SOAP-Schnittstelle der DFN-PKI

Version 4.0 (28.11.2018)

dfnpca@dfn-cert.de

Inhaltsverzeichnis

1 Einleitung

1.1 Schnittstellenbeschreibung

In der DFN-PKI wird eine SOAP-Schnittstelle angeboten, die alle Funktionen der Webschnittstellen nicht nur für Menschen, sondern auch für Software aufrufbar anbietet. Die Umsetzung ist so gestaltet, dass ein Arbeitsschritt in der Webschnittstelle jeweils einem Prozeduraufruf in der SOAP-Schnittstelle entspricht. Dies ermöglicht die Bedienung der DFN-PKI auch durch selbst entwickelte Software, die einen SOAP-Client beinhalten. Eine solche Software kann z.B. Zertifikate in einer Stapelverarbeitung beantragen und genehmigen, wobei als Quelle der Benutzerdaten eine lokale Datenbank verwendet wird.

Bei der Entwicklung einer solchen Software ist stets zu beachten, dass die umgesetzten Prozesse auch konform zur Policy der DFN-PKI sind. Insbesondere ist eine Schlüsselerzeugung für Nutzer durch z.B. die Registrierungsstelle nur zulässig, wenn die verwendeten Verfahren detailliert in einer Erklärung zum Zertifizierungsbetrieb beschrieben und die Gefahr der unbefugten Verwendung vn Nutzerschlüsseln durch Dritte minimiert wurde.

Um zu klären, ob die geplanten Verfahren konform zur Policy der DFN-PKI sind, wenden Sie sich bitte unbedingt an dfnpca@dfn-cert.de.

1.2 Endpunkte der Kommunikation

Die SOAP-Schnittstelle ist analog zu der Webschnittstelle in eine öffentliche und eine Registrierungsschnittstelle unterteilt. Diese Trennung ist in der SOAP-Schnittstelle durch unterschiedliche Endpunkte der Kommunikation und auch durch unterschiedliche XML-Namensräume realisiert. Die Implementierung der öffentlichen und der Registrierungsschnittstelle sind erreichbar unter (wobei <*caname*> jeweils durch den konkreten Installationsnamen der CA ersetzt werden muss):

- https://pki.pca.dfn.de/<*caname*>/cgi-bin/pub/soap/DFNCERT/Public
- https://ra.pca.dfn.de/<*caname*>/cgi-bin/ra/soap/DFNCERT/Registration
- https://ra.pca.dfn.de/<*caname*>/cgi-bin/ra/soap/DFNCERT/Domains

Die Schnittstellen sind dokumentiert in der WSDL (Web Service Description Language) unter:

- https://pki.pca.dfn.de/<*caname*>/cgi-bin/pub/soap?wsdl=1
- https://ra.pca.dfn.de/<*caname*>/cgi-bin/ra/soap?wsdl=1
- https://ra.pca.dfn.de/<*caname*>/cgi-bin/ra/soap/DFNCERT/Domains?wsdl=1

1.3 Zeichensatz

Bei der Kommunikation mit der Schnittstelle kann ein Unicode-Zeichensatz in Form von *UTF-8* oder die westeuropäische Kodierung *ISO-8859-1* verwendet werden. Wichtig ist dabei, dass die angegebene Kodierung in dem Attribut *encoding* der XML-Deklaration in der ersten Zeile mit der tatsächlichen Kodierung der Nachricht übereinstimmt. Der Server antwortet immer mit dem Zeichensatz *UTF-8*.

Signaturen können nur dann korrekt von der CA verifiziert werden, wenn die signierten Daten im Zeichensatz *ISO-8859-1* vorliegen. Die zu signierenden Daten werden von den Aufrufen *getRawRequest* und *getRawRevocationRequest* in genau dieser Kodierung zurückgegeben. Die Daten werden dazu als Wert vom Typ *xsd:base64Binary* übertragen. Diese BASE64-kodierte Übertragung stellt sicher, dass die Daten Byte für Byte genau so ankommen, wie diese von der CA signiert erwartet werden. Es muss darauf geachtet werden, dass ein Client die Daten 1:1 signiert und diese nicht unbeachtet in *UTF-8* umgewandelt werden, was in manchen Programmiersprachen der Standard für Zeichenketten ist. Eine Signatur über einen Antrag mit deutschen Umlauten könnte z.B. dadurch fehlschlagen.

1.4 Client-Authentifizierung

Der Zugang zu der öffentlichen Schnittstelle ist wie für die Webschnittstelle standardmäßig nicht durch eine Authentifizierung beschränkt. Jeder Zugriff auf die Registrierungsschnittstelle ist durch eine HTTPS-Client-Authentifizierung mit einem X.509-Zertifikat (das RA-Zertifikat) gesichert. Ein SOAP-Client muss bei der Authentifizierung immer die komplette Zertifizierungskette senden (d.h. mit allen CA-Zertifikaten).

1.5 Registrierungsstellen

Einer CA können in der DFN-PKI beliebig viele Registrierungsstellen (RAs) untergeordnet sein. Jeder Registrierungsstelle sind ein oder mehrere Namensräume zugeordnet, in denen Zertifikate beantragt und ausgestellt werden dürfen. Die Registrierungsstellen werden mit dezimalen Nummern *0...n* bezeichnet. Immer vorhanden ist die RA mit der Nummer 0, in der alle Anträge untergeordneter RAs eingesehen werden können. Jedes RA-Operator-Zertifikat ist einer RA-Nummer (RA_ID) eindeutig zugeordnet. Diese RA_ID ist nicht Bestandteil des Zertifikats selbst, sondern nur als Information zu einem Zertifikat in der Datenbank der CA abgespeichert.

Die Nummer muss in der SOAP-Schnittstelle nur auf der öffentlichen Schnittstelle bei einem Antrag mittels *newRequest* und *newRevocationRequest* übergeben werden. Für die Funktionalität der Registrierungsstelle in der SOAP-Schnittstelle entscheidet das verwendete RA-Operator-Zertifikat darüber, welche RA angesteuert wird. Über das Feld *RALoginID* aus der Struktur *DFNCERTTypesCAInfo* kann ein Client nach dem Aufruf der Methode *getCAInfo* die RA_ID erhalten, mit der dieser auf dem Server identifiziert wurde.

Eine Liste aller verfügbaren Registrierungsstellen unterhalb einer CA kann mit einem Aufruf *getCAInfo* abgerufen werden. Die Antwort enthält Informationen über alle konfigurierten Registrierungsstellen mit den jeweils gültigen DN-Prefixen.

Der Parameter RaID muss bei der Implementierung einer Client-Anwendung flexibel gehalten werden, da dieser im Testbereich der DFN-PKI von der produktiven Umgebung abweicht.

1.6 Fehlerbehandlung

Generell liefert jeder Aufruf bei einem Fehler eine SOAP-Fault-Nachricht. Das Element *Faultstring* enthält eine Fehlermeldung im Klartext, die für eine direkte Anzeige für den Benutzer geeignet ist. Diese Fehlermeldung könnte in Zukunft geändert werden und ein Client darf sich nicht auf den Inhalt der Meldung verlassen.

2 Zertifizierung über SOAP

2.1 Rollen im Zertifizierungsprozess

Ein SOAP-Client kann beide Rollen, die des Zertifikatnehmers und die der Registrierungsstelle übernehmen. Eine Anwendung kann einen Antrag generieren und hochladen, später genehmigen und das Zertifikat nach dessen Ausstellung abholen. In der nachfolgenden Beschreibung wird davon ausgegangen, dass ein SOAP-Client beide Rollen einnimmt und damit auch beide Schnittstellen bedient.

2.2 Einzelne Zertifikate

Der Beantragungsprozess von Zertifikaten ist in der SOAP-Schnittstelle an die Vorgehensweise in der HTML-Webschnittstelle angelehnt. Zunächst erzeugt der Client ein Schlüsselpaar und ein PKCS#10-Antrag. Ein Aufruf von *newRequest* entspricht dann dem Ausfüllen des Formulars für Serveranträge auf der Webseite und übermittelt den Antrag mit allen weiteren Informationen an den Server. Der Rückgabewert ist die Seriennummer des Antrags auf dem Server, wodurch der Client den Antrag für alle späteren Aktionen eindeutig identifizieren kann.

Ein Aufruf von *getRawRequest* liefert anschließend die Daten des Antrags in einer speziellen Form, die von der RA signiert und wieder zurück an die CA übermittelt werden muss. Dabei müssen diese Daten zunächst mit einem gültigen RA-Operator-Zertifikat signiert und ein PKCS#7-Container mit einer Signatur erzeugt werden. Der PKCS#7-Container wird dann im PEM-Format bei einem Aufruf von *approveRequest* übergeben, wodurch der Antrag genehmigt wird. Nachdem der Antrag genehmigt wurde, wird das beantragte Zertifikat durch die CA nach einem nicht definierten Zeitabstand ausgestellt. Daher wird als letzte Aktion *getCertificateByRequestSerial* in gewissen Zeitabständen aufgerufen, bis der Server das ausgestellte Zertifikat zurückliefert. Die Zeitabstände dürfen nicht zu gering gewählt werden. 15 Sekunden sind ein akzeptabler Wert. Der Client sollte einen Timeout-Mechanismus benutzen, um nach einer bestimmten Anzahl von Versuchen mit einem Fehler abzubrechen.

