt wird.

Ersetzen der Berechtigungen gegen alte Version Dieser Angriff geht davon aus, dass einem Benutzer Berechtigungen entzogen wurden, z. B. als Vertreter eines Versicherten. Dieser Knoten modelliert Möglichkeiten, wie ein Angreifer versuchen kann, den Systemzustand vor der Löschung wiederherzustellen und somit die gelöschte Berechtigung zurückzuerlangen.

Festplatten löschen Um eine Notwendigkeit für das Einspielen eines Backups zu schaffen, kann ein Angreifer die Festplatten des epa4All-Systems löschen. Wenn die einzige Alternative ein Datenverlust ist, nimmt der Betreiber evtl. in Kauf, dass ältere Daten eingespielt werden.

Festplatten verschlüsseln - Ransomware Das Verschlüsseln der Festplatten führt zu einem Datenverlust, da ein Zugriff auf die Daten nicht mehr möglich ist. Hierdurch kann der Betreiber dazu gezwungen werden, ein Backup einzuspielen, um die unverschlüsselten Daten wiederherzustellen.

Festplatten zerstören Die physikalische Zerstörung von Festplatten kann den Betreiber des Aktensystems dazu zwingen, ein Backup einzuspielen, um die Daten auf den neuen Festplatten wiederherzustellen.

gelöschte Berechtigung wieder einspielen Dieser Knoten repräsentiert diverse Möglichkeiten, wie ein Angreifer gelöschte Berechtigungen wieder einspielen kann. Durch einen solchen Angriff wird der Entzug der entsprechenden Berechtigungen rückgängig gemacht.

Kryptanalytischer Angriff Dieser Knoten modelliert die tatsächliche Durchführung des kryptanalytischen Angriffs auf ein Signaturverfahren, das verwendet wird, um die Authentizität und Integrität der Entitlements, welche die Berechtigungen des aktuellen Benutzers abbilden, zu schützen.

Kryptographische Angriffe auf Signatur Dieser Knoten fasst alle kryptografischen Angriffe auf die Signatur zusammen, die verwendet wird, um die Authentizität und Integrität der Entitlements, welche die Berechtigungen des aktuellen Benutzers abbilden, zu schützen. Diese Angriffe können sowohl das Signaturverfahren selbst betreffen, d. h. den logischen Algorithmus, als auch seine Implementierung in konkreter Hardware oder Software.

Logischen Fehler auslösen Dieser Knoten fasst alle Angriffe zusammen, bei denen die Daten auf den Festplatten des epa4All-Systems auf logischer Ebene gelöscht oder unbrauchbar gemacht werden sollen. Damit unterscheidet sich der Angriff von physikalischen Angriffen, welche z. B. die Festplatten hardwareseitig zerstören.

Netzwerkseitiger Replay-Angriff Vorherige legitime Aktionen in einem System wie bspw. die Vergabe von Berechtigungen können von einem Angreifer wiederholt werden, wenn keine hinreichenden Schutzmaßnahmen existieren. Hierzu kann der Angreifer bspw. ein zuvor von einem legitimen Benutzer versendetes und vom Angreifer aufgezeichnetes Netzwerkwerkpaket erneut an den Server senden.

Physikalischen Fehler auslösen Dieser Knoten umfasst alle Aktionen, die ein Angreifer auslösen kann, um einen physikalischen Fehler auszulösen oder Schaden zu verursachen, der den Betreiber dazu zwingt, ein Backup einzuspielen.

Signatur brechen Dieser Knoten beschreibt den tatsächlichen Einsatz eines Quantencomputers (und eines entsprechenden Quantenalgorithmus), um eine gefälschte Signatur zu erzeugen. Im Falle einer RSA-Signatur kann der Quantencomputer z. B. genutzt werden, um durch Faktorisierung den privaten Schlüssel zu rekonstruieren und damit eine neue Signatur zu erzeugen.

Social Engineering Das Einspielen von Backups kann auch durch Social Engineering ausgelöst werden, indem bspw. einem zuständigen Administrator vorgespielt wird, dass es einen Systemausfall gegeben habe und die Daten im Produktivsystem nicht mehr korrekt seien. In diesem Fall kann der Administrator evtl. dazu verleitet werden, die korrekten Daten durch eine ältere Version aus einem Backup zu ersetzen.

Zugang zu Quantencomputer erhalten Um einen kryptografischen Algorithmus mittels Quantencomputer zu brechen, muss der Angreifer Zugriff zu einem Quantencomputer erhalten. Während spezialisierte Quantencomputer (z. B. Quantum Annealer) bereits kommerziell erhältlich sind, sind allgemeine Quantencomputer, die beliebige Quantenalgorithmen ausführen können, bisher nicht mit Speichergrößen (Qubits) verfügbar, die realistische Angriffe erlauben. Es wird jedoch davon ausgegangen, dass die Weiterentwicklung der Quantencomputer nur eine Frage der Zeit ist. Ob diese Quantencomputer dann für die im vorliegenden Projekt betrachteten Angreifer erhältlich oder erschwinglich sind, ist Spekulation. Es wird davon ausgegangen, dass Quantencomputer prinzipiell theoretisch verfügbar sein könnten.

5.4.17 LL11: Manipulation der Daten außerhalb der VAU

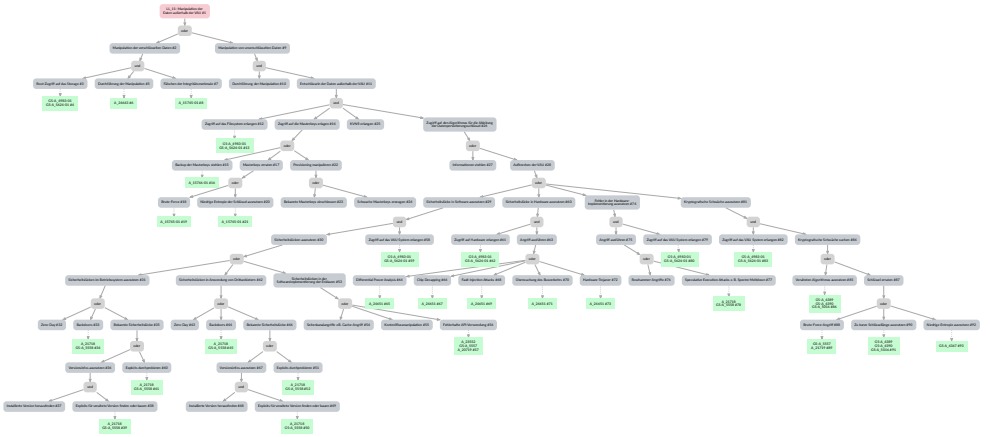

Abb. 5.16:
LL11: Manipulation der Daten außerhalb der VAU

Die Daten der Patientenakte werden außerhalb der VAU verschlüsselt abgespeichert. Ein Ziel der Angreifer kann es daher sein, die Daten außerhalb der VAU zu manipulieren, um die Integrität und die Verfügbarkeit der Daten zu verletzen.

Manipulation der verschlüsselten Daten Die Angreifer könnten versuchen, die verschlüsselten Daten zu manipulieren. Zwar können die Angreifer die Daten in der verschlüsselten Form nicht lesen, aber durch die Manipulation der verschlüsselten Daten werden diese auch für den Betreiber des Aktensystems nicht mehr lesbar. In diesem Fall ist die Verfügbarkeit der Daten das primäre Ziel der Angreifer.

Root-Zugriff auf das Storage Die Angreifer benötigen zunächst Zugriff auf das System, in welchem die Daten abgelegt werden. Mitarbeitende des Betreibers des Aktensystems können über einen legitimen Zugriff auf die Systeme verfügen.

Durchführung der Manipulation Im nächsten Schritt erfolgt die Manipulation der Daten mithilfe von Bitflips.

Fälschen der Integritätsmerkmale Ergänzend dazu könnte der Angreifer versuchen, vorhandene Integritätsmerkmale zu fälschen.

Manipulation von unverschlüsselten Daten Eine weitere Möglichkeit für die Angreifer besteht darin, die Daten zunächst zu entschlüsseln.

LL12: Entschlüsseln der Daten außerhalb der VAU Um die Daten inhaltlich zu verändern, müssten die Angreifer zunächst in der Lage sein, die Daten zu entschlüsseln. Die Beschreibung dazu befindet sich in Abschnitt 5.4.18

Durchführung der Manipulation In diesem Schritt erfolgt die eigentliche Manipulation der entschlüsselten Daten.

5.4.18 LL12: Entschlüsseln der Daten außerhalb der VAU

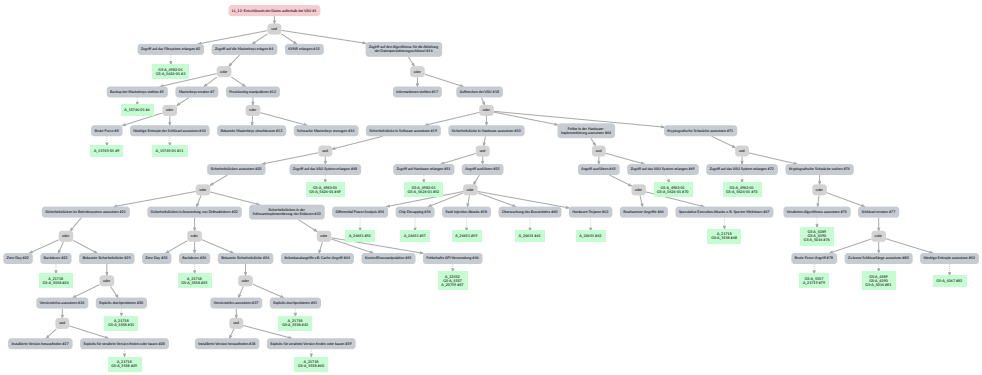

Abb. 5.17:
LL12: Entschlüsseln der Daten außerhalb der VAU

Die Daten werden außerhalb der VAU verschlüsselt abgespeichert. Ein weiteres Ziel für die Angreifer ist daher, die Daten zu entschlüsseln. Zur Entschlüsselung benötigen die Angreifer Zugriff auf das Filesystem, Zugriff auf die Masterkeys, sie benötigen die Krankenversichertennummer (KVNR) der Person, dessen Daten entschlüsselt werden sollen und zuletzt benötigen die Angreifer Informationen über den Algorithmus, wie aus den Masterkeys und der KVNR die Datenpersistierungsschlüssel abgeleitet werden. Mit den Datenpersistierungsschlüsselen können die Daten entschlüsselt werden.

