

Anbindung externer Webanwendung an PDF- AS-WEB 4.0

Anbindung einer externen Webanwendung an PDF-AS-WEB 4.0

Version 0.4, 06.06.2014

Andreas Fitzek – andreas.fitzek@egiz.gv.at
Christian Maierhofer – christian.maierhofer@egiz.gv.at

Zusammenfassung: Dieses Dokument beschreibt die Anbindung externer Webanwendung an PDF-AS-WEB ab der Version 4.0. Es beschreibt die verfügbaren Parameter der PDF-AS-WEB Komponenten, sowie die Parameter die von PDF-AS-WEB an die externe Webanwendung übergeben werden.

Inhaltsverzeichnis

| | |
|--|---|
| 1 Beschreibung der Schnittstelle | 3 |
| 1.1 Übergabe per User Agent..... | 3 |
| 1.2 Upload per SOAP Schnittstelle..... | 5 |
| 1.3 Serversignatur per SOAP Schnittstelle | 6 |
| 2 Beschreibung der Implementation..... | 7 |
| 2.1 PDF-AS-WEB Servlets | 7 |
| 2.1.1 Sign Servlet | 7 |
| 2.1.2 ProvidePDF Servlet | 8 |
| 2.1.3 PDFData Servlet..... | 8 |
| 2.2 Parameteruebergabe an die externe Webanwendung | 9 |
| 2.2.1 Abrufen des originalen PDF Dokuments | 9 |
| 2.2.2 Fehlerseite der externen Webanwendung | 9 |
| 2.2.3 Benachrichtigung nach erfolgreicher Signatur..... | 9 |

Abbildungsverzeichnis

| | |
|--|---|
| Abbildung 1 - Prozessablauf zur Anbindung einer externen Webanwendung und PDF-AS-WEB via User Agent | 4 |
| Abbildung 2 - Prozessablauf zur Anbindung einer externen Webanwendung und PDF-AS-WEB via SOAP Schnittstelle..... | 5 |

1 Beschreibung der Schnittstelle

PDF-AS-WEB bietet unterschiedliche Möglichkeiten zur Anbindung einer externen Webanwendung:

- **Übergabe per User Agent:** In dieser Möglichkeit wird der Benutzer von der externen Webanwendung an PDF-AS-WEB weitergeleitet. PDF-AS-WEB bezieht in diesem Fall das zu signierende PDF von der externen Webanwendung.
- **Upload per SOAP Schnittstelle:** In dieser Möglichkeit wird das zu signierende PDF per SOAP Service von der externen Webanwendung direkt an PDF-AS-WEB hochgeladen. Als Antwort erhält die externe Webanwendung eine URL an die sie den Benutzer weiterleiten muss.
- **Serversignatur per SOAP Schnittstelle:** In dieser Möglichkeit wird das zu signierende PDF per SOAP Service von der externen Webanwendung direkt an PDF-AS-WEB hochgeladen. Als Antwort erhält die externe Webanwendung das signierte Dokument. Diese Möglichkeit funktioniert nur wenn das PDF Dokument direkt vom Server signiert werden soll.

1.1 Übergabe per User Agent

In Abbildung 1 - Prozessablauf zur Anbindung einer externen Webanwendung und PDF-AS-WEB wird der Ablauf der Signatur eines PDF Dokumentes einer externen Webanwendung durch PDF-AS-WEB dargestellt.

