Spezifikation ePVS V1.0

Herausgeber:
KV Telematik GmbH

Inhaltsverzeichnis

Änderungshistorie

Vers.	Datum	Autor	Kap.	Änderung	Status
0.1	27. 05. 2015	B. Bresser	alle	Initiale Erstellung	in Arbeit
0.9	29. 07. 2015	B. Bresser	alle	Fertigstellung Kommentierungsversion	zur Kommentierung freigegeben
1.0	13. 08. 2015	B. Bresser	alle	Geringfügige Modifikationen	Kommentierungsversion
1.1	15. 12. 2015	B. Bresser		Audit-Dokumentation	in Arbeit
1.1	18.01.2016	B. Bresser	alle	kleinere redaktionelle und sachliche Korrekturen und Klarstellungen	in Arbeit

Herausgeber:
KV Telematik GmbH

1 Vorbemerkungen

Dieses Dokument dient der Spezifikation des KV-Connect Anwendungsdienstes „elektronische Privatliquidation“. In den einzelnen Abschnitten werden die technischen Anforderungen näher erläutert, so dass eine Implementierung in Softwaresysteme der Ärzte leicht möglich ist.

1.1 Geltungsbereich

Die vorliegende Spezifikation gilt für alle Praxisverwaltungssysteme (PVS), die Abrechnungen außerhalb des kassenärztlichen Vertragsrahmens an Dienstleister für Privatliquidationen erzeugen und abliefern. Sie beschreibt den Prozess von der Aufbereitung der vorher erzeugten Berichtsdokumente für den Versand über den Nachrichtenaufbau, den Versand sowie den Empfang und den Inhalt von Quittungsdateien auf der Arzt-Seite.

Auf Seiten der Annahmestelle(n), also PADline oder eine der im Anhang bezeichneten Stellen direkt, wird der Prozess des Nachrichtenempfangs, der Prüfung der Nachrichten sowie der Erzeugung und des Versands von Quittungsnachrichten beschrieben.

Die Spezifikation gilt für alle Praxisverwaltungssysteme (PVS), die in KV-Gebieten eingesetzt werden, die die Übermittlung von Privatliquidationen über KV-Connect ermöglichen.

1.2 Referenzen

- [PP KVC]: Dokumentation zu KV-Connect im KV-Connect Partnerportal (https://partnerportal.kv-telematik.de)
- [KVC-Anb]: Anbindung an KV-Connect (Anbindung an KV-Connect-3_1_2.pdf) (Anbindung an KV-Connect)
- [KBV_ITA_VGEX]: KBV_ITA_VGEX_Anforderungskatalog_eDMP, KBV, (ftp://ftp.kbv.de/ita-update/Medizinische-Dokumentationen dort: "KBV_ITA_VGEX_Anforderungskatalog_eDMP.pdf")

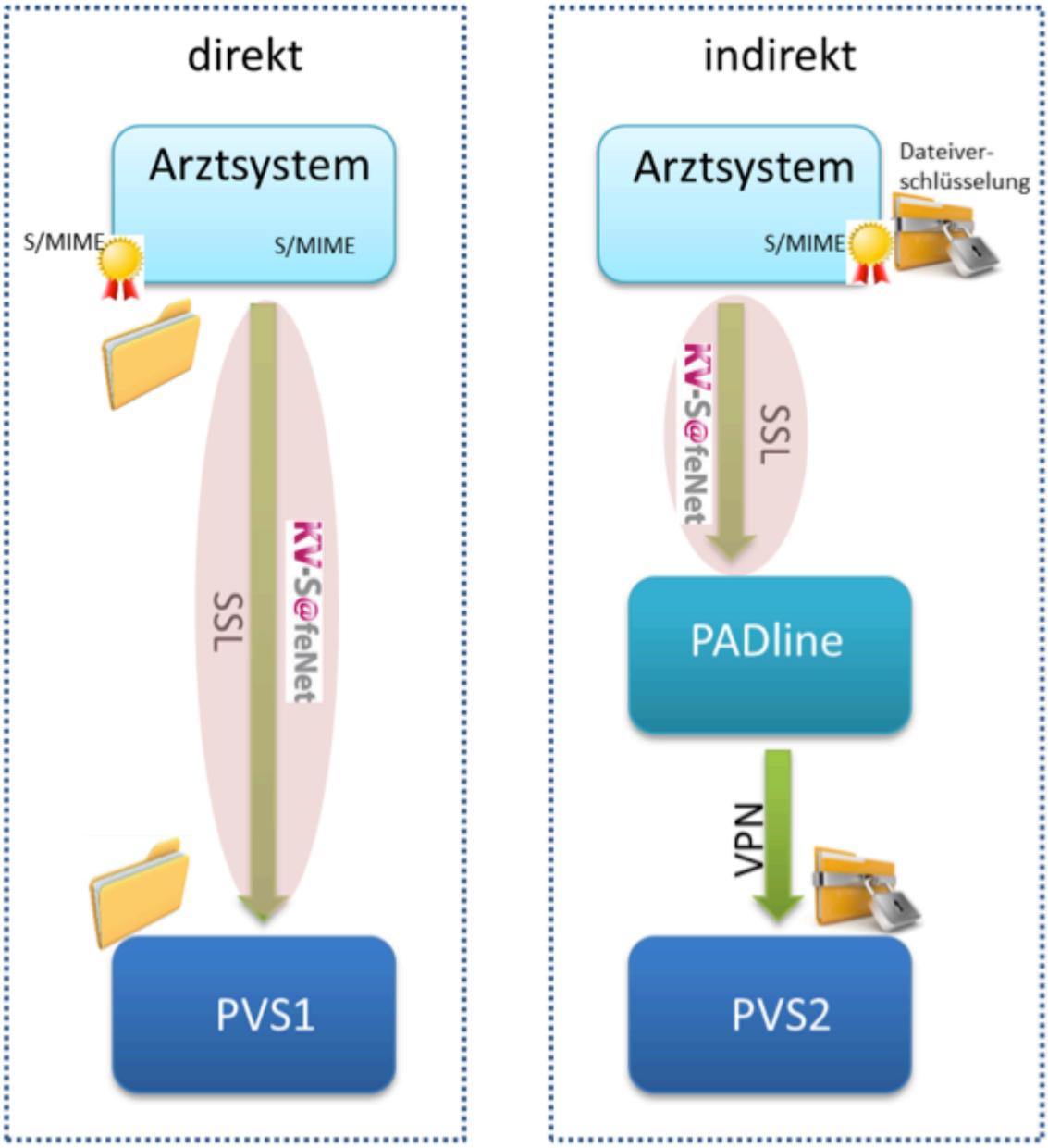
1.3 Bezug zur Auditierung

Die Implementierung aller KV-Connect-Anwendungen durch die Softwarehäuser werden im Rahmen einer Auditierung überprüft. Auditierungs-Kriterien, die sich auf die vorliegende Spezifikation beziehen, werden in den nachstehenden Kapiteln explizit als Auditierungs-Kriterien AK-n hervorgehoben. Wie z.B.

[PVSSM020] : Jede Sendung MUSS genau ein ZIP-Archiv mit Nutzdaten enthalten.

2 Der Übertragungsweg

Abrechnungsstellen, die dieses Verfahren (ePVS mit KV-Connect) unterstützen, können zum einen direkt erreicht werden oder zum andern indirekt über eine zentrale Servicestelle. Bei beiden Verfahren werden die Daten, so wie bei KV-Connect immer vorgeschrieben, vor dem Versand S/MIME verschlüsselt. Der Transport selbst erfolgt innerhalb eines SSL-gesicherten Tunnels. Die Nutzdaten werden zusammengefasst und zusätzlich noch einmal verschlüsselt. Dabei werden die Daten bei direkter Übertragung für den gleichen Empfänger verschlüsselt, der auch der Adressat der KV-Connect-Nachricht ist. Bei der indirekten Übertragung dagegen wird für den letztendlichen Empfänger der Nutzdaten verschlüsselt, somit kann die Servicestelle die Daten nicht einsehen. Der innere Container wird in der Folge von der Servicestelle an den eigentlichen Adressaten übertragen, der mittels seines privaten Schlüssels die für ihn vorgesehenen Daten zugänglich machen kann.



Wie die einzelnen Abrechnungsdienstleister zu erreichen sind, ist der folgenden Tabelle zu entnehmen.

Abrechnungsdienstleister	Übertragungsweg	KV-Connect-Adresse
PVS Baden-Württemberg	direkt	

Abrechnungsdienstleister	Übertragungsweg	KV-Connect-Adresse
PVS Rhein-Ruhr	indirekt	

3 Die Datenlieferung

Mit Datenlieferung werden die Daten, die zum Empfänger gesendet werden, bezeichnet. Dabei handelt es sich immer um eine komprimierte Datei (Zip-Datei), die eine Auftragsdatei , eine Abrechnungsdatei (PAD, PADneXt, BDT , Arztbrief) sowie optionale Anhänge zur Abrechnungsdatei enthält. Die Abrechnungsdatei und die möglichen Anhänge sind entsprechend zu verschlüsseln, siehe hierzu Abschnitt 4.

Die einzelnen Dateinamen enthalten jeweils eine Transfernummer. Diese Nummer wird fortlaufend vom Absender für einen Empfänger vergeben und muss der angegebenen Transfernummer in der Auftragsdatei entsprechen. Durch das Auswerten dieser Nummer können auf der Empfangsseite automatisch Lücken in der Verarbeitung erkannt werden.

Werden PADneXt Daten geliefert, gelten die Notationen und die Konvention der aktuellen PADneXt Schnittstellenbeschreibung (siehe auch "Komprimierte Daten").

3.1 Auftragsdateien

Auftragsdateien sind generell ohne Zusatzverschlüsselung zu übertragen. Pro Datenlieferung existiert genau eine Auftragsdatei. Der Name der Auftragsdatei setzt sich zusammen aus der Kundennummer, dem Erstellungsdatum, einem Kennzeichen über den Nachrichtentyp (ADL), einer Transfernummer und der Bezeichnung „auf“, gefolgt von der Endung „.xml“. Die Felder werden jeweils mit dem Zeichen „_“ getrennt.

Auftragsdatei	
Aufbau	*<KdNr.>* **_** *<Erstelldatum>* **_** *<Nachrichtentyp>* **_** *<Transfernr.>* **_auf.xml**
Format	*nnnnnnnn_JJJJMMTT_XXX_mmmmmm_auf.xml* *n: Die Nummern sind rechtsbündig und ggf. mit führenden Nullen anzugeben.* *JJJJMMTT: Datumsangabe im Format Jahr, Monat und Tag.X: Buchstabe in Großschrift.*
Beispiel	00123456_20150226_ADL_000001_auf.xml

3.2 Nutzdaten

Unter Nutzdaten werden alle Abrechnungsdateien plus optionaler Anhänge verstanden. Diese sind zunächst zu komprimieren und anschließend zu verschlüsseln (siehe Abschnitt 4). Die Eigenschaften der Dateien werden in der Auftragsdatei beschrieben. Der Name der Datei, die die Nutzdaten enthält setzt sich zusammen aus der Kundennummer, dem Erstellungsdatum, einem Kennzeichen über den Nachrichtentyp (ADL), einer Transfernummer und der Bezeichnung „nutz“, gefolgt von der gängigen Endung („.pad“, „.dat“, „.bdt“, „.xml“ oder andere). Die Felder werden jeweils mit dem Zeichen „_“ getrennt.

Nutzdaten, versch. Nachrichtentypen im XML Format	
Aufbau	*<KdNr.>* **_** *<Erstelldatum>* **_** *<Nachrichtentyp>* **_** *<Transfernr.>* **_nutz.<Extension>**
Format	nnnnnnnn_JJJJMMTT_XXX_mmmmmm_nutz.xml *n: Die Nummern sind rechtsbündig und ggf. mit führenden Nullen anzugeben.* *JJJJMMTT: Datumsangabe im Format Jahr, Monat und Tag.X: Buchstabe in Großschrift.*
Beispiel	00123456_20150226_ADL_000001_nutz.dat

Der Dateiname für Anhänge setzt sich zusammen aus der Kundennummer, der Transfernummer und einer pro Datenübermittlung zu generierenden Anhangsnummer.

Nutzdaten, Anhänge im PDF-, JPEG oder TIFF Format	
Aufbau	*<KdNr.>* _ *<Transfernr.>* _ *<LfdNr.>* **.[pdf|jpg|tiff|txt]**
Format	nnnnnnnn_mmmmmm_kkk.pdf nnnnnnnn_mmmmmm_kkk.jpg *n: Die Nummern sind rechtsbündig und ggf. mit führenden Nullen anzugeben.*
Beispiel	00123456_000001_001.pdf 00123456_000001_002.jpg 00123456_000001_003.tiff 00123456_000001_004.txt

3.3 Komprimierte Daten

Jede Datenlieferung ist immer als eine komprimierte Datei an den Empfänger zu übertragen (Zip Datei). Neben dem Vorteil, dass die Transfergröße der Daten erheblich reduziert wird, ist auch die Analyse bei Übertragungsfehlern einfacher zu realisieren, da nur eine Datei übertragen wird und nicht mehrere. Der Dateiname besitzt den gleichen Aufbau wie die Auftrag- und Nutzdaten. Die Dateien innerhalb der Zip-Datei sind ohne Pfadangaben anzugeben.

Datenlieferung: Dateiname	
Aufbau	*<KdNr.>* _ *<Erstelldatum>* _ *<Nachrichtentyp>* _ *<Transfernr.>* **_kvc.zip** bzw. *<KdNr.>*_*<Erstelldatum>*_*<Nachrichtentyp>*_*<Transfernr.>* **_padx.zip** (f. PADneXt-Dateien
Format	nnnnnnnn_JJJJMMTT_XXX_mmmmmm_kvc.zip bzw. nnnnnnnn_JJJJMMTT_XXX_mmmmmm_padx.zip *n: Die Nummern sind rechtsbündig und ggf. mit führenden Nullen anzugeben.* *JJJJMMTT: Datumsangabe im Format Jahr, Monat und Tag.* *X: Buchstabe in Großschrift.*
Beispiel	00123456_20150226_ADL_000001_kvc.zip

Der Inhalt der komprimierten Datei besteht aus der unverschlüsselten Auftragsdatei und den verschlüsselten Nutzdaten (Abrechnungsdatei und optionale Anhänge).

4 Datei(zusatz)verschlüsselung

Generell werden beim Transport mit KV-Connect alle Inhaltsdaten verschlüsselt. Die Anwendung "ePVS" sieht jedoch ergänzend gegebenenfalls eine weitere Vertraulichkeitsebene vor. Als (Zusatz-) Verschlüsselungsverfahren kommt PKCS#7 zum Einsatz. Für die Verschlüsselung wird der öffentliche Schlüssel des Empfängers benötigt, bei der Entschlüsselung ist der jeweilige zugehörige private Schlüssel notwendig. Die verfahrensspezifischen Merkmale wie Schlüssellängen, Kodierungsrichtlinien und Algorithmen werden dabei in der Nachricht selbst gespeichert.

4.1 Aufbau der Datei

Der Inhalt der Datenlieferung ist komprimiert und besteht immer aus genau zwei Dateien: die unverschlüsselte Auftragsdatei und die in einer Datei komprimierten und verschlüsselten Nutzdatendateien. Dazu sind alle Nutzdatendateien zunächst in einer Datei zu komprimieren und dann zu verschlüsseln. Damit der Dateiname des Archivs sich von dem Namen der Datenlieferungsdatei unterscheidet, wird das Kürzel *_dat* im Dateinamen angegeben. Weiterhin wird im Namen nach PADline- und PADneXt-Dateien unterschieden. Werden PADneXt-Daten geliefert, dann gelten die Notationen und Konventionen der aktuellen PADneXt-Schnittstellenbeschreibung.

Datenlieferung: Inhalt, Nachrichtentyp ADL	
Aufbau	***(Zusatz-)Verschlüsselte Datei:*** *<KdNr.>_<Erstelldatum>_<Nachrichtentyp>_<Transfernr.>***_kvc.zip** bzw. *<KdNr.>_<Erstelldatum>_<Nachrichtentyp>_<Transfernr.>***_padx.zip**
Format	**Inhalt der o.a. Zip-Datei (Datenlieferung):** nnnnnnnn_JJJJMMTT_XXX_mmmmmm_auf.xml : *Auftragsdatei* nnnnnnnn_JJJJMMTT_XXX_mmmmmm_dat_kvc.zip.p7m : *Nutzdaten verschlüsselt u. kompr.* **Inhalt der p7m-Datei:** nnnnnnnn_JJJJMMTT_XXX_mmmmmm_dat_kvc.zip : *Nutzdaten komprimiert* **Inhalt der Zip-Datei (nur Nutzdaten):** nnnnnnnn_JJJJMMTT_XXX_mmmmmm_nutz.<Extension> nnnnnnnn_mmmmmm_kkk.pdf nnnnnnnn_mmmmmm_kkk.jpg
Beispiel	**Finale ZIP-Datei** (äquivalent für _padx.zip-Dateien)**:** 00123456_20150226_ADL_000001_kvc.zip **Inhalt der o.a. Zip-Datei (Datenlieferung):** 00123456_20150226_ADL_000001_auf.xml 00123456_20150226_ADL_000001_dat_kvc.zip.p7m **Inhalt der p7m-Datei:** 00123456_20150226_ADL_000001_dat_kvc.zip **Inhalt der Zip-Datei (nur Nutzdaten):** 00123456_20150226_ADL_000001_nutz.dat 00123456_000001_001.pdf 00123456_000001_002.jpg

Das folgende Beispiel zeigt die einzelnen durchzuführenden Schritte, ausgehend von den Rohdaten bis zur komprimierten und verschlüsselten Übertragungsdatei. Die Komprimierung der Nutzdaten ist dabei vor der Verschlüsselung vorzunehmen. Die verschlüsselte Datei bekommt die Endung p7m und ist nun zusammen mit der Auftragsdatei in einem neuen Archiv zu speichern.

In dem Beispiel wird eine gültige Datenlieferung für einen Leistungserbringer mit der Kundennummer 123456 vom 26. Februar 2015 angegeben. Laut Transfernummer handelt es sich um die 42. Datensendung und enthält neben den Rechnungsdaten noch zwei Anhänge. Die Nutzdaten sind dabei verschlüsselt in der Datenlieferung zu hinterlegen.

1. Dateien der Datenlieferung

```
Auftragsdatei: 00123456_20150226_ADL_000042_auf.xml

Nutzdaten:          00123456_20150226_ADL_000042_nutz.dat

Anhang 1:           00123456_000042_001.pdf

Anhang 2:           00123456_000042_002.jpg
```

2. Komprimierung der Daten

```
Nur die Nutzdaten und die beiden Anhänge sind zunächst zu komprim

Neue Datei: 00123456_20090526_ADL_000042_dat_kvc.zip

Inhalt:

00123456_20150226_ADL_000042_nutz.dat

00123456_000042_001.pdf

00123456_000042_002.jpg
```

3. (Zusatz-)Verschlüsselung

```
Die Zip-Datei wird nach PKCS#7 mit dem öffentlichen Schlüssel des

Neue Datei: 00123456_20150226_ADL_000042_dat_kvc.zip.p7m
```

4. Datenlieferung

```
Zum Schluss ist eine neue Archivdatei zu erstellen, die die o. a.
 verschlüsselte Nutzdatendatei und die Auftragsdatei enthält.


Neue Datei: 00123456_20150226_ADL_000042_kvc.zip

Inhalt:

00123456_20150226_ADL_000042_auf.xml

00123456_20150226_ADL_000042_dat_kvc.zip.p7m
```

4.2 Schlüsselservice

Damit die Dateien verschlüsselt werden können ist ein Public-Key der jeweiligen Abrechnungsstelle notwendig. Um immer den aktuellen Schlüssel zu verwenden, ist ein Web-Service eingerichtet worden, über den man den aktuellen Schlüssel automatisch beziehen kann.

Die dafür benötigte ID der Abrechnungsstelle findet man in den Schlüsseltabellen.

Wichtig:	**Für die innere Verschlüsselung ist der öffentliche Schlüssel zu verwenden, der dem logischen Empfänger (Abrechnungsstelle) der KV-Connect-Nachricht zugeordnet ist. Beim *indirekten* Versand ist dies daher ein anderer Schlüssel als für die äußere KV-Connect Verschlüsselung (an die PADline).** **Beim *direkten* Versand erfolgen innere und äußere Verschlüsselung mit dem gleichen Schlüssel, daher ist die innere Verschlüsselung hier optional. Es wird jedoch empfohlen der Einfachheit halber immer mit der inneren Zusatzverschlüsselung zu arbeiten.**

Bei **Zugriff aus dem KV-SafeNet** lautet die URL für den Schlüsselservice beispielhaft:

https://padtransfer.kv-safenet.de/zertifikat/v1/300.pem

Der Dateiname besteht aus der ID der Abrechnungsstelle (siehe Schlüsseltabellen) für die Sie den Schlüssel benötigen und dem Format als Dateiendung (.pem oder .cer).

In dem Beispiel erhalten Sie den public Key der PADline (ID = 300) im PEM-Format.

Möchten Sie lieber ein Zertifikat im CER-Format dann lautet die URL beispielhaft:

https://padtransfer.kv-safenet.de/zertifikat/v1/300.cer

Einige ID’s repräsentieren eine Bezirksstelle einer PVS. Wenn dort dann z.B. die ID 801 angegeben wird, wird der Key der Hauptgeschäftsstelle, in diesem Fall 800, geliefert.

Bei **Zugriff über das Internet** werden ID und Format durch Parameter angegeben. Ohne Format Parameter erhält man das PEM-Format, die URL sieht dann so aus:

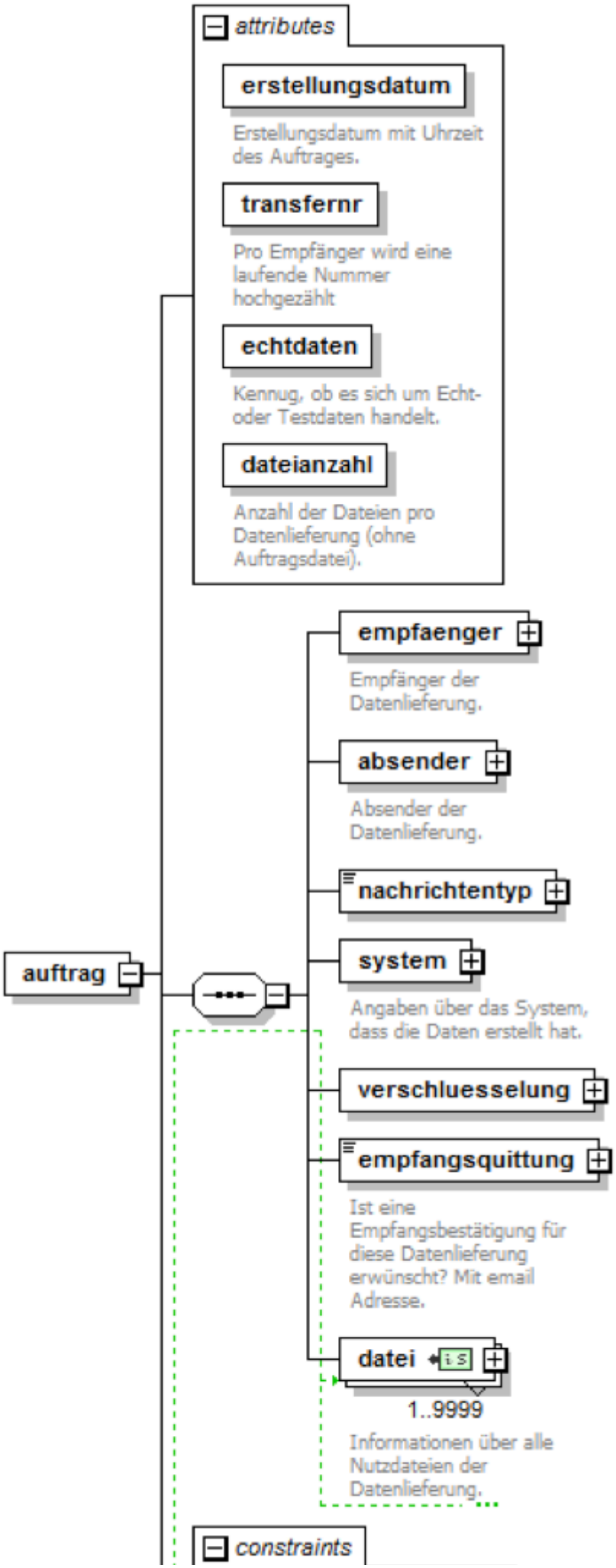
https://webservice.padline.de/zertifikat/v1?id=300

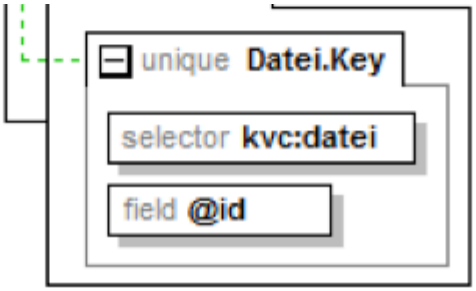
und für das CER-Format:

https://webservice.padline.de/zertifikat/v1?id=300&format=cer

5 Auftragsdatei

Die Auftragsdatei kann nur einmal pro Datenlieferung vorkommen und enthält alle Informationen zu den Nutzdaten, um diese automatisiert verarbeiten zu können. Alle Attribute und Elemente sind als Mussfelder definiert, d. h. Es sind alle aufgeführten Datenfelder anzugeben. Mit "auftrag" wird das Wurzelelement für die Auftragsdatei bezeichnet und kann demnach nur einmal pro Datei vorkommen.


attributes
erstellungsdatum
Erstellungsdatum mit Uhrzeit des Auftrages.
transfernr
Pro Empfänger wird eine laufende Nummer hochgezählt
echtdaten
Kennug, ob es sich um Echt- oder Testdaten handelt.
dateianzahl
Anzahl der Dateien pro Datenlieferung (ohne Auftragsdatei).
auftrag
empfaenger
Empfänger der Datenlieferung.
absender
Absender der Datenlieferung.
nachrichtentyp
system
Angaben über das System, dass die Daten erstellt hat.
verschluesselung
empfangsquittung
Ist eine Empfangsbestätigung für diese Datenlieferung erwünscht? Mit email Adresse.
datei
1..9999
Informationen über alle Nutzdateien der Datenlieferung.
constraints



Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
@erstellungsdatum	1	dateTime	Erstellungsdatum mit Uhrzeit des Auftrages.	
@transfernr	1	posInt(6)	Pro Empfänger wird eine laufende Nummer pro Datenlieferung hochgezählt.	
@echtdaten	1	boolean	Kennung, ob es sich um Echt- oder Testdaten handelt. **0**: Testdaten (dürfen nicht im produktiven System verarbeitet werden). **1**: Echtdaten	
@dateianzahl	1	posInt(3)	Anzahl der Dateien pro Datenlieferung (exkl. Auftragsdatei), maximal 999 erlaubt.	
empfaenger	1	complex	Angaben über den Empfänger der Daten, hier wird zwischen dem tatsächlichen Empfänger und der Datenannahmestelle unterschieden.	5.1
absender	1	complex	Hier werden alle Informationen über den Absender und Ersteller der Daten gespeichert.	5.2
nachrichtentyp	1	complex	Enthält genauere Angaben über die Art der Daten mit Angabe einer Versionskennung.	5.3
system	1	complex	Angaben über die Software, die diese Daten erstellt hat.	5.4
verschluesselung	1	complex	Kennzeichen, ob die Nutzdaten in verschlüsselter Form oder im Klartext vorliegen.	5.5
empfangsquittung	1	complex	Angabe, ob der Absender eine Quittierung für diese Datenlieferung erhalten möchte.	5.6
datei	1..9999	complex	Alle Informationen über die einzelnen Dateien dieser Datenlieferung.	5.7

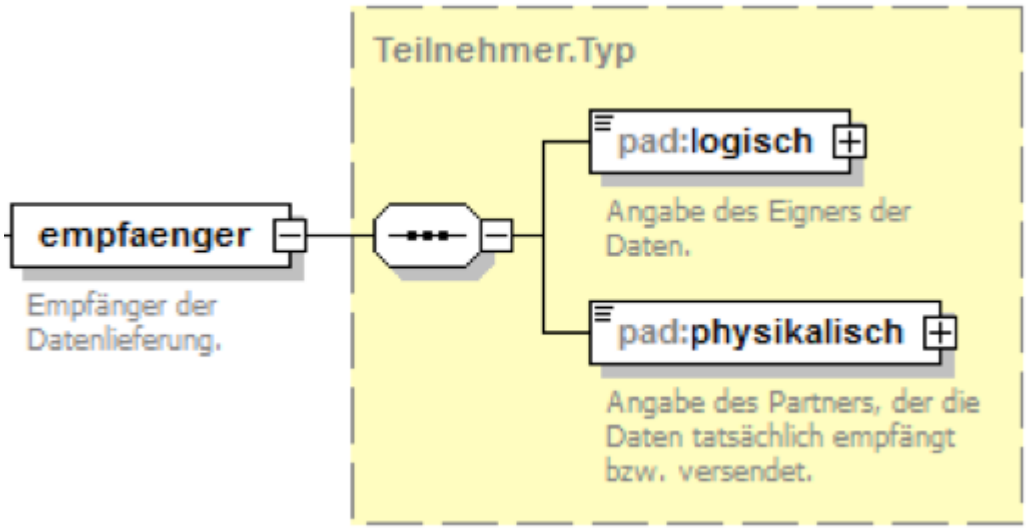
```
<auftrag transfernr="12345" erstellungsdatum="2001-12-17T09:30:47Z" echtdaten
       <empfaenger> … </empfaenger>
       <absender> …  </absender>
       <nachrichtentyp version="02.10">ADL</nachrichtentyp>
       <system> … </system>
       <verschluesselung verfahren="0"/>
       <empfangsquittung> … </empfangsquittung>
       <datei erstellungsdatum="2001-12-17T09:30:47Z" id="12345"> … </datei>
</auftrag>
```

5.1 Empfänger

Es bleibt dem umsetzenden Softwarehaus überlassen, ob es aus den verfügbaren Informationen zum Patienten die KV-Connect-Adresse der zuständigen Liquidationsstelle zwingend auswählt, diese als Default-Einstellung in einer Auswahl-Box setzt oder dem Anwender die Auswahl aus der Adressliste vollständig überlässt. Es muss außerdem sichergestellt werden, dass die richtige Auswahl bei der Unterscheidung des „logischen“ und des „physikalischen“ Empfängers sachgerecht funktioniert.

Es SOLLTE vom Primärsystem eine Auswahl-Box mit Default-Vorauswahl angeboten werden.

Das Element "empfaenger" setzt sich aus zwei Einzelelementen zusammen, die eine identische Struktur besitzen: logisch und physikalisch. Die Unterscheidung ermöglicht es Routinginformationen anzugeben, um verschlüsselte Daten über eine Zwischenstelle zum eigentlichen Empfänger zu senden. Die Identifizierung der Partner wird über unterschiedliche Attribute in den Elementen vorgenommen, die jeweils eindeutig sind. Je nach Transferrichtung (vom oder zum Arzt) sind die Attribute anzugeben.



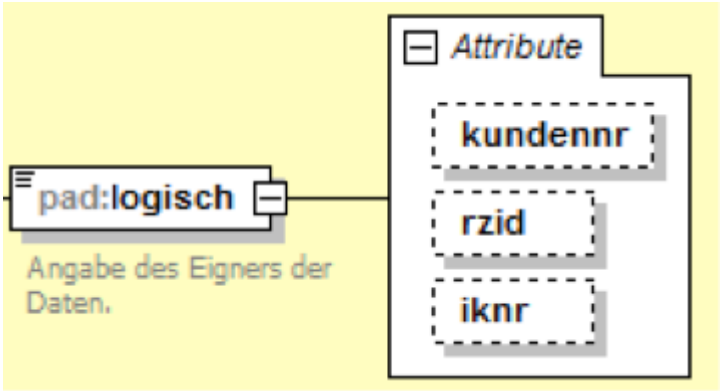
Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
logisch	1	complex		

Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
			Angabe des Empfängers, für den die Nutzdaten zur Verarbeitung bestimmt sind (für diesen werden die Daten auch verschlüsselt).	
physikalisch	1	complex	Die Datenlieferung wird an den physikalischen Empfänger gesendet.	

```
<empfaenger>
       <logisch> … </logisch>
       <physikalisch> …  </physikalisch>
</empfaenger>
```

5.1.1 Empfänger.logisch

Unter dem logischen Empfänger wird der eigentliche Adressat der Daten verstanden, unabhängig vom Transferweg. Ggf. werden für diesen Empfänger die Daten verschlüsselt und sind somit auch nur von diesem zu entschlüsseln. Bei einer Übertragung von Arztdaten an eine Abrechnungsstelle, wird diese über eine eindeutige Identifikationsnummer adressiert.



Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
@kundennr	0..1	posInt (20)	Eindeutige Kundennummer des Arztes oder der Praxis bei der Abrechnungsstelle	
@rzid	0..1	posInt (4)	Eindeutige Identifikationsnummer der Abrechnungsstelle oder des Rechenzentrums Die gültigen Nummern sind in den Schlüsseltabellen angegeben	7.1
@iknr	0..1	posInt (9)	Eindeutige IK-Nummer der Praxis, des Krankenhauses oder der Abrechnungsstelle	

Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
logisch	1	string	Angabe des logischen Empfängernamen	

```
<!-- Beispiel für Transfer Arzt zur PVS Niedersachsen / Lüneburg -->
<logisch rzid="0205" iknr="220330010">PVS Niedersachsen/Lüneburg</logisch>

<!-- Beispiel für Transfer Abrechenzentrum zum Arzt -->
<logisch kundennr="4711">Dr. Müller</logisch>
```

5.1.2 Empfänger.physikalisch

Der physikalische Empfänger gibt den Empfänger an, an den die Nachricht gesendet wird. Dort wird der logische Empfänger ausgewertet und die Daten werden u. U. entsprechend weitergeleitet. Wenn die Daten nicht über eine Vermittlungsstelle übermittelt werden, sondern direkt zum Empfänger, sind diese Angaben identisch mit den Daten für den logischen Empfänger der Nachricht.

Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
@kundennr	0..1	posInt (20)	Eindeutige Kundennummer des Arztes oder der Praxis bei der Abrechnungsstelle	
@rzid	0..1	posInt (4)	Eindeutige Identifikationsnummer der Abrechnungsstelle oder des Rechenzentrums Die gültigen Nummern sind in den Schlüsseltabellen angegeben	7.1
@iknr	0..1	posInt (9)	Eindeutige IK-Nummer der Praxis, des Krankenhauses oder der Abrechnungsstelle	
physikalisch	1	string	Angabe des physikalischen Empfängernamen. Wenn der Partner die Daten direkt entgegen nimmt, ist dieser Eintrag identisch mit dem Wert für den logischen Empfänger.	

```
<!-- Beispiel für Transfer Arzt zur PVS über die Zentrale PADline -->
<physikalisch rzid="0300" iknr="660330076">PADline GmbH</physikalisch>

<!-- Beispiel für Transfer PVS zum Arzt (direkt) -->
<physikalisch kundennr="4711">Dr. Müller</physikalisch>
```

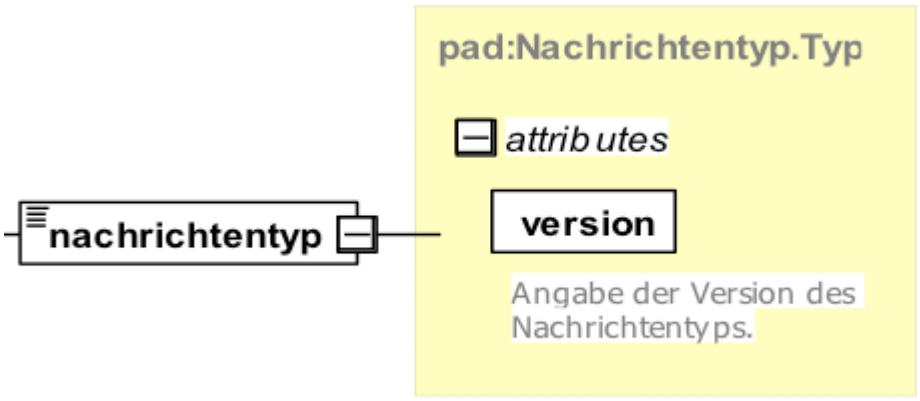
5.2 Absender

In dem Element "absender" werden die Angaben zu dem Ersteller der Datenlieferung angegeben. Die Struktur ist identisch mit der unter dem Element "empfaenger" angegebenen Definition (siehe Auftragsdatei, erster Unterpunkt). Der logische Absender enthält z. B. die Informationen des Leistungserbringers (Name mit Kundennummer der Abrechnungsstelle oder IK Nummer). Der Inhalt im physikalischen Absender ist nur abweichend, wenn der Nachrichtentransport über einen Vermittler durchgeführt wird, über den auch Quittungsnachrichten zurück gesendet werden müssen.

```
<!-- Beispiel für Transfer Arzt zur PVS -->
<absender>
	<logisch kundennr="3344">Dr. Musterarzt</logisch>
	<physikalisch kundennr="3344">Dr. Musterarzt</physikalisch>
</absender>
```

5.3 Nachrichtentyp

Der Nachrichtentyp gibt die Art der Nutzdaten an und definiert die Zusammenstellung der einzelnen Klassen. Die Angabe in der Auftragsdatei ist für die automatische Verarbeitung notwendig.

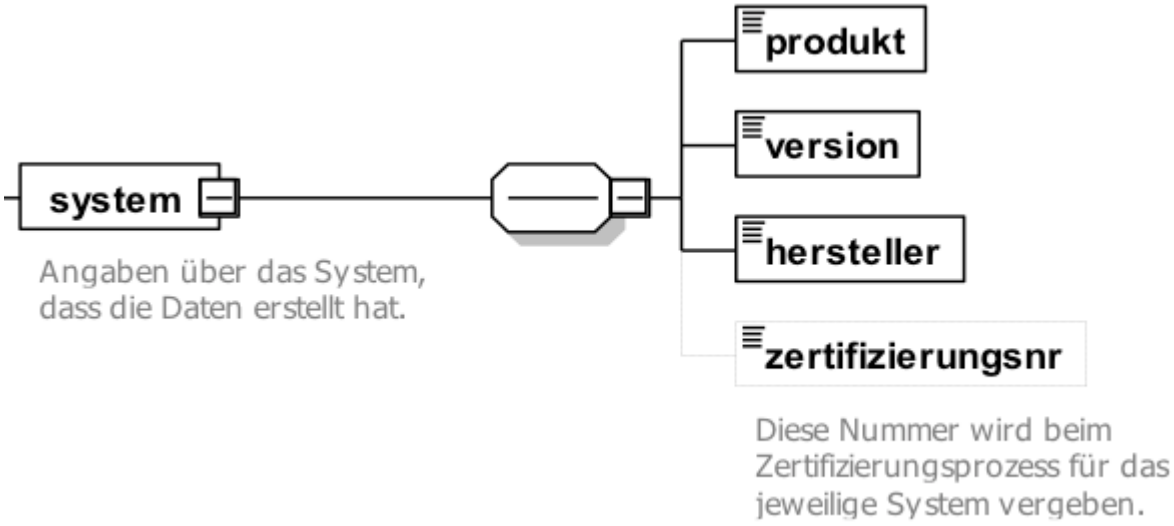


Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
nachrichtentyp	1	string(4)	Angabe des Nachrichtentyps Die folgenden Werte sind bisher definiert: **ADL: Arzt – DatenlieferungQADL: Quittung Arzt - Datenlieferung**	
@version	1	string(5)	Angabe der Versionsnummer für den jeweiligen Nachrichtentyp Das Format besteht aus einer Haupt- und einer Nebenversionsnummer, getrennt von einem „.“ Jede Nummer ist als zweistellige Zahl definiert und wird in den Daten ggf. mit einer vorangestellten 0 angegeben	

```
<!-- Datenlieferung mit Rechnungsdaten an die Abrechnungsstelle -->

<nachrichtentyp version="01.0">ADL</nachrichtentyp>

<!--Kennung für Quittungsnachricht von der Abrechnungsstelle -->

<nachrichtentyp version="01.0">QADL</nachrichtentyp>
```

5.4 System

In der Klasse "system" werden Informationen über das Softwaresystem des Herstellers gespeichert. Diese Angaben können für die automatisierten Prozessabläufe bei der Abrechnungsstelle eine wichtige Rolle spielen.

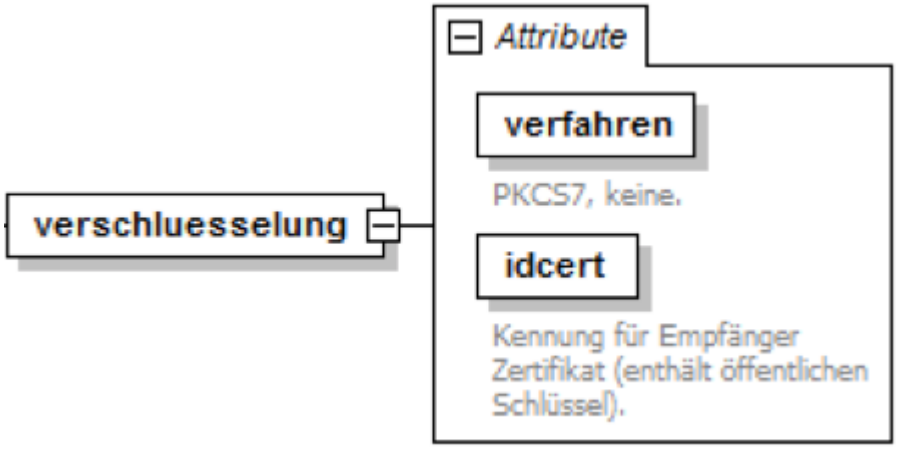


Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
Produkt	1	string (40)	Produktname	
Version	1	string (20)	Produktversion	
Hersteller	1	string (40)	Name des Softwareherstellers	
zertifizierungsnr	0..1	string (20)	Diese Nummer wird beim Zertifizierungsprozess für das jeweilige System vergeben	

```
<system>
	<produkt>RG-Info</produkt>
	<version>3.0</version>
	<hersteller>quadcore GmbH</hersteller>
	<zertifizierungsnr>PADx-AIS-2009-001</zertifizierungsnr>
</system>
```

5.5 Verschlüsselung

Kennzeichen, ob die Nutzdaten in verschlüsselter Form oder im Klartext vorliegen. Die Nutzdaten über den indirekten Weg sind immer verschlüsselt zu übertragen (Kennung mit „1“ angeben). Bei einer direkten Übertragung sind die Attribute entsprechend mit dem Kennzeichen „0“ anzugeben.

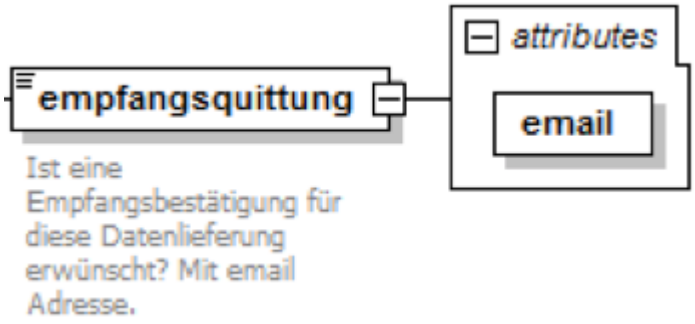


Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
@verfahren	1	decimal (1)	Die folgenden Werte sind zulässig: **0**: keine Verschlüsselung der Nutzdaten**1**: Nutzdaten sind nach PKCS#7 verschlüsselt	
@idcert	0..1	string (128)	Kennung für Empfängerzertifikat, das den öffentlichen Schlüssel enthält.	

```
<!--Datenlieferung ohne Verschlüsselung -->
<verschluesselung verfahren="0" idcert="0"/>

<!--Datenlieferung mit Verschlüsselung -->
<verschluesselung verfahren="1" idcert="C0:C8:F7:A0:33:20:A2:D4:2E:27:65:73:4
```

5.6 Empfangsquittung

Der Inhalt im Feld "empfangsquittung" gibt an, ob der Ersteller der Daten eine Quittierung der Datenlieferung wünscht oder nicht. Die Quittungsnachricht wird über ein separates Schema definiert, das in Kapitel ?0? näher beschrieben ist.



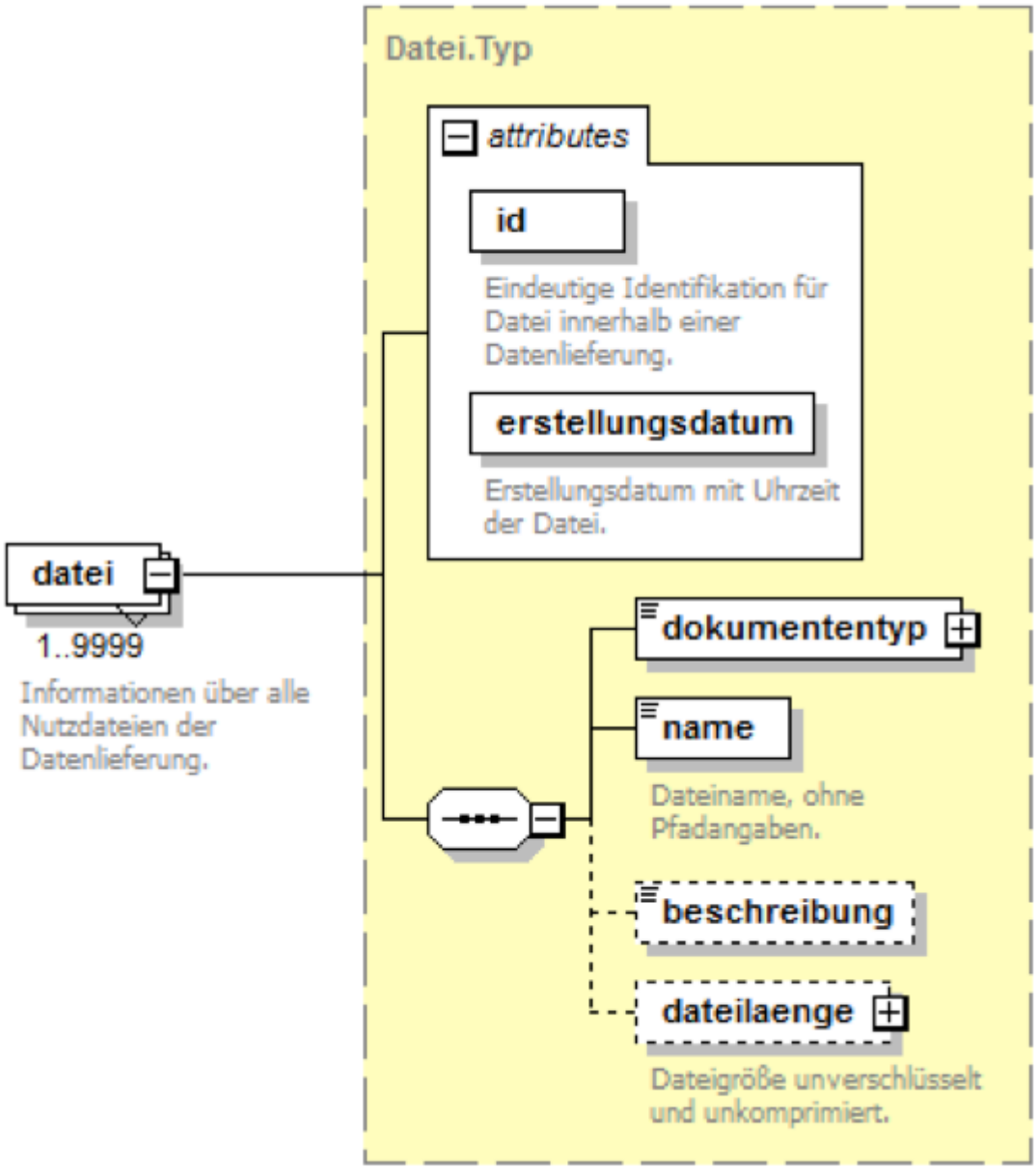
Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
empfangsquittung	1	boolean	**0**: Keine Quittierung erforderlich.**1**: Datenlieferung soll vom Empfänger quittiert werden.	
@email	1	string (100)	Angabe der email Adresse, nur notwendig wenn eine Quittierung erfolgt.(**KV-Connect-Adresse**)	

```
<!--Datenlieferung soll quittiert werden -->

<empfangsquittung>1</empfangsquittung>
```

5.7 Datei

In der Struktur datei werden alle Informationen über eine Datei innerhalb der Datenlieferung angegeben. Mit diesen Informationen sind einige Konsistenzprüfungen auf der Empfängerseite möglich.

Zum Beispiel werden durch die technischen Prozesse der Komprimierung, Verschlüsselung und Transport die Nutzdaten auf der Erstellerseite im Format verändert (inhaltlich bleiben sie unberührt). Auf der Empfängerseite sind diese Vorgänge entsprechend umzukehren. Um die Konsistenz der Daten zu gewährleisten, ist diese Struktur auszuwerten und mit den vorliegenden Daten zu vergleichen.



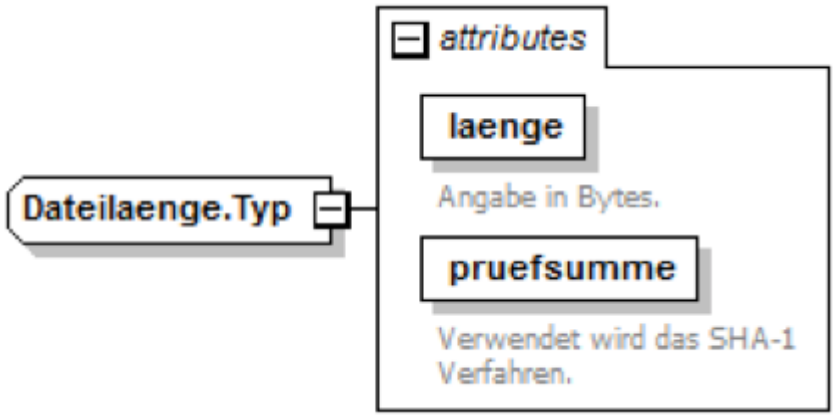
Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
@id	1	string (40)	Eindeutiger Bezeichner für die Datei innerhalb einer Datenlieferung Die Zugehörigkeit zu einer Rechnung bzw. Abrechnungsfall wird über diesen Bezeichner definiert (in den Rechnungsdaten wird auf diesen referenziert)	
@erstellungsdatum	1	dateTime	Datum und Uhrzeit der Erstellung der Datei	
dokumententyp @format	1 0..1	string (10) string (10)	Kennzeichen für Nutzdaten (PADneXt, PAD, Anhang) Angabe eines MIME-Typ (z. B. jpeg oder pdf)	7.2 7.1
name	1		Dateiname	

Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
		string (40)		
beschreibung	0..1	string (60)	Beschreibungstext für die Datei	
dateilaenge	0..1	complex	Angabe der unkomprimierten und unverschlüsselten Dateigröße	5.7.1

```
<datei erstellungsdatum="2001-12-17T09:30:47Z" id="1239">
       <dokumententyp>PADneXt</dokumententyp>
       <name>00123456_20090526_ADL_000042_padx.xml</name>
       <beschreibung>Optionaler Beschreibungstext</beschreibung>
</datei>
```

5.7.1 Dateilänge

In "Dateilaenge" wird die originale Dateigröße angegeben (ohne Kompression und Verschlüsselung). Diese Dateilänge muss sich auf der Empfängerseite nach der Umkehrung der Prozesse ergeben.



Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
@laenge	1	unsignedLong	Angabe der Dateigröße in Bytes der unverschlüsselten und unkomprimierten Datei	
@pruefsumme	1	string(40)	Prüfsumme im Format eines Hashwertes nach dem SHA-1 Verfahren	

```
<!--Datenlieferung einer Datei mit der Größe von 42.353 Bytes -->

<dateilaenge laenge="42353" pruefsumme="1fdsff4354325jhkjhdkjh3894234jkfdhh32
```

5.8 Beispiel

Im folgenden Beispiel wird eine Auftragsdatei für eine Datenlieferung von Rechnungen dargestellt, die von Dr. Müller, Kunde mit der Nummer 4711 bei der PVS Niedersachsen in Lüneburg, elektronisch an seine PVS versendet werden soll. Die Nachricht wird über die PADline GmbH als Vermittlungsstelle übertragen. Das tatsächliche Übertragungsverfahren zwischen den Instanzen spielt hierfür keine Rolle.

Dr. Müller ist schon längere Zeit Kunde bei der PVS und hat schon einige PADneXt Daten versendet, die aktuelle Datenlieferung hat die Nummer 42 bekommen. Sie wurde am 17.05.2011 erstellt und enthält 1.312 Rechnungen. Dr. Müller hat mit seiner PVS vereinbart ggf. zusätzliche Informationen als Anhang im PDF Format mitzuliefern. In diesem Fall wurde ein Anhang mit zusätzlichen Abrechnungshinweisen für die PVS erstellt. Die Daten wurden verschlüsselt und eine Empfangsquittung angefordert (die Antwortnachricht ist unter Quittung, Unterpunkt 4 angegeben).

```
<!-- Auftragsdatei für Datenlieferung mit Rechnungen vom Arzt zur PVS -->

<Auftrag transfernr="42" echtdaten="true" erstellungsdatum="2011-05-17T09:29:
         dateianzahl="2" xsi:schemaLocation="http://padinfo.de/ns/pad padx_au
         xmlns="http://padinfo.de/ns/pad" xmlns:xsi="http://www.w3.org/2001/X

       <empfaenger>

              <logisch iknr="220330010" rzid="205">PVS Niedersachsen/Lüneburg

              <physikalisch rzid="300">PADline GmbH</physikalisch>

       </empfaenger>

       <absender>

              <logisch kundennr="4711">Dr. Müller</logisch>

              <physikalisch kundennr="4711">Dr. Müller</physikalisch>

       </absender>

       <nachrichtentyp version="02.10">ADL</nachrichtentyp>

       <system>

              <produkt>AIS 123</produkt>

              <version>3.00</version>

              <hersteller>Medicus KG</hersteller>

              <zertifizierungsnr>PADx-AIS-2010-099</zertifizierungsnr>

       </system>

       <verschluesselung verfahren="1" idcert="C0:C8:F7:A0:33:20:A2:D4:2E:27:

       <empfangsquittung>1</empfangsquittung>
```

```
	<datei erstellungsdatum="2011-05-17T08:30:47Z" id="2011-42-0001">
		<dokumententyp>PADneXt</dokumententyp>
		<name>00004711_20110517_ADL_000042_padx.xml</name>
		<beschreibung>Rechnungsdaten über PADneXt Schnittstelle.</besch
		<dateilaenge pruefsumme="5158b469699c3ae7bae26ee7470665798c632e
	</datei>
	<datei erstellungsdatum="2011-05-17T08:45:17Z" id="2011-42-0002">
		<dokumententyp format="pdf">Anhang</dokumententyp>
		<name>00004711_000042_001.pdf</name>
		<beschreibung>Zus. Abrechnungshinweise zu den Rechnungen.</besc
		<dateilaenge pruefsumme="7739bb69b99cfae7bde269e7465665794c6322
	</datei>
</Auftrag>
```

6 Abrechnungshinweise

Ärzte übermitteln häufig Abrechnungshinweise zu den Abrechnungsdateien. Das sind Texte mit denen sie spezielle Wünsche zu einzelnen Abrechnungen dokumentieren.

Solche Texte sind bitte als Anhang zu sehen und verschlüsselt innerhalb der Nutzdaten zu übertragen.

Der Dateiname muss der PADneXt-Spezifikation entsprechen. Als Endung der Datei ist „.txt“ zu verwenden.

Beispiel:

00123456_000001_001.txt

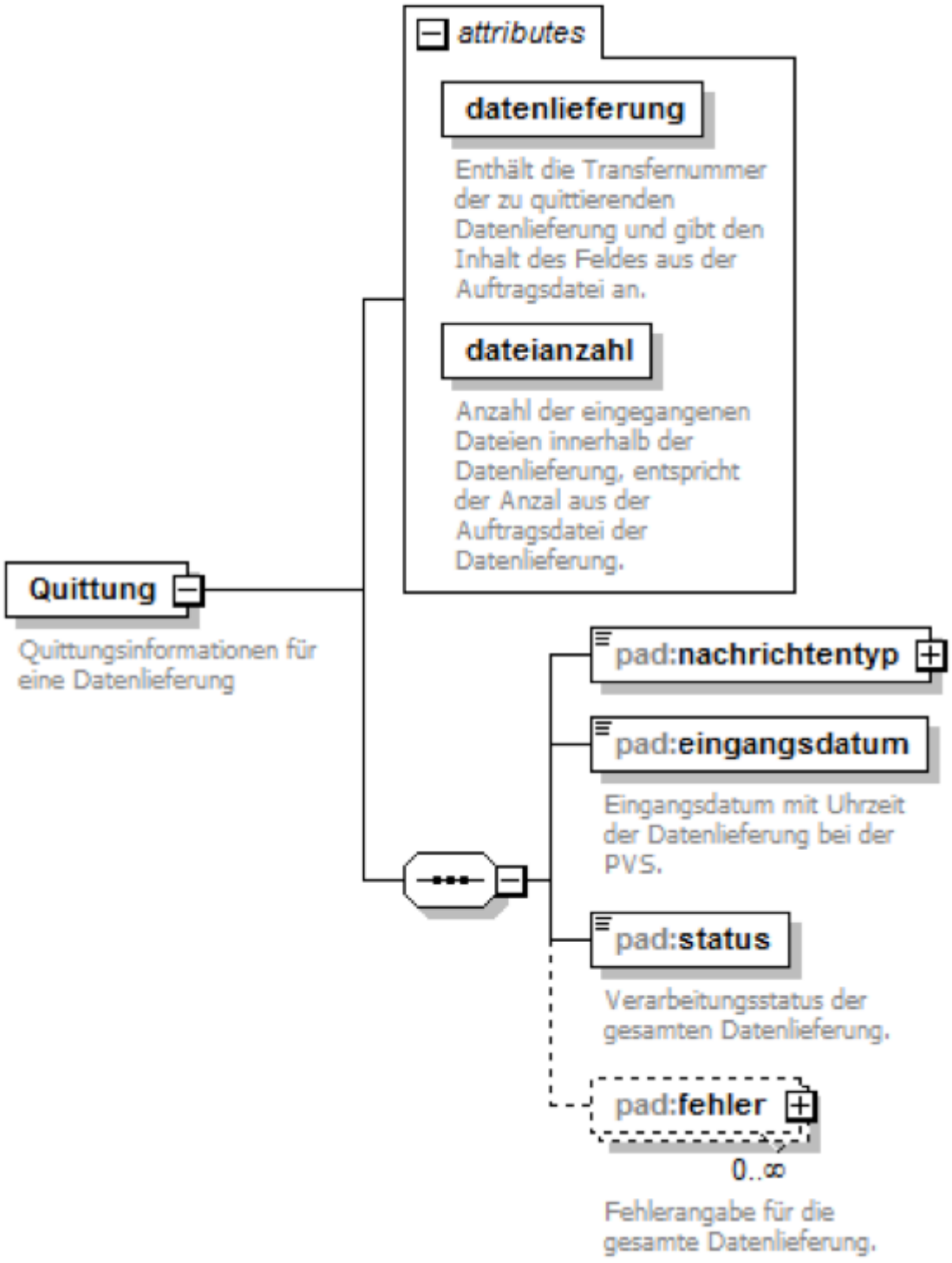
7 Quittung

Nach Eingang der Echtabrechnung in der Abrechnungsstelle bzw. der Servicestelle, wird die Abrechnung für den produktiven Verarbeitungsprozess zur Verfügung gestellt.

Der Arzt erhält als Rückmeldung pro Datenlieferung eine Quittungsnachricht, die „nur“ als normale KV-Connect-Nachricht verschlüsselt ist. Die einer indirekten Lieferung entsprechende Zusatzverschlüsselung entfällt also in jedem Fall. Die Quittungsnachricht enthält keine rechnungsspezifischen Informationen und erfolgt demnach nicht auf Rechnungsebene. Die Quittungsnachricht wird als ZIP-Datei übertragen und besteht aus der Auftragsdatei und der Quittungsdatei selbst.

7.1 Inhalte der Quittung

Äquivalent zur Nachricht selbst sind die Inhalte der Quittung streng definiert.



Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
@datenlieferung	1	posInt(6)	Enthält die Transfernummer der zu quittierenden Datenlieferung und gibt den Inhalt des Feldes aus der Auftragsdatei an (aus der Datenlieferung vom Kunden an die Abrechnungsstelle)	
@dateianzahl	1	posInt(3)	Anzahl der eingegangenen Dateien innerhalb der Datenlieferung (nur Nutzdaten, also ohne Auftragsdatei)	
nachrichtentyp	1	*string(4)*	Angabe des Nachrichtentyps	5.3

Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
@version	1	*string(5)*	Für dieses Klassenmodell ist der folgende fest vorgegeben: **QADL**: Quittung Arzt – Datenlieferung Angabe der Versionsnummer des NachrichtentAyps	
eingangsdatum	1	dateTime	Eingangsdatum mit Uhrzeit der Datenlieferung bei der Abrechnungsstelle	
status	1	posInt	Verarbeitungsstatus der gesamten Datenlieferung Die folgenden Werte sind bisher definiert: **0**: Die gesamte Datenlieferung konnte verarbeitet werden **1**: Technische Fehler aufgetreten **2**: Auftragsdatei enthält Fehler	
fehler	0..*	complex	Fehlerangabe für die gesamte Datenlieferung	7.1

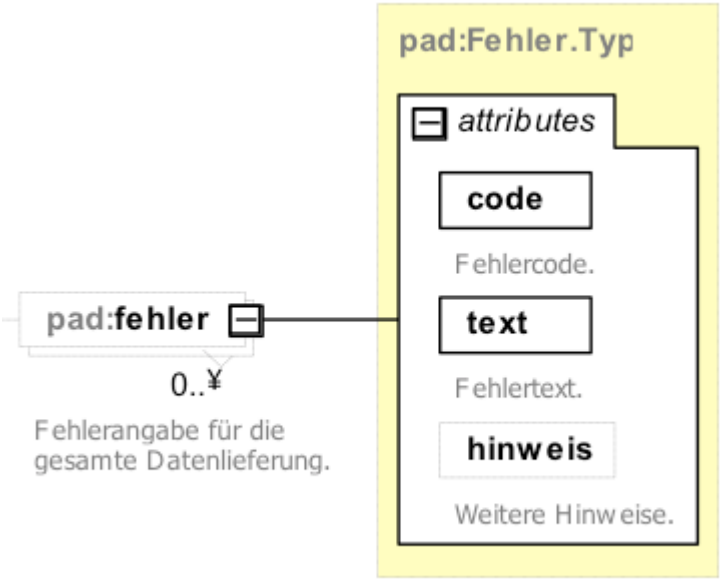
7.2 Namenskonvention

Die Dateinamen der Quittungsdateien entsprechen dem gleichen Schema wie die beim Nutzdatenformat:

Quittung: Nachrichtentyp QADL	
Aufbau	(Zusatz-)Verschlüsselte Datei: <KdNr.>_<Erstelldatum>_<Nachrichtentyp>_<Transfernr.>**_kvc.zip**
Format	**Inhalt der o.a. Zip-Datei (Quittung):** nnnnnnnn_JJJJMMTT_XXX_mmmmmm_auf.xml : Auftragsdatei nnnnnnnn_JJJJMMTT_XXX_mmmmmm_padx.xml : Quittung selbst
Beispiel	**Finale ZIP-Datei:** 1.1.1.1.1.1 00004711_20150517_QADL_000042_kvc.zip **Inhalt der o.a. Zip-Datei (Quittung):** 00004711_20150517_QADL_000042_auf.xml 00004711_20150517_QADL_000042_padx.xml

7.3 Fehler

In der optionalen Struktur fehler werden nähere Angaben zum Fehlerauftreten bei der Abrechnungsstelle gegeben. Das Element kann beliebig oft innerhalb einer Quittungsnachricht vorkommen.



Element / Attribut	Kardi-nalität	Datentyp	Erläuterung	Kap.
code	1	posInt	Fehlercode	
text	1	*string*	Fehlertext	
hinweis	0..1	*string*	Weitere Hinweise zum aufgetreten Fehler z.B. Angabe des Dateinamens in dem der Fehler aufgetreten ist	

Fehlerliste	
Technische Fehler (100-199)	
Code	**Beschreibung**
101	Dateilücke erkannt (Transfernummer nicht korrekt)
102	Daten wurden bereits verarbeitet (Dublette)
103	Daten konnten nicht dekomprimiert werden
104	Dateinamen entsprechen nicht den Namenskonventionen
Fehler in Auftragsdatei (200-299)	

Fehlerliste	
200	Auftragsdatei validiert nicht gegen das Schema
201	Falscher Empfänger (Eintrag in Auftragsdatei)

```
<!-- Beispiel für Validierungsfehler -->

<fehler code="200" text="Auftragsdatei validiert nicht gegen das Schema."/>
```

7.4 Quittungsbeispiel

Im folgenden Beispiel wird eine Quittungsnachricht dargestellt, die von der PVS Niedersachsen (Lüneburg) an ihren Kunden Dr. Müller mit der Kundennummer 4711 gesendet werden soll. Die Nachricht wird über die PADline GmbH als Vermittlungsstelle übertragen. Der tatsächliche Übertragungsweg zum Kunden spielt hierfür keine Rolle.

Es handelt sich um die erste Quittungsnachricht, die an Dr. Müller gesendet wird (Transfernummer=1), es wird nur eine Datenlieferung quittiert und zwar die mit der Transfernummer 42 (Dr. Müller war schon vor der Einführung der Quittungsfunktion Kunde bei der PVS und hat Daten elektronisch übertragen). Die Datenlieferung Nummer 42 wurde von Dr. Müller am 17.05.2011 an die Abrechnungsstelle versendet. Beim Empfänger der KV-Connect-Nachricht sind die Daten am 17.05.2011 um 9:33:47 Uhr eingegangen und konnten fehlerfrei verarbeitet werden.

```
<!-- Auftragsdatei für u.a. Quittungsnachricht -->

<Auftrag transfernr="1" echtdaten="true" erstellungsdatum="2011-05-17T09:36:1
         dateianzahl="1" xsi:schemaLocation="http://padinfo.de/ns/kvc/kvc_auf
         xmlns="http://padinfo.de/ns/kvc"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

       <empfaenger>

              <logisch kundennr="4711">Dr. Müller</logisch>

              <physikalisch kundennr="4711">Dr. Müller</physikalisch>

       </empfaenger>

       <absender>

              <logisch iknr="220330010" rzid="205">PVS Niedersachsen/Lüneburg

              <physikalisch rzid="300">PADline GmbH

       </physikalisch>

       </absender>

       <nachrichtentyp version="01.0">QADL</nachrichtentyp>

       <system>

              <produkt>KV-Connect-Service</produkt>
```

```
            <version>1.00</version>
            <hersteller>PADline GmbH</hersteller>
      </system>
      <verschluesselung verfahren="0"/>
      <empfangsquittung>0</empfangsquittung>
      <datei erstellungsdatum="2011-05-17T09:35:14Z" id="2011-4711-0001">
            <dokumententyp>PADneXt</dokumententyp>
            <name>00004711_20110517_QADL_000001_padx.xml</name>
            <beschreibung>Quittung für Datenlieferung über PADneXt.</beschr
            <dateilaenge pruefsumme="5158bb69b99cfae7bde269e7465665794c6322
      </datei>
</Auftrag>
```

```
<!-- Quittungsnachricht -->
<Quittung datenlieferung="42"
          dateianzahl="1" xsi:schemaLocation="http://padinfo.de/ns/pad kvc_qa
          xmlns="http://padinfo.de/ns/kvc"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <nachrichtentyp version="01.0">QADL</nachrichtentyp>
      <eingangsdatum>2011-05-17T09:33:47Z</eingangsdatum>
      <status>0</status>
</Quittung>
```

8 Aufbau der KV-Connect Nachrichten

Zur Erleichterung der Verarbeitung von KV-Connect Nachrichten werden diese mit Anwendungs- und Nachrichten-spezifischen X-Attributen und Content-Descriptions angereichert, die die Nachrichten als Ganzes aber auch deren einzelne Bestandteile kennzeichnen. Die eingesetzten Attribute entstammen einem Pool von Attributen, die zentral für alle KV-Connect-Anwendungen dokumentiert und gepflegt werden.

In der hier beschriebenen Anwendung kommen die folgenden X-Attribute zur Anwendung:

Header-Attribute	
X-KVC-Sendersystem: <Systembezeichnung>;<Version>	Name und Version des versendenden Primär-Systems
X-KVC-Dienstkennung: ePVS; Lieferung;V1.0	Nachrichten-Klassifizierung: ePVS-Lieferung
X-KVC-Dienstkennung: ePVS; Rueckmeldung;V1.0	Nachrichten-Klassifizierung: Rückmeldung (fachliche Quittung) zu einer ePVS-Einsendung
Segment-Attribute	
Content-Description: ePVS-Archiv	MIME-Segment der Einsendungs-Nachricht, das das Dokumentationsarchiv enthält
Content-Description: ePVS-Rueckmeldung	MIME-Segment der Rueckmeldungs-Nachricht, das die Datei mit der Antwortdaten enthält

Im Aufgabenbereich dieser Spezifikation, also bei der Abwicklung der Privatliquidation nach diesem Modell (ePVS) sind zwei Nachrichtenwege zu betrachten.

8.1 Übertragung zur Abrechnungsstelle

Bei der Übertragung an die Abrechnungsstelle kann, wie oben beschrieben, die direkte und indirekte Übertragung erforderlich sein. Die Auswahl, welches der beiden Verfahren gewählt wird ist durch das Primärsystem zu realisieren.

[PVSSN005] : **Das System des Einsenders MUSS in der Lage sein, bei ausgehenden Nachrichten zu entscheiden (oder vom Nutzer die Entscheidung abzufragen), ob eine Übertragung direkt oder indirekt erfolgen soll.**

Nach den in den vorhergehenden Kapiteln beschriebenen Regeln ist pro Übertragung/Nachricht eine Auftragsdatei und die Nutzdatei zu erzeugen. Die Nutzdatei wird mit dem öffentlichen Schlüssel des endgültigen (logischen) Adressaten (siehe Datei(zusatz)verschlüsselung und Auftragsdatei) verschlüsselt. Der endgültige (logische) Adressat der Nachricht ist im Fall der direkten Übertragung identisch mit dem Empfänger der KV-Connect-Nachricht, im indirekten Fall sind KV-Connect-Adressat und endgültiger Adressat disjunkt. Es ist zu beachten, dass die beiden Schlüssel/Zertifikate einer Identität im KV-Connect-System nicht identisch sind mit denen des Systems der Privat-Abrechnungsstellen.

[PVSSN010] : Das System des Einsenders MUSS in der Lage sein, den öffentlichen Schlüssel des endgültigen Adressaten (abhängig davon, ob eine direkte oder indirekte Übertragung vorgesehen ist) vom Schlüsselservice (Datei(zusatz)verschlüsselung Unterpunkt 2) abzuholen und die Nutzdaten mit diesem Schlüssel zu verschlüsseln.

Für die Spezifikation der KV-Connect-Übertragung wird davon ausgegangen, dass die beiden erforderlichen Dateien, die Auftragsdatei und das verschlüsselte Nutzdatenarchiv, bereits vorliegen.

Diese beiden Dateien werden in ein ZIP-Archiv verpackt. ZIP-Archive sind im Allgemeinen binäre Dateien und werden vor bzw. im Lauf des Versands mit „Email-Mechanismen“ bei der Erzeugung von MIME-Formaten BASE64-codiert. Die Erstellung der ZIP-Dateien soll äquivalent zu anderen KV-/KBV-Anwendungen bei allen Nutzungen des ZIP-Formats in dieser Spezifikation auf das Format „InfoZIP“ (http://www.info-zip.org) zurückgreifen (siehe z.B. [KBV_ITA_VGEX]).

[PVSSM015] : Das System des Einsenders MUSS in der Lage sein, aus den oben beschriebenen Dateien eine ZIP-Datei nach den Vorgaben von Kap. 3 bis 7 für den Versand und im Namensschema „nnnnnnnn_JJJJMMTT_XXX_mmmmmm_kvc.zip“ bzw. „nnnnnnnn_JJJJMMTT_XXX_mmmmmm_padx.zip“ zu erzeugen.

Der fertige innere MIME-BLOB der Nachricht hat mit den entsprechenden Boundaries sowie den Metadaten (insbesondere der „Content-Description: ePVS-Archiv) etwa folgendes Aussehen:

[PVSSM020] : Der MIME-Segment-Header des ePVS-Nachrichteninhalts MUSS das Attribut "Content-Description: ePVS-Archiv" enthalten.

```
--------------040506000303090304090700
Content-Type: application/x-zip-compressed;
 name="00123456_20150226_ADL_000042_kvc.zip"
Content-Transfer-Encoding: base64
Content-Description: ePVS-Archiv
Content-Disposition: attachment;
 filename="00123456_20150226_ADL_000042_kvc.zip"

UEsDBBQACAAIAMR4nEYAAAAAAAAAAAAAAAAkABAAMDAxMjM0NTZfMjAxNTAyMjZfQURMXzAw
MDA0Ml9hdWYueG1sVVgMAMKFP1XAhT9V9QEUAJVV227bOBB9Tr6C1bskirpZhCzAiNvuYpOi
2CyKoi8GJVEWYYnyklKc+Nvylh/rUJILe9Nmdw3YEKmZM2fOXJy+s220Gqpesa0uWc8Fql6e
FVrDo2wEr7ga5Ba1okd/8qKWcOASPXQtWqljj46DQp+/3CPbzq7TGQbBj9TgKNXSCoiFwK83
...
...
...
VdqHP1VQSwECFQMUAAgACADEeJxGb8+AFWcAAACrAAAALwAMAAAAAAAAAAAABApIHbAwAAX19N
QUNPU1gvLl8wMDEyMzQ1Nl8yMDE1MDIyNl9BRExfMDAwMDQyX2F1Zi54bWxVWAgAwoU/VcCF
P1VQSwECFQMUAAgACAAVeZxGRPe9sCOQAQAAkAEALAAMAAAAAAAAAABApIGvBAAAMDAxMjM0
NTZfMjAxNTAyMjZfQURMXzAwMDA0Ml9kYXRfa3ZjLnppcC5wN21VWAgAlIY/VVqGP1VQSwUG
```

```
AAAAAAQABABwAQAAPJUBAAAA

--------------040506000303090304090700--
```

Damit daraus eine gültige KV-Connect Nachricht wird ist der Inhalt (einschließlich des leeren Bodys) mit der Identität des KV-Connect-Absenders zu signieren. Für die Signatur ist als Hash-Algorithmus mindestens SHA 128 (derzeit, siehe immer auch Kryptographische Standards der REST-Schnittstelle (Version 2016)) zu verwenden. Die resultierende S/MIME-signed Message ist dann verschlüsselt an die KV-Connect-Identität des logischen Empfängers zu versenden. Es ist empfehlenswert, die Nachricht nicht nur an/für den Empfänger zu verschlüsseln, sondern auch an den Absender. Nach der Signatur hat die Nachricht folgende Form:

```
Content-Type: multipart/signed; protocol="application/pkcs7-signature";
micalg=sha128; (Kommentar: Änderungen berücksichtigen (siehe oben))
boundary="------------ms090104040704010109000702"

This is a cryptographically signed message in MIME format.

--------------ms090104040704010109000702
Content-Type: multipart/mixed;
boundary="------------040506000303090304090700"

This is a multi-part message in MIME format.
--------------040506000303090304090700
Content-Type: text/plain; charset=iso-8859-15
Content-Transfer-Encoding: quoted-printable

--------------040506000303090304090700
Content-Type: application/x-zip-compressed;
name="00123456_20150226_ADL_000042_kvc.zip"
Content-Transfer-Encoding: base64
Content-Description: ePVS-Archiv
Content-Disposition: attachment;
filename="00123456_20150226_ADL_000042_kvc.zip"

UEsDBBQACAAIAMR4nEYAAAAAAAAAAAAAAAAkABAAMDAxMjM0NTZfMjAxNTAyMjZfQURMXzAw

MDA0Ml9hdWYueG1sVVgMAMKFP1XAhT9V9QEUAJVV227bOBB9Tr6C1bskirpZhCzAiNvuYpOi

2CyKoi8GJVEWYYnyklKc+Nvylh/rUJILe9Nmdw3YEKmZM2fOXJy+s220Gqpesa0uWc8Fql6e

FVrDo2wEr7ga5Ba1okd/8qKWcOASPXQtWqljj46DQp+/3CPbzq7TGQbBj9TgKNXSCoiFwK83

...

...

...

VdqHP1VQSwECFQMUAAgACADEeJxGb8+AFWcAAACrAAAALwAMAAAAAAAAAABApIHbAwAAX19

NQUNPU1gvLl8wMDEyMzQ1Nl8yMDE1MDIyNl9BRExfMDAwMDQyX2F1Zi54bWxVWAgAwoU/Vc

CFP1VQSwECFQMUAAgACAAVeZxGRPe9sCOQAQAAkAEALAAMAAAAAAAAAABApIGvBAAAMDAxM

jM0NTZfMjAxNTAyMjZfQURMXzAwMDA0Ml9kYXRfa3ZjLnppcC5wN21VWAgAlIY/VVqGP1VQ

SwUGAAAAAAQABABwAQAAPJUBAAAA
```

```
----------040506000303090304090700--
----------ms090104040704010109000702
Content-Type: application/pkcs7-signature;
    name="smime.p7s"
Content-Transfer-Encoding: base64
Content-Disposition: attachment;
    filename="smime.p7s"
Content-Description: S/MIME Cryptographic Signature
MIAGCSqGSIb3DQEHAqCAMIACAQExCzAJBgUrDgMCGgUAMIAGCSqGSIb3DQEHAQAAoIIUaDCC
BC4wggMWoAMCAQICAgEMMA0GCSqGSIb3DQEBBQUAMHExCzAJBgNVBAYTAkRFMRwwGgYDVQQK
ExNEZXV0c2NoZSBUZWxla29tIEFHMR8wHQYDVQQLExZULVRlbGVTZWMgVHJ1c3QgQ2VudGVy
...
...
...
egVyYsDlSioQQ0IWzN+dVdI0iYOL8v3ByH/cBWA9xsHd4OFPkeLurHepbaAkpvnFG+3cyW6Y
BbPP892m/rWg8SpvU9YdUxXkY9vtOQVhbWJ4zxagLoE4yJCMBZs4skCF60GPdB3HrVXXZDZY
iRVNHFXavxeGtMxrFzVDAAAAAAAA
----------ms090104040704010109000702--
```

Die blau markierten Bereiche sind die Boundaries der elektronischen Signatur, die grünen Bereiche sind die Boundaries der signierten MIME-Nachricht (eine Mail mit leerem Mailbody mit einem Attachment). Die grauen Bereiche dienen der Übersicht und können auch weggelassen werden. (Farbgebung korrigieren)

Der entstandene MIME-BLOB wird in eine verschlüsselte S/MIME-Nachricht überführt. Diese resultierende (verschlüsselte) S/MIME-Nachricht, der Body der eigentlichen KV-Connect-Nachricht, wird mit den für den Transport wesentlichen Metadaten versehen. Zunächst ist die zu übermittelnde Datei in eine MIME-Datei entsprechend RFC 2045 bzw. 2046 einzubetten.

[PVSSM025] : Der Nachrichten-Header MUSS die "X-KVC-Dienstkennung: ePVS;Lieferung;V1.0 " enthalten.

[PVSSM030] : Der Nachrichten-Header MUSS ein Attribut "X-KVC-Sendersystem:" entsprechend [KVC-Anb] enthalten.

[PVSSM035] : Das Subject der Einsendung MUSS identisch mit der Dienstkennung sein ("Subject: ePVS;Lieferung;V1.0 ").

Den makroskopischen Aufbau der MIME-Datei zeigt beispielhaft die folgende Box.

Das Attribut "X-KVC-Dienstkennung" ist mit dem Wert "ePVS;Lieferung;V1.0" zu belegen,
das Attribut "X-KVC-Sendersystem" entsprechend der Vorgaben von [KVC-Anb].

Das Subject erhält den festen Eintrag "ePVS;Lieferung;V1.0".

```
Date: Tue, 28 Apr 2015 16:04:14 +0200

From: josef.jost.kvwl@kv-safenet.de

Reply-To: josef.jost.kvwl@kv-safenet.de

MIME-Version: 1.0

To: padline.kvtg@kv-safenet.de

Message-ID: <553F935E.8030907@kv-safenet.de>

X-KVC-Dienstkennung: ePVS;Lieferung;V1.0

X-KVC-Sendersystem: MySystem;V2.01

Subject: ePVS;Lieferung;V1.0

Content-Type: application/pkcs7-mime; name="smime.p7m"; smime-type=enveloped-

Content-Transfer-Encoding: base64

Content-Disposition: attachment; filename="smime.p7m"

Content-Description: Mit S/MIME verschluesselte Nachricht


MIAGCSqGSIb3DQEHA6CAMIACAQAxggGRMIIBjQIBADB1MGcxCzAJBgNVBAYTAkRFMRMwEQYD

VQQKEwpGcmF1bmhvZmVyMSEwHwYDVQQLExhGcmF1bmhvZmVyIENvcnBvcmF0ZSBQS0kxIDAe

BgNVBAMTF0ZyYXVuaG9mZXIgVXNlciBDQSAyMDA3AgogQ/EjAAAAAMrfMA0GCSqGSIb3DQEB

AQUABIIBAAWqKsqYwNqm8HEan3DzgzW2mzGXyMx6HjKxDkAjWIH09KIfK0KDOKL8KXkdiJdR

...

...

...

/YFcJMrXzGV2P9iHl1HRReOwtwaR1v+tgF3RjS2K2vWWQzMs1HuDHX6N0l1hADvyw3dUUMHH

iaWYEhWW5gQKW33tTjxD3GismqiRpgRGFeKxlNdC5v/caWmfCf6zOLst094YXvtJy2t1k3Ow

1IYP1Wg5EzpwOkeBjtzimZ57JcVO19vFZCkmg2eJI+JPjc64Ni3l9Tww4phSSGgqkVbLI+FY

GMDH17yscX3WGl1OAC8JjzfrBAj1Mhz1UHAfFQAAAAAAAAAAAA=
```

Der binäre Inhalt (die verschlüsselte Datei) ist BASE64-codiert. Die so entstandene Datei wird durch das Versandsystem des Primärsystems (entweder über den KV-Connect-Client oder eine direkte Verbindung zur REST-Schnittstelle des KV-Connect-Servers) an den KV-Connect-Server übertragen.

[PVSEM040] : Die Entschlüsselung der KV-Connect-Nachricht „ePVS Lieferung“ MUSS fehlerfrei möglich sein.

[PVSSM045] : Die Prüfung der Signatur der KV-Connect-Nachricht „ePVS Lieferung“ MUSS fehlerfrei möglich sein.

8.1.1 Empfangsbestätigung

Eine automatische Quittierung per MDN erfolgt nicht. Da in jedem Fall eine fachliche Quittung /Rueckmeldung vorgesehen ist, soll und muss eine automatische Quittung in Form einer MDN unterbleiben.

8.2 Übertragung der Rückmeldung (fachliche Quittung) an den einliefernden Arzt

Zu jeder Übertragung einer Abrechnungsdatei wird von der jeweils zuständigen Annahmestelle eine Auftragsdatei und eine fachliche Empfangsbestätigung nach den Vorgaben des Kap. Quittung erzeugt. Die beiden Dateien werden in einem ZIP-Archiv zusammengefasst.

Die Basis für die Übertragung der binären gezippten Rueckmeldung ist die Formatierung als MIME-BASE64-Containers in der folgenden Form:

[PVSSM050] : Der MIME-Segment-Header der Quittungsdatei MUSS ein Attribut "Content-Description: ePVS-Rueckmeldung" enthalten.

```
--------------090002030909070003090806
Content-Type: application/x-zip-compressed; name="QADL.zip"
Content-Transfer-Encoding: base64
Content-Description: ePVS-Rueckmeldung
Content-Disposition: attachment; filename="QADL.zip"

UEsDBBQACAAIAIqJnEYAAAAAAAAAAAAAAAAmABAAMDAwMDQ3MTFfMjAxMTA1MTdfUUFETF8w
MDAwMDFfcGFkeC54bWxVWAwAeqM/VVOjP1X1ARQAdZAxT8MwEIXn+lcY746dpKgiciwhMYah
ggGxVJbtJhbJJcROUvj1mEArdeh27/S9e3cn7ijF+8mFMEHtQelmdLoJmFKJxLmPjQoWWmeP
...
...
...
gWgCAABRQURMLUF1ZnRyYWcueG1sVVgIAJylP1WRpT9VUEsBAhUDFAAIAAgAu4qcRm/PgBVn
AAAAqwAAABsADAAAAAAAAAAAAQKSBNQUAAF9fTUFDT1NYLy5fUUFETC1BdWZ0cmFnLnhtbFVY
CACcpT9VkaU/VVBLBQYAAAAABQAFAK0BAAD1BQAAAAA=
--------------090002030909070003090806--
```

Damit aus diesem MIME-Segment eine gültige KV-Connect Nachricht wird, ist der Inhalt (einschließlich des leeren Bodys) mit der Identität des KV-Connect-Absenders zu signieren. Für die Signatur ist als Hash-Algorithmus SHA128 (derzeit, siehe auch Kryptographische Standards der REST-Schnittstelle (Version 2016)) zu verwenden. Die resultierende S/MIME signed Message ist dann verschlüsselt an die

KV-Connect-Identität des Empfängers, also des Absenders der Liquidation, zu versenden. Es ist empfehlenswert, die Nachricht nicht nur an/für den Empfänger zu verschlüsseln, sondern auch an den Absender. Nach der Signatur hat die Nachricht folgende Form:

```
Content-Type: multipart/signed; protocol="application/pkcs7-signature";
                        micalg=sha128; boundary="------------ms02040803090005

This is a cryptographically signed message in MIME format.

--------------ms020408030900050202070307
Content-Type: multipart/mixed; boundary="------------090002030909070003090806

This is a multi-part message in MIME format.
--------------090002030909070003090806
Content-Type: text/plain; charset=iso-8859-15
Content-Transfer-Encoding: quoted-printable

--------------090002030909070003090806
Content-Type: application/x-zip-compressed; name="QADL.zip"
Content-Transfer-Encoding: base64
Content-Description: ePVS-Rueckmeldung
Content-Disposition: attachment; filename="QADL.zip"

UEsDBBQACAAIAIqJnEYAAAAAAAAAAAAAAAAmABAAMDAwMDQ3MTFfMjAxMTA1MTdfUUFETF8w
MDAwMDFfcGFkeC54bWxVWAwAeqM/VVOjP1X1ARQAdZAxT8MwEIXn+lcY746dpKgiciwhMYah
ggGxVJbtJhbJJcROUvj1mEArdeh27/S9e3cn7ijF+8mFMEHtQelmdLoJmFKJxLmPjQoWWmeP
...
...
...
gWgCAABRQURMLUF1ZnRyYWcueG1sVVgIAJylP1WRpT9VUEsBAhUDFAAIAAgAu4qcRm/PgBVn
AAAAqwAAABsADAAAAAAAAAAAQKSBNQUAAF9fTUFDT1NYLy5fUUFETC1BdWZ0cmFnLnhtbFVY
CACcpT9VkaU/VVBLBQYAAAAABQAFAK0BAAD1BQAAAAA=
```

```
--------------090002030909070003090806--


--------------ms020408030900050202070307
Content-Type: application/pkcs7-signature; name="smime.p7s"
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="smime.p7s"
Content-Description: S/MIME Cryptographic Signature

MIAGCSqGSIb3DQEHAqCAMIACAQExCzAJBgUrDgMCGgUAMIAGCSqGSIb3DQEHAQAAoIIUaDCC
BC4wggMWoAMCAQICAgEMMA0GCSqGSIb3DQEBBQUAMHExCzAJBgNVBAYTAkRFMRwwGgYDVQQK
...
...
...
boq449AtkImG4a/nIwdCMkvSRTaiZh8F6PkGzPqgoRe9PuvKeNz37bjdcZkHmHrAQHpss1cd
SM8jbfIItB3R1D2ON0Bpp1DVW7cjQBDftiPw0sW/e8YvCoxv0TebuNwHILOQZuuTErjkqmXl
A1ryomsBCO26v6IumszxAAAAAAAA
--------------ms020408030900050202070307--
```

Man erkennt die Boundaries der elektronischen Signatur, im Inneren sind die Boundaries der signierten MIME-Nachricht (eine Mail mit leerem Mailbody mit einem Attachment) zu sehen.

[PVSSM055] : Die Rueckmeldung erzeugende Anwendung MUSS in der Lage sein, für die Quittung ein ZIP-Archiv nach den Namenskonventionen dieser Spezifikation zu erzeugen.

Der entstandene MIME-BLOB wird in eine verschlüsselte S/MIME-Nachricht überführt. Diese resultierende (verschlüsselte) S/MIME-Nachricht, der Body der eigentlichen KV-Connect-Nachricht, wird mit den für den Transport wesentlichen Metadaten versehen. Zunächst ist die zu übermittelnde Datei in eine MIME-Datei entsprechend RFC 2045 bzw. RFC 2046 einzubetten.

[PVSSM060] : Der Nachrichten-Header MUSS die "X-KVC-Dienstkennung: ePVS;Rueckmeldung;V1.0" enthalten.

[PVSSM065] : Der Nachrichten-Header MUSS ein Attribut "X-KVC-Sendersystem:" entsprechend [KVC-Anb] enthalten.

[PVSSM070] : Das Subject der Rueckmeldung MUSS identisch mit der Dienstkennung sein ("X-KVC-Dienstkennung: ePVS;Rueckmeldung;V1.0").

Jede ePVS-Nachricht mit der ein ePVS-Archiv übertragen wird ist durch eine eindeutige Nachrichten-ID charakterisiert. Um eine verbesserte Organisation der Quittungsverarbeitung zu ermöglichen (möglichst schon auf dem Server) wird die Nachrichten-ID der die Empfangsbestätigung auslösenden Nachricht in den Header der Empfangsbestätigung übernommen.

[PVSSM075] : Der Nachrichten-Header der Empfangsbestätigung MUSS das Attribut "in-reply-to:" mit der Nachrichten-ID der beantworteten ePVS-Nachricht enthalten.

Den makroskopischen Aufbau der MIME-Datei zeigt beispielhaft die folgende Box.

Das Attribut "X-KVC-Dienstkennung" ist mit dem Wert "ePVS;Rueckmeldung;V1.0" zu belegen, das Attribut "X-KVC-Sendersystem" entsprechend der Vorgaben von [KVC-Anb].

Das Subject erhält den festen Eintrag "ePVS;Rueckmeldung;V1.0".

```
Date: Wed, 29 Apr 2015 12:34:48 +0200

Message-ID: <5540B3C8.8060988@ibmt.fraunhofer.de>

From: padline.kvtg@kv-safenet.de

Reply-To: padline.kvtg@kv-safenet.de

MIME-Version: 1.0

To: josef.jost.kvwl@kv-safenet.de

X-KVC-Dienstkennung: ePVS;Rueckmeldung;V1.0

X-KVC-Sendersystem: AnotherSystem;V4.00

In-Reply-To: <553F935E.8030907@kv-safenet.de>

Subject: ePVS;Rueckmeldung;V1.0

Content-Type: application/pkcs7-mime; name="smime.p7m"; smime-type=enveloped-

Content-Transfer-Encoding: base64

Content-Disposition: attachment; filename="smime.p7m"

Content-Description: Mit S/MIME verschluesselte Nachricht


MIAGCSqGSIb3DQEHA6CAMIACAQAxggGRMIIBjQIBADB1MGcxCzAJBgNVBAYTAkRFMRMwEQYD

VQQKEwpGcmF1bmhvZmVyMSEwHwYDVQQLExhGcmF1bmhvZmVyIENvcnBvcmF0ZSBQS0kxIDAe

BgNVBAMTF0ZyYXVuaG9mZXIgVXNlciBDQSAyMDA3AgogQ/EjAAAAAMrfMA0GCSqGSIb3DQEB

...

...

...

bBXxob81FutgDgZSzt+G4vY6Yr7djDDkMTUwZPuQ9x4rBH5gJAYVKdgU/OiiPgOzQiykFX9p

DWq9K6PvgcsTWAkbbhF+Ae2VbrP/YDQNqV4pgO9V/+EmJh9J2Rof9hv94Tf+NbBrIUpPtK90
```

```
UfjhjbTbGpgTWgcN+Q/MwR9MZQ9ajqe00btC0qvIcQ4ECCp1D7dc3lrDAAAAAAAAAAAAAA==
```

Der binäre Inhalt (die verschlüsselte Datei) ist BASE64-codiert. Die so entstandene Datei wird durch das Versandsystem des Primärsystems (entweder über den KV-Connect-Client oder eine direkte Verbindung zur REST-Schnittstelle des KV-Connect-Servers) an den KV-Connect-Server übertragen.

[PVSEM075] : **Die Entschlüsselung der KV-Connect-Nachricht „ePVS Rueckmeldung“ MUSS fehlerfrei möglich sein.**

[PVSSM080] : Die Prüfung der Signatur der KV-Connect-Nachricht „ePVS Rueckmeldung“ MUSS fehlerfrei möglich sein.

8.3 Bearbeitung der Rueckmeldung (fachlichen Quittung)

Die Erzeugung und Übertragung der Rueckmeldung zu einer Privatabrechnung dient der Sicherung der Prozessabläufe gegen unbemerkte Fehler und gegebenenfalls der Fehlerkorrektur. Um im laufenden Praxisbetrieb die Kontrolle über die internen Abläufe zu erhalten, ist es erforderlich, dass die von der Abrechnungsstelle produzierten Empfangsbestätigungen auch ausgewertet werden. Nur wenn eine Empfangsbestätigung eine erfolgreiche Übersendung der Abrechnung signalisiert, kann diese vom lokalen System als abgeschlossen kategorisiert werden. Und nur wenn die Rueckmeldung, die einen Fehler signalisiert auch ausgewertet wird, kann das System automatisiert eine vernünftige Fehlerbehandlung anstoßen.

Aus diesem Grund **sollte** das Primärsystem, das den Dienst „ePVS mit KV-Connect“ unterstützt, regelmäßig, solange nicht alle ePVS-Vorgänge erfolgreich abgeschlossen sind, abfragen, ob eine Nachricht an das lokale System (den lokalen Nutzer) vorliegt. Da diese Vorgabe jedoch sehr tief in die internen Verarbeitungsstrukturen des Primärsystems eingreift, wird sie nicht als zwingend kategorisiert sondern als Empfehlung.

Allerdings muss das Primärsystem, das für die Anwendung „ePVS mit KV-Connect“ zertifiziert werden soll in der Lage sein, KV-Connect-Nachrichten mit der Dienstkennung „ePVS;Empfangsbestaetigung;V1.0“, die an das lokale System (also entweder die lokale Einrichtung oder an eines der Mitglieder der lokalen Betriebsstätte) adressiert sind, abzuholen, die Quittung und die Auftragsdatei aus der Nachricht zu extrahieren und Quittung und Auftragsdatei in einem definierten Verzeichnis abzulegen. Das System muss außerdem die Möglichkeit bieten, die Empfangsbestätigung bzw. deren sachlichen Inhalt in geeigneter Form darzustellen.

[PVSEM085] : Das Primärsystem des Abrechners MUSS in der Lage sein, KV-Connect-Nachrichten mit der Dienstkennung „ePVS;Rueckmeldung;V1.0“ abzuholen, die Empfangsbestätigung aus der Nachricht zu extrahieren und die Rueckmeldung in einem definierten Verzeichnis abzulegen.

[PVSEN090] : Das Primärsystem des Abrechners MUSS in der Lage sein, den sachlichen Inhalt der „ePVS-Rueckmeldung“ (Erfolg, Fehler, ...) für den Benutzer verständlich darzustellen.

9 Schlüsseltabellen

9.1 Anhangsformate

Wert	Erläuterung
pdf	Portable Document Format (Dokumentenformat von Adobe).
jpeg	Joint Photographic Experts Group (verlustbehaftete komprimierte Bilddateien).
tiff	Tagged Image File Format (universelles Pixelbild-Format).

9.2 Dokumententyp

Wert	Erläuterung
PADneXt	Kennzeichnet die Datei als PADneXt Datei (XML-basierend).
PAD	Kennzeichnet die Datei als PAD Datei (ursprüngliche Schnittstellenformat, nicht XML-basierend).
BDT	Kennzeichnet die Datei als BDT Datei (ist vorab mit der Abrechnungsstelle abzustimmen)
Anhang	Beliebiger Dateninhalt mit Zusatzinformationen zu einer Rechnung.

9.3 RZ ID

RZ ID	Name
100	PVS Bremen
101	Bezirksstelle Bremen
102	Bezirksstelle Bremerhaven
200	PVS Niedersachsen
201	Bezirksstelle Aurich
202	Bezirksstelle Braunschweig
203	Bezirksstelle Göttingen
204	Bezirksstelle Hannover
205	Bezirksstelle Lüneburg
206	Bezirksstelle Oldenburg
207	Bezirksstelle Osnabrück
208	Bezirksstelle Stade
209	Bezirksstelle Verden
210	Bezirksstelle Wilhelmshaven
300	PADline GmbH
400	PVS Westfalen-Nord
500	PVS Westfalen-Süd
600	PVS Sachsen
601	Bezirksstelle Chemnitz
602	Bezirksstelle Dresden
603	Bezirksstelle Leipzig

700	PVS Limburg-Lahn
800	PVS Mosel-Saar
801	Bezirksstelle Neunkrichen
802	Bezirksstelle Trier
900	PVS Südwest
901	Bezirksstelle Karlsruhe
902	Bezirksstelle Mannheim
1000	Aev München
1100	PVS Südbaden
2000	PVS Schleswig-Holstein / Hamburg
2001	Bezirksstelle Bad Doberan
2002	Bezirksstelle Bad Segeberg
2003	Bezirksstelle Hamburg
2100	PVS Rhein-Ruhr
2101	Bezirksstelle Aachen
2102	Bezirksstelle Berlin / Brandenburg
2103	Bezirksstelle Cottbus
2104	Bezirksstelle Düsseldorf
2105	Bezirksstelle Köln
2106	Bezirksstelle Moers
2107	Bezirksstelle Mülheim
2108	Bezirksstelle Potsdam
2109	Bezirksstelle Wuppertal
2200	PVS Büdingen

2201	Bezirksstelle Büdingen
2202	Bezirksstelle Giessen
2203	Bezirksstelle Kassel
2204	Bezirksstelle Mainz
2205	Bezirksstelle Nürnberg
2206	Bezirksstelle Weimar
2207	Bezirksstelle Würzburg
2208	Bezirksstelle München
2300	PVS Baden-Württemberg
3000	PVS Verbandsgeschäftsstelle Berlin
4000	ARC Abrechnungs-Centrum Dr. Pellengahr e.K.
4100	BFS health finance GmbH
4200	DZR – Deutsches zahnärztliches Rechenzentrum GmbH
4300	EOS Health Honorarmanagement AG
4400	ZA – Zahnärztliche Abrechnungsgesellschaft AG
4500	mediserv Abrechnung und Service für Heilberufe GmbH
8000	PAS Dr. Hammerl GmbH & Co. KG
9999	quadcore GmbH (für Testbetrieb definiert)

10 Referenzen

- [PP KVC]: Dokumentation zu KV-Connect im KV-Connect Partnerportal (https://partnerportal.kv-telematik.de)
- [KVC-Anb]: Anbindung an KV-Connect (Link 1: Anbindung an KV-Connect bzw. Link 2: Spezifikation Anbindung an KV-Connect)

11 Prüfregeln

Die im Rahmen der KV-Connect-Anwendung versandten Privatabrechnungen. Eingehende Abrechnungen **MÜSSEN** durch die Software der Annahmestelle (vor der Zuführung zu deren internem Prozess) einer mehrstufigen formalen Prüfung unterzogen werden. Ein Teil dieser Prüfungen im Produktivbetrieb können beim KV-Connect-Audit nicht nachvollzogen werden (die interne Struktur und die interne Verarbeitungslogik der PVS ist für die KVTG nicht transparent). Die im Rahmen des KV-Connect-Audits für den Transport vorgesehenen Prüfungen und Prüfregeln MÜSSEN allerdings in identischer Art und Weise durch den „Prüf-Client“ nachgebildet werden, gegen den Softwarehäuser ihre Systeme testen können und gegen den auch die Auditierung erfolgt.

"Technische Fehler" sind nicht Gegenstand dieser "Prüfregeln".

Die Prüfregeln für den Quittungsempfang sollen ein Mindestmaß von gesicherter Verarbeitung quer zu allen Primärsystemen für den betroffenen Anwender gewährleisten.

11.1 Regeln für das Senden von Einsendungen

11.1.1 Prüfregel [PVSSN005]

Das Primärsystem muss die Entscheidung ermöglichen, ob ein Versand an die Private Abrechnungsstelle nach dem direkten oder indirekten Verfahren erfolgen soll oder darf.

Diese Entscheidung kann durch interne Konfiguration/Konfektionieren des Primärsystems realisiert werden oder über Software-Auswahldialoge, die dem Nutzer eine sichere Wahl des Verfahrens ermöglicht.

11.1.2 Prüfregel [PVSSN010]

Die Infrastruktur der Privatabrechner besitzt eine eigene PKI, deren Credentials (derzeit) nicht identisch sind mit denen der Infrastruktur der KVTG. Beim indirekten Versand der Abrechnung ist der personenbezogenen Inhalt der Nachrichten mit einem öffentlichen Schlüssel / für einen privaten Schlüssel aus dieser parallelen Sicherheitsinfrastruktur zu verschlüsseln. Für die Zertifizierung als „ePVS-Anwendung“ muss das Primärsystem in der Lage sein, den/die erforderlichen Schlüssel / Zertifikate des physikalischen Adressaten zu identifizieren, über die Mechanismen aus „4.2 Schlüsselservice“ bereitzustellen und die zu übermittelnden Informationen für diesen Empfänger zu verschlüsseln.

Das Primärsystem SOLLTE in der Lage sein, zusätzlich an sich selbst zu verschlüsseln. Die Identifikation des Adressaten in diesem Sinn kann durch interne Konfiguration/Konfektionieren des Primärsystems realisiert werden oder über Software-Auswahldialoge, die dem Nutzer eine sichere Identifikation des Empfängers in der parallelen Infrastruktur ermöglicht.

11.1.3 Prüfregel [PVSSM015]

Die internen Prozesse der Privatabrechnung setzen für die Funktionsfähigkeit des Verfahrens Datenstrukturen und Dateinamenskonventionen voraus, die eingehalten werden müssen. Die einschlägigen Regeln sind in den Kapiteln 3 bis 7 beschrieben. Das Ergebnis dieser Maßnahmen ist in jedem Fall ein ZIP-Archiv mit einem Namen nach der Konvention „nnnnnnnn_JJJJMMTT_XXX_mmmmmm_kvc.zip“ bzw. „nnnnnnnn_JJJJMMTT_XXX_mmmmmm_padx.zip“ im Fall von PADneXt. Im Rahmen der Auditierung der KV-Connect-Anwendung „ePVS mit KV-Connect“ werden die Inhalte dieser Datei nicht mehr weiter geprüft. Das Prüfkriterium besteht darin, dass die Anwendung in der Lage sein MUSS, für jede ePVS-Nachricht ein ZIP-Archiv nach den Namenskonventionen dieser Spezifikation zu erzeugen.

11.1.4 Prüfregel [PVSSM020]

Der MIME-Segment-Header des ePVS-Nachrichteninhalts MUSS das Attribut "Content-Description: ePVS-Archiv" zur Charakterisierung des Inhaltes enthalten.

11.1.5 Prüfregel [PVSSM025]

Das Dienstmerkmal „X-KVC-Dienstkennung: ePVS; V1.0; Lieferung“ MUSS im Nachrichten-Header enthalten sein.

11.1.6 Prüfregel [PVSSM030]

Der Nachrichten-Header MUSS ein Attribut "X-KVC-Sendersystem:" entsprechend [KVC-Anb] enthalten.

11.1.7 Prüfregel [PVSSM035]

Das Subject der Einsendung MUSS ein Subject enthalten, dessen Inhalt identisch mit der Dienstkennung ist ("Subject: ePVS; V1.0; Lieferung ").

11.1.8 Prüfregel [PVSEM040]

Bei der Entschlüsselung der Nachricht darf kein Fehler auftraeten.

11.1.9 Prüfregel [PVSEM045]

Bei der Prüfung der Signatur der Nachricht darf kein Fehler auftreten.

11.2 Regeln für das Senden von Empfangsbestätigungen

11.2.1 Prüfregel [PVSSM050]

Der MIME-Segment-Header des Inhalts der Quittungsdatei MUSS das Attribut "Content-Description: ePVS-Rueckmeldung" zur Charakterisierung des Inhaltes enthalten.

11.2.2 Prüfregel [PVSSM055]

Die Anwendung MUSS in der Lage sein, für jede ePVS-Rueckmeldung ein ZIP-Archiv nach den Namenskonventionen dieser Spezifikation zu erzeugen.

11.2.3 Prüfregel [PVSSM060]

Das Dienstmerkmal „X-KVC-Dienstkennung: ePVS;Rueckmeldung;V1.0“ MUSS im Nachrichten-Header enthalten sein.

11.2.4 Prüfregel [PVSSM065]

Der Nachrichten-Header MUSS ein Attribut "X-KVC-Sendersystem:" entsprechend [KVC-Anb] enthalten.

11.2.5 Prüfregel [PVSSM070]

Das Subject der Quittung MUSS ein Subject enthalten, dessen Inhalt identisch mit der Dienstkennung ist ("Subject: ePVS;Rueckmeldung;V1.0").

11.2.6 Prüfregel [PVSSM075]

Der Nachrichten-Header der Empfangsbestätigung MUSS das Attribut "in-reply-to:" mit der Nachrichten-ID der beantworteten ePVS-Nachricht enthalten. Jede ePVS-Nachricht mit der ein ePVS-Archiv übertragen wird ist durch eine eindeutige Nachrichten-ID charakterisiert. Um eine verbesserte Organisation der Quittungsverarbeitung zu ermöglichen (möglichst schon auf dem Server) wird die Nachrichten-ID der die Empfangsbestätigung auslösenden Nachricht in den Header der Empfangsbestätigung übernommen.

11.2.7 Prüfregel [PVSEM080]

Bei der Entschlüsselung der Rueckmeldung darf kein Fehler auftraten.

11.2.8 Prüfregel [PVSEM085]

Bei der Prüfung der Signatur der Rueckmeldung darf kein Fehler auftreten.

11.3 Regeln für das Empfangen von Empfangsbestätigungen

Die Auswertung der Rueckmeldung ist i.A. für den absendenden Arzt ein wesentliches Werkzeug zur Organisation des gesamten Geschäftsprozesses

11.3.1 Prüfregel [PVSEM090]

Das Primärsystem des Abrechners MUSS in der Lage sein, KV-Connect-Nachrichten mit der Dienstkennung „ePVS;Rueckmeldung;V1.0“ abzuholen, die Empfangsbestätigung aus der Nachricht zu extrahieren und die Empfangsbestätigung in einem definierten Verzeichnis abzulegen

11.3.2 Prüfregel [PVSEN095]

Das Primärsystem des Abrechners MUSS in der Lage sein, den sachlichen Inhalt der „ePVS-Rueckmeldung“ (Erfolg, Fehler, ...) für den Benutzer verständlich darzustellen.