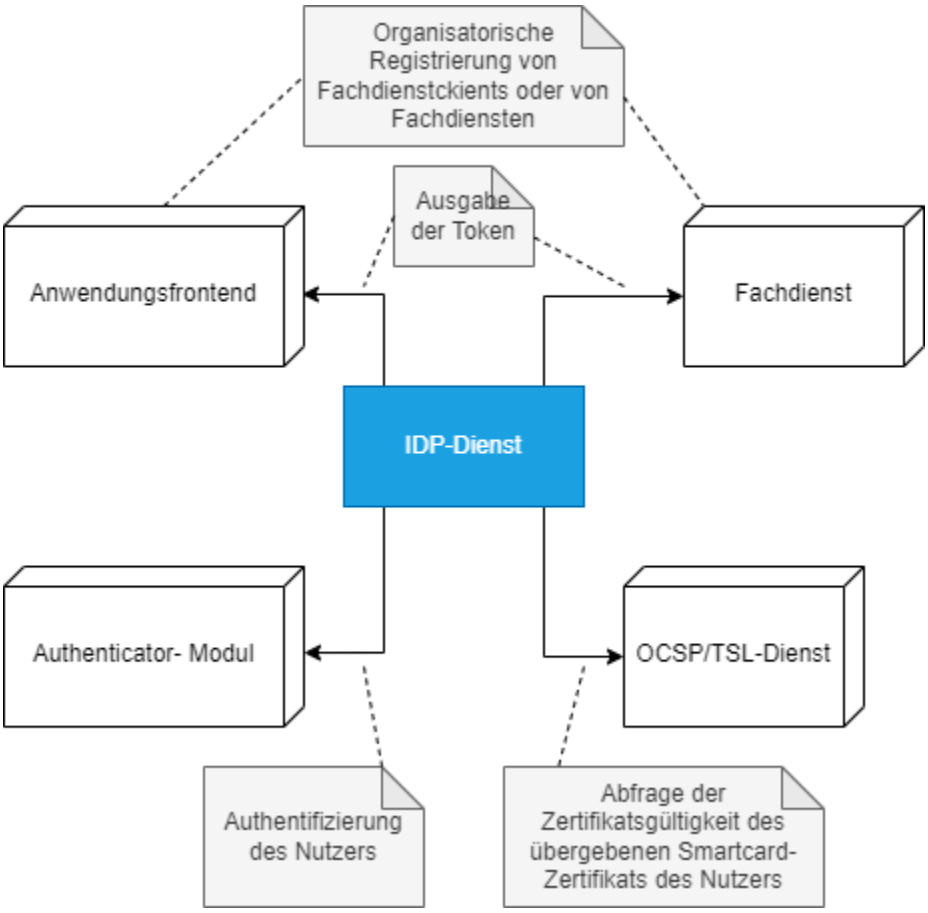
IDP-Dienst als Smartcard-IDP für Fachanwendungen

Smartcard-IDP

Die Kernfunktionalität des IDP-Dienstes als Identity Provider ist die Validierung einer Smartcard (eGK, HBA oder SMC-B), welche ein Nutzer zur Authentisierung einsetzt. Bevor ein Nutzer eine Fachanwendung nutzen kann, authentisiert er sich mit dieser Smartcard und seiner PIN. Im Zuge der Authentisierung wird das Zertifikat der Smartcard zum IDP-Dienst übertragen. Der IDP-Dienst übernimmt für den Fachdienst die Aufgabe der Authentisierung des Nutzers. Der IDP-Dienst fasst die idNummer (Telematik-ID bzw. KVNR bei Versicherten) sowie weitere für den Fachdienst notwendige Attribute in signierten JSON Web Token (*ID-Token oder Access-Token*) zusammen. Fachdienste müssen somit keine Identifikation und Authentisierung des Nutzers selbst implementieren, sondern können sich darauf verlassen, dass der Besitzer des bei ihnen vorgetragenen *Token* bereits authentisiert wurde. Des Weiteren stellt der IDP-Dienst sicher, dass die vom Nutzer vorgetragenen Attribute (aus dem Signaturzertifikat) gültig sind. Der IDP-Dienst prüft, ob das vorgetragene X.509-nonQES-Authentisierungs-Zertifikat der verwendeten Prozessor-Chipkarte (eGK, HBA oder SMC-B) für die vorgesehene Laufzeit des *Tokens* zeitlich gültig, ob dessen Integrität sichergestellt ist und ob die Smartcard nicht gesperrt wurde. Der IDP-Dienst stellt nur *Token* aus, welche auf gültigen AUT-Zertifikaten (d. h. C.CH.AUT, C.HP.AUT oder C.HCI.AUT) basieren. Dem Ablauf der Authentisierung liegt der OAuth2.0 Standard [RFC6749] zugrunde.