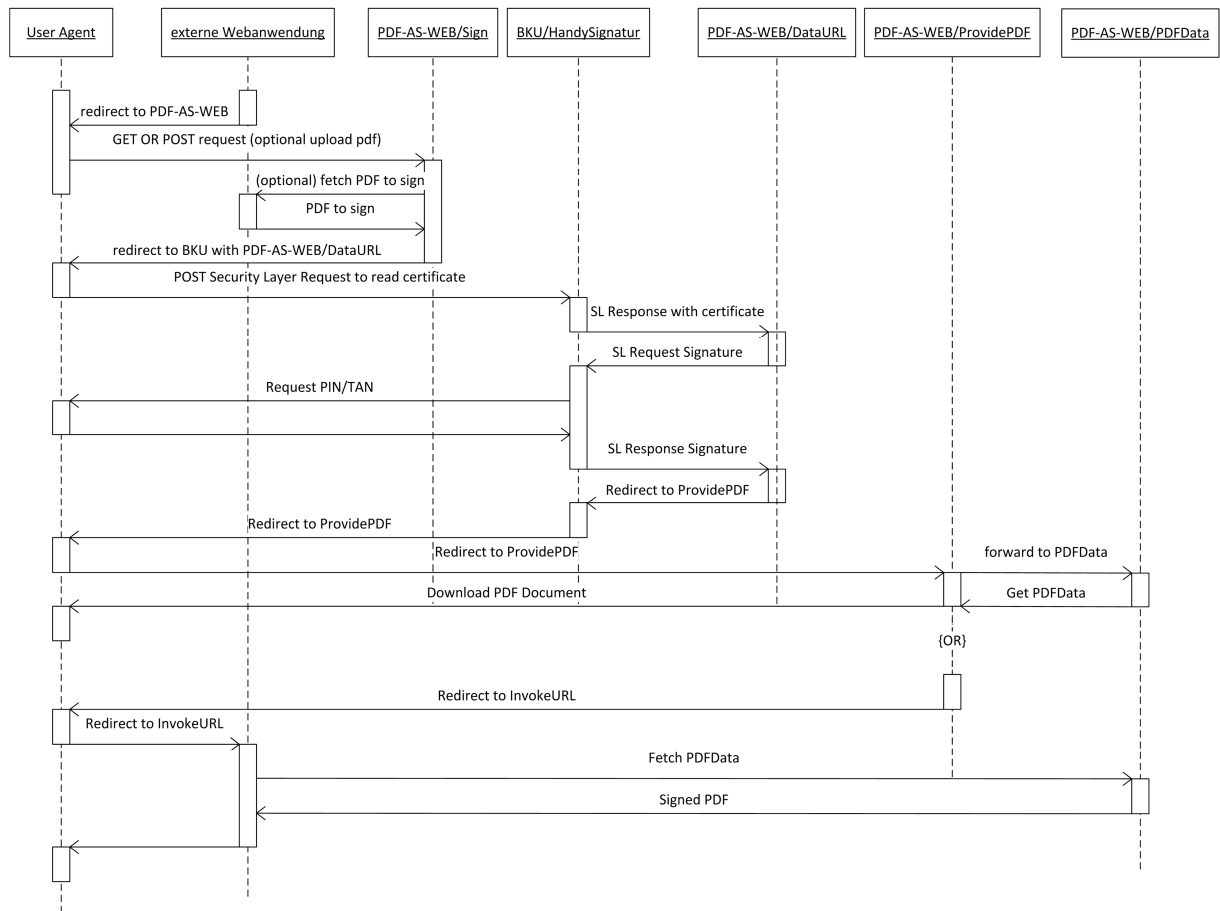


Abbildung 1 - Prozessablauf zur Anbindung einer externen Webanwendung und PDF-AS-WEB via User Agent

Im ersten Schritt leitet die externe Webanwendung den Benutzer zu PDF-AS-WEB um. Im Zuge dieser Weiterleitung werden PDF-AS-WEB diverse Parameter übergeben. Eine detaillierte Beschreibung der Parameter folgt später. Im ersten Request des Benutzers an PDF-AS-WEB kann entweder das zu signierende PDF Dokument per POST Request hochgeladen werden oder eine URL angegeben werden von der PDF-AS-WEB das PDF Dokument herunterladen soll. Sobald PDF-AS-WEB im Besitz des zu signierenden PDF Dokumentes ist wird die Signaturerstellung vorbereitet. PDF-AS-WEB bereitet einen Security Layer Request vor und leitet den Benutzer zur der ausgewählten Bürgerkartenumgebung um. Die Bürgerkartenumgebung bearbeitet die Security Layer Requests und sendet das Ergebnis per Data URL an PDF-AS-WEB. Wenn die Signatur erzeugt wurde antwortet PDF-AS-WEB mit einer Weiterleitung an das ProvidePDF Servlet mit entsprechender Data URL. Somit wird der Benutzer von der Bürgerkartenumgebung auf das ProvidePDF Servlet von PDF-AS-WEB weitergeleitet. Je nachdem ob die externe Webanwendung eine InvokeURL angegeben hat, wird der Benutzer auf diese URL weitergeleitet, oder es wird dem Benutzer angeboten das PDF Dokument direkt von PDF-AS herunter zu laden. Ein signiertes PDF Dokument kann nur ein einziges Mal von

PDF-AS-WEB heruntergeladen werden. Wird der Benutzer wieder an die externe Webanwendung weitergeleitet, so bekommt diese einen Link um das signierte PDF von PDF-AS-WEB herunterzuladen. In diesem letzten Schritt hat die Webanwendung auch die Möglichkeit den SHA-256 Hash der originalen Signatur Daten an PDF-AS-WEB zu schicken. Nur wenn die Originaldaten des signierten PDF Dokumentes zu diesem Hash passen, wird das PDF-Dokument von PDF-AS geliefert. Diese Überprüfung ist allerdings optional.

