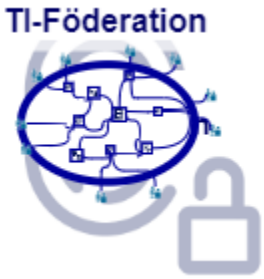
Fachanwendungen der TI-Föderation



Systemüberblick

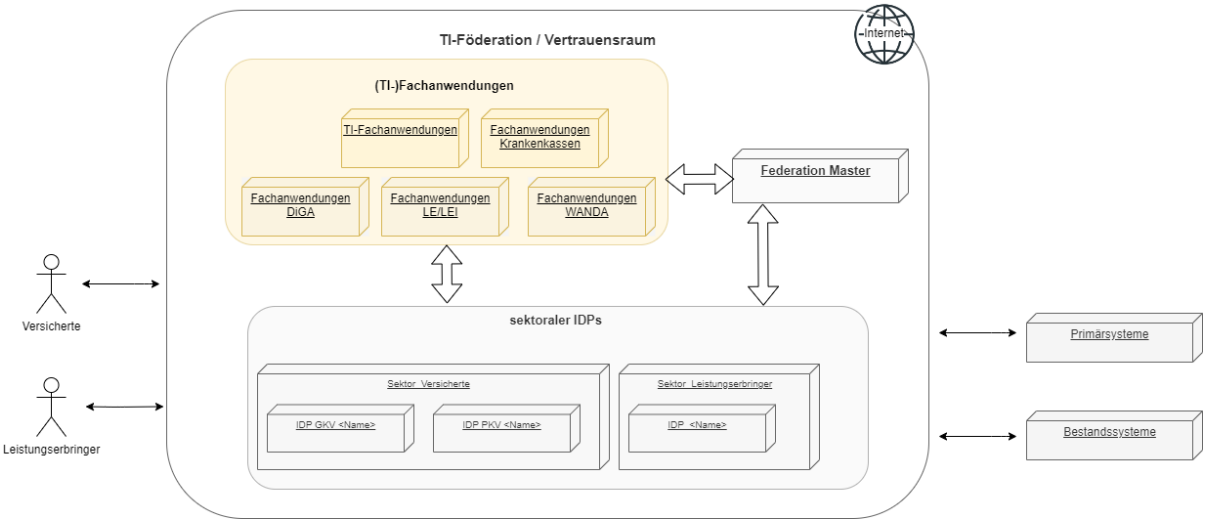
Unter Fachanwendungen im Kontext der TI-Föderation werden medizinische Anwendungen verstanden, die zur Nutzerauthentifizierung die sektoralen Identity Provider der TI-Föderation nutzen. Fachanwendungen (teilweise auch als Fachdienste bezeichnet) sind nach aktueller Definition:

- TI-Anwendungen (E-Rezept, elektronische Patientenakte, KIM, TIM)
- Digitale Gesundheitsanwendungen (DiGA)
- Fachanwendungen der Krankenkassen
- Fachanwendungen für Leistungserbringer und Institutionen
- Weitere Anwendungen für den Datenaustausch in der Telematikinfrastruktur (WANDA)

Ein weiteres Kriterium der Fachanwendungen ist deren Zielgruppe und damit verbunden die im Rahmen einer Nutzerauthentifizierung zielgruppenspezifische Daten- und Sicherheitsanforderungen:

- gesetzlich Versicherte
- privat Versicherte
- Leistungserbringer (Ärzte, Hebammen, Physiotherapeuten, u.ä.)
- Leistungserbringerinstitutionen (Krankenhäuser, Praxen, Apotheken u.ä.)

Zum jetzigen Zeitpunkt sind Fachanwendungen vorgesehen, die gesetzlich zur Integration der TI-Föderation verpflichtet sind. Dies sind insbesondere TI-Anwendungen und Digitale Gesundheitsanwendungen (DiGA). Es werden aktuell in Abstimmungen mit dem BMG, den gesetzlichen Krankenkassen und weiteren Stakeholdern Anforderungen und Kriterien für weitere Anwendungsgruppen erarbeitet.



