Addendum „Authentifizierung an der RST-Schnittstelle“ zur XML-Spezifikation V2.1.1

Version	1.0
Klassifizierung	S1 - öffentlich
Status	freigegeben
Gültig ab	04.03.2024

Herausgegeben von:

Referat 126 – Implantateregister Deutschland
Bundesministerium für Gesundheit
https://www.bundesgesundheitsministerium.de/implantateregister-deutschland.html

Rochusstraße 1, 53123 Bonn
Postanschrift: 53107 Bonn

Helpdesk der Register- und Vertrauensstelle:
support-implantateregister@d-trust.net
Tel.: 030 2598-4316

Inhaltsverzeichnis

1. Einführung

Dieses Addendum ersetzt den in Kapitel 2.2 der Technischen Dokumentation zur XML-Spezifikation in der Version 2.1.1 beschriebenen Prozess zur Authentifizierung. Es beschreibt einen aktualisierten Prozess der Authentifizierung zur Datenübermittlung an das IRD über die Registerstelleschnittstelle (RST-Schnittstelle). Dieser Prozess wird ab der 10. Kalenderwoche (ab 04.03.2024) zunächst in der Referenz- und anschließend in der Produktivumgebung des IRD verfügbar sein. Die Aktualisierung ist notwendig aufgrund der Weiterentwicklung des genutzten Gematik-Plugins der Bundesdruckerei. Der bisherige Authentifizierungsprozess wird mindestens bis Ende 2024 weiterhin unterstützt. Es wird rechtzeitig informiert, ab wann dies nicht mehr der Fall sein wird. Dennoch wird die Umstellung auf den hier beschriebenen Prozess so bald wie möglich empfohlen.

Die Anmeldung über die Implantateregister-Meldeanwendung erfolgt wie bisher mit dem gematik-Authenticator.

2. Authentifizierung an der RST-Schnittstelle

Grundsätzliche Voraussetzung ist die Nutzung der Telematikinfrastruktur (Details s. Kapitel 4.4 der Technischen Dokumentation 2.1.1). Bei der Datenübermittlung ist für die Authentifizierung eine beim IRD registrierte SMC-B-Karte erforderlich (s. Kapitel 2.2 der Technischen Dokumentation 2.1.1). Ein eHBA ist für die Datenübermittlung nicht erforderlich, wohl aber für die erstmalige Beantragung einer SMC-B-Karte bei einem zugelassenen Kartenherausgeber.

Erfolgt die Datenübermittlung über eine Software der Gesundheitseinrichtung, so ist der Kern des Verfahrens analog zu dem des E-Rezeptes. Es muss für die Datenübermittlung an das IRD jedoch erweitert werden, da das IRD einen eigenen Identity Provider (IDP) verwendet: den IDP der Registerstelle.

Bei Nutzung eines Webclients ist die Installation des gematik-Authenticators mindestens in der Version 4.1 erforderlich.

Dieses Kapitel beschreibt den neuen Ablauf der Authentifizierung bei der Registerstelle des IRD basierend auf dem OAuth2-Authorization Flow für Software der Gesundheitseinrichtung. Zur Meldung einer implantatbezogenen Maßnahme muss die Software auf den Webservice der Registerstelle zugreifen. Für den Zugriff ist ein Token vom IDP der Registerstelle notwendig. Dieser IDP setzt

zusätzlich auf die von der Bundesdruckerei entwickelte und beim Organspende-Register eingesetzte Technologie[1] auf.

Die Änderungen im Vergleich zum bisherigen Ablauf betreffen Schritt 4 (signed challenge) sowie die in den Schritten 5 und 6 aufgerufenen Endpunkte in Bezug auf die „nextStepUrl“. Diese Änderungen sind im Text rot markiert.

