

Design und Implementierung eines Localhost Signaturgateways

David Derler¹, Christof Rath¹, Moritz Horsch², Tobias Wich³

david.derler@iaik.tugraz.at, christof.rath@iaik.tugraz.at, horsch@cdc.informatik.tu-darmstadt.de,
tobias.wich@ecsec.de



Überblick

- Motivation
- Unser Beitrag
- FutureID Client
 - Add-on Framework
- eSign Plugin
 - Architektur
 - Server-/Clientseitige Signaturerstellung
 - Signaturvalidierung
- Zusammenfassung

Motivation

- Digitale Signatur
 - Schlüsselkomponente im elektronischen Geschäftsverkehr
- Vielzahl an Signaturkarten und Applikationen
- Hürden für Benutzerakzeptanz
 - Verschiedene Programmabläufe
 - Unterschiedliche Handhabung
- Steigerung von Vertrauen, Transparenz und Akzeptanz
 - ...durch flexible Open Source Signaturanwendung
 - Breite Unterstützung von
 - Signaturkarten/Signaturszenarien

Unser Beitrag

- FutureID Projekt
 - Privacy-Friendly Open Source Identity Management
 - FutureID Client¹
 - auf Basis des Open eCard² Framework
 - modulare Applikation
 - Vielzahl an Signaturkarten unterstützt (ISO 24727, bzw. eCard-API Framework)
- eSign Services
 - Add-on zum FutureID Client
 - Signaturerstellung über FutureID Client
 - Bliebige andere Wege
 - z.B.: serverseitige Signaturerstellung
 - Sign-/Verify-Anfragen via OASIS DSS

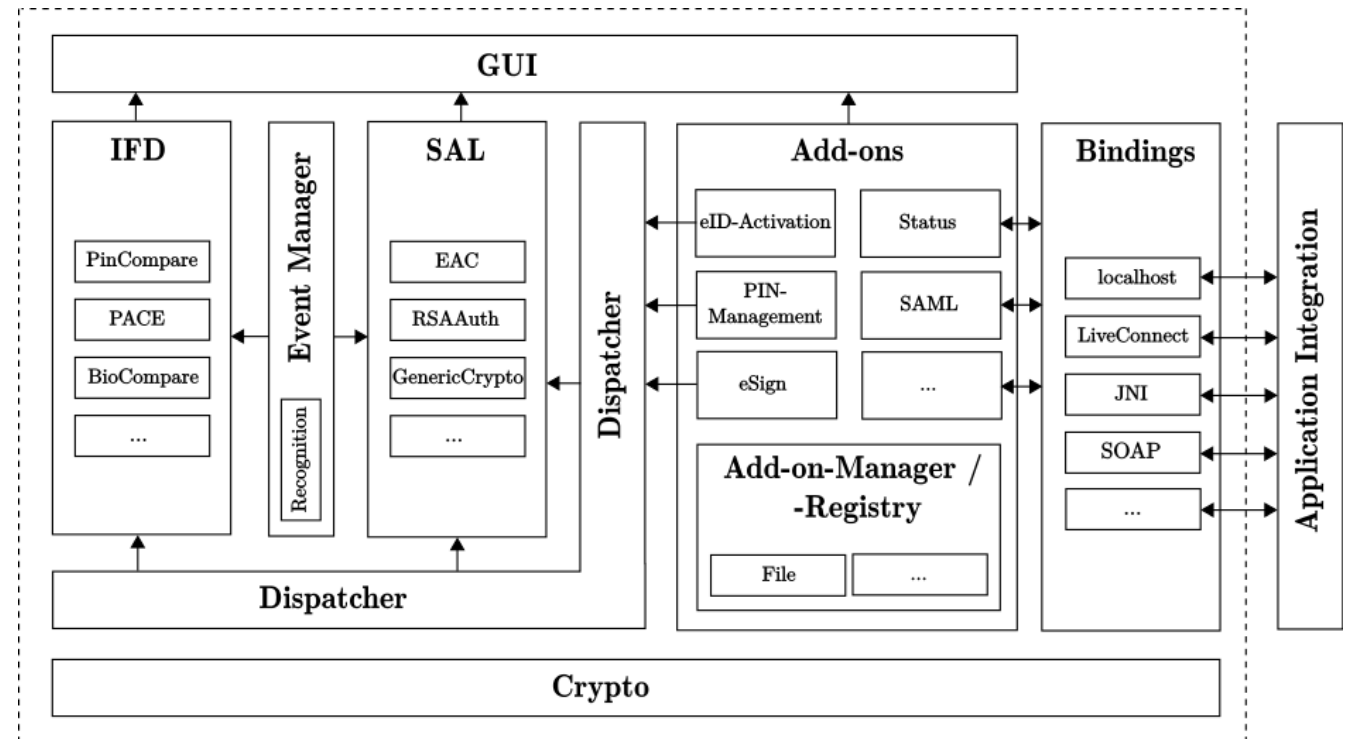


¹<http://www.futureid.eu>, ²<http://www.openecard.org>

FutureID Client

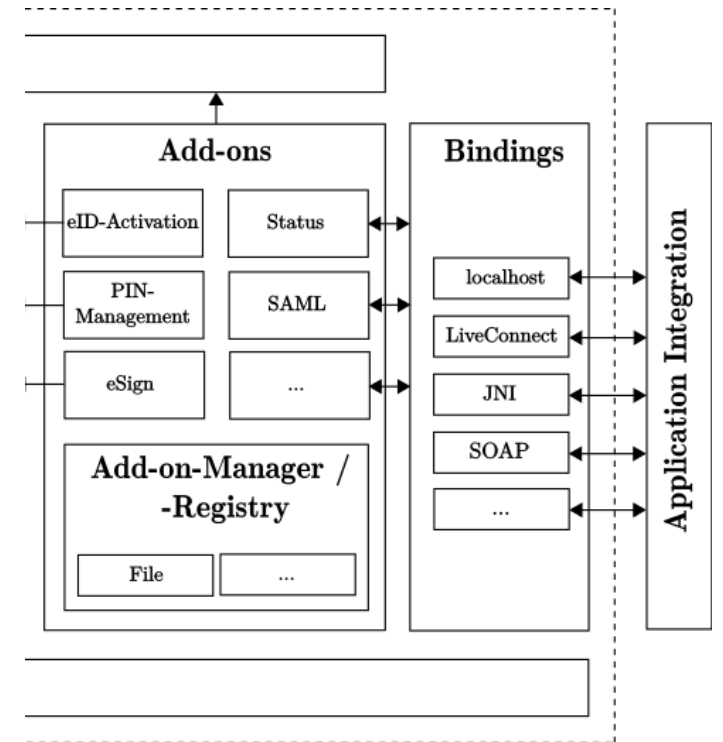
- Schnittstelle
 - Zw. Benutzer und Applikationen
 - Für Signaturerstellung bzw. Authentisierung

