



Project **TeleTrusT**

Date
26.06.2009

Author(s)
Haralambie Dumistracel,
Carsten Zimmermann,
Carolin Paulheim

Distribution
mtG, TeleTrusT

Title **Reference PKI - API Specification**

Type **GEN / General Document**

Name **Reference PKI - API Specification**

Version **0.6**

Security **Public**

API Specification for the
reference PKI.

TABLE OF CONTENTS

1	Reference PKI – API Specification	5
1.1	Introduction.....	5
1.1.1	Service-oriented PKI.....	5
1.1.2	PKI Entities	5
1.1.3	Certificate profiles	6
1.1.4	Communication protocol.....	6
1.1.5	Authentication & Authorization.....	6
1.1.6	Multi-client capability	7
1.1.7	Events	7
1.1.8	Statistic.....	7
1.2	API design.....	8
1.2.1	Error handling	8
1.2.2	Search functions.....	8
1.3	End-Entity functions.....	10
1.3.1	requestCertificate (send a certificate request)	10
1.3.2	retrieveAccessControlGroups (multi client capability).....	12
1.3.3	retrieveCertificateProfiles (certificate definitions).....	13
1.3.4	retrieveRequests (check request status)	14
1.3.5	retrieveCertificates (download certificates)	16
1.3.6	revokeCertificates (certificate revocation).....	19
1.3.7	retrieveCaCertificates (provide CA certificate hierarchies)	20
1.3.8	retrieveCRLs (provide certificate revocation lists).....	21
1.4	Registration-Authority functions.....	23
1.4.1	processCertificateRequest (edit/approve/reject/postpone/activate).....	23
1.4.2	unlockCertificates (undo temporary revocation)	23
1.4.3	generateCRLs (certificate revocation lists generation)	24
1.4.4	retrieveStatistic (retrieve statistic data).....	24
1.5	API – administration	27
1.6	Formats.....	28
	REFERENCES.....	30
	ATTACHMENTS.....	30
	VERSIONSINFORMATIONS	30

INDEX OF TABLES

Table 1 Search filter parameters.....	9
Table 2 Search result container fields.....	9
Table 3 Parameter details for the requestCertificate function - CertificateRequest.....	11
Table 4 Example for the requestCertificate function.....	12
Table 5 Parameter description for the retrieveAccessControlGroups function - ACGSearchFilter.....	13
Table 6 Return value details for the retrieveAccessControlGroups function - AccessControlGroup.....	13
Table 7 Return value details for the retrieveAccessControlGroups function - CertificateProfil.....	14
Table 8 Parameter description for the retrieveRequests function - RequestSearchFilter.....	15
Table 9 Return value details for the retrieveRequests function - CertificateRequest.....	15
Table 10 Example for the retrieveRequests function.....	16
Table 11 Parameter description for the retrieveCertificates function - CertificateSearchFilter.....	17
Table 12 Parameter description for the retrieveCertificates function - DownloadFormat.....	17
Table 13 Return value details for the retrieveCertificates function - Certificate.....	18
Table 14 Example for the retrieveCertificates function.....	18
Table 15 Parameter description for the revokeCertificates function - CertificateSearchFilter.....	19
Table 16 Parameter description for the revokeCertificates function - RevocationData.....	19
Table 17 Example for the revokeCertificates function.....	20
Table 18 Parameter description for the retrieveCaCertificates function - CaCertificateFilter.....	20
Table 19 Return value details for the retrieveCaCertificates function - CaCertificate.....	21
Table 20 Parameter description for the retrieveCRLs function - CRLSearchFilter.....	22
Table 21 Return value details for the retrieveCRLs function - CRL.....	22
Table 22 Return value details for the processCertificateRequest function - PKIResponse.....	23
Table 23 Parameter description for the unlockCertificates function - CertificateSearchFilter.....	24
Table 24 Parameter description for the retrieveStatistic function - StatisticQuery.....	25
Table 25 Return value details for the retrieveStatistic function - StatisticData.....	26
Table 26: Error codes.....	28
Table 27: Certificate request status.....	28
Table 28: Access control group status.....	28
Table 29: Certificate download formats.....	28
Table 30: Certificate status.....	29
Table 31: CRL types.....	29
Table 32: Revocation reasons.....	29
Table 33: Certificate request data type.....	29

