Gesellschaft für Telematikanwendungen der Gesundheitskarte mbH

Elektronische Gesundheitskarte und Telematikinfrastruktur

Spezifikation Autorisierung ePA

Version:	1.3.0
Revision:	167250
Stand:	02.10.2019
Status:	freigegeben
Klassifizierung:	öffentlich
Referenzierung:	gemSpec_Autorisierung

Dokumentinformationen

Änderungen zur Vorversion

Anpassungen des vorliegenden Dokumentes im Vergleich zur Vorversion können Sie der nachfolgenden Tabelle entnehmen.

Dokumentenhistorie

Version	Stand	Kap./ Seite	Grund der Änderung, besondere Hinweise	Bearbeitung
1.0.0	18.12.18		freigegeben	gematik
1.1.0	15.05.19		Einarbeitung Änderungsliste P18.1	gematik
1.2.0	28.06.19		Einarbeitung Änderungsliste P19.1	gematik
			Einarbeitung Änderungsliste P20.1/2	gematik
1.3.0	02.10.19		freigegeben	gematik

Inhaltsverzeichnis

1 Einordnung des Dokumentes 6
1.1 Zielsetzung 6
1.2 Zielgruppe 6
1.3 Geltungsbereich 6
1.4 Abgrenzungen 6
1.5 Methodik 7
2 Systemüberblick 8
3 Systemkontext 9
3.1 Akteure und Rollen 9
3.2 Nachbarsysteme 12
3.3 Tokenbasierte Autorisierung 13
4 Zerlegung der Komponente Autorisierung 14
5 Übergreifende Festlegungen 15
5.1 Datenschutz und Datensicherheit 15
5.2 Verwendete Standards 19
5.3 Protokollierung 20
5.4 Fehlerbehandlung in Schnittstellenoperationen 22
5.5 Nicht-Funktionale Anforderungen 24
5.5.1 Skalierbarkeit 24
5.5.2 Performance 24
5.5.3 Mengengerüst 24
6 Funktionsmerkmale 25
6.1 Übergreifende Festlegungen 25
6.2 Schnittstellen der Komponente Autorisierung 27
6.2.1 Schnittstelle I_Authorization 29
6.2.1.1 Operationsdefinition I_Authorization::getAuthorizationKey 29
6.2.1.2 Umsetzung I_Authorization::getAuthorizationKey 31
6.2.2 Schnittstelle I_Authorization_Insurant 32
6.2.2.1 Operationsdefinition I_Authorization_Insurant::getAuthorizationKey 32
6.2.2.2 Umsetzung I_Authorization_Insurant::getAuthorizationKey 35
6.2.3 Schnittstelle I_Authorization_Management 36
6.2.3.1 Operationsdefinition I_Authorization_Management::putAuthorizationKey . 36
6.2.3.2 Umsetzung I_Authorization_Management::putAuthorizationKey 38

6.2.3.3 Operationsdefinition I_Authorization_Management::checkRecordExists...39
6.2.3.4 Umsetzung I_Authorization_Management::checkRecordExists.....40
6.2.3.5 Operationsdefinition I_Authorization_Management::getAuthorizationList..40
6.2.3.6 Umsetzung I_Authorization_Management::getAuthorizationList.....41
6.2.4 Schnittstelle I_Authorization_Management_Insurant.....41
6.2.4.1 Operationsdefinition I_Authorization_Management_Insurant::putAuthorizationKey 42
6.2.4.2 Umsetzung I_Authorization_Management_Insurant::putAuthorizationKey 43
6.2.4.3 Operationsdefinition I_Authorization_Management_Insurant::deleteAuthorizationKey.....46
6.2.4.4 Umsetzung I_Authorization_Management_Insurant::deleteAuthorizationKey.....47
6.2.4.5 Operationsdefinition I_Authorization_Management_Insurant::replaceAuthorizationKey.....48
6.2.4.6 Umsetzung I_Authorization_Management_Insurant::replaceAuthorizationKey.....50
6.2.4.7 Operationsdefinition I_Authorization_Management_Insurant::getAuditEvents 51
6.2.4.8 Umsetzung I_Authorization_Management_Insurant::getAuditEvents.....52
6.2.4.9 Operationsdefinition I_Authorization_Management_Insurant::putNotificationInfo.....52
6.2.4.10 Umsetzung I_Authorization_Management_Insurant::putNotificationInfo .54
6.2.4.11 Operationsdefinition I_Authorization_Management_Insurant::getAuthorizationList 54
6.2.4.12 Umsetzung I_Authorization_Management_Insurant::getAuthorizationList 56
6.3 Berechtigungstypen der Autorisierung 57
6.4 Hardware-Merkmal der Komponente Autorisierung 58
6.5 Geräteverwaltung 58
6.5.1 Freischaltprozess neuer Geräte.....58
6.5.2 Geräteadministration.....61
6.6 Freischaltprozess Vertretereinrichtung 62
7 Informationsmodell 65
7.1 Namensräume.....66
7.2 SAML-Profil und Tokeninhalte.....66
8 Verteilungssicht 70
9 Anhang A – Verzeichnisse.....71
9.1 Abkürzungen 71
9.2 Glossar.....71
9.3 Abbildungsverzeichnis 71
9.4 Tabellenverzeichnis 72
9.5 Referenzierte Dokumente 72
9.5.1 Dokumente der gematik.....72

9.5.2 Weitere Dokumente 73

1 Einordnung des Dokumentes

1.1 Zielsetzung

Das vorliegende Dokument spezifiziert die Anforderungen an die Komponente "Autorisierung" des Produkttyps ePA-Aktensystem. Die Komponente Autorisierung ist verantwortlich für die zentrale Verwaltung des empfängerbezogenen verschlüsselten Schlüsselmaterials.

1.2 Zielgruppe

Das Dokument richtet sich an Hersteller und Anbieter der Komponente "Autorisierung" für die Nutzung in einem ePA-Aktensystem sowie an Hersteller und Anbieter von Produkttypen ePA, die Schnittstellen der Komponente "Autorisierung" nutzen.

1.3 Geltungsbereich

Dieses Dokument enthält normative Festlegungen zur Telematikinfrastruktur des deutschen Gesundheitswesens. Der Gültigkeitszeitraum der vorliegenden Version und deren Anwendung in Zulassungs- oder Abnahmeverfahren wird durch die gematik GmbH in gesonderten Dokumenten (z.B. Dokumentenlandkarte, Produkttypsteckbrief, Leistungsbeschreibung) festgelegt und bekannt gegeben.

Schutzrechts-/Patentrechtshinweis

Die nachfolgende Spezifikation ist von der gematik allein unter technischen Gesichtspunkten erstellt worden. Im Einzelfall kann nicht ausgeschlossen werden, dass die Implementierung der Spezifikation in technische Schutzrechte Dritter eingreift. Es ist allein Sache des Anbieters oder Herstellers, durch geeignete Maßnahmen dafür Sorge zu tragen, dass von ihm aufgrund der Spezifikation angebotene Produkte und/oder Leistungen nicht gegen Schutzrechte Dritter verstoßen und sich ggf. die erforderlichen Erlaubnisse/Lizenzen von den betroffenen Schutzrechtsinhabern einzuholen. Die gematik GmbH übernimmt insofern keinerlei Gewährleistungen.

1.4 Abgrenzungen

Spezifiziert werden in dem Dokument die von der Komponente bereitgestellten (angebotenen) Schnittstellen. Benutzte Schnittstellen werden hingegen in der Spezifikation desjenigen Produkttypen beschrieben, der diese Schnittstelle bereitstellt. Auf die entsprechenden Dokumente wird referenziert (siehe auch Anhang A5).

Die vollständige Anforderungslage für den Produkttyp ergibt sich aus weiteren Konzept- und Spezifikationsdokumenten. Diese sind in dem Produkttypsteckbrief des Produkttyps <ePA-Aktensystem> verzeichnet.

Nicht Bestandteil des vorliegenden Dokumentes sind die Festlegungen zum Themenbereich Betrieb.

1.5 Methodik

Anforderungen als Ausdruck normativer Festlegungen werden durch eine eindeutige ID sowie die dem RFC 2119 [RFC2119] entsprechenden, in Großbuchstaben geschriebenen deutschen Schlüsselworte MUSS, DARF NICHT, SOLL, SOLL NICHT, KANN gekennzeichnet.

Da in dem Beispielsatz „Eine leere Liste DARF NICHT ein Element besitzen.“ die Phrase „DARF NICHT“ semantisch irreführend wäre (wenn nicht ein, dann vielleicht zwei?), wird in diesem Dokument stattdessen „Eine leere Liste DARF KEIN Element besitzen.“ verwendet. Die Schlüsselworte werden außerdem um Pronomen in Großbuchstaben ergänzt, wenn dies den Sprachfluss verbessert oder die Semantik verdeutlicht.

Anforderungen werden im Dokument wie folgt dargestellt:
<AFO-ID> - <Titel der Afo>
Text / Beschreibung
[<=]

Dabei umfasst die Anforderung sämtliche zwischen Afo-ID und Textmarke [<=] angeführten Inhalte.

2 Systemüberblick

Der Autorisierungsdienst ePA ist eine Komponente des Produkttyps ePA-Aktensystem. Die Systemzerlegung der Fachanwendung ePA in Komponenten und Produkttypen sowie die Verteilung der Komponenten auf Produkttypen der Telematikinfrastruktur (TI) ist in [gemSysL_ePA#2.1] und in [gemSysL_ePA#4.1] definiert.

Die Komponente Autorisierungsdienst ePA verwaltet das empfängerverschlüsselte Schlüsselmaterial der Nutzer eines Aktenkontos eines Versicherten (kryptografische Autorisierung). Mit dem Vorhandensein einer kryptografischen Berechtigung ist ein Nutzer in der Lage, auf den symmetrischen Aktenschlüssel sowie den Kontextschlüssel zuzugreifen. Um dieses Schlüsselmaterial für den Zugriff auf medizinische Daten und Dokumente eines Versicherten zu nutzen, benötigt ein Nutzer ggfs. zusätzlich Berechtigungen auf Objektebene in anderen Komponente und Produkttypen, die die Daten und Dokumente des Versicherten verwalten.

3 Systemkontext

Der folgende Abschnitt setzt die Komponente Autorisierung in den Systemkontext der Fachanwendung ePA.

3.1 Akteure und Rollen

Die Komponente Autorisierung wird als Provider technischer Schnittstellen von weiteren technischen Komponenten und Produkttypen der Fachanwendung ePA aufgerufen. Diese weiteren Komponenten und Produkttypen nutzen die Schnittstellen der Komponente Autorisierung im Zusammenhang von fachlichen Anwendungsfällen der Nutzer der Fachanwendung ePA.

Die Nutzer sind dabei gesetzlich Versicherte, Leistungserbringerinstitutionen und Kostenträger, welche durch ihre jeweilige Karte der TI repräsentiert werden. Über eine kartenbasierte Authentifizierungsbestätigung authentisieren sie sich gegenüber der Komponente Autorisierung. Ein Spezialfall des gesetzlichen Versicherten ist der berechtigte Vertreter.

Für die oben genannten Nutzer verwaltet die Komponente Autorisierung empfängerbezogen verschlüsseltes Schlüsselmaterial

- für Versicherte, plus den Spezialfall des Vertreters - verschlüsselt für die individuelle KVNR
- für Leistungserbringerinstitutionen und Kostenträger - verschlüsselt für die individuelle Telematik-ID

Die Komponente Autorisierung wird je nach Erfordernis zur Laufzeit von einem Administrator administriert. Gemäß der Festlegungen des Rollenmodells "Personenkreise der Telematikinfrastruktur" in [gemKPT_Arch_TIP] haben Anbieter, Betreiber und Administratoren keinen Zugriff auf medizinische Daten der Anwendungen des §291a SGB V [SGB V]. Die Komponente Autorisierung speichert personenbezogene Informationen, jedoch keine medizinischen Daten im Sinne des § 291a SGB V [SGB V].

Das folgende Bild gibt eine Übersicht der durch die Schnittstellen realisierten Anwendungsfälle zur Schlüsselverwaltung der Komponente Autorisierung. Zur Vereinfachung sind die Anwendungsfälle der Protokollierung und Geräteverwaltung nicht dargestellt.

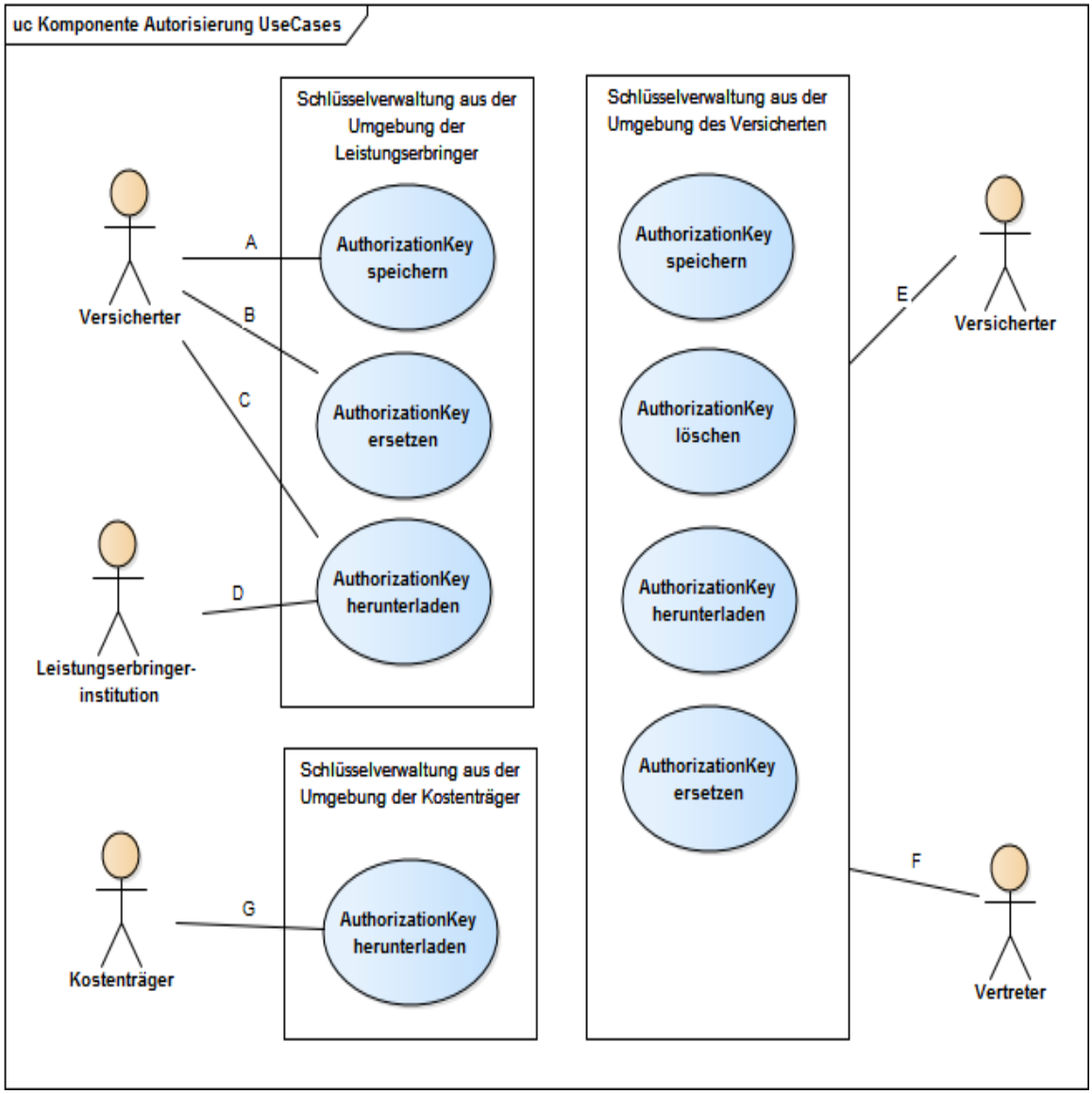


Abbildung 1: Anwendungsfälle der Schlüsselverwaltung nach Umgebung

Die Berechtigung für Anwendungsfälle der Schlüsselverwaltung durch einen Nutzer unterscheidet sich nach Umgebung. Dem Versicherten stehen in der Umgebung der Leistungserbringer keine Anwendungsfälle zum Löschen bestehender Berechtigungen zur Verfügung, da ihm dort kein geeignetes Benutzerinterface zur Verfügung steht. Ein Ersetzen des Schlüsselmaterials erfolgt bei Vergabe einer Änderungsberechtigung für eine Leistungserbringerinstitution, wenn bspw. die Gültigkeitsdauer der Berechtigung angepasst wird.

Eine Leistungserbringerinstitution kann auf das für sie hinterlegte Schlüsselmaterial lesend zugreifen. Analog kann ein Kostenträger nur auf das für ihn hinterlegte Schlüsselmaterial lesend zugreifen.

In der Umgebung des Versicherten hat ein Versicherter vollen Zugriff auf das hinterlegte Schlüsselmaterial mit folgender Ausnahme - ein Versicherter darf das eigene Schlüsselmaterial für die eGK des Versicherten nicht löschen. Ein Vertreter führt Anwendungsfälle der Vertretung ausschließlich in der Umgebung eines Versicherten aus.

Ebenso darf der Vertreter nicht das Schlüsselmaterial des Versicherten löschen und auch nicht Schlüsselmaterial für andere eGK-Inhaber hinzufügen (kein Einrichten weiterer Vertretungen durch einen Vertreter).

Ergänzende Informationen zu Bezeichnern und Datentypen finden sich im Informationsmodell in Abschnitt 7.

Tabelle 1: Anwendungsfälle der Schlüsselverwaltung nach Umgebung

Assoziation	Actor	Regel zur Identifikation des Nutzers*
A	Versicherter	subject-id == OwnerKVNR == ActorID
B		
C		
D	Leistungserbringer-institution	subject-id == ActorID != OwnerKVNR (für HBA – erst in Folgestufe) organization-id == ActorID != OwnerKVNR (für SMC-B)
E	Versicherter	subject-id == OwnerKVNR
F	Vertreter	subject-id == ActorID != OwnerKVNR (beim Verwalten des Vertretungsschlüssels) subject-id != ActorID != OwnerKVNR (beim Verwalten aller übrigen Schlüssel)
G	Kostenträger	organization-id == ActorID != OwnerKVNR (für SMC-B KTR)

* `subject-id/organization-id` *ist Teil der Authentication- bzw. AuthorizationAssertion (als Behauptung gemäß [gemSpec_TBAuth#TAB_TBAuth_02_1/2]),* `OwnerKVNR` *ist ein Attribut der KeyChain (vgl. Kap. 7 Informationsmodell), der mehrere AuthorizationKeys untergeordnet werden,* `ActorID` *meint hier den Teil des AuthorizationKeys der dessen Besitzer identifiziert, (einige Schnittstellenoperationen verfügen über einen Parameter* `ActorID`*, dieser ist hier jedoch nicht Gegenstand der Betrachtung)*

Der Versicherte wird beim Einsatz der eGK in der Umgebung der Leistungserbringer (Anwendungsfälle A und B) und in Anwendungsfällen aus der Umgebung des Versicherten (Anwendungsfälle zu E) anhand der KVNR als subject-id eines AuthenticationTokens erkannt. Diese stimmt gleichzeitig mit der OwnerKVNR des Eigentümers der Akte überein. Im Regelfall existiert für den Versicherten ein AuthorizationKey mit der KVNR des Versicherten als ActorID. Im Zustand der Kontoeröffnung und bei Anbieterwechsel wird das Schlüsselmaterial für den Versicherten extern erzeugt. Ein Nicht-Vorhandensein eines AuthorizationKeys für den Versicherten

wird nicht als Fehler behandelt, sondern als Autorisierung im Zusammenhang mit Anwendungsfällen der Kontoverwaltung.

Eine Leistungserbringerinstitution wird bei Einsatz einer SMC-B (Anwendungsfälle C und D) anhand ihrer Telematik-ID aus der organization-id eines AuthenticationTokens erkannt. Für diese Telematik-ID muss ein AuthorizationKey mit gleichlautender ActorID vorhanden sein, andernfalls ist diese Leistungserbringerinstitution nicht autorisiert. Das gleiche gilt für die Kostenträger (Anwendungsfälle G und H).