Zugriff auf das Filesystem erlangen Bevor die Verschlüsselung der Daten durch die Angreifer vorgenommen werden kann, benötigen die Angreifer Zugriff auf das Dateisystem mit den verschlüsselten Daten. Mitarbeiter des Betreibers des Aktensystems verfügen möglicherweise über diese Berechtigungen und kommen daher als Innentäter infrage.

Zugriff auf die Masterkeys erlagen In den Hardware-Sicherheitsmodulen (HSM) werden die Masterkeys erzeugt. Für die Angreifer ist der Zugriff auf diese Masterkeys ein wichtiges Zwischenziel.

Backup der Masterkeys stehlen Die Masterkeys müssen durch Datensicherung abgesichert werden, andernfalls würden die Masterkeys bei einem Defekt des HSM verloren gehen und die verschlüsselten Daten könnten nicht mehr entschlüsselt werden. Angreifer könnten diesen Umstand nutzen und die Masterkeys aus der Datensicherung entwenden.

Masterkeys erraten Angreifer könnten versuchen, durch verschiedene Verfahren die Masterkeys zu erraten.

Brute Force Eine Methode zum Erraten der Masterkeys ist das Brute-Force-Verfahren.

Niedrige Entropie der Schlüssel ausnutzen Falls die Generierung der Masterkeys auf einer niedrigen Entropie basiert, könnten Angreifer die Masterkeys, z. B. vorhersehbare Zufallszahlen, erraten.

Provisioning manipulieren Eine weitere Option für Angreifer kann die Bereitstellung der Masterkeys sein.

Bekannte Masterkeys einschleusen Angreifern könnten versuchen, bereits bekannte Masterkeys in das HSM einzubringen.

Schwache Masterkeys erzeugen Alternativ können die Angreifer versuchen, das HSM zu manipulieren, um schwache Masterkeys zu erzeugen.

KVNR erlangen Angreifer müssen die KVNR des Opfers besitzen, um die Datenpersistierungsschlüssel ableiten zu können. Die KVNR sind grundsätzlich nicht geheim, jedoch bedeutet das nicht, dass sie den Angreifern einfach zugänglich sind.

Zugriff auf den Algorithmus für die Ableitung der Datenpersistierungsschlüssel Ergänzend zu den Masterkeys und der KVNR müssen die Angreifer den Algorithmus zur Ableitung der Datenpersistierungsschlüssel kennen.

Informationen stehlen Die Informationen für den Algorithmus können die Angreifer bei dem Betreiber des Aktensystems stehlen. Innentätern bei den Betreibern könnte der Algorithmus bereits bekannt sein.

LL1: Aufbrechen der VAU Alternative könnten Angreifer versuchen, den Algorithmus aus der VAU zu extrahieren. Dazu wird der Zugriff auf die VAU benötigt. Die Beschreibung dazu befindet sich in Abschnitt 5.4.7

6 Sicherheitslücken und Handlungsempfehlungen

Ziel des Projekts war vorrangig die Identifikation möglicher Sicherheitslücken und etwaigen Verbesserungsbedarfs in der Architektur des Aktensystems im Hinblick auf IT-Sicherheit. Es wird betont, dass in diesem Projekt lediglich die Anforderungen der gematik untersucht wurden, jedoch keine konkrete Implementierung der epa4all oder ihrer Komponenten.

Im vorliegenden Kapitel werden die entsprechenden Ergebnisse präsentiert. Es sei an dieser Stelle explizit darauf verwiesen, dass die Prüfung sich auftragsgemäß auf die dokumentierte Architektur sowie die Anforderungen an die Umsetzung der einzelnen Komponenten beschränkte. Konkrete Implementierungen wurden nicht geprüft. Somit bezieht sich der Begriff "Schwachstelle" auf eine mögliche Schwachstelle, die in einer Implementierung entstehen kann, wenn eine der folgenden Bedingungen zutrifft.

1. Eine ungeeignete Anforderung wird umgesetzt
2. Eine eigentlich notwendige, aber in der Spezifikation der epa4all fehlende Anforderung, wird nicht umgesetzt.

Unklar oder widersprüchlich formulierte Anforderungen oder Schwachstellen, bei denen davon auszugehen ist, dass sie lediglich Mängel in der Dokumentation darstellen, werden gesondert in Kapitel 7 beleuchtet.

Es wird empfohlen, eine Prüfung der konkreten Implementierung z. B. als Penetrationstest oder Code Review durchzuführen, sobald entsprechende Umsetzungen der Anforderungen in Software oder Hardware bereitstehen. Durch die Modellierung der Angriffsbäume auf Basis der Konzeptprüfung werden mögliche Pfade sichtbar, die Angreifer für sich nutzen könnten, die im Rahmen eines Tests der Implementierung besonders betrachtet werden müssen.

Die Bewertung der Schwachstellen erfolgt anhand der folgenden Einstufung: hoch, mittel und niedrig. Für jede Schwachstelle wird begründet, warum die jeweilige Einstufung gewählt wurde. Eine Bewertung nach dem Common Vulnerability Scoring System (CVSS) ist nicht möglich, da es sich ausschließlich um konzeptionelle Schwachstellen handelt.

6.1 Übersicht der Schwachstellen

Die folgende Tabelle stellt eine Übersicht der gefundenen Schwachstellen dar. In den anschließenden Kapiteln wird jede Schwachstelle im Detail erläutert.

Bezeichner	Verfahren	Referenz	Bewertung	Abschnitt
Bewertung von Schwachstellen	Angriffsbaum	gemSpec_DS_Anbieter	Hoch	6.2
Backup der Masterkeys	Angriffsbaum	gemSpec_DS_Anbieter	Hoch	6.3
Angriff auf die Verfügbarkeit durch Innentäter	Angriffsbaum	gemSpec_DS_Anbieter	Hoch	6.4
Sichere Entwicklungsprozesse	Angriffsbaum	gemSpec_DS_Hersteller	Hoch	6.5
IDS/IPS für sektorale IDP	Angriffsbaum	gemSpec_IDP_Sek	Mittel	6.6
Verfahren zum Einreichen von Widersprüchen	Angriffsbaum	gemSpec_Aktensystem _ePAfueralle	Mittel	6.7
Sicherheitsanforderungen für Primärsysteme	Angriffsbaum	Zuordnung nicht möglich	Mittel	6.8
Offline-Datensicherung	Angriffsbaum	gemSpec_DS_Anbieter	Mittel	6.9
Unspezifizierte ECC-Schlüssellängen	Clusteranalyse	gemProdT_SigD_PTV	Mittel	6.10
Entropie der User-Session-ID	Angriffsbaum	gemSpec_Krypt	Mittel	6.11
Verpflichtende und systemübergreifende Penetrationstests	Angriffsbaum	gemSpec_DS_Anbieter	Niedrig	6.12
Auslesen von Versionsinformationen	Angriffsbaum	gemSpec_DS_Anbieter gemSpec_DS_Hersteller	Niedrig	6.13
Austrittsverfahren für Mitarbeiter von Herstellern	Angriffsbaum	gemSpec_DS_Hersteller	Niedrig	6.14
Annullierung von Zugangsdaten ausscheiden der Organisationen	Angriffsbaum	Zuordnung nicht möglich	Niedrig	6.15
Cache-Angriffe auf die VAU	Angriffsbaum	gemSpec_DS_Anbieter gemSpec_DS_Hersteller	Niedrig	6.16
Kontrollflussmanipulation in der VAU	Angriffsbaum	gemSpec_DS_Hersteller	Niedrig	6.17
Keine Maßnahmen gegen Click-/Tap-Jacking im Frontend	Angriffsbaum	gemSpec_ePA_FdV	Niedrig	6.18
Keine Maßnahmen gegen Session Fixation	Angriffsbaum	Zuordnung nicht möglich	Niedrig	6.19
Fehlende Erkennung von Jailbreak/Root im Frontend	Angriffsbaum	ZgemSpec_ePA_FdV	Niedrig	6.20
Fehlende Erkennung veralteter Geräte im Frontend	Angriffsbaum	Zuordnung nicht möglich	Niedrig	6.21
Eingeschränkte Betriebssystemhärtung	Clusteranalyse	gemAnbT_SigD_ATV	Niedrig	6.22

6.2 Bewertung von Schwachstellen

Referenz:

gemSpec_DS_Anbieter

Schwachstelle:

Anbieter von Diensten der Telematikinfrastruktur, darunter auch Anbieter der Aktensysteme, sind dazu verpflichtet, Schwachstellen in ihren IT-Systemen zu bewerten und anschließend Gegenmaßnahmen in Abhängigkeit der Kritikalität umzusetzen.[4] Laut Sicherheitsanforderung A_21716 kann die Bewertung von Schwachstellen an Wochenenden und Feiertagen bis zu 72 Stunden verzögert erfolgen. Da erst nach Bewertung der Schwachstellen die Umsetzung der Gegenmaßnahmen in Abhängigkeit ihrer Kritikalität erfolgen kann, könnte es vorkommen, dass auch eine Schwachstelle mit der CVSS-Bewertung von 9,0 oder höher erst mit einer Verzögerung von 72 Stunden behoben wird. Handelt es sich bei der Schwachstelle um eine Remote-Code-Execution (RCE) in einer aus dem Internet

4 Sicherheitsanforderung A_21716 und A_21718

erreichbaren Komponente, könnte das System nach 72 Stunden bereits kompromittiert worden sein.

Es ist zu beachten, dass diese Anforderung unabhängig davon gilt, ob eine Schwachstelle auf Basis der vorliegenden Informationen bereits aktiv von Angreifern ausgenutzt wird. Die Bewertung und Behebung der Schwachstelle kann essenziell sein, um laufende Angriffe zu stoppen oder auch nur zu erkennen, dass eine Schwachstelle kritisch genug ist, um bis zur Behebung spezielle Detektionsmechanismen (z. B. IDS-Muster) anzuwenden.

