Zertifikatsprofile für X.509 Attributzertifikate V2.3.2

Bundesärztekammer, Berlin

	Datum	Name, Abteilung, Firma
Autor, Ansprechpartner		Georgios Raptis
Status (HPC-Projektbüro)	12.05.11	Freigegeben

Versionshistorie				
Version	**Datum**	**Bearbeiter**	**Änderungen**	**Bemerkungen**
0.1	21.09.05	Georgios Raptis (RS)		Initiale Version
0.4	24.10.05	Georgios Raptis (RS)		Dokument finalisiert, QS erforderlich
0.41	25.10.05	DL	-Ergänzung um Verweis auf mögliche admissionAuthority -QS	
0.5.0	08.10.08	Georgios Raptis	Neue OIDs, Literaturverzeichnis	
0.5.1	10.10.08	Dirk Schladweiler	QS	
2.3.1	10.03.09	Georgios Raptis	Konsolidierung zum Paket V2.3.1 Common-PKI 2.0 und neue RFCs	Flexibilisierung von Attributen und PolicyOIDs. Attribute, OIDs und URLs sind Konfigurationsdaten und können von der BÄK ohne Anpassung des Dokumentes geändert werden
2.3.1	23.03.09 29.05.09	Jessica Beyer	QS	

2.3.2	12.05.11	Dirk Schladweiler	Aktualisierung der Referenzen	

Inhalt

Tabellenverzeichnis

1 Zertifikatsprofile für X.509 Attributzertifikate, Übersicht und Inhalte

Zertifikatstyp	**Qualifiziertes Attributzertifikat zum qualifizierten Basis-Signaturzertifikat**
Aussteller	ZDA-qCA für Ärzte
acinfo	
Version	DEFAULT v1 (0)[1]
Subject	baseCertificateID (Issuer und SerialNumber des qualifizierten Basis-Signaturzertifikats)
Issuer	CN=<ZDA> qCA für Ärzte 1:PN, O=<ZDA>,C=DE gleicher Issuer wie das qualifizierte Basis-Signaturzertifikat[2]
Signature	OID des verwendeten Signaturalgorithmus
SerialNumber	Zertifikatsseriennummer
AttCertValidityPeriod	= validity des unter baseCertificateID referenzierten qualifizierten Basis-Signaturzertifikats
Attributes	admission
Extensions	siehe Tabelle 2
SignatureAlgorithm	
SignatureAlgorithm	OID des verwendeten Algorithmus (bzgl. Signatur des Zertifizierers)
SignatureValue	
SignatureValue	Bit-String (octetString)

Tabelle 1: Grundprofil der qualifizierten Attributszertifikate mit Grundstruktur und Referenzierung des Basis-Signaturzertifikates.

[1] Nach Common-PKI [externCommonPKI] und abweichend vom [externRFC3281] nur v1 (Begründung: s. Common-PKI). Bitte Version v1 nicht kodieren, sie ist laut ASN.1-Definition DEFAULT.

[2] Für die explizite Abweichung zum [externRFC3281] s. [externCommonPKI] Part 1 S. 44 Note 7.

Name des Feldes	**Qualifiziertes Attributzertifikat zum qualifizierten Basis-Signaturzertifikat**
Extensions	
AuthorityKeyIdentifier	Key Identifier vom öffentlichen Schlüssel des Ausstellers
CertificatePolicy	[1]Zertifikatsrichtlinie: Richtlinienkennung=id-commonpki-cp-accredited [1]Zertifikatsrichtlinie: Richtlinienkennung= s. Kap. 4.9.2 [1,1]Richtlinienqualifierinformationen: Richtlinienqualifier Id=CPS Qualifier: http://www.e-arztausweis.de/policies/EE_policy.html
CrlDistributionPoints (URL-konforme Kodierung erforderlich)	CRL-DP des Ausstellers
ValidityModel	id-validityModel-chain {1 3 6 1 4 1 8301 3 5 1}
optional: additionalInformation	ggf. Text gem. Policy
QCStatements	QcCompliance, KEIN EuLimitValue!
authorityInfoAccess	method=OCSP, URI des jeweiligen OCSP-Responders
Attributes	
Admission	gesetzt, s. u.

Tabelle 2: Zertifikatsprofil für Attributzertifikate – Extensions und Attribute

Die URLs und DNs in der Tabelle sind nicht normativ und können geändert werden.

2 Rahmenbedingungen für Attributzertifikate

Für die elektronischen Arztausweise können qualifizierte Attributzertifikate zu den qualifizierten Signaturzertifikaten ausgestellt werden. Die Struktur der qualifizierten Signaturzertifikate wird in [baekCerts] beschrieben. Aussteller eines qualifizierten Attributzertifikats soll der Aussteller des korrespondierenden qualifizierten Signaturzertifikats sein. Die Attributzertifikate beinhalten das Berufsgruppenattribut oder ein anderes von der Ärztekammer bestätigtes Attribut in einem „admission"-Attribut. Andere Attribute werden im Dokument [baekConfigData] bekanntgegeben. Attributzertifikate dürfen keine Einschränkungen beinhalten[3]. Attributzertifikate werden nur in Verbindung mit qualifizierten Signaturzertifikaten ausgestellt und haben die gleiche Gültigkeit. Es ist möglich, dass das zugehörige Public-Key Signaturzertifikat ebenfalls eine Admission-Extension enthält.