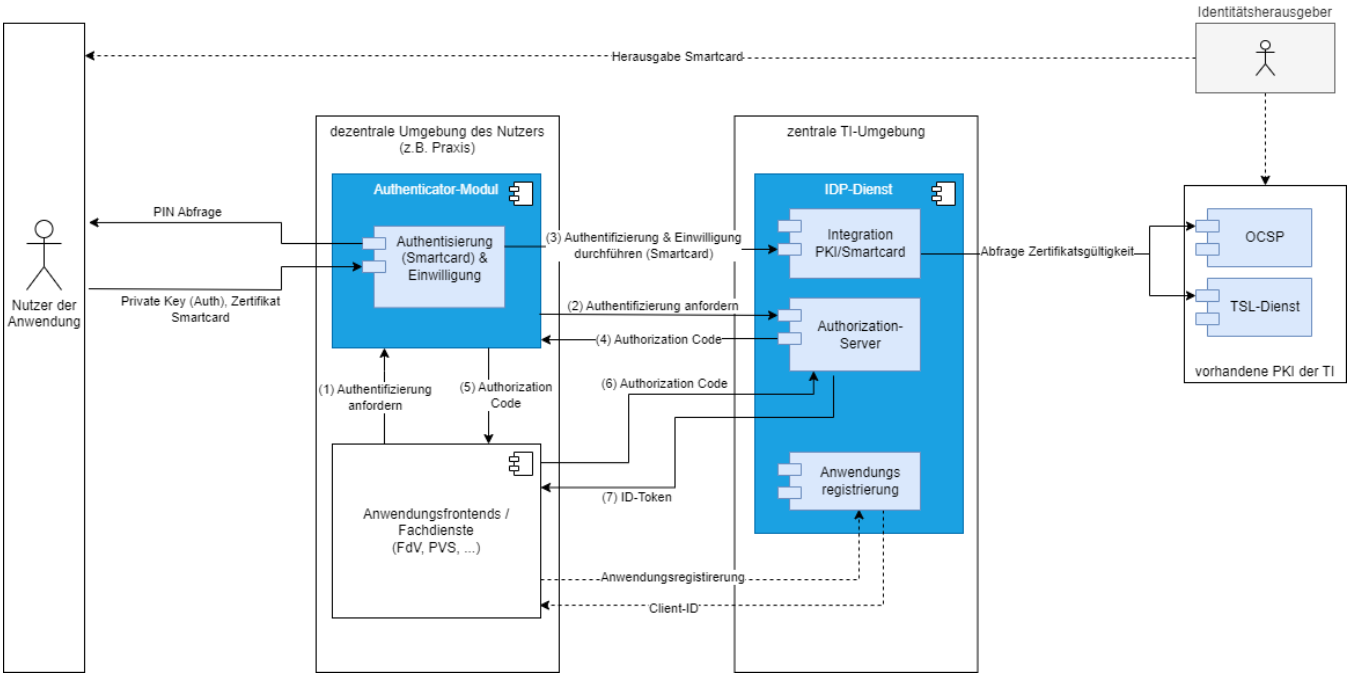
Die untere Abbildung beschreibt den Systemkontext aus Sicht des IDP-Dienstes. Anwendungsfrontends von Fachdiensten oder Fachdienste registrieren sich am IDP-Dienst über einen organisatorischen Prozess. Nur registrierte Dienste können im Ablauf der Nutzerauthentifizierung Token vom IDP-Dienst erhalten. Das Authenticator-Modul liefert die Daten zur Authentifizierung des Nutzers an den IDP-Dienst. Der IDP-Dienst prüft die Gültigkeit der Smartcard des Nutzers gegen den OCSP/TSL-Dienst der Public Key Infrastructure (PKI) der gematik und stellt *Token* für registrierte Anwendungsfrontend bzw. Fachdienste aus.



Der IDP-Dienst führt die Authentisierung des Nutzers durch und stattet diesen mit einem *ID-Token* gemäß [openid-connect-core 1.0] und einem *Acces s-Token* gemäß [RFC6749-section 1.4] aus. In der Rolle des IDP-Dienstes als Smartcard-IDP für weitere Anwendungen des Gesundheitswesens ist für diese nur das *ID-Token* relevant. Für die E-Rezept-App und den Zugriff auf den Fachdienst wird auch das *Access-Token* verwendet. Das sogenannte *SSO-Token* basierend auf [RFC7519] kommt nur im Rahmen der sicherheitsbegutachteten E-Rezept-App als eine Art *Refresh-Token* zum Einsatz. Gewählt wird aus Sicherheitsaspekten für alle Anfragen der *Authorization Code Grant* gemäß [RFC6749-section 4.1]. Zum Schutz vor verschiedenen Angriffsszenarien wird dabei der der PKCE-Flow (Proof Key for Code Exchange by OAuth Public Clients) gemäß [RFC7636]] eingesetzt. Der IDP-Dienst wird zentral und bei Bedarf auf unterschiedlicher Hardware verteilt betrieben. Das Authenticator-Modul wird grundsätzlich auf dezentraler Hardware zusammen mit dem Konnektor oder auf dem mobilen Endgerät des Nutzers betrieben. Der IDP-Dienst stellt unterschiedliche Endpunkte bereit.

Diese statisch adressierten Endpunkte umfassen:

- Discovery-Endpunkt ("OAuth 2.0 Authorization Server Metadata")
- Redirection-Endpunkt (Teil des "The OAuth 2.0 Authorization Framework") [RFC6749-section 3.1.2]
- Authorization-Endpunkt (Teil des "The OAuth 2.0 Authorization Framework") [RFC6749]
- Token-Endpunkt [RFC6749-section 3.2]
    - Ausstellung von ID-Token
    - Ausstellung von Access-Token [RFC6749-section 1.4] Nur für E-Rezept, nicht relevant für weitere Anwendungen
    - Ausstellung von SSO-Token [RFC7519] Nur für E-Rezept, nicht relevant für weitere Anwendungen



Der IDP-Dienst stellt eine Basisleistung innerhalb der TI dar und soll die sichere Identifikation der Akteure anhand der ihnen bereitgestellten Identifikationsmittel (Smartcards) ermöglichen. Der Standard lässt hierbei die Einbringung weiterer Identity Provider und unterschiedlichsten Authentisierungsverfahren zu, ohne dass Fachdienste hierfür eine Änderung der Zugangsmechanismen realisieren müssen. Die Umsetzung basiert grundsätzlich auf [OpenID Connect Core 1.0] und [OpenID Connect Discovery 1.0]. Weitere zu beachtende Standards sind:

- [RFC7519] - Request for Comments JWT (JSON Web Token),
- [RFC7515] - JWS (JSON Web Signature)
- [RFC7516] - JWE (JSON Web Encryption)
- [RFC7517] - JWK (JSON Web Key)
- [RFC7518] - JWA (JSON Web Algorithm)
- [RFC7033] - WebFinger
- [RFC6750] - OAuth 2.0 Bearer
- [RFC7521] - OAuth 2.0 Assertion
- [RFC7523] - OAuth 2.0 JWT Profile
- [RFC6749] - OAuth 2.0 Responses

Die untere Abbildung beschreibt den Systemkontext aus Sicht des IDP-Dienstes. Das Authenticator-Modul liefert die Daten zur Authentifizierung des Nutzers an den IDP-Dienst. Bei positiver Validierung – gegen den OCSP/TSL-Dienst der Public Key Infrastructure (PKI) der gematik – liefert der IDP-Dienst einen *Authorization Code* zurück. Für den Fachdienst E-Rezept liefert der IDP-Dienst ebenso einen *SSO-Token*, wodurch das eRezept-FdV für einen gewissen Zeitraum einen weiteren *Authorization Code* ohne erneute Nutzerauthentifizierung erhalten kann. Der Client registriert sich innerhalb eines organisatorischen Prozesses am IDP-Dienst und erlangt gegen Vorlage des *Authorization Code* einen *ID-Token* und einen *Access-Token*.