Da es sich um öffentliche Anforderungen handelt, muss davon ausgegangen werden, dass Angreifern diese Maximalzeit bekannt ist. Somit könnten Angreifer bewusst versuchen, bei der Ausnutzung von Zero-Day-Schwachstellen eine maximale Verzögerung herbeizuführen, indem sie Angriffe unter Ausnutzung dieser Schwachstellen an Tagen vor einem verlängerten Wochenende durchführen. Selbst wenn der Angriff und die ausgenutzte Schwachstelle erkannt werden, ist die Verzögerung immer noch ein signifikanter Vorteil für den Angreifer.

Bewertung:

Die Schwachstelle wird als **"hoch"** eingestuft. Die Softwareschwachstellen können sich in Komponenten befinden, die aus dem Internet erreichbar sind. Eine kritische RCE-Schwachstelle kann von Angreifern innerhalb von 72 Stunden bereits erfolgreich ausgenutzt worden sein.

Handlungsempfehlung:

Der Analysezeitraum für die Bewertung von Schwachstellen an Wochenenden und Feiertagen sollte deutlich verringert und an den Analysezeitraum für Werktage angeglichen werden. Es sollte ein Notdienst eingerichtet werden, um Schwachstellenmeldungen auch am (verlängerten) Wochenende bearbeiten zu können.

An dieser Stelle sei zudem auf die Regelungen des Cyber Resilience Act (CRA) der EU verwiesen, welcher für Produkte mit digitalen Elementen Meldefristen an die zuständigen Behörden vorschreibt. Im CRA ist die erste Meldung bereits nach 24 Stunden vorgeschrieben, selbst wenn es sich um ein "unkritisches" Produkt im Consumer-Bereich handelt. Es wird empfohlen, für kritische Systeme im Gesundheitsbereich unabhängig von einer (separat zu prüfenden) formalen Geltung des CRA mindestens die im CRA in der Breite vorgesehenen Standards anzuwenden.

6.3 Backup der Masterkeys

Referenz:

gemSpec_DS_Anbieter

Schwachstelle:

Es gibt keine Maßnahmen, die eine Rollentrennung des Personals bei den Betreibern des Aktiensystems bezüglich HSM und Masterkeys vorschreiben. Ein Mitarbeiter mit Zugang zum Rechenzentrum darf gleichzeitig für die Sicherung der Masterkeys zuständig sein. Ein Innentäter könnte so das HSM in den Servern und die Backups der Masterkeys gleichzeitig zerstören. Die Daten könnten in diesem Fall nicht mehr von einer versicherten Person entschlüsselt werden.

Bewertung:

Die Schwachstelle wird als **"hoch"** eingestuft. Die Zerstörung der Masterkeys führt zum unwiederbringlichen Verlust der Daten.

Handlungsempfehlung:

Es wird empfohlen, ein Rollenkonzept für die Trennung zwischen Mitarbeitern mit Zugriff auf das Rechenzentrum und Zugriff auf die Datensicherung der Masterkeys zu entwerfen und den Betreibern des Aktensystems vorzuschreiben.

6.4 Angriff auf die Verfügbarkeit durch Innentäter

Referenz:

gemSpec_DS_Anbieter

Schwachstelle:

Innentäter haben im Rahmen ihrer Tätigkeit Zugriff auf kritische IT-Systeme. Einzelne Innentäter können die Verfügbarkeit der elektronischen Patientenakte negativ beeinflussen, indem sie Komponenten, wie z. B. das Access Gateway oder zentrale Komponenten des sektoralen IDP im Rechenzentrum physisch angreifen oder sie remote per Management-Engine herunterfahren.

Bewertung:

Die Schwachstelle wird als **"hoch"** eingestuft. Einzelne Innentäter können die Verfügbarkeit des Aktensystems beeinträchtigen.

Handlungsempfehlung:

Es wird empfohlen, den Betreibern des Aktensystems und des sektoralen IDPs Sicherheitsanforderungen vorzuschreiben, um technisch auszuschließen, dass ein einzelner Innentäter zentrale Komponenten zerstören oder herunterfahren kann. Es sollte seitens der Betreiber sichergestellt werden, dass ein einzelner Mitarbeiter nicht ausreichend Berechtigungen besitzt, um alle Komponenten, die für den reibungslosen Betrieb benötigt werden, herunterzufahren. Auch die Zutrittsberechtigungen für die Rechenzentren sollten so vergeben werden, dass ein Mitarbeiter nicht alle Systeme gleichzeitig physisch angreifen kann.

6.5 Sichere Entwicklungsprozesse

Referenz:

gemSpec_DS_Hersteller

Schwachstelle:

Verwendet ein Zulieferer (Softwareentwickler oder Hardwarehersteller) einen unsicheren Entwicklungsprozess, der bspw. jedem Mitarbeiter erlaubt, unbemerkt beliebige Änderungen am Code durchzuführen, können Schadcode oder Backdoors eingeschleust werden. Dies kann durch nicht vertrauenswürdige Mitarbeiter beim Zulieferer, aber auch durch

Angriffe auf unzureichend geschützte Systeme des Zulieferers geschehen. Im Beispiel des Schreibzugriffs auf den Quellcode müsste nur ein einzelner Entwicklerrechner kompromittiert werden, um diesen Zugriff im Sinne des Angreifers auszunutzen. Rein auf Ebene der gematik ist es nur schwer möglich, solche Supply-Chain-Angriffe zu erkennen.

Bewertung:

Die Schwachstelle wird als **"hoch"** eingestuft. Der Entwicklungsprozess ist ein wichtiger Grundstein für die Sicherheit von Software. Supply-Chain-Angriffe haben in letzter Zeit stark zugenommen[5].

Handlungsempfehlung:

Es wird empfohlen, den Zulieferern Vorgaben zu sicheren Entwicklungsprozessen und zur Absicherung ihrer Entwicklungssysteme zu machen. In einigen Branchen wie z. B. der Automobilindustrie, ist die Einhaltung bestimmter Standards entlang der Zulieferkette bereits Standard. Diese Empfehlung betrifft sowohl den Betrieb der eigenen IT-Systeme beim Zulieferer als auch die Tätigkeiten und Prozesse zur Erstellung des geforderten Arbeitsergebnisses für die gematik.

Bei der Verwendung von Open-Source-Komponenten wird empfohlen, die entsprechenden OSS-Projekte eingehend zu prüfen. Eine Bibliothek sollte nur verwendet werden, wenn das Projekt einen definierten und angemessenen Entwicklungsprozess verwendet. Leitfragen können sein, wer Codeänderungen durchführen oder annehmen kann (Maintainer), wie oft die Maintainer wechseln, wie neue Maintainer eingesetzt werden, wie der Code neuer Beitragender geprüft wird, wie die offiziellen Builds erzeugt werden, usw. Ebenso müssen verwendete Komponenten kontinuierlich bzgl. bekannter Sicherheitslücken überwacht werden. Sobald eine Lücke bekannt wird, welche die Sicherheit der eigenen Software beeinträchtigen kann, muss die Komponente aktualisiert werden. Lässt sich dies nicht unmittelbar realisieren, müssen andere Abwehrmaßnahmen, z. B. das temporäre Deaktivieren einer Funktion, ergriffen werden. Wird eine verwendete Komponente nicht mehr weiter betreut (end-of-life, deprecated), muss sie möglichst zeitnah ausgetauscht werden.

Die Anforderungen an Entwicklungsprozesse und an den Umgang mit verwendeten Komponenten sollten mindestens den Anforderungen entsprechen, die der Cyber Resilience Act (CRA) der EU an Produkte mit digitalen Elementen richtet (einschließlich "unkritischer" Comsumer-Produkte), unabhängig von einer formalen Anwendbarkeit des CRA auf die epa4all. Der CRA sieht bspw. das Vorhalten einer Software-Stückliste (SBOM) und die Prüfung auf bekannte Sicherheitslücken vor. Ebenso sind Produkte so zu entwerfen und zu entwickeln, dass sie auf Basis einer vom Hersteller zu entwickelnden Risikoanalyse ein angemessenes Sicherheitsniveau bieten. Hieraus ergeben sich Anforderungen an den Entwicklungsprozess, z. B. bei Design-Entscheidungen, aber auch bei der Dokumentation und Begründung einer verwendeten Systemarchitektur bis hinunter zur Ebene technischer Komponenten.

Referenz auf Standards:

Diese Gefahr ist auch vom BSI im IT-Grundschutz als Teil Elementargefährdung G 0.21 ("Manipulation von Hard- oder Software") umfasst. In den Konzepten des IT-Grundschutzes ist der sichere Entwicklungsprozess als CON.8 aufgeführt. Wir verweisen insbesondere auf die Konzepte CON.8.A17, CON.8.A.18 und CON.8.A19 für Systeme mit erhöhtem Schutzbedarf, was auf das hier vorliegende System zutrifft. Darin werden Sicherheitsaudits und

5 Beispielangriff: Manipulation der xz-Komprimierungsbibliothek zwecks Angriff auf OpenSSH

Integritätsprüfungen der Entwicklungsumgebung sowie die Auswahl vertrauenswürdiger Entwicklungswerkzeuge gefordert.

Im Wortlaut beziehen sich diese Konzepte rein auf die Entwicklungsumgebung und nicht auf die verarbeiteten Daten (Quellcode, kompilierte Binärdateien). Der Schutz dieser Daten wird jedoch konzeptionell durch andere Maßnahmen im Grundschutz dargestellt, da aus Sicht der Entwicklerfirma der Schutz des Quellcodes vor Manipulation äquivalent zum allgemeinen Schutz der Unternehmensdaten vor Manipulation ist. Anders ausgedrückt wäre ein solcher Schutz gewährleistet, wenn der Hersteller den IT-Grundschutz oder ein äquivalentes ISMS umsetzt.

In anderen sicherheitskritischen Bereichen (z.B. Luft- und Raumfahrtindustrie) ist es üblich, von Zulieferen entweder den Nachweis eines ISMS (IT-Grundschutz, ISO 27001 oder vergleichbar) oder ein äquivalentes Sicherheitsniveau einzufordern. Ist ein Zulieferer nicht zertifiziert, wird oftmals eine Checkliste übergeben, in welcher der Zulieferer zu jeder Anforderung der Standards angibt, welche äquivalenten Maßnahmen er ergreift.

6.6 IDS/IPS für sektorale IDP

Referenz:

gemSpec_IDP_Sek

Schwachstelle:

Sektorale IDPs sind aus dem Internet erreichbar und können leicht Opfer von Brute-Force-Angriffen werden. Anbieter des sektoralen IDP haben keine Verpflichtung, ein Intrusion Detection System (IDS) oder Intrusion Prevention System (IPS) zu implementieren. Durch das Fehlen der Systeme werden Brute-Force-Angriffe auf das System möglicherweise nicht erkannt oder nicht automatisch verhindert.