Die Bundesärztekammer kann neue Inhalte der Admission-Extension, wie z. B. professionOIDs, professionItems und ggf. registrationNumber für die Attributzertifikate bekanntgeben bzw. bestehende Inhalte ändern, die dann entsprechend von den ZDAs in die Attributzertifikate aufgenommen werden. Dafür ist keine Änderung dieses Dokumentes erforderlich, da diese Daten als Konfigurationsdaten definiert werden. Außerdem kann die Bundesärztekammer entscheiden, auf ein Attributzertifikat zu verzichten oder die Ausstellung eines Attributzertifikates (mit einem Berufsgruppen- oder Facharzt- oder anderes Attribut) im Ermessen der Ärztekammer und des Antragstellers zu stellen. Auch dafür ist keine Änderung dieses Dokumentes sondern lediglich eine entsprechende Mitteilung erforderlich. Konfigurationsdaten werden in [baekConfigData] definiert.

Das korrespondierende Basis-Signaturzertifikat wird mittels „baseCertificateID" referenziert. Die Aufnahme einer SubjectDN ist laut [externCommonPKI] Part 9 unzulässig.

Als gesetzliche Anforderung gilt, dass das Attributzertifikat bei Sperrung des korrespondierenden Basis-Signaturzertifikats gesperrt werden muss. Bei elektronischen Arztausweisen muss eine Sperrung des qualifizierten Attributzertifikats (z. B. bei Entzug der Approbation) auch die Sperrung des korrespondierenden Basis-Signaturzertifikats (sowie aller anderen Zertifikate auf dem eArztausweis) zur Folge haben, sofern das Attributzertifikat ausschließlich das Berufsgruppenattribut „Ärztin/Arzt" enthält.

[3] Die Aufnahme einer Einschränkung im Attributzertifikat hätte zur Folge, dass ein „LiabilityLimitationFlag" im Signaturzertifikat gesetzt werden müsste und das Attributzertifikat in jeder Signatur zwingend aufgenommen werden müsste.

3 Übersicht und Beschreibung von Zertifikatsstrukturen

Die folgenden Tabellen enthalten eine Übersicht über die einzelnen Zertifikatsstrukturen mit einer kurzen Beschreibung. Die Kritikalität einer Extension wird mit „!!“ aufgeführt; nicht gekennzeichnete Extensions sind nicht kritisch. Ein Client (Software), der eine „kritische“ Extension nicht versteht, muss das Zertifikat ablehnen.

Name der Struktur	**Semantik**	**Inhalt**
acinfo		
Version	X.509-Versionsnummer	0 (Version 1) DEFAULT
SerialNumber	Zertifikatsseriennummer. Eindeutig für alle von einem CA-Zertifikat ausgestellten Zertifikate	z. B. „1234567890“, muss als positiver Integer kodiert sein (MSB=0), max. 20 bytes lang
Signature	Kennzeichner (OID) für den Algorithmus für die Signatur des Zertifikates	z. B. sha256WithRsaEncryption
Issuer	Name des Ausstellers des Zertifikates (CA-Zertifikat)	Es soll der Issuer des Basis-zertifikats verwendet werden
AttCertValidityPeriod	Gültigkeitszeitraum des Attributzertifikates	Ende der Gültigkeit = Ende der Gültigkeit des Basiszertifikats
Subject	baseCertificateID	Referenz auf das Basiszertifikat als IssuerDN und serialNumber
Attributes	siehe Tabelle 4	admission
Extensions	siehe Tabelle 4	
SignatureAlgorithm		
SignatureAlgorithm	OID des Signatur-algorithmus, mit dem das Zertifikat signiert wurde	z. B. sha256WithRSAEncryption
SignatureValue		
SignatureValue	Signatur der Zertifizierungsstelle	Bit-String

Tabelle 3: Grundstruktur der Attributzertifikate

Name der Struktur	Semantik		Inhalt
Extensions			
AuthorityKeyIdentifier		Informationen zur Identifikation des öffentlichen Schlüssels des CA-Zertifikates	nur keyIdentifier = OCTET STRING, 20 Bytes, SHA-1 aus dem public-key ohne Tag- und Längen-bytes (s. [externCommonPKI])
CertificatePolicy		Policy der Zertifizierungsstelle für das Zertifikat. Identifikationsmerkmal für eArztausweis-Zertifikate	s. Tabelle 2
AdditionalInformation		Weitere Informationen nicht einschränkender Natur über das Zertifikat (nicht über den Zertifikatsinhaber)	optional. enthält ggf. Text, welches das Zertifikat als e-Arztausweis-Zertifikat ausweist.