Akteure und Rollen

Im Systemkontext des IDP-Dienstes interagieren verschiedene Akteure (Nutzer und aktive Komponenten) in unterschiedlichen Rollen. Einige dieser Rollen sind im OIDC-Standard verankert.

Akteur	Rolle
Nutzer (z. B. Versicherte, Ärzte)	"OIDC" - Resource Owner
IDP-Dienst	"OIDC" - OpenID Provider (OP)
Authenticator-Modul / Authenticator	"OIDC" - Komponente mit Frontend zur Nutzerauthentifizierung
OCSP / TSL	Validierungsdienste
Fachdienst - Authorization-Server	"OIDC" - Relying Party (RP) bzgl. IDP-Dienst
Fachdienst - Fachliche Services mit UI (Fachdaten und -Prozesse)	Fachanwendung mit UI, welche der Nutzer nach seiner Authentifizierung nutzen möchte

Nutzer (Rolle: Resource Owner)

Der Resource Owner ist der Nutzer, welcher auf die beim Fachdienst (Resource Server) für ihn bereitgestellten Daten (Protected Resources) zugreift. Der Resource Owner verfügt über die folgenden Komponenten:

- Endgerät des Nutzers
- Authenticator-Modul
- Anwendungsfrontend

Fachdienst (Rolle: Resource Server)

Der Resource Server ist der Fachdienst, der dem Nutzer (Resource Owner) Zugriff auf seine Fachdaten (Protected Resources) gewährt. Der Fachdienst, der die geschützten Fachdaten (Protected Resources) anbietet, ist in der Lage, auf Basis von *Token* Zugriff für das Anwendungsfrontend z u gewähren. Ein solches Token repräsentiert die delegierte Identifikation des Resource Owners.

Anwendungsfrontend/Authenticator-Modul kombiniert in einer Applikation (Rolle: Client)

Der Client greift mit dem Authenticator-Modul und dem Anwendungsfrontend (OIDC Relying Party bzw. OAuth2 Client) auf Fachdienste (Resource Server) und ihre geschützten Fachdaten (Protected Resource) zu. Das Anwendungsfrontend kann auf einem Server als Webanwendung (Primärsystem als Terminalserver), auf einem Desktop-PC oder einem mobilen Gerät (z.B. Smartphone) ausgeführt werden.

IDP-Dienst (Rolle: Authorization Server)

Der Authorization Server authentifiziert den Resource Owner (Nutzer) und stellt *ID-Token* bzw. *Access-Token* (E-Rezept) für den vom Resource Owner erlaubten Anwendungsbereich (*scope*) aus, welche dieser wiederum beim Fachdienst einreicht.

Kurzbezeichnung der Schnittstellen des IDP-Dienstes:

Kurzzeichen	Schnittstelle
AUTH	Authorization-Endpunkt
TOKEN	Token-Endpunkt
REDIR	Redirection-Endpunkt
DD	Discovery Document-Endpunkt

Weitere Akteure im Kontext IDP-Dienst sind:

Fachdaten (Rolle: Protected Resource)

Die geschützten Fachdaten, welche vom Fachdienst (Resource Server) angeboten werden.

Kontext Leistungserbringer und Leistungserbringer Institutionen (LE/LEI)

Leistungserbringer und Leistungserbringer-Institutionen nutzen aktuell dezentrale Telematikinfrastruktur-Komponenten mit Kartenleser und Konnektoren.
In der Systemlandschaft der LE/LEI wird analog zum Authenticator-Modul im Versichertenkontext eine Komponente benötigt, welche in Kommunikation mit dem IDP-Dienst die Nutzerauthentifizierung durchführt. Zu diesem Zweck hat die gematik den Authenticator entwickelt. Der Authen ticator authentisiert Inhaber von HBA bzw. SMC-B über die im Kartenleser gesteckte und durch PIN-Eingabe freigeschaltete Karte gegenüber beliebigen Anwendungen und Diensten.

Details zur Installation, Konfiguration und Verwendung sind über die Authenticator-Dokumentation verfügbar.

Alternativ kann die Funktion des Authenticator-Modul auch direkt in ein Primärsystem integriert und mit der Anwendungslogik gekoppelt werden. Dies ist beim E-Rezept der Fall.

Registrierung von Fachanwendungen

Um ein Anwendungsfrontend nutzen zu können, muss dieses am IDP-Dienst registriert sein. Die Registrierung der Fachdienste am IDP-Dienst erfolgt über einen organisatorischen Prozess.
Diese Registrierung erfolgt einmalig für die Anwendung bzw. den Dienst und muss nicht bei Updates wiederholt werden. Die Registrierung des Fachdienstes beinhaltet dabei auch die Abstimmung der Claims und die Gültigkeitsdauer der erstellten Token, wobei der Fachdienst seinen Bedarf an den gewünschten Attributen erklärt. Anpassungen an den Claims bedürfen einer erneuten Abstimmung und Registrierung.

IDP-Registrierungsformular:

gematik

Registrierung Anwendung am IDP - Beantragung

Service Umgebung	RU
Antragstyp	Neuanlage
Zulassungsschlüssel	ZLS_WANDA_Klicken oder tippen Sie hier, um die Ziffern Ihres Zulassungsschlüssels einzugeben.
Für welche Anwendung benötigen Sie die Fachdienst-Registrierung? (kurze Beschreibung des Anwendungsfalls)	
erwartete Anfragen (pro 5 Minuten)	
Antragsteller	Vorname Nachname des Antragstellers Organisationsname Straße Hausnummer PLZ Stadt E-Mail-Adresse des Antragstellers
Antwort-URL (redirect_uri für den Redirect vom IDP zum Fachdienst)	
Empfänger-URI (ID des Ziel-Dienstes)	
Claims	ACCESS_TOKEN (A_19985-01) / ID_TOKEN (A_20706) Folgende personenbezogenden Claims werden beantragt (bitte gewünschte (Mehrfach-) Auswahl ankreuzen): ☐ professionOID ☐ Vorname ☐ Nachname ☐ Organisationsname ☐ ID-Nummer (Telematik-ID)
Lebensdauer eines ausgestellten Access Token (in Sekunden; max. 300 Sekunden)	