Folgende URL werden im Laufe des Prozesses aufgerufen:

- IRD-URL
  - RU: ru.ir-d.de
  - PU: rst.ir-d.de
- Webservice-URL = <IRD-URL>/validation[2]
- Gematik-IDP-URL
  - RU: idp-ref.zentral.idp.splitdns.ti-dienste.de
  - PU: idp.zentral.idp.splitdns.ti-dienste.de

Weitere Parameter für die Requests an die IDP:

- Webservice-Client-ID: z. B. WS_BI
- Gematik-Client-ID:
  - RU: GEMBundeImpvv8er3v8R
  - PU: GEMBundeImpzv4vrz3bP

Bei jedem Request zum gematik-IDP muss gemäß der gematik Spezifikation[3] der Header „User-Agent“ wie folgt gesetzt werden:

- <Produktname>/<Produktversion> <Herstellername>/<Gematik-Client-ID>

Für die <Gematik-Client-ID> verwenden Sie entsprechend der Umgebung (Referenz- oder Produktivumgebung) den jeweils oben angegebenen Wert. Für die übrigen Bezeichnungen nehmen Sie den Namen und die Version ihres Softwareprodukts, sowie den Namen Ihres Unternehmens als Hersteller.

[1] https://github.com/Bundesdruckerei-GmbH/Gematik-IDP
[2] Die Authentifizierung ist über alle Endpunkte möglich. Der Endpunkt „validation“ ist hier nur beispielhaft angegeben.
[3] https://gemspec.gematik.de/docs/gemSpec/gemSpec_eRp_AdV/latest/ Regel A_20014-01

Die Authentifizierung läuft wie folgt ab:

1. Die Software der Gesundheitseinrichtung sendet einen Authentication-Request an den IDP der Registerstelle per GET-Methode im folgenden Format:

   GET https://<IRD-URL>/auth/realms/IRD/protocol/openid-connect/auth?
   response_type=code
   &client_id=<Webservice-Client-ID>
   &redirect_uri=https://<Webservice-URL>
   & code_challenge=<Code Verifier[4] SHA256-hashed Base64-encoded>
   &code_challenge_method=S256
   &state=<random String> (optional)
   &scope=openid

2. Es finden einige Redirects statt. Die Ausführung der Redirects muss von der Software der Gesundheitseinrichtung ermöglicht werden. Nach den Redirects wird als Response HTTP 200 mit Content-Type text/html zurückgegeben.
   Innerhalb dieser Datei ist der URL-encodierte Eintrag direkt nach „URL=authenticator://?challenge_path=<Gesuchter https Aufruf>“ herauszufiltern.

   Beispiel für solch einen Eintrag (für eine bessere Lesbarkeit ohne URL Encoding):

   <meta http-equiv="refresh" content="1;
   URL=authenticator://?challenge_path=https://<Gematik-IDP-URL>/auth?
   client_id=<Gematik-Client-ID>
   &response_type=code
   &redirect_uri=https://<IRD-URL>/auth/realms/IRD/broker/gematik-idp/endpoint/result
   &state=88d24c3a-e4b6-4f90-a191-9be0a9bfab63__WS_BI__c6jbgfo5jmA
   &scope=irdIdp openid Institutions_ID
   &code_challenge=nogDvv9vB-XIzLxrz8TlWJXNHQiQdX4qbqGoS_sAwMI
   &code_challenge_method=S256
   &nonce=lqaaW-P23tiUH8eT2wpDVg
   &callback=direct"
   id="openAuthenticator">

   Bitte beachten:
   a. Im Scope ist hier „Institutions_ID“ angegeben. Dies besagt, dass für die nachfolgende Kommunikation mit dem gematik-IDP die Daten der SMC-B-Karte zu benutzen sind. Für die weitere Verarbeitung ist „Institutions_ID“ aus der URL zu entfernen.
   b. Am Ende der URL findet sich „&callback=direct“. Für die weitere Verarbeitung ist dieser Teil ebenfalls zu entfernen.
   c. Der Wert des Parameters „state“ ist für spätere Verwendung zwischenzuspeichern (im obigen Beispiel entspräche dies dem Wert 88d24c3a-e4b6-4f90-a191-9be0a9bfab63__WS_BI__c6jbgfo5jmA).
   d. Die „code_challenge“ in der obenstehenden URL ist nicht die gleiche wie im Schritt 1. Der Wert in der obenstehenden URL darf nicht geändert werden. Wenn der Wert geändert wird, schlägt die Authentifizierung durch den gematik-IDP fehl.