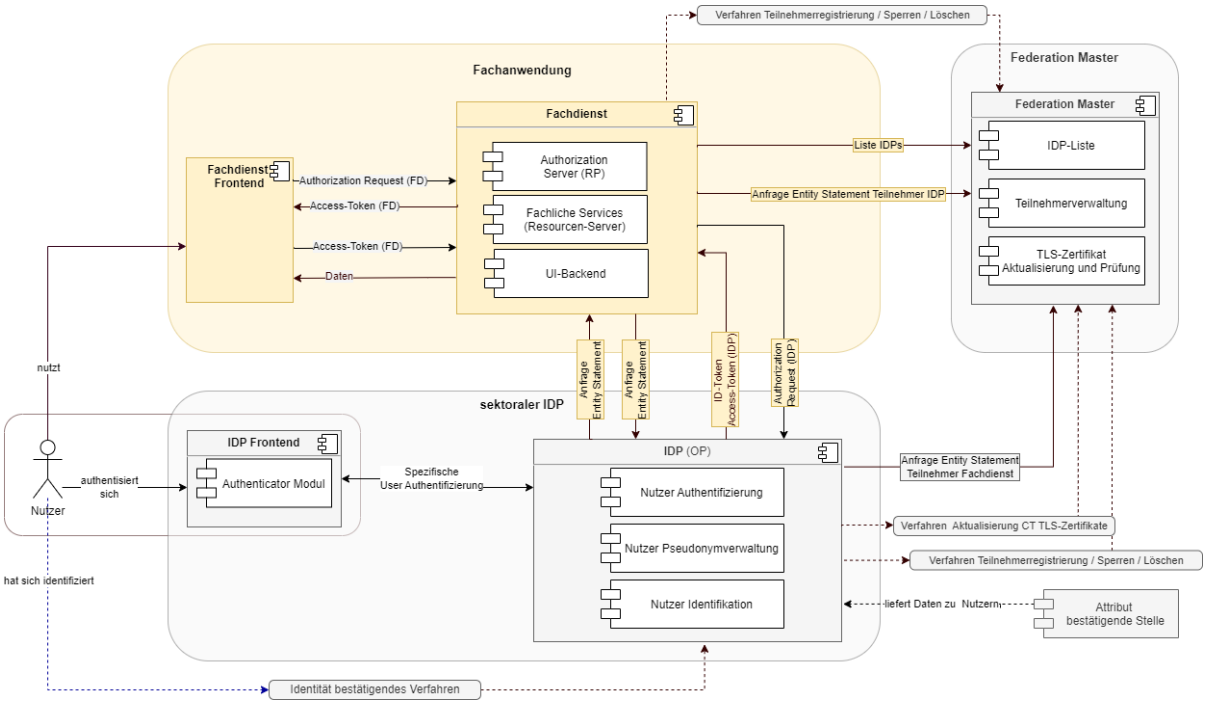
Eine Fachanwendung in der TI-Föderation besteht im Allgemeinen aus den folgenden Komponenten (siehe auch Fachliche & technische Grundlagen sowie Standards):

- Authorization-Server: Entsprechend dem OAuth2.0-Standard ist die Autorisierung eines Nutzers getrennt vom eigentlichen Zugriff auf Fachdaten und -prozesse. Der Authorization-Server des Fachdienstes implementiert diese Autorisierung. Dazu muss der Nutzer der Fachanwendung erst einmal authentisiert werden. Die Nutzerauthentisierung erfolgt nach dem OpenID Connect (OIDC) Standard. Im Kontext dieses Standards ist der Authorization-Server des Fachdienstes eine Relying Party (RP). Er stößt die Authentisierung des Nutzers beim sektoralen IDP an und erhält als Ergebnis einen Authorization Code, den er gegen ein *ID-Token* (und *Access-Token)* beim sektoralen

IDP eintauscht. Der Authorization-Server des Fachdienstes verwendet die Informationen aus dem *ID-Token* für die Feststellung der Zugriffsrechte des authentisierten Nutzers auf die Ressourcen des Fachdienstes. Ist der Datenzugriff berechtigt, stellt der Authorization-Server des Fachdienstes ein eigene *Access-Token* (und ggf. ein *Refresh-Token)* für das Anwendungsfrontend aus. Erst dann kann der Nutzer über das Anwendungsfrontend auf die Anwendungsfälle des Fachdienstes zugreifen.