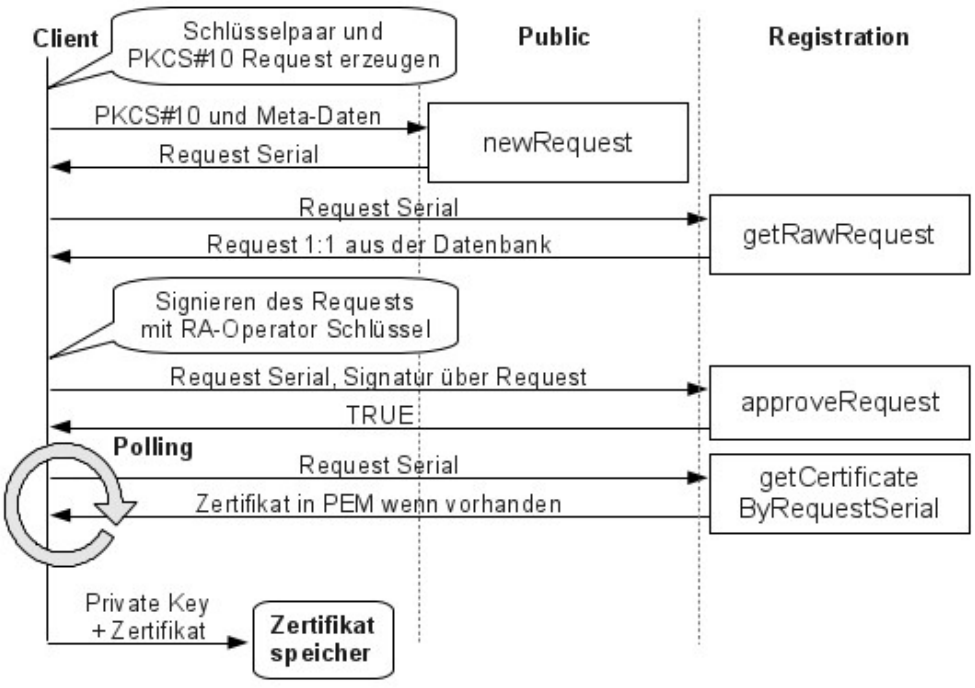

Abbildung 1: Ablauf einer Zertifizierung mit der SOAP-Schnittstelle

Abbildung 1 zeigt den Ablauf einer Zertifizierung über die SOAP-Schnittstelle. Die Beschreibungen an den Pfeilen stellen dabei die Parameter bzw. Rückgabewerte der in den Kästen dargestellten Aufrufe dar. Die Akteure sind horizontal abgetrennt: Client, Public und Registration sind physikalisch getrennte Kommunikationspartner. Bei dem dargestellten Zertifikatspeicher kann es sich eine PKCS#12-Datei oder eine SmartCard bzw. USB-Token handeln.

2.3 Mehrere Zertifikate

Bei der Beantragung von mehreren Zertifikaten über die SOAP-Schnittstelle (z.B. Batch-Betrieb) wird empfohlen, zunächst alle Zertifikatanträge zu übertragen, diese zu signieren und die entstehenden Antragsnummern in einer Liste vorzuhalten. Anschließend sollte das Abfragen für jedes Zertifikat in der Liste einzeln durchgeführt werden, wobei bei dem zuerst beantragten Zertifikat begonnen werden sollte. Dadurch ergibt sich ein höherer Durchsatz, weil in der DFN-PKI alle genehmigten Anträge in bestimmten Zeitabständen als Block verarbeitet werden.

Abbildung 2 zeigt den Ablauf bei einer Zertifizierung mit mehreren Anträgen. Es werden zunächst die Anträge 1..n mittels *newRequest* übertragen, die entstandenen Seriennummern in einer Liste gespeichert, der jeweilige Antrag durch *getRawRequest* zur Signatur ermittelt, der Antrag signiert und dann mittels *approveRequest* genehmigt.

Anschließend wird damit begonnen, nach dem Zertifikat zu Antrag 1 mittels *getCertificateByRequestSerial* zu fragen. Dies wird solange wiederholt, bis das Zertifikat ausgestellt wurde und vom Server geliefert wurde. Zwischen den einzelnen Aufrufen von *getCertificateByRequestSerial* muss eine Pause von 15 Sekunden eingehalten werden.

Mit den weiteren Anträgen bis n wird nun genauso verfahren.

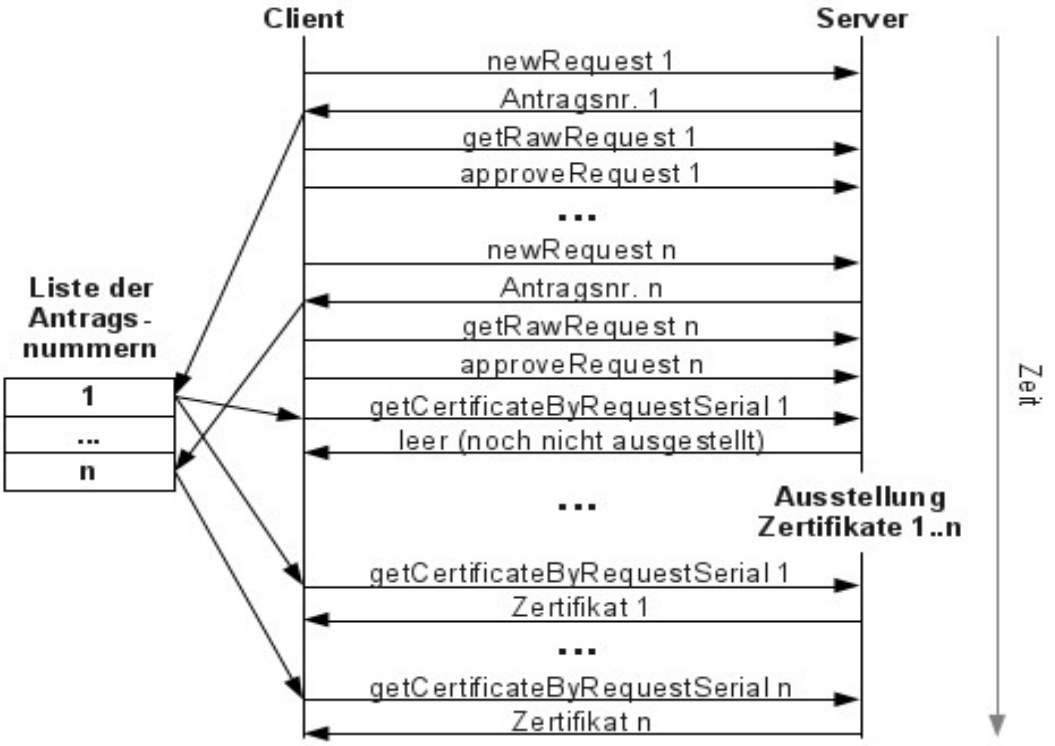

Abbildung 2: Ablauf einer Zertifizierung mit mehreren Anträgen

3 Client-Anwendungen

3.1 Anforderungen an eine Client SOAP-Implementierung

Um mit der SOAP-Schnittstelle der DFN-PKI kommunizieren zu können, muss die SOAP Implementierung eines Clients folgende Punkte unterstützen:

- SOAP Version 1.1
- Binding für den Kommunikationsstil rpc/encoded
- SSL-Client-Authentifizierung

3.2 Der DFN-PKI-Client

Für Java wurde eine Clientbibliothek für die SOAP-Schnittstelle der DFN-PKI entwickelt, die in eigenen Anwendungen integriert werden kann. Der DFN-PKI-Client kapselt die SOAP-Aufrufe und bietet ein etwas abstrakteres API.

Der Client eignet sich gut für die Verbreitung durch z.B. Java Webstart. Weiterhin bietet der Client zusätzliche Crypto-Funktionen wie z.B. das Erstellen eines PKCS#10-Antrags oder einer PKCS#7-Signatur an.

Der DFN-PKI-Client setzt die Bibliothek BouncyCastle aus dem gleichnamigen Projekt als Security Provider voraus.

3.2.1 Integration in Anwendungen

Alle Funktionen zur Kommunikation mit den Servern der DFN-PKI besitzen keine weiteren Abhängigkeiten und können mit einer Java VM ab Version 1.5 verwendet werden. Um die Methoden für die Erstellung von PKCS#10-Anträgen oder der Erstellung einer Signatur in einem PKCS#7-Container nutzen zu können, muss die BouncyCastle Crypto-API eingebunden werden. Die erforderlichen Dateien können von der Homepage des BouncyCastle-Projekts (http://www.bouncycastle.org) unter einer der MIT-X11-Lizenz ähnlichen Lizenz bezogen werden (Bitte beachten Sie die Lizenzinformationen auf der Webseite des BouncyCastle-Projekts). Um den Client in eigenen Anwendungen zu verwenden, muss dessen JAR-Archiv sowie die JAR-Archive des Bouncy Castle Crypto-API im Klassenpfad eingetragen werden.

Die Bibliothek *soapclient.jar* enthält die wichtige Klasse *de.dfncert.tools.DFNPKIClient*, die die wichtigsten Methoden für die Kommunikation mit der DFN-PKI sowie die Verwendung eines RA-Operator-Zertifikats kapselt. Die Klasse muss nur mit dem Namen der anzusprechenden CA instanziert werden und kann das RA-Zertifikat aus einer PKCS#12-Datei oder einem PKCS#11-Gerät (z.B. USB-Token) für die Aktionen der Registrierungsstelle laden. Detaillierte Informationen und Verwendungsbeispiele befinden sich in der Javadoc-Dokumentation zu der soapclient-Bibliothek.