Der Vertreter wird zunächst als Versicherter mit eigener eGK anhand der KVNR als subject-id eines AuthenticationTokens erkannt. In der Wahrnehmung einer Vertretung (Anwendungsfälle F) ist seine KVNR ungleich der OwnerKVNR des Eigentümers der Akte. Für seine KVNR muss ein AuthorizationKey mit gleichlautender ActorID vorhanden sein, andernfalls ist der Vertreter für den Zugriff nicht autorisiert.

3.2 Nachbarsysteme

Der folgende Abschnitt beschreibt die Positionierung der Komponente Autorisierung im Kontext der Fachanwendung ePA.

Die folgende Abbildung zeigt die Beziehung zu benachbarten Produkttypen innerhalb der Fachanwendung mit den von der Komponente Autorisierung bereitgestellten Schnittstellen.

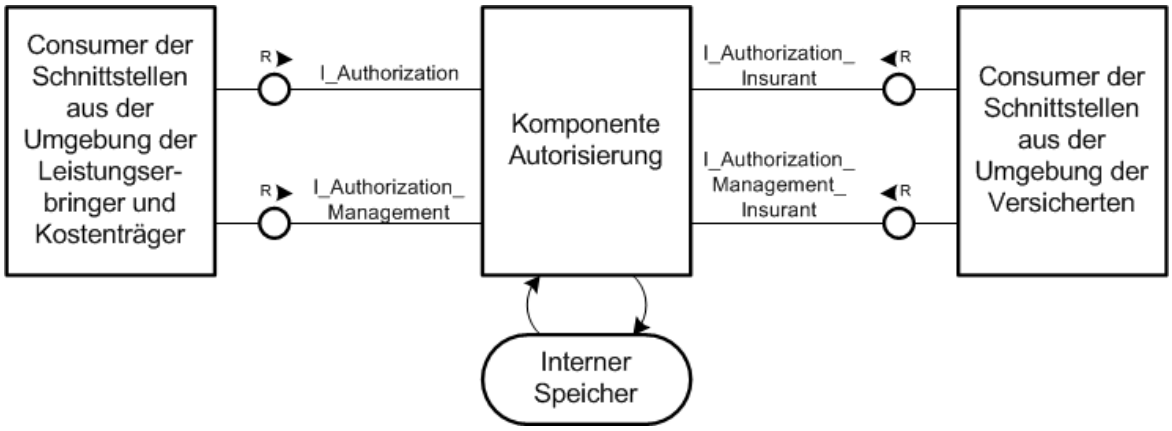


Abbildung 2: Komponente Autorisierung, benachbarte Komponenten und Produkttypen

Die Komponente Autorisierung stellt die Schnittstellen `I_Authorization` und `I_Authorization_Management` zur Nutzung aus der Umgebung der Leistungserbringer und Kostenträger bereit. Von dort werden sie aus der Secure Consumer Zone aufgerufen.

Die Schnittstellen `I_Authorization_Insurant` und `I_Authorization_Management_Insurant` werden aus der Personal Zone in der Umgebung des Versicherten aufgerufen. In dieser Umgebung nutzt der Versicherte das ePA-Frontend des Versicherten auf einem Gerät des Versicherten.

Die Komponente Autorisierung wird als Teil des Produkttyps ePA-Aktensystem in der Provider Zone der Telematikinfrastruktur betrieben. Sie verfügt über einen logischen, internen Speicher, an den in diesem Dokument keine Umsetzungsanforderungen gestellt werden. Er dient der Persistierung der im Informationsmodell (siehe 7-Informationsmodell) strukturierten Inhalte.

A_13956 - Komponente Autorisierung -Separierung der Schnittstellen für verschiedene Umgebungen
Die Komponente Autorisierung MUSS die Bereitstellungspunkte der Schnittstellen für die Nutzung durch benachbarte Komponenten und Produkttypen aus verschiedenen Einsatzumgebungen voneinander separieren. **[<=]**

Diese Separierung kann beispielsweise umgesetzt werden durch die Erreichbarkeit der Schnittstellen über verschiedene Netzwerkadressen.

3.3 Tokenbasierte Autorisierung

Die Komponente Autorisierung bietet eine Single-Sign-On (SSO)-Lösung an, um einem zuvor authentifizierten Nutzer den Zugriff auf weitere Ressourcen zu ermöglichen. Hierbei wird nach einer erfolgreichen Autorisierung eine Autorisierungsbestätigung (AuthorizationAssertion gemäß SAML 2.0 Assertions [SAML2.0]) ausgestellt.

Für die Initialisierung sowie für den Zugriff auf den Aktenkontext eines Versicherten erwartet die Komponente Dokumentenverwaltung eine gültige Assertion von der Komponente Autorisierung. Die Assertion wird ungültig, wenn der Aktenkontext eines Versicherten geschlossen wird oder der Gültigkeitszeitraum der Assertion abgelaufen ist.

4 Zerlegung der Komponente Autorisierung

Eine detaillierte Zerlegung der Komponente Autorisierung wird nicht vorgegeben. Gleichwohl muss die Komponente Autorisierung privates Schlüsselmaterial in einem HSM speichern, das den Anforderungen einer bestimmten Prüftiefe entspricht. Auf eine grafische Darstellung wird an dieser Stelle verzichtet.

5 Übergreifende Festlegungen

5.1 Datenschutz und Datensicherheit

Im folgenden Abschnitt werden die für die Komponente Autorisierung notwendigen Anforderungen für den Schutz personenbezogener Daten bzw. Anforderungen für den Schutz von Daten beschrieben, um beispielsweise vor Datenmanipulation oder Datenverlust zu schützen.

A_14417 - Komponente Autorisierung - Akzeptieren von Identitätsbestätigungen
Die Komponente Autorisierung MUSS Identitätsbestätigungen (AuthenticationAssertion) als ungültig mit dem Fehler ASSERTION_INVALID ablehnen, wenn die Identität des Ausstellers (`Issuer`) nicht als vertrauenswürdiger Dienst für die Durchführung einer Authentifizierung konfiguriert ist oder dessen X.509-Signatur-Zertifkat nicht zu der Signatur der Identitätsbestätigung passt.
[<=]

A_13990 - Komponente Autorisierung - Vorgaben für Identitätsbestätigung
Die Komponente Autorisierung MUSS eine übergebene Identitätsbestätigung (AuthenticationAssertion) als ungültig mit dem Fehler ASSERTION_INVALID ablehnen, wenn diese nicht konform zu den Vorgaben der Tabelle [gemSpec_TBAuth#TAB_TBAuth_03 Identitätsbestätigung] ist.**[<=]**

A_14688 - Komponente Autorisierung - Prüfung einer Identitätsbestätigung
Die Komponente Autorisierung MUSS eine übergebene Identitätsbestätigung (AuthenticationAssertion) als ungültig mit dem Fehler ASSERTION_INVALID ablehnen, die nach einer Prüfung gemäß [gemSpec_TBAuth#A_15557] (vgl. auch gemSpec_TBAuth#3.2 Prüfen von Identitätsbestätigungen) als nicht gültig betrachtet wird. Insbesondere MUSS die Komponente Autorisierung das Signaturzertifikat der Ausstelleridentität eines Vertrauensraums außerhalb des Vertrauensraums der Komponente Autorisierung mittels [gemSpec_PKI#TUC_PKI_018] mit den folgenden Parametern prüfen:

Parameter	**Belegung für SAML 2.0 Assertions des Fachmoduls ePA**	
Zertifikat	Signaturzertifikat (eingebettet in Identitätsbestätigung) C.HCI.OSIG	
PolicyList	oid_smc_b_osig	
intendedKeyUsage	nonRepudiation	
intendedExtendedKeyUsage	(leer)	
OCSP-Graceperiod	60 Minuten	

Offline-Modus	nein	
Prüfmodus	OCSP	

Das Signaturzertifikat muss anhand der Zertifikatsprüfung für [mathematisch gültig UND zeitlich gültig UND online gültig] befunden werden.

[<=]

A_17839 - Komponente Autorisierung - Prüfung der Empfänger-Rolle
Die Komponente Autorisierung MUSS beim Aufruf einer der Operation

- I_Authorization~~_Management~~::getAuthorizationKey

den übergebenen Parameter `AuthenticationAssertion` dahingehend prüfen, ob mindestens eine `ProfessionOID` der ZertifikatsExtension `Admission` gemäß [gemSpec_PKI#Tab_PKI_226] im Signaturzertifikat C.HCI.OSIG `/saml2:Assertion/ds:Signature/ds:KeyInfo/ds:X509Data/ds:X509Certificate` in der Liste der zulässigen Autorisierungsempfänger-Rollen gemäß [gemSpec_OID#Tab_PKI_402] und [gemSpec_OID#Tab_PKI_403]

- oid_praxis_arzt
- oid_zahnarztpraxis
- oid_praxis_psychotherapeut
- oid_krankenhaus
- oid_oeffentliche_apotheke
- oid_krankenhausapotheke
- oid_bundeswehrapotheke
- oid_mobile_einrichtung_rettungsdienst
- oid_kostentraeger

enthalten ist und sofern nicht, die Operation mit dem Fehler AUTHORIZATION_ERROR abbrechen.

[<=]

Ist die AuthenticationAssertion vom Aktensystem selbst erstellt worden (/`saml2:Assertion/ds:Signature/ds:KeyInfo/ds:X509Data/ds:X509Certificate` enthält das Signaturzertifikat C.FD.SIG des Aktensystems), entfällt die Rollenprüfung, da die Rolle des Versicherten bereits durch Komponente Authentisierung Versicherter geprüft wurde.

A_17840 - Komponente Autorisierung Vers. - Prüfung Identitätswechsel des Versicherten
Die Komponente Autorisierung MUSS eine übergebene `AuthenticationAssertion` für einen Versicherten (Das SAML:Assertion/SAML:AttributeStatement/SAML:Attribute `urn:gematik:subject:su`

bject-id enthält eine KVNR) dahingehend prüfen, ob die in der Behauptung urn:gematik:subject:authreference mit der serialNumber des zur Authentifizierung verwendeten AUT- bzw. AUT_ALT-Zertifikats in der Liste der bekannten AUT-Referenzen an der KeyChain des im RecordIdentifier benannten Aktenkontos ist und falls nicht,
MUSS die Komponente Autorisierung den Versicherten sowie im Vertretungsfall zusätzlich den Vertreter über die Nutzung eines neuen Authentisierungsmittels in einer E-Mail-Nachricht an die hinterlegte E-Mailadresse NotificationInfo des Versicherten bzw. des Vertreters informieren. Anschließend MUSS die benannte serialNumber in die WhiteList der AUT-Referenzen an der KeyChain des im RecordIdentifier benannten Aktenkontos übernommen werden.

[<=]

Nutzt der Versicherte ein im Aktensystem bisher unbekanntes Authentisierungsmittel (z.B. eine Folge-eGK) erhält er eine E-Mailbenachrichtigung, der Anwendungsfall wird nicht unterbrochen. Es obliegt dem Versicherten die Legitimität des Zugriffs bzw. des Authentisierungsmittels zu prüfen und sich gegebenenfalls mit dem ePA-Aktenanbieter und seiner Kasse in Verbindung zu setzen.

Nutzt der Vertreter des Versicherten ein bisher unbekanntes Authentisierungsmittel, erhalten sowohl der Versicherte als auch der Vertreter eine Benachrichtigung.

A_17655 - Komponente Autorisierung – Prüfung von Identitätsbestätigungen des Aktensystems
Die Komponente Autorisierung MUSS sicherstellen, dass Identitätsbestätigungen für Versicherte nur akzeptiert werden, wenn das zugehörige Signaturzertifikat zeitlich gültig ist, nicht gesperrt wurde und nach dem Zertifikatsprofil C.FD.SIG auf die Identität der Komponente Authentisierung Versicherter ausgestellt wurde.
[<=]

Dies kann durch eine aktuell gehaltene Konfiguration vertrauenswürdiger Zertifikate umgesetzt werden und ersetzt eine detaillierte Prüfung des Signaturzertifikats gemäß [gemSpec_TBAuth#A_15557], um die Prüfung solcher vom ePA-Aktensystem selbst ausgestellten Identitätsbestätigungen zu vereinfachen.

Eine Prüfung von Identitätsbestätigungen gemäß den Festlegungen für TBAuth bezieht sich auf Identitätsbestätigungen für Leistungserbringerinstitutionen und Kostenträger. . .

A_14270 - Komponente Autorisierung - Zugriff aus der Umgebung des Versicherten
Die Komponente Autorisierung MUSS Zugriffe auf Daten eines Versicherten aus der Personal Zone heraus verhindern, wenn das verwendete Gerät des Versicherten nicht in der Liste der bekannten/freigeschalteten Geräte vorhanden ist.**[<=]**

Bei Zugriffen aus der Umgebung des Versicherten wird ein Identitätsmerkmal des verwendeten Geräts abgefragt (DeviceID). Bei Zugriffen aus der Umgebung der Leistungserbringer erfolgt dies nicht, da hier als zugreifende Geräte ausschließlich zugelassene Konnektoren mit geprüfter Fachlogik zum Einsatz kommen. Ebenso wird keine Geräteidentität für den Zugang der Kostenträger über ihr jeweiliges Rechenzentrum geprüft, da auch hier ausschließlich zugelassene Produkttypen in einer kontrollierten Betriebsumgebung zum Einsatz kommen.

A_14402 - Komponente Autorisierung - Integritätsschutz für Autorisierungsbestätigungen

Die Komponente Autorisierung MUSS jede ausgestellte Autorisierungsbestätigung mit dem privaten Schlüssel der Ausstelleridentität C.FD.SIG in seiner fachlichen Rolle oid_epa_authz gemäß [gemSpec_OID] signieren.**[**<=**]**

A_14740 - Komponente Autorisierung - TLS-Identität innerhalb der TI
Die Komponente Autorisierung MUSS sich beim TLS-Verbindungsaufbau an den Schnittstellen innerhalb der TI mit der technischen Rolle oid_epa_authz der TLS-Identität C.FD.TLS-S authentisieren.**[**<=**]**

A_14529 - Komponente Autorisierung - Absicherung gegenüber dem Internet
Die Komponente Autorisierung MUSS alle Operationsaufrufe der Schnittstellen I_Authorization_Insurant und I_Authorization_Management_Insurant auf Wohlgeformtheit und Zulässigkeit gemäß Protokoll SOAP 1.2 prüfen und bei Schema-, Semantik- oder Protokollverletzungen eine aufgerufene Operation mit dem HTTP-Statuscode 400 gemäß [RFC-7231] abbrechen.**[**<=**]**

Die Prüfung der eingehenden Nachrichten auf Syntax-, Semantik- und Protokollverletzungen soll insbesondere den Angriffstypen *XML Injection*, *XPath Query Tampering* und *XML External Entity Injection* entgegenwirken.

Im Fall der Sperrung der Signaturidentität der Komponente Autorisierung, darf diese nicht für die Ausstellung einer Autorisierungsbestätigung genutzt werden. Da diese Identität aus dem gleichen Vertrauensraum stammt wie die Signaturidentität der Identitätsbestätigung eines Authentisierungsdienstes im gleichen Aktensystem, dürfen in diesem Fall auch keine Identitätsbestätigungen des gleichen Vertrauensraums mehr akzeptiert werden.

A_16260 - Komponente Autorisierung - Periodische Prüfung Signaturidentität
Die Komponente Autorisierung MUSS den Sperrstatus der eigenen Signaturidentität C.FD.SIG mittels [gemSpec_PKI#TUC_PKI_018] periodisch (einmal täglich) prüfen:

Parameter	Belegung
Zertifikat	Signaturzertifikat C.FD.SIG der Komponente Autorisierung
PolicyList	oid_fd_sig
intendedKeyUsage	digitalSignature
intendedExtendedKeyUsage	(leer)
OCSP-Graceperiod	60 Minuten
Offline-Modus	nein
Prüfmodus	OCSP

Das Signaturzertifikat muss anhand der Zertifikatsprüfung für [mathematisch gültig UND zeitlich gültig UND online gültig] befunden werden.**[**<=**]**

A_16261 - Komponente Autorisierung - Keine Autorisierung bei gesperrter Signaturidentität

Die Komponente Autorisierung MUSS das Ausstellen einer Autorisierungsbestätigung mit dem Fehler INTERNAL_ERROR abbrechen, wenn das Signaturzertifikat der Komponente Autorisierung gemäß einer Statusprüfung nach [A_16260] nicht gültig ist.**[<=]**

A_16262 - Komponente Autorisierung - Keine Identitätsbestätigung bei gesperrter Signaturidentität
Die Komponente Autorisierung MUSS alle Identitätsbestätigungen aller Issuer des gleichen Vertrauensraums der Signaturidentität C.FD.SIG der Komponente Autorisierung mit dem Fehler INTERNAL_ERROR als ungültig ablehnen, wenn das Signaturzertifikat der Komponente Autorisierung gemäß einer Statusprüfung nach [A_16260] nicht gültig ist.
[<=]

5.2 Verwendete Standards

Für die Sicherstellung der Interoperabilität wird auf verwendete Standards zurückgegriffen.
Durch die Verwendung des IHE-Frameworks (Integrating the Healthcare Enterprise) zum einheitlichen Datenaustausch im Gesundheitssystem ist die Verwendung von SAML zum Austausch von Authentisierungsinformationen notwendig.

Für die Übertragung von Nachrichten zwischen dem Fachmodul und den Teilkomponenten von ePA wird das vom W3C standardisierte Protokoll SOAP 1.2 in Verbindung mit HTTP verwendet.

A_13801 - Komponente Autorisierung - Verwendung von SAML 2.0
Die Komponente Autorisierung MUSS Authentisierungsbestätigung im Format SAML 2.0 Assertions [SAML2.0] unterstützen.
[<=]

A_13802 - Komponente Autorisierung - Ausstellung im Format SAML 2.0
Die Komponente Autorisierung MUSS Autorisierungsbestätigungen im Format SAML 2.0 Assertions [SAML2.0] ausstellen.**[<=]**

A_14969 - Komponente Autorisierung - Kodierung in UTF-8
Die Komponente Autorisierung MUSS bei der Erstellung von XML-Fragmenten das Encoding UTF-8 verwenden.
[<=]

A_17760 - Komponente Autorisierung - AuthenticationAssertion im SOAP-Header
Die Komponente Autorisierung MUSS die Identitätsbestätigungen eines Nutzers (AuthenticationAssertion) im Header eines eingehenden SOAP-Requests akzeptieren.
[<=]

A_17761 - Komponente Autorisierung - Verwendung des SAML Token Profile 1.1 für Web Services Security bei SAML 2.0 Assertions
Die Komponente Autorisierung MUSS die Anforderungen aus [WSS-SAML] umsetzen, wenn eine SAML 2.0 Assertion Teil einer SOAP 1.2-Eingangsnachricht ist.
[<=]

A_17762 - Komponente Autorisierung - Verwendung von SOAP Message Security 1.1

Die Komponente Autorisierung MUSS die Sicherheitsanforderungen aus SOAP Message Security 1.1 [WSS] für die Verarbeitung von SOAP 1.2-Nachrichten umsetzen.
[<=]

A_17763 - Komponente Autorisierung - Unterstützung von Profilen der Web Services Interoperability Organization (WS-I)
Die Komponente Autorisierung MUSS das WS-I Basic Profile V2.0 [WSIBP], das WS-I Basic Security Profile Version V1.1 [WSIBSP] sowie das WS-I Attachment Profile V1.0 [WSIAP] für die Kommunikation über Web Services berücksichtigen.
[<=]

5.3 Protokollierung

Die Anforderungen an die Protokollierung für die Komponente Autorisierung leiten sich aus dem Konzept der Protokollierung aus [gemSysL_ePA#2.5.5] ab.

A_14403 - Komponente Autorisierung - Verwaltungsprotokollierung Autorisierung
Die Komponente Autorisierung MUSS beim Aufruf einer der folgenden Operationen:

- I_Authorization_Insurant::getAuthorizationKey
- I_Authorization_Management::putAuthorizationKey
- I_Authorization_ManagementI_Authorization_Management_Insurant::putAuthorizationKey
- I_Authorization_Management_Insurant::deleteAuthorizationKey
- I_Authorization_Management_Insurant::replaceAuthorizationKey
- I_Authorization_Management_Insurant::getAuditEvents
- I_Authorization_Management_Insurant::putNotificationInfo
- I_Authorization_Management_Insurant::getAuthorizationList

je einen Eintrag im Verwaltungsprotokoll für den Versicherten gemäß [gemSpec_DM_ePA#A_14471] mit folgenden vom Operationsaufruf abhängigen Parameterwerten vornehmen: `UserID`, `UserName`, `ObjectID`, `ObjectName`, `DeviceID`.