Bewertung:

Die Schwachstelle wird als **"mittel"** eingestuft. Der sektorale IDP ist aus dem Internet erreichbar, ein Angriff kann daher als wahrscheinlich betrachtet werden. Ein erfolgreicher Brute-Force-Angriff auf die Zugangsdaten führt zu einer Kompromittierung des gesamten Systems.

Handlungsempfehlung:

Die gematik sollte eine Verpflichtung zum Schutz vor Angriffen aus dem Internet für die Betreiber von sektoralen IDPs festlegen. Dieser Schutz sollte IDS- bzw. IPS-Systeme umfassen.

6.7 Verfahren zum Einreichen von Widersprüchen

Referenz:

gemSpec_Aktensystem_ePAfueralle

Schwachstelle:

Es gibt für das Verfahren zum Einlegen von Widersprüchen oder Rücknahme von Widersprüchen keine Mindestsicherheitsanforderungen durch die gematik. Die gematik weist explizit darauf hin, dass das Verfahren nicht Bestandteil der Spezifikation ist. An dieser Stelle besteht jedoch die Gefahr, dass der Prozess, welcher jedem Kostenträger selbst überlassen wird, Lücken enthalten könnte. Diese Lücken könnten Angreifer ausnutzen, um die Patientenakten von Versicherten zu löschen, indem sie unbefugt einen Widerspruch einlegen. Eine Wiederherstellung einer solchen unrechtmäßig gelöschten Akte wäre nicht möglich.

Bewertung:

Die Schwachstelle wird als **"mittel"** eingestuft. Das Einlegen eines Widerspruchs führt zur Löschung der gesamten ePA des Versicherten. Angreifer könnten dieses Ziel durch Social Engineering erreichen. Die Hürde für den Angreifer steigt jedoch mit der Anzahl der Opfer stark an.

Handlungsempfehlung:

Die gematik sollte einen Prozess für die Kostenträger definieren, wie ein Widerspruch eingelegt werden kann. So werden Mindestsicherheitsanforderungen berücksichtigt und es wird ein einheitlicher Prozess etabliert.

6.8 Sicherheitsanforderungen für Primärsysteme

Referenz:

Zuordnung nicht möglich

Schwachstelle:

Es gibt keine verpflichtenden Sicherheitsanforderungen für die Entwicklung der Primärsysteme durch die gematik. Hersteller könnten unbeabsichtigt oder beabsichtigt Fehler in das Primärsystem einbauen, wodurch das Aktensystem gefährdet werden könnte. Ferner könnten Angreifer den Leistungserbringer (z. B. ein Krankenhaus) angreifen und ein Primärsystem missbrauchen.

Bewertung:

Die Schwachstelle wird als **"mittel"** eingestuft. Primärsysteme sind ein zentraler Akteur im Zusammenspiel mit der ePA und es gibt keine verpflichtenden Sicherheitsanforderungen. Die Softwarequalität ist unbekannt und gleichzeitig werden Leistungserbringerinstitution (z. B. Krankenhäuser) immer wieder Opfer von Cyberangriffen.

Handlungsempfehlung:

Die gematik sollte verpflichtende Anforderungen für Primärsysteme einführen, um die elektronische Patientenakte besser vor Implementierungsfehlern und Angriffen aus der Umgebung der Leistungserbringer zu schützen. Der bereits vorhandene Leitfaden "Implementierungsleitfaden Primärsysteme – Telematikinfrastruktur (TI)" sollte verpflichtend durch die Hersteller der Primärsysteme umgesetzt werden. Ergänzend sollten folgende Mindestanforderungen an das Primärsystem gestellt werden:

- Ein sicheres Verfahren zur Benutzerauthentifizierung
- Das Erzwingen von sicheren Passwörtern durch eine Passwortrichtlinie[6]
- Brute-Force-Schutz für die Benutzerauthentifizierung
- Automatisches Sperren von Benutzersessions bei Inaktivität
- Vorgaben für einen sicheren Softwareentwicklungsprozess
- Vorgaben zur Protokollierung von und ggf. zur Erkennung von Anomalien, um z. B. Missbrauch von Accounts durch entwendete Zugangsdaten erkennen und Gegenmaßnahmen ergreifen zu können
- Regelmäßige Durchführung von Penetrationstests oder Red-Teaming-Aktivitäten zur Messung des tatsächlichen Sicherheitsniveaus

6.9 Offline-Datensicherung

Referenz:

gemSpec_DS_Anbieter

Schwachstelle:

Es gibt keine Maßnahmen, die eine Offline-Datensicherung verpflichtend vorschreiben. Das IT-Grundschutz-Kompendium des BSI sieht ebenfalls keine Verpflichtung für eine Offline-Datensicherung vor. Angreifer können ohne Offline-Datensicherung einfacher an die Backupdaten gelangen und diese löschen. Falls keine Rollentrennung und damit gleichbedeutend eine Trennung von Berechtigungen bei den Betreibern des Aktensystems umgesetzt ist, kann eine Person Zugriff auf die Produktionsdaten und die Backupdaten haben. Ein Innentäter oder ein Außentäter mit den Berechtigungen eines Innentäters könnte so Produktions- und Backupdaten gleichzeitig löschen. Gleiches gilt für eine Malware, die unter den Berechtigungen eines Mitarbeiters agiert.

Bewertung:

Die Schwachstelle wird als **"mittel"** eingestuft. Ransomware-Angriffe sind eine reale Bedrohung. Im Falle eines Angriffs sind Backups die einzige Möglichkeit, alle Daten wiederherzustellen. Die sichere Aufbewahrung von Backups ist daher unerlässlich.

Handlungsempfehlung:

Die gematik sollte die Betreiber des Aktensystems verpflichten, Offline-Datensicherungen zu implementieren. Des Weiteren sollte auf eine Rollentrennung zwischen Backup-Administratoren und Administratoren der Produktivumgebung geachtet werden.

6 Entfällt, falls keine Passwörter für die Benutzerauthentifizierung verwendet werden.

6.10 Unspezifizierte ECC-Schlüssellängen

Referenz:

gemProdT_SigD_PTV

Schwachstelle:

Es fehlen konkrete Anforderungen an die Schlüssellängen für ECC-Verfahren, obwohl deren Verwendung in A_17370-01 gefordert wird. Die Schwachstelle wird als **"mittel"** eingestuft. Während die Verwendung von ECC-Verfahren grundsätzlich positiv ist, kann das Fehlen konkreter Schlüssellängenanforderungen zu Implementierungen führen, die nicht dem aktuellen Stand der Technik entsprechen. Dies könnte langfristig die Sicherheit der verschlüsselten Daten gefährden, insbesondere angesichts der schnellen Entwicklungen in der Kryptoanalyse.

Handlungsempfehlung:

Es wird empfohlen, eine spezifische Mindestschlüssellänge für ECC-Verfahren festzulegen, beispielsweise 256 Bit gemäß aktuellen BSI-Empfehlungen. Die Anforderung sollte präzise formuliert und in die bestehenden Spezifikationen integriert werden.

6.11 Entropie der User-Session-ID

Referenz:

gemSpec_Krypt

Schwachstelle:

Es bestehen keine Sicherheitsanforderungen hinsichtlich der Entropie von Session-IDs. Eine zu niedrige Entropie macht Session-IDs vorhersagbar und erhöht die Anfälligkeit gegenüber Brute-Force-Angriffen. Das Erraten der Session-IDs kann dazu führen, dass Angreifer die Session eines Versicherten übernehmen.

Bewertung:

Die Schwachstelle wird als **"mittel"** eingestuft. Angriffe auf zu kurze User-Session-IDs sind mit Brute-Force-Angriffen möglich.

Handlungsempfehlung:

Es wird empfohlen, eine Mindestanforderung für die Entropie von Session-IDs zu erlassen. Session-IDs sollten eine Mindestlänge von 128 Bit haben.

6.12 Verpflichtende und systemübergreifende Penetrationstests

Referenz:

gemSpec_DS_Anbieter

Schwachstelle:

Im Rahmen der Produktgutachten wird alle drei Jahre ein Penetrationstest von den Gutachtern durchgeführt. Ein Scope für die Penetrationstests ist dabei allerdings nicht festgelegt. Zwischenzeitliche Penetrationstests seitens der gematik sind nicht verpflichtend.

Bewertung:

Die Schwachstelle wird als **"niedrig"** eingestuft. Sicherheitslücken oder Fehlkonfigurationen der werden möglicherweise nur in einem Rhythmus von drei Jahren entdeckt.

Handlungsempfehlung:

Penetrationstests sind ein effektives Werkzeug, um Schwachstellen und Fehlkonfigurationen an IT-Systemen aufzuspüren. Der Umfang und der Scope der Penetrationstests sollten daher im Rahmen der Produktgutachten genau definiert werden. Die gematik sollte zusätzlich jährliche produktübergreifende Penetrationstests verpflichtend einführen.

6.13 Auslesen von Versionsinformationen

Referenz:

gemSpec_DS_Anbieter, gemSpec_DS_Hersteller

Schwachstelle:

Es sind keine Maßnahmen definiert, um das Auslesen von Versionsinformationen zu verhindern bzw. zu erschweren. Angreifer können Informationen zu Softwareversionen auslesen. Die daraus gewonnenen Informationen sind eine wichtige Grundlage für Angreifer, um die Angriffe auf die Systeme vorzubereiten. Dies ist insbesondere dann relevant, wenn in einer verwendeten Komponente eine Sicherheitslücke öffentlich bekannt wurde, und die entsprechende Komponente im Produktivsystem bisher nicht aktualisiert wurde.

Bewertung:

Die Schwachstelle wird als **"niedrig"** eingestuft. Das Verbergen von Versionsinformationen verhindert keine Angriffe, kann sie aber erschweren.