- UI-Backend: Das UI-Backend nimmt Anfragen der UI entgegen und stellt der UI Daten zur Anzeige bereit. Je nach technischer Implementierung wird die gesamte UI mit den Daten im UI-Backend erstellt und dem Anwendungsfrontend präsentiert (serverside rendered progressive web-app) oder das UI-Backend liefert Daten und die UI Aufbereitung erfolgt im Anwendungsfrontend (native APP, single page web-app). Das UI-Backend sollte auf der Serverseite des Fachdienst getrennt von Authorization-Server und den Business-Services des Fachdienstes implementiert werden.
- Resource-Server: Nach OAuth2.0-Standard implementiert der Resource-Server die eigentliche fachliche Logik (Business-Services) des Fachdienstes.

Schnittstellen



Die folgende Tabelle enthält einen kurzen Überblick über die Schnittstellen der Komponenten des Fachdienstes. Detailliertere Informationen zum den Abläufen und den Schnittstellen findet man im App2App-Flow, Web2App-Flow und 2-Geräte-Flow.

Komponenten des Fachdienstes	Schnittstelle	Komponente	Beschreibung
Anwendungsfrontend			
	Initiale Nutzeraktion zur Nutzung der Fachanwendung (anwendungsspezifisch)	UI-Backend	Der Nutzer möchte den Fachdienst verwenden. Eine Interaktion des Nutzers mit dem Anwendungsfrontend löst ein Ereignis aus, um den Nutzer für den Zugriff zu autorisieren.
UI-Backend / Anwendungsfrontend			
	Anfrage Liste registrierter IDPs (anwendungsspezifisch)	Authorization-Server	Das UI-Backend bzw. Anwendungsfrontend erfragt beim Authorization-Server des Fachdienstes die Liste der in der TI-Föderation registrierten sektoralen Identity Provider.
	Authorization-Request	Authorization-Server	Das UI-Backend bzw. Anwendungsfrontend startet die Nutzerautorisierung. Dazu sendet es dem Authorization-Server des Fachdienstes einen Authorization-Request mit einer Code-Challenge sowie den zur Nutzerauthentisierung gewählten IDP. Als Ergebnis liefert der Authorization-Server nach erfolgreicher Nutzerauthentisierung und Prüfung der Berechtigungen (Autorisierung) dem UI-Backend (bzw. je nach technischer Implementierung dem Anwendungsfrontend) ein Access-Token als Zugriffserlaubnis auf die Business-Services des Fachdienstes.