LIST OF FIGURES

Figure 1 PKI Entities.....	5
Figure 2 Certificate profile.....	6
Figure 3 Access control group hierarchy.....	7
Figure 4 Error handling - return objects hierarchy.....	8
Figure 5 Search filter and result container.....	8
Figure 6 Parameter diagram for the requestCertificate function.....	10
Figure 7 Return value diagram for the requestCertificate function.....	11
Figure 8 Parameter diagram for the retrieveAccessControlGroups function.....	12

Figure 9 Return value diagram for the retrieveAccessControlGroups function	13
Figure 10 Return value diagram for the retrieveAccessControlGroups function.....	14
Figure 11 Parameter diagram for the retrieveRequests function.....	14
Figure 12 Return value diagram for the retrieveRequests function.....	15
Figure 13 Parameter diagram for the retrieveCertificates function	16
Figure 14 Return value diagram for the retrieveCertificates function.....	18
Figure 15: Parameter diagram for the revokeCertificates function	19
Figure 16 Parameter diagram for the retrieveCaCertificates function.....	20
Figure 17 Return value diagram for the retrieveCaCertificates function.....	21
Figure 18 Parameter diagram for the retrieveCRLs function.....	21
Figure 19 Return value diagram for the retrieveCRLs function	22
Figure 20: Return value diagram for the processCertificateRequest function.....	23
Figure 21: Parameter diagram for the unlockCertificates function.....	24
Figure 22 Parameter diagram for the retrieveStatistic function	25
Figure 23 Return value diagram for the retrieveStatistic function	25

1 Reference PKI - API Specification

1.1 Introduction

Due to complex processes and interfaces, PKI systems have suffered from relatively low adoption rates. An implementation of a PKI environment is faced with a lot of effort and expenses as well as a high need for skilled personnel.

It is therefore the main goal of the "Reference PKI" to drastically simplify the organisational overhead (processes), to reduce the complexity of the interfaces and to improve the user experience. The effort to implement a PKI environment should be reduced to a minimum and a reduced complexity should allow even inexperienced users to use PKI security services. The goal is to seamlessly integrate PKI processes into application processes.

The following subchapters of the introduction explain basic concepts and patterns which are reflected in the API specification.

1.1.1 Service-oriented PKI

The security functionality offered by a PKI should be offered in a service-oriented manner, to allow provisioning and management of the PKI across web services and service oriented architectures.

The PKI services are to hide complex protocols and implementations and to make the security processes transparent to the user.

1.1.2 PKI Entities

The following figure shows the various entities which are involved in PKI based applications. Such applications can use certificates for secure authentication of their participating end entities, as well as for realization of encryption or digital signature functionality.

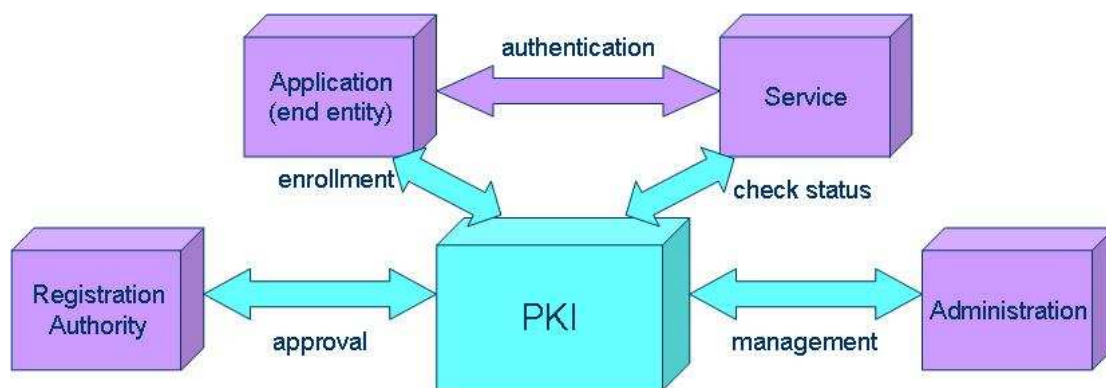


Figure 1 PKI Entities

Application (end-entity)

Clients of the PKI. Have access to the following functions: request, retrieve, renew, revoke and activate certificates (see section 1.3).

Service

Also a client of the PKI: Accesses the following functions: verifies the status of certificates provided by end-entities during the application process.

Registration Authority

Optional mediator between the Certificate Authority and the end-entity – manages end-entity certificate requests and end-entity certificates; accesses the following functions: approval, revocation and renewal (see section 1.4).

Administration

Manages the PKI/CA itself; issues and manages signers and CA certificates etc (see section 1.5).

Certificate Authority

The PKI itself; issues the certificates.