Handlungsempfehlung:

Die gematik sollte Maßnahmen für die Hersteller und Betreiber von Diensten der TI vorschreiben, um das Auslesen von Versionsinformationen zu verhindern bzw. zu erschweren. Folgende Maßnahmen eignen sich für diesen Zweck:

- Vermeiden von Versionsinformationen auf Webseiten (auch im HTML-Code)
- Sicherstellen, dass Debugging- und Logdateien keine Versionsinformationen enthalten
- Vermeiden von Standard-Bannern bei Servern
- Anpassung der Webserverkonfiguration (HTTP-Header)
- Einsatz von Web Application Firewalls (WAF)

6.14 Austrittsverfahren für Mitarbeiter von Herstellern

Referenz:

gemSpec_DS_Hersteller

Schwachstelle:

Ein unzureichendes Austrittsverfahren für ehemalige Mitarbeiter von Herstellern oder Entwicklern kann erhebliche Sicherheitsrisiken mit sich bringen. Bleiben Zugänge zu sensiblen Systemen und Daten auch nach dem Ausscheiden eines Mitarbeiters bestehen, können diese missbräuchlich genutzt werden. Ehemalige Mitarbeiter könnten absichtlich oder unabsichtlich Schadcode einbringen, sensible Informationen preisgeben oder weiterhin Zugang zu vertraulichen Daten haben. Solche Risiken entstehen insbesondere, wenn die Prozesse zur Deaktivierung von Benutzerkonten, dem Entzug von Zugriffsrechten und der Rückgabe von Firmenausstattung nicht konsequent umgesetzt werden. Die Umsetzung aller BSI-Grundschutz-Maßnahmen aus der Anforderung GS-A_4983-01 beinhaltet eine Maßnahme zum Austrittsverfahren, allerdings nur für Anbieter.

Bewertung:

Die Schwachstelle wird als **"niedrig"** eingestuft. Ehemalige Mitarbeiter könnten bösartigen Code implementieren. Interne Qualitätskontrollen und Testverfahren sollten in der Lage sein, manipulierten Quellcode zu erkennen.

Handlungsempfehlung:

Es wird empfohlen, ein strukturiertes und umfassendes Austrittsverfahren auch für ehemalige Mitarbeiter von Herstellern zu implementieren und regelmäßig zu überprüfen. Dieses Verfahren sollte folgende Punkte umfassen:

- **Deaktivierung von Zugängen:** Sofortige Deaktivierung aller Benutzerkonten und Zugriffsrechte in IT-Systemen, Netzwerken und Anwendungen, die dem Mitarbeiter zur Verfügung standen.
- **Rückgabe von Ausrüstung:** Sicherstellung der Rückgabe aller Firmenausstattung, einschließlich Laptops, Mobiltelefone, Zugangskarten und sonstiger Hardware.
- **Entfernung von Berechtigungen:** Entfernung aller physischen und digitalen Zugriffsberechtigungen, einschließlich Zugang zu Büros, Serverräumen und sicheren Bereichen.
- **Dokumentation:** Dokumentation des Austrittsprozesses und der durchgeführten Maßnahmen, um die Nachverfolgbarkeit und Überprüfung zu gewährleisten.
- **Schulung und Sensibilisierung:** Schulung der Personalabteilung und IT-Verantwortlichen zur konsequenten Umsetzung des Austrittsverfahrens und Sensibilisierung für potenzielle Sicherheitsrisiken.
- **Kommunikation:** Benachrichtigung relevanter Abteilungen und Partner über den Austritt des Mitarbeiters, um sicherzustellen, dass alle notwendigen Maßnahmen ergriffen werden.

Durch die Implementierung eines soliden Austrittsverfahrens können potenzielle Sicherheitsrisiken minimiert und die Integrität der Systeme und Daten geschützt werden. Regelmäßige Überprüfungen und Aktualisierungen des Verfahrens stellen sicher, dass es den aktuellen Sicherheitsanforderungen entspricht und effektiv bleibt.

6.15 Annullierung von Zugangsdaten ausscheidender Organisationen

Referenz:

Zuordnung nicht möglich

Schwachstelle:

Die unzureichende Annullierung von Zugangsdaten ausscheidender Anbieter oder Leistungserbringer kann erhebliche Sicherheitsrisiken zur Folge haben. Wenn ehemalige Anbieter oder Leistungserbringer weiterhin Zugang zu sensiblen Systemen und Daten haben, besteht die Gefahr des Missbrauchs dieser Zugänge. Ehemalige Dienstleister könnten absichtlich oder unabsichtlich Schaden anrichten, sensible Informationen kompromittieren oder weiterhin auf vertrauliche Daten zugreifen. Solche Risiken entstehen besonders dann, wenn keine klaren Prozesse zur Deaktivierung von Zugangsdaten und Entzug von Berechtigungen existieren oder diese Prozesse nicht konsequent umgesetzt werden.

Bewertung:

Die Schwachstelle wird als **"niedrig"** eingestuft. Angreifer benötigen für die Durchführung des Angriffs Zugriff auf den HBA und die SMC-B sowie auf den Konnektor. Der HBA und die SMC-B verfallen automatisch nach 5 Jahren. Im Einzelfall kann diese Zeitspanne jedoch zu groß sein.

Handlungsempfehlung:

Um diese Risiken zu minimieren, wird empfohlen, ein klares und striktes Verfahren zur Annullierung von Zugangsdaten ausscheidender Dienste und Leistungserbringer zu implementieren. Dieses Verfahren sollte folgende Punkte umfassen:

- **Zeitnahe Deaktivierung:** Sofortige Deaktivierung aller Zugangsdaten und Berechtigungen in IT-Systemen, Netzwerken und Anwendungen, die dem ausscheidenden Leistungserbringer zur Verfügung standen, sobald das Ausscheiden bekannt ist.
- **Vertragliche Regelungen:** Aufnahme von Klauseln in Verträge mit Leistungserbringern, die eine sofortige Mitteilungspflicht bei Wechseln oder Ausscheiden von Mitarbeitern, sowie eine umgehende Deaktivierung von Zugängen vorsehen.
- **Kontinuierliche Überwachung:** Regelmäßige Überprüfung und Aktualisierung der Zugangslisten, um sicherzustellen, dass nur aktuelle und autorisierte Leistungserbringer Zugriff auf die Systeme haben.
- **Benachrichtigung relevanter Stellen:** Information aller relevanten Abteilungen und Partner über das Ausscheiden des Leistungserbringers, um sicherzustellen, dass alle notwendigen Maßnahmen zur Zugangsannullierung ergriffen werden.
- **Audit und Dokumentation:** Dokumentation des gesamten Prozesses der Annullierung von Zugangsdaten und regelmäßige Audits, um die Einhaltung der Sicherheitsrichtlinien zu überprüfen.

Durch die Implementierung eines robusten Verfahrens zur Annullierung von Zugangsdaten wird sichergestellt, dass ausscheidende Dienste und Leistungserbringer keinen unbefugten Zugriff auf elektronische Patientenakten und andere sensible Daten haben. Dies schützt nicht nur die Integrität der Daten, sondern auch die Privatsphäre der Patienten und die Sicherheit des gesamten Gesundheitssystems.

6.16 Cache-Angriffe auf die VAU

Referenz:

gemSpec_DS_Hersteller, gemSpec_DS_Anbieter

Schwachstelle:

Die sicheren Enklaven der VAU können verwundbar gegenüber Cache-Angriffen sein. Angreifer nutzten dazu die Cache-Hierarchie eines Prozessors, um anhand von wiederkehrenden Mustern Informationen zu extrahiert. [7] Dieser Angriff setzt voraus, dass ein Angreifer bereits Zugang zu dem System hat, auf welchem die VAU ausgeführt wird.

Bewertung:

Die Schwachstelle wird als **"niedrig"** eingestuft. Die Komplexität, die Dauer und die Voraussetzungen für den Angriff sind sehr hoch.

Handlungsempfehlung:

Es wird empfohlen, Maßnahmen gegen Cache-Angriffe zu definieren. Folgen Maßnahmen eignen sich für diesen Zweck:

- **Zufällige Speicherzugriffe auf Anwendungsebene:** Die Hersteller der VAU-Images können Algorithmen für zufällige Speicherzugriffe auf Anwendungsebene implementieren.
- **Address Space Layout Randomization (ASLR):** ASLR ist eine Technik, um Applikationen einen zufälligen Adressbereich zuzuweisen. Diese Funktion kann auf Betriebssystemebene aktiviert werden.

6.17 Kontrollflussmanipulation in der VAU

Referenz:

gemSpec_DS_Hersteller

Schwachstelle:

Es gibt keine Sicherheitsanforderungen, die einer Kontrollflussmanipulation innerhalb der VAU entgegenwirken. Angreifer mit Zugang zu dem System könnten den Kontrollfluss der Applikationen innerhalb der VAU verändern, um schädlichen Code auszuführen oder eine Privilege Escalation herbeizuführen.

7 https://dl.acm.org/doi/10.1145/3065913.3065915

Bewertung:

Die Schwachstelle wird als **"niedrig"** eingestuft. Die Komplexität, die Dauer und die Voraussetzungen für den Angriff sind sehr hoch.

Handlungsempfehlung:

Es wird empfohlen, die Hersteller des VAU-Images zu verpflichten, Control-Flow Integrity (CFI) Technologien zu berücksichtigen. Beispiele dafür sind Microsoft Control Flow Guard (CFG) und LLVM-CFI. Die in Abschnitt 6.16 erwähnte ASLR erhört zusätzlich den Aufwand für Angreifer bei einer Kontrollflussmanipulation.

Alternativ können innerhalb der VAU auch Programmiersprachen und Laufzeitumgebungen verwendet werden, die ein höheres Maß an Speichersicherheit garantieren, z. B. durch Verwendung von RUST statt C.

6.18 Keine Maßnahmen gegen Click-/Tap-Jacking im Frontend

Referenz:

gemSpec_ePA_FdV

Schwachstelle:

Es gibt keine Sicherheitsanforderung gegen Clickjacking bzw. Tap-Jacking innerhalb des Frontends, welches auch oft auch als "UI Redressing" bezeichnet wird. Dadurch kann ein Angreifer eine manipulierte Benutzeroberfläche über das tatsächliche User-Interface der angegriffenen Benutzeroberfläche legen und Benutzer können so getäuscht werden, auf Elemente zu tippen, die sie für einen Teil der legitimen App halten, während sie tatsächlich andere Aktionen ausführen.

Bewertung:

Die Schwachstelle wird als **"niedrig"** eingestuft. Die Voraussetzungen für diesen Angriff sind relativ hoch. Ein Angreifer muss zunächst in der Lage sein, die Benutzeroberfläche zu manipulieren, z. B. indem er den Benutzer dazu bringt, eine Malware-App auf demselben Gerät zu installieren. Dennoch sind solche Angriffe in der Literatur bekannt.