Name der Struktur	Semantik		Inhalt
Extensions			
CrlDistributionPoints		Informationen, wie und wo die zugehörige Sperrliste (CRL) bezogen werden kann und wer der Aussteller dieser Liste ist	s. Tabelle 2
QcStatements		Informationen, dass das Zertifikat qualifiziert i. S. der EU-Direktive ist	Konformität zur EU-Direktive, Es dürfen KEINE Beschränkungen des Basiszertifikats aufgenommen werden. Evtl. Beschränkungen müssen ins Basiszertifikat aufgenommen werden
AuthorityInfoAccess		Quelle für Status-informationen für die Validierung des Zertifikats	URL des zuständigen OCSP-Responders
ValidityModel		Beschreibt das zugrunde liegende Gültigkeitsmodell für die Zertifikate und Signaturen	OID für das Kettenmodell id-validityModel-chain {1 3 6 1 4 1 8301 3 5 1}
Attributes			
Admission		Information über das Berufsgruppenattribut oder über ein anderes Attribut	enthält admissionAuthority (zuständige Landesärzte-kammer) und professionItem als OID (s. Kap. 4.8.1) und Text (Ärztin/Arzt) oder einen anderen OID und Text (Konfigurationsdaten, s. [baekConfigData])

Tabelle 4: Extensions und Attribute mit Wert

4 Beschreibung der Zertifikatsfelder und gültige Werte

4.1 Grundstruktur

Die Zertifikate sind konform zur Spezifikation X.509 v3 [externRFC3281] und zur [externCommonPKI] Spezifikation. Sie enthalten folgende ASN.1-Strukturen, deren spezifische Ausprägung von Zertifikatstyp abhängt (s. Übersicht):

```
AttributeCertificate  ::=  SEQUENCE
{
        acinfo                  AttributeCertificateInfo,
        signatureAlgorithm      AlgorithmIdentifier,
        signatureValue          BIT STRING
}
```

Der Kennzeichner (OID) des Algorithmus, mit dem das Zertifikat signiert worden ist, ist in der Struktur `signatureAlgorithm` enthalten. Das Feld `signatureValue` enthält die Signatur der Zertifizierungsstelle.

Das `acinfo` enthält alle Elemente des Zertifikats (ohne die Signatur der Zertifizierungsstelle), in der Form, wie sie von der Zertifizierungsstelle signiert werden. Insbesondere ist die Referenz zum Basiszertifikat sowie das (Berufsgruppen- o.ä.) Attribut enthalten. Die ASN.1-Struktur von `AttributeCertificateInfo` ist:

```
AttributeCertificateInfo   ::= SEQUENCE {
     version                   AttCertVersion DEFAULT v1,
     subject                   CHOICE {
          baseCertificateID    [0] EXPLICIT IssuerSerial,
          subjectName          [1] EXPLICIT GeneralNames}
     issuer                    GeneralNames,
     signature                 AlgorithmIdentifier,
     serialNumber              CertificateSerialNumber,
     attrCertValidityPeriod    AttCertValidityPeriod,
     attributes                SEQUENCE OF Attribute,
     issuerUniqueID            UniqueIdentifier OPTIONAL,
     extensions                Extensions OPTIONAL
}
```

Der `issuerUniqueID` darf nicht gesetzt werden. Für die hier beschriebenen Attributzertifikate wird im Feld `attributes` einzig das Attribut `admission` verwendet. Der

`AlgorithmIdentifier` für das Feld `signature` muss mit dem `AlgorithmIdentifier` vom Feld `signatureAlgorithm` der übergeordneten Struktur `AttributeCertificate` übereinstimmen.

4.2 Version, Feld `Version`

Das Feld `Version` definiert die Version des Attributzertifikats und hat den Wert 0 (DEFAULT) für v1. Da der Wert 0 DEFAULT ist, darf es nach DER nicht kodiert werden.

4.3 Seriennummer des Zertifikats, Feld `SerialNumber`

Das Feld `SerialNumber` enthält die Seriennummer des Zertifikats, kodiert als signed Integer (also MSB=0) und darf eine Maximallänge von 20 Octets haben [externRFC3281]. Die Seriennummer muss für alle Zertifikate eines Issuers (CA-Zertifikat) eindeutig sein.

4.4 Erlaubte Signaturalgorithmen, Feld `Signature`

Das Feld `Signature` definiert den zugrunde liegenden Signaturalgorithmus, der benutzt wurde, um das Zertifikat zu signieren.

ASN.1-Struktur:

```
AlgorithmIdentifier ::= SEQUENCE
{
          algorithm             OBJECT IDENTIFIER
          parameters            ANY DEFINED BY algorithm OPTIONAL
}
```

Gültige Werte:

Folgende Signaturalgorithmen sind derzeit zulässig (laut aktuellem Algorithmenkatalog der Bundesnetzagentur [externAlgCat]):

- sha256WithRSAEncryption OBJECT IDENTIFIER ::= { 1 2 840 113549 1 1 11 }
- sha512WithRSAEncryption OBJECT IDENTIFIER ::= { 1 2 840 113549 1 1 13 }

oder die entsprechenden OIDs für PKCS#1 version 2.1 (RSASSA-PSS und RSAES-OAEP).

Derzeit (03.2009) muss sha256WithRSAEncryption verwendet werden. Die Signaturalgorithmen müssen stets den gesetzlichen Anforderungen entsprechen und können aus diesem Grund ohne Anpassung dieses Dokumentes in Abstimmung mit der BÄK geändert werden.

Optional können auch ECDSA-Signaturalgorithmen zugelassen werden.

Im Feld `parameters` können zusätzliche Parameter definiert werden, wenn sie benötigt werden bzw. zulässig sind, bspw. Kurvenparameter für ECDSA-Algorithmen.

