

## **PDF Signaturen für eSchKG Beilagendokumente**

### Anforderungs-Spezifikation

2. Mai 2012 / --zw

## 1 Zweck

Dieses Dokument spezifiziert die Erweiterung des Message Handlers um eine automatische Signierfunktion für PDF-Dokumente im Rahmen des eSchKG Verbundes.

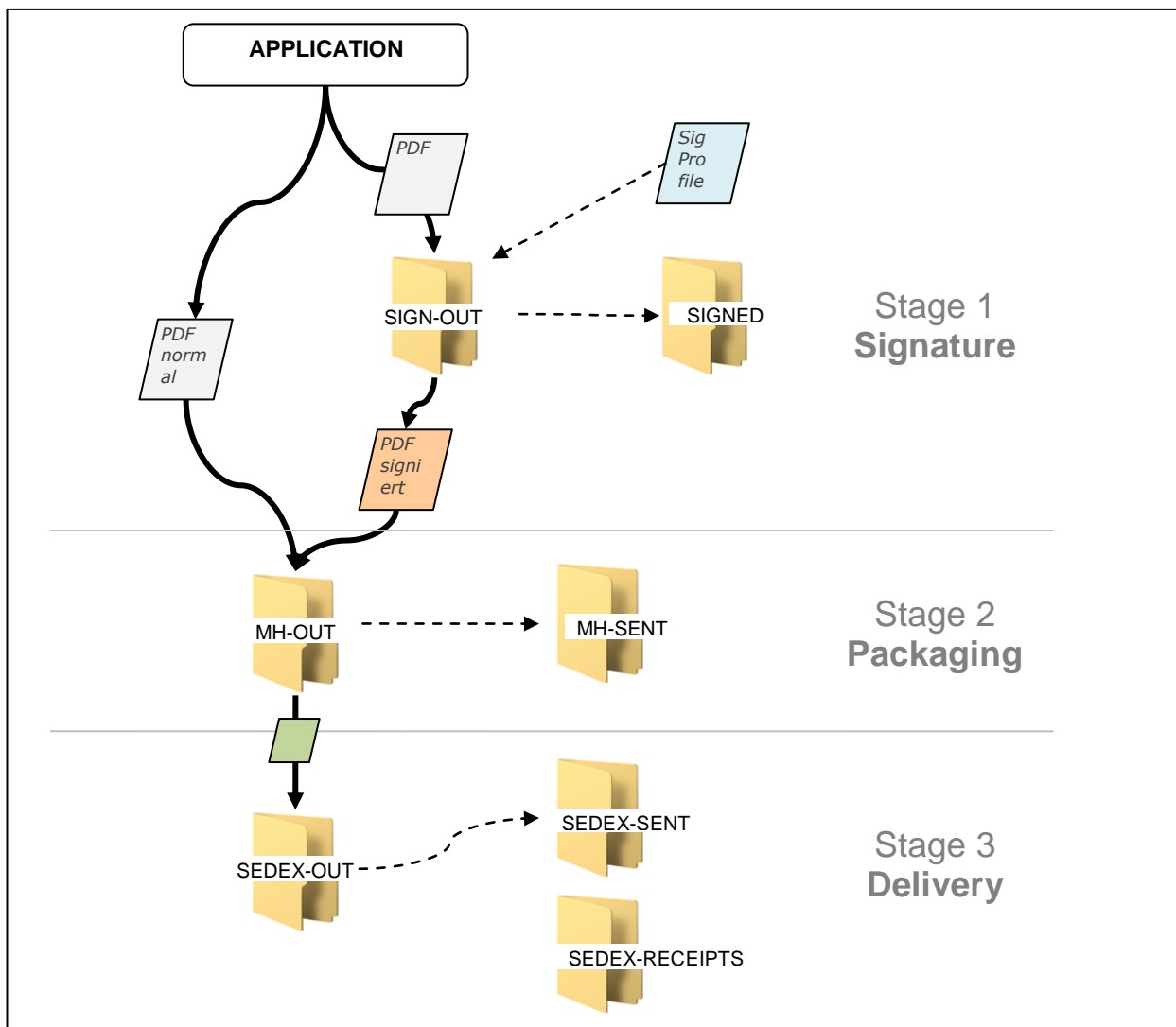
## 2 Ausgangslage

eSchKG 2.0 erlaubt die Übermittlung von PDF Beilagen, sowohl als Bestandteil einer Eingabe des Gläubigers als auch einer umfassenden Antwort des Betreibungsamtes. Während es dem Gläubiger freisteht, seine PDF Unterlagen zu signieren, ist die Situation beim Betreibungsamt anders. Bestimmte Unterlagen sind entscheidend für die weitere Verwendung durch den Gläubiger, z.B. ein elektronisches Gläubigerdoppel als Beilage zu einem Rechtsöffnungsbegehren an ein Gericht. Solche Unterlagen müssen daher vom Betreibungsamt signiert sein.