- IFD
- SAL
- Dispatcher
- Add-ons
- Bindings



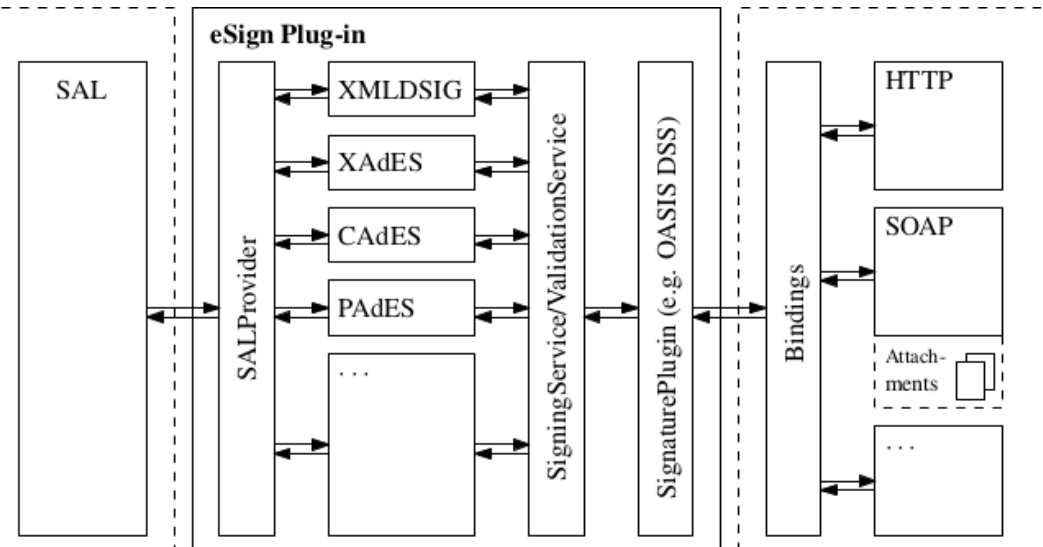
Add-on Framework

- Application Extensions
 - Explizit vom Benutzer ausgeführt
 - z.B.: PIN Management von Chipkarten
- Protokoll Plug-ins
 - Für IFD bzw. SAL (z.B.: Authentisierungsprotokolle)
- Application Plug-ins
 - z.B.: eSign Plug-in
- Plug-ins reagieren auf an sie adressierte Daten
- Kommunikation über Bindings
 - Extern: Transportprotokoll (z.B.: HTTP)
 - Intern: Abstraktes Nachrichtenformat
 - Beliebige Transportprotokolle



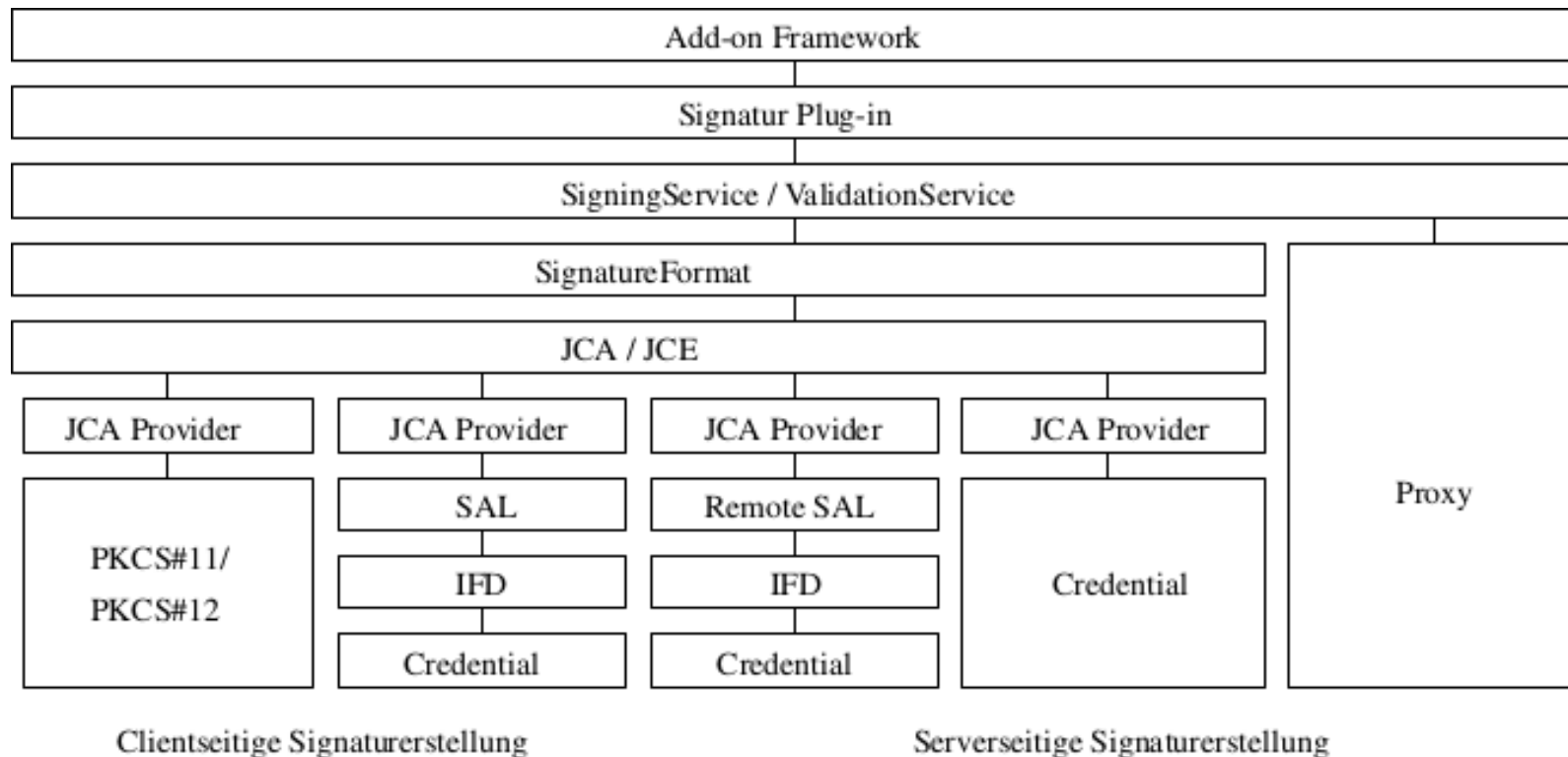
eSign Plug-in im Kontext des FutureID Client

