# Basismodul SGD, Version 1.16.0

## 1 Basismodul SGD, Version 1.16.0

## 2 Überblick

Dieses Modul stellt eine Implementierung der Basisfunktionen da, die in [gemSpec_SGD] beschrieben sind. Es werden die dem Protokoll zugrundeliegenden Funktionen bereit gestellt (sgd-protocol), ein Modul für Client-/Server-Interfaces für Apache CXF (sgd-cxf), eine Crypto-Library (sgd-crypto) und eine Server-Implementierung (sgd-server).

Spezifikationsbasis ist “Release 3.1.3 Online-Produktivbetrieb” mit [gemSpec_Krypt], Version 2.16.0.

## 3 Voraussetzungen

- Es wird Lombok verwendet. Um in einer IDE also mit dem Quelltext arbeiten zu können wird ein entsprechendes Plugin benötigt.
- Die Entwicklung wurde mit Java 11 durchgeführt. Abwärtskompatibilität wird nicht gewährleistet.
- Maven wird als Buildtool in Version 3.6.1 verwendet.

## 4 Aufbau des Moduls

Die Implementierung erfolgt in Java (Version 11). Buildsystem ist Apache Maven (Version 3.6.1).

- **contracts** enthält die XSD-Schemas der gematik. Dieses Packet ist NICHT Teil des sgd-Reactors, muss also manuell gebaut werden.
- **sgd** ist das maven reactor module, das die einzelnen Teilmodule in einem Build baut.
- **sgd-protocol** bildet die Funktionalitäten des SGD-Protokolls ab. Enthalten sind
  - Server- und Client-Funktionalitäten zum Ver- und Entschlüsseln der Nachrichten
  - Konkrete Schlüsselableitungsfunktionalitäten des SGD-HSM
  - Grundsätzliche Funktionen zum Fehlerhandling

Alle benötigten kryptografischen Funktionen werden über ein Interface `SgdCryptoUtilities` angesprochen. Die konkrete Implementierung des Interfaces ist damit unabhängig von der verwendeten Krypto-Bibliothek. Eine Beispielhafte Implementierung findet sich in diesem Modul unter `SgdTestCryptoUtilities` und unter sgd-crypto

- **sgd-crypto** ist eine beispielhafte Implementierung der für das SGD-Modul benötigten Krypto-Funktionalitäten. Es basiert auf BouncyCastle. Diese Implementierung ist eine Kopie von `SgdTestCryptoUtilities`.
- **sgd-cxf** enthält das für einen CXF-Server und/oder -Client notwendige Interface.
- **sgd-server** ist eine Beispielimplementierung eines Servers. Es wird keine Datenbank benötigt, die Konfiguration erfolgt über die application.yaml. Der Server kann beispielsweise in einer Entwicklungsumgebung verwendet werden.

## 5 Release Notes

**1.16.0**

- Verbesserung: Die beigelegten Zertifikate des Moduls sgd-server wurden ausgetauscht. Diese Zertifikate sind in der ECC-TSL des TU/RU Vertrauensraums enthalten. (Keine funktionale Änderung)

**1.15.0**

- Change: In GetAuthenticationToken sendet der Client nun auch den H-Wert in der Challenge selbst mit. Der Server kann diesen Wert auch validieren (Siehe Flag performServerSidedHCheck in SchluesselGenerierungsDienst)

**1.14.0**

- Bugfix: Der Namensraumbezeichner für das Element encryptedKeyContainer der inneren Verschlüsselungsschicht wurde korrigiert.

**1.13.0**

- Verbesserung: Die beigelegten Zertifikate des Moduls sgd-server wurden ausgetauscht. Die neu enthaltenen Zertifikate sind in der TSL des TU/RU Vertrauensraums enthalten. (Keine funktionale Änderung)

**1.12.0**

- Verbesserung: Die Klasse SgdVector wurde um verbesserte Methoden zum Abrufen der OwnerKvnr und der AuthorizedId erweitert.

**1.11.0**

- Bugfix: Die Erstellung der “A-Zeichenkette” aus A_18026 wurde fehlerhaft durchgeführt: Es werden nun die Hash-Werte der Public-Keys der konkateniert. Dies ist ein Breaking-Change!

**1.10.0**

- Verbesserung: Es kann nun das Server-Zertifikat gegen die TSL geprüft werden. Um die Prüfung umzusetzen muss die Methode `isSgdHsmCertificateInTsl` entsprechend erweitert werden.
- Erweiterung: Die Änderungen nach 3.1.3 sind eingearbeitet. Darunter fällt insbesondere die veränderte Fehlerbehandlung nach A_18988

**1.4.0**

- Verbesserung: Die Server-Interfaces wurden Aufgespalten für SGD-1 und -2
- Bugfix: Das benötigte Package de.gematik.ti:contracts liegt nun dem ZIP bei

**1.3.0**

- Bugfix: Der SGD-Server kontrolliert nun immer die übergebenen AuthenticationTokens sowie deren Reihenfolge
- Bugfix: Der SGD-Client übergibt nun die AuthenticationTokens in der korrekten Reihenfolge
- Bugfix: Fehler auf Applikations-Ebene werden nun mit HTTP 200 - OK übermittelt

**1.2.0**

- Die Konvertierung von ECC-Schlüsseln für die JSON-Schnittstelle erfolgt nun Spezifikationskonform. Eine Länge von 64 Zeichen ist nicht mehr garantiert!

**1.1.0**

- sgd-server wurde als neues Modul eingeführt
- Diverse kleinere Umstrukturierungen in den DTO-Klassen an den Schnittstellen
- Neues Modul sgd-cxf
- Fix an den Tests im sgd-protocol: Alle benötigten Dateien sind nun Teil des ZIPs
- Release 3.1.2 wird nun unterstüzt. Einzige relevante Änderung ist der ohnehin schon vorweggenommene Change AssociatedDataType base64->string

**1.0.0**

- Initialer Release