4.5 Gültigkeitszeitraum, Feld `attCertValidityPeriod`

Der Gültigkeitszeitraum des Attributzertifikates wird im Feld `AttCertValidityPeriod` definiert. Erlaubte Werte sind in Tabelle 1 eingetragen. Ein Attributzertifikat darf nicht länger gültig sein, als die zugrunde liegende Algorithmen nach [externAlgCat] und als sein korrespondierendes Basiszertifikat.

```
AttCertValidityPeriod ::= SEQUENCE
{
          notBeforeTime         GeneralizedTime
          notAfterTime          GeneralizedTime
}
```

Die Zeit wird immer als `GeneralizedTime` im Format YYYYMMDDhhmmssZ kodiert (Achtung! Unterschied in der Kodierung zwischen Attributzertifikat und Basiszertifikat!). Durch die Benutzung des Kettenmodells für alle Zertifikate darf ein Zertifikat länger gültig sein, als sein Aussteller, wenngleich dies für die Zertifikate im eArztausweis durch eine Festlegung der Bundesärztekammer eingeschränkt werden kann.

4.6 Aussteller des Zertifikats, Feld `Issuer`

Der Name des signierenden Zertifikats (Ausstellers) wird im Feld `Issuer` geführt. Es muss exakt mit dem Inhalt des Subject-Feldes des Ausstellers inkl. Kodierung und Reihenfolge der Strukturen übereinstimmen (so dass die Hashwerte beider Strukturen übereinstimmen), auch wenn ggf. Abweichungen im [externRFC3281] toleriert werden. Als Aussteller eines Attributzertifikats soll der Aussteller des korrespondierenden Basiszertifikats verwendet werden.

Folgende Attribute werden benutzt. Die erlaubten Werte sind in der Tabelle 1 aufgeführt:

- `countryName` (DE, ISO 3166 Code)
- `organizationName` (gemäß Tabelle 1)
- optional `organizationalUnitName`
- `commonName` (gemäß Tabelle 1)

Die ASN.1-Struktur:

```
Name ::= CHOICE {RDNSequence}

RDNSequence ::= SEQUENCE OF RelativeDistinguishedName

RelativeDistinguishedName ::= SET OF AttributeTypeAndValue

AttributeTypeAndValue ::= SEQUENCE
{
     type            AttributeType
     value           AttributeValue
}
```

```
AttributeType ::== OBJECT IDENTIFIER
AttributeValue ::= ANY DEFINED BY AttributeType
```

Die erlaubten Attribute werden in Tabelle 5 spezifiziert (id-at = 2.5.4).

Attribut	OID	Kodierung	max. Länge
commonName	{id-at 3}	UTF8	64
organizationName	{id-at 10}	UTF8	64
(organizationalUnitName)	{id-at 11}	UTF8	64
countryName	{id-at 6}	PrintableString	2

Tabelle 5: Attribute im IssuerDN

4.7 Zertifikatsinhaber, Feld `Subject`

Im Feld `Subject` wird die Zuordnung des Attributzertifikats zum Basiszertifikat hergestellt. Die Zuordnung zur Person des Zertifikatsinhabers wird dann durch das Basiszertifikat hergestellt. Es können mehrere Attributzertifikate zu einem Basiszertifikat ausgestellt werden, jedoch nur eins für das Berufsgruppenattribut. Das Subject wird zwingend als `baseCertificateID` aufgeführt.

Die ASN.1-Struktur vom Feld `subject`:

```
subject  CHOICE {
baseCertificateID     [0] EXPLICIT IssuerSerial,
subjectName           [1] EXPLICIT GeneralNames }

IssuerSerial ::= SEQUENCE {
      issuer GeneralNames,
      serial CertificateSerialNumber,
      issuerUID UniqueIdentifier OPTIONAL }
```

Als `CHOICE` muss `baseCertificateID` verwendet werden. `issuer` ist der Aussteller, `serial` ist die Seriennummer des korrespondierenden Basiszertifikats. Der optionale `issuerUID` darf nicht gesetzt werden.

4.8 Attribute, Feld `Attributes`

Das Feld `Attributes` definiert Eigenschaften („Attribute“) der Person, der das Attributzertifikat (und das korrespondierende Signaturzertifikat) gehört. In den hier beschriebenen Attributzertifikaten wird das Attribut „Admission“ mit dem OID {1 3 36 8 3 3} gesetzt.

4.8.1 Admission (1.3.36.8.3.3)

Das Admission-Attribut kann das Berufsgruppenattribut „Ärztin/Arzt“ oder ein anderes Attribut (Definition im Ermessen der BÄK) sowohl als Text als auch in Form eines maschinenlesbaren OID enthalten. Es wird in allen Attributzertifikaten aufgenommen.

ASN.1-Struktur:

```
id-commonpki-at-admission OBJECT IDENTIFIER ::= { commonpki-at 3 }
id-commonpki-at-namingAuthorities OBJECT IDENTIFIER ::= { commonpki-
at 11 }
AdmissionSyntax                 ::= SEQUENCE {
     admissionAuthority         GeneralName OPTIONAL,
     contentsOfAdmissions       SEQUENCE OF Admissions}

Admissions                      ::= SEQUENCE {
     admissionAuthority         [0] EXPLICIT GeneralName OPTIONAL,
     namingAuthority            [1] EXPLICIT NamingAuthority
OPTIONAL,
     professionInfos            SEQUENCE OF ProfessionInfo}

NamingAuthority                 ::= SEQUENCE {
     namingAuthorityId          OBJECT IDENTIFIER OPTIONAL,
     namingAuthorityUrl         IA5String OPTIONAL,
     namingAuthorityText        DirectoryString (SIZE(1..128))
OPTIONAL }

ProfessionInfo                  ::= SEQUENCE {
     namingAuthority            [0] EXPLICIT NamingAuthority
OPTIONAL,
     professionItems            SEQUENCE OF DirectoryString
(SIZE(1..128)),
     professionOIDS             SEQUENCE OF OBJECT IDENTIFIER
OPTIONAL,
     registrationNumber         PrintableString (SIZE(1..128))
OPTIONAL,
     addProfessionInfo          OCTET STRING OPTIONAL }
```