[4] Der Code Verifier ist ein random-String mit 43 bis 128 Zeichen. Der Code Verifier sollte eine hohe Entropie aufweisen.

3. Die Software der Gesundheitseinrichtung sendet den in Schritt 2 ermittelten Request an den gematik-IDP:

   GET https://<Gematik-IDP-URL>/auth?
   client_id=<Gematik-Client-ID>
   &response_type=code
   &redirect_uri=https://<IRD-URL>/auth/realms/IRD/broker/gematik-idp/endpoint/result
   &state=88d24c3a-e4b6-4f90-a191-9be0a9bfab63__WS_BI__c6jbgfo5jmA
   &scope=irdIdp openid
   &code_challenge=nogDvv9vB-XIzLxrz8TlWJXNHQiQdX4qbqGoS_sAwMI
   &code_challenge_method=S256
   &nonce=lqaaW-P23tiUH8eT2wpDVg

   Bitte beachten:
   a. in dem obigen Request ist „Institutions_ID“ aus dem Parameter „scope“ entfernt!
   b. In dem obigen Request ist ebenfalls „&callback=direct“ aus dem URL entfernt!

   Als Antwort wird vom gematik-IDP bei HTTP-Status 200 eine JSON-Datei mit Challenge und User Consents zurückgeliefert.

4. Gemäß den Vorgaben der gematik wird aus der Antwort auf Schritt 3 von der Software der Gesundheitseinrichtung eine „signed-challenge“ erstellt und an den gematik-IDP geschickt:

   POST https://<Gematik-IDP-URL>/auth
   HTML Form URL Encoded: application/x-www-form-urlencoded
   Form item: "signed_challenge"=
   "eyJleHAiOjE2ODE4MjYzMTIsImVwayI6eyJrdHkiOiJFQyIsImNydiI6IkJQLTI1NiIsIngiOiJjMUZTb3NWalQ1VENUZXRkM1llZlMwVVZRMlVzbUtpWnlrb1J2OHVDVGY0IiwieSI6IktSUEFGdWFKdk9rU054THJnbVh5cEVBbVVkSjg1NTNHQzRZWjF0U09HVDAifSwiY3“

   Der IDP der gematik antwortet der Software der Gesundheitseinrichtung mit HTTP 302 FOUND und Location (Hinweis: Automatische Redirects müssen dazu abgeschaltet sein):

   https://<IRD-URL>/auth/realms/IRD/broker/gematik-idp/endpoint/result?
   code=eyJhbGciOiJkaXIiLC....IwGT627euBCZiS95vxu1eA
   &state=88d24c3a-e4b6-4f90-a191-9be0a9bfab63__INA__c6jbgfo5jmA

   Diese URL wird am Ende um „&cardType=SMC-B“ ergänzt und per GET-Request abgesetzt:

   GET https://<IRD-URL>/auth/realms/IRD/broker/gematik-idp/endpoint/result?
   code=eyJhbGciOiJkaXIiLC....IwGT627euBCZiS95vxu1eA
   &state=88d24c3a-e4b6-4f90-a191-9be0a9bfab63__INA__c6jbgfo5jmA&cardType=SMC-B

   Dies resultiert in einem HTTP 200 als Antwort.

   Hinweis: Die gematik sendet bei einem Fehler im Body der Response ein JSON- Dokument mit Details zu dem Fehler mit zurück.

5. Nachdem in Schritt 4 der Antwort-Status 200 empfangen worden ist, sendet die Software der Gesundheitseinrichtung per GET-Methode folgenden Request an den IDP der Registerstelle:

GET https://<IRD-URL>/auth/realms/IRD/broker/gematik-idp/endpoint/status?
state=88d24c3a-e4b6-4f90-a191-9be0a9bfab63__WS_BI__c6jbgfo5jmA

Hierbei muss der Wert des „state“ Parameters dem in Schritt 2 gespeicherten Wert entsprechen.