Kontakt: IDP-Registrierung@gematik.de

Ablaufbeschreibung IDP-Dienst als Smartcard-IDP

Vorbereitende Maßnahmen

Vorbedingung	Beschreibung
Kartenausgabe	Durch den Identitätsherausgeber wurde dem Nutzer eine Smartcard (eGK, HBA, SMC-B) + PIN ausgestellt.
Registrierung	Über einen organisatorischen Prozess hat sich ein Anwendungsclient oder eine Fachanwendung bei IDP-Dienst registriert. Ob sich alle Clients (Frontends) einer Fachanwendung (z.B. E-Rezept) oder nur der Fachdienst beim IDP-Dienst registriert hängt von der konkreten Implementierung des jeweiligen Fachdienstes ab. Zur Registrierung werden unter anderem Angaben über den vom Fachdienst benötigten Scope (Umfang der nötigen Attribute) am IDP-Dienst hinterlegt. Der IDP-Dienst erzeugt bei der Registrierung eine Client_ID. Über die Client_ID wird während des Ablaufs das registrierte Anwendungsfrontend bzw. der registrierte Fachdienst vom IDP-Dienst identifiziert.
Konfiguration	Der Fachdienst und das Authenticator-Modul haben das Discovery Dokument eingelesen und kennen damit die Uniform Ressource Identifier (URI) und die öffentlichen Schlüssel der vom IDP-Dienst angebotenen Endpunkte. Fachdienste sollten zusätzlich eine regelmäßige OCSP-Validierung auf die Signaturzertifikate des IDP-Dienstes durchführen.

Das Authenticator-Modul bzw. die Authenticator-Anwendung stellen die Verbindung zwischen Kartenleser mit SmartCard, dem Konnektor und dem IDP-Dienst her. Die Nutzerauthentifizierung erfolgt durch stecken der SmartCard und PIN-Eingabe am Kartenleser.

Für die Implementierung gibt es folgende Möglichkeiten

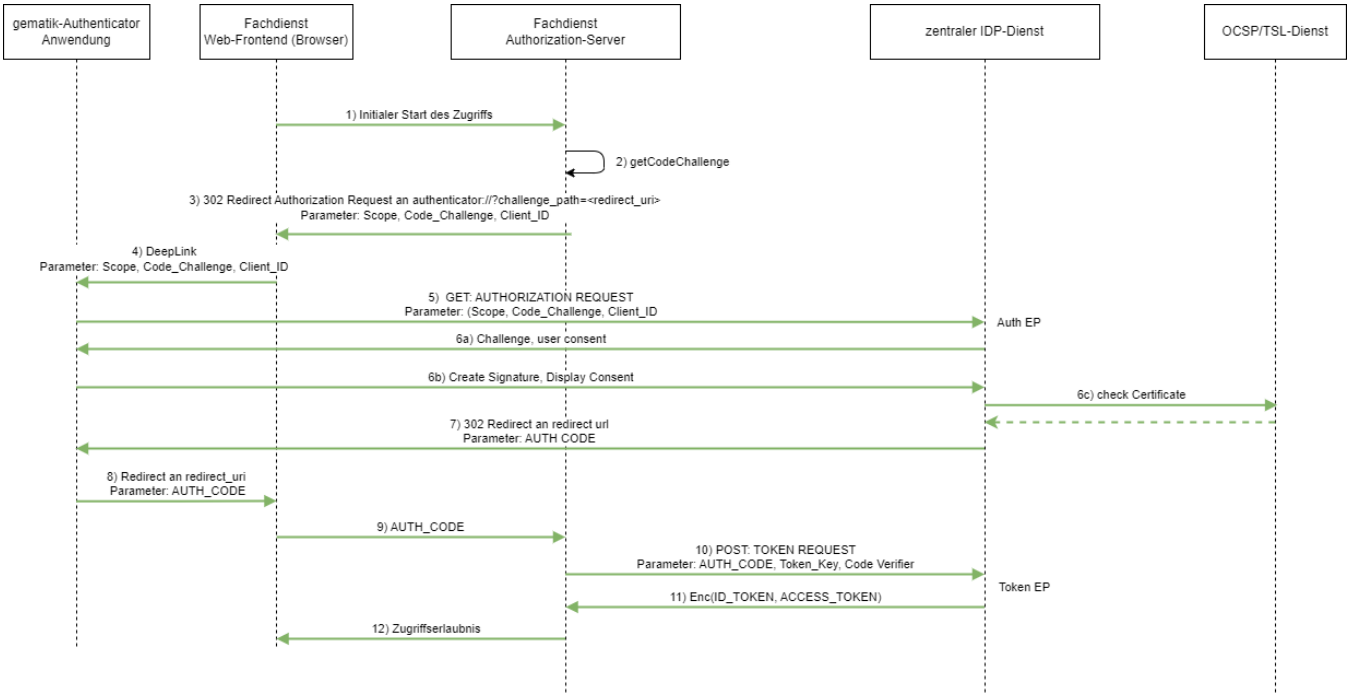
- Nutzung des gematik-Authenticator zur Nutzerauthentifizierung

  Der gematik-Authenticator "... ist eine Desktop-Anwendung mit grafischer Benutzerschnittstelle, welche zunächst unter Windows - später auch unter Mac OS und Linux - lauffähig ist und aus Anwendungen (typisch: Web-Anwendungen) heraus aufgerufen wird. Seine Aufgabe ist die Authentisierung des Nutzers an einem Identity Provider (IdP) mittels Smartcards der TI (HBA, SMC-B) und Konnektors/KT (2-Faktor-Authentisierungsverfahren). Für die Authentifizierung und Delegation wird das OAuth2-basierte Protokoll von OpenID Connect unterstützt, um mit dem zentralen IDP der TI (vormals: Smartcard IDP) oder auch einem anwendungsspezifischen IdP (im Falle des Organspenderegisters) zu interagieren."