Gültige Werte:

Es wird genau eine admissionAuthority auf der oberste globale Ebene des Attributes gesetzt. Diese besteht aus einem DistinguishedName (X.501-Name) mit folgenden Elementen:

- Country=“DE“ (kodiert als printableString)

- Organisation="Landesärztekammer Nordrhein" (Beispiel), UTF-8-kodiert

und bezeichnet die „Bestätigende Stelle" (bestätigende Ärztekammer) für das Attribut. Als admissionAuthority-Bezeichner der jeweiligen, bestätigenden Ärztekammern sind ausschließlich die vollständigen ausgesprochenen Namen der in [baekWebdienst] im XML-Antragsdatensatz mit dem Element `<xsd:simpleType name="kammer">` beschriebenen Werte zu verwenden. [Anmerkung: BZÄK-Bezirksärztekammer; LÄK-Landesärztekammer; ÄK-Ärztekammer]

Es wird genau eine „Admissions"-Struktur aufgenommen, die genau ein ProfessionInfo enthält. Weder eine admissionAuthority noch eine NamingAuthority werden auf diesem Level gesetzt.

Die ProfessionInfo enthält genau ein ProfessionItem und genau einen ProfessionOID.

Das ProfessionItem enthält den UTF-8-kodierten String „Ärztin/Arzt" oder einen anderen von der BÄK definierten Text.

Der ProfessionOID enthält den OID für „Ärztin/Arzt" nach der folgenden Tabelle oder einen anderen von der BÄK definierten OID:

Karte	ProfessionOID	Name
HPC Version 2.1.1 Entwicklerkarte	1.3.6.1.4.1.24796.4.11.1	id-baek-at-namingAuthorityÄrzteschaft-Ärztin/Arzt
HPC Version 2.1.1 Testkarte	1.3.6.1.4.1.24796.4.11.1	id-baek-at-namingAuthorityÄrzteschaft-Ärztin/Arzt
HPCqsig	1.3.6.1.4.1.24796.4.11.1	id-baek-at-namingAuthorityÄrzteschaft-Ärztin/Arzt
HPC Version 2.3.0 alle Karten	1.2.276.0.76.4.30	Ärztin/Arzt

Die Telematik-ID wird nicht im Attributzertifikat aufgenommen.

Weitere OIDs und professionItems können von der BÄK ohne Änderung dieses Dokumentes definiert werden und den ZDAs mitgeteilt werden. Sie müssen dann von den ZDAs implementiert werden.

4.9 Zertifikatserweiterungen, Feld `Extensions`

In der Struktur „Extensions" werden Zertifikatserweiterungen aufgenommen.

Die ASN.1-Struktur:

```
Extensions ::= SEQUENCE SIZE (1..MAX) OF Extension

Extension ::= SEQUENCE
{
     extnId          OBJECT IDENTIFIER
     critical        BOOLEAN DEFAULT FALSE
     extnValue       OCTET STRING
}
```

In `extnId` wird den Typ der in `extnValue` enthaltenen Extension definiert. Der boolean `critical` zeigt an, ob die Extension kritisch ist. Wenn eine Extension als `critical` gesetzt ist, muss ein Client, der sie nicht versteht, das Zertifikat ablehnen.

In Tabelle 6 werden die Extensions aufgeführt, die in den Attributzertifikaten verwendet werden.

Extension	OID	critical?
AuthorityKeyIdentifier	{2 5 29 35}	non critical
CertificatePolicies	{2 5 29 32}	non critical
CRLDistributionPoints	{2 5 29 31}	non critical
QCStatements	{1.3.6.1.5.5.7.3}	non critical
additionalInformation	{1 3 36 8 3 15}	non critical
authorityInfoAccess	{1.3.6.1.5.5.7.1}	non critical
ValidityModel	{1 3 6 1 4 1 8301 3 5}	non critical

Tabelle 6: Extensions

4.9.1 AuthorityKeyIdentifier (2.5.29.35)

Die nicht-kritische Extension AuthorityKeyIdentifier (OID: {2 5 29 35}) dient zur Identifizierung des öffentlichen Schlüssels des Ausstellers.

ASN.1-Struktur:

```
AuthorityKeyIdentifier ::= SEQUENCE
{
      keyIdentifier               [0] IMPLICIT KeyIdentifier OPTIONAL
      authorityCertIssuer         [1] IMPLICIT GeneralNames OPTIONAL
      authorityCertSerialNumber   [2] IMPLICIT CertificateSerialNumber
                                                          OPTIONAL
}
```

Gültige Werte:

In der Extension wird nur das Feld keyIdentifier verwendet. Der Wert enthält den SHA-1 Hashwert über den `subjectPublicKey` (ohne Tag, Länge, Padding-Bits) des Ausstellers.