1.2 Upload per SOAP Schnittstelle

Die Endpunkt für das SOAP Webservice in PDF-AS-WEB befindet sich unter „/wssign“. Unter der URL „/wssign?wsdl“ kann die Web Service Description abgefragt werden. Dieses WSDL dient als Dokumentation für das SOAP Service.

In Abbildung 2 - Prozessablauf zur Anbindung einer externen Webanwendung und PDF-AS-WEB via SOAP Schnittstelle ist der Ablauf einer Signatur per SOAP Schnittstelle beschrieben.

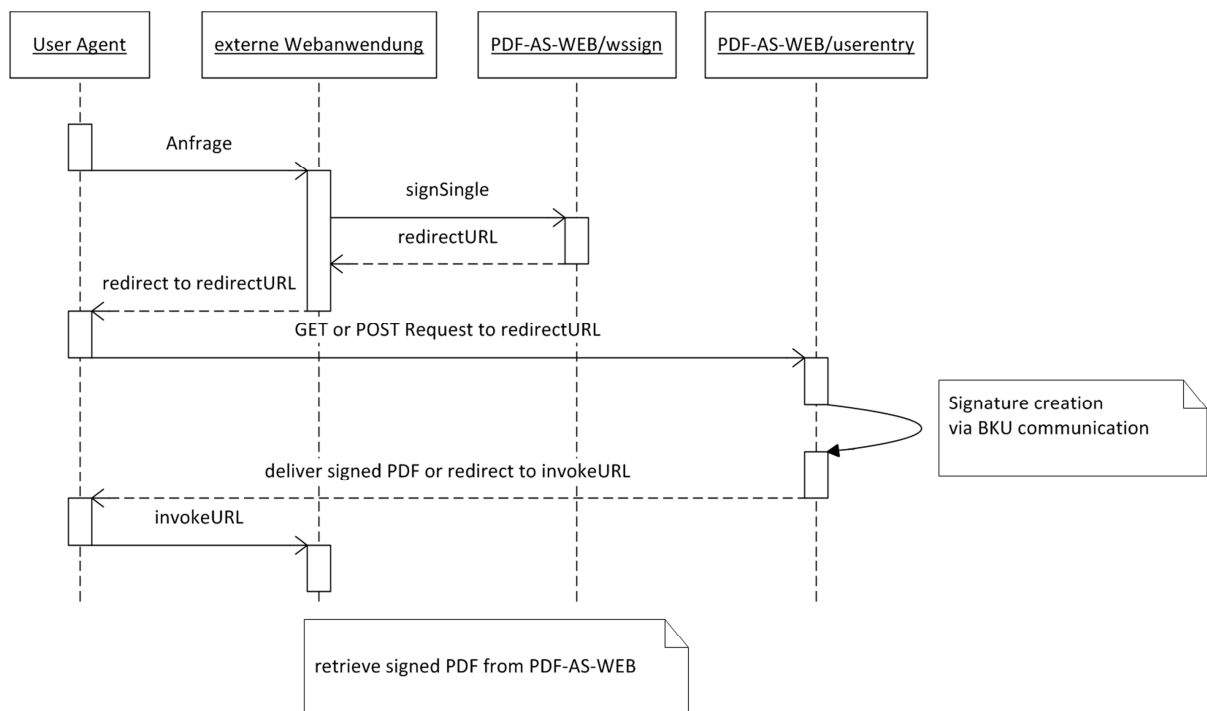


Abbildung 2 - Prozessablauf zur Anbindung einer externen Webanwendung und PDF-AS-WEB via SOAP Schnittstelle

Die externe Webanwendung startet den Signaturvorgang in dem ein SOAP Request an PDF-AS-WEB geschickt wird. Dieser Request enthält das zu signierende PDF Dokument, sowie den zu verwendenden connector, die invoke-url, die invoke-error-url und die