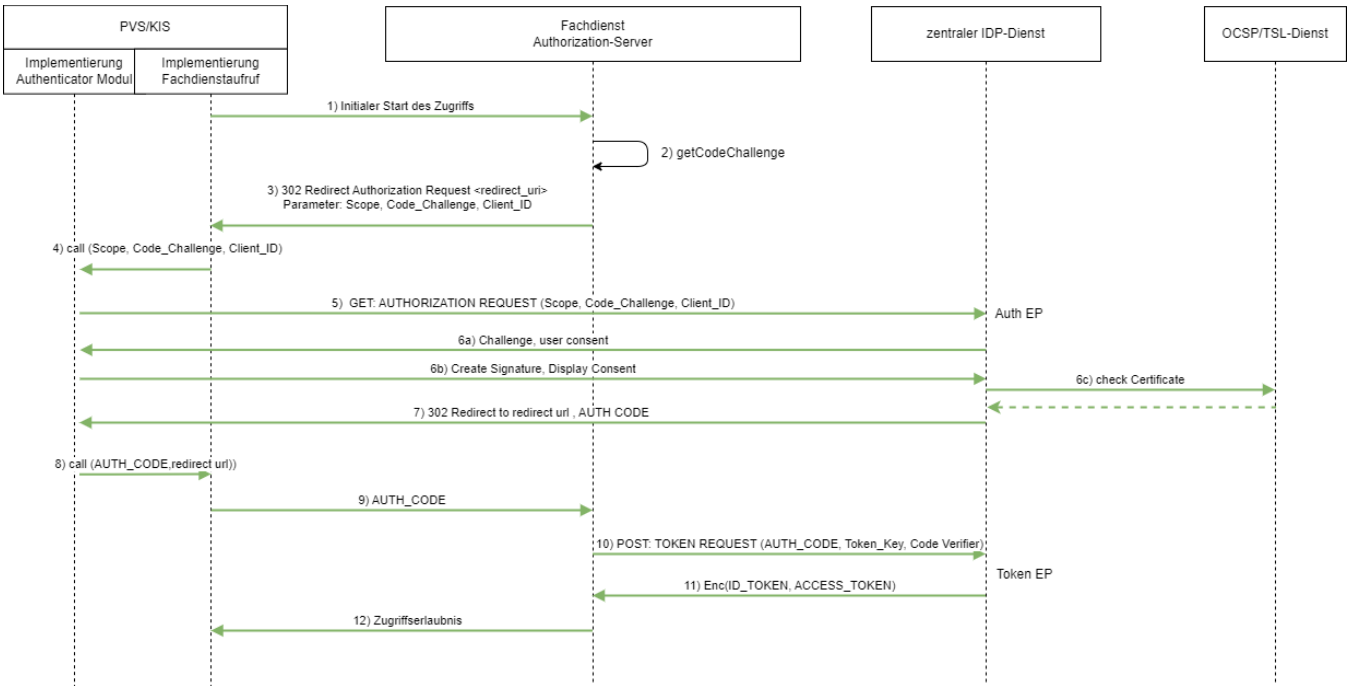
- Eigenimplementierung eines Authenticator-Moduls

Bei der Eigenimplementierung muss die Funktionalität des gematik-Authenticator selbst implementiert und in das eigene System integriert werden. Im Kontext der Umsetzung des E-Rezept wurden eigene Authenticator Lösungen in PVS und KIS sowie die E-Rezept-App integriert.

Ablauf der Nutzerauthentifizierung mit der gematik-Authenticator-Anwendung



Ablauf der Nutzerauthentifizierung aus einem Praxissystem (PVS) oder Klinikinformationssystem (KIS) mit integriertem Authenticator-Modul



Schritt	Beschreibung
1	Das Anwendungsfrontend überträgt seine Initiale Zugriffsanfrage an den Fachdienst. Idealerweise als OAuth2 Request

2	Der Authorisierungsserver des Fachdienstes erzeugt sich einen zufälligen *Code Verifier* und bildet darüber den Hash *Code Challenge* mit dem Hash-Algorithmus S256. Dann formuliert er den vollständigen OpenID Authorization Request an den IDP-Dienst. (Optional erzeugt er sich zusätzlich eine zufällige *nonce*)
3	Der Authorisierungsserver des Fachdienstes antwortet auf die Initiale Zugriffsanfrage des Anwendungsfrontend mit den notwendigen Parametern um einen vollständigen Authorization Request an den Authorization-Endpunkt des IDP-Dienstes zu stellen. (hier bietet sich ein 302 Redirect an)
4	(a) gematik-Authenticator: Das Anwendungsfrontend übermittelt den Authorization Request an das vorgesehene Authenticator-Modul. Im Falle des gematik-Authenticators gelten hier dessen Schnittstellenbeschreibungen (gematik Authenticator SDK Dokumentation) - Aufruf mittels "authenticator://" (b) Authenticator-Eigenimplementierung: Das Anwendungssystem (PVS/KIS) ruft sein Authenticator-Modul (je nach technischer Umsetzung) auf.
5	Das Authenticator-Modul überträgt den Authorization Request inklusive der *Code Challenge* weiter an den Authorization-Endpunkt des IDP-Dienstes.
6	(a) Der Authorization-Endpunkt stellt den einmalig den Consent (Zustimmung des Nutzers zur Verarbeitung der angezeigten Daten) zu den mit dem entsprechenden Fachdienst vereinbarten Claims zusammen und überträgt *Challenge-Token* und Consent-Abfrage *User-Consent* zum Authenticator-Modul. (b) Das Authenticator-Modul fordert den Nutzer einmalig zur Consent-Freigabe auf mittels Smartcard und PIN-Eingabe und verwendet die PIN, um die *Challenge-Token* von der Smartcard signieren zu lassen. Das Authenticator-Modul überträgt dann das signierte *Challenge-Token* mit dem Smartcard-Zertifikat, verschlüsselt mittels PuK_IDP_ENC, an den IDP-Dienst. (c) Der IDP-Dienst prüft das Smartcard-Zertifikat des Nutzers gegen OCSP/TSL-Dienst.
7	Der Authorization-Endpunkt entschlüsselt und validiert die signierte Challenge *Session-ID*, *Challenge* und *Signatur*. Die Signatur wird anhand des im "x5c"-Header mitgelieferten Authentifizierungszertifikats der Smartcard validiert. Die Validierung des Zertifikats erfolgt durch Prüfung gegen OCSP/TSL-Dienst der gematik-PKI. Der Authorization-Endpunkt erstellt den *Authorization Code* und überträgt diesen entsprechend der für diese *Client_Id* registrierten *redirect_uri* an das Authenticator-Modul.
8	(a) gematik-Authenticator: Das Authenticator-Modul überträgt den *Authorization Code,* der *redirect_uri* folgend, an den Fachdienst. (b) Authenticator-Eigenimplementierung: Das Anwendungssystem (PVS/KIS) verarbeitet den *Authorization Code* je nach technischer Umsetzung.
9	(a) gematik-Authenticator: Entweder das Anwendungsfrontend reagiert über einen eigenen "Deeplink" auf die Rückgabe des *Authorization Code* oder dieser wird in einem eigenen Browserfenster zum Authorisierungsserver des Fachdienstes übermittelt welcher ihn anhand des "state" einer Anfrage zuordnen kann. (b) Authenticator-Eigenimplementierung: Das Anwendungssystem (PVS/KIS) übergibt den *Authorization Code* je nach technischer Umsetzung an den aufrufenden Fachdienst.
10	Der Authorisierungsserver des Fachdienstes erzeugt sich einen AES256-"Token-Key", verknüpft ihn mit dem *Code Verifier* zum *Key Verifier* und sendet diesen unter Nutzung des öffentlichen Schlüssels PUK_IDP_ENC verschlüsselt zusammen mit dem *Authorization Code* zum Token-Endpunkt des IDP-Dienstes.
11	Der Token-Endpunkt entschlüsselt und validiert den *Key Verifier*, entnimmt aus diesem den *Code Verifier* und gleicht diesen mit der *Code Challenge* zu diesem dem *Authorization Code* ab. Der Token-Endpunkt erzeugt die erforderlichen Token, signiert sie mit seinem privaten Schlüssel PrK_IDP_SIG und verschlüsselt sie mit dem „Token-Key“ des Anwendungsfrontends, welchen er dem *Key Verifier* entnommen hat.| Der Token-Endpunkt überträgt die *ID-Token* und *Access-Token* an den Authorisierungsserver des Fachdienstes
12	Der Fachdienst entschlüsselt das *ID-Token* entsprechend dem gewählten „Token-Key“ . Der Fachdienst validiert das *ID-Token* anhand des öffentlichen Schlüssels *PuK-Token* des Token-Endpunktes. (Optional validiert er zusätzlich die im Authorization Request übermittelte *nonce*). Fachdienste mit TI-Zugriff sollten zusätzlich eine OCSP-Validierung auf die Signatur durchführen. (Anderen Fachdiensten steht der OCSP-Responder hierfür bald auch im Internet zur Verfügung) Der Fachdienst zieht die Claims (d. h. die Key/Value-Paare im Payload eines Tokens) aus dem *ID-Token* und gibt bei positiver Validierung für das Anwendungsfrontend Zugriff auf die Fachdaten frei. Idealerweise über ein eigenes OAuth2 *Access-Token*.