	Datenaustausch (anwendungsspezifisch)	Resource Server	Der Datenaustausch zwischen UI-Backend bzw. Anwendungsfrontend und den Business-Services des Fachdienstes ist anwendungsspezifisch. Allerdings wird das Access-Token in der Schnittstellenkommunikation benötigt. Der Resource Server prüft damit die Zugriffsberechtigung.
Authorization-Server			
	Anfrage Entity Statement Federation Master	Federation Master	Der Authorization-Server erfragt über die /.well-known Schnittstelle des Federation Master dessen Entity Statement
	Anfrage der Liste registrierter IDPs	Federation Master	Der Authorization-Server erfragt über die im Entity Statement veröffentlichte Schnittstelle des Federation Master die Liste der in der TI-Föderation registrierten Identity Provider.
	Teilnehmerauskunft zu Identity Provider	Federation Master	Der Authorization-Server erfragt über die im Entity Statement veröffentlichte Schnittstelle des Federation Master Auskunft zu einem registrieren Identity Provider.
	Anfrage Entity Statement Identity Provider	Identity Provider	Der Authorization-Server erfragt über die /.well-known Schnittstelle des Identity Provider dessen Entity Statement
	Authentication-Request	Identity Provider	Aufgrund einer Nutzeranfrage über das UI-Backend oder Anwendungsfrontend sendet der Authorization-Server einen "Pushed-Authorization-Request" an die im Entity Statement veröffentlichte Schnittstelle des Identity Provider. Dieser leitet den Prozess der Nutzerauthentisierung ein. Dazu muss der Nutzer Interaktionen mit der Clientkomponente des Identity Provider - dem Authenticator-Modul - durchführen. Der Prozess beginnt mit einer Antwort auf dem PAR an den Authorization-Server des Fachdienstes und endet bei erfolgreicher Authentisierung mit der Rückgabe eines Authorization-Code an den Authorization-Server des Fachdienstes.
	Abruf des ID-Token	Identity Provider	Der Authorization-Server des Fachdienstes stellt einen Request an den Identity Provider in dem er diesem den Authorization-Code übergibt. Nach Prüfung erstellt der Identity Provider ein ID-Token (und ein Access-Token) und gibt dem Authorization-Server diese als Antwort auf dessen Request zurück. Aufgrund der Informationen aus dem ID-Token prüft der Authorization-Server die Rechte des anfragenden Nutzers zur Ausführung der Business-Services des Fachdienstes. Ist die Prüfung positiv, so antwortet der Authorization-Server des Fachdienstes auf die ursprüngliche Nutzeranfrage vom UI-Backend oder Anwendungsfrontend mit einem eigenen Access-Token. Dieses Access-Token muss UI-Backend oder Anwendungsfrontend bei jeder Kommunikation mit Business-Services - also dem Ressource-Server - zum Nachweis der Berechtigung mit übergeben.
Resource Server			
	Datenaustausch (anwendungsspezifisch)	UI-Backend / Anwendungsfrontend	Der Resource Server prüft bei jedem Request aus dem UI-Backend oder Anwendungsfrontend des übergebene Access-Token. Bei gültigem Token führt der Resource Server die angeforderte Funktion aus. Bei ungültigem Token antwortet der Resource Server dem UI-Backend / Anwendungsfrontend mit einer Fehlermeldung.

Anforderungen an einen Fachdienst für die Teilnahme an der Föderation

Die Anforderungen für die Telematik Infrastruktur sind in Spezifikationen der gematik veröffentlicht. Für einen Fachdienst in der TI-Föderation sind vor allem diese Spezifikationen relevant

- gemSpec_IDP_FD - Anforderungen an einen Fachdienst der TI-Föderation
  - Alle Kapitel sind wichtig für die Umsetzung einen Fachdienstes für die TI-Föderation.
  - Ausnahme: Kapitel "5 Anbindung eines Fachdienstes an den zentralen IDP-Dienst" - Dieser Spezialfall für die Anmeldung von Leistungserbringern und Leistungserbringerinstitutionen ist derzeit nur für wenige Fachdienste relevant.
- gemSpec_IDP_Sek - Anforderungen an den sektoralen IDP der TI-Föderation
  - Kapitel "4 Funktionsmerkmale" - enthält Informationen zu den Schnittstellen des sektoralen IDP, sowie zur Identifikation und Authentisierung von Nutzern
  - Kapitel "5 Anforderungen an Authenticator-Module sektoraler IDPs" - enthält Informationen zum Authenticator-Modul
  - Kapitel "7 Anhang B - Abläufe"
- gemSpec_IDP_FedMaster - Anforderungen an den Federation Master der TI-Föderation
  - Kapitel "3 Funktionsmerkmale" enthält wichtige Informationen zu den Anwendungsfällen und Schnittstellen des Federation Master "IDP-Liste bereitstellen", "Entity Statement bereitstellen" und "Schlüssel verwalten".
  - Kapitel "4.2 - Organisatorische Prozesse am Federation Master" enthält wichtige Informationen zur Teilnehmerregistrierung.
- gemSpec_IDP_Frontend - Spezifikation Identity Provider - Frontend
  - Kapitel "9 Nutzung sektoraler Identity Provider" - enthält wichtige Informationen zur Kommunikation eines Anwendungsfrontend mit dem Authenticator-Modul eines sektoralen IDP.
- gemSpec_Krypt - kryptografische Anforderungen
- gemSpec_Perf - Anforderungen an Performance und betriebliche Anforderungen

Im Folgenden werden einige wichtige Informationen aus den Spezifikationen erläutert.