- Konzept eines Signaturgateway
 - OASIS DSS Anfragen
- Kommunikation über Bindings
- Abstraktion der Signaturerstellung
 - Signing-/ValidationService
 - SALProvider



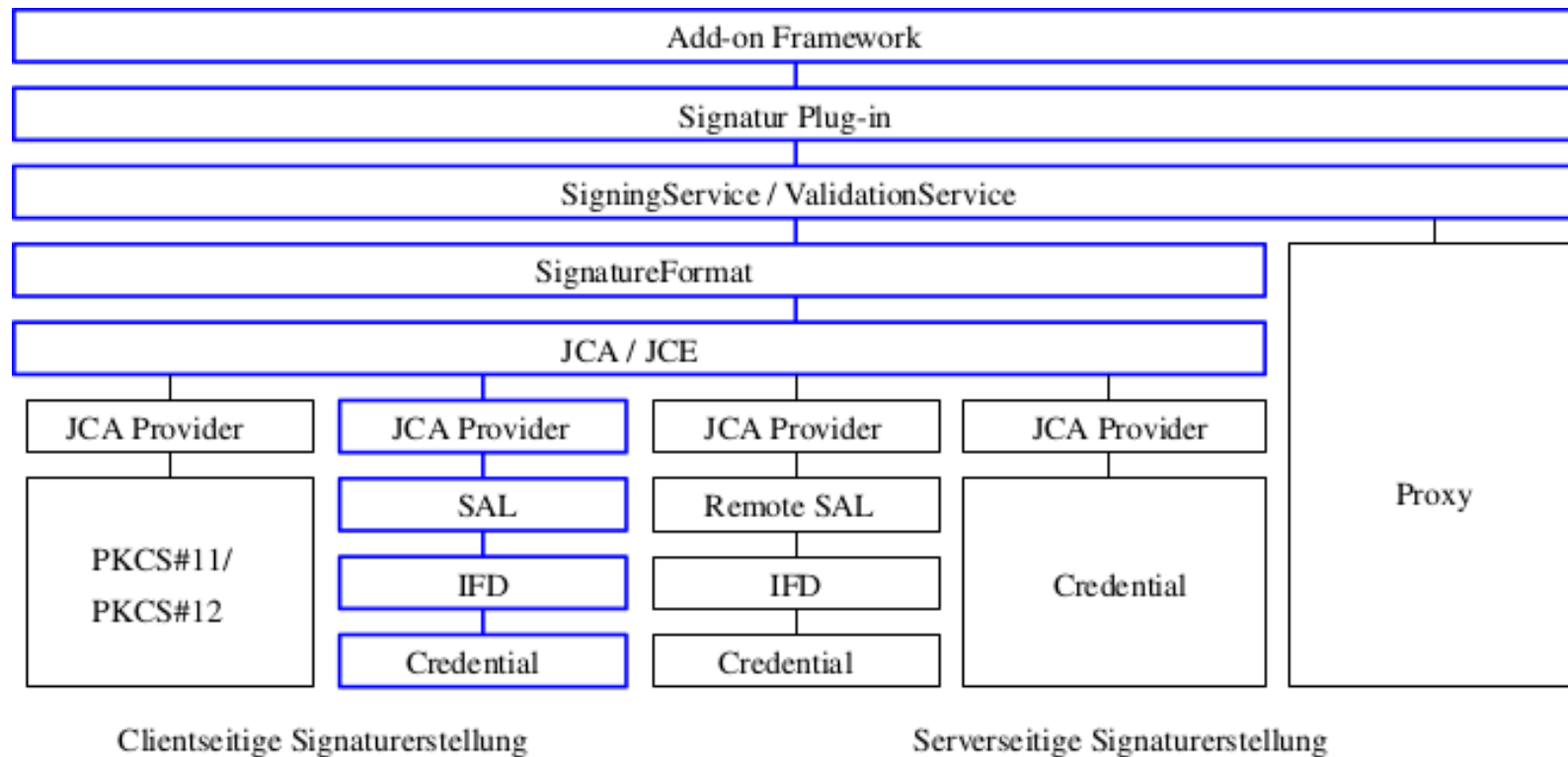
eSign Plug-in Architektur

- Vielseitige Möglichkeiten zur Signaturerstellung



eSign Plug-in Architektur

- Clientseitige Signaturerstellung (SAL Provider)