Begriffsdefinition

Für die Signatur des Discovery Document und der Token werden Zertifikate der Komponenten-PKI der TI verwendet. Die folgende Tabelle enthält die Abkürzungen (für die privaten Schlüssel PrK und für öffentliche Schlüssel PUK) der verschiedenen Endpunkte des IDP-Dienstes und deren Verwendung.

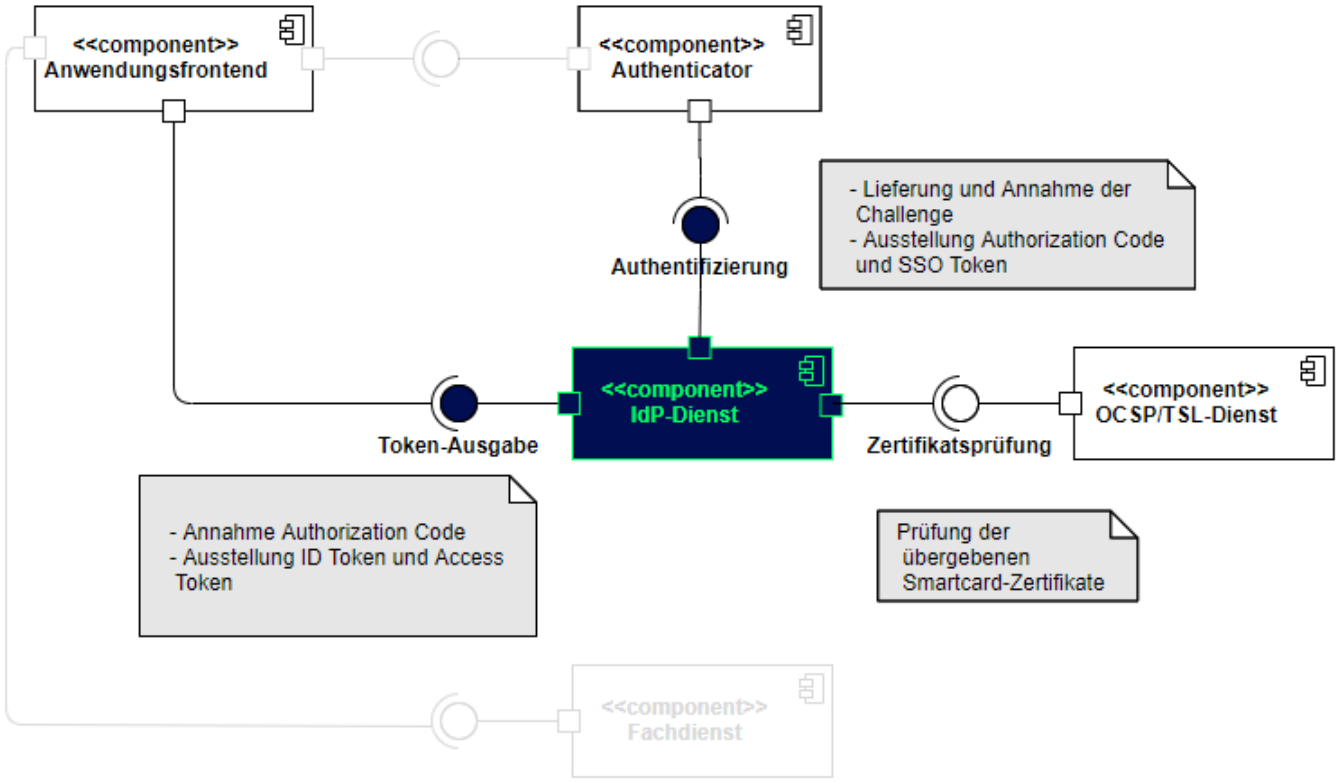
Endpunkt / URI	PuK / PrK	Aufgaben
Authorization-Endpunkt (AUTH) / URI_AUTH (authorization_endpoint)	PuK_IDP_SIG	• für die Signaturprüfung des *Challenge-Token* durch das Authenticator-Modul • kodiert in einem FD.SIG-Zertifikat
	PuK_IDP_ENC	• für die Verschlüsselung der signierten Challenge durch das Authenticator-Modul
	PrK_IDP_SIG	• zum Signieren des *Challenge-Token*
	PrK_IDP_ENC	• zum Entschlüsseln der signierten Challenge
Discovery-Endpunkt (DISC) / URI_DISC	PuK_DISC_SIG	• für die Signaturprüfung des Discovery Document durch den Authorization-Server • kodiert in einem FD.SIG-Zertifikat
	PrK_DISC_SIG	• zum Signieren des Discovery Document
Token-Endpunkt (TOKEN) / URI_TOKEN (token_endpoint)	PuK_IDP_SIG	• für die Signaturprüfung des *ID-Token* durch den Authorization-Server • kodiert in einem FD.SIG-Zertifikat
	PuK_IDP_ENC	• für die Verschlüsselung des *Key Verifier* durch durch den Authorization-Server
	PrK_IDP_SIG	• zum Signieren des *ID-Token* und des *Access-Token*
	PrK_IDP_ENC	• für die Entschlüsselung des *Key Verifier*