4.9.2 CertificatePolicies (2.5.29.32)

Die Extension enthält die Referenzen zu den zugrunde liegenden Policies für die Zertifikate. Die CP und die zugehörige CPS wird über den OID der CP des Zertifikates, sowie über einen Link (URL) für die CPS des Trustcenters, referenziert. Die Extension ist „non-critical“. Qualifizierte Attributzertifikate müssen als Policy-OID id-commonpki-cp-accredited enthalten, neben einem zweiten OID (s. Tabelle), der die Policy der Bundesärztekammer für die e-Arztausweise definiert. Weitere Policies (z. B. eine gemeinsame Policy aller Leistungserbringer) können ebenso aufgenommen werden. Die PolicyOIDs und PolicyURLs sind Konfigurationsdaten, die von der BÄK in [baekConfigData] definiert und geändert werden können, ohne dass es einer Anpassung dieses Dokumentes bedarf. Wenn Anforderungen (z. B. Sicherheitsanforderungen) einer aufgeführten Policy durch eine andere aufgeführte Policy im selben Zertifikat „aufgeweicht“ werden, dann gelten stets die jeweiligen schärferen Anforderungen.

Folgende OIDs sind derzeit (03.2009) gültig:

HPC Version 2.1.1, nicht qualifiziert

- Entwicklerkarten, HPC-Version 2.1.1:
  - 1.3.6.1.4.1.24796.1.2: id-baek-cp-hbaEntwicklerkarteArzt (Inhaber ist KEIN Arzt, nur für Entwicklerkarten)
- Test-HBA[Arzt], HPC-Version 2.1.1:
  - 1.3.6.1.4.1.24796.1.1: id-baek-cp-hbaTestkarteArzt (Inhaber ist Ärztin/Arzt)

HPCqsig (kompatibel zu Version 2.1.1), qualifiziert

- eArztausweise (qualifiziert)
  - 1.3.6.1.4.1.24796.1.10: id-baek-cp-eArztausweisV1 (Policy (Version 1) für den elektronischen Arztausweis mit qualifizierter Signatur)
  - 1.3.36.8.1.1: id-commonpki-cp-accredited

Das Attributzertifikat der qualifizierten HPCqsig-Karte hat also zwei OIDs (id-baek-cp-eArztausweisV1 und id-commonpki-cp-accredited). Die Policy id-baek-cp-eArztausweisV1 entspricht der Gemeinsamen Policy [leoGemPolicy] und darf nur für elektronische Arztausweise mit qualifizierter Signatur, die von einer Ärztekammer als solche freigegeben wurden, verwendet werden.

HPC Version 2.3.0

- Entwicklerkarten
  - 1.2.276.0.76.4.112: policy-muster-010000-cp (Unpersonalisiertes Zertifikat, Inhaber ist KEIN Arzt)
  - 1.2.276.0.76.4.73: hba-sig

Das Attributzertifikat der Entwicklerkarten der HPC-Spec Version 2.3.0 hat also zwei Policy-OIDs: policy-muster-010000-cp und hba-sig.

- eArztausweise
  - 1.2.276.0.76.4.62: policy-hba-010000-cp (Gemeinsame Policy für die Ausgabe der HPC, [leoGemPolicy])
  - 1.2.276.0.76.4.72: hba-qes
  - 1.3.36.8.1.1: id-commonpki-cp-accredited

Das qualifizierte Attributzertifikat der eArztausweise der HPC-Spec Version 2.3.0 hat also drei Policy-OIDs.

Karte	**PolicyOID**
HPC Version 2.1.1, Entwicklerkarte	1.3.6.1.4.1.24796.1.2： id-baek-cp-hbaEntwicklerkarteArzt (Inhaber ist KEIN Arzt, nur für Entwicklerkarten)
HPC-Version 2.1.1, Test-HBAs[Arzt]	1.3.6.1.4.1.24796.1.1: id-baek-cp-hbaTestkarteArzt (Inhaber ist Ärztin/Arzt),
HPCqsig	1.3.6.1.4.1.24796.1.10: id-baek-cp-eArztausweisV1 (Inhaber ist Ärztin/Arzt) 1.3.36.8.1.1: id-commonpki-cp-accredited
HPC Version 2.3.0, Entwicklerkarte	1.2.276.0.76.4.112: policy-muster-010000-cp 1.2.276.0.76.4.73: hba-sig
HPC Version 2.3.0,	TBD

Testkarte	
HPC Version 2.3.0, eArztausweis	1.2.276.0.76.4.62: policy-hba-010000-cp 1.2.276.0.76.4.72: hba-qes 1.3.36.8.1.1: id-commonpki-cp-accredited

ASN.1-Struktur:

```
CertificatePolicies ::= SEQUENCE SIZE (1..MAX) OF PolicyInformation

PolicyInformation ::= SEQUENCE
{
     policyIdentifier     CertPolicyId
     policyQualifiers     SEQUENCE SIZE (1..MAX) OF
PolicyQualifierInfo OPTIONAL
}

CertPolicyId ::= OBJECT IDENTIFIER

PolicyQualifierInfo ::= SEQUENCE {
     policyQualifierId PolicyQualifierId,
     qualifier ANY DEFINED BY policyQualifierId
}

PolicyQualifierId ::= OBJECT IDENTIFIER
{id-qt-cps | id-qt-unotice }

CPSUri ::= IA5String
```

Der Wert vom `policyIdentifier` ist der in der Tabelle 2 aufgeführter OID. `policyQualifierId` hat den Wert `id-qt-cps`. Die URL auf die CPS der Bundesärztekammer wird als Wert im `CPSUri` aufgenommen (s. Tabelle 2).