Handlungsempfehlung:

Um diese Risiken zu minimieren, wird empfohlen Sicherheitsanforderungen hinzuzufügen, die die Benutzeroberfläche des Frontends gegen Clickjacking bzw. Tap-Jacking schützen können. Auf der Ebene von Web-basierten Benutzeroberflächen sind dazu wichtige Techniken der Einsatz von **X-Frame-Options** und **Content Security Policy (CSP)**. Über X-Frame-Options kann gesteuert werden, welche Websites die zu schützenden Inhalte in einem Frame anzeigen dürfen. Die DENY-Option kann so z. B. verhindernd, dass die Benutzeroberfläche von anderen Websites in Frames eingebettet wird, über die andere grafische Elemente zur Täuschung gelegt werden. Ebenso ermöglicht eine strikte CSP genau festzulegen, welche Ressourcen und Inhalte von welchen Quellen geladen werden dürfen. So kann mit der frame-ancestors Direktive kontrolliert werden, welche Domains Inhalte in Frames anzeigen dürfen. Zudem bietet eine strikte CSP auch Schutzmöglichkeiten gegen Injektionsangriffe und ist daher eine wichtige zusätzliche Schutzmaßnahme.

Für Android-Apps sollten zudem die von Google beschriebenen Maßnahmen gegen Tap-Jacking[8] berücksichtigt werden.

6.19 Keine Maßnahmen gegen Session Fixation

Referenz:

Zuordnung nicht möglich

Schwachstelle:

Für das Frontend gibt es keine Sicherheitsanforderung, die verhindert, dass ein Angreifer eine gültige Session-ID auf dem Gerät eines Opfers anlegt, bevor das Opfer sich authentifiziert. Nachdem das Opfer die ihm zugewiesene Session-ID verwendet und sich eingeloggt hat, könnte der Angreifer diese Session-ID nutzen, um Zugriff auf das Konto des Opfers zu erhalten, da er die Session-ID bereits kennt.[9]

Bewertung:

Die Schwachstelle wird als **"niedrig"** eingestuft. Die Voraussetzungen und der Aufwand für diesen Angriff sind hoch. Der Angreifer benötigt zunächst Zugriff auf das Endgerät des Benutzers.

Handlungsempfehlung:

Da es sich hierbei um eine typische Schwachstelle im Session-Management handelt, sollte eine Sicherheitsanforderung aufgenommen werden, sodass das Frontend sicherstellt, dass jede Sitzung nur von einem einzigen Client ausgelöst werden kann und eine bestehende Sitzung nicht über einen Link oder Cookies weitergegeben werden kann.

6.20 Fehlende Erkennung von Jailbreak/Root im Frontend

Referenz:

gemSpec_ePA_FdV

Schwachstelle:

Es gibt keine Anforderung zur Erkennung von Jailbreak (iOS) oder Root (Android) auf mobilen Geräten der Benutzer. Solche modifizierten Geräte können potenziell ein höheres Sicherheitsrisiko darstellen, da sie Schutzmechanismen des Betriebssystems umgehen und somit anfälliger für Malware oder unbefugten Zugriff sein können.

Bewertung:

Die Schwachstelle wird als **"niedrig"** eingestuft. Obwohl Jailbreak-/Root-Geräte potenzielle Sicherheitsrisiken bergen, liegt die Verantwortung für die Gerätesicherheit primär beim Benutzer.

8 https://developer.android.com/privacy-and-security/risks/tapjacking
9 https://cwe.mitre.org/data/definitions/384.html

Handlungsempfehlung:

Es wird empfohlen, eine Jailbreak-/Root-Erkennung zu implementieren, die lediglich als Warnmechanismus für den Benutzer dient. Die App sollte:

1. Bei der Erkennung eines Jailbreak-/Root-Geräts eine Warnmeldung anzeigen, die den Benutzer über mögliche Sicherheitsrisiken informiert.
2. Dem Benutzer ermöglichen, die Warnung zu bestätigen und die App dennoch zu nutzen.
3. Keine Funktionen blockieren, sondern dem Benutzer die Entscheidung überlassen, ob er die App auf seinem modifizierten Gerät verwenden möchte.
4. In den Nutzungsbedingungen oder der Datenschutzerklärung darauf hinweisen, dass die Verwendung der App auf Jailbreak-/Root-Geräten auf eigenes Risiko erfolgt.

Diese Vorgehensweise respektiert die Entscheidungsfreiheit des Benutzers, während sie gleichzeitig das Bewusstsein für potenzielle Sicherheitsrisiken schärft.

6.21 Fehlende Erkennung veralteter Geräte im Frontend

Referenz:

Zuordnung nicht möglich

Schwachstelle:

Es gibt keine Anforderung zur Erkennung veralteter Betriebssystemversionen bei Android oder iOS auf mobilen Geräten der Benutzer oder von veralteten Browser-Versionen beim Zugriff per Web-Oberfläche. Veraltete Software besitzt oftmals öffentlich bekannte Sicherheitslücken und kann somit anfälliger für Malware oder unbefugten Zugriff sein.

Bewertung:

Die Schwachstelle wird als **“niedrig”** eingestuft. Obwohl veraltete Geräte potenzielle Sicherheitsrisiken bergen, liegt die Verantwortung für die Gerätesicherheit primär beim Benutzer.

Handlungsempfehlung:

Es wird empfohlen, eine Erkennung veralteter Endgeräte zu implementieren, die lediglich als Warnmechanismus für den Benutzer dient. Die App sollte:

1. Bei der Erkennung einer veralteten Geräteversion eine Warnmeldung anzeigen, die den Benutzer über mögliche Sicherheitsrisiken informiert.
2. Dem Benutzer ermöglichen, die Warnung zu bestätigen und die App dennoch zu nutzen.
3. Keine Funktionen blockieren, sondern dem Benutzer die Entscheidung überlassen, ob er die App auf seinem veralteten Gerät verwenden möchte.
4. In den Nutzungsbedingungen oder der Datenschutzerklärung darauf hinweisen, dass die Verwendung der App auf veralteten Geräten auf eigenes Risiko erfolgt.

Diese Vorgehensweise respektiert die Entscheidungsfreiheit des Benutzers, während sie gleichzeitig das Bewusstsein für potenzielle Sicherheitsrisiken schärft.

6.22 Eingeschränkte Betriebssystemhärtung

Referenz:

gemAnbT_SigD_ATV

Schwachstelle:

Die Anforderung zur Betriebssystemhärtung (GS-A_4316) ist nur dem Signaturdienst zugeordnet, nicht aber anderen relevanten Komponenten.

Bewertung:

Die Schwachstelle wird als **"mittel"** eingestuft. Die Beschränkung der Betriebssystemhärtung auf den Signaturdienst von GS-A_4316 lässt andere potenziell verwundbare Komponenten ungeschützt. Dies könnte Angriffsvektoren eröffnen, die die Sicherheit des Gesamtsystems gefährden, auch wenn der Signaturdienst selbst gut geschützt ist.

Handlungsempfehlung:

Es wird empfohlen, die Anforderung zur Betriebssystemhärtung auf alle sicherheitsrelevanten Komponenten des ePA-Systems auszuweiten. Die Vorgabe sollte einheitlich formuliert und für alle betroffenen Komponenten verbindlich gemacht werden.

7 Dokumentenbasierte Problemstellen

In diesem Kapitel werden potenzielle Problemstellen in den Anforderungen für Hersteller und Anbieter der Komponenten Signaturdienst, IDP-Dienst, sektoraler IDP-Dienst und Aktensystem identifiziert und analysiert. Ziel ist es, Unklarheiten, Redundanzen und Inkonsistenzen aufzudecken, die sich aus den verschiedenen Anbieter- und Produkttypvorschriften (ATV/PTV) für diese spezifischen Komponenten ergeben können. Jeder identifizierte Punkt wird nach folgendem Schema behandelt:

- Referenz: Angabe der betroffenen Komponente(n)
- Beschreibung: Detaillierte Erläuterung der identifizierten Problematik unter Angabe der relevanten Dokumente und Abschnitte, die sich speziell auf die genannten Komponenten beziehen.
- Handlungsempfehlung: Konkrete Vorschläge zur Klärung, Bereinigung oder Harmonisierung der Anforderungen für die jeweilige Komponente.

Diese Analyse soll dazu beitragen, die Anforderungslandschaft für diese Dienste und Systeme zu optimieren, Missverständnisse zu vermeiden und die Umsetzung für deren Hersteller und Anbieter zu erleichtern. Die hier aufgeführten Punkte können als Grundlage für weitere Diskussionen und mögliche Anpassungen der Spezifikationen dieser spezifischen Komponenten dienen.

7.1 Übersicht gefundener Problemstellen

Bezeichner	Referenz	Abschnitt
Mehrere TLS-Versionsanforderungen	übergreifend	7.2
Unklare Vorgaben zu Zertifikatstypen und Schnittstellendefinitionen	gemSpec_SigD	7.3
Unzureichende Notfallwiederherstellungsanforderungen	gemSpec_SigD	7.4
Unklares Verhältnis zwischen BSI-Grundschutz und gematik-Anforderungen	übergreifend	7.5
Unklare Definition personenbezogener Daten in Fehlermeldungen	gemSpec_Aktensystem_ePAfueralle gemSpec_IDP_Sek	7.6
Unklare Definition der Erkennung nicht standardmäßiger Aktennutzung	gemSpec_Aktensystem_ePAfueralle	7.7
Redundante Anforderung zur Gültigkeitsdauer von Vertreterbefugnissen	gemSpec_Aktensystem_ePAfueralle	7.8

7.2 Mehrere TLS-Versionsanforderungen

Referenz:

Übergreifend

Beschreibung:

Die Anforderungen an TLS-Versionen sind über mehrere Vorgaben verteilt (GS-A_4385, A_18467, GS-A_4387, A_18464).

Handlungsempfehlung:

Es wird empfohlen, die Anforderungen an TLS-Versionen in einer einzigen, klaren Vorgabe zusammenzufassen. Diese sollte die Verwendung von TLS 1.2 und 1.3 erlauben und ältere Versionen explizit verbieten.