Entity Statement

Jeder Teilnehmer der TI-Föderation muss Auskunft über sich in Form eines standardkonformen Entity Statement abgeben. Ein Entity Statement ist immer ein signiertes JWT (Beispiel des JWT des Entity Statement eines sektoralen IDP - siehe Umgebungen, Referenzimplementierungen, Codebeispiele sektoraler IDP). Die Anforderungen an das Entity Statement eines Fachdienstes können in den gematik Spezifikationen gemSpec_IDP_FD und den Anlagen von gemSpec_IDP_Sek nachgeschlagen werden.

Die Tabelle zeigt ein Beispiel für das Entity Statement eines Fachdienstes (Auszug aus gemSpec_IDP_Sek Kapitel 7):

Name	Werte	Anmerkungen
iss	URL	URL des Fachdienstes (seine client_id bei Anfragen an sektorale IDPs)
sub	URL	URL des Fachdienstes (=iss)
iat	Alle time Werte in Sekunden seit 1970, RFC 7519 Sect.2	Das Entity Statement ist gültig ab
exp	Alle time Werte in Sekunden seit 1970, RFC 7519 Sect.2	Das Entity Statement ist gültig bis
jwks	JWKS Objekt	Schlüssel für die Signatur des Entity Statement
authority_hints	[string]	iss Bezeichnung des Federation Master
metadata {		Begin des Blocks für die Metadaten des Fachdienstes
openid_relying_party {		Begin des Blocks für die Metadaten des Fachdienstes als Relying Party
signed_jwks_uri	URL	Unter der URL kann das Schlüsselset des Fachdienstes als JWKS abgerufen werden. Es enthält Schlüssel für die Signatur des Entity Statement, die TLS Client Schlüssel und Zertifikate (x5c, use = sig) und für die Verschlüsselung der ID_TOKEN (use = enc). Der claim ist optional - gemäß https://openid.net/specs/openid-connect-federation-1_0.html#name-openid-connect-and-oauth2-m
jwks	Liste von JWKS Objekten	Optional - gemäß https://openid.net/specs/openid-connect-federation-1_0.html#name-openid-connect-and-oauth2-m für den Fall das ein Fachdienst signed_jwks_uri nicht anbieten kann.
organization_name	String	Optional: Name der Organisation die hinter dem Fachdienst steht
client_name	String	Name des Fachdienstes (redundant zum claim "name" in den "Federation Entity" claims)
logo_uri	URL	Optional: URL, unter der das Logo der Organisation oder des Fachdienstes abgerufen werden kann
redirect_uris	[URL]	Liste aller URLs, zu denen die Antwort des Identity Provider über redirect propagiert werden kann. Ein Authorization-Request des Fachdienstes muss im eine URL aus dieser Liste als Parameter redirect_uri enthalten.
response_types	[code]	Der claim "response_types" enthält nur "code"
client_registration_types	[automatic]	gemäß OpenID Connect Federation 1.0 (section-4.1)
grant_types	[authorization_code]	OpenID Connect Dynamic Client Registration 1.0 (section-2)
require_pushed_authorization_requests	true	OAuth 2.0 Pushed Authorization Requests (section-6)
token_endpoint_auth_method	self_signed_tls_client_auth	
default_acr_values	"gematik-ehealth-loa-high" oder "gematik-ehealth-loa-substantial"	Das default_acr_values gibt an, auf welchem Vertrauensniveau der Nutzer authentisiert werden muss wenn im Authorization Request der claim "acr_values" nicht belegt ist.

id_token_signed_response_alg	ES256	Weitere Werte sind möglich
id_token_encrypted_response_alg	ECDH-ES	Weitere Werte sind möglich
id_token_encrypted_response_enc	A256GCM	Weitere Werte sind möglich
scope	[string]	Der claim "scope" ist ein String mit space-delimited scope values. Die scope values sagen aus, welche Informationen über den Nutzer der Fachdienst im ID-Token übermittelt bekommen möchte. Die von der TI-Föderation unterstützten scopes sind in der Spezifikation gemSpec_IDP_Sek definiert. Beispiele für unterstützte scopes sind • openid scope value nach OIDC-Spezifikation • urn:telematik:display_name Anzeigename des Nutzers (bsteht i.d.R. aus Vorname + Nachname) • urn:telematik:versicherter Telematik-ID für Versicherte
}		Ende des Blocks für die Metadaten des Fachdienstes als Relying Party
federation_entity {		Begin des Blocks für die Metadaten des Fachdienstes als Teilnehmer der Föderation
name	string	Optional: Der Name des Fachdienstes wird z. B. in der Consent-Freigabe des Anwenders genutzt. Der claim ist redundant zum claim "client_name".
contacts	string	Optional: Kontaktdaten, z.B. für Support-Anfragen
homepage_uri	URL	Optional: URL der Homepage der Organisation oder des Fachdienstes
}		Ende des Blocks für die Metadaten des Fachdienstes als Teilnehmer der Föderation
}		Ende des Blocks für die Metadaten