1.1.3 Certificate profiles

Public key certificates are electronic documents which incorporate a digital signature to bind an identity (e.g. a person) to a public key in a secure manner. The signature is the signature of a “Certification Authority” (CA), the entity which issues certificates. A certificate is then used e.g. to verify that a digital signature was created by an individual.

Public key certificates can be built using various format standards: X.509, CV, etc. These formats can become very complex and therefore the format details should be made transparent to the user. To achieve this, the API specification uses the concept of so-called “certificate profiles”.

A “certificate profile” is defined by the Certificate Authority which signs the certificates and an XML format definition describing the certificate structure and default values for certain certificate entries. When a certificate is requested via the API, a certificate profile is referenced by a corresponding identifier. The requesting application has only to deal with a minimum of information which is necessary for the application.

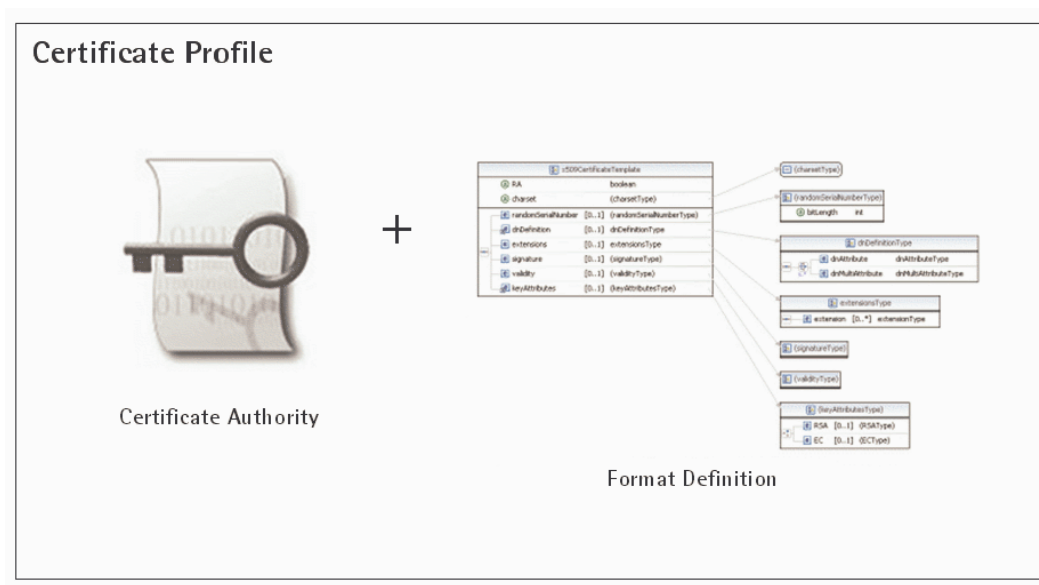


Figure 2 Certificate profile

1.1.4 Communication protocol

Service oriented architectures are commonly built using web service standards. The reference PKI will publish its interfaces as SOAP Web services.

1.1.5 Authentication & Authorization

The SOAP Web services will be published over HTTPS (minimum version: SSL3).

The PKI should provide means to implement the following two levels of authentication and authorization:

- application level – at this level the PKI authenticates and authorises applications which communicate with the PKI via the API. Examples for such applications are web front ends used by RA administrators, or web front ends provided for end entities, applications which realize a revocation hotline, and so on.
- user level – at this level the PKI authenticates and authorises users via certificates. This level is a mandatory requirement for users of access-controlled PKI functionality (e.g. RA or administrative functions). For end-entity or service users, this authentication is optional.

Both application and user credentials are to be included in every call made to a PKI service.

Application security shall be implemented using WS-Security (WS-I Basic Security Profile).

User security shall be implemented by using SSL client authentication (every service call will then authenticate and authorise the certificate provided by a user gaining access to access-controlled functions).

1.1.6 Multi-client capability

The multi-client capability refers to the ability of a PKI system to enable hierarchically organized groups provided with separately defined access rights to resources and management of certificates. The highest possible access control group is defined by the first level of authentication and authorization, namely the application. Each access control group can contain other access control groups, creating the possibility to fine-grain access to resources and certificates.