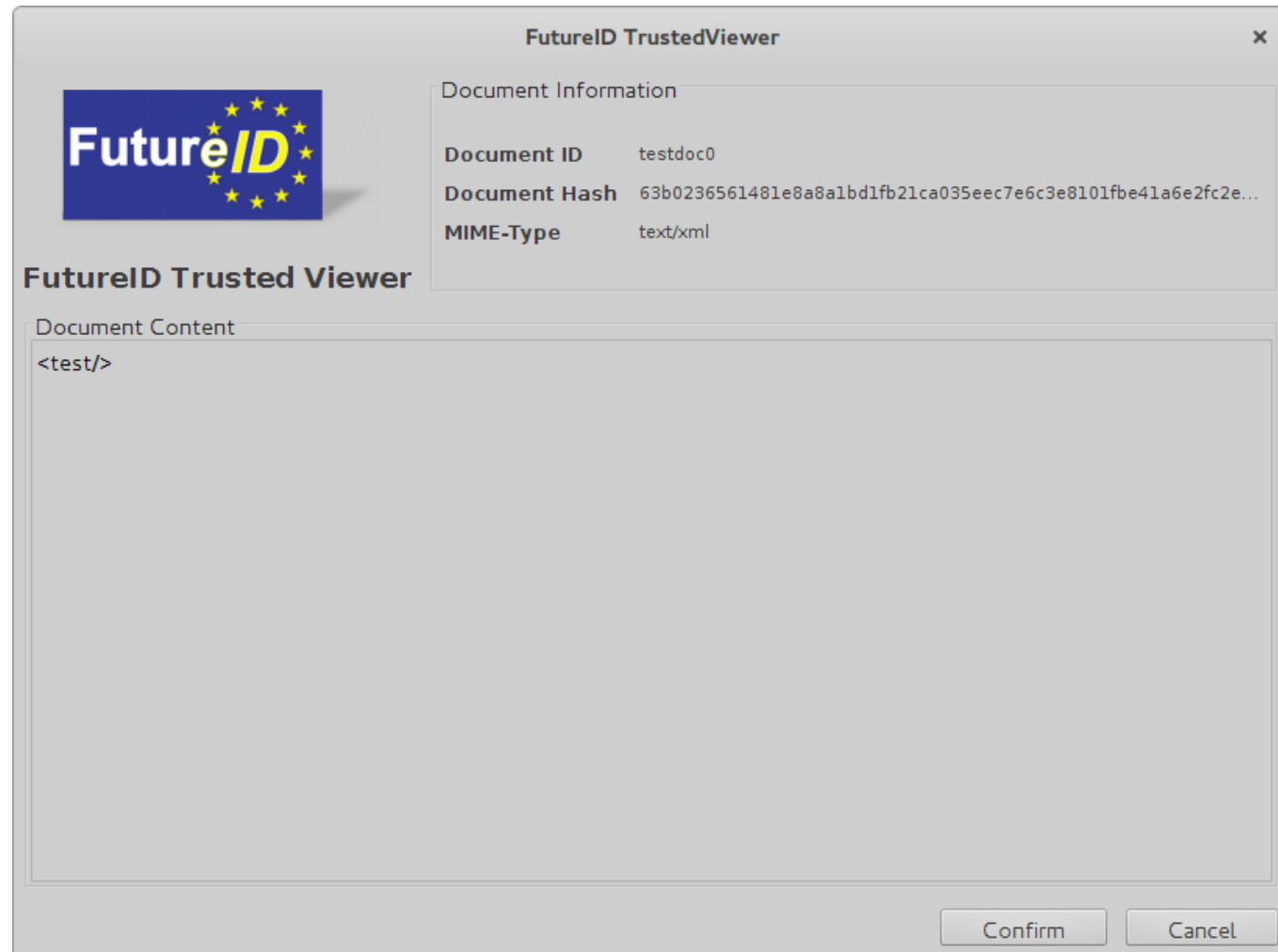
Clientseitige Signaturerstellung

- Zwei Schritte
 - Schlüsselmaterial abrufen
 - Public Key, (Referenz auf) Private Key
 - Abstrahiert durch `ProviderInitializationProxy`
 - Auch on-demand Signaturen möglich
 - Signaturwert berechnen
 - Java Cryptography Architecture (JCA)
 - `Signature` Interface
- SAL Provider
 - Authentifizierung durchführen (z.B. PIN)
 - Signaturanfrage an SAL
 - Verwendet von `SignaturFormats`