[<=]

Der Aufruf der Operation `I_Authorization::getAuthorizationKey` aus der Umgebung der Leistungserbringer und der Kostenträger wird nicht protokolliert.

A_14427 - Komponente Autorisierung - Verwaltungsprotokollierung Gerät hinzufügen
Die Komponente Autorisierung MUSS beim Hinzufügen eines Geräts in die Liste der registrierten Geräte einen Eintrag im Verwaltungsprotokoll für den Versicherten vornehmen.**[<=]**

A_15753 - Komponente Autorisierung - Verwaltungsprotokollierung E-Mail-Adresse ändern
Die Komponente Autorisierung MUSS das manuelle Ändern der Benachrichtigungsadresse (z.B. durch den Anbieter im Supportfall) im Verwaltungsprotokoll des Versicherten protokollieren.**[<=]**

A_15754 - Komponente Autorisierung - Verwaltungsprotokollierung AuthorizationKey ändern
Die Komponente Autorisierung MUSS das manuelle Ändern eines AuthorizationKey in der KeyChain eines Kontos des Versicherten (z.B. durch den Anbieter im Supportfall) im Verwaltungsprotokoll des Versicherten protokollieren.**[<=]**

A_14188 - Komponente Autorisierung - Umfang Verwaltungsprotokoll
Die Komponente Autorisierung MUSS dem Versicherten oder berechtigten Vertreter die Einträge des Verwaltungsprotokolls gemäß der Festlegung in [gemSpec_DM_ePA#A_14471] übergeben:

Tabelle 2: Parameter des Verwaltungsprotokolls

Protokoll-parameter	Parameterwerte gemäß aufgerufener Operation
UserID	Wert des AttributStatements der im Operationsaufruf übergebenen AuthenticationAssertion, welches in `SAML:Assertion/SAML:AttributeStatement/SAML:Attribute/@Name` als AuthenticationAssertion für Leistungserbringer und Versicherte mit „`urn:gematik:subject:subject-id`“ bzw. als AuthenticationAssertion für Leistungserbringerinstitutionen und Kostenträger mit "`urn:gematik:subject:organization-id`" benannt ist.
UserName	Wert aus `SAML:Assertion/SAML:Subject/SAML:NameID` der im Operationsaufruf übergebenen AuthenticationAssertion
ObjectID	`RecordIdentifier`-Parameter des Operationsaufrufs *Hinweis: Bei Aufruf von Operationen ohne diesen Parameter wird der Wert im Protokolleintrag nicht belegt.*
ObjectName	`ActorID` des im Operationsaufruf gelesenen, gespeicherten oder geänderten AuthorizationKey *Hinweis: Bei Aufruf von Operationen ohne Bezug zu einem AuthorizationKey wird der Wert im Protokolleintrag nicht belegt (z.B. getAuditEvents).*
DeviceID	`DeviceID`-Parameter `DeviceIdType::Displayname` des Operationsaufrufs *Hinweis: Bei Aufruf der Operationen der Schnittstelle I_Authorization_Management gibt es den Parameter nicht, DeviceID wird im Protokolleintrag demzufolge nicht belegt.*

[<=]

A_14189 - Komponente Autorisierung - Protokollierung Schutz vor Manipulation
Die Komponente Autorisierung MUSS sicherstellen, dass die Verwaltungsprotokolldaten gegen Veränderung und unberechtigtes Löschen geschützt sind.
[<=]

A_14404 - Komponente Autorisierung - Löschen von Protokolleinträgen
Die Komponente Autorisierung MUSS für jeden bekannten RecordIdentifier Protokolleinträge des Verwaltungsprotokolls - außer den 50 jüngsten Einträgen - am Ende des auf ihre Generierung folgenden Kalenderjahres löschen.**[<=]**

5.4 Fehlerbehandlung in Schnittstellenoperationen

Bei Fehlern in der internen Verarbeitung oder fachlichen Fehlern in der Nutzung der von der Komponente Autorisierung bereitgestellten Schnittstellen werden Operationsaufrufe mit gematik-Fehlermeldungen gemäß der Definition in [gemSpec_OM] beantwortet. Die Fehlermeldungen werden als SOAP-Fault gemäß [TelematikError.xsd] strukturiert. Abweichend von den Festlegungen in [gemSpec_OM] sind zu meldende Fehler wie folgt mit Informationen zu füllen.

A_15068 - Komponente Autorisierung - Fehlername
Die Komponente Autorisierung MUSS in einer GERROR-Fehlermeldung gemäß [TelematikError.xsd] den in der Operationsdefinition festgelegten Fehlernamen `Name` im Feld `tel:Error/tel:Trace/tel:EventID` verwenden.**[<=]**

Die folgende Abbildung illustriert das Schema der GERROR-Struktur in TelematikError.xsd:

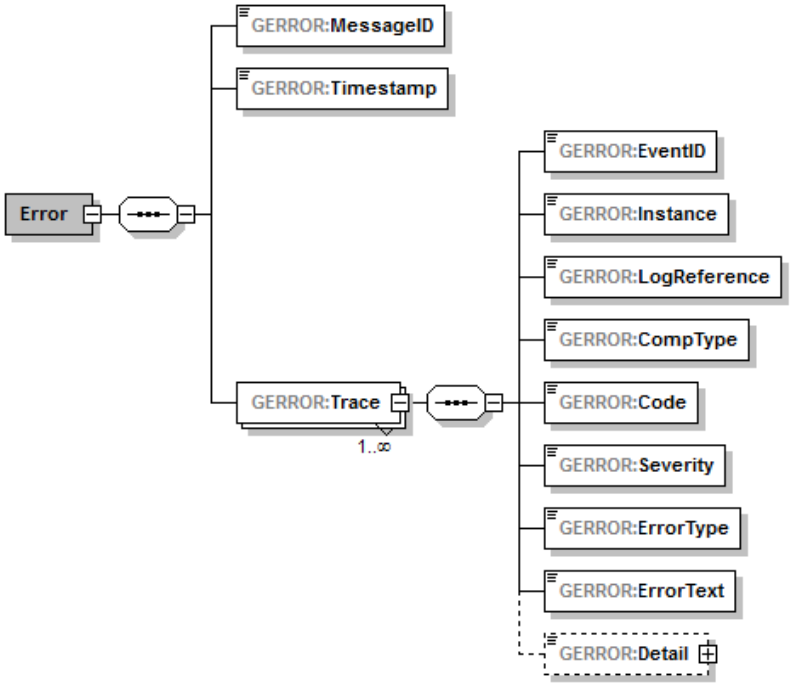


Abbildung 3: GERROR-Struktur zur Rückgabe einer Fehlermeldung

A_15069 - Komponente Autorisierung - Fehlertext
Die Komponente Autorisierung MUSS in einer GERROR-Fehlermeldung gemäß [TelematikError.xsd] den in der Operationsdefinition festgelegten Fehlerdetailtext `Fehlertext` im Feld `tel:Error/tel:Trace/tel:ErrorText` verwenden.**[<=]**

A_15101 - Komponente Autorisierung - Fehlernummer
Die Komponente Autorisierung MUSS in einer GERROR-Fehlermeldung gemäß [TelematikError.xsd] die folgenden Fehlercodes im Feld `tel:Error/tel:Trace/tel:Code` verwenden:

Tabelle 3: Fehlercodes zu Fehlern gemäß Operationsdefinition

Name	Fehlercode
TECHNICAL_ERROR	7900
KEY_ERROR	7910
SYNTAX_ERROR	7930
ASSERTION_INVALID	7940
DEVICE_UNKNOWN	7950
ACCESS_DENIED	7960
AUTHORIZATION_ERROR	7970
REPRESENTATIVE_ PENDING	7980

[<=]

Die Operationsdefinitionen der Schnittstellen der Komponente Autorisierung beschränken die Liste möglicher Fehler auf fachliche Fehler. Daneben sind weitere, technische Gründe für Fehler anderer Art denkbar. Für diese kann der Hersteller der Komponente einen generischen Fehler für den Transport geeigneter Fehlerinformationen (z.B. für Supportzwecke) verwenden.

A_15102 - Komponente Autorisierung - Herstellerspezifische Fehlermeldungen
Die Komponente Autorisierung MUSS komponenteninterne und herstellerspezifische Fehlermeldungen in einer GERROR-Fehlermeldung gemäß [TelematikError.xsd] mit folgender Festlegung transportieren:

Tabelle 4: Herstellerspezifische Fehlerdefinition

GERROR-Element	Herstellerspezifisch zu belegen
tel:Error/tel:Trace/tel:Code	Fester Wert: "7900"
tel:Error/tel:Trace/tel:EventID	Fester Wert: "TECHNICAL_ERROR"
tel:Error/tel:Trace/tel:ErrorText	Je Fehlerfall zufällig gewählte Fehlernummer

[<=]

A_15249 - Komponente Autorisierung - Herstellerspezifische Fehlermeldungen Detailtext
Die Komponente Autorisierung MUSS Details zu herstellerspezifischen Fehlermeldungen ausschließlich in einem internen Fehlerprotokoll und zusammen mit der zum Zeitpunkt des Fehlers gewählten zufälligen Fehlernummer speichern.**[<=]**

Die herstellerspezifische und je Fehlerfall zufällig gewählte Fehlernummer dient der Kapselung von Implementierungs- und Fehlerbehebungsdetails und zum Auffinden der Fehlermeldungsdetails in einem internen Fehlerprotokoll im Supportfall.

5.5 Nicht-Funktionale Anforderungen

5.5.1 Skalierbarkeit

Die für die Komponente Autorisierung relevanten Informationen zur Skalierbarkeit sind in [gemSpec_Perf] zu entnehmen.

5.5.2 Performance

Die durch die Komponente Autorisierung zu erfüllende Performance-Anforderung befinden sich in [gemSpec_Perf].

5.5.3 Mengengerüst

Das für die Komponente Autorisierung relevante Mengengerüst befindet sich in [gemSpec_Perf].

6 Funktionsmerkmale

Die Komponente Autorisierung realisiert die Funktionsmerkmale der kryptografischen Autorisierung und eine Geräteverwaltung. Das Funktionsmerkmal der Autorisierung wird über die Implementierung der Schnittstellen `I_Authorization`, `I_Authorization_Management`, `I_Authorization_Insurant` und `I_Authorization_Management_Insurant` realisiert.

Die Nutzung des Funktionsmerkmals der Geräteverwaltung durch den Versicherten erfolgt über einen separaten Verwaltungszugang abseits der `I_Authorization`*-Schnittstellen. Dieser Zugang ist für den Versicherten über das Internet erreichbar.

6.1 Übergreifende Festlegungen

Im Folgenden werden übergreifende Festlegungen formuliert, die in allen Operationen umgesetzt werden.

Wenn im Folgenden die KVNR als ActorID, OwnerKVNR oder subject-id referenziert wird ist immer der unveränderliche Anteil als 10-stellige Kennung gemeint.

A_14469 - Komponente Autorisierung - Identifizierung des Versicherten anhand einer AuthenticationAssertion
Die Komponente Autorisierung MUSS jeden Versicherten anhand des unveränderlichen Teils der KVNR als `urn:gematik:subject:subject-id` in `SAML:Assertion/SAML:AttributeStatement/SAML:Attribute/@Name` einer übergebenen, gültigen AuthenticationAssertion eindeutig identifizieren, wenn die `subject-id` mit der OwnerKVNR zu einem im Operationsaufruf angegebenen RecordIdentifier übereinstimmt.

[<=]

A_14499 - Komponente Autorisierung - Identifizierung einer Institution anhand einer AuthenticationAssertion
Die Komponente Autorisierung MUSS jede Leistungserbringerinstitution und jeden Kostenträger anhand der Telematik-ID als `urn:gematik:subject:organization-id` in `SAML:Assertion/SAML:AttributeStatement/SAML:Attribute/@Name` einer übergebenen, gültigen AuthenticationAssertion eindeutig identifizieren, wenn für diese ein AuthorizationKey zu einem im Operationsaufruf angegebenen RecordIdentifier existiert.

[<=]

A_14500 - Komponente Autorisierung - Identifizierung eines Vertreters anhand einer AuthenticationAssertion
Die Komponente Autorisierung MUSS einen berechtigten Vertreter anhand seiner KVNR als `urn:gematik:subject:subject-id` in `SAML:Assertion/SAML:AttributeStatement/SAML:Attribute/@Name` einer übergebenen, gültigen AuthenticationAssertion eindeutig identifizieren, wenn

die `subject-id` ungleich der OwnerKVNR zu einem im Operationsaufruf angegebenen RecordIdentifier ist und für die KVNR der AuthenticationAssertion ein AuthorizationKey zu der im Operationsaufruf angegebenen RecordIdentifier existiert.

[<=]

A_14434 - Komponente Autorisierung - Prüfung der Schnittstellenparameter
Die Komponente Autorisierung MUSS in jeder Operation alle übergebenen Eingangsparameter auf Konformität zum Schema AuthorizationService.xsd prüfen und bei Nichtkonformität die jeweilige Operation mit dem Fehler TECHNICAL_ERROR gemäß den Festlegungen zur Fehlerbehandlung abbrechen.
[<=]

A_14369 - Komponente Autorisierung - Prüfung des Geräts des Versicherten
Die Komponente Autorisierung MUSS in allen Operationen der Schnittstellen `I_Authorization_Insurant` und `I_Authorization_Management_Insurant` anhand des Wertes `DeviceID::Device` prüfen, ob das vom Nutzer verwendete Gerät in der Geräteliste des AuthorizationKeys des Nutzers bekannt/freigeschaltet ist und andernfalls die Operation mit dem Fehler DEVICE_UNKNOWN abbrechen, in dessen SOAP-Error in `tel:Error/tel:Trace/tel:ErrorText` eine gemäß [gemSpec_Autorisierung#A_17866] generierte `phr:DeviceID::Device` einfügen und den Freischaltprozess neuer Geräte auslösen.
[<=]

Greift ein Nutzer mit einem Gerät erstmalig auf die in A_14369 genannten Schnittstellen zu, sind die Elemente phr:DeviceID@ und phr:DeviceID::Device in den aufgerufenen Operationen ggfs. leer bzw. enthalten eine Zeichenkette der Länge 0 ("").

A_14634 - Komponente Autorisierung - Prüfung auf vorhandenen AuthorizationKey
Die Komponente Autorisierung MUSS eine aufgerufene Operationen mit dem Standardfehler KEY_ERROR abbrechen, wenn es zu fachlichen Fehlern in Lese- oder Schreiboperationen eines AuthorizationKey kommt oder dieser für einen in der ActorID benannten Nutzer in der KeyChain eines benannten RecordIdentifier nicht vorhanden ist.**[<=]**

A_14768 - Komponente Autorisierung - Prüfung auf Berechtigung
Die Komponente Autorisierung MUSS eine aufgerufene Operation mit dem Standardfehler ACCESS_DENIED abbrechen, wenn ein über die `subject-id` bzw. `organization-id` einer AuthenticationAssertion identifizierter Nutzer eine Operation auf einem im RecordIdentifier benannten Datensatz aufruft, für den kein AuthorizationKey hinterlegt und er nicht der Eigentümer ist, d.h. `OwnerKVNR != subject-id` bzw. `organization-id` und es existiert kein AuthorizationKey mit `ActorID == subject-id` bzw. `organization-id`.**[<=]**

Der Fehler ACCESS_DENIED wird ebenso erwartet, wenn im jeweiligen Aufrufparameter ein RecordIdentifier mit einer falschen HomeCommunityID übergeben wird.

A_16487 - Komponente Autorisierung - Prüfung auf Tokenherkunft
Die Komponente Autorisierung MUSS jeden Aufruf an den Schnittstellen *I_Authorization_Insurant* und *I_Authorization_Management_Insurant* mit dem Fehler

ACCESS_DENIED ablehnen, der mittels einer AuthenticationAssertion erfolgt, die nicht aus dem Vertrauensraum der Komponente Autorisierung erfolgt.**[<=]**

A_15620 - Komponente Autorisierung - Read-only bei suspendiertem Konto
Die Komponente Autorisierung MUSS die folgenden Operationen mit dem Standardfehler ACCESS_DENIED abbrechen,
wenn der RecordState der KeyChain des im Aufrufparameter der Operation benannten `RecordIdentifier` den Zustand `SUSPENDED` ausweist:

- I_Authorization_Management::putAuthorizationKey
- I_Authorization_ManagementI_Authorization_Management_Insurant::putAuthorizationKey
- I_Authorization_Management_Insurant::deleteAuthorizationKey
- I_Authorization_Management_Insurant::replaceAuthorizationKey
- I_Authorization_Management_Insurant::putNotificationInfo

[<=]

A_17102 - Komponente Autorisierung - Maximale Berechtigungsstufe für Konto-Eigentümer
Die Komponente Autorisierung MUSS sicherstellen, dass der AuthorizationType am hinterlegten AuthorizationKey des Versicherten immer "DOCUMENT_AUTHORIZATION" lautet.
[<=]

Damit soll verhindert werden, dass ein zur Umschlüsselung berechtigter Vertreter fälschlich einen ungültigen oder einschränkenden AuthorizationKey für den Versicherten hinterlegt. Dies berührt nicht die Ausstellung einer AuthorizationAssertion mit ACCOUNT_AUTHORIZATION für den Fall eines nicht vorhandenen AuthorizationKey bei Kontoaktivierung/-umzug.

6.2 Schnittstellen der Komponente Autorisierung

Das Funktionsmerkmal 'Autorisierung' der Komponente Autorisierung wird durch die in der folgenden Tabelle beschriebenen Schnittstellen mit den jeweiligen Operationen umgesetzt.

Tabelle 5: Schnittstellen der Komponente Autorisierung

Schnittstellen der Komponente Autorisierung	
I_Authorization	
`getAuthorizationKey`	Mit der Operation `getAuthorizationKey` wird das für einen Berechtigten verschlüsselte Schlüsselmaterial für ein konkretes Aktenkonto eines Versicherten in der Leistungserbringer-Umgebung und durch den Kostenträger heruntergeladen.
I_Authorization_Management	
`putAuthorizationKey`	Mit der Operation `putAuthorizationKey` wird das für einen Berechtigten verschlüsselte Schlüsselmaterial für ein konkretes Aktenkonto eines Versicherten im Aktensystem ePA gespeichert.
`checkRecordExists`	Mit der Operation `checkRecordExists` kann ein anderer Anbieter bei einem Anbieter einer Aktenlösung den Status und die Existenz eines Aktenkontos über die KVNR eines Versicherten abfragen.
`getAuthorizationList`	Die Operation `getAuthorizationList` liefert die Liste aller OwnerKVNRs des Aktensystems, in denen für die anfragende Institution ein AuthorizationKey hinterlegt ist. (horizontale Abfrage)
I_Authorization_Insurant	
`getAuthorizationKey`	Mit der Operation `getAuthorizationKey` wird das für einen Berechtigten verschlüsselte Schlüsselmaterial (kryptografische Berechtigung) für ein konkretes Aktenkonto eines Versicherten in der Personal-Zone heruntergeladen.
I_Authorization_Management_Insurant	
`putAuthorizationKey`	Mit der Operation `putAuthorizationKey` wird das für einen Berechtigten verschlüsselte Schlüsselmaterial AuthorizationKey für ein konkretes Aktenkonto eines Versicherten im ePA-Aktensystem gespeichert.
`deleteAuthorizationKey`	Mit der Operation `deleteAuthorizationKey` kann ein authentifizierter Versicherter bzw. ein berechtigter Vertreter die kryptografische Berechtigung für einen Nutzer innerhalb seines Aktenkontos löschen.

`replaceAuthorizationKey`	Mit der Operation `replaceAuthorizationKey` kann ein authentifizierter Versicherter bzw. ein berechtigter Vertreter das für eine alte eGK verschlüsselte Schlüsselmaterial durch neues für eine Folgekarte verschlüsseltes Schlüsselmaterial ersetzen.
`getAuditEvents`	Mit der Operation `getAuditEvents` kann ein authentifizierter Versicherter bzw. ein berechtigter Vertreter das Verwaltungsprotokoll der Komponente Autorisierung auslesen.
`putNotificationInfo`	Mit der Operation `putNotificationInfo` kann ein authentifizierter Versicherter bzw. ein berechtigter Vertreter die eigene, im Benachrichtigungskanal hinterlegten Daten aktualisieren.
`getAuthorizationList`	Die Operation `getAuthorizationList` liefert die Liste aller AuthorizationKeys zu einer angefragten Akte eines Versicherten. (vertikale Abfrage)

6.2.1 Schnittstelle I_Authorization

Diese Schnittstelle setzt die in [gemSysL_Fachanwendung_ePA#4.2.2.2] definierte Schnittstelle `I_Authorization` technisch um.