4.9.3 CRLDistributionPoints (2.5.29.31)

Die Extension gibt den CRL-Issuer an und die URL, die die CRL enthält. Sie ist „nicht-kritisch“.

ASN.1-Struktur:

```
CrlDistributionPoints ::= SEQUENCE SIZE (1..MAX) OF
CrlDistributionPoint

CrlDistributionPoint ::= SEQUENCE
{
     distributionPoint     [0]   EXPLICIT DistributionPointName
OPTIONAL
     reasons               [1]   IMPLICIT ReasonFlags OPTIONAL
     cRLIssuer             [2]   IMPLICIT GeneralNames OPTIONAL
}

DistributionPointName ::= CHOICE
{
     fullName                    [0] IMPLICIT GeneralNames
```

```
     nameRelativeToCRLIssuer     [1] IMPLICIT
RelativeDistinguishedName
}
```

Gültige Werte:

Im `fullName` wird die vollständige URL (LDAP oder HTTP oder beides, URL-konform kodiert) aufgeführt, unter welche die aktuelle CRL abgerufen werden kann (s. auch Tabelle 2). Die Felder `reasons` und `nameRelativeToCRLIssuer` werden nicht verwendet. Das Feld `cRLIssuer` muss laut [externCommonPKI] aufgenommen werden, da indirekte CRLs ausgestellt werden.

Anmerkungen:

Der Wert einer CRL ist im SigG-Kontext sowie auch für Authentisierungs- und Verschlüsselungszertifikate, die für die e-Arztausweise den SigG-Anforderungen genügen müssen, zumindest für eine erste Überprüfung eines Zertifikates, zweifelhaft. Der Grund ist, dass ausgestellte Zertifikate, die nicht im Verzeichnisdienst freigeschaltet wurden, nicht in einer CRL aufgenommen werden können, obwohl sie ungültig sind. Eine Verweigerung zur Freischaltung kann z. B. entstehen, wenn der Empfänger die Chipkarte nie bekommen hat (jemand anders hat sie abgefangen) oder wenn die Transport-PIN gebrochen war. Solche Zertifikate sind mathematisch korrekt aber formal „nicht existent“ und somit ungültig. Sie können auch nicht in der CRL aufgenommen werden, weil damit suggeriert würde, dass sie bis zur Sperrung gültig wären (eine rückwirkende Sperrung ist laut SigG nicht zulässig). Es wird somit hingewiesen, dass eine CRL-Prüfung nicht ausreicht, um die Gültigkeit eines Zertifikates zu einem bestimmten Zeitpunkt zu ermitteln. Eine OCSP-Prüfung ist erforderlich. Wenn allerdings ein Zertifikat einmal als „vorhanden“ über OCSP deklariert wurde, dann kann man künftig dafür auch eine CRL-Prüfung vornehmen.

Der CRL-Issuer kann sich im SigG-Kontext ändern. Dies ist z. B. möglich, wenn die Person, auf die das Zertifikat personalisiert ist, gestorben ist und somit auch nicht mehr unterschreiben kann. Laut SigG/SigV ist es nicht zulässig, ein bereits vergebenes Pseudonym (z.B. CRL-Signer 1:PN) einer zweiten Person zuzuordnen. Der DistinguishedName im Feld cRLSigner ist jedoch fest. Es ist also möglich, dass ein anderer CRL-Signer, als der im CRL-DP Eingetragene, die CRL unterschreiben muss. Dies ist dann der Fall, wenn ein Trustcenter aus Gründen der Hochverfügbarkeit zwei Standorte betreibt und SigG-konformen Chipkarten mit nicht-kopierbaren Schlüsseln einsetzt. Der zweite Standort muss dann einen anderen CRLSigner einsetzen als der erste Standort. Folgender Lösungsweg wird dafür sowohl für qualifizierte als auch für alle nicht-qualifizierten-Zertifikate definiert, gemäß Common-PKI Spezifikation [externCommonPKI] Part 9 S. 27ff:

Der Überprüfer einer CRL soll die CRL mittels dem URL im CRL-DP herunterladen und mit den in Common-PKI beschriebenen Validierungsalgorithmen überprüfen. Wenn der CRLSigner, der die CRL unterschrieben hat, nicht identisch ist mit dem CRLSigner, der im Zertifikat eingetragen ist, dann soll überprüft werden, ob das Zertifikat (dessen Gültigkeit mit Hilfe der CRL geprüft wird) und der CRLSigner, der die CRL signiert hat, aus der selben Zertifizierungshierarchie stammen. Wenn dies der Fall ist und auch alle anderen Prüfschritte positiv verlaufen, dann soll die CRL als gültig betrachtet werden.

4.9.4 QCStatements (1.3.6.1.5.5.7.3)

Die Kennzeichnung des Attributzertifikats als qualifiziertes Attributzertifikat nach der entsprechenden EU-Direktive (1999/93/EC) erfolgt mit der QCStatements-Extension und dem entsprechenden OID (s. gültige Werte). Eintragungen, welche die Nutzung des öffentlichen Schlüssels des Basis-Signaturzertifikats einschränken, dürfen im Attributzertifikat nicht aufgenommen werden, sondern müssen im zugehörigen Signaturzertifikat enthalten sein. D. h. in der QCStatements-Extension des Attributzertifikats darf eine Begrenzung für finanzielle Transaktionen nicht aufgenommen werden, sondern muss im Signaturzertifikat eingetragen werden[4]. Die Extension ist nicht-kritisch.