Unter der signed_jwks_uri (oder alternativ als zusätzliches jwks unterhalb von metadata/openid_relying_party) finden sich dann die weiteren Schlüssel des Fachdienstes

Schlüssel für die TLS Client Authentisierung sind erkennbar am gesetztem x5c Wert und use=sig.

Schlüssel für die Verschlüsselung der übertragenen ID_Token durch den IDP sind erkennbar an use=enc.

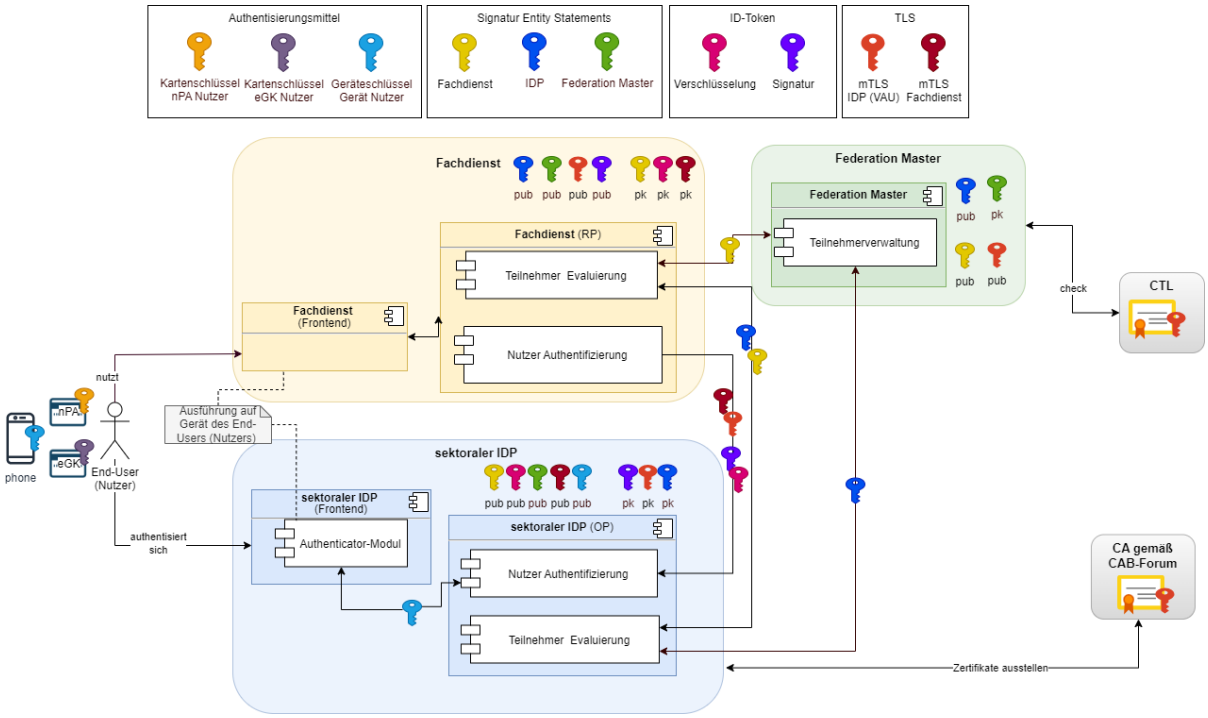
Name	Werte	Beispiel	Anmerkungen
keys {			
kty		EC	
kid		Fachdienst007-42 / Fachdienst007-69	
crv		P-256	
x		qAOdPQROkHfZY1daGofOmSNQWpYK8c9G2m2Rbkpbd4c /	
y		G_7fF-T8n2vONKM15Mzj4KR_shvHBxKGjMosF6FdoPY / ...	
use		sig / enc	nach JSON Web Key (section-4.2) Der Fachdienst listet sowohl sig als auch enc Schlüssel
x5c		MIIDQjCCAiqgAwIBAgIGATz /FuLiMA0GCSqGSIb3DQEBBQUAMGIxCzAJBgNVBAYTAlVTMQswCQYDVQQIEwJ DTzEPMA0GA1UEBxMGRGVudmVyMRwwGgYDVQQKExNQaW5nI...	Zertifikat für die TLS Client Authentisierung des Fachdienst gegenüber dem IDP
}			
iss	URL	"https://Fachdienst007.de"	URL des IDP (variabel je Mandant /Kasse)
iat	Alle time Werte in Sekunden seit 1970, RFC 7519 Sect.2,	1645484401	