Die Schnittstelle stellt dem Fachmodul eine Operation zum Bezug eines Autorisierungs-Tokens für bereits authentifizierte Leistungserbringer und Kostenträger bereit, um die ePA-Komponente Dokumentenverwaltung verwenden zu können.

6.2.1.1 Operationsdefinition I_Authorization::getAuthorizationKey

A_14045 - Komponente Autorisierung - I_Authorization::getAuthorizationKey
Die Komponente Autorisierung MUSS die Operation `I_Authorization::getAuthorizationKey` gemäß der folgenden Signatur implementieren:

Tabelle 6: I_Authorization::getAuthorizationKey Definition

Operation	**I_Authorization::getAuthorizationKey**
Beschreibung	Mit dieser Operation wird für einen authentifizierten Nutzer eine Autorisierung des Zugriffs auf Daten eines Versicherten geprüft.
Formatvorgaben	Die Definition der Ein- und Ausgabeparameter erfolgt in [AuthorizationService.xsd]. Diejenigen Parameter, die im XSD-Schema optional gekennzeichnet, aber hier nicht aufgeführt sind, werden in der Operation an dieser Schnittstelle nicht verwendet.

Eingangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**
`AuthenticationAssertion`	Die `AuthenticationAssertion` ist eine von einem Identitiy Provider ausgestellte Authentifizierungsbestätigung für einen Nutzer.	SAML Assertion im SOAP-Header des Requests	-
`RecordIdentifier`	Der `RecordIdentifier` referenziert ein konkretes Aktenkonto eines Versicherten bei einem Anbieter. Mit diesem wird der Datensatz der kryptografischen Autorisierung in der Komponente Autorisierung für den anfragenden Nutzer lokalisiert.	String	-
Ausgangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**
`AuthorizationAssertion`	Die `AuthorizationAssertion` ist eine signierte Autorisierungsbestätigung für einen Nutzer und enthält Informationen über die Art und den Umfang der in der Komponente Autorisierung hinterlegten Autorisierung.	SAML Assertion base64-codiert	-
`AuthorizationKey`	Die kryptografische Autorisierung eines Nutzers.	AuthorizationKeyType	ja
Fehlermeldungen			
Name	**Fehlertext**	**Details**	
TECHNICAL_ERROR	Zufallszahl		
ASSERTION_INVALID	Authentifizierungsbestätigung ungültig	Die Authentifizierungsbestätigung des aufrufenden Nutzers wird nicht akzeptiert.	
ACCESS_DENIED	Zugriff verweigert	Die Operation ist mit den angegebenen Parametern nicht zulässig.	

KEY_ERROR	Fehler im Schlüsseldatensatz	Kein Datensatz für diesen Nutzer für den benannten RecordIdentifier vorhanden.
REPRESENTATIVE_PENDING	Vertretungsberechtigung erfordert Freischaltung	Die Vertretung kann erst wahrgenommen werden, wenn diese über den Freischaltprozess autorisiert wurde.
AUTHORIZATION_ERROR	Autorisierung nicht zulässig	Die zu hinterlegte Berechtigtenrolle ist nicht zulässig.

Sollte bei der Autorisierung für einen Versicherten kein zugehöriger Datensatz gefunden werden, darf dies nicht mit einer technischen Fehlermeldung behandelt werden. Hierfür MUSS eine sprechende Information (fachliches Ereignis) geliefert werden.

[<=]

6.2.1.2 Umsetzung I_Authorization::getAuthorizationKey

Die folgenden Anforderungen beschreiben die Umsetzung der Operation `I_Authorization::getAuthorizationKey`. Dabei gelten die übergreifenden Festlegungen zur Prüfung der Eingangsparameter.

A_17790 - Komponente Autorisierung LE - Vertretung wahrnehmen Freischaltprüfung
Die Komponente Autorisierung MUSS bei Wahrnehmung einer Vertretung für einen Versicherten mittels `I_Authorization::getAuthorizationKey` (`subject-id` der AuthenticationAssertion != `OwnerKVNR`) vor der Herausgabe prüfen, ob ein wartender Vertreter-Freischaltprozess für [OwnerKVNR des benannten RecordIdentifiers, subject-id als ActorID] aktiv ist und falls ja, die Operation mit dem Fehler REPRESENTATIVE_PENDING abbrechen.

[<=]

A_13917 - Komponente Autorisierung LE - Ausstellen einer Autorisierungsbestätigung
Die Komponente Autorisierung MUSS in der Operation `I_Authorization::getAuthorizationKey` bei Vorhandensein eines *AuthorizationKey* in der *KeyChain* des benannten `RecordIdentifier` für den mittels `AuthenticationAssertion` authentifizierten Nutzer (`subject-ID` bzw. `organization-id == ActorID`) eine *AuthorizationAssertion* gemäß der Festlegung in [A_14491] ausstellen und diese in der Ausgangsnachricht der Operation zurückgeben.
Der Wert für [AuthorizationType] in der AuthorizationAssertion MUSS dem Wert des hinterlegten AuthorizationKey genau dieses authentifizierten Nutzers entsprechen.
[<=]

A_17662 - Komponente Autorisierung LE - Codierung der Autorisierungsbestätigung
Die Komponente Autorisierung MUSS die erstellte und signierte Autorisierungsbestätigung in der Response der Operation `I_Authorization::getAuthorizationKey` Base64-codiert zurückgeben. **[<=]**

A_13692 - Komponente Autorisierung LE - Herausgabe kryptografischer Berechtigung des Nutzers
Die Komponente Autorisierung MUSS in der Operation `I_Authorization::getAuthorizationKey` bei Vorhandensein eines *AuthorizationKey* in der *KeyChain* des benannten `RecordIdentifier` für den mittels `AuthenticationAssertion` authentifizierten Nutzer (`subject-ID` bzw. `organization-id == ActorID`) den `AuthorizationKey` in der Ausgangsnachricht der Operation zurückgeben.**[<=]**

A_14643 - Komponente Autorisierung LE - Aktivierung bei Kontoeröffnung in der Umgebung der Leistungserbringer
Die Komponente Autorisierung MUSS dem authentifizierten Versicherten als Eigentümer der Akte (`subject-ID == OwnerKVNR` für den benannten `RecordIdentifier`) eine Autorisierungsbestätigung mit AuthorizationType = ACCOUNT_AUTHORIZATION gemäß [A_14491] ausstellen, wenn für seine OwnerKVNR kein Schlüsseldatensatz AuthorizationKey in der KeyChain vorhanden ist.
[<=]

A_15618 - Komponente Autorisierung LE - Autorisierung bei suspendiertem Konto
Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization::getAuthorizationKey` bei Vorhandensein eines *AuthorizationKey* in der *KeyChain* des benannten `RecordIdentifier` für den mittels `AuthenticationAssertion` authentifizierten Nutzer (`subject-id` = *ActorID* des *AuthorizationKey*) eine Autorisierungsbestätigung mit AuthorizationType = ACCOUNT_AUTHORIZATION gemäß [A_14491] ausstellen, wenn der RecordState der KeyChain des benannten `RecordIdentifier` den Zustand `SUSPENDED` ausweist.**[<=]**

6.2.2 Schnittstelle I_Authorization_Insurant

Diese Schnittstelle setzt die in [gemSysL_ePA] definierte Schnittstelle `I_Authorization_Insurant` technisch um.

Die Schnittstelle `I_Authorization_Insurant` stellt Operationen zur Autorisierungsprüfung auf das Vorhandensein von kryptografischem Schlüsselmaterial für einen Nutzer des Aktenkontos eines Versicherten bereit. Sie stellt dem Frontend des Versicherten eine Schnittstelle zum Abruf eines Autorisierungs-Tokens für bereits authentifizierte Versicherte bereit.

6.2.2.1 Operationsdefinition I_Authorization_Insurant::getAuthorizationKey

A_14042 - Komponente Autorisierung - I_Authorization_Insurant::getAuthorizationKey

Die Komponente Autorisierung MUSS die Operation `I_Authorization_Insurant::getAuthorizationKey` gemäß der folgenden Signatur implementieren:

Tabelle 7: I_Authorization_Insurant::getAuthorizationKey Definition

Operation	**I_Authorization_Insurant::getAuthorizationKey**		
Beschreibung	Mit dieser Operation wird für einen authentifizierten Nutzer eine Autorisierung des Zugriffs auf Daten eines Versicherten geprüft.		
Formatvorgaben	Die Definition der Ein- und Ausgabeparameter erfolgt in [AuthorizationService.xsd]. Diejenigen Parameter, die im XSD-Schema optional gekennzeichnet, aber hier nicht aufgeführt sind, werden in der Operation an dieser Schnittstelle nicht verwendet.		
Eingangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**
`AuthenticationAssertion`	Die `AuthenticationAssertion` ist eine von einem Identitiy Provider ausgestellte Authentifizierungsbestätigung für einen Nutzer.	SAML Assertion im SOAP-Header des Requests	-
`RecordIdentifier`	Der `RecordIdentifier` referenziert ein konkretes Aktenkonto eines Versicherten bei einem Anbieter. Mit diesem wird der Datensatz der Autorisierung in der Komponente Autorisierung für den anfragenden Nutzer lokalisiert.	String	-
`DeviceID`	Die `DeviceID` enthält die Gerätekennung eines vom Nutzer verwendeten Geräts.	DeviceIdType	-
Ausgangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**

AuthorizationAssertion	Die `AuthorizationAssertion` ist eine signierte Autorisierungsbestätigung für einen Nutzer und enthält Informationen über die Art und den Umfang der in der Komponente Autorisierung hinterlegten Autorisierung.	SAML Assertion mit AuthorizationDecision Statement base 64-codiert	-
AuthorizationKey	Die kryptografische Autorisierung eines Nutzers.	AuthorizationKeyType	ja
Fehlermeldungen			
Name	**Fehlertext**	**Details**	
TECHNICAL_ERROR	Zufallszahl		
ASSERTION_INVALID	Authentifizierungsbestätigung ungültig	Die Authentifizierungsbestätigung des aufrufenden Nutzers wird nicht akzeptiert.	
ACCESS_DENIED	Zugriff verweigert	Die Operation ist mit den angegebenen Parametern nicht zulässig.	
KEY_ERROR	Fehler im Schlüsseldatensatz	Kein Datensatz für diesen Nutzer für den benannten RecordIdentifier vorhanden.	
DEVICE_UNKOWN	generierte `phr:DeviceID::Device`	Das vom Nutzer verwendete Gerät des Versicherten ist nicht bekannt und muss freigeschaltet werden.	
REPRESENTATIVE_PENDING	Vertretungsberechtigung erfordert Freischaltung	Die Vertretung kann erst wahrgenommen werden, wenn diese über den Freischaltprozess autorisiert wurde.	

Sollte bei der Autorisierung für einen Versicherten kein zugehöriger Datensatz gefunden werden, darf dies nicht mit einer technischen Fehlermeldung behandelt werden. Hierfür MUSS eine sprechende Information (fachliches Ereignis) geliefert werden.

[<=]

6.2.2.2 Umsetzung I_Authorization_Insurant::getAuthorizationKey

Die folgenden Anforderungen beschreiben die Umsetzung der Operation `I_Authorization_Insurant::getAuthorizationKey`. Dabei gelten die übergreifenden Festlegungen zur Prüfung der Eingangsparameter.

A_17789 - Komponente Autorisierung Vers. - Vertretung wahrnehmen Freischaltprüfung

Die Komponente Autorisierung MUSS bei Wahrnehmung einer Vertretung für einen Versicherten mittels `I_Authorization_Insurant::getAuthorizationKey` (`subject-id` der AuthenticationAssertion != `OwnerKVNR`) vor der Herausgabe prüfen, ob ein wartender Vertreter-Freischaltprozess für [OwnerKVNR des benannten RecordIdentifiers, subject-id als ActorID] aktiv ist und falls ja, die Operation mit dem Fehler REPRESENTATIVE_PENDING abbrechen.

[<=]

A_14436 - Komponente Autorisierung Vers. - Ausstellen einer Autorisierungsbestätigung

Die Komponente Autorisierung MUSS in der Operation `I_Authorization_Insurant::getAuthorizationKey` bei Vorhandensein eines *AuthorizationKey* in der *KeyChain* des benannten `RecordIdentifier` für den mittels `AuthenticationAssertion` authentifizierten Nutzer [subject-id der AuthenticationAssertion == ActorID des vorhandenen AuthorizationKey] eine *AuthorizationAssertion* gemäß der Festlegung in [A_14491] ausstellen und diese in der Ausgangsnachricht der Operation zurückgeben.
Der Wert für [AuthorizationType] in der AuthorizationAssertion MUSS dem Wert des hinterlegten AuthorizationKey genau dieses authentifizierten Nutzers entsprechen.
[<=]

A_17663 - Komponente Autorisierung Vers. - Codierung der Autorisierungsbestätigung

Die Komponente Autorisierung MUSS die erstellte und signierte Autorisierungsbestätigung in der Response der Operation `I_Authorization_Insurant::getAuthorizationKey` Base64-codiert zurückgeben.
[<=]

A_14439 - Komponente Autorisierung Vers. - Herausgabe kryptografischer Berechtigung des Nutzers

Die Komponente Autorisierung MUSS in der Operation `I_Authorization_Insurant::getAuthorizationKey` bei Vorhandensein eines *AuthorizationKey* in der *KeyChain* des benannten `RecordIdentifier` für den mittels `AuthenticationAssertion` authentifizierten Versicherten oder Vertreter (`subject-id == ActorID`) den `AuthorizationKey` des authentifizierten Nutzers in der Ausgangsnachricht der Operation zurückgeben.
[<=]

A_14644 - Komponente Autorisierung Vers. - Aktivierung bei Kontoeröffnung in der Umgebung des Versicherten
Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Insurant::getAuthorizationKey` dem authentifizierten Versicherten als Eigentümer der Akte (`subject-ID == OwnerKVNR` für den benannten `RecordIdentifier`) eine Autorisierungsbestätigung mit AuthorizationType = ACCOUNT_AUTHORIZATION gemäß [A_14491] ausstellen, wenn für seine OwnerKVNR kein Schlüsseldatensatz AuthorizationKey in der KeyChain vorhanden ist. **[<=]**

A_15619 - Komponente Autorisierung Vers. - Autorisierung bei suspendiertem Konto
Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Insurant::getAuthorizationKey` bei Vorhandensein eines *AuthorizationKey* in der *KeyChain* des benannten `RecordIdentifier` für den mittels `AuthenticationAssertion` authentifizierten Nutzer (`subject-id` = *ActorID* des *AuthorizationKey*) eine Autorisierungsbestätigung mit AuthorizationType = ACCOUNT_AUTHORIZATION gemäß [A_14491] ausstellen, wenn der RecordState der KeyChain des benannten `RecordIdentifier` den Zustand `SUSPENDED` ausweist.**[<=]**

6.2.3 Schnittstelle I_Authorization_Management

Diese Schnittstelle setzt die in [gemSysL_ePA] definierte Schnittstelle `I_Authorization_Management` technisch um.

Die Schnittstelle `I_Authorization_Management` dient dazu, kryptografische Berechtigungen im Autorisierungsdienst eines Aktensystems zu verwalten.

6.2.3.1 Operationsdefinition I_Authorization_Management::putAuthorizationKey

A_14180 - Komponente Autorisierung - I_Authorization_Management::putAuthorizationKey
Die Komponente Autorisierung MUSS die Operation `I_Authorization_Management::putAuthorizationKey` gemäß der folgenden Signatur implementieren:

Tabelle 8: I_Authorization_Management::putAuthorizationKey - Definition

Operation	**I_Authorization_Management::putAuthorizationKey**
Beschreibung	Mit der Operation wird das für einen Berechtigten verschlüsselte Schlüsselmaterial für ein konkretes Aktenkonto eines Versicherten im ePA-Aktensystem gespeichert.
Formatvorgaben	Die Definition der Ein- und Ausgabeparameter erfolgt in [AuthorizationService.xsd]. Diejenigen Parameter, die im XSD-Schema optional gekennzeichnet, aber hier nicht aufgeführt sind, werden in der Operation an dieser Schnittstelle nicht verwendet.
Eingangsparameter	

Name	Beschreibung	Typ	opt.
AuthenticationAssertion	Die AuthenticationAssertion ist eine von einem Identitiy Provider ausgestellte Authentifizierungsbestätigung für einen Nutzer.	SAML Assertion im SOAP-Header des Requests	-
RecordIdentifier	Der RecordIdentifier referenziert ein konkretes Aktenkonto eines Versicherten bei einem Anbieter. Mit diesem wird der Datensatz der Autorisierung in der Komponente Autorisierung für den anfragenden Nutzer lokalisiert.	String	-
AuthorizationKey	Die kryptografische Autorisierung eines Nutzers, bestehend aus Listen von verschlüsselten Schlüsseln. Details zur Struktur finden sich im Kapitel 7 zum Informationsmodell.	AuthorizationKeyType	-
Fehlermeldungen			
Name	**Fehlertext**	**Details**	
TECHNICAL_ERROR	Zufallszahl	.	
ASSERTION_INVALID	Authentifizierungsbestätigung ungültig	Die Authentifizierungsbestätigung des aufrufenden Nutzers wird nicht akzeptiert.	
ACCESS_DENIED	Zugriff verweigert	Die Operation ist mit den angegebenen Parametern nicht zulässig.	

[<=]

6.2.3.2 Umsetzung I_Authorization_Management::putAuthorizationKey

Die folgenden Anforderungen beschreiben die Umsetzung der Operation `I_Authorization_Management::putAuthorizationKey`. Dabei gelten die übergreifenden Festlegungen zur Prüfung der Eingangsparameter.

A_14212 - Komponente Autorisierung LE - Speicherung kryptografische Berechtigung des Nutzers

Die Komponente Autorisierung MUSS in der Operation `I_Authorization_Management::putAuthorizationKey` den im Eingangsparameter übergebenen `AuthorizationKey` als *AuthorizationKey* der KeyChain des im Eingangsparameter benannten `RecordIdentifier` speichern bzw. ersetzen, falls für die im `AuthorizationKey` benannte ActorID bereits ein *AuthorizationKey* in der KeyChain des benannten `RecordIdentifier` existiert.**[<=]**

A_14441 - Komponente Autorisierung LE - Berechtigungsprüfung Schlüsselhinterlegung

Die Komponente Autorisierung MUSS beim Aufruf der Operation `I_Authorization_Management::putAuthorizationKey` anhand der KVNR der `AuthenticationAssertion` und des `RecordIdentifier` prüfen, ob für den aufrufenden Nutzer ein AuthorizationKey mit `ActorID == subject-ID~~/organization-id~~` hinterlegt ist, und falls nicht, die Operation mit dem Fehler ACCESS_DENIED abbrechen.**[<=]**

Mit dieser Prüfung wird sichergestellt, dass nur Versicherte bzw. Vertreter einen Schlüssel für einen Berechtigten hinterlegen können. Eine Berechtigung wird nicht von einer Leistungserbringerinstitution oder von einem Kostenträger hinterlegt.

A_14587 - Komponente Autorisierung LE - Initiale Schlüsselhinterlegung Kontoeröffnung

Die Komponente Autorisierung MUSS die Operation `I_Authorization_Management::putAuthorizationKey` mit dem Fehler ACCESS_DENIED abbrechen, sofern für den Eigentümer der Akte noch kein AuthorizationKey vorhanden ist und der zu speichernde `AuthorizationKey` des Aufrufparameters für einen anderen Nutzer als den Eigentümer des `RecordIdentifier` (ActorID != OwnerKVNR) gespeichert werden soll.**[<=]**

Mit dieser Anforderung soll verhindert werden, dass die Akte genutzt wird, bevor das Schlüsselmaterial für den Versicherten erzeugt und hinterlegt wurde. Die benannte Konstellation liegt im Rahmen der Kontoeröffnung und bei einem Aktenumzug vor. Das Schlüsselmaterial für den Versicherten wird im Schritt der Kontoaktivierung erzeugt, welcher auf den Schritt der Kontoinitialisierung folgt.