3.2.2 Kryptografische Hilfsmethoden

In der Klasse *de.dfncert.tools.Cryptography* werden einige statische Methoden angeboten, die der Erzeugung eines Zertifikatantrags oder einer Signatur dienen. Diese Methoden sind in ihren Parametern sehr einfach gehalten und kapseln den Aufwand von immer wiederkehrenden Aufgaben. Die Methoden sind in der Javadoc-Dokumentation beschrieben.

3.2.3 Sperrprüfung

Üblicherweise sollten in der Java VM die Mechanismen zur Sperrprüfung der Zertifikate beim Aufbau einer SSL-Verbindung eingeschaltet werden. Hierzu kann die Methode *setCheckRevocation(boolean bCheckRevocation)* in der Klasse *de.dfncert.tools.DFNPKIClient* verwendet werden, die dann die Systemproperty *com.sun.net.ssl.checkRevocation* entsprechend setzt.

Wichtig: Der DFNPKIClient setzt niemals selbsttätig *com.sun.net.ssl.checkRevocation*!

3.2.4 Quelltext-Beispiel

Siehe Javadoc-Dokumentation von der Klasse de/dfncert/tools/DFNPKIClient.java

4 Funktionsreferenz der öffentlichen Schnittstelle

4.1 Zertifikate beantragen

4.1.1 newRequest

RaID	xsd:int	Nummer der RA, 0 für die Master-RA
PKCS10	xsd:string	Der Zertifikatantrag im PEM-Format
AltNames	xsd:string[]	Subject Alternative Names in der Form ("typ:wert", …)
Role	xsd:string	Die Rolle des beantragten Zertifikats
Pin	xsd:string	Das Sperrkennwort für das Zertifikat als SHA-1 Hash
AddName	xsd:string	Vollständiger Name des Antragstellers
AddEMail	xsd:string	E-Mail Adresse des Antragstellers
AddOrgUnit	xsd:string	Abteilung des Antragstellers
Publish	xsd:boolean	Veröffentlichung des Zertifikats
Rückgabe	**xsd:int**	**Die Seriennummer des hochgeladenen Antrags**

Lädt einen PKCS#10-Antrag mit Parametern zum Server in der durch den Parameter RaID angegebenen RA hoch (siehe hierzu Kapitel *Registrierungsstellen in der DFN-PKI*). Der Parameter *PKCS10* muss einen PKCS#10-Zertifikatantrag im PEM-Format (inklusive des Headers *-----BEGIN CERTIFICATE REQUEST-----* und des entsprechenden Footers) enthalten. Zu beachten ist, dass Erweiterungen, die in der PKCS#10-Struktur enthalten sind, nicht ausgewertet werden. Alternative Namen müssen stattdessen über den Parameter *AltNames* angegeben werden.

Im Parameter *AltNames* kann eine Liste mit beliebig vielen Namen angegeben werden, die als X509v3-Extension „Subject Alternative Name" eingetragen werden. Die Liste muss dabei Zeichenketten enthalten, die als Schlüssel/Wert-Paar getrennt durch einen Doppelpunkt aufgebaut sind: *<typ>:<wert>*. Für *<typ>* werden dabei folgende Werte unter Beachtung der Groß-/Kleinschreibung unterstützt:

email	Alternative E-Mail Adresse
DNS	Eintrag eines alternativen DNS-Namen (z.B. CNAME eines Servers)
IP	Eine IP-Adresse
URI	Ein Unique Resource Identifier wie z.B. eine URL
Microsoft_UPN	Ein Principle Name für die SmartCard Anmeldung bei Microsoft

In dem Parameter *Role* muss eine von der CA unterstützte Rolle für das Zertifikat angegeben werden. Die Rolle beinflusst die Erweiterungen in den ausgestellten Zertifikaten und kann in Absprache mit der DFN-PCA beliebig eingerichtet werden.

In dem Parameter *Pin* muss ein SHA1-Fingerabdruck des Sperrpassworts enthalten sein. Die Angabe muss hier in hexadezimaler Form als Zeichenkette erfolgen. Die Parameter *AddName*, *AddEmail* und *AddOrgUnit* können den Namen, die E-Mail-Adresse und die Abteilung des An-

tragstellers enthalten. Zu beachten ist, dass für einen Zerfifikatnehmer immer eine E-Mail-Adresse verfügbar sein muss. Falls im Subject-DN oder den alternativen Namen keine E-Mail-Adresse angegeben wurde, muss der Parameter *AddEMail* zwingend ausgefüllt sein. Der Parameter *Publish* legt fest, ob das ausgestellte Zertifikat öffentlich im LDAP-Server und in der Webschnittstelle zur Verfügung steht.
Alle Parameter dieser Funktion unterliegen einer Syntaxprüfung:

- Der Antrag ist enthält einen korrekten PKCS#10-Antrag und ist korrekt selbst signiert
- Der Subject-DN in dem Antrag entspricht den Konventionen der CA
- Die beantragte Rolle wird von der CA unterstützt
- Die PIN liegt in einem korrekten SHA-1 Fingerabdruck vor (40 Zeichen, Hexstring)
- Die erweiterten Parameter sind eingetragen und enthalten sinnvolle Werte (Prüfung auf E-Mail Syntax, leere Werte)
- Die Veröffentlichung ist vereinbar mit der Policy der CA

Der Rückgabewert dieses Aufrufes ist die Seriennummer des Antrags in der CA. Dieser wird in vielen Aufrufen der Registrierungsschnittstelle zum referenzieren eines Zertifikatantrags benutzt.

4.1.2 newRevocationRequest

RaID	xsd:int	Nummer der RA, 0 für die Master-RA
Serial	xsd:integer	Die Seriennummer des Zertifikats
Reason	xsd:string	Der Grund für die Sperrung
Pin	xsd:string	Das Sperrkennwort für das Zertifikat als SHA-1 Hash
Rückgabe	**xsd:int**	**Die Seriennummer des neuen Sperrantrags**

Stellt einen Sperrantrag für ein bestimmtes Zertifikat. Der Grund für die Sperrung kann auch leer sein. In der DFN-PKI wird der Grund für die Sperrung zur Zeit nicht in die Sperrlisten als Erweiterung aufgenommen.

4.1.3 getRequestPrintout

RaID	xsd:int	Nummer der RA, 0 für die Master-RA
Serial	xsd:int	Die Seriennummer des Zertifikatantrags
Format	xsd:string	Rückgabeformat
Pin	xsd:string	Das Sperrkennwort für das Zertifikat als SHA-1 Hash
Rückgabe	**xsd:base64-Binary**	**Der Ausdruck des Zertifikatantrags**

Liefert analog zu dem Aufruf *getRequestPrintout* in der Registrierungsstelle den Ausdruck des Zertifikatantrags. Als Format wird dabei momentan lediglich *application/pdf* für die Rückgabe eines PDF-Dokuments unterstützt. Da in der öffentlichen Schnittstelle keine Authentifizierung mit einem RA-Operator-Zertifikat erfolgt, muss hier zusätzlich der Parameter *RaID* angegeben werden. Weiterhin muss im Parameter *Pin* die Sperr-PIN des Zertifikatantrags übergeben werden, um so eine Abfrage durch Unbefugte zu verhindern.

4.1.4 getCertificateByRequestSerial

RaID	xsd:int	Nummer der RA, 0 für die Master-RA
Serial	xsd:integer	Die Seriennummer des Zertifikats
Pin	xsd:string	Das Sperrkennwort für das Zertifikat als SHA-1 Hash

Rückgabe	**xsd:string**	**Das ausgestellte Zertifikat im PEM-Format**

Liefert analog zu dem Aufruf *getCertificateByRequestSerial* in der Registrierungsstelle das Zertifikat zu einem bestehenden Antrag. Der Antrag wird dabei durch die Nummer referenziert, die von *newRequest* zurückgegeben wurde. Falls (noch) kein Zertifikat zu der Antragsnummer Serial existiert, wird eine leere Zeichenkette zurückgeliefert. Anders als in der Registrierungsstelle müssen hier aufgrund fehlender Authentifizierung durch ein RA-Operator-Zertifikat die RA-Nummer *RaID* und die Sperr-PIN des Zertifikatantrags *Pin* angegeben werden.

4.1.5 getValidDomains

RaID	xsd:int	Nummer der RA, 0 für die Master-RA
Type	xsd:string	Domain-Typ: 'server' oder 'email'
Rückgabe	**tns1:DFN-CERTTypes-ValidDo-main[]**	**Liste mit Domain-Einträgen**

Liefert alle gültigen Domain-Einträge für die gewünschte RA-ID und den gewünschten Typ zurück. Dabei werden beim Typ 'server' alle 'server' und 'server-host', beim Typ 'email' alle 'email' und 'email-host' Domain-Einträge zurückgegeben. Wird kein Typ angegeben werden alle gültigen Domain-Einträge zurückgegeben.

4.1.6 getRequestInfo

RaID	xsd:int	Nummer der RA
Serial	xsd:int	Die Seriennummer des Zertifikatantrags
Pin	xsd:string	Das Sperrkennwort für das Zertifikat als SHA-1 Hash
Rückgabe	**tns:DFNCERT TypesRequest Info**	**Struktur mit Informationen über den Antrag**

Liefert detailierte Informationen über einen Zertifikatantrag in einer *DFNCERTTypesRequest*-Struktur. Diese enthält alle veränderbaren Parameter sowie nicht veränderbaren Informationen über den Antrag.