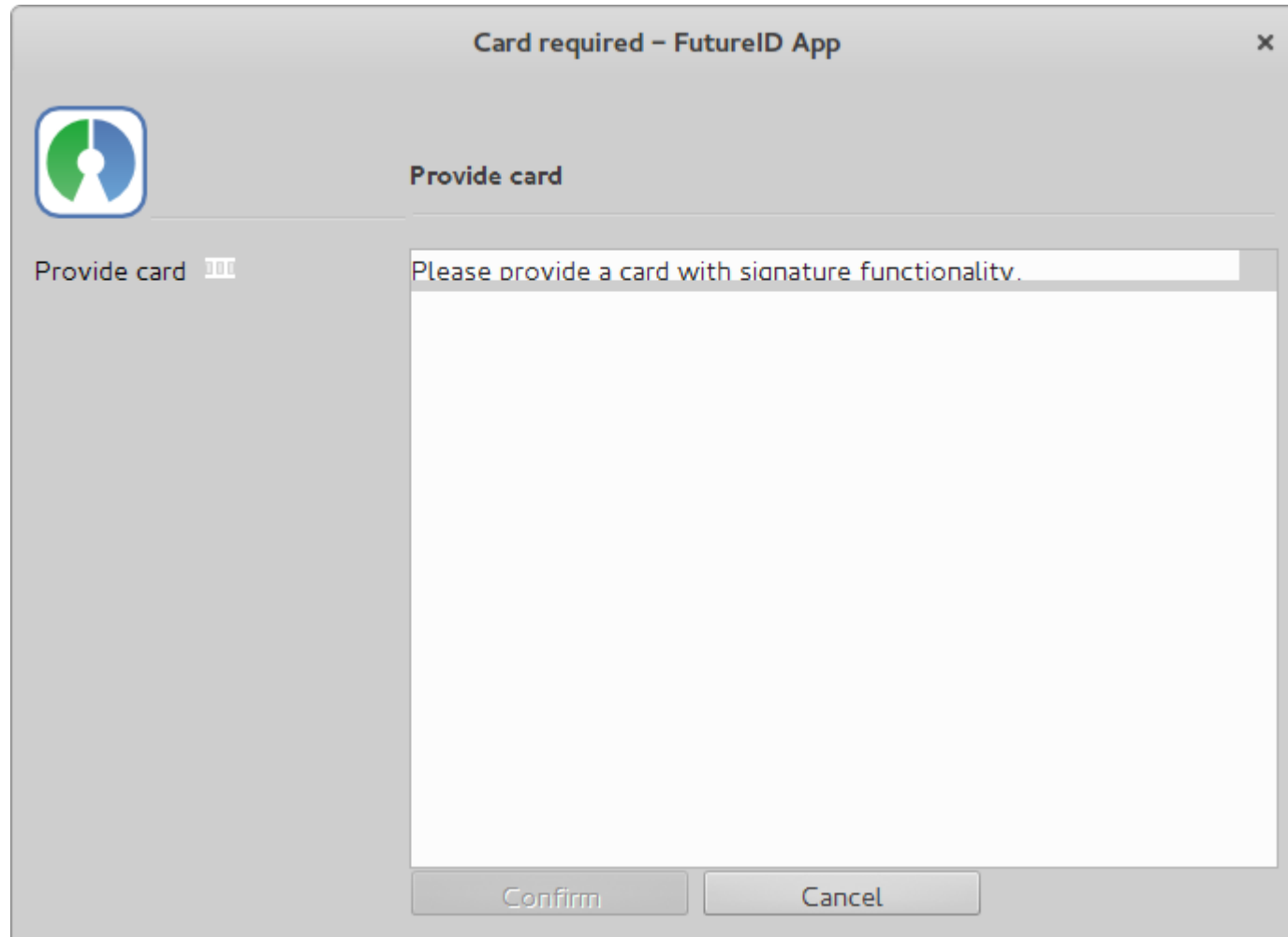
Beispiel

```
<dss:SignRequest RequestID="TestRequest"
  xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema"
  xmlns:futureid="urn:eu:futureid">
  <dss:OptionalInputs>
    <dss:SignatureType>urn:ietf:rfc:3275</dss:SignatureType>
    <dss:IncludeObject WhichDocument="testdoc0" ObjId="Document1"/>
  </dss:OptionalInputs>
  <dss:InputDocuments>
    <dss:Document ID="testdoc0" RefURI="#Document1" RefType="DSSReference">
      <dss:Base64XML>PHRlc3Q+PC90ZXN0Pg==</dss:Base64XML>
    </dss:Document>
  </dss:InputDocuments>
</dss:SignRequest>
```