Die Dokumente sollen nicht von der Applikation signiert werden müssen. Vielmehr soll durch eine Erweiterung des MessageHandlers (MH) eine neue Funktionalität zum Signieren von Beilagen bereit gestellt werden.

## 3 Anforderungen

### 3.1 Funktionsprinzip



Der neue MH kombiniert die Funktionalitäten aus dem aktuellen MH und dem BatchSigner (BS). Die Hierarchie der Verarbeitung erfolgt gemäss den Stages:

1. **Signature.** Soll ein PDF Dokument nicht signiert werden, so wird es direkt in das Ausgangspostfach des MH geschrieben (MH-OUT). Falls es signiert werden soll, ist es in das dafür vorgesehene Verzeichnis zu schreiben (SIGN-OUT). Dessen Inhalt wird in regelmässigen Abständen an eine Signaturfunktion weitergegeben. Signierte Dokumente können (müssen aber nicht) in ein dafür konfiguriertes Verzeichnis archiviert werden (SIGNED);
2. **Packaging.** Signierte und nicht-signierte Dokumente werden für den Sedex-Versand aufbereitet.
3. **Delivery.** Versand nach Sedex-Manier.

Beim Empfang wird kein Unterschied zwischen signierten und nicht-signierten Dokumenten gemacht.

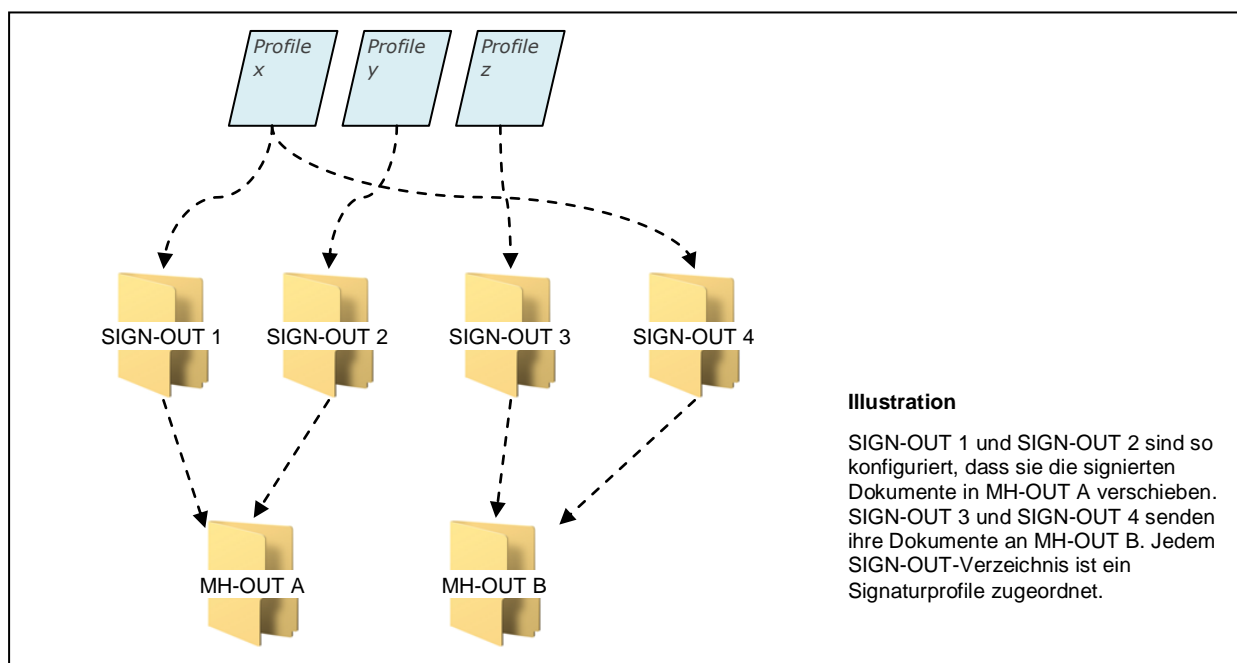
### 3.2 Signatur-Profiles

Für die Signatur werden Soft-Zertifikate (PKCS12) verwendet. Das Schlüsselmaterial ist gemäss dem aktuellen BS zu konfigurieren, d.h. es sind anzugeben: Zertifikatstyp (*certificatetype*), Speicherort des Zertifikats (*certificatefile*), das Passwort (*certificatepassword*) sowie das Profile (*signaturprofile*).