Signaturposition. Also Antwort empfängt die Webanwendung eine URL. Der Benutzer muss an diese URL weitergeleitet werden damit PDF-AS-WEB eine Benutzerschnittstelle für die Bürgerkartenumgebung zur Verfügung hat. Nachdem das Dokument durch den Benutzer signiert wurde, kann dieser das PDF Dokument von PDF-AS-WEB herunterladen, bzw. falls eine *invoke-url* definiert wurde, wird der Benutzer an diese URL weitergeleitet. Somit hat die Webanwendung die Möglichkeit das signierte Dokument von PDF-AS-WEB zu beziehen.

1.3 Serversignatur per SOAP Schnittstelle

Es wird dasselbe Web Service wie unter 1.2 verwendet um Serversignaturen zu erstellen. Der einzige Unterschied ist in diesem Fall, dass als Connector Parameter im SOAP Request „jks“ oder „moa“ angegeben werden müssen. PDF-AS-WEB verwendet dann einen Software Keystore, bzw. eine MOA-SS Instanz im Hintergrund zur Erzeugung der Signatur.

2 Beschreibung der Implementation

2.1 PDF-AS-WEB Servlets

Hier werden die Parameter beschrieben, die PDF-AS-WEB externen Webanwendungen zur Steuerung anbietet.

2.1.1 Sign Servlet

Das Sign Servlet ist der Haupteinstiegspunkt in PDF-AS-WEB. Dieses Servlet kann entweder mittels GET oder POST Methode aufgerufen werden. Es muss entweder der **pdf-file** Parameter oder der **pdf-url** Parameter angegeben werden.

- **„connector“**: Dieser Parameter gibt an welches Signaturgerät verwendet werden soll. (**Pflichtparameter**) Mögliche Werte sind:
 - **„bku“**: Es soll die lokale Bürgerkartenumgebung verwendet werden.
 - **„onlinebku“**: Es die MOCCA online als Bürgerkartenumgebung verwendet werden.
 - **„mobilebku“**: Es soll die Handy Signatur verwendet werden.
 - **„moa“**: Es soll eine MOA-SS Instanz verwendet werden.
- **„invoke-app-url“**: Dieser Parameter gibt die Fertigstellungsurl der externen Webanwendung an. Der Benutzer wird nach Abschluss des Signaturvorgangs wieder auf diese URL umgeleitet. (**optionaler Parameter**)
- **„invoke-app-error-url“**: Dieser Parameter gibt die Fehlerurl der externen Webanwendung an. Im Fehlerfall wird der Benutzer auf diese URL umgeleitet. (**optionaler Parameter**)
- **„locale“**: Die Locale die PDF-AS-WEB verwenden soll. Wird kein Parameter angegeben werden die Locale vom Benutzerbrowser übernommen. (**optionaler Parameter**)Mögliche Werte sind:
 - „DE“
 - „EN“
- **„num-bytes“**: Dieser Parameter gibt die Anzahl der Bytes des zu signierenden PDF Dokuments an. Wenn dieser Parameter angegeben wird, aber die Signaturdaten nicht entsprechen wird ein Fehler generiert. (**optionaler Parameter**)
- **„sig-type“**: Dieser Parameter gibt das zu verwendete Signaturprofil von PDF-AS an. Wird dieser Parameter nicht angegeben, wird das Standardsignaturprofil verwendet. (**optionaler Parameter**) Mögliche Standardwerte sind:
 - SIGNATURBLOCK_DE
 - SIGNATURBLOCK_EN
 - SIGNATURBLOCK_SMALL_DE
 - SIGNATURBLOCK_SMALL_EN