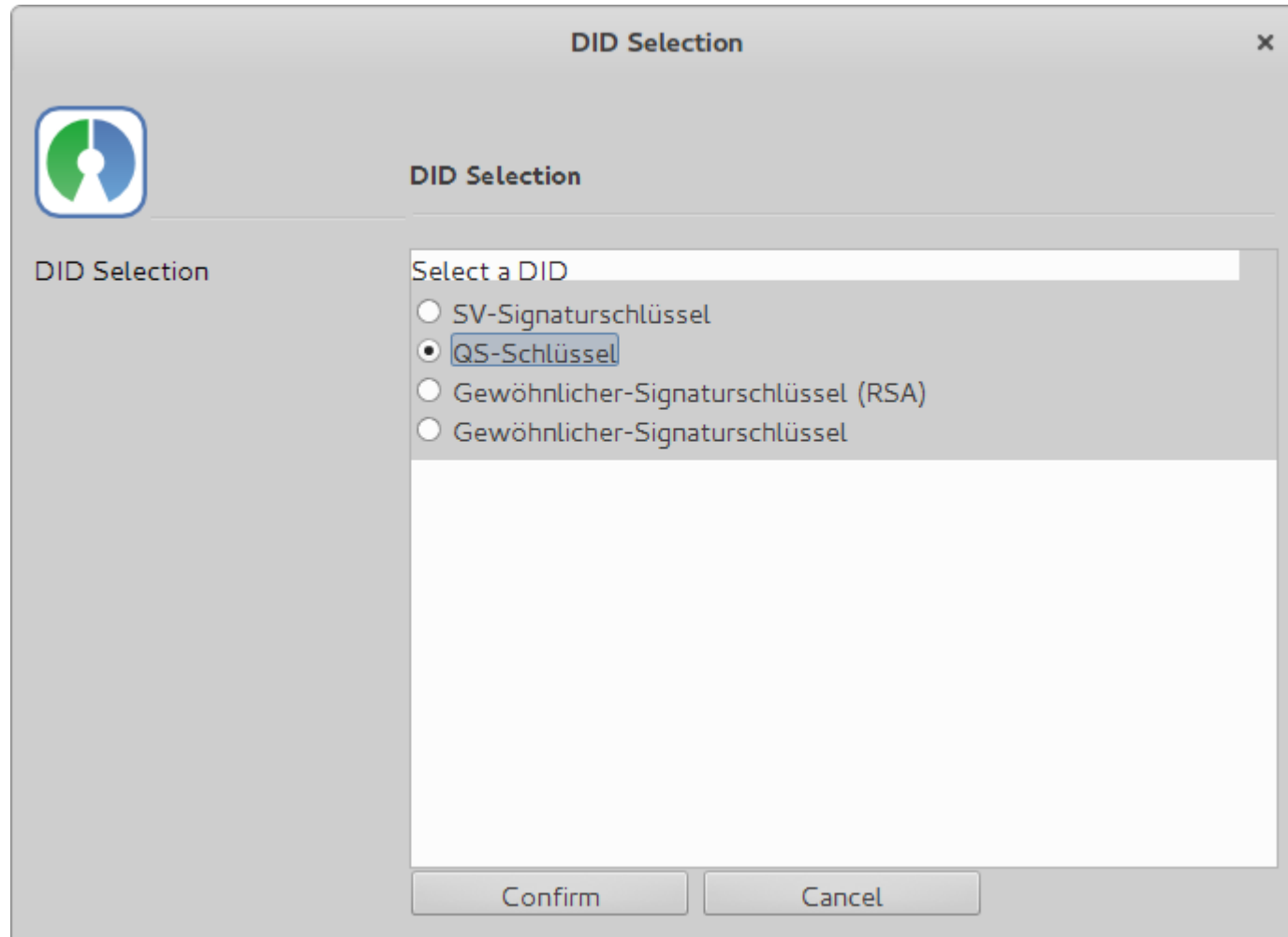
Beispiel



Beispiel

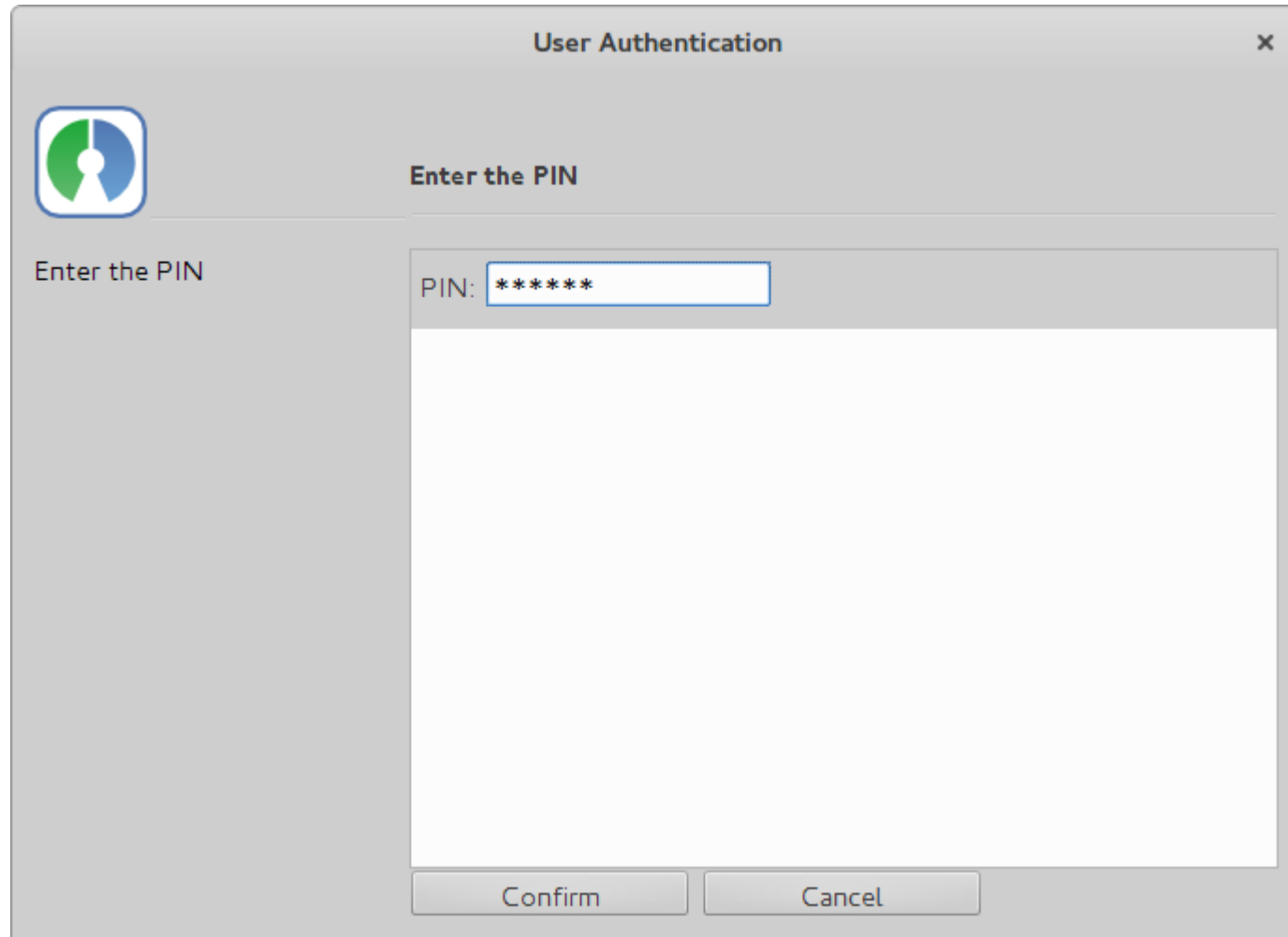


Beispiel



Beispiel

User Authentication ×



The image shows a standard Windows-style dialog box titled "User Authentication" with a close button (X) in the top right corner. On the left side, there is a circular icon with a green and blue design. Below the icon, the text "Enter the PIN" is displayed. The main area of the dialog is titled "Enter the PIN" and contains a text input field with the label "PIN:" and the text "*****" inside it. At the bottom of the dialog, there are two buttons: "Confirm" and "Cancel".