A_14737 - Komponente Autorisierung LE - Initiale Schlüsselhinterlegung für den Versicherten

Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Management::putAuthorizationKey` durch den Versicherten (subject-id (KVNR) der `AuthenticationAssertion == OwnerKVNR`) im Rahmen der initialen Schlüsselhinterlegung während der Kontoaktivierung das `validTo`-Datum des übergebenen `AuthorizationKey` vor der Speicherung mit einem technischen Datum gleichbedeutend mit "unendlich" (z.B.*31.12.9999*) ersetzen.**[<=]**

A_14999 - Komponente Autorisierung LE - Zustandswechsel bei Schlüsselhinterlegung für den Versicherten

Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Management::putAuthorizationKey` durch den Versicherten (subject-id (KVNR) der `AuthenticationAssertion == OwnerKVNR`) bei erfolgreichem Abschluss der initialen Schlüsselhinterlegung für den Versicherten während der Kontoaktivierung den Zustand `RecordState` der KeyChain des Versicherten von `REGISTERED` auf den Wert `ACTIVATED` setzen.
[<=]

6.2.3.3 Operationsdefinition I_Authorization_Management::checkRecordExists

A_14965 - Komponente Autorisierung - I_Authorization_Management::checkRecordExists
Die Komponente Autorisierung MUSS die Operation `I_Authorization_Management::checkRecordExists` gemäß der folgenden Signatur implementieren:

Tabelle 9: I_Authorization_Management::checkRecordExists - Definition

Operation	**I_Authorization_Management::checkRecordExists**		
Beschreibung	Die Operation liefert den Status eines Aktenkontos eines via KVNR benannten Versicherten.		
Formatvorgaben	Die Definition der Ein- und Ausgabeparameter erfolgt in [AuthorizationService.xsd]. Diejenigen Parameter, die im XSD-Schema optional gekennzeichnet, aber hier nicht aufgeführt sind, werden in der Operation an dieser Schnittstelle nicht verwendet.		
Eingangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**
`KVNR`	Der unveränderliche Teil der Krankenversicherungsnummer eines gesetzlich Versicherten	String	-
Ausgangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**
`RecordState`	Statuswert zur Existenz eines Aktenkontos in der Komponente Autorisierung zu einer angefragten KVNR	RecordStateType	-
Fehlermeldungen			
Name	**Fehlertext**	**Details**	
TECHNICAL_ERROR	Zufallszahl		

[<=]

6.2.3.4 Umsetzung I_Authorization_Management::checkRecordExists

Die folgenden Anforderungen beschreiben die Umsetzung der Operation `I_Authorization_Management::checkRecordExists`. Dabei gelten die übergreifenden Festlegungen zur Prüfung der Eingangsparameter.

A_14966 - Komponente Autorisierung LE - Abfrage Aktenexistenz
Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Management::checkRecordExists` den Wert des `RecordState` des Datensatzes KeyChain eines Konto zurückliefern, wenn zu einer angefragten `KVNR` ein Datensatz KeyChain mit `OwnerKVNR == KVNR` existiert und andernfalls den Statuswert `UNKNOWN` zurückgeben.**[<=]**

6.2.3.5 Operationsdefinition I_Authorization_Management::getAuthorizationList

A_17110 - Komponente Autorisierung - I_Authorization_Management::getAuthorizationList
Die Komponente Autorisierung MUSS die Operation `I_Authorization_Management::getAuthorizationList` gemäß der folgenden Signatur implementieren:

Tabelle 10: I_Authorization_Management::getAuthorizationList - Definition

Operation	**I_Authorization_Management::getAuthorizationList**		
Beschreibung	Die Operation liefert eine Liste der OwnerKVNRs von Konten im Aktensystem, in denen die anfragende Identität berechtigt ist.		
Formatvorgaben	Die Definition der Ein- und Ausgabeparameter erfolgt in [AuthorizationService.xsd]. Diejenigen Parameter, die im XSD-Schema optional gekennzeichnet, aber hier nicht aufgeführt sind, werden in der Operation an dieser Schnittstelle nicht verwendet.		
Eingangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**
`AuthenticationAssertion`	Die `AuthenticationAssertion` ist eine von einem Identitiy Provider ausgestellte Authentifizierungsbestätigung für einen Nutzer.	SAML Assertion im SOAP-Header des Requests	-
Ausgangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**

AuthorizationInfoList	Liste der OwnerKVNRs von Konten im Aktensystem, in denen für die Telematik-ID der anfragenden Leistungserbringerinstitution bzw. der Kostenträger ein AuthorizationKey aktuell vorhanden ist.	AuthorizationInfo[0..*]	-
Fehlermeldungen			
Name	**Fehlertext**	**Details**	
ASSERTION_INVALID	Die übergebene AuthenticationAssertion ist ungültig.	z.B. abgelaufen oder Misstrauen in Signatur des Tokens	
TECHNICAL_ERROR	Zufallszahl		

[<=]

6.2.3.6 Umsetzung I_Authorization_Management::getAuthorizationList

Die folgenden Anforderungen beschreiben die Umsetzung der Operation `I_Authorization_Management::getAuthorizationList`. Dabei gelten die übergreifenden Festlegungen zur Prüfung der Eingangsparameter.

A_17111 - Komponente Autorisierung LE - Abfrage Berechtigungsliste
Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Management::getAuthorizationList` die Liste aller OwnerKVNRs ermitteln, in deren KeyChain für die `organization-id` der gültigen AuthenticationAssertion ein AuthorizationKey vorhanden ist (`organization-id` == `ActorID`) und diese Liste als AuthorizationInformation [`OwnerKVNR` + `validTo` am jeweiligen AuthorizationKey der ActorID je KeyChain] zurückgeben.
[<=]

6.2.4 Schnittstelle I_Authorization_Management_Insurant

Diese Schnittstelle setzt die in [gemSysL_ePA] definierte Schnittstelle `I_Authorization_Management_Insurant` technisch um.

Die Schnittstelle `I_Authorization_Management_Insurant` stellt Operationen zur Verwaltung von kryptografischen Berechtigungen im Autorisierungsdienst eines Aktensystems bereit.

6.2.4.1 Operationsdefinition I_Authorization_Management_Insurant::putAuthorizationKey

A_14672 - Komponente Autorisierung - I_Authorization_Management_Insurant::putAuthorizationKey

Die Komponente Autorisierung MUSS die Operation `I_Authorization_Management_Insurant::putAuthorizationKey` gemäß der folgenden Signatur implementieren:

Tabelle 11: I_Authorization_Management_Insurant::putAuthorizationKey - Definition

Operation	**I_Authorization_Management_Insurant::putAuthorizationKey**		
Beschreibung	Mit dieser Operation wird für einen Berechtigten verschlüsseltes Schlüsselmaterial für ein konkretes Aktenkonto eines Versicherten im Aktensystem gespeichert.		
Formatvorgaben	Die Definition der Ein- und Ausgabeparameter erfolgt in [AuthorizationService.xsd]. Diejenigen Parameter, die im XSD-Schema optional gekennzeichnet, aber hier nicht aufgeführt sind, werden in der Operation an dieser Schnittstelle nicht verwendet.		
Eingangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**
`AuthenticationAssertion`	Die `AuthenticationAssertion` ist eine von einem Identitiy Provider ausgestellte Authentifizierungsbestätigung für einen Nutzer.	SAML Assertion im SOAP-Header des Requests	-
`RecordIdentifier`	Der `RecordIdentifier` referenziert ein konkretes Aktenkonto eines Versicherten bei einem Anbieter. Mit diesem wird der Datensatz der Autorisierung in der Komponente Autorisierung für den anfragenden Nutzer lokalisiert.	String	-
`AuthorizationKey`	Die kryptografische Autorisierung eines Nutzers, bestehend aus Listen von verschlüsselten Schlüsseln. Details zur Struktur finden sich im Kapitel 7 zum Informationsmodell.	AuthorizationKeyType	-
`DeviceID`	Die `DeviceID` enthält die Gerätekennung eines vom Nutzer verwendeten Gerätes.	DeviceIdType	-

`NotificationInfoRepresentative`	Mit diesem Parameterhinterlegt der Versicherte eine Benachrichtigungsadresse der Geräteverwaltung des mittels AuthorizationKey berechtigten Vertreters.	String	ja

Fehlermeldungen

Name	**Fehlertext**	**Details**
TECHNICAL_ERROR	Zufallszahl	
ASSERTION_INVALID	Authentifizierungsbestätigung ungültig	Die Authentifizierungsbestätigung des aufrufenden Nutzers wird nicht akzeptiert.
KEY_ERROR	Fehler im Schlüsseldatensatz	Es ist bereits ein Datensatz vorhanden.
SYNTAX_ERROR	Fehlerhafte Aufrufparameter	Es wurde ein fehlerhafter Aufrufparameter übergeben.
DEVICE_UNKNOWN	generierte `phr:DeviceID::Device`	Das vom Nutzer verwendete Gerät des Versicherten ist nicht bekannt und muss freigeschaltet werden.

[<=]

6.2.4.2 Umsetzung I_Authorization_Management_Insurant::putAuthorizationKey

Die folgenden Anforderungen beschreiben die Umsetzung der Operation `I_Authorization_Management_Insurant::putAuthorizationKey`. Dabei gelten die übergreifenden Festlegungen zur Prüfung der Eingangsparameter.

A_14446 - Komponente Autorisierung Vers. - Speicherung kryptografische Berechtigung des Nutzers

Die Komponente Autorisierung MUSS in der Operation `I_Authorization_Management_Insurant::putAuthorizationKey` den im Eingangsparameter übergebenen `AuthorizationKey` als *AuthorizationKey* der KeyChain des im Eingangsparameter benannten `RecordIdentifier` speichern, sofern kein *AuthorizationKey* für die ActorID zu diesem RecordIdentifier bereits vorhanden ist, und andernfalls die Operation mit der Fehlermeldung KEY_ERROR abbrechen.

[<=]

A_14447 - Komponente Autorisierung Vers. - Berechtigungsprüfung Schlüsselhinterlegung
Die Komponente Autorisierung MUSS beim Aufruf der Operation `I_Authorization_Management_Insurant::putAuthorizationKey` an hand der subject-id (KVNR) der `AuthenticationAssertion` und des `RecordIdentifier` prüfen, ob für den aufrufenden Nutzer ein AuthorizationKey mit ActorID = KVNR hinterlegt ist und falls nicht, die Operation mit dem Fehler ACCESS_DENIED abbrechen.**[<=]**

Mit dieser Prüfung wird sichergestellt, dass nur Versicherte sowie berechtigte Vertreter Schlüsselmaterial für Versicherte, Leistungserbringerinstitutionen und Kostenträger hinterlegen können, die selbst bereits über einen AuthorizationKey verfügen.

A_18184 - Komponente Autorisierung Vers. - Prüfung auf Vertretungsberechtigung für Prüfidentität
Die Komponente Autorisierung MUSS bei Hinterlegung einer Vertretungsberechtigung durch Aufruf der Operation `I_Authorization_Management_Insurant::putAuthorizationKey` mit (subject-id der AuthenticationAssertion != ActorID des Übergabeparameters AuthorizationKey und ActorID des Übergabeparameters AuthorizationKey != OwnerKVNR) prüfen, ob die Hinterlegung für eine Prüfidentität gemäß [gemSpec_PK_eGK#Card-G2-A_3820] erfolgen soll und falls ja, den Anwendungsfall mit dem Fehler TECHNICAL_ERROR abbrechen.**[<=]**

Die Erkennung auf eine Prüfidentität kann über die Auswertung der ActorID des zu berechtigenden Vertreters erfolgen, wobei diese als Prüf-KVNR anhand der Bildungsregel "`4 oder mehr gleiche aufeinander folgende Ziffern`" eindeutig zu erkennen ist.

A_17670 - Komponente Autorisierung Vers. - Freischaltprozess Vertreterberechtigung
Die Komponente Autorisierung MUSS bei Hinterlegung einer Vertretungsberechtigung durch Aufruf der Operation `I_Authorization_Management_Insurant::putAuthorizationKey` mit (subject-id der AuthenticationAssertion != ActorID des Übergabeparameters AuthorizationKey und ActorID des Übergabeparameters AuthorizationKey != OwnerKVNR) die Operation abschließen, sofern kein technischer oder fachlicher Fehler dies verhindert und anschließend den Freischaltprozess für Vertretereinrichtung starten (6.6- Freischaltprozess Vertretereinrichtung), sofern für die im Übergabeparameter AuthorizationKey benannte ActorID noch kein AuthorizationKey in der Komponente Autorisierung für die im `RecordIdentifier` benannte OwnerKVNR vorhanden ist.
[<=]

A_18750 - Komponente Autorisierung Vers. - Begrenzung zu registrierender Vertreter
Die Komponente Autorisierung MUSS bei Hinterlegung einer Vertretungsberechtigung durch Aufruf der Operation `I_Authorization_Management_Insurant::putAuthorizationKey` (vgl. A_17670) prüfen, ob die maximale Anzahl von fünf Vertretern erreicht wurde. Trifft dies zu, MUSS der Anwendungsfall mit dem Fehler TECHNICAL_ERROR abgebrochen werden. Eine Prüfung MUSS berücksichtigen, ob zum Zeitpunkt der Vertretungsregistrierung Freischaltprozesse gestartet wurden bzw. im Gange sind. Diese

Prozesse sind in der maximalen Anzahl an Vertretern zu berücksichtigen.
[<=]

A_15752 - Komponente Autorisierung Vers. - Benachrichtigungskanal für Geräteverwaltung E-Mail-Format
Die Komponente Autorisierung MUSS die Operation `I_Authorization_Management_Insurant::putAuthorizationKey` mit dem Fehler SYNTAX_ERROR abbrechen, wenn der Parameter `NotificationInfoRepresentative` nicht leer und nicht gemäß [RFC-5322] formatiert ist.**[<=]**

A_14318 - Komponente Autorisierung Vers. - Benachrichtigungskanal für Geräteverwaltung
Die Komponente Autorisierung MUSS einen in der Operation `I_Authorization_Management_Insurant::putAuthorizationKey` übergebenen optionalen Parameter `NotificationInfoRepresentative` als Benachrichtigungsadresse der Geräteverwaltung für den im Parameter AuthorizationKey durch ActorID benannten Nutzer übernehmen.**[<=]**

A_14615 - Komponente Autorisierung Vers. - Initiale Schlüsselhinterlegung Kontoeröffnung
Die Komponente Autorisierung MUSS die Operation `I_Authorization_Management_Insurant::putAuthorizationKey` mit dem Fehler ACCESS_DENIED abbrechen, sofern für den Eigentümer der Akte noch kein AuthorizationKey vorhanden ist, und der zu speichernde `AuthorizationKey` des Aufrufparameters für einen anderen Nutzer als den Eigentümer des `RecordIdentifier` (ActorID != OwnerKVNR) gespeichert werden soll.**[<=]**

Mit dieser Anforderung soll verhindert werden, dass die Akte genutzt wird, bevor das Schlüsselmaterial für den Versicherten erzeugt und hinterlegt wurde. Die benannte Konstellation liegt im Rahmen der Kontoeröffnung und bei einem Aktenumzug vor. Das Schlüsselmaterial für den Versicherten wird im Schritt der Kontoaktivierung erzeugt, welcher auf den Schritt der Kontoinitialisierung folgt.

A_14736 - Komponente Autorisierung Vers. - Initiale Schlüsselhinterlegung für den Versicherten
Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Management_Insurant::putAuthorizationKey` durch den Versicherten (subject-id (KVNR) der `AuthenticationAssertion == OwnerKVNR`) im Rahmen der initialen Schlüsselhinterlegung während der Kontoaktivierung das `validTo`-Datum des übergebenen `AuthorizationKey` vor der Speicherung mit einem technischen Datum gleichbedeutend mit "unendlich" (z.B.*31.12.9999*) ersetzen.**[<=]**

A_15000 - Komponente Autorisierung Vers. - Zustandswechsel bei Schlüsselhinterlegung für den Versicherten
Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Management_Insurant::putAuthorizationKey` durch den Versicherten (subject-id (KVNR) der `AuthenticationAssertion == OwnerKVNR`) bei erfolgreichem Abschluss der initialen Schlüsselhinterlegung für den Versicherten während der Kontoaktivierung den Zustand `RecordState` der KeyChain des Versicherten von `REGISTERED` bzw. `REGISTERED_FOR_MIGRATION` auf den Wert `ACTIVATED` setzen.**[<=]**

6.2.4.3 Operationsdefinition I_Authorization_Management_Insurant::deleteAuthorizationKey

A_14674 - Komponente Autorisierung - I_Authorization_Management_Insurant::deleteAuthorizationKey

Die Komponente Autorisierung MUSS die Operation `I_Authorization_Management_Insurant::deleteAuthorizationKey` gemäß der folgenden Signatur implementieren:

Tabelle 12: I_Authorization_Management_Insurant::deleteAuthorizationKey - Definition

Operation	**I_Authorization_Management_Insurant::deleteAuthorizationKey**		
Beschreibung	Mit dieser Operation kann ein authentifizierter Nutzer bzw. ein berechtigter Vertreter das im Aktenkonto hinterlegte kryptografische Schlüsselmaterial für einen benannten Nutzer löschen.		
Formatvorgaben	Die Definition der Ein- und Ausgabeparameter erfolgt in [AuthorizationService.xsd]. Diejenigen Parameter, die im XSD-Schema optional gekennzeichnet, aber hier nicht aufgeführt sind, werden in der Operation an dieser Schnittstelle nicht verwendet.		
Eingangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**
`AuthenticationAssertion`	Die `AuthenticationAssertion` ist eine von einem Identitiy Provider ausgestellte Authentifizierungsbestätigung für einen Nutzer.	SAML Assertion im SOAP-Header des Requests	-
`RecordIdentifier`	Der `RecordIdentifier` referenziert ein konkretes Aktenkonto eines Versicherten bei einem Anbieter. Mit diesem wird der Datensatz der Autorisierung in der Komponente Autorisierung für den anfragenden Nutzer lokalisiert.	String	-
`ActorID`	Identifikator des Nutzers, für den der hinterlegte Datensatz `AuthorizationKey` gelöscht werden soll.	String	-
`DeviceID`	Die DeviceID enthält die Gerätekennung eines vom Nutzer verwendeten Gerätes.	DeviceIdType	-

Fehlermeldungen		
Name	**Fehlertext**	**Details**
TECHNICAL_ERROR	Zufallszahl	
ASSERTION_INVALID	Authentifizierungsbestätigung ungültig	Die Authentifizierungsbestätigung des aufrufenden Nutzers wird nicht akzeptiert.
KEY_ERROR	Fehler im Schlüsseldatensatz	Kein Datensatz vorhanden
ACCESS_DENIED	Zugriff verweigert	Die Operation ist mit den angegebenen Parametern nicht zulässig.
DEVICE_UNKNOWN	generierte `phr:DeviceID::Device`	Das vom Nutzer verwendete Gerät des Versicherten ist nicht bekannt und muss freigeschaltet werden.

[<=]

6.2.4.4 Umsetzung I_Authorization_Management_Insurant::deleteAuthorizationKey

Die folgenden Anforderungen beschreiben die Umsetzung der Operation `I_Authorization_Management_Insurant::deleteAuthorizationKey`. Dabei gelten die übergreifenden Festlegungen zur Prüfung der Eingangsparameter.

A_14451 - Komponente Autorisierung Vers. - Prüfen Löschberechtigung
Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Management_Insurant::deleteAuthorizationKey` prüfen, ob der in der `AuthenticationAssertion` benannte Nutzer über einen *AuthorizationKey* mit AuthorizationType = DOCUMENT_AUTHORIZATION für den benannten `RecordIdentifier` verfügt, und andernfalls die Operation mit der Fehlermeldung ACCESS_DENIED abbrechen.

[<=]

A_14452 - Komponente Autorisierung Vers. - Löschen des AuthorizationKeys
Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Management_Insurant::deleteAuthorizationKey` den Datensatz AuthorizationKey des Nutzers löschen, der im Aufrufparameter als `ActorID` (Telematik-ID oder KVNR für Vertreter) benannt wurde.**[<=]**

A_14453 - Komponente Autorisierung Vers. - Löschverbot für Versichertenschlüssel
Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Management_Insurant::deleteAuthorizationKey` das Löschen verhindern, wenn der im Aufrufparameter als `ActorID` benannte Datensatz gleich der `OwnerKVNR` des Versicherten als Eigentümer der Akte ist, und die Operation mit der Fehlermeldung ACCESS_DENIED abbrechen.**[<=]**

A_14552 - Komponente Autorisierung Vers. - Löschen veralteter Schlüssel
Die Komponente Autorisierung MUSS alle `AuthorizationKey` löschen, deren `validTo`-Datum älter als die aktuelle Systemzeit der Komponente Autorisierung sind und das Löschen mit den folgenden Parametern protokollieren:

- UserID = interner, systemseitig wählbarer Identifikator
- UserName = Automatische Löschung nach Ablauf der Berechtigungsdauer
- ObjectID = RecordIdentifier des betroffenen Kontos
- ObjectName = ActorID des gelöschten AuthorizationKey.