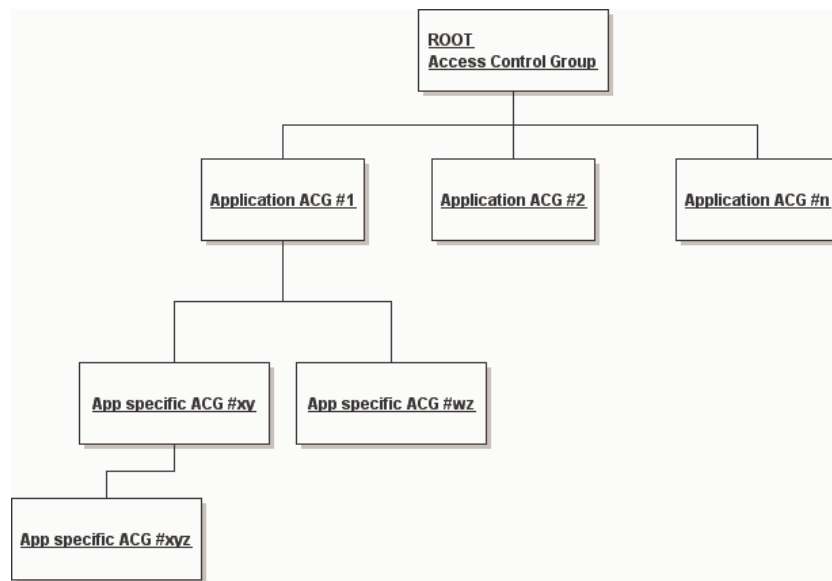


Figure 3 Access control group hierarchy

The access control group hierarchy is not to be confused with certificate hierarchies. By using access control groups, clients of the PKI have the option to organise and granulate their resources and certificates for better administration and access control.

1.1.7 Events

The event handling mechanism describes a configurable response of the system on defined events. The events and responses act as a rule of conduct. Both are defined by an administrator of the PKI. The response is performed when the event is triggered.

An event could be a request, a revocation or an approval of a certificate; a response could be the sending of a notification email or the publishing of a certificate at a LDAP server and so forth.

A user can opt to register for all notifications related to a request (and its subsequent certificates) by providing an email address upon sending the certificate request to the PKI.

1.1.8 Statistic

Detailed statistic information regarding the certificates managed by the PKI should be provided. The statistic data can be queried for a specific point in time or for a specific time period. The statistic functions will provide numbers for certificate lifecycle events (approve, revoke, etc.) along with numbers for specific certificate stati (active, valid, expired, revoked).

Statistic functionality is strongly connected with the multi client capability. The statistic data can be retrieved only by users belonging to certain access control groups.

1.2 API design

The API is use case oriented – this is the most important requirement for the PKI services: to match exact users' needs and to avoid imposing complex processes. The “PKI Reference”-API is built based on requirements created by concrete, real life usage scenarios.

All service functions use objects as parameters and return values.

The PKI services are stateless – the PKI will not save states between two consecutive calls.

1.2.1 Error handling

All return objects for the PKI service functions contain an error object, which signals if an error occurred. It also contains detailed information about the error.

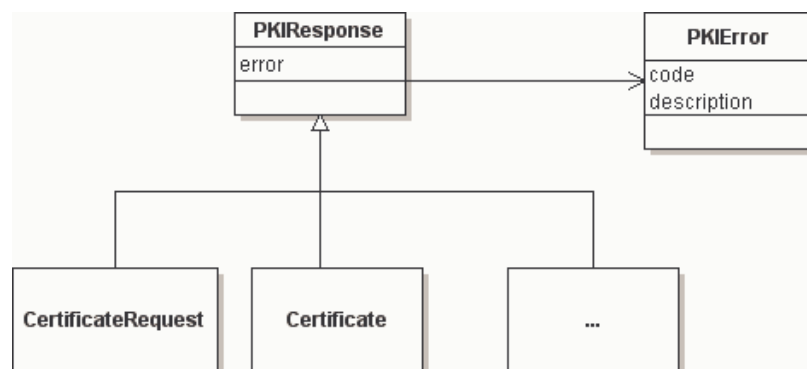


Figure 4 Error handling – return objects hierarchy

1.2.2 Search functions

The PKI service offers a range of search methods, which all share the same characteristics: they all have a filter parameter and return a container object containing a list of found objects.