Enter the PIN

PIN: *****

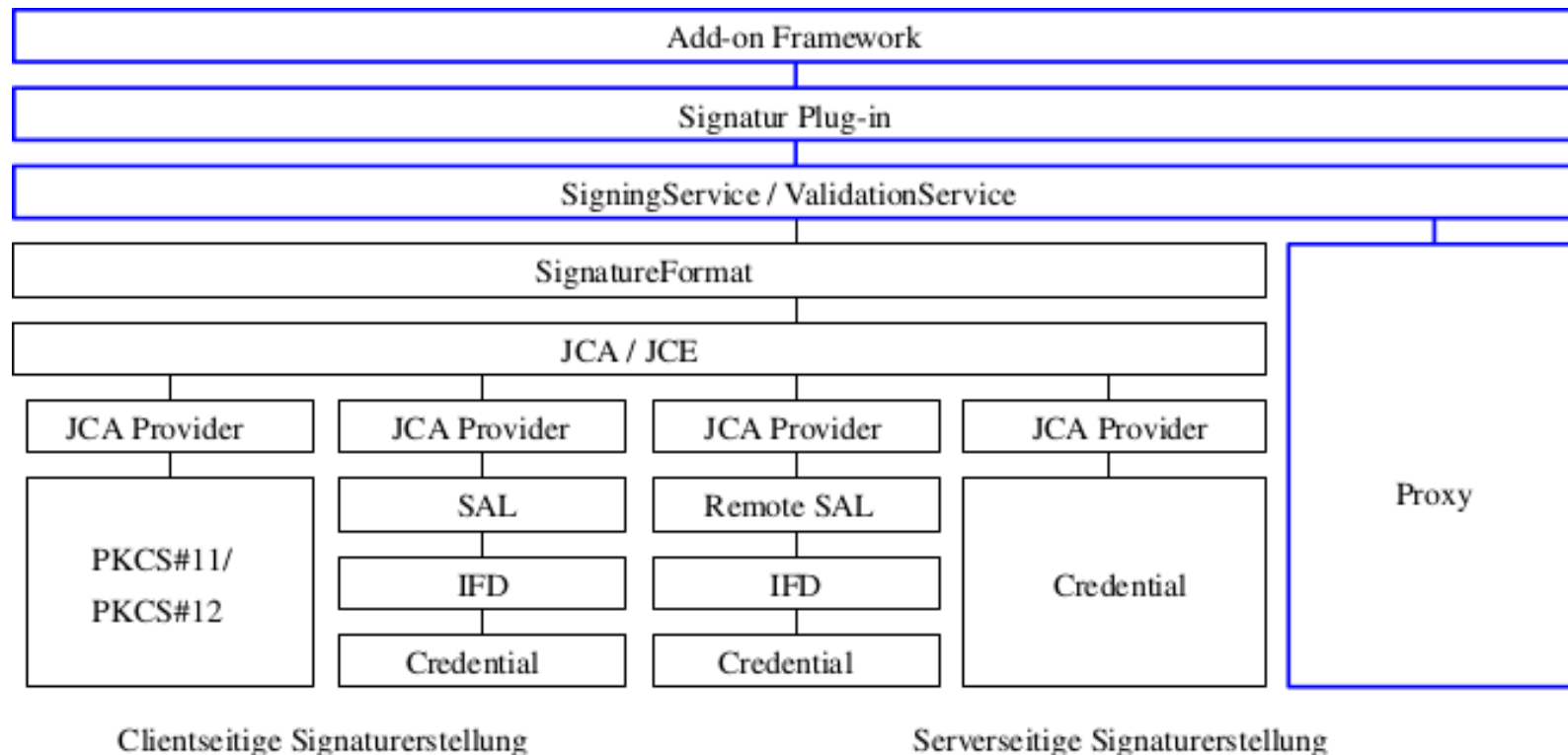
Confirm Cancel

Beispiel

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<dss:SignResponse xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema"
  Profile="urn:futureid:esign" RequestID="TestRequest">
  <dss:Result>
    <dss:ResultMajor>urn:oasis:names:tc:dss:1.0:resultmajor:Success</dss:ResultMajor>
  </dss:Result>
  <dss:SignatureObject>
    <dss:Base64Signature>PD94bWwgdmVyc2lv...</dss:Base64Signature>
  </dss:SignatureObject>
</dss:SignResponse>
```


eSign Plug-in Architektur

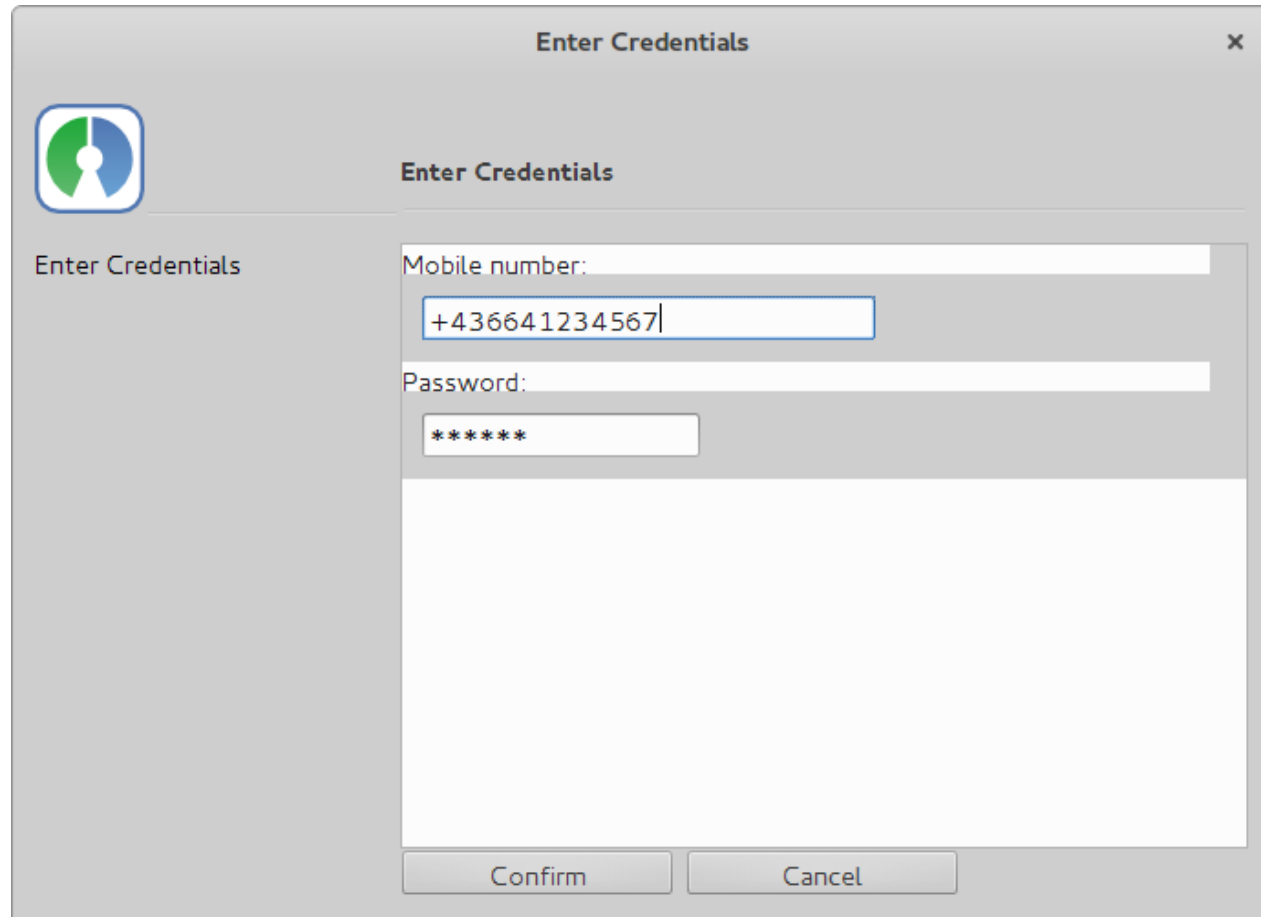
- Serverseitige Signaturerstellung (Österr. Handysignatur)



Serverseitige Signaturerstellung

- Österreichische Handysignatur
 - Dokument muss auf Server übertragen werden
 - Signatur wird dort erstellt
 - Security-Layer Anfragen
- Transformation
 - OASIS DSS – Security-Layer
 - CreateXMLSignatureRequest
 - CreateCMSSignatureRequest
- Benutzerinteraktion
 - Eingabe von Mobilnummer und Passwort
 - TAN Eingabe

Beispiel



Enter Credentials

Enter Credentials

Mobile number:

+436641234567

Password:

Confirm Cancel

Beispiel

Enter TAN

Enter TAN

Reference Value: Boh1mEaK4O

Enter TAN:

Confirm Cancel

Signaturvalidierung

- ValidationService delegiert Anfragen
 - an Webservice
- Vertrauensstatus von Zertifikaten
 - Entscheidungen werden zentral getroffen
 - Nutzer muss die Entscheidungen nicht treffen
 - Bessere Usability
 - Trustanchors per Policy überschreibbar

Beispiel

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<dss:SignResponse xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema"
  Profile="urn:futureid:esign" RequestID="TestRequest">
  <dss:Result>
    <dss:ResultMajor>urn:oasis:names:tc:dss:1.0:resultmajor:Success</dss:ResultMajor>
  </dss:Result>
  <dss:SignatureObject>
    <dss:Base64Signature>PD94bWwgdmVyc2lv...</dss:Base64Signature>
  </dss:SignatureObject>
</dss:SignResponse>
```

Beispiel

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<dss:VerifyResponse xmlns:dss="urn:oasis:names:tc:dss:1.0:core:schema" Profile="urn:futureid:esign" RequestID="TestRequest">
  <dss:Result>