Werden alle Teildienste auf einem Server gemeinsam betrieben, so können diese dasselbe Schlüsselmaterial für Verschlüsselung bzw. Signaturen verwenden. Werden Teildienste auf unterschiedlichen physischen oder logischen Servern betrieben, so sind die Endpunkte mit eigenem Schlüsselmaterial auszustatten.
Die URL des Discovery Document URI_DISC stellt somit den zentralen Anlaufpunkt dar, anhand dessen alle weiteren „statischen“ Dienste (Endpunkte des IDP-Dienstes) adressiert werden können.
Bei allen extern genutzten Schlüsseln handelt es sich um ECC-Schlüsselpaare der Kurve brainpoolP256r1. Für IDP_ENC ist im Gegensatz zu den anderen beiden Schlüsseln keine Bestätigung als Zertifikat vorgesehen. Die maximale Einsatzdauer des Schlüsselpaares liegt analog zum IDP_SIG und DISC_SIG bei maximal 5 Jahren.

Für die Transportverschlüsselung (TLS) werden Internet-Zertifikate von Mitgliedern des CA/Browser Forum (https://cabforum.org/members/) eingesetzt.

Schnittstellenbeschreibung des IDP-Dienstes

Der IDP-Dienst bietet zahlreiche Schnittstellen gegenüber unterschiedlichen Akteuren inner- und außerhalb der TI an, weswegen es notwendig ist, die einzelnen Schnittstellen so zu beschreiben, dass andere Akteure deren Funktionsweise leichter verstehen können. Nachfolgende Abbildung skizziert die Schnittstellen des IDP-Dienstes. Komponenten und Schnittstellen, welche nicht direkt vom IDP-Dienst genutzt werden, sind in der Abbildung grau hinterlegt.



Die erste Token-bezogene Anfrage an den Authorization Server des IDP-Dienstes geht am Authorization-Endpunkt [RFC6749-section-3.1] ein. Das Authenticator-Modul reicht dort am Endpunkt den Consent mit der Challenge ein, mit welchem die Token erstellt werden sollen, und erhält den Authorization-Code zurück, falls die Prüfung der signierten Challenge und die Prüfung des übergebenen Smartcard-Zertifikats am OCSP/TSL-Dienst positiv ausfallen.

Gemäß Schritt 10 in der Ablaufbeschreibung reicht der Fachdienst den Authorization-Code am Token-Endpunkt [RFC6749-section-3.2] # section-3.2] des IDP-Dienstes ein. Der IDP-Dienst überprüft den Authorization-Code und stellt bei positiver Validierung einen ID-Token und einen Access-Token aus. Bei der ersten Kontaktaufnahme erzeugt der Authorization Server des IDP-Dienstes die Subject-Session, welche im weiteren Verlauf als Zeitpunkt der letzten Authentisierung gegen die eGK oder den HBA gewertet wird. Basierend darauf dürfen weitere Access-Token für andere Fachdienste ausgegeben werden, wenn das jeweils vorliegende *claim* durch die dem Authorization Server des IDP-Dienstes vorliegenden Informationen bedient werden kann. Ist der Zeitpunkt der letzten Authentisierung zu lange her oder wird das Authenticator-Modul zum ersten Mal gestartet, muss eine Authentisierung erfolgen. Die Implementierung des [E-Rezept] kann als Referenz der Anbindung einer Fachanwendung an den IDP-Dienst dienen.

Die folgenden Kapitel dienen der Darstellung der Zusammenhänge.

Funktionsmerkmale des IDP-Dienstes

Authorization Server Metadata (Discovery Document)

Der Authorization Server des IDP-Dienst dient dazu, bestehende Identitäten zu prüfen und das Prüfungsergebnis in einer einheitlichen Form abgestimmt und durch zusätzliche Mechanismen gesichert bereitzustellen. Basis dieser Dienstleistung ist ein vertrauenswürdiges Verzeichnis, aus welchem hervorgeht, an welchen Schnittstellen dieser Dienst oder seine Teildienste erreichbar sind, wie diese Schnittstellen abgesichert sind und woher man die zur Etablierung der gewünschten Sicherheit erforderlichen Materialien beziehen kann. Gemäß dem verwendeten Standard OpenID Connect mit OAuth 2.0 kommen JSON Web Token (JWT), JSON Web Encryption (JWE), JSON Web Signature (JWS) und JSON Web Key (JWK) zum Einsatz.

Um nutzenden Anwendungen eine einheitliche Bezugsquelle für die Adressierung von Schnittstellen zu schaffen, werden die für alle Akteure grundlegenden Schnittstellen im sogenannten Discovery Document zusammengefasst und dort unter der URI_DISC gemäß [RFC8414 "OAuth 2.0 Authorization Server Metadata"] veröffentlicht. Alle Akteure, welche den IDP-Dienst nutzen wollen, sind angehalten, dieses Discovery Document zu lokalisieren, herunterzuladen, zu prüfen und den Inhalt in den geplanten Betrieb einzubeziehen.