[<=]

6.2.4.5 Operationsdefinition I_Authorization_Management_Insurant::replaceAuthorizationKey

A_14325 - Komponente Autorisierung - I_Authorization_Management_Insurant::replaceAuthorizationKey
Die Komponente Autorisierung MUSS die Operation `I_Authorization_Management_Insurant::replaceAuthorizationKey` gemäß der folgenden Signatur implementieren:

Tabelle 13: I_Authorization_Management_Insurant::replaceAuthorizationKey - Definition

Operation	**I_Authorization_Management_Insurant::replaceAuthorization Key**		
Beschreibung	Mit dieser Operation kann ein authentifizierter Nutzer bzw. ein berechtigter Vertreter das für eine alte eGK verschlüsselte Schlüsselmaterial durch neues für eine Folgekarte verschlüsseltes Schlüsselmaterial ersetzen.		
Formatvorgaben	Die Definition der Ein- und Ausgabeparameter erfolgt in [AuthorizationService.xsd]. Diejenigen Parameter, die im XSD-Schema optional gekennzeichnet, aber hier nicht aufgeführt sind, werden in der Operation an dieser Schnittstelle nicht verwendet.		
Eingangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**
`AuthenticationAssertion`	Die `AuthenticationAssertion` ist eine von einem Identitiy Provider ausgestellte	SAML Assertion im SOAP-Header des Requests	-

	Authentifizierungsbestätigung für einen Nutzer.		
RecordIdentifier	Der `RecordIdentifier` referenziert ein konkretes Aktenkonto eines Versicherten bei einem Anbieter. Mit diesem wird der Datensatz der Autorisierung in der Komponente Autorisierung für den anfragenden Nutzer lokalisiert.	String	-
NewAuthorizationKey	Die kryptografische Autorisierung eines Nutzers, bestehend aus Listen von verschlüsselten Schlüsseln. Details zur Struktur finden sich im Kapitel 7 zum Informationsmodell.	AuthorizationKeyType	-
DeviceID	Die `DeviceID` enthält die Gerätekennung eines vom Nutzer verwendeten Gerätes.	DeviceIdType	-
Fehlermeldungen			
Name	**Fehlertext**	**Details**	
TECHNICAL_ERROR	Zufallszahl		
KEY_ERROR	Fehler im Schlüsseldatensatz	Kein Datensatz vorhanden.	
ASSERTION_INVALID	Authentifizierungsbestätigung ungültig	Die Authentifizierungsbestätigung des aufrufenden Nutzers wird nicht akzeptiert.	
DEVICE_UNKNOWN	generierte `phr:DeviceID::Device`	Das vom Nutzer verwendete Gerät des Versicherten ist nicht bekannt und muss freigeschaltet werden.	
ACCESS_DENIED	Zugriff verweigert	Die Operation ist mit den angegebenen Parametern nicht zulässig.	

[<=]

6.2.4.6 Umsetzung I_Authorization_Management_Insurant::replaceAuthorizationKey

Die folgenden Anforderungen beschreiben die Umsetzung der Operation `I_Authorization_Management_Insurant::replaceAuthorizationKey`. Dabei gelten die übergreifenden Festlegungen zur Prüfung der Eingangsparameter.

A_14454 - Komponente Autorisierung Vers. - Prüfung Datensatz für bestehenden AuthorizationKey
Die Komponente Autorisierung MUSS für die Operation `I_Authorization_Management_Insurant::replaceAuthorizationKey` prüfen, ob ein *AuthorizationKey* für den benannten `RecordIdentifier` und den in der `AuthenticationAssertion` benannten Nutzer (subject-id == ActorID des vorhandenen AuthorizationKey) hinterlegt ist, und andernfalls die Operation mit der Fehlermeldung ACCESS_DENIED abbrechen.**[<=]**

A_14455 - Komponente Autorisierung Vers. - Ersetzen des AuthorizationKeys
Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Management_Insurant::replaceAuthorizationKey` den Datensatz *AuthorizationKey* desjenigen Nutzers durch den übergebenen `NewAuthorizationKey` ersetzen, der im Aufrufparameter als *ActorID* (Telematik-ID oder KVNR) benannt wurde und für den ein `AuthorizationKey` vorhanden ist.**[<=]**

A_15120 - Komponente Autorisierung Vers. - Fixierung des AuthorizationType für Vertreter
Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Management_Insurant::replaceAuthorizationKey` prüfen, ob ein Vertreter seinen eigenen Schlüssel ersetzt (`OwnerKVNR != subject-id == ActorID` des vorhandenen AuthorizationKey `== ActorID` in `NewAuthorizationKey`) und in diesem Fall den AuthorizationType des vorhandenen AuthorizationKey in den zu speichernden `NewAuthorizationKey` übernehmen.
Die Komponente Autorisierung MUSS die Operation mit dem Fehler ACCESS_DENIED abbrechen, wenn ein lediglich zur Umschlüsselung berechtigter Vertreter (RECOVERY_AUTHORIZATION im hinterlegten AuthorizationKey des Vertreters) versucht einen anderen AuthorizationKey zu ersetzen als den eigenen oder den des Versicherten.

[<=]

A_15889 - Komponente Autorisierung Vers. - Prüfung KVNR bei Schlüsselwechsel für den Versicherten
Die Komponente Autorisierung MUSS den Aufruf der Operation `I_Authorization_Management_Insurant::replaceAuthorizationKey` durch den Versicherten als Eigentümer der Akte (ActorId des übergebenen `AuthorizationKey == OwnerKVNR` für den benannten `RecordIdentifier`) mit der Fehlermeldung ACCESS_DENIED abbrechen, wenn der unveränderliche Teil der KVNR des Versicherten im übergebenen AuthorizationKey nicht übereinstimmt mit dem unveränderlichen Teil der KVNR des Versicherten im bereits gespeicherten AuthorizationKey.
[<=]

6.2.4.7 Operationsdefinition I_Authorization_Management_Insurant::getAuditEvents

A_14676 - Komponente Autorisierung - I_Authorization_Management_Insurant::getAuditEvents

Die Komponente Autorisierung MUSS die Operation `I_Authorization_Management_Insurant::getAuditEvents` gemäß der folgenden Signatur implementieren:

Tabelle 14: I_Authorization_Management_Insurant::getAuditEvents - Definition

Operation	**I_Authorization_Management_Insurant::getAuditEvents**		
Beschreibung	Mit dieser Operation kann ein authentifizierter Versicherter bzw. ein berechtigter Vertreter das Verwaltungsprotokoll der Autorisierungskomponente auslesen.		
Formatvorgaben	Die Definition der Ein- und Ausgabeparameter erfolgt in [AuthorizationService.xsd]. Diejenigen Parameter, die im XSD-Schema optional gekennzeichnet, aber hier nicht aufgeführt sind, werden in der Operation an dieser Schnittstelle nicht verwendet.		
Eingangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**
`AuthenticationAssertion`	Die `AuthenticationAssertion` ist eine von einem Identitiy Provider ausgestellte Authentifizierungsbestätigung für einen Nutzer.	SAML Assertion im SOAP-Header des Requests	-
`RecordIdentifier`	Der `RecordIdentifier` referenziert ein konkretes Aktenkonto eines Versicherten bei einem Anbieter. Mit diesem wird der Datensatz der Autorisierung in der Komponente Autorisierung für den anfragenden Nutzer lokalisiert.	String	-
`DeviceID`	Die DeviceID enthält die Gerätekennung eines vom Nutzer verwendeten Gerätes.	DeviceIdType	-
Ausgangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**
`AuditEventList`	Liste der Verwaltungsprotokolleinträge des im RecordIdentifier referenzierten Aktenkontos	AuditMessage [0..*]	-

Fehlermeldungen		
Name	**Fehlertext**	**Details**
TECHNICAL_ERROR	Zufallszahl	
ASSERTION_INVALID	Authentifizierungsbestätigung ungültig	Die Authentifizierungsbestätigung des aufrufenden Nutzers wird nicht akzeptiert.
DEVICE_UNKNOWN	generierte `phr:DeviceID::Device`	Das vom Nutzer verwendete Gerät des Versicherten ist nicht bekannt und muss freigeschaltet werden.

[<=]

6.2.4.8 Umsetzung I_Authorization_Management_Insurant::getAuditEvents

Die folgenden Anforderungen beschreiben die Umsetzung der Operation `I_Authorization_Management_Insurant::getAuditEvents`. Dabei gelten die übergreifenden Festlegungen zur Prüfung der Eingangsparameter.

A_14394 - Komponente Autorisierung Vers. - Auslesen Verwaltungsprotokoll
Die Komponente Autorisierung MUSS beim Aufruf der Operation `I_Authorization_Management_Insurant::getAuditEvents` dem anhand einer `AuthenticationAssertion` authentifizierten Nutzer die Liste aller zum angefragten `RecordIdentifier` verfügbaren Verwaltungsprotokolleinträge gemäß [gemSpec_DM_ePA#A_14471] zurückliefern, wenn der Wert von `DeviceID::Device` des Aufrufparameters gleich dem Wert "`urn:gematik:fa:phr:1.0:device:device-id`" einer für diesen Nutzer ausgestellten `Autorisierungsbestätigung` der in der Komponente Autorisierung gespeicherten Sessiondaten für diesen Nutzer ist.**[<=]**

Damit wird sichergestellt, dass das Auslesen des Verwaltungsprotokolls nur gestattet wird, wenn zuvor eine Autorisierungsbestätigung für diesen Nutzer ausgestellt wurde.

6.2.4.9 Operationsdefinition I_Authorization_Management_Insurant::putNotificationInfo

A_14344 - Komponente Autorisierung - I_Authorization_Management_Insurant::putNotificationInfo
Die Komponente Autorisierung MUSS die Operation `I_Authorization_Management_Insurant::putNotificationInfo` gemäß der folgenden Signatur implementieren:

Tabelle 15: I_Authorization_Management_Insurant::putNotificationInfo - Definition

Operation	**I_Authorization_Management_Insurant::putNotificationInfo**

Beschreibung	Mit dieser Operation kann ein authentifizierter Versicherter bzw. ein berechtigter Vertreter seine im Benachrichtigungskanal hinterlegte Adresse aktualisieren.		
Formatvorgaben	Die Definition der Ein- und Ausgabeparameter erfolgt in [AuthorizationService.xsd]. Diejenigen Parameter, die im XSD-Schema optional gekennzeichnet, aber hier nicht aufgeführt sind, werden in der Operation an dieser Schnittstelle nicht verwendet.		
Eingangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**
`AuthenticationAssertion`	Die `AuthenticationAssertion` ist eine von einem Identitiy Provider ausgestellte Authentifizierungsbestätigung für einen Nutzer.	SAML Assertion im SOAP-Header des Requests	-
`RecordIdentifier`	Der `RecordIdentifier` referenziert ein konkretes Aktenkonto eines Versicherten bei einem Anbieter. Mit diesem wird der Datensatz der Autorisierung in der Komponente Autorisierung für den anfragenden Nutzer lokalisiert.	String	-
`DeviceID`	Die `DeviceID` enthält die Gerätekennung eines vom Nutzer verwendeten Gerätes.	DeviceIdType	-
`NewNotificationInfo`	`NewNotificationInfo` beinhaltet die neue Benachrichtigungsadresse, die für den authentifizierten Nutzer gespeichert werden soll.	String	-
Fehlermeldungen			
Name	**Fehlertext**	**Details**	
TECHNICAL_ERROR	Zufallszahl		
SYNTAX_ERROR	Fehlerhafte Aufrufparameter	Es wurde ein fehlerhafter Aufrufparameter übergeben.	
DEVICE_UNKNOWN	generierte `phr:DeviceID::Device`	Das vom Nutzer verwendete Gerät des Versicherten ist nicht	

		bekannt und muss freigeschaltet werden.
ACCESS_DENIED	Zugriff verweigert	Die Operation ist mit den angegebenen Parametern nicht zulässig.

[<=]

6.2.4.10 Umsetzung I_Authorization_Management_Insurant::putNotificationInfo

Die folgenden Anforderungen beschreiben die Umsetzung der Operation `I_Authorization_Management_Insurant::putNotificationInfo`. Dabei gelten die übergreifenden Festlegungen zur Prüfung der Eingangsparameter.

A_14715 - Komponente Autorisierung Vers. - Aktualisierung Benachrichtigungsadresse

Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Management_Insurant::putNotificationInfo` den Wert des Parameters `NotificationInfoRepresentative` als Benachrichtigungsadresse des in der `AuthenticationAssertion` benannten Nutzers für den hinterlegten AuthorizationKey des Nutzers `(subject-id` der AuthenticationAssertion `== ActorID` des AuthorizationKey) speichern.**[<=]**

A_14716 - Komponente Autorisierung Vers. - E-Mail-Format

Die Komponente Autorisierung MUSS die Operation `I_Authorization_Management_Insurant::putNotificationInfo` mit dem Fehler SYNTAX_ERROR abbrechen, wenn der Parameter `NewNotificationInfo` nicht gemäß [RFC-5322] formatiert ist.
[<=]

Mit dieser Funktion kann ein Versicherter oder ein berechtigter Vertreter seine persönliche Benachrichtigungsadresse zur Gerätefreischaltung ändern. Sowohl für Versicherte als auch deren berechtigte Vertreter sind vor deren jeweiligem Zugriff Benachrichtigungsadressen vorhanden, da diese Operation ohne Gerätefreischaltung über ihre Adresse nicht aufrufbar ist.

Für Versicherte wird die Benachrichtigungsadresse initial im Rahmen der Kontoeröffnung hinterlegt. Für Vertreter erfolgt die initiale Hinterlegung der Benachrichtigungsadresse durch den Versicherten mittels `I_Authorization_Management_Insurant::putAuthorizationKey` während der Vergabe der Zugriffsberechtigung.

6.2.4.11 Operationsdefinition I_Authorization_Management_Insurant::getAuthorizationList

A_17113 - Komponente Autorisierung - I_Authorization_Management_Insurant::getAuthorizationList

Die Komponente Autorisierung MUSS die Operation `I_Authorization_Management_Insurant::getAuthorizationList` gemäß der folgenden Signatur implementieren:

Tabelle 16: I_Authorization_Management_Insurant::getAuthorizationList - Definition

Operation	I_Authorization_Management_Insurant::getAuthorizationList		
Beschreibung	Die Operation liefert eine Liste aller AuthorizationKeys eines Kontos im Aktensystems, als Liste aller Berechtigten in einem Aktenkonto.		
Formatvorgaben	Die Definition der Ein- und Ausgabeparameter erfolgt in [AuthorizationService.xsd]. Diejenigen Parameter, die im XSD-Schema optional gekennzeichnet, aber hier nicht aufgeführt sind, werden in der Operation an dieser Schnittstelle nicht verwendet.		
Eingangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**
`AuthenticationAssertion`	Die `AuthenticationAssertion` ist eine von einem Identitiy Provider ausgestellte Authentifizierungsbestätigung für einen Nutzer.	SAML Assertion im SOAP-Header des Requests	-
`RecordIdentifier`	Der `RecordIdentifier` referenziert ein konkretes Aktenkonto eines Versicherten bei einem Anbieter. Mit diesem wird der Datensatz der Autorisierung in der Komponente Autorisierung für den anfragenden Nutzer lokalisiert.	String	-
`DeviceID`	Die `DeviceID` enthält die Gerätekennung eines vom Nutzer verwendeten Gerätes.	DeviceIdType	-
Ausgangsparameter			
Name	**Beschreibung**	**Typ**	**opt.**
`AuthorizationKeyList`	Liste der AuthorizationKeys des per RecordIdentifier identifizierten Kontos.	AuthorizationKeyType[0..*]	-
Fehlermeldungen			
Name	**Fehlertext**	**Details**	

TECHNICAL_ERROR	Zufallszahl	
DEVICE_UNKNOWN	generierte `phr:DeviceID::Device`	Das vom Nutzer verwendete Gerät des Versicherten ist nicht bekannt und muss freigeschaltet werden.
ACCESS_DENIED	Zugriff verweigert	Die Operation ist mit den angegebenen Parametern nicht zulässig.

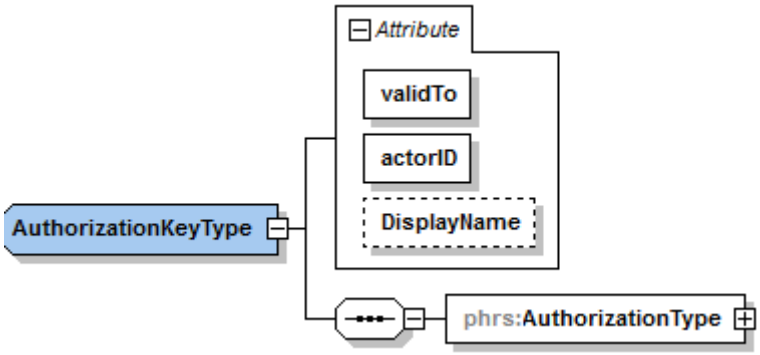
[<=]

6.2.4.12 Umsetzung I_Authorization_Management_Insurant::getAuthorizationList

A_17115 - Komponente Autorisierung Vers. - Berechtigung für Berechtigungsliste
Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Management_Insurant::getAuthorizationList` prüfen, ob für den in der AuthenticationAssertion benannten User ein AuthorizationKey in der Keychain der mittels RecordIdentifier benannten Akte vorhanden ist (`subject-id` == `ActorID`) und andernfalls die Operation mit ACCESS_DENIED abbrechen.
[<=]

A_17114 - Komponente Autorisierung Vers. - Abfrage Berechtigungsliste

Die Komponente Autorisierung MUSS bei Aufruf der Operation `I_Authorization_Management_Insurant::getAuthorizationList` die Liste aller AuthorizationKey in der KeyChain der im RecordIdentifier benannten Akte mit Ausnahme des AuthorizationKey des Eigentümers der Akte (für alle zurückgegebenen AuthorizationKey MUSS gelten: `ActorID` != `OwnerKVNR`) in der folgenden Struktur zurückgeben



Das heißt, in der Rückgabe an den Aufrufenden werden alle relevanten AuthorizationKeys jeweils ohne das Element EncryptedKeyContainer zurückgegeben. **[<=]**

6.3 Berechtigungstypen der Autorisierung

Der Berechtigungstyp (`AuthorizationType`) steuert den Zugriff auf weitere Ressourcen für einen authentisierten Nutzer. Der Berechtigungstyp wird beim Hinzufügen des Schlüsselmaterials für einen Nutzer in der Autorisierungskomponente hinterlegt.

Es wird zwischen drei Typen unterschieden, die in der folgenden Tabelle beschrieben sind:

Tabelle 17: Berechtigungstypen für AuthorizationType

AuthorizationType	**Beschreibung**
DOCUMENT_AUTHORIZATION (Dokumentenautorisierung)	Es wird für einen authentisierten Nutzer eine Autorisierungsbestätigung ausgestellt, die für den Zugang zur Dokumentenverwaltung notwendig ist.
RECOVERY_AUTHORIZATION (Umschlüsselungsautorisierung)	Es wird einem authentisierten Nutzer die Verwendung des hinterlegten Schlüssels zur lokalen Umschlüsselung gestattet. Mit dieser Autorisierungsbestätigung ist kein Zugriff auf die Komponente Dokumentenverwaltung möglich
ACCOUNT_AUTHORIZATION (Betreiberwechselautorisierung)	Es wird dem authentisierten Nutzer eine Autorisierungsbestätigung ausgestellt, mit dem in der Komponente Dokumentenverwaltung nur ein eingeschränkter Zugriff auf Daten des Versicherten möglich ist.

6.4 Hardware-Merkmal der Komponente Autorisierung

Es müssen die privaten Schlüssel der Ausstelleridentität für Autorisierungsbestätigungen sowie der TLS-Server-Identität sicher gespeichert werden.