Bei Bedarf können mehrere SIGN-OUT Verzeichnisse konfiguriert werden, jedes hat einen eigenen Namen und ein dazugehöriges Signatur-Profil. (Ein Signatur-Profil hingegen kann von unterschiedlichen SIGN-OUT Verzeichnissen gleichzeitig verwendet werden.)

### 3.3 Relation zwischen SIGN-OUT und MH-OUT

Zu jedem MH-OUT gehört eine bestimmte Anzahl SIGN-OUT Verzeichnisse, mindestens jedoch eines. Ein bestimmtes SIGN-OUT Verzeichnis gehört zu genau einem MH-OUT Verzeichnis. In dieses sind die signierten Dokumente zu verschieben.



### 3.4 Adressierung

Signierte Dokumente erhalten den Suffix "-signed". Ansonsten gilt die Namenskonvention des eSchKG-Verbundes für XML und PDF-Dateien, also <recID>\_<send\_ID>\_<uuid>.<ext>.

Aus dem PDF Dokument

7-4-1\_7-76-1\_f81d4fae-7dec-11d0-a765-00a0c91e6bf6.pdf wird

7-4-1\_7-76-1\_f81d4fae-7dec-11d0-a765-00a0c91e6bf6-signed.pdf

### 3.5 Monitoring

Der neue MH unterstützt Logging und Monitoring, so dass die Anwendung jederzeit feststellen kann, ob Dokumente nach der Verschiebung in ein SIGN-OUT-Verzeichnis tatsächlich unterzeichnet und versandt worden sind.

### 3.6 Installation

Die Installation des neuen MH ist als Gesamtpaket zu konzipieren für Windows (XP, Win7, Vista) und Unix. Der Installationsvorgang ist intuitiv und der Benutzer wird dabei geführt. Es sind zwei Installationsoptionen zu realisieren:

- **full** = es werden eigene Runtimes für JVM, Groovy, etc. installiert und verwendet;
- **custom** = es wird auf bestehende Installationen Bezug genommen. Dabei ist besondere Unterstützung während des Installationsvorganges zu gewähren, z.B. Setzen von PATH-Variablen etc.

### 3.7 Auto-Konfiguration

Nach der Installation des "MH-Pakets" kann der Benutzer seine Konfiguration anhand von config-Files vornehmen.

Ausserdem ist ein Wizard zu realisieren, der den Benutzer im Dialogverfahren durch die Konfiguration führt und automatisch alle benötigten Konfigurationsdateien generiert (neu schreibt). Darunter fallen:

- Konstellation der Sedex-Adapter, d.h. entweder Installation eines eigenen Adapters nur für eSchKG oder aber Verwendung eines bestehenden Adapters;
- Definition von Anzahl und Benennung der SIGN-OUT Verzeichnisse;
- Definition von Anzahl und Benennung der MH-OUT Verzeichnisse;
- daraus hergeleitet die Auflösungsroutine `inboxNameResolver.groovy`;
- Spezielle Behandlung von "eSchKG-Installationen" des neuen MH: Vordefinierte Werte für den Nachrichtentyp (d.h. 10301) und vorgefertigte Auflösungs-Scripts (`resolver.groovy`) gemäss der Standard-Namenskonvention im eSchKG Verbund;
- Regelmässigkeit der Routine-Aufrufe (cron-Jobs);
- Logging-Optionen;
- Bei Beendigung des Wizards ist eine Checkliste / Todo-Liste zu erzeugen, worin die noch zu erledigenden Punkte aufgeführt sind, falls es welche gibt.

Zu prüfen ist ggf. eine Realisierung des Wizards im Web.

### 3.8 Dokumentation

Die Dokumentation umfasst den gesamten Installations- und Konfigurationsprozess, einschliesslich einer detaillierten Beschreibung der Konfigurationsdateien und der darin enthaltenen Optionen. Ausserdem sind Muster-Beispiele zu zeigen.

Das Ziel ist ein "Sorglospaket" für IT-Verantwortliche, die ihre Anwendung an den eSchKG Verbund anschliessen müssen. Darum ist eine zusätzliche Checkliste für Problembehandlungen zu liefern (z.B. fehlende Zugriffsrechte in Verzeichnisse, usw.).

Es sind Best-Practice für die Applikationsintegration zu erstellen. Diese zeigt auf, wie aus Applikationssicht vorzugehen ist, damit rechtzeitig auf Störungen reagiert werden kann, z.B. bei Problemen mit dem Signaturvorgang, dem SEDEX-Versand, oder auch der Abholungsmeldung (z.B. nach Ablauf einer bestimmten Frist).