4.1.7 getCAInfo

RaID	xsd:int	Nummer der RA
Rückgabe	**tns:DFNCERT TypesCAInfo**	**Struktur mit Informationen über die CA**

Liefert Informationen über die Zertifizierungsstelle für die gewünschte RA-ID. Dazu gehören der Installationsname der CA, der volle Name der CA sowie die komplette Kette mit allen übergeordneten CA-Zertifikaten
und dem Zertifikat der CA. Damit kann bei einer Personalisierung z.B. in einer PKCS#12-Struktur die komplette CA-Kette hinterlegt und ausgeliefert werden.

Weiterhin sind in der Struktur alle Informationen über die erlaubten Namensräume der untergeordneten Registrierungstellen enthalten. Dadurch kann ein Client beim Erzeugen eines Zertifikatantrags einen erlaubten Präfix (für C und O) an den Subject-DN anhängen.

5 Funktionsreferenz der Registrierungsschnittstelle

5.1 Objekt-Informationen abfragen

5.1.1 getCAStatus

Rückgabe	**tns:DFNCERTTypesCAStatus**	**Struktur mit Informationen über neue Elemente**

Liefert Informationen über den Status einer Zertifizierungsstelle. Dazu gehören die Anzahl der neuen Zertifikatanträge und Sperranträge. Diese können z.B. von einem Client in bestimmten Zeitintervallen abgefragt werden, worauf dieser bei neuen Anträgen den Benutzer informieren kann.

5.1.2 getCAInfo

Rückgabe	**tns:DFNCERTTypesCAInfo**	**Struktur mit Informationen über die CA**

Liefert Informationen über die Zertifizierungsstelle. Dazu gehören der Installationsname der CA, der volle Name der CA sowie die komplette Kette mit allen übergeordneten CA-Zertifikaten und dem Zertifikat der CA. Damit kann bei einer Personalisierung z.B. in einer PKCS#12-Struktur die komplette CA-Kette hinterlegt und ausgeliefert werden.

Weiterhin sind in der Struktur alle Informationen über die erlaubten Namensräume der untergeordneten Registrierungstellen enthalten. Dadurch kann ein Client beim Erzeugen eines Zertifikatantrags einen erlaubten Präfix (für C und O) an den Subject-DN anhängen.

5.1.3 searchItems2

Type	xsd:string	Die Art der Einträge, die gesucht werden sollen
Status	xsd:string	Der Status der gesuchten Einträge
Role	xsd:string	Die Rolle der gesuchten Einträge. Darf 'null' sein, dann wird die Suche nicht nach der Rolle eingeschränkt.
DesiredRaID	xsd:int	Die RA-ID, für die Einträge zurückgeliefert werden sollen. Darf 'null' sein, dann werden alle Einträge, für die der angemeldete RA-Operator eine Berechtigung hat, zurückgegeben.
LastSerial	xsd:integer	Seriennummer, ab der die Suche fortgesetzt werden soll. Darf 'null' sein, dann wird bei der größten Seriennummer gestartet.
Limit	xsd:int	Die Anzahl der Einträge die zurückgeliefert werden sollen
Rückgabe	**tns:DFNCERTTypesExtendedObjectInfo[]**	**Liste von Informationsobjekten**

Sucht Informationen über Einträge aus der Datenbank. Diese Funktion kann Informationen über verschiedene Typen suchen, wobei die Suche nach einer gewünschten Rolle bzw. einer bestimmten RA-ID eingeschränkt werden kann, sofern der angemeldete RA-Operator die Berechtigung für die gewünschte RA-ID hat. Die Ergebnisliste enthält nicht die konkreten Einträge, sondern nur ausgewählte Teil-Informationen die für eine Listendarstellung ausreichen.

Der Parameter *Status* bestimmt den Zustand der gesuchten Objekte: Für Type = *request* sowie *Type = crr* sind die Werte *NEW, PENDING, APPROVED, DELETED, ARCHIVED* und für Type = *certificate* die Werte *VALID, REVOKED* möglich.

Für große Ergebnislisten kann über die Parameter *LastSerial* und *Limit* die Position und Größe der Ergebnisliste reguliert werden. Damit kann z.B. eine Liste in mehreren Schritten in einer GUI aufgebaut werden.

Diese Funktion ersetzt searchItems, searchExtendedItems, searchItemsByRole sowie searchItemsForRaID.

5.1.4 searchItems

Type	xsd:string	Die Art der Einträge, die gesucht werden sollen
Status	xsd:string	Der Status der gesuchten Einträge
Offset	xsd:int	Die Position ab der zurückgeliefert werden soll
Limit	xsd:int	Die Anzahl der Einträge die zurückgeliefert werden sollen
Rückgabe	**tns:DFNCERTTypesObjectInfo[]**	**Liste von Informationsobjekten**

Sucht Informationen über Einträge aus der Datenbank. Diese Funktion kann Informationen über verschiedene Typen suchen. Die Ergebnisliste enthält nicht die konkreten Einträge, sondern nur ausgewählte Teil-Informationen die für eine Listendarstellung ausreichen.

Der Parameter *Status* bestimmt den Zustand der gesuchten Objekte: Für Type = *request* sowie *Type = crr* sind die Werte *NEW, PENDING, APPROVED, DELETED, ARCHIVED* und für Type = *certificate* die Werte *VALID, REVOKED* möglich.

Für große Ergebnislisten kann über die Parameter *Offset* und *Limit* die Position und Größe der Ergebnisliste reguliert werden. Damit kann z.B. eine Liste in mehreren Schritten in einer GUI aufgebaut werden.

Diese Funktion sollte nicht mehr verwendet werden. Stattdessen soll searchItems2 verwendet werden.

5.1.5 SearchItemsByRole

Type	xsd:string	Die Art der Einträge, die gesucht werden sollen
Status	xsd:string	Der Status der gesuchten Einträge
Role	Xsd:string	Die Rolle der gesuchten Einträge
Offset	xsd:int	Die Position ab der zurückgeliefert werden soll
Limit	xsd:int	Die Anzahl der Einträge die zurückgeliefert werden sollen
Rückgabe	**tns:DFNCERTTypesExtendedObjectInfo[]**	**Liste von Informationsobjekten**

s. searchItems.
Zusätzlich kann hier nach einer bestimmten Rolle gesucht werden, um z.B. Zertifikate von Teilnehmer-Service-Mitarbeitern zu suchen, und die Ergebnisliste enthält die RA Nummer der ge-

fundenen Einträge sowie die Anzahl der noch nicht bestätigten E-Mail-Adressen.

Diese Funktion sollte nicht mehr verwendet werden. Stattdessen soll searchItems2 verwendet werden.

5.1.6 SearchExtendedItems

Type	xsd:string	Die Art der Einträge, die gesucht werden sollen
Status	xsd:string	Der Status der gesuchten Einträge
Offset	xsd:int	Die Position ab der zurückgeliefert werden soll
Limit	xsd:int	Die Anzahl der Einträge die zurückgeliefert werden sollen
Rückgabe	**tns:DFNCERTTypesExtendedObjectInfo[]**	**Liste von Informationsobjekten**

s. searchItems, wobei hier die Ergebnisliste, die RA Nummer der Objekte sowie die Anzahl der noch nicht bestätigten E-Mail-Adressen enthält.

Diese Funktion sollte nicht mehr verwendet werden. Stattdessen soll searchItems2 verwendet werden.

5.1.7 SearchItemsForRaID

Type	xsd:string	Die Art der Einträge, die gesucht werden sollen
Status	xsd:string	Der Status der gesuchten Einträge
Offset	xsd:int	Die Position ab der zurückgeliefert werden soll
Limit	xsd:int	Die Anzahl der Einträge die zurückgeliefert werden sollen
DesiredRaID	xsd:int	Die RA-ID, für die Einträge zurückgeliefert werden sollen
Rückgabe	**tns:DFNCERTTypesExtendedObjectInfo[]**	**Liste von Informationsobjekten**

s. searchExtendeItems.
Zusätzlich kann die Suche auf eine bestimmte RA-ID eingeschränkt werden, sofern der angemeldete RA-Operator die Berechtigung für die gewünschte RA-ID hat.

Diese Funktion sollte nicht mehr verwendet werden. Stattdessen soll searchItems2 verwendet werden.

5.1.8 getRequestData

Serial	xsd:int	Seriennummer des Zertifikatantrags
Rückgabe	**tns1:DFNCERTTypesRequestData**	**Struktur mit Informationen über den Zertifikatantrag**

Liefert Informationen über den Zertifikatantrag in einer *DFNCERTTypesRequestData*-Struktur.

Diese enthält den PKCS#10-Request sowie weitere Daten, die benötigt werden, um einen neuen Antrag per *newRequest* zu stellen. Enthält der Zertifikatantrag keinen PKCS#10-Request, da der Schlüssel bei der Zertifikatbeantragung im Browser generiert wurde, wird ein Fehler zurückgegeben.