7.3 Unklare Vorgaben zu Zertifikatstypen und Schnittstellendefinitionen

Referenz: gemSpec_SigD

Beschreibung:

In A_17369 und A_17369-01 werden unterschiedliche Zertifikatstypen genannt (C.CH.AUT_ALT vs. C.CH.SIG), ohne klare Angabe, ob beide Typen parallel unterstützt werden sollen. Ähnlich gilt das auch für die Schnittstellendefinitionen in A_17238-01 und A_17238-02.

Handlungsempfehlung:

Es wird empfohlen, die Anforderungen zu Zertifikatstypen zu vereinheitlichen und klar zu spezifizieren, welche Typen unterstützt werden müssen. Zudem sollten die unterschiedlichen Schnittstellendefinitionen zu einer konsistenten Version zusammengeführt werden.

7.4 Unzureichende Notfallwiederherstellungsanforderungen

Referenz: gemSpec_SigD

Beschreibung

Die Anforderungen zu Backup- und Wiederherstellungsprozessen im Falle eines Ausfalls oder einer Kompromittierung sind nur oberflächlich definiert.

Handlungsempfehlung:

Es wird empfohlen, detaillierte Anforderungen für Notfallwiederherstellungsprozesse zu formulieren. Diese sollten konkrete Vorgaben für Backup-Verfahren, Wiederherstellungsprozesse und Reaktionszeiten beinhalten. Begriffe wie “erheblich” und “unverzüglich” in GS-A_5555 und GS-A_5556 sollten präzise definiert werden.

Ebenso sollte ein klarer Prozess definiert werden, wann und von wem eine Wiederherstellung ausgelöst wird. Hierdurch wird vermieden, dass ein Angreifer bspw. durch Social Engineering das Wiedereinspielen alter Daten erreicht.

7.5 Unklares Verhältnis zwischen BSI-Grundschutz und gematik-Anforderungen

Referenz:

Übergreifend

Beschreibung:

Es ist nicht eindeutig festgelegt, welche Vorgaben Vorrang haben, wenn spezifische gematik-Anforderungen von BSI-Grundschutz-Empfehlungen abweichen.

Handlungsempfehlung:

Es wird empfohlen, eine klare Hierarchie der Anforderungen zu definieren. Dabei sollte festgelegt werden, dass spezifische gematik-Anforderungen Vorrang vor allgemeinen BSI-Grundschutz-Vorgaben haben, falls Widersprüche bestehen. Diese Regelung sollte in die übergreifenden Richtlinien aufgenommen werden.

7.6 Unklare Definition personenbezogener Daten in Fehlermeldungen

Referenz:

gemSpec_Aktensystem_ePAfueralle, gemSpec_IDP_Sek

Beschreibung:

Die Definition, welche Daten in Fehlermeldungen als personenbezogen gelten (GS-A_3813), ist nicht präzise genug und berücksichtigt möglicherweise nicht alle relevanten Metadaten.

Handlungsempfehlung:

Es wird empfohlen, eine detaillierte und umfassende Definition von personenbezogenen Daten im Kontext von Fehlermeldungen zu erarbeiten. Diese sollte auch indirekt personenbezogene Metadaten berücksichtigen und klare Richtlinien für die Gestaltung von Fehlermeldungen vorgeben.

7.7 Unklare Definition der Erkennung nicht standardmäßiger Aktennutzung

Referenz:

gemSpec_Aktensystem_ePAfueralle

Beschreibung:

Die Anforderungen A_15155 und A_15154 zur Erkennung von nicht standardmäßiger Aktennutzung sind unklar definiert. A_15154 fordert die Ermittlung einer Standard-Aktennutzung, während A_15155 verlangt, dass Abweichungen von dieser Standard-Aktennutzung erkannt werden sollen. Allerdings fehlen konkrete Kriterien und Metriken für die Definition einer "Standard-Aktennutzung" sowie klare Schwellenwerte oder Indikatoren für die Erkennung von Abweichungen.

Handlungsempfehlung:

- Präzisierung der Definition der "Standard-Aktennutzung" durch Festlegung konkreter, messbarer Kriterien (z. B. durchschnittliche Anzahl von Zugriffen pro Zeiteinheit, typische Zugriffszeiten, Art der abgerufenen Daten).
- Definition klarer Schwellenwerte oder statistischer Abweichungen, die als nicht standardmäßige Nutzung gelten sollen (z. B. Zugriffe, die mehr als drei Standardabweichungen vom Durchschnitt abweichen).
- Spezifizierung, welche Arten von Abweichungen als besonders kritisch einzustufen sind und unmittelbare Maßnahmen erfordern.
- Ergänzung von Anforderungen zur regelmäßigen Überprüfung und Anpassung der Erkennungsparameter, um auf sich ändernde Nutzungsmuster reagieren zu können.
- Vorgaben zur Protokollierung und Meldung erkannter Abweichungen, einschließlich eines definierten Eskalationsprozesses.

7.8 Redundante Anforderung zur Gültigkeitsdauer von Vertreterbefugnissen

Referenz:

gemSpec_Aktensystem_ePAfueralle

Beschreibung:

Die Anforderung A_24537 zur Gültigkeitsdauer von Befugnissen impliziert die Anforderung A_24536. A_24536 legt fest, dass die Gültigkeitsdauer der Befugnisse für Vertreter bis zu deren Entzug gilt. A_24537 definiert eine Standardgültigkeitsdauer für Befugnisse, die Vertreterbefugnisse bereits einschließt. Somit impliziert A_24537 die in A_24536 geforderte Regelung.

Handlungsempfehlung:

Die redundante Anforderung A_24536 sollte entfernt werden. A_24537 kann zusätzlich überarbeitet werden, um die Regelung für Vertreterbefugnisse noch deutlicher hervorzuheben.

8 Bewertung der Rollentrennung

Innentäter bei Betreiberorganisationen stellen eine große Herausforderung bei der Absicherung der elektronischen Patientenakte dar. Betriebsinterne Täter können über weitreichende Berechtigungen und Insiderwissen verfügen. Um auch diese Bedrohung auf ein Minimum zu reduzieren, ist eine Rollentrennung für bestimmte Aufgaben unbedingt notwendig. Die Sicherheitsanforderungen für die elektronische Patientenakte beinhalten daher mehrere sinnvolle Maßnahmen zur Rollentrennung:

8.1 Rollentrennung ePA-Aktensystem und IDP-Dienst

Die Maßnahme A_24986 verhindert, dass sich ein einzelner Mitarbeiter einen gültigen ID-Token ausstellen könnte und so einen vermeintlich legitimen Zugriff auf die Daten anderer versicherter Personen erhalten könnte. Entsprechend der Sicherheitsanforderung ist dies durch geeignete technische und organisatorische Maßnahmen umzusetzen. Dabei sollte aber berücksichtigt werden, dass organisatorische Maßnahmen wie z. B. Dienstanweisungen (oder andere Verhaltensvorschriften) zu Regelungen von Arbeitsabläufen alleine nicht ausreichend sind, da Mitarbeiter, die dem eigenen Unternehmen gegenüber negativ eingestellt sind, die Vorschriften missachten werden.

8.2 Rollentrennung ePA-Aktensystem und sektoraler IDP

Die Maßnahme A_25149-01 verhindert, dass ein einzelner Mitarbeiter den Zugriff über das Frontend des Versicherten erhalten kann. Genau wie bei der Sicherheitsanforderung A_24986 ist allerdings darauf zu achten, dass geeignete technische Maßnahmen zur Rollentrennung angewendet werden.

8.3 Rollentrennung zur Gewährleistung der Vertraulichkeit und der Integrität

In der Sicherheitsanalyse wurden keine Hinweise auf fehlende Maßnahmen zur Rollentrennung gefunden, die einem einzelnen Mitarbeiter des Betreibers des Aktensystems den Zugriff auf sensible Versichertendaten ermöglichen. Es ist jedoch darauf hinzuweisen, dass dies in der Praxis aufgrund von Sicherheitslücken oder Implementierungsfehlern nicht mehr der Fall sein kann.

8.4 Rollentrennung zur Gewährleistung der Verfügbarkeit

Sicherheitsanforderungen zur Gewährleistung der Schutzziele Vertraulichkeit und Integrität sind in den Spezifikationen enthalten. Die Angriffsbäume offenbaren jedoch zwei Schwächen, die aufgrund fehlender Rollentrennung negative Auswirkungen auf die Verfügbarkeit der elektronischen Patientenakte haben können. In Abschnitt 6.3 wird dargestellt, dass ein einzelner Mitarbeiter mit Zugang zum Rechenzentrum das HSM zerstören könnte. Wenn derselbe Mitarbeiter auch Zugriff auf die Backups der HSMs besitzt, könnte er auch diese zerstören. Die Daten sind dann unwiederbringlich verschlüsselt, was zu einem Denial of Service der ePA führt. Eine weitere Schwäche wird in Abschnitt 6.4 aufgezeigt. Ein einzelner Mitarbeiter beim Betreiber des Aktensystems oder beim Betreiber des sektoralen IDP könnte ein Denial of Service auslösen, in dem er alle Server für das Access Gateway oder den sektoralen IDP per Remote herunterfährt oder im Rechenzentrum ausschaltet.

8.5 Handlungsempfehlung für die Rollentrennung

Aus der Bewertung der Rollentrennung können zwei Handlungsempfehlungen abgeleitet werden:

- In den Sicherheitsanforderungen sollte deutlicher formuliert werden, dass organisatorische Maßnahmen wie Dienstanweisung alleine nicht ausreichend sind, um eine Rollentrennung zu etablieren.
- Die Erhaltung der Verfügbarkeit muss stärker berücksichtigt werden. Einzelne Mitarbeiter dürfen nicht in der Lage sein, alle zentralen Komponenten herunterzufahren oder zu administrieren. Dies kann durch eine entsprechende Vergabe von Berechtigungen (Zugangsberechtigungen für das Rechenzentrum, Berechtigungen für privilegierte Nutzer auf den Servern und Berechtigungen für die Management Engine) aufseiten der Betreiber umgesetzt werden.
- Die gematik sollte im Rahmen ihrer Auditrechte prüfen, ob die Rollentrennung auf Berechtigungsebene durch die Anbieter umgesetzt wurde.
- Zur Absicherung der HSM-Backups (Backup der Masterkeys aus dem HSMs) sollte die gematik ein 4-Augen-Prinzip einführen. Der gesamte Backup-Prozess sollte von einer Vertrauensperson der gematik begleitet werden. Auch die Lagerung der Backups sollte so gestaltet werden, dass eine einzelne Person keinen Zugriff auf die Backups der HSMs besitzt.