A_14366 - Komponente Autorisierung - Verwendung eines HSM
Die Komponente Autorisierung MUSS das private Schlüsselmaterial der Ausstelleridentität C.FD.SIG und der TLS-Server-Identität C.FD.TLS-S in einem HSM speichern.**[**<=**]**

6.5 Geräteverwaltung

Die Komponente Autorisierung setzt zusätzlich zur kryptografischen Autorisierung eine Geräteautorisierung um. Dazu wird bei Zugriffen aus der Umgebung des Versicherten (über das Internet) geprüft, ob ein Versicherter bzw. berechtigter Vertreter ein bekanntes Gerät für den Zugriff nutzt. Ist das Gerät unbekannt, wird ein Freischaltprozess über einen separaten Benachrichtigungskanal gestartet. Die Erkennung erfolgt auf Basis einer im Gerät des Versicherten gebildeten DeviceID, welche in den Operationsaufrufen mitgeschickt werden muss. Die DeviceId als `DeviceIdType` gemäß [PHR_Common.xsd] enthält neben der eigentlichen Gerätekennung `Device`, welche für den Abgleich bekannter Geräte verwendet wird, einen `DisplayName`, der dem Nutzer die Verwaltung seiner genutzten Geräte erleichtert.

Die Umsetzung erfolgt in der Komponente Autorisierung, da eine vorgelagerte zustandslose Komponente der Authentifizierung von Nutzern, ggfs. nicht über einen Speicher zur Verwaltung von Gerätekennungen je Benutzerkonto verfügt bzw. dieser für diesen Zweck erst geschaffen werden müsste.

Die Prüfung auf ein autorisiertes Gerät erfolgt vor der Herausgabe des in der Komponente Autorisierung gespeicherten Schlüsselmaterials.

Für die Benachrichtigung mit anschließender Freischaltung werden E-Mails mit generierten URLs auf generierte HTML-Webseiten verwendet, da E-Mail aus Usability-Sicht am komfortabelsten erscheint und diese Methoden in verschiedensten Diensten im Internet etabliert und den Versicherten sehr wahrscheinlich bekannt sind.

6.5.1 Freischaltprozess neuer Geräte

Der Freischaltprozess dient dazu, ein Endgerät des Versicherten in der Komponente Autorisierung zu registrieren. Der folgende Ablauf zeigt informativ einen möglichen Ablauf des Freischaltprozesses.

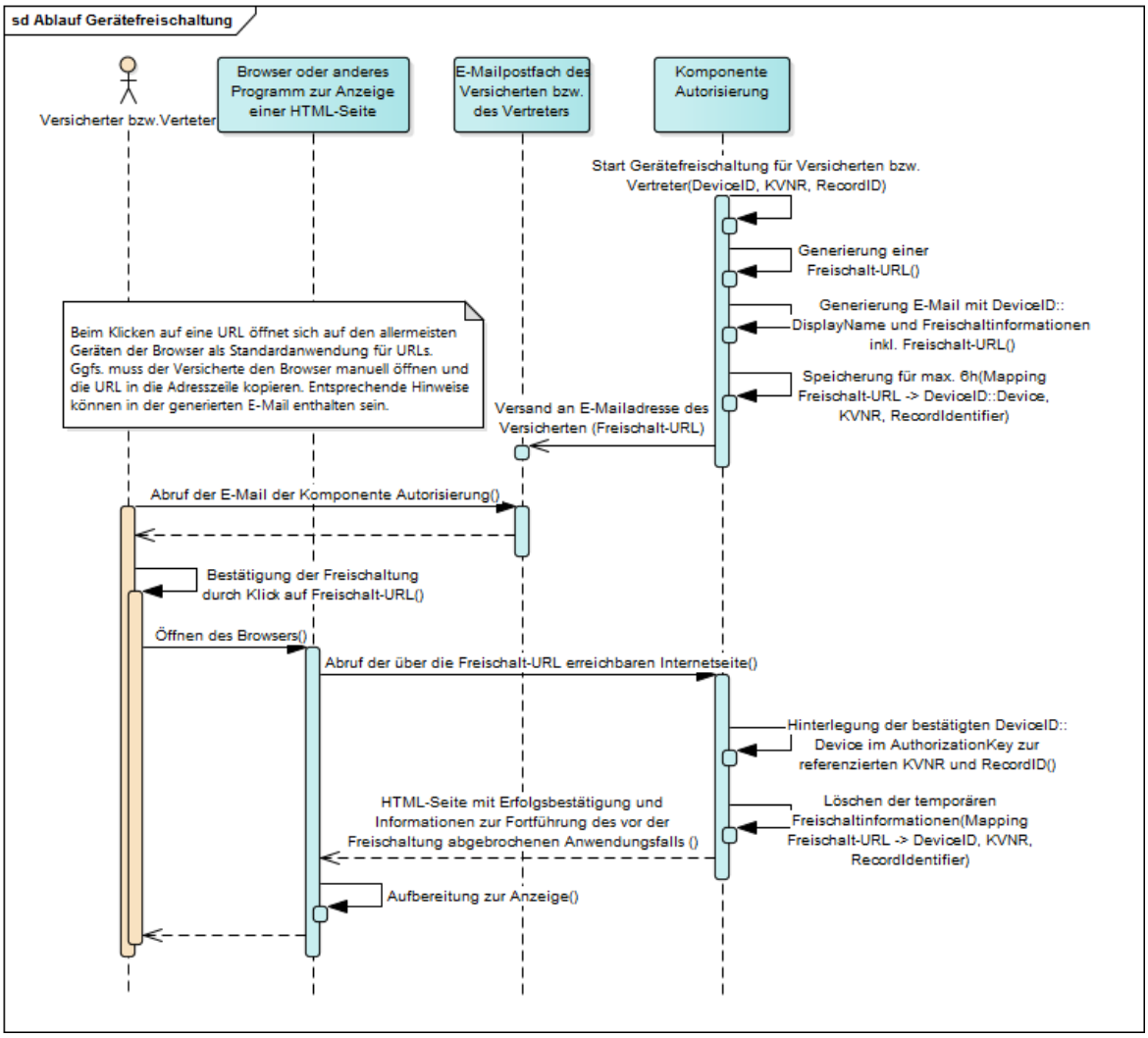


Abbildung 4: Informativer Ablauf des Geräte-Freischaltprozesses

Die Komponente Autorisierung startet den Freischaltprozess für jedes über `DeviceID::Device` identifizierte Gerät, das für den AuthorizationKey eines per KVNR identifizierten Versicherten bzw. Vertreter zu einer genannten RecordID als unbekannt gilt. D.h. ein vom Vertreter im eigenen Aktenkonto verwendetes Gerät kann dort bereits registriert sein, im Rahmen der Vertretung eines anderen Versicherten kann das gleiche Gerät am Vertretungsschlüssel unbekannt sein. In diesem Fall ist der Freischaltprozess für die Wahrnehmung der Vertretung erforderlich.

Die Komponente Autorisierung generiert zu einem Freischaltprozess einen eindeutigen Link auf Basis von Zufallszahlen und verschickt ihn an die vom Nutzer hinterlegte Benachrichtigungs-E-Mail-Adresse. Durch Klicken auf diesen Link erhält der Versicherte bzw. Vertreter eine Webseite, mit der Bitte um Bestätigung der Freischaltung des genutzten Geräts. Nach Erhalt der Freischaltbestätigung fügt die Komponente Autorisierung das per DeviceID identifizierte Gerät zum AuthorizationKey des Versicherten bzw. Vertreters hinzu.

A_17866 - Komponente Autorisierung - Generierung Device-Kennung für unbekanntes Gerät des Versicherten

Die Komponente Autorisierung MUSS bei Aufruf einer Operation der Schnittstellen `I_Authorization_Insurant` und `I_Authorization_Management_Insurant` mit einem für den aufrufenden Nutzer im benannten `RecordIdentifier` unbekanntem Parameter `phr:DeviceID::Device` eine 256 Bit Zufallszahl (base64-kodiert) mit einer Mindestentropie von 120 Bit und Erzeugung gemäß [gemSpec_Krypt#GS-A_4367] erzeugen, diese als `phr:DeviceID::Device` für den aufrufenden Nutzer im benannten RecordIdentifier konfigurieren und den Freischaltprozess gemäß [gemSpec_Autorisierung#A_14515] starten.

[<=]

Mit der Generierung der Device-Kennung auf Basis einer Zufallszahl je Konto ergibt sich, dass die Verwendung eines Geräts in verschiedenen Konten (z.B. eigenes Konto + Vertretungsberechtigung in einem anderen Konto) zur Erzeugung zweier verschiedener Device-IDs führt, die im jeweiligen Aufrufkontext zu verwenden sind.

A_17947 - Komponente Autorisierung - Gültigkeitszeitraum und Löschung der Devicekennung
Die Komponente Autorisierung MUSS jede generierte und in einem Aktenkonto gespeicherte Device-Kennung `phr:DeviceID::Device` nach 2 Jahren löschen und darf Nutzeranfragen mit dieser Device-Kennung nach diesem Zeitpunkt nicht mehr akzeptieren.
[<=]

Daraus folgt, dass nach zwei Jahren eine Neuregistrierung des verwendeten Geräts erforderlich ist. Ein möglicher Zeitraum der Inaktivität des Geräts ist dabei irrelevant

A_14515 - Komponente Autorisierung - Freischaltprozess Freischalt-URL
Die Komponente Autorisierung MUSS im Freischaltprozess eine Freischalt-URL erzeugen, die einzig aus dem FQDN der Komponente Autorisierung und einer Zufallszahl (base64-kodiert) mit mindestens 120 Bit Entropie und Erzeugung gemäß [gemSpec_Krypt#GS-A_4367] besteht und diese Freischalt-URL an die E-Mail-Adresse am `AuthorizationKey` des via `KVNR` einer `AuthenticationAssertion` referenzierten Nutzers zum angefragten `RecordIdentifier` verschicken.**[<=]**

A_14518 - Komponente Autorisierung - Freischaltprozess Freischalt-URL Transportsicherheit
Die Komponente Autorisierung MUSS in der generierten Freischalt-URL das https-Protokoll verwenden.
[<=]

A_14520 - Komponente Autorisierung - Freischaltprozess Webseite zu Freischalt-URL
Die Komponente Autorisierung MUSS bei Aufruf einer generierten Freischalt-URL durch einen Versicherten bzw. Vertreter mit einer HTML-Seite mit folgendem Inhalt über den transportverschlüsselten Kanal der https-Freischalt-URL antworten:

- DeviceID::DisplayName des freizuschaltenden Geräts
- Zeitpunkt des Starts des Freischaltprozesses
- RecordIdentifier
- Bestätigungslink (submit) zur endgültigen Freischaltung des Geräts

[<=]

A_14521 - Komponente Autorisierung - Freischaltprozess DeviceID hinzufügen
Die Komponente Autorisierung MUSS bei Abruf des Bestätigungslinks eines aktiven Freischaltprozesses die generierte `phr:DeviceID::Device` zum `AuthorizationKey` eines `RecordIdentifiers` des über `KVNR` einer `AuthenticationAssertion` identifizierten Versicherten bzw. Vertreters hinzufügen und den Freischaltprozess für den Vorgang zu DeviceID, KVNR und RecordIdentifier beenden.
[<=]

A_14522 - Komponente Autorisierung - Freischaltprozess beenden
Die Komponente Autorisierung MUSS den Vorgang eines Freischaltprozesses zu `DeviceID, KVNR` und `RecordIdentifier` nach 6 Stunden Wartezeit beenden.**[<=]**

A_14523 - Komponente Autorisierung - Freischaltprozess Löschen nach Beendigung
Die Komponente Autorisierung MUSS beim Beenden des Vorgangs eines Freischaltprozesses die generierte Freischalt-URL und alle dazugehörigen temporären Daten löschen.**[<=]**

6.5.2 Geräteadministration

Mit der Geräteadministration wird dem Nutzer die Möglichkeit gegeben, seine Endgeräte zu verwalten.

A_14364 - Komponente Autorisierung - Geräteverwaltung
Die Komponente Autorisierung MUSS dem authentifizierten Versicherten über eine ~~grafische Web-Oberfläche~~ Web-Schnittstelle folgende Funktionen zur Verfügung stellen:

- Sperren von registrierten Geräten, so dass ein Zugriff über diese Geräte bis zur Entsperrung nicht möglich ist,
- Entsperren von gesperrten Geräten, so dass ein Zugriff über diese Geräte möglich ist,
- Deregistrieren von Geräten, so dass ein Zugriff über diese Geräte erst nach erneuter erfolgreicher Freischaltung möglich ist sowie
- das Vergeben einer alternativen Bezeichnung für ein registriertes Gerät.

[<=]

A_15438 - Komponente Autorisierung - Keine negative Beeinflussung des Aktensystems durch die Geräteverwaltung
Die Komponente Autorisierung MUSS sicherstellen, dass das Web-Frontend zur Geräteverwaltung der Komponente Autorisierung so geschützt wird, dass keine negative Beeinflussung des Aktensystems über diese Schnittstelle möglich ist.**[<=]**

A_14595 - Komponente Autorisierung - Pflegeprozess Geräteverwaltung
Die Komponente Autorisierung MUSS die interne Liste aller bekannten Geräte derart pflegen, dass ein Gerät nach spätestens einem Jahr nach der letzten Nutzung des Gerätes automatisch aus der Liste der registrierten Geräte gelöscht wird, und bei anschließender Verwendung durch einen Versicherten als unbekanntes Gerät über den Freischaltprozess neu freizuschalten ist.**[<=]**

A_15551 - Komponente Autorisierung - Deregistrierung in fremden Konten

Die Komponente Autorisierung MUSS sicherstellen, dass der Versicherte nur diejenigen registrierten Geräte verwalten kann, die der Versicherte oder ein Vertreter in seinem Konto verwendet. Eine Deregistrierung eines Gerätes in einem Konto DARF NICHT automatisch zu einer Deregistrierung in einem anderen Konto führen (z.B. im Konto eines anderen Versicherten, für das der Versicherte Vertretungsrechte besitzt).**[**<=**]**

A_15755 - Komponente Autorisierung - Protokollierung Geräteverwaltung
Die Komponente Autorisierung MUSS alle Vorgänge der Geräteverwaltung im Verwaltungsprotokoll des Versicherten protokollieren.**[**<=**]**

6.6 Freischaltprozess Vertretereinrichtung

Die Komponente Autorisierung führt eine zusätzliche Autorisierung durch den Versicherten bei Einrichtung einer Vertretung für einen Vertreter durch. Der Versicherte wird aufgefordert, auf einen Link in einer E-Mail zu klicken, um die Speicherung eines AuthorizationKey für einen Vertreter zu autorisieren, den er über `I_Authorization_Management_Insurant::putAuthorizationKey` für diesen Vertreter hinterlegt. Die E-Mail mit dem Link zur Freischaltung wird an die E-Mail-Adresse des Versicherten geschickt, die auch für die Gerätefreischaltung des Versicherten verwendet wurde. Der folgende Ablauf zeigt informativ einen möglichen Ablauf des Freischaltprozesses.

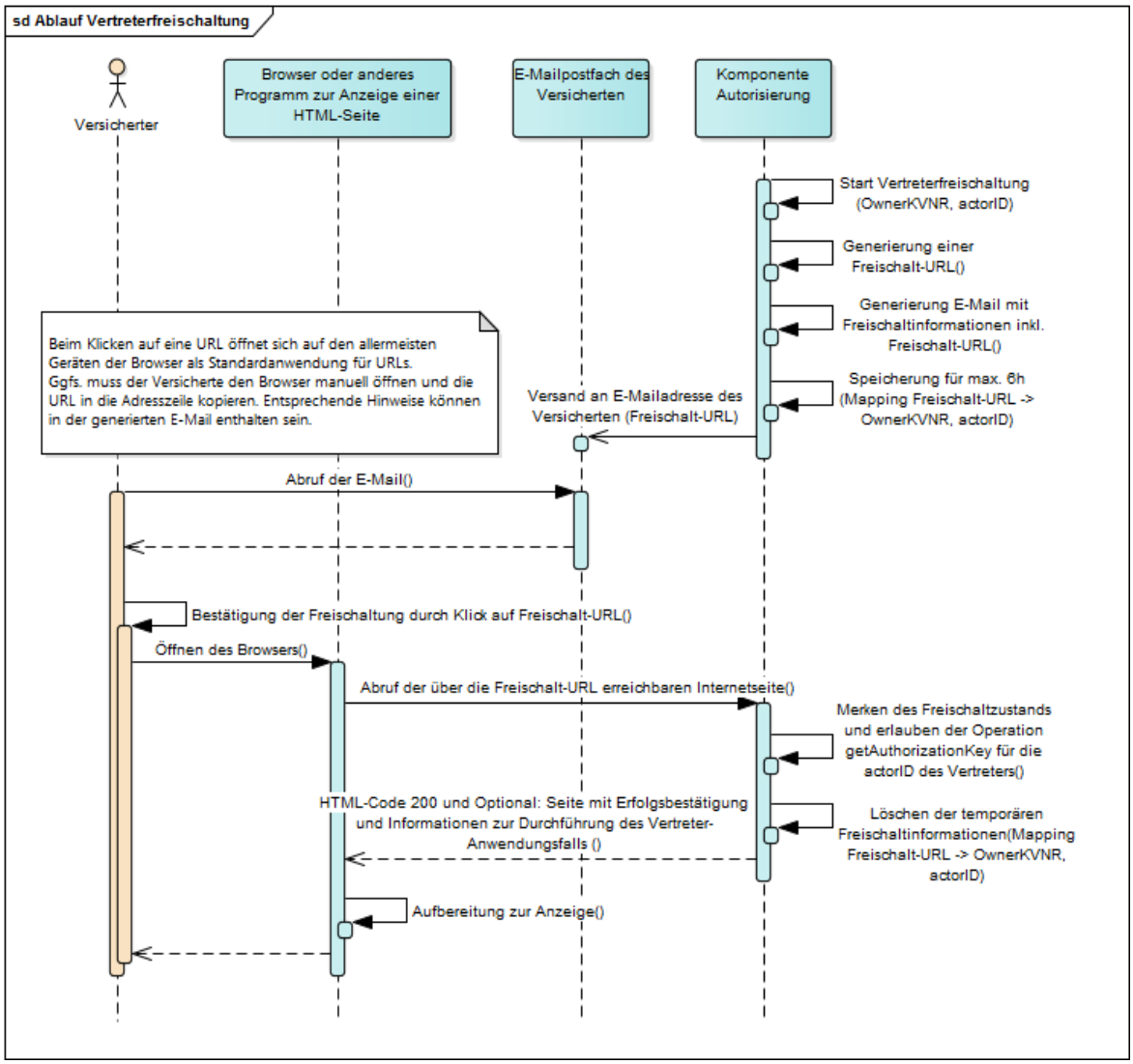


Abbildung 5: Informativer Ablauf des Freischaltprozesses für Vertretung

Die Komponente Autorisierung startet den Freischaltprozess wenn der Versicherte mittels `I_Authorization_Management_Insurant::putAuthorizationKey` für einen konkreten mittels KVNR identifizierten Vertreter (als `ActorID` am AuthorizationKey) erstmalig eine Berechtigung hinterlegen möchte. Die Operation wird zunächst erfolgreich abgeschlossen, sofern kein fachlicher oder technischer Fehler dies verhindert. Dem Vertreter wird der Zugriff auf diesen Schlüssel jedoch solange verwehrt, wie der Versicherte noch nicht auf einen Freischaltlink in einer generierten Freischalt-E-Mail klickt. Die Komponente Autorisierung generiert zum Freischaltprozess der Vertretung einen eindeutigen Link auf Basis von Zufallszahlen und verschickt ihn an die vom Versicherten hinterlegte Benachrichtigungs-E-Mail-Adresse.

Durch Klicken auf diesen Link signalisiert der Versicherte der Komponente Autorisierung, dass die Hinterlegung eines AuthorizationKey für die KVNR d.h. `ActorID` des Vertreters rechtmäßig ist. Die Komponente Autorisierung speichert diesen Freischaltzustand für die `ActorID` des Vertreters und teilt dem Versicherten über die mittels Freischaltlink abgerufene Webseite mit, dass der UseCase des Schlüsselabrufs mittels `I_Authorization~~_Management~~_Insurant::getAuthorizationKey` durch den

Vertreter nun autorisiert ist. Der Vertreter kann nun den hinterlegten Schlüssel abrufen und eine Vertretung wahrnehmen.