5.2 Zertifikatanträge bearbeiten

5.2.1 approveRequest

Serial	xsd:int	Die Seriennummer des Zertifikatantrags
Content	xsd:base64-Binary	Der signierte Inhalt, Rückgabe von getRawRequest
Signature	xsd:string	PKCS#7 mit Signatur über den Antrag im PEM-Format
Rückgabe	**xsd:boolean**	**Bei Erfolg true**

Genehmigt einen Zertifikatantrag anhand dessen Seriennummer und einer Signatur über den Antrag. Für die Signatur muss als Eingabe der Antrag in genau der Form vorliegen, wie er von *getRawRequest* geliefert wurde. Die Signatur muss als abgetrennte Signatur (detached) in einem PKCS#7-Container im BASE64-kodierten PEM-Format vorliegen. Der Header *-----BEGIN PKCS7-----* muss mit dem entsprechenden Footer vorhanden sein.

5.2.2 deleteRequest

Serial	xsd:int	Die Seriennummer des Zertifikatantrags
Rückgabe	**xsd:boolean**	**Bei Erfolg true**

Löscht einen Zertifikatantrag. Der Antrag wird zunächst nicht endgültig gelöscht, sondern erhält den Status *DELETED*. Nach einer Vorhaltefrist wird der Antrag dann automatisch endgültig gelöscht.

5.2.3 renewRequest

Serial	xsd:int	Die Seriennummer des Zertifikatantrags
Rückgabe	**xsd:int**	**Seriennummer des erneuerten Antrags**

Erneuert einen Zertifikatantrag. Der Antrag mit der Seriennummer *Serial* muss sich dafür entweder in den archivierten oder den gelöschten Anträgen befinden. Der Antrag wird kopiert und unter einer neuen Seriennummer neu gespeichert, die als Rückgabewert übergeben wird. Alle Daten des alten Antrags werden kopiert, bis auf das Ende der im alten Antrag festgelegten Laufzeit. Die Funktion ist für eine Rezertifizierung einsetzbar.

5.2.4 renewRequestSetPublishIfNeeded

Serial	xsd:int	Die Seriennummer des Zertifikatantrags
Rückgabe	**Tns1:DFN-CERTTypes-RenewRequestResult**	**Datenstruktur mit Angaben zum erneuerten Antrag**

Erneuert einen Zertifikatantrag. Der Antrag mit der Seriennummer *Serial* muss sich dafür entweder in den archivierten oder den gelöschten Anträgen befinden. Der Antrag wird kopiert und unter einer neuen Seriennummer neu gespeichert, die innerhalb der zurückgegebenen Daten-

struktur übergeben wird. Alle Daten des alten Antrags werden kopiert, bis auf das Ende der im alten Antrag festgelegten Laufzeit sowie der Wert für die Veröffentlichung des Zertifikats (Publish), sofern der Wert nicht zu den Richtlinien zur Veröffentlichung eines Zertifikats passt. D.h. wurde im bestehenden Zertifikat der Veröffentlichung nicht zugestimmt, aber die Richtlinien zur Veröffentlichung eines Zertifikats erfordern nun die Veröffentlichung, wird im erneuerten Zertifikat der Veröffentlichung zugestimmt. Der Aufrufer muss sich um die Nutzereinwilligung zur Veröffentlichung des Zertifikats kümmern. Die Funktion ist für eine Rezertifizierung einsetzbar.

5.2.5 getRawRequest

Serial	xsd:int	Die Seriennummer des Zertifikatantrags
Rückgabe	**xsd:base64-Binary**	**Der komplette zu signierende Zertifikatantrag**

Liefert die Daten, die von einem Client für die Genehmigung des Antrags zu signieren sind. Zu beachten ist, dass die Rückgabe als Datentyp *xsd:base64Binary* erfolgt, damit die Zeichenkodierung auch von Umlauten in der Kodierung *ISO-8859-1* erhalten bleibt und nicht von der SOAP-Implementierung des Clients automatisch in die lokale Kodierung überführt wird.

5.2.6 getRequestInfo

Serial	xsd:int	Die Seriennummer des Zertifikatantrags
Rückgabe	**tns:DFNCERTTypesRequestInfo**	**Struktur mit Informationen über den Antrag**

Liefert detailierte Informationen über einen Zertifikatantrag in einer *DFNCERTTypesRequest*-Struktur. Diese enthält alle veränderbaren Parameter sowie nicht veränderbaren Informationen über den Antrag. Aufgrund dieser Vermischung besitzt dieser Aufruf einen anderen Parameter als der entsprechende Aufruf zum Setzen der Parameter (*setRequestParameters*).

5.2.7 getExtendedRequestInfo

Serial	xsd:int	Die Seriennummer des Zertifikatantrags
Rückgabe	**tns:DFNCERTTypesExtendedRequestInfo**	**Struktur mit Informationen über den Antrag**

Liefert detailierte Informationen über einen Zertifikatantrag in einer *DFNCERTTypesExtendedRequest*-Struktur. Diese enthält alle veränderbaren Parameter sowie nicht veränderbaren Informationen über den Antrag. Aufgrund dieser Vermischung besitzt dieser Aufruf einen anderen Parameter als der entsprechende Aufruf zum Setzen der Parameter (*setExtendedRequestParameters*).

5.2.8 getRequestPrintout

Serial	xsd:int	Die Seriennummer des Zertifikatantrags
Format	xsd:string	Das gewünschte Format (MIME-Type) des Ausdrucks
Rückgabe	**xsd:base64-Binary**	**Der Ausdruck des Zertifikatantrags**

Liefert einen Ausdruck des Zertifikatantrags mit der Seriennummer *Serial* in der Form, wie dieser auch einem Nutzer beim Beantragen eines Zertifikats präsentiert wird. Dabei kann ein MI-

ME-Type für das gewünschte Format des Rückgabewerts angegeben werden. Momentan wird hierfür ausschließlich *application/pdf* unterstützt.

5.2.9 setRequestParameters

Serial	xsd:int	Die Seriennummer des Zertifikatantrags
RequestParameters	tns:DFNCERTTypesRequestParameters	Eine Struktur mit gewünschten Werten
Rückgabe	**xsd:boolean**	**Bei Erfolg true**

Setzt die veränderbaren Teile eines Antrags anhand einer *DFNCERTTypesRequestParameters* Struktur. Dazu gehören z.B. die Rolle des Zertifikats, der Subject-DN und der Gültigkeitszeitraum. Alle dieser veränderbaren Parameter unterliegen einer Plausibilitätsprüfung. Dies betrifft den Synatx, sowie die Aussagen der Werte (z.B. wird der gewünschte Gültigkeitszeitraum gegen die Laufzeit der CA oder der maximalen Laufzeit der beantragten Rolle geprüft). Falls diese Plausibilitätsprüfung für mindestens einen gewünschtenWert fehlschlägt, wird der Antrag nicht bearbeitet und der Aufruf liefert einen Fehler.

5.2.10 setExtendedRequestParameters

Serial	xsd:int	Die Seriennummer des Zertifikatantrags
RequestParameters	tns:DFNCERTTypesExtendedRequestParameters	Eine Struktur mit gewünschten Werten
Rückgabe	**xsd:boolean**	**Bei Erfolg true**

Setzt die veränderbaren Teile eines Antrags anhand einer *DFNCERTTypesExtendedRequestParameters* Struktur (s. auch setRequestParameters).

5.2.11 sendConfirmationEMail

Serial	xsd:int	Die Seriennummer des Zertifikatantrags
EMails	xsd:string[]	Liste mit E-Mail-Adressen, an die eine Bestätigungs-E-Mail versendet werden soll
Rückgabe	**xsd:boolean**	**Bei Erfolg true**

Sendet für jede E-Mail-Adresse aus der übergebenen Liste eine Bestätigung-E-Mail für den Zertifikatantrag mit der gegebenen Seriennummer. Diese Bestätigungs-E-Mail enthält einen Bestätigungs-Link, mit dem überprüft werden kann, ob die E-Mail-Adresse dem zukünftigen Zertifikatinhaber zugeordnet ist. Bevor der Bestätigungs-Link nicht für alle E-Mail-Adressen aus der übergebenen Liste aufgerufen wurde, kann der Zertifikatantrag nicht genehmigt werden.

5.3 Zertifikatinformationen einholen

5.3.1 getCertificate

Serial	xsd:integer	Die Seriennummer des Zertifikats
Rückgabe	**xsd:string**	**Das gewünschte Zertifikat im PEM-Format**

Liefert ein Zertifikat im PEM-Format aus der Datenbank anhand dessen Seriennummer zurück. Der Header *-----BEGIN CERTIFICATE-----* ist mit dem entsprechenden Footer in der Ausgabe enthalten.

5.3.2 getCertificateByRequestSerial

Serial	xsd:int	Die Seriennummer des Zertifikatantrags
Rückgabe	**xsd:string**	**Das gewünschte Zertifikat im PEM-Format**

Liefert ein Zertifikat aus der Datenbank anhand der Seriennummer des Zertifikatantrags, mit dem das gesuchte Zertifikat beantragt wurde. Diese Funktion ist sinvoll einsetzbar nach der Beantragung eines Zertifikats durch *newRequest*, da dieser Aufruf die Seriennummer des Antrags zurückliefert.

5.3.3 getCertificateInfo

Serial	xsd:integer	Die Seriennummer des Zertifikats
Rückgabe	**tns:DFNCERTTypesCertificateInfo**	**Meta-Informationen über das Zertifikat**

Liefert Meta-Informationen über das Zertifikat mit der Seriennummer Serial. Darin enthalten sind Informationen, die nicht Bestandteil des Zertifikats sind, sondern in der Datenbank der CA gespeichert sind. Dazu gehören: die zugehörige Seriennummer des Antrags, alle Seriennummern von Zertifikaten mit dem gleichen DN, die Einstellung zur Veröffentlichung sowie die Rolle des Zertifikats.