Die Antwort ist ein HTTP 200 mit einer JSON-Datei als Content. Diese Datei enthält in dem Attribut nextStepUrl die URL für den nächsten Schritt.

```
{
  "currentStep":"RECEIVED_SMCB_DATA",
  "nextStepUrl":“https://<IRD-URL>/auth/realms/IRD/broker/gematik-idp/endpoint/nextStep?state=88d24c3a-e4b6-4f90-a191-9be0a9bfab63__WS_BI__c6jbgfo5jmA“
}
```

Mögliche Fehler:

- Es wird der Status 202 zurückgeliefert und keine nextStepUrl. Dies bedeutet, dass der interne Anmeldevorgang am IDP der Registerstelle noch nicht vollständig abgeschlossen ist. Der Request sollte nach einer kurzen Wartezeit wiederholt werden. Dieser Fehler kann auftreten, wenn am Ende von Schritt 4 nicht auf die Antwort mit Status 200 gewartet worden ist.
- Es wird ein anderer Status als 200 oder 202 zurückgeliefert: Dies bedeutet, dass ein unerwarteter Fehler bei der Anmeldung aufgetreten ist. Die Anmeldung sollte abgebrochen und eine neue Anmeldung gestartet werden.
- Es wird kein Code zurückgeliefert, jedoch eine URL welche „first-broker-login“ enthält: Es sollte das Helpdesk der Register- und Vertrauensstelle kontaktiert werden.

6. Die Software der Gesundheitseinrichtung sendet den in Schritt 5 ermittelten Request (nextStepUrl) an den IDP der Registerstelle:

GET https://<IRD-URL>/auth/realms/IRD/broker/gematik-idp/endpoint/nextStep?state=88d24c3a-e4b6-4f90-a191-9be0a9bfab63__WS_BI__c6jbgfo5jmA

Der IDP der Registerstelle antwortet der Software der Gesundheitseinrichtung mit HTTP 302 FOUND und Location (Hinweis: Automatische Redirects müssen dazu abgeschaltet sein):

https://<Webservice-URL>?
state=1266ac6e-4dea-4e0f-a3e1-c43360fb6ddb
&session_state=88d24c3a-e4b6-4f90-a191-9be0a9bfab63
&code=954ab355-ef72-4acb-8b1b-1311bfd65c8a.342b4b3b-321a-4177-8b48-bc5616c2c7d1.07a1561a-9e68-4f7b-ba5d-00710a0279d3

Dieser URL ist der authorization-code (Parameter „code“) zu entnehmen und für das Abholen des Tokens im folgenden Schritt zu verwenden.

7. Die Software der Gesundheitseinrichtung fordert bei dem IDP der Registerstelle ein Token an:

POST https://<IRD-URL>/auth/realms/IRD/protocol/openid-connect/token
HTML Form URL Encoded: application/x-www-form-urlencoded
Form item: "grant_type" = "authorization_code"
Form item: "code" = "954ab355-ef72-4acb-8b1b-1311bfd65c8a.342b4b3b-321a-4177-8b48-bc5616c2c7d1.07a1561a-9e68-4f7b-ba5d-00710a0279d3"
Form item: "code_verifier" = <erstellter Code Verifier>
Form item: "redirect_uri" = https://<Webservice-URL>
Form item: "client_session_state" = "jMVjtbZxoOJRnM_qMBf9da_KSB5aOqZUZsnEiv-L“
Form item: "client_session_host" = "desktop-p6jnpnm" (optional)
Form item: "client_id" = <Webservice-Client-ID>

Die Form-Items client_session_state und client_session_host sind dabei vom Primärsystem sinnvoll zu befüllen.
Die Antwort ist ein HTTP 200 mit einer JSON-Datei als Content. Diese Datei enthält das gewünschte Access-Token:

{"access_token":"eyJhbGciOiJSU.....32NuMv0lIJw",
"expires_in":300,
"refresh_expires_in":3600,
"refresh_token":"eyJhbGciOiJIUzI1N.....XWL1n912lEmib0s",
"token_type":"Bearer",
"id_token":"eyJhbGciOiJSUz.....Nzv2cOZCw36w",
"not-before-policy":1681825865,
"session_state":"16d90479-e0b5-4015-a8b1-abf98565ee29",
"scope":"openid ird"}