    <dss:ResultMajor>urn:oasis:names:tc:dss:1.0:resultmajor:Success</dss:ResultMajor>
    <dss:ResultMessage xml:lang="en-US" xmlns:xml="http://www.w3.org/XML/1998/namespace">
      Basic Validation: SUCCESS/SUCCESS, msg:
        Identification of the signer's certificate (ISC): SUCCESS/SUCCESS, msg:
          Certificate digest: INVALID/FORMAT_FAILURE, msg: Signature does not contain certificate digest.
          Check IssuerSerial: SUCCESS/SUCCESS, msg: No IssuerSerial field present in the signature
          Identified signer certificate: SUCCESS/SUCCESS, msg: Subject DN: serialNumber=103033557143,givenName=David,SN=Derler,CN=David Derler,C=AT
        X509 Certificate validation wrapper for basic validation: SUCCESS/SUCCESS, msg:
          X509 Certificate validation (XCV): SUCCESS/SUCCESS, msg:
            Certificate 1: SUCCESS/SUCCESS, msg: Subject DN: CN=a-sign-premium-mobile-03,OU=a-sign-premium-mobile-03,O=A-Trust Ges. f. Sicherheitssysteme im
              elektr. Datenverkehr GmbH,C=AT
            Certificate 0: SUCCESS/SUCCESS, msg: Subject DN: serialNumber=103033557143,givenName=David,SN=Derler,CN=David Derler,C=AT
          Cryptographic verification (CV): SUCCESS/SUCCESS, msg:
            Integrity check of signed dataobjects: SUCCESS/SUCCESS, msg:
            Cryptographic verification of the signature: SUCCESS/SUCCESS, msg:
          Signature Acceptance Validation (SAV): SUCCESS/SUCCESS, msg:
            Signature constraints: SUCCESS/SUCCESS, msg:
            Cryptographic constraints: SUCCESS/SUCCESS, msg:
              Hashing algorithm constraint: SUCCESS/SUCCESS, msg:
              Key length constraint: SUCCESS/SUCCESS, msg:
          </dss:ResultMessage>
        </Result>
      <OptionalOutputs xmlns="urn:oasis:names:tc:dss:1.0:core:schema"/>
    </VerifyResponse>
```

Zusammenfassung

- Flexibles Framework
 - Modularer Aufbau
 - Standardisiertes Anfrage-/Antwortformat
- Einfache Integration in bestehende Applikation
- Signaturerstellung/-prüfung abstrahiert
 - Signaturwertberechnung über den SAL des FutureID Client
 - Beliebige andere Wege
- Einheitliches Look and Feel
 - Unabhängig von der Art der Signaturerstellung