5.4 Verwalten von Sperranträgen

5.4.1 newRevocationRequest

Serial	xsd:integer	Die Seriennummer des Zertifikats
Reason	xsd:string	Der Grund für die Sperrung
Rückgabe	**xsd:int**	**Die Seriennummer des neuen Sperrantrags**

Stellt einen Sperrantrag für ein bestimmtes Zertifikat. Der Grund für die Sperrung kann auch leer sein. In der DFN-PKI wird der Grund für die Sperrung zur Zeit nicht in die Sperrlisten als Erweiterung aufgenommen.

5.4.2 approveRevocationRequest

Serial	xsd:int	Die Seriennummer des Sperrantrags
Content	xsd:base64-Binary	Rückgabewert von getRawRevocationRequest
Signature	xsd:string	PKCS#7 Signatur über den Sperrantrag
Rückgabe	**xsd:boolean**	**Bei Erfolg true**

Äquivalent zum Genehmigen eines Zertifikatantrags durch *approveRequest* genehmigt dieser Aufruf einen Sperrantrag. Der Inhalt des Parameters *Content* wird durch einen Aufruf von *getRawRevocationRequest* geliefert.

5.4.3 getRawRevocationRequest

Serial	xsd:int	Die Seriennummer des Sperrantrags
Rückgabe	**xsd:base64-binary**	**Sperrantrag aus der Datenbank der CA**

Liefert die Daten, wie sie zur Genehmigung eines Sperrantrags durch einen Client zu signieren sind.

5.4.4 getRevocationInfo

Serial	xsd:int	Die Seriennummer des Sperrantrags
Rückgabe	**tns:DFNCERT-TypesRevocationInfo**	

Liefert Meta-Informationen über den Sperrantrag mit der Seriennummer Serial, wie die Seriennummer des zu sperrenden Zertifikats, das Genehmigungsdatum sowie den Status des Sperrantrags.

5.5 Verwalten von erlaubten Domain-Namen

5.5.1 listDomains

RaID	xsd:int	Die RA deren Domain-Namen gelistet werden sollen
Rückgabe	**DFNCERTTypesDomainListResult**	**Alle Domain-Einträge und die Zugriffsrechte der angeforderten RA**

Listet alle erlaubten und aktuell beantragten Domain-Namen der angegebenen RA auf. In der zurückgegebenen Struktur sind alle Domain-Einträge, die aktuellen Zugriffsrechte der RA sowie eine aktuelle Prüfsumme der Liste enthalten.

Das Feld Result enthält eine Liste von Domain-Einträgen. Diese Einträge geben jeweils Aufschluss über den Namen, die Sichtbarkeit auf den Antragsseiten, die Verwendbarkeit in Server- und E-Mail-Domain-Namen, ob diese Domain bereits durch die DFN-PCA freigegeben wurde sowie das Datum der Freigabe.
Das Feld für die Zugriffsrechte ACL enthält eine weitere Struktur, die in dem Feld Allowed eine Liste von erlaubten Aktionen aufführt. Aktuell sind hier die Werte *edit-server* und *edit-email* definiert, die bei Vorhandensein angeben, dass diese RA selbst Server- und E-Mail-Domain-Namen bearbeiten darf. Diese Rechte können nur durch die DFN-PCA bearbeitet werden.

Das Feld Change enthält eine aktuelle Prüfsumme über alle Domain-Einträge. Diese muss bei anderen Aufrufen als Parameter angegeben werden und dient dazu, eine Änderung an der Liste durch eine andere RA zu erkennen. Wird bei einem anderen Aufruf eine nicht aktuelle Prüfsumme angegeben, schlägt der Aufruf fehl. In diesem Fall muss eine Anwendung erst erneut wieder die aktuelle Änderungsprüfsumme durch einen Aufruf von listDomains erhalten und prüfen, ob die angeforderte Aktion nicht mit der aktuellen Liste kollidiert.

5.5.2 listExtendedDomains

RaID	xsd:int	Die RA, deren Domain-Namen gelistet werden sollen
Rückgabe	**DFNCERTTy-**	**Alle Domain-Einträge und die Zugriffsrechte der ange-**

	pesExtended-Domain-ListResult	**forderten RA**

Listet, wie auch listDomains, alle erlaubten und aktuell beantragten Domain-Namen der angegebenen RA auf, wobei noch weitere Informationen wie z.B. die Prüfmethode enthalten sind.

5.5.3 requestDomain

RaID	xsd:int	Die RA in der ein Domain-Name beantragt werden sollen
Name	xsd:string	Beantragter Domain-Name
Type	xsd:string	Typ des Domain-Eintrags (server[-host] oder email[-host])
Public	xsd:boolean	Sichtbar auf den Antragsseiten
Change	xsd:string	Aktuelle Änderungsprüfsumme
Rückgabe	**xsd:string**	**Neue Änderungsprüfsumme**

Beantragt einen neuen Domain-Namen zur Freischaltung durch die DFN-PCA in einer bestimmten RA. Im Parameter Name wird dazu der Domain-Name angegeben. Für Domain-Namen sind hier keine Wildcards (*) erlaubt. Für E-Mail-Domain-Namen (Domain-Name mit Type=email oder Type=email-host), erfolgt keine separate Freischaltung durch die DFN-PCA, sondern diese Namen können sofort in Zertifikatanträgen verwendet werden. Für E-Mail-Adressen mit einem Domain-Namen, der nicht vorher durch requestDomain eingetragen wurde, wird bei Antragstellung eine Bestätigungs-E-Mail versendet. Diese E-Mail enthält einen Bestätigungs-Link, der aufgerufen werden muss, um die Zuordung einer E-Mail-Adresse zum Antragsteller zu bestätigen.

Im Parameter Type sind folgende Werte möglich:

server	Es wird ein Domain-Name für Server beantragt. Es sollen alle Hostnamen inklusive Subdomains vor diesem Namen erlaubt sein.
server-host	Es wird genau ein FQDN für Serverzertifikate erlaubt.
email	Es wird ein Domain-Name für EMail-Adressen beantragt. Es sind alle Adressen vor dieser Domain und beliebige Subdomains zugelassen. Beispiel: Eingetragen wird „dfn.de“. Gültige Adressen sind dann „pki@dfn.de“ sowie „pki@pca.dfn.de“.
email-host	Es wird genau eine Domain für EMail-Adressen beantragt. Es sind beliebige EMail-Adressen aber keine beliebigen Subdomains erlaubt.

Im Parameter Change muss die aktuelle Änderungsprüfsumme übergeben werden. Die erste Änderungsprüfsumme erhält eine Anwendung immer durch einen Aufruf von listDomains. Danach muss die Anwendung immer die aktuell bekannte Änderungsprüfsumme zwischenspeichern und mit den Rückgabewerten der Aufrufe aktualisieren.

5.5.4 deleteDomain

RaID	xsd:int	Die RA des Domain-Namen, der gelöscht werden soll
Name	xsd:string	Name des zu löschenden Domain-Eintrags
Type	xsd:string	Typ des Domain-Eintrags (server[-host] oder email[-host])
Change	xsd:string	Letzte Änderungsprüfsumme

Rückgabe	**xsd:string**	**Aktuelle Änderungsprüfsumme**

Löscht den angegebenen Domain-Eintrag ohne vorherige Prüfung durch die DFN-PCA in der angegebenen RA. Der zu löschende Domain-Eintrag wird dazu über den Namen in Parameter Name und den Typ in Parameter Type (siehe dazu requestDomain) referenziert. Wenn der Eintrag nicht existiert, liefert der Aufruf einen Fehler.

5.5.5 deleteDomain2

RaID	xsd:int	Die RA des Domain-Namen, der gelöscht werden soll
Name	xsd:string	Name des zu löschenden Domain-Eintrags
Type	xsd:string	Typ des Domain-Eintrags (server[-host] oder email[-host])
Change	xsd:string	Letzte Änderungsprüfsumme
Rückgabe	**tns1:DFN-CERTTypes-DeleteDomain2Result**	**Aktuelle Änderungsprüfsumme und ggf. Liste der gültigen Zertifikate zu diesem Domain-Namen.**

Siehe deleteDomain. Allerdings wird hier zunächst überprüft, ob es noch gültige Zertifikate gibt, die den Domain-Namen, der gelöscht werden soll, enthalten. Falls es noch welche gibt, wird der Domain-Eintrag nicht gelöscht und es wird eine Liste der gefundenen Zertifikate zurückgegeben.

5.5.6 getTLDs

Rückgabe	**tns1:DFNCERTTypesTLDs**	**Liste der Top-level-domains**

Gibt die Liste der konfigurierten Top-level-domains zurück.

5.5.7 getCertificatesForDomain

RaID	xsd:int	Die RA des Domain-Namen
Name	xsd:string	Domain-Name
Type	xsd:string	Typ des Domain-Eintrags (server[-host] oder email[-host])
Status	xsd:string	Status der Zertifikate ('VALID' oder 'REVOKED')
Rückgabe	**tns1:ArrayOfDFNCERTTypesShortCertInfo**	**Liste der gültigen Zertifikate zu diesem Domain-Namen.**

Gibt eine Liste mit Zertifikat-Informationen über Zertifikate zurück, die den angefragten Domain-Namen enthalten. Hierbei kann nach gültigen oder revozierten Zertifikaten gesucht werden.