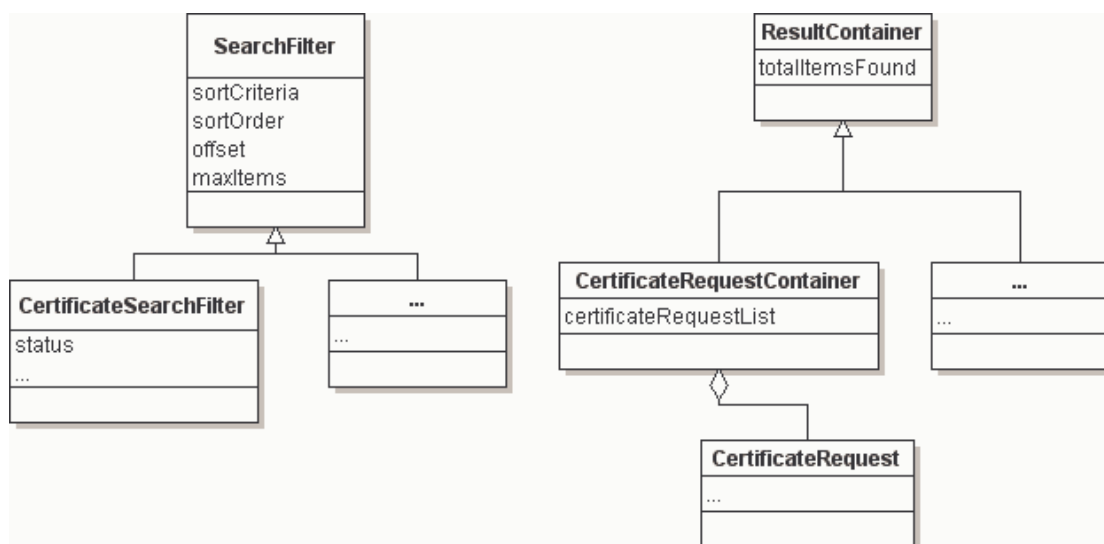


Figure 5 Search filter and result container

The filter parameter contains besides the actual search criteria, a few common search attributes like “sort order”, “sort criteria”, “offset” and “maximum number of results to return”.

Name	Type	Description
sortCriteria	String	Specifies criteria for sorting.
sortOrder	String	Specifies sort order: "asc" for ascending order and "desc" for descending order.
offset	Integer	Defines the start index for the results – only objects with a greater index in the result list will be returned.
maxItems	Integer	Defines the maximum size of the result list. This parameter limits the number of results.

Table 1 Search filter parameters

Name	Type	Description
totalItemsFound	Integer	Returns the total number of items found (not the size of the result list).

Table 2 Search result container fields

1.3 End-Entity functions

1.3.1 requestCertificate (send a certificate request)

REQUIREMENTS

- transparent selection of certificate profiles
- request one or more certificates simultaneously (synchronous certificate management)
- request X.509 and/or CV certificates
- use key pairs generated by the client
- generate key pairs for the client (with key backup)
- allow various formats for the public key (PKCS#10, SPKAC, plain key)
- allow users to specify validity
- request renewal of existing certificates

PARAMETERS

- CertificateRequest - object encapsulating data relevant for this function:

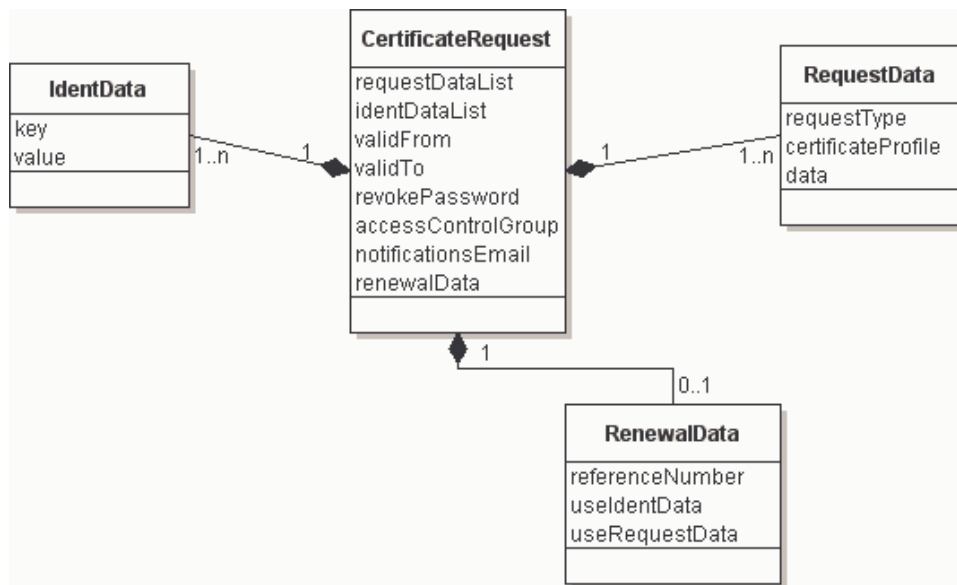


Figure 6 Parameter diagram for the requestCertificate function

Name	Type	Required	Description
requestDataList	Array	Yes	Array containing data required for building certificates. For each entry in the array a certificate will be built.
.requestType	Integer	Yes	Type of request. See Table 33 for possible values.
.certificateProfile	Integer	Yes	ID of the desired certificate profile. See sections 1.1.3 and 1.3.3 for more details.
.data	String	Yes	Depending on the type of request can contain the public key. See Table 33 for details.
identDataList	Array	No	Optional parameter with identification information for this certificate request. Consists of key-value pairs and can be used to provide values to be used when building the certificate (as defined by the certificate profile). See section 1.3.3.
.key	String	Yes	The key to for an identification value.
.value	String	Yes	The identification value.
revokePassword	String	Yes	Revoke password to be queried on certificate revocation request.
validFrom	Date	No	Optional valid from date.
validTo	Date	No	Optional valid to date.
accessControlGroup	Integer	No	Optional assignment to an access control group. See sections 1.1.6 and 1.3.2 for more details.
notificationsEmail	String	No	Email address to be used for sending notifications. See section 1.1.7 for more details.
renewalData	Object	No	Data for a renewal request.
.referenceNumber	Long	Yes	Reference number of the request to be renewed.
.useIdentData	Boolean	No	Use original identification data of the request to be renewed.
.useRequestData	Boolean	No	Use original request data of the request to be renewed.

Table 3 Parameter details for the requestCertificate function - CertificateRequest

RETURN VALUE

- CertificateRequest - object containing only a reference number for the inserted request

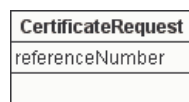


Figure 7 Return value diagram for the requestCertificate function

EXAMPLE

```

<ns1:requestCertificate>
  <request xsi:type="ns2:CertificateRequest" >
    <identDataList soapenc:arrayType="ns2:IdentData[3]" xsi:type="soapenc:Array">
      <identDataList xsi:type="ns2:IdentData">
        <key xsi:type="soapenc:string">id.commonname</key>
        <value xsi:type="soapenc:string">M. Mustermann</value>
      </identDataList>
      <identDataList xsi:type="ns2:IdentData">
        <key xsi:type="soapenc:string">id.country</key>
        <value xsi:type="soapenc:string">DE</value>
      </identDataList>
      <identDataList xsi:type="ns2:IdentData">
        <key xsi:type="soapenc:string">id.email</key>
        <value xsi:type="soapenc:string">max@mustermann.de</value>
      </identDataList>
    </identDataList>
    <requestDataList soapenc:arrayType="ns2:RequestData[1]" xsi:type="soapenc:Array">
      <requestDataList xsi:type="ns2:RequestData">
        <certificateProfile xsi:type="soapenc:long">341</certificateProfile>
        <data xsi:type="soapenc:string">[BASE64 encoded public key]</data>
        <requestType xsi:type="soapenc:long">1</requestType>
      </requestDataList>
    </requestDataList>
    <revokePassword xsi:type="soapenc:string">jdka381x</revokePassword>
  </request>
</ns1:requestCertificate>

```

Table 4 Example for the requestCertificate function

1.3.2 retrieveAccessControlGroups (multi client capability)

REQUIREMENTS

- retrieve available access control groups, for optional assignment to a certificate request (section 1.1.6)

PARAMETERS

- ACGSearchFilter - contains criteria to search for access control groups (also see Search functions).

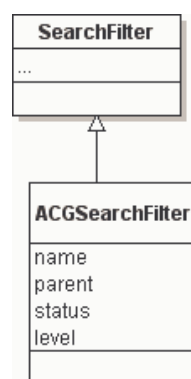


Figure 8 Parameter diagram for the retrieveAccessControlGroups function

Name	Type	Required	Description
name	String	No	Name fragment use to identify the access control group. Only ACGs with names containing this text will be retrieved.
parent	Long	No	Id of a parent access control group. Only ACGs having this parent will be retrieved.
status	Long	No	Access control group status (see Table 28: Access control group status for possible values). Only ACGs having this status will be retrieved.
level	Long	No	Hierarchy level of the access control groups (0 - highest available level). Only ACGs having this level will be retrieved.

Table 5 Parameter description for the retrieveAccessControlGroups function - ACGSearchFilter

RETURN VALUE

- ACGContainer – the search result container for access control groups