Sichere Kommunikation

Die Kommunikation zwischen den Teilnehmern der Föderation ist über kryptografische Verfahren gesichert.

- Die Kommunikation zwischen Fachdienst und Identity Provider erfolgt über mTLS
- Entity Statements sind signiert
- Die vom Federation Master abfragbare Liste der registrierten Identity Provider ist signiert
- Das unter der im claim "signed_jwks_uri" genannten URL ladbare Schlüsselset ist signiert
- Ausgestellte ID-Token und Access-Token sind signiert und verschlüsselt

Die Schlüssel und Zertifikate sind über das Entity Statement des jeweiligen Teilnehmer verfügbar (Root-claim "jwks" und Metadata-claim "signed_jwks_uri"). Die Angaben über die supporteten Signatur- und Verschlüsselungsalgorithmen sind ebenfalls über die Entity Statements der Teilnehmer öffentlich zugänglich.



Weitere Maßnahmen erhöhen die Sicherheit in der Teilnehmerkommunikation gegen Angriffsszenarien :

- state [OAuth 2.0 for Native Apps (section-8.9)] Der Ersteller des Authorization-Request generiert einen Zufallswert, der als state-Parameter Teil des Authorization-Request ist. Die Antwort auf den Request muss den state-Parameter unverändert an den Aufrufer zurückgeben. Der Aufrufer verifiziert den Wert. Ein state-Parameter wird sowohl in der Kommunikation des Anwendungsfrontend mit dem Authorization-Server des Fachdienstes, als auch in der Kommunikation des Authorization-Server des Fachdienstes mit dem Identity Provider gesetzt.
- nonce Der Authorization-Server des Fachdienstes generiert einen Zufallswert, der als nonce-Parameter Teil des Authorization-Request ist. Dieser nonce-Wert muss vom Identity-Provider als claim im ID-Token an den Authorization-Server des Fachdienstes zurückgegeben werden. Dieser prüft den Rückgabewert gegen den Ausgangswert.
- PKCE [RFC7636 - Proof Key for Code Exchange by OAuth Public Clients] Sowohl in der Kommunikation des Anwendungsfrontend mit dem Authorization-Server des Fachdienstes, als auch in der Kommunikation des Authorization-Server des Fachdienstes mit dem Identity Provider wird PKCE eingesetzt. Dies verhindert das Angreifer welche trotz der weiteren Maßnahmen in den Besitz eines Auth_Code geraten sind diesen gegen ein valides Token eintauschen können.

Zugang zur Testumgebung

Fachdienste können die Integration der TI-Föderation testen. Hierzu steht sowohl eine Referenzimplementierung des Federation Masters als auch ein von der gematik entwickelte Implementierung eines sektoralen IDPs bereit. Um Ende-zu-Ende Tests durchzuführen, ist eine Registrierung in der Testumgebung notwendig. Alle Informationen hierzu finden Sie hier: Fachdienste Test-Umgebungen

Perspektivisch werden auch Tests gegen Referenzimplementierungen der sektoralen IDPs der Kostenträger inkl. Authenticator-Apps möglich sein. Die Bereitstellung beider Komponenten befindet sich aktuell noch in Abstimmung.