5.5.8 getValidationParameter

Name	xsd:string	Domain-Name
Rückgabe	**tns1:Array-OfDFNCERT-TypesValida-tionParame-ter**	**Liste der möglichen Validierungs-Parameter für die angefragte Domain.**

Gibt zu einem Domain-Namen eine Liste mit den möglichen Validierungs-Parametern zurück.

5.5.9 setValidationParameter

RaID	xsd:int	RA-ID des Domain-Eintrags
Name	xsd:string	Domain-Name
Type	xsd:string	Typ (server, server-host)
Method	xsd:string	Prüfmethode (2-Domain-Contact-Mail-SOA, 2-Domain-Contact-Mail-Whois, 4-Constructed-Mail)
EmailLocal	xsd:string	Lokaler Part der E-Mail-Adresse, an die eine Challenge-E-Mail versendet werden soll.
EmailDomain	xsd:string	Domain Part der E-Mail-Adresse, an die eine Challenge-E-Mail versendet werden soll.
Change	xsd:string	Letzte Änderungsprüfsumme
Rückgabe	**xsd:string**	**Aktuelle Änderungsprüfsumme**

Die Prüfmethode gibt an, nach welchem Verfahren der angefragte Domain-Name validiert werden soll. Bei der Methode „2-Domain-Contact-Mail-SOA" wird eine Challenge-E-Mail an den Zonenkontakt der Domain gesendet. Bei der Methode „2-Domain-Contact-Mail-Whois" wird eine Challenge-E-Mail an eine Kontakt-Adresse, die im whois hinterlegt ist, gesendet. Bei der Methode „4-Constructed-Mail" eine Challenge-E-Mail an eine E-Mail-Adresse, die wie folgt konstruiert wird: Der lokale Part der E-Mail-Adresse darf „admin", „hostmaster", „webmaster", „administrator" oder „postmaster" sein, der domain Part ist ein zur Domain passender Authorization Domain Name. Ein Authorization Domain Name ist dabei einer der Namen zwischen der angefragten und der Base Domain (Domain, die bei einer Registry eingetragen ist).

5.5.10 sendChallengeEMail

RaID	xsd:int	RA-ID des Domain-Eintrags
Name	xsd:string	Domain-Name
Type	xsd:string	Typ (server, server-host)
Change	xsd:string	Letzte Änderungsprüfsumme
Rückgabe	**tns1:DFN-CERTTypes-SendChallen-geEMailResult**	**Struktur, die die aktualle Änderungsprüfsumme sowie das Datum, an dem die Challenge-E-Mail versendet wurde, enthält.**

6 Datenstrukturenreferenz

6.1 Datenstrukturen für RA-Informationen

6.1.1 DFNCERTTypesCAStatus

RequestNew-Count	xsd:int	Anzahl der neuen Zertifikatanträge
RevocationNe-wCount	xsd:int	Anzahl der neuen Sperranträge

6.1.2 DFNCERTypesCAInfo

RALoginID	xsd:int	RA_ID des Clients nach Authentifizierung
RAInfos	tns:DFNCERT TypesRAInfo[]	Liste mit Informationen über alle zu dieser CA gehörenden Registrierungsstellen
CAChain	xsd:string[]	Das aktuelle CA-Zertifikat und Kette im PEM-Format
Roles	xsd:string[]	Liste mit allen von dieser CA unterstützten Rollen-Namen

6.1.3 DFNCERTypesRAInfo

ID	xsd:int	RA-Nummer des Eintrags
Name	xsd:string	Installationsname der CA
DNPrefixes	xsd:string[]	Liste mit allen erlaubten Namensräumen

6.2 Datenstrukturen für Objekt-Informationen

6.2.1 DFNCERTTypesCertificateInfo

RequestSerial	xsd:int	Seriennummer des passenden Requests zu diesem Zertifikat
Publish	xsd:boolean	Wurde das Zertifikat veröffentlicht?
Role	xsd:string	Die Rolle des Zertifikats in der CA
Status	xsd:string	Der Status des Zertifikats (VALID oder REVOKED)
PEM	Xsd:string	Das Zertifikat im PEM-Format

6.2.2 DFNCERTTypesShortCertInfo

RaID	xsd:int	RA-Nummer des Zertifikats
Serial	xsd:integer	Seriennummer des Zertifikats
SubjectDN	xsd:String	Der Subject-DN des Zertifikats
NotAfter	xsd:dateTime	Das Ablaufdatum des Zertifikats („gültig bis", Zeitzone:

		UTC)

6.2.3 DFNCERTTypesObjectInfo

Serial	xsd:integer	Seriennummer des Eintrags
Subject	xsd:string	Subject-DN des Eintrags
EMail	xsd:string	E-Mail Adresse entweder aus Subject-DN, wenn dort nicht vorhanden aus Additional Email
Role	xsd:string	Beantragte Rolle des Eintrags
Date	xsd:dateTime	Für Anträge Datum des Eingangs und für Zertifikate NotAfter (entspricht "gültig bis", Zeitzone: UTC)

6.2.4 DFNCERTTypesExtendedObjectInfo

Serial	xsd:integer	Seriennummer des Eintrags
Subject	xsd:string	Subject-DN des Eintrags
Email	xsd:string	E-Mail Adresse entweder aus Subject-DN, wenn dort nicht vorhanden aus Additional Email
Role	xsd:string	Beantragte Rolle des Eintrags
Date	xsd:dateTime	Für Anträge Datum des Eingangs und für Zertifikate NotAfter (entspricht "gültig bis", Zeitzone: UTC)
UnconfirmedEMails	xsd:int	Anzahl der noch nicht bestätigten E-Mail-Adressen
RaID	xsd:int	RA Nummer des Eintrags
AdditionalName	xsd:string	Name des Antragsstellers
AdditionalEMail	xsd:string	Kontakt-E-Mail-Adresse des Antragsstellers
AdditionalOrgUnit	xsd:string	Abteilung des Antragsstellers
NotBefore	xsd:dateTime	Gültigkeitsbeginn des Zertifikats (Zeitzone: UTC)

6.3 Datenstrukturen für Informationen über Zertifikatanträge

6.3.1 DFNCERTTypesRequestParameters

RaID	xsd:int	RA Nummer des Antrags
Subject	xsd:string	Subject DN des Antrags
SubjectAltNames	xsd:string[]	Subject Alternative Names als Array von Strings. Format ist: ("typ:wert", …)
Role	xsd:string	Die Rolle des beantragten Zertifikats
NotBefore	xsd:dateTime	Gültigkeitsbeginn des Zertifikats (Zeitzone: UTC)
NotAfter	xsd:dateTime	Gültigkeitsende des Zertifikats (Zeitzone: UTC)

AdditionalName	xsd:string	Name des Antragstellers
AdditionalEMail	xsd:string	E-Mail-Adresse des Antragstellers
AdditionalUnit	xsd:string	Abteilung des Antragstellers

6.3.2 DFNCERTTypesExtendedRequestParameters

RaID	xsd:int	RA Nummer des Antrags
Subject	xsd:string	Subject DN des Antrags
SubjectAltNa-mes	xsd:string[]	Subject Alternative Names als Array von Strings. Format ist: ("typ:wert", …)
Role	xsd:string	Die Rolle des beantragten Zertifikats
NotBefore	xsd:dateTime	Gültigkeitsbeginn des Zertifikats (Zeitzone: UTC)
NotAfter	xsd:dateTime	Gültigkeitsende des Zertifikats (Zeitzone: UTC)
AdditionalName	xsd:string	Name des Antragstellers
AdditionalEMail	xsd:string	Kontakt-E-Mail Adresse des Antragstellers
AdditionalUnit	xsd:string	Abteilung des Antragstellers
ValidityDays	xsd:int	Gültigkeitsdauer des Zertifikats in Tagen
EmailAddresses	tns:Email[]	Liste mit allen E-Mail-Adressen des gegebenen Antrags, die in das Zertifikat aufgenommen werden sollen (für Anträge, die nach dem 1.7.2014 gestellt wurden)

6.3.3 Email

local	xsd:string	Lokaler-Part der E-Mail-Adresse
domain	xsd:string	Domain-Part der E-Mail-Adresse
state	EMailState	Status des Eintrags
requestSerial	xsd:int	Antragsnummer des zugehörigen Zertifikatantrags
location	EMailLocation	Ort, an dem die E-Mail-Adresse im Zertifikat steht
lastSendDate	xsd:dateTime	Zeitpunkt, an dem die letzte Bestätigungs-E-Mail versendet wurde (Zeitzone: UTC)
stateChange-Date	xsd:dateTime	Zeitpunkt, an dem der State verändert wurde. (Zeitzone: UTC)

Der Parameter EMailState ist vom Typ xsd:string und kann die folgenden Werte annehmen:

PENDING	Bestätigung für die E-Mail-Adresse steht noch aus
REJECTED	E-Mail-Adresse wurde zurückgewiesen
CONFIRMED	E-Mail-Adresse wurde durch den Nutzer bestätigt
WHITELISTED	E-Mail-Adresse wurde durch die Whitelist bestätigt

Der Parameter EMailLocation ist vom Typ xsd:string und kann die folgenden Werte annehmen:

DN	E-Mail-Adresse befindet sich im DN
SAN	E-Mail-Adresse befindet sich im Subject-AltName
DN_AND_SAN	E-Mail-Adresse sowohl im DN als auch im Subject-AltName

6.3.4 DFNCERTTypesRequestInfo

Serial	xsd:int	Seriennummer des Antrags
SameDNSerials	xsd:int[]	Liste mit Seriennummern von Zertifikaten, die den gleichen Subject DN tragen
Status	xsd:string	Status des Eintrags (NEW, PENDING, RENEW, APPROVED, DELETED, ARCHIVED)
Parameters	tns:DFNCERT RequestPara-meters	Struktur mit allen veränderbaren Parametern eines Zertifikatantrags
PublicKey	xsd:string	Der öffentliche Schlüssel des Antrags in OpenSSL-Ausgabeformat
PublicKeyAlgo-rithm	xsd:string	Der verwendete Algorithmus des Schlüssels
PublicKeyDi-gest	xsd:string	Ein SHA-1 über den öffentlichen Schlüssel
PublicKey-Length	xsd:int	Die Länge des öffentlichen Schlüssels in Bit
Publish	xsd:boolean	Flagge für die Veröffentlichung des Antrags
SignatureAlgo-rithm	xsd:string	Der verwendete Algorithmus bei der Signatur des Antrags
DateSubmitted	xsd:string	Datum an dem der Antrag einging (Zeitzone: UTC)
DateApproved	xsd:string	Das Datum der Genehmigung des Antrags (Zeitzone: UTC)
DateDeleted	xsd:string	Das Datum der Löschung des Antrags (Zeitzone: UTC)

Die Werte für 'Status' haben folgende Bedeutung:

NEW:
Ein Zertifikatantrag (Request) ist im Zustand NEW, wenn er initial neu erzeugt worden ist.
RENEW:
Ein Request, der von einem achivierten Request abgeleitet worden ist ("Kopie").
PENDING:
Ein neuer oder erneuerter Request, der von einem CAO1 verändert worden ist.
DELETED:
Ein gelöschter Request. Von diesem wurde kein Zertifikat erzeugt.
APPROVED:
Ein freigegebener Request. Aus diesem Request basierend soll nun ein Zertifikat erzeugt werden.
ARCHIVED:
Ein Request, von dem ein Zertifikat erzeugt worden ist.

6.3.5 DFNCERTTypesExtendedRequestInfo

Serial	xsd:int	Seriennummer des Antrags
SameDNSerials	xsd:int[]	Liste mit Seriennummern von Zertifikaten, die den gleichen Subject DN tragen
Status	xsd:string	Status des Eintrags (s. DFNCERTTypesRequestInfo)
Parameters	tns:DFNCERT ExtendedRe-	Struktur mit allen veränderbaren Parametern eines Zertifikatantrags

	questParame-ters	
PublicKey	xsd:string	Der öffentliche Schlüssel des Antrags in OpenSSL-Ausgabeformat
PublicKeyAlgo-rithm	xsd:string	Der verwendete Algorithmus des Schlüssels
PublicKeyDi-gest	xsd:string	Ein SHA-1 über den öffentlichen Schlüssel
PublicKey-Length	xsd:int	Die Länge des öffentlichen Schlüssels in Bit
Publish	xsd:boolean	Flagge für die Veröffentlichung des Antrags
SignatureAlgo-rithm	xsd:string	Der verwendete Algorithmus bei der Signatur des Antrags
DateSubmitted	xsd:string	Datum an dem der Antrag einging
DateApproved	xsd:string	Das Datum der Genehmigung des Antrags
DateDeleted	xsd:string	Das Datum der Löschung des Antrags
SignerCertifica-teSerial	xsd:integer	Seriennummer des Unterzeichner-Zertifikats
SignerCN	xsd:string	Subject-CN des Unterzeichner-Zertifikats

6.3.6 DFNCERTTypesRequestData

Serial	xsd:int	Seriennummer des Antrags
RaID	xsd:int	Nummer der RA, 0 für die Master-RA
PKCS10	xsd:string	Der Zertifikatantrag im PEM-Format
AltNames	tns1:ArrayOf-String	Subject Alternative Names in der Form ("typ:wert", …)
Role	xsd:string	Die Rolle des beantragten Zertifikats
AddName	xsd:string	Vollständiger Name des Antragstellers
AddEMail	xsd:string	E-Mail Adresse des Antragstellers
AddOrgUnit	xsd:string	Abteilung des Antragstellers
Publish	xsd:boolean	Veröffentlichung des Zertifikats

6.3.7 DFNCERTTypesRenewRequestResult

Serial	xsd:int	Seriennummer des Antrags
Server	xsd:boolean	Flag, das angibt, ob es sich um einen Antrag für ein Server-Zertifikat handelt.
Publish	xsd:boolean	Veröffentlichung des Zertifikats
HasChanged	xsd:boolean	Flag, das angibt, ob der Wert von Publish beim erneuerten Antrag geändert wurde.

6.4 Datenstrukturen für Sperranträge

6.4.1 DFNCERTTypesRevocationParameters

Reason	xsd:string	Grund der Sperrung

6.4.2 DFNCERTTypesRevocationInfo

Status	xsd:string	Status des Eintrags
Serial	xsd:int	Seriennummer des Eintrags
CertificateSerial	xsd:integer	Seriennummer des zu sperrenden Zertifikats
RaID	xsd:int	RA-Nummer des Eintrags
DateSubmitted	xsd:string	Datum an dem der Antrag einging (Zeitzone: UTC)
DateApproved	xsd:string	Das Datum der Genehmigung des Antrags (Zeitzone: UTC)
DateDeleted	xsd:string	Das Datum der Löschung des Antrags (Zeitzone: UTC)
Parameters	tns1:DFNCERTTypesRevocationParameters	Parameter für einen Sperrantrag (Grund der Sperrung)

6.5 Datenstrukturen für Domain-Verwaltung

6.5.1 DFNCERTTypesDomain

Name	xsd:string	Domain-Name
Type	xsd:string	Typ (server, server-host, email oder email-host)
Secret	xsd:boolean	Versteckt vor der Öffentlichkeit
Approved	xsd:boolean	Freigegeben
ApprovedDate	xsd:dateTime	Freigabezeitpunkt (Zeitzone: UTC)

6.5.2 DFNCERTTypesExtendedDomain

Name	xsd:string	Domain-Name
Type	xsd:string	Typ (server, server-host, email oder email-host)
Secret	xsd:boolean	Versteckt vor der Öffentlichkeit
Approved	xsd:boolean	Freigegeben
ApprovedDate	xsd:dateTime	Freigabezeitpunkt (Zeitzone: UTC)
Method	xsd:string	Prüfmethode, nach der die Domain validiert wurde/werden soll
BRVersion	xsd:string	Versionsnummer der Baseline-Requirements, auf die sich die Prüfmethode bezieht
Challenge-MailAddress	xsd:string	E-Mail-Adresse, an die die Challenge-E-Mail versendet wird/wurde
LastChallenge-MailSent	xsd:dateTime	Datum, an dem die letzte Challenge-E-Mail versendet wurde (Zeitzone: UTC)
ValidUntil	xsd:dateTime	Gültigkeitsende (Zeitzone: UTC)

6.5.3 DFNCERTTypesDomainACL

RaID	xsd:int	RA_ID für die diese Liste gilt
Allowed	xsd:string[]	Liste erlaubter Aktionen (Whitelist)

6.5.4 DFNCERTTypesDomainListResult

Change	xsd:string	Aktuelle Änderungsprüfsumme
Result	tns1:Array-OfDFNCERT TypesDomain	Liste mit gefundenen Domain-Einträgen
ACL	DFNCERT TypesDomain ACL	Zugriffsrechte für angeforderte RA_ID

6.5.5 DFNCERTTypesExtendedDomainListResult

Change	xsd:string	Aktuelle Änderungsprüfsumme
Result	tns1:Array-OfDFNCERT TypesExten-dedDomain	Liste mit gefundenen Domain-Einträgen
ACL	DFNCERT TypesDomain ACL	Zugriffsrechte für angeforderte RA_ID

6.5.6 DFNCERTTypesDeleteDomain2Result

Change	xsd:string	Aktuelle Änderungsprüfsumme
CertInfos	tns1:Array-OfDFNCERTTy-pesShortCer-tInfo	Liste mit gefundenen Zertifikat-Einträgen

6.5.7 DFNCERTTypesTLDs

TLDs	xsd:string[]	Liste mit den Top-level-domains

6.5.8 DFNCERTTypesValidDomain

Name	xsd:string	Domain-Name
Type	xsd:string	Typ (server, server-host, email oder email-host)

6.5.9 DFNCERTTypesValidationParameter

Name	xsd:string	Domain-Name
Method	xsd:string	Prüfmethode (2-Domain-Contact-Mail-SOA oder 4-Construc-

		ted-Mail)
Email	xsd:string	E-Mail-Adresse, die zur Prüfmethode passt.
ADNs	tns1:ArrayOf-String	Liste der zur Prüfmethode passenden Authorization Domain Names

Bzgl. Prüfmethoden siehe setValidationParameter.

6.5.10 DFNCERTTypesSendChallengeEMailResult

Change	xsd:string	Aktuelle Änderungsprüfsumme
LastChallenge-EMailSent	xsd:dateTime	Datum, an dem die Challenge-E-Mail gesendet wurde. (Zeit-zone: UTC)