A_17672 - Komponente Autorisierung - Freischaltprozess Vertretung Freischalt-URL
Die Komponente Autorisierung MUSS im Freischaltprozess Vertretereinrichtung eine Freischalt-URL erzeugen, die einzig aus dem FQDN der Komponente Autorisierung und einer Zufallszahl (base64-kodiert) mit mindestens 120 Bit Entropie und Erzeugung gemäß [gemSpec_Krypt#GS-A_4367] besteht und diese Freischalt-URL an die E-Mail-Adresse des via `OwnerKVNR` referenzierten Versicherten verschicken.
[<=]

A_17673 - Komponente Autorisierung - Freischaltprozess Vertretung Freischalt-URL Transportsicherheit
Die Komponente Autorisierung MUSS in der generierten Freischalt-URL das https-Protokoll verwenden.
[<=]

A_17674 - Komponente Autorisierung - Freischaltprozess Vertretung getAuthorizationKey erlauben
Die Komponente Autorisierung MUSS bei Abruf des Bestätigungslinks eines aktiven Freischaltprozesses zur `OwnerKVNR` und `ActorId` des zukünftigen Vertreters die Operation `I_Authorization~~_Management~~_Insurant::getAuthorizationKey` für das Abrufen eines AuthorizationKey durch den Vertreter (`ActorId` = KVNR des zukünftigen Vertreters) erlauben und den Freischaltprozess für den Vorgang zu `OwnerKVNR` und `ActorID` beenden.
[<=]

Damit wird die Operation `I_Authorization~~_Management~~_Insurant::getAuthorizationKey` bei zukünftigen Aufrufen durch den Vertreter für die freigeschaltete `ActorID` nicht mehr mit Fehler REPRESENTATIVE_PENDING abgebrochen.

A_17677 - Komponente Autorisierung - Freischaltprozess Vertretung Information
Die Komponente Autorisierung KANN in der HTTP-Response zum URL-Aufruf der Vertreterfreischaltung eine Meldung über die erfolgreiche Freischaltung an den aufrufenden Versicherten zurückgeben.
[<=]

A_17675 - Komponente Autorisierung - Freischaltprozess Vertretung beenden
Die Komponente Autorisierung MUSS den Vorgang eines Freischaltprozesses Vertretung zur `OwnerKVNR` und `ActorID` nach 6 Stunden Wartezeit beenden.
[<=]

A_17676 - Komponente Autorisierung - Freischaltprozess Vertretung Löschen nach Beendigung
Die Komponente Autorisierung MUSS beim Beenden des Vorgangs eines Freischaltprozesses die generierte Freischalt-URL und alle dazugehörigen temporären Daten löschen.
[<=]

7 Informationsmodell

Das folgende Informationsmodell der Autorisierung gibt eine Übersicht über die verwendeten Objekte mit ihren Eigenschaften und Beziehungen zueinander.

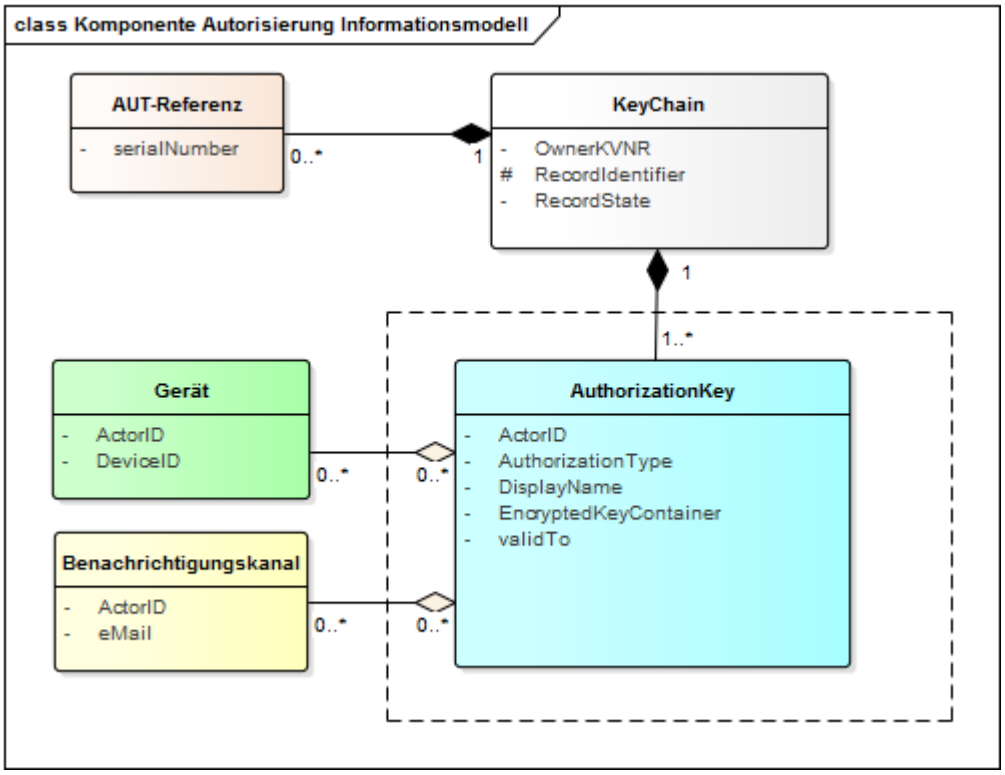


Abbildung 6: Informationsmodell der intern verwalteten Daten

Das blau dargestellte Element bildet den verwalteten `AuthorizationKey`, der vom Versicherten für jeden berechtigten Nutzer in der Komponente Autorisierung hinterlegt wird, das Element `EncryptedKeyContainer` enthält dabei das mit dem Empfängerschlüssel individuell verschlüsselte Schlüsselmaterial der Akte (Akten- und Kontextschlüssel). Die Summe aller AuthorizationKeys zu einem über den `RecordIdentifier` identifizierten Konto eines über die `OwnerKVNR` identifizierten Versicherten bildet das logische Element des "Schlüsselrings" `KeyChain`. Zu einem über `ActorID` identifizierten Nutzer wird eine Liste autorisierter `Geräte` (grün dargestellt) geführt, die bei Zugriffen aus der Umgebung des Versicherten die Zulässigkeit des genutzten `Geräts` prüfen lässt. Für den Fall eines unbekannten und somit nicht in der Liste zulässiger Geräte enthaltenen Geräts wird ein Freischaltprozess über einen `Benachrichtigungskanal` gestartet. Die Zuordnung der Benachrichtigungsadressen zum jeweiligen Nutzer ist im Bild gelb dargestellt.

Für Versicherte und deren Vertreter wird der unveränderliche Teil der `KVNR` (VersichertenID) der eGK als `ActorID` verwendet. Für den Versicherten wird genau diese ID auch als OwnerKVNR genutzt, um den jeweiligen Versicherten als Eigentümer einer Akte zu identifizieren. Für Leistungserbringerinstitutionen und Kostenträger wird die

Telematik-ID als ActorID verwendet. Für Leistungserbringerinstitutionen sowie für die Kostenträger wird keine Liste autorisierter Geräte und keine Liste von Benachrichtigungskanälen geführt. Die Eigenschaft validTo bezeichnet ein Gültigkeitsende-Datum, an welchem ein AuthorizationKey systemseitig automatisch gelöscht wird. Für den Versicherten als Eigentümer der Akte wird ein technisches Ende-Datum gleichbedeutend mit "unendlich" automatisch gesetzt. Für alle anderen AuthorizationKeys wird das Datum clientseitig belegt und definiert das Ende der vom Versicherten vergebenen Berechtigung. Mit dem optionalen `Displayname` je AuthorizationKey kann vom Versicherten ein lesbarer Name für eine Berechtigung vergeben werden, auf LE-Seite und den Abruf durch Kostenträger wird das Feld vollständig ignoriert.

Mittels der Angabe des `RecordIdentifiers` und der `ActorID` *(Telematik-ID/KVNR)* kann der zugehörige `AuthorizationKey` eines Berechtigten gefunden werden. Der `AuthorizationKey` enthält eine Liste verschlüsselter `Akten-` und `Kontextschlüssel`.

Das Element AUT-Referenz speichert in einer WhiteList die serialNumber der zur Authentisierung durch Versicherte in einer Akte verwendeten AUT- bzw. AUT_ALT-Zertifikate. Über diese Liste wird die Verwendung einer bisher unbekannten kryptografischen Identität erkannt und der Versicherte bzw. der Vertreter über den Benachrichtigungskanal informiert.

7.1 Namensräume

Für die Schnittstellen der Komponente Autorisierung werden die in der folgenden Tabelle definierten XML-Präfixe verwendet, um den Namensraum des XML-Dokumentes zu beschreiben.

Tabelle 18: Namensräume

Präfix	Namensraum
xmlns:phrs	http://ws.gematik.de/fd/phrs/AuthorizationService/v1.0
xmlns:SAML	urn:oasis:names:tc:SAML:2.0:assertion
xmlns:ds	http://www.w3.org/2000/09/xmldsig#
xmlns:xenc	http://www.w3.org/2001/04/xmlenc#

7.2 SAML-Profil und Tokeninhalte

In diesem Abschnitt werden die Inhalte der auszustellenden AuthorizationAssertion festgelegt. Eine AuthorizationAssertion wird für einen mittels AuthenticationAssertion

authentifizierten Nutzer ausgestellt. Aus dessen AuthenticationAssertion werden identifizierende Attribute in die AuthorizationAssertion übernommen.

A_14491 - Komponente Autorisierung - Inhalte AuthorizationAssertion
Die Komponente Autorisierung MUSS Autorisierungsbestätigungen als SAML2-Assertion gemäß den Festlegungen der folgenden Tabelle ausstellen:

Tabelle 19: Inhalte Autorisierungsbestätigung

Assertion Element	Usage Convention	Beschreibung
Issuer	[FQDN des ePA-Aktensystems der TI] + "/authz"	Aussteller des Tokens
Signature	[nonQES-Signatur des SAML-Tokens]	nonQES-Signatur des SAML-Tokens gemäß [SAML 2.0], die mit dem privaten Schlüssel der Ausstelleridentität C.FD.SIG der Komponente Autorisierung gemäß [gemSpec_Krypt#A_17206] erstellt wird. Das Element `ds:Signature/ds:KeyInfo/ds:X509Data/ds:X509Certificate` muss das zugehörige C.FD.SIG Zertifikat enthalten
Subject		
NameID	[SubjectDN der SMC-B] oder [SubjectDN der eGK]	wird übernommen aus der übergebenen *AuthenticationAssertion*
SubjectConfirmation		
@Method	urn:oasis:names:tc:SAML:2.0:cm:bearer	Protokoll zur Authentisierung
Conditions		
@NotBefore	[Systemzeit der Komponente Autorisierung]	Zeitpunkt, ab wann die Assertion nutzbar ist.
@NotOnOrAfter	[Systemzeit der Komponente Autorisierung + 15 Minuten]	Zeitpunkt, zu dem die Gültigkeit der Assertion endet.
AudienceRestriction		Liste der Server, für die das Token ausgestellt wird.

		Audience	[FQDN des ePA-Aktensystems der TI]	Adresse des ePA-Aktensystems aus der aktuellen Konfiguration
AuthnStatement				
	@AuthnInstant		[Systemzeit der Komponente Autorisierung]	Systemzeitpunkt bei Erstellung des Tokens
AuthzDecisionStatement				
	@Ressource		[RecordIdentifier]	RecordIdentifier der Akte, für die eine Autorisierungsbestätigung für den Nutzer ausgestellt wird.
	@Decision		Permit	
	Action		[AuthorizationType]	String gemäß der Autorisierungsentscheidung über den authentifizierten Nutzer
		@Namespace	"http://ws.gematik.de/fa/phr/v1.0"	
AttributeStatement				
	Attribute			
		Name	Resource ID "urn:oasis:names:tc:xacml:1.0:resource:resource-id"	
		AttributeValue	[RecordIdentifier]	RecordIdentifier der Akte, für die eine Autorisierungsbestätigung für den Nutzer ausgestellt wird.
	Attribute			
		Name	Gerätekennung "urn:gematik:fa:phr:1.0:device:device-id"	Nur bei mittels ActorID authentifizierten Versicherten, bei Abruf durch Leistungserbringer und Kostenträger entfällt dieses Attribut.
		AttributeValue	[DeviceID::Device]	Die DeviceID::Device ist über die ActorID des AuthorizationKey referenziert, der über die KVNR des Versicherten einer übergebenen AuthenticationAssertion gefunden

				wird.
	Attribute			
		Name	Zustand des Kontos "urn:gematik:fa:phr:1.0:status:status-id"	
		AttributeValue	[RecordState]	Wert der Eigenschaft RecordState der KeyChain des via RecordIdentifier benannten Kontos.
	Attribute			
		Name	**VersichertenID** "urn:gematik:subject:subject-id" oder **Telematik-ID** "urn:gematik:subject:organization-id"	Benutzerkennung für den die AuthorizationAssertion ausgestellt wird.
		AttributeValue	[Telematik-ID] oder [10-stelliger, unveränderlicher Teil der KVNR]	wird übernommen aus der AuthenticationAssertion

[<=]

8 Verteilungssicht

Eine Darstellung der hardwareseitigen Verteilung des Produkttyps bzw. seiner Teilsysteme und der Einbettung in die physikalische Umgebung wird nicht benötigt.

9 Anhang A – Verzeichnisse

9.1 Abkürzungen

Kürzel	Erläuterung
SAML	Security Assertion Markup Language
WS	Web Services
PKCS	Public-Key Cryptography Standards
ePA-FdV	ePA-Frontend des Versicherten, welches das das ePA-Modul FdV inkludiert
IHE	Integrating the Healthcare Enterprise
WSDL	Web Services Description Language
KVNR	Krankenversichertennummer

9.2 Glossar

Begriff	Erläuterung
HSM	Hardware Security Module, Gerät zur sicheren Speicherung kryptografischen Schlüsselmaterials
ePA-Modul FdV	Modul der dezentralen ePA-Fachlogik zur Nutzung durch den Versicherten in einem ePA-Frontend des Versicherten

Das Glossar wird als eigenständiges Dokument (vgl. [gemGlossar]) zur Verfügung gestellt.

9.3 Abbildungsverzeichnis

Abbildung 1: Anwendungsfälle der Schlüsselverwaltung nach Umgebung 10
Abbildung 2: Komponente Autorisierung, benachbarte Komponenten und Produkttypen 12
Abbildung 3: GERROR-Struktur zur Rückgabe einer Fehlermeldung 22
Abbildung 4: Informativer Ablauf des Geräte-Freischaltprozesses 59

Abbildung 5: Informativer Ablauf des Freischaltprozesses für Vertretung.....63
Abbildung 6: Informationsmodell der intern verwalteten Daten.....65

9.4 Tabellenverzeichnis

Tabelle 1: Anwendungsfälle der Schlüsselverwaltung nach Umgebung.....11
Tabelle 2: Parameter des Verwaltungsprotokolls.....21
Tabelle 3: Fehlercodes zu Fehlern gemäß Operationsdefinition.....23
Tabelle 4: Herstellerspezifische Fehlerdefinition.....23
Tabelle 5: Schnittstellen der Komponente Autorisierung.....28
Tabelle 6: I_Authorization::getAuthorizationKey Definition.....29
Tabelle 7: I_Authorization_Insurant::getAuthorizationKey Definition.....33
Tabelle 8: I_Authorization_Management::putAuthorizationKey - Definition.....36
Tabelle 9: I_Authorization_Management::checkRecordExists - Definition.....39
Tabelle 10: I_Authorization_Management::getAuthorizationList - Definition.....40
Tabelle 11: I_Authorization_Management_Insurant::putAuthorizationKey - Definition.....42
Tabelle 12: I_Authorization_Management_Insurant::deleteAuthorizationKey - Definition 46
Tabelle 13: I_Authorization_Management_Insurant::replaceAuthorizationKey - Definition 48
Tabelle 14: I_Authorization_Management_Insurant::getAuditEvents - Definition.....51
Tabelle 15: I_Authorization_Management_Insurant::putNotificationInfo - Definition.....52
Tabelle 16: I_Authorization_Management_Insurant::getAuthorizationList - Definition 55
Tabelle 17: Berechtigungstypen für AuthorizationType.....57
Tabelle 18: Namensräume.....66
Tabelle 19: Inhalte Autorisierungsbestätigung.....67
Tabelle 20: Referenzierte Dokumente der gematik.....73
Tabelle 21: Referenzierte externe Dokumente.....73

9.5 Referenzierte Dokumente

9.5.1 Dokumente der gematik

Die nachfolgende Tabelle enthält die Bezeichnung der in dem vorliegenden Dokument referenzierten Dokumente der gematik zur Telematikinfrastruktur. Der mit der vorliegenden Version korrelierende Entwicklungsstand dieser Konzepte und

Spezifikationen wird pro Release in einer Dokumentenlandkarte definiert. Version und Stand der referenzierten Dokumente sind daher in der nachfolgenden Tabelle nicht aufgeführt. Deren zu diesem Dokument jeweils gültige Versionsnummer ist in der aktuellen, von der gematik veröffentlichten Dokumentenlandkarte enthalten, in der die vorliegende Version aufgeführt wird.

Tabelle 20: Referenzierte Dokumente der gematik

[Quelle]	Herausgeber: Titel
[gemGlossar]	gematik: Glossar der Telematikinfrastruktur
[gemSysL_ePA]	gematik. Systemspezifisches Konzept ePA
[AuthorizationService.wsdl]	Schnittstellendefinition Komponente Autorisierung
[AuthorizationService.xsd]	Schemadefinition der Schnittstellen der Komponente Autorisierung
[TelematikError.xsd]	Schemadefinition Fehlermeldungen TelematikError
[PHR_Common.xsd]	Schemadefinition für übergreifende ePA-Datentypen
[gemKPT_Arch_TIP]	Konzept Architektur der TI-Plattform
[gemSpec_Perf]	Spezifikation Performancevorgaben und Mengengerüst
[gemSpec_Krypt]	Spezifikation der in der TI zulässigen kryptografischen Verfahren
[gemSpec_OID]	Spezifikation Festlegung von OIDs
[gemSpec_OM]	Spezifikation Operation und Maintenance
[gemSpec_PKI]	Übergreifende Spezifikation PKI
[gemSpec_TB_Auth]	Übergreifende Spezifikation Tokenbasierte Authentisierung
[gemSpec_TSL]	Spezifikation der Schnittstelle des TSL-Dienstes

9.5.2 Weitere Dokumente

Tabelle 21: Referenzierte externe Dokumente

[Quelle]	Herausgeber (Erscheinungsdatum): Titel

[SAML2.0]	Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0 http://docs.oasis-open.org/security/saml/v2.0/
[SOAP]	W3C (2007): SOAP Version 1.2 Part 1: Messaging Framework (Second Edition), https://www.w3.org/TR/soap12-part1/
[WSDL]	W3C: Web Services Description Language (WSDL) 1.1 https://www.w3.org/TR/wsdl.html
[WSDL11SOAP12]	W3C (2006): WSDL 1.1 Binding Extension for SOAP 1.2, https://www.w3.org/Submission/wsdl11soap12/
[WSIBP]	Web-Services Interoperability Consortium (2010): WS-I Basic Profile V2.0 (final material), http://ws-i.org/Profiles/BasicProfile-2.0-2010-11-09.html
[WS-Trust1.4]	WS-Trust 1.4 http://docs.oasis-open.org/ws-sx/ws-trust/v1.4/errata01/os/ws-trust-1.4-errata01-os-complete.pdf
[WSS]	OASIS (2006): Web Services Security: SOAP Message Security 1.1 (WS-Security 2004), http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf
[WSS-SAML]	OASIS (2006): Web Services Security: SAML Token Profile 1.1, https://www.oasis-open.org/committees/download.php/16768/wss-v1.1-spec-os-SAMLTokenProfile.pdf
[XSPA]	OASIS: Cross-Enterprise Security and Privacy Authorization (XSPA) Profile of Security Assertion Markup Language (SAML) for Healthcare Version 2.0 http://docs.oasis-open.org/xspa/saml-xspa/v2.0/saml-xspa-v2.0.html
[SGB V]	BGBl. I S.2477 (20.12.1988): Sozialgesetzbuch, Fünftes Buch Zuletzt geändert durch Art. 4 G v. 14.4.2010 I 410 Gesetzliche Krankenversicherung
[RFC-5322]	Internet Message Format - Format für E-Mail-Adressen https://tools.ietf.org/html/rfc5322
[RFC5280]	Internet X.509 Public Key Infrastructure Certificate Prüfung von Zertifikaten entlang einer Zertifikatskette (inkl. Cross-Zertifikaten) bis zu einem Vertrauensanker (Root-CA) https://tools.ietf.org/html/rfc5280#page-71