Der IDP-Dienst gibt alle verwendeten Adressen in Form von URL gemäß [RFC1738] an und veröffentlicht diese in einem Discovery Document gemäß [RFC8414 # section-2] im Internet. Das im Internet bereitgestellte Discovery Document stellt die URI der angebotenen Fachdienste im Internet mit dort auflösbaren Adressen bereit. Es gibt je ein internes und externes (public) "Discovery Document". Diese unterscheiden sich in den darin angebotenen URI, welche gleichlautend im Host-Anteil auf unterschiedliche Domänen bzw. Top-Level-Domain (TLD) verweisen. Der IDP-Dienst prüft alle von ihm im Discovery Document angebotenen URL ständig auf deren Erreichbarkeit.

Aufbau des Discovery Documents

Der IDP-Dienst stellt ein über das Internet erreichbares Discovery Document gemäß [RFC8414] und [openid-connect-discovery-1_0] zur Verfügung. D as Discovery Document enthält Informationen, wie die Schnittstellen des IDP-Dienstes zu erreichen sind. Ebenso enthält das Discovery Document die Links, unter denen die öffentlichen Schlüssel für Verschlüsselung und Signatur abfragbar sind. Das Discovery Document des IDP-Dienstes enthält die in der Tabelle dargestellten Attribute, deren Belegung z. T. fest vorgegeben ist:

Name	Wert	Beispiel	Anmerkungen
issuer	URL	https://idp-ref.app.ti-dienste.de	URL, unter welchem der IDP-Dienst erreichbar ist
jwks_uri	URI	https://idp-ref.app.ti-dienste.de/certs	URI für den Abruf von „PUK_IDP_ENC“ sowie des öffentlichen Schlüssels und des Zertifikats von „PUK_IDP_SIG“
uri_disc	URL	https://idp-ref.app.ti-dienste.de/.well-known/openid-configuration	URL, unter welcher das Discovery Document bereitgestellt wird
uri_pair	URI	https://idp-pairing-ref.zentral.idp.splitdns.ti-dienste.de/pairings	URI des Authorization-Endpunktes für Requests mit Pairing-Daten zum einrichten einer Gerätebindung (wird durch das E-Rezept FdV verwendet)
authorization_endpoint	URL	https://idp-ref.app.ti-dienste.de/auth	URI des Dienstes und des öffentlichen Verschlüsselungsschlüssels des Authorization-Endpunktes gemäß [RFC6749]
sso_endpoint	URL	https://idp-ref.app.ti-dienste.de/auth/sso_response	URI des Authorization-Endpunktes für Requests mit SSO-Token (dieser wird nur durch das E-Rezept FdV verwendet.)
auth_pair_endpoint	URL	https://idp-ref.app.ti-dienste.de/auth/alternative	URL des Authorization-Endpunkts zur Authentisierung mit einer Gerätebindung (dieser ist nur im Internet verfügbar und wird durch das E-Rezept FdV verwendet)
token_endpoint	URL	https://idp-ref.app.ti-dienste.de/token	URI des Token-Endpunktes gemäß [RFC6749]
uri_puk_idp_enc	URI	https://idp-ref.app.ti-dienste.de/certs/puk_idp_enc	URI der JWK Objekte für Schlüssel und Zertifikates zur Verschlüsselung
uri_puk_idp_sig	URI	https://idp-ref.app.ti-dienste.de/certs/puk_idp_sig	URI der JWK Objekte für Schlüssel und Zertifikates zur Signatur
kk_app_list_uri	URL	https://idp-ref.app.ti-dienste.de/directory/kk_apps	URL zum Abruf aller registrierten Krankenkassen-APPs (wird nur durch das E-Rezept FdV verwendet)
third_party_authorization_endpoint	URL	https://idp-ref.app.ti-dienste.de/extauth	Der sektorale IDP muss hier den IDP-Dienst mit all jenen "*kk_app_redirect_uri*"-Adressen registrieren, die er unterstützt (nur für IDP-Dienst Fasttrack).
subject_types_supported	["pairwise"]	-	Das JSON-Array enthält die Liste der vom IDP-Dienst unterstützen subject types. Der IDP-Dienst unterstützt nur subject type mit dem Identifier "pairwise". Das bedeutet, dass für jeden Client ein anderer Wert für *sub* (Attribut im ID-Token für das Pseudonym des Nutzers) mit dem Ziel generiert wird, eine Korrelation der Aktivitäten von Nutzern zu verhindern.
id_token_signing_alg_values_supported	["BP256R1"]	-	Das JSON-Array enthält die Liste der vom IDP-Dienst unterstützen JWS Signatur-Algorithmen (alg values) zur Verschlüsselung der *claims* im ID-Token.
response_types_supported	["code"]	-	Das JSON-Array enthält die Liste der vom IDP-Dienst unterstützen OAuth 2.0 response_type. Der IDP-Dienst unterstützt nur response_type "code".
scopes_supported	[<scope Fachdienst>, <scope Fachdienst>, <scope Fachdienst>]	-	Das JSON-Array enthält die Liste der vom IDP-Dienst unterstützten *scopes*. Jeder beim IDP-Dienst registrierte Fachdienst hat einen eindeutigen *scope*. Alle diese *scopes* müssen im discovery document in der Liste *scopes_supported* enthalten sein.
response_modes_supported	["query"]	-	Das JSON-Array enthält die Liste der vom IDP-Dienst unterstützten response_mode. Der IDP-Dienst unterstützt nur response_mode "query".
grant_types_supported	["authorization_code"]	-	Das JSON-Array enthält die Liste der vom IDP unterstützten Grant Type. Der IDP unterstützt nur Grant Type "authorization_code".

acr_values_supported	["gematik-ehealth-loa-high"]	-	Der IDP unterstützt nur acr_values "gematik-ehealth-loa-high".
code_challenge_methods_supported	["S256"]	-	Das JSON-Array enthält die Liste der vom IDP unterstützten Proof Key for Code Exchange (PKCE) [RFC7636] Methoden. Der IDP unterstützt nur "S256".
exp	Alle time Werte in Sekunden seit 1970, RFC 7519 Sect.2	16753320006 Beispielhafte Gültigkeit von 24h	Gültigkeitsdauer des Discovery Documents.
iat	Alle time Werte in Sekunden seit 1970, RFC 7519 Sect.2	1675245600 01.02.2023 10:00:00	Zeitpunkt der Ausstellung des Discovery Documents.