- AMTSSIGNATURBLOCK_DE
- AMTSSIGNATURBLOCK_EN
- AMTSSIGNATURBLOCK_DE_SMALL
- AMTSSIGNATURBLOCK_EN_SMALL
- „**pdf-file**“: Dieser Parameter gibt den File Parameter an beim upload des PDF Dokuments mittels POST Methode. (**optional Parameter**)
- „**pdf-url**“: Dieser Parameter gibt die URL an, von der das zu signierende PDF Dokument heruntergeladen werden soll. (**optional Parameter**)
- „**sig-pos-p**“: Mit diesem Parameter kann die Seite auf der die Signatur positioniert werden soll festgelegt werden. (**optional Parameter**)
- „**sig-pos-w**“: Mit diesem Parameter kann die Breite des Signaturblocks festgelegt werden. (**optional Parameter**)
- „**sig-pos-x**“: Mit diesem Parameter kann die X Koordinate der Position des Signaturblocks festgelegt werden. (**optional Parameter**)
- „**sig-pos-y**“: Mit diesem Parameter kann die Y Koordinate der Position des Signaturblocks festgelegt werden. (**optional Parameter**)
- „**sig-pos-r**“: Mit diesem Parameter kann die Rotation (in Grad) des Signaturblocks festgelegt werden. (**optional Parameter**)
- „**sig-pos-f**“: Mit diesem Parameter kann die Footerhoehe bei automatischer Platzierung des Signaturblocks vorgegeben werden. (**optional Parameter**)

2.1.2 ProvidePDF Servlet

Das Provide PDF Servlet laesst keine Parameter zu. Das Verhalten wird durch die Parameter gesteuert, die beim Aufruf des SignServlet uebergeben wurden.

2.1.3 PDFData Servlet

Das PDFData Servlet returniert, falls vorhanden, das signierte PDF Dokument. Vorsicht: Dieses Servlet kann nur ein einziges Mal pro Signaturvorgang aufgerufen werden. Nachdem das signierte Dokument abgeholt wurde, werden die Daten des signierten PDF Dokumentes geloescht.

- „**origdigest**“: Dieser Parameter kann den Hex kodierten SHA256 Wert des Originaldokuments enthalten. Wenn dieser Parameter angegeben wird, prueft PDF-AS-WEB, ob der Wert mit dem Wert der originalen Daten uebereinstimmt. Nur wenn dieser uebereinstimmt, werden die Signaturdaten returniert. Wenn nicht, wird ein Fehler generiert. (**optional Parameter**)

2.2 Parameteruebergabe an die externe Webanwendung

Hier werden die Parameter beschrieben, die PDF-AS-WEB an die externe Webanwendung uebergibt.

2.2.1 Abrufen des originalen PDF Dokuments

Wenn PDF-AS-WEB das PDF Dokument von der externen Webanwendung herunterladen soll. Werden von PDF-AS keine Parameter uebergeben, sondern nur die URL die mit **pdf-url** an das Sign Servlet uebergeben wurde aufgerufen. PDF-AS-WEB bietet ein Whitelisting um pdf-urls zu filtern.

2.2.2 Fehlerseite der externen Webanwendung

Wenn in PDF-AS-WEB eine externen Fehlerurl mit dem Parameter **invoke-app-error-url** uebergeben wurde und in PDF-AS-WEB ein Fehler auftritt, wird der Benutzer auf diese URL weitergeleitet. Diese URL wird mittels GET Methode aufgerufen und 2 Parameter werden uebergeben:

- „**error**“: Dieser Parameter enthaelt einen Text mit einer einfachen Fehlermeldung
- „**cause**“: Dieser Parameter enthaelt einen Text mit einer detaillierteren Fehlerbeschreibung

PDF-AS-WEB bietet ein Whitelisting um Fehlerurls zu filtern.

2.2.3 Benachrichtigung nach erfolgreicher Signatur

Wenn in PDF-AS-WEB eine Fertigstellungsurl mit dem Parameter **invoke-app-url** uebergeben wurde, leitet PDF-AS-WEB nach einer erfolgreichen Signatur den Benutzer an diese URL weiter. Dieser URL wird mittels GET Methode aufgerufen und 2 Parameter werden uebergeben:

- „**pdfurl**“: Die URL unter der das signierte Dokument abgerufen werden kann.
- „**pdflength**“: Die Anzahl der Bytes die das signierte Dokument hat.

PDF-AS-WEB bietet ein Whitelisting um Fertigstellungsurl zu filtern.

Dokumentenhistorie

| Version | Datum | Autor(en) | Anmerkung |
|---------|------------|----------------------|----------------------------------|
| 0.1 | 21.01.2014 | Andreas Fitzek | Initiale Version |
| 0.2 | 19.02.2014 | Christian Maierhofer | Review |
| 0.3 | 05.06.2014 | Andreas Fitzek | Erweiterungen SOAP Schnittstelle |
| 0.4 | 06.06.2014 | Andreas Fitzek | sig-type Werte hinzugefügt |