9 Anhang

9.1 TL1: Unbefugtes Lesen der Akte (komplett)

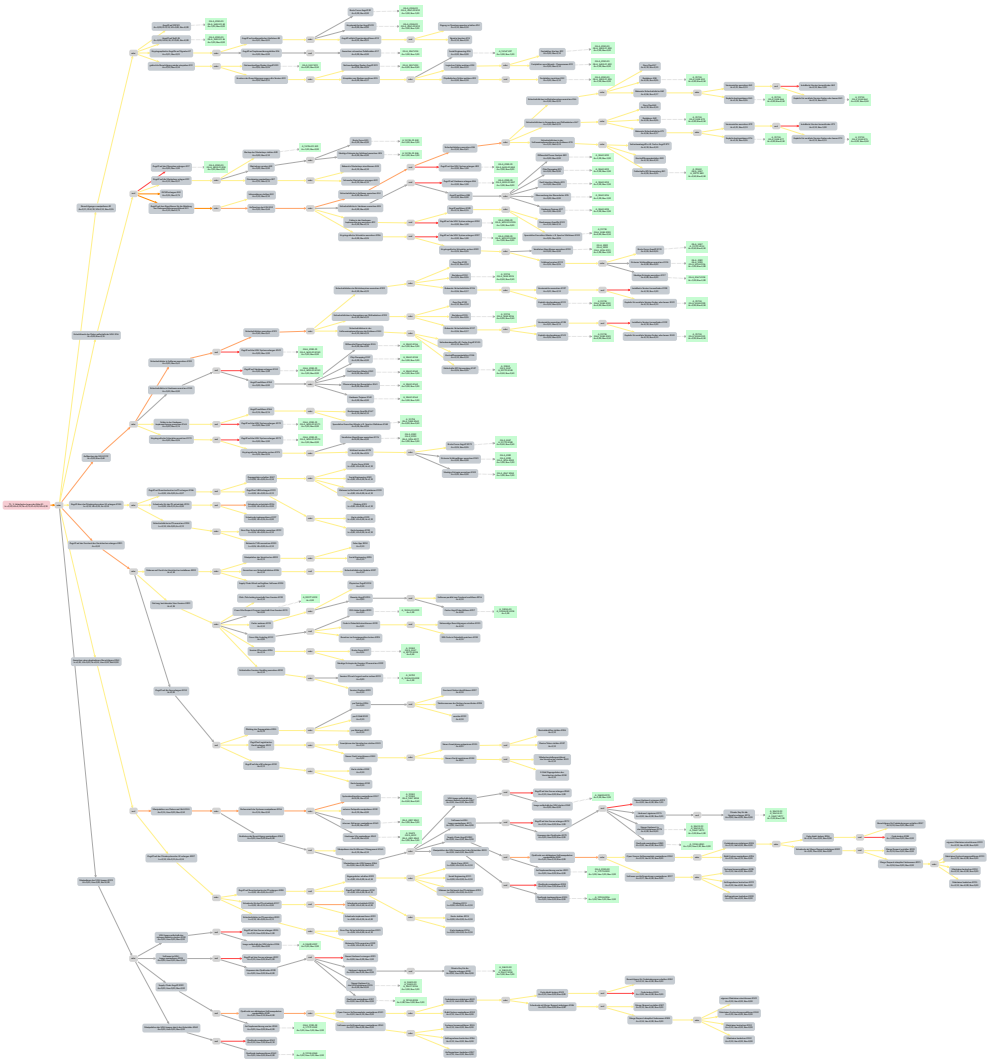

Abb. 9.1:
TL1: Unbefugtes Lesen der Akte (komplett)

9.2 TL2: Unbefugtes Manipulieren der Akte (komplett)

Abb. 9.2:
TL2: Unbefugtes Manipulieren der Akte (komplett)

9.3 TL3: Unbefugtes Löschen der Akte (komplett)

Abb. 9.3:
TL3: Unbefugtes Löschen der Akte (komplett)

9.4 TL4: Denial-of-Service-Angriff auf Akte (komplett)

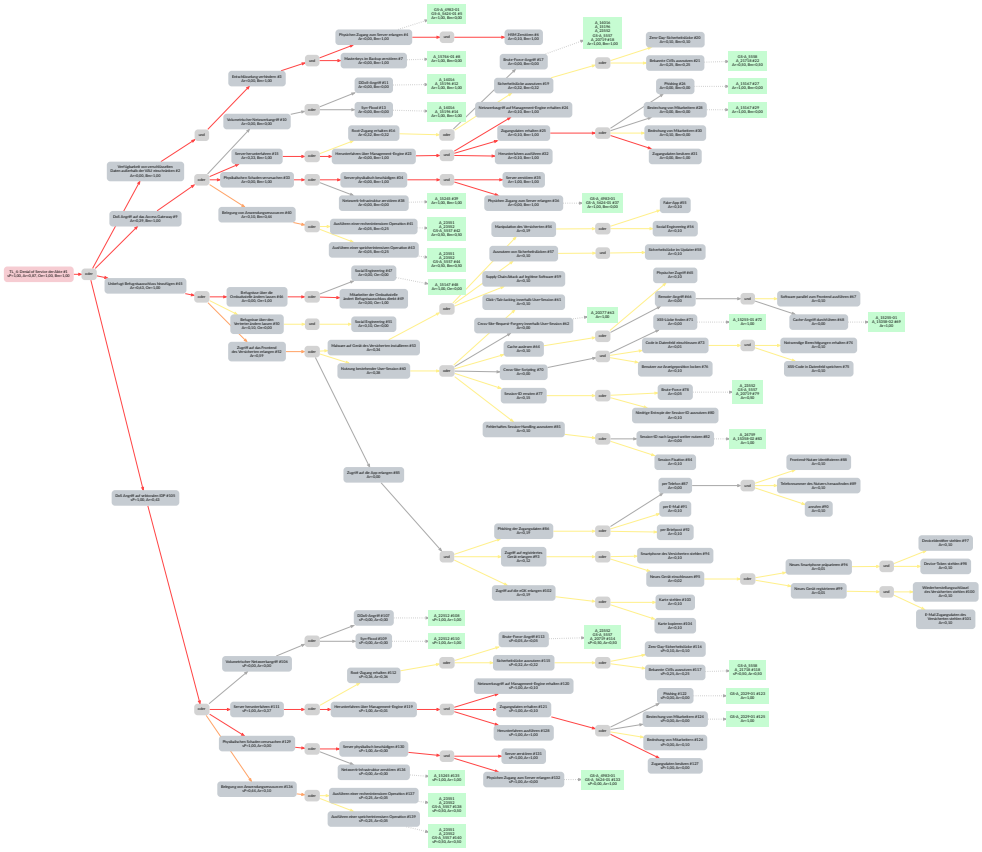

Abb. 9.4:
TL4: Denial-of-Service-Angriff auf Akte (komplett)

9.5 TL5: Aufdecken von Behandlungen und Krankheiten (komplett)

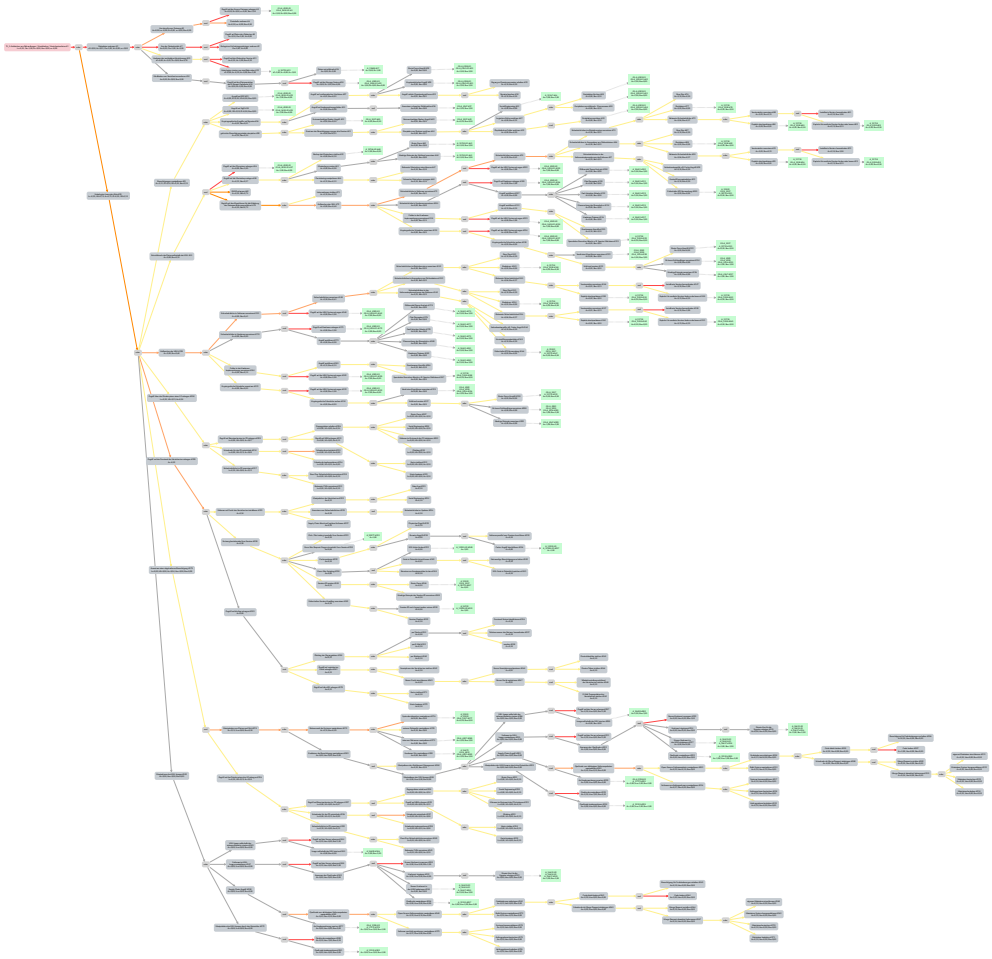

Abb. 9.5:
TL5: Aufdecken von Behandlungen und Krankheiten (komplett)