ASN.1-Struktur:

```
QCStatements ::= SEQUENCE OF QCStatement

QCStatement ::= SEQUENCE
{
      statementId                ObjectIdentifier
      statementInfo              ANY DEFINDED BY statementId OPTIONAL
}
```

Gültige Werte:

Die Extension muss genau ein QCStatement mit dem OID id-etsi-qcs-QcCompliance {id-etsi-qcs 1} ({0 4 0 1862 1 1}) als `statementId` enthalten. Ein `statementInfo` wird nicht gesetzt.

4.9.5 AdditionalInformation (1 3 36 8 3 15)

Diese optionale Extension gibt weitere Informationen (nicht einschränkender Natur) über die Verwendung des Attributzertifikates an. Sie ist nicht-kritisch.

ASN.1-Struktur:

```
AdditionalInformationSyntax ::= DirectoryString
```

Der `DirectoryString` darf maximal 2048 bytes lang sein und muss UTF-8-kodiert sein.

Gültige Werte:

In Abhängigkeit vom Herausgabemodell könnte der Inhalt der Extension beispielsweise lauten: “Zertifikat als Teil eines elektronischen Arztausweises, herausgegeben durch die zuständige Landesärztekammer”

[4] Wenn einschränkende Einträge im Attributzertifikat aufgenommen werden, dann muss das Attributzertifikat zwingend in jeder Signatur mitsigniert werden. Außerdem muss ein LiabilityLimitationFlag im Signaturzertifikat aufgenommen werden.

4.9.6 AuthorityInfoAccess (1.3.6.1.5.5.7.1)

Die Extension AuthorityInfoAccess enthält Angaben über die Quelle von Statusinformationen für die Validierung des Zertifikates. Für Statusinformationen für die e-Arztausweise wird das OCS-Protokoll verwendet. Die Extension ist nicht-kritisch und enthält die URL des zuständigen OCSP-Responders.

ASN.1-Struktur:

```
AuthorityInfoAccessSyntax ::= SEQUENCE (1..MAX) OF AccessDescription

AccessDescription ::= SEQUENCE
{
      accessMethod          OBJECT IDENTIFIER
      accessLocation        GeneralName
}
```

Gültige Werte:

Als `accessMethod` wird `id-ad-ocsp` {1 3 6 1 5 5 7 48 1} gesetzt. In der `accessLocation` wird die URL des für das Zertifikat zuständigen OCSP-Responders aufgenommen (URL-konform kodiert).

4.9.7 ValidityModel (1.3.6.1.4.1.8301.3.5)

Die Extension `ValidityModel` gibt das Gültigkeitsmodell an, das von einem Client für die Zertifikatsprüfung und Signaturvalidierung verwendet werden muss. Der verwendete OID kennzeichnet das Kettenmodell. Die Extension ist nicht kritisch.

Der OID der Extension `ValidityModel` ist `id-validityModel` {1 3 6 1 4 1 8301 3 5}.

ASN.1-Struktur:

```
ValidityModel::= SEQUENCE
{
      validityModelId       OBJECT IDENTIFIER
      validityModelInfo     ANY DEFINED BY validityModelId OPTIONAL
}
```

Gültige Werte:

Als `validityModelId` wird der Wert für das Kettenmodell `id-validityModel-chain` {1 3 6 1 4 1 8301 3 5 1} für alle Zertifikatstypen festgelegt.

Das Feld `validityModelInfo` wird nicht verwendet.

4.10 Das Feld `signatureAlgorithm`

Das Feld `signatureAlgorithm` entspricht dem Feld `signature`, wie im Abschnitt 4.4 „Erlaubte Signaturalgorithmen, Feld `Signature`“ beschrieben.

4.11 Das Feld `signatureValue`

Die Signatur auf das `tbsCertificate` wird im Feld `signatureValue` als BIT STRING aufgenommen.

5 Literatur

[externRFC3281] An Internet Attribute Certificate, Farrell, S.; Housley, R.; Request for Comments (RFC) 3281, April 2002.

[externCommonPKI] Common PKI Specification for Interoperable Applications, T7 & TeleTrusT, Version 2.0, 20.01.2009

[externAlgCat] Bekanntmachung zur elektronischen Signatur nach dem Signaturgesetz und der Signaturverordnung (Übersicht über geeignete Algorithmen) vom 17.11.2008, veröffentlicht am 27.01.2009 im Bundesanzeiger Nr. 13 S. 346, Bundesnetzagentur. http://www.bundesnetzagentur.de/media/archive/15549.pdf.

[baekCerts] Zertifikatsprofile für X.509 Basiszertifikate; Version 2.3.2; 12.05.11

[baekWebdienst] Webdienst-Spezifikation; Version 2.3.3; 12.05.11

[leoGemPolicy] Gemeinsame Policy für die Herausgabe der HPC; Version 0.9.3w2; 03.03.06, Bundesärztekammer, Bundespsychotherapeutenkammer, Bundeszahnärztekammer, Kassenzahnärztliche Bundesvereinigung, Werbe- und Vertriebsgesellschaft Deutscher Apotheker mbH

[baekConfigData] Konfigurationsdaten für die PKI der elektronischen Arztausweise, Version 2.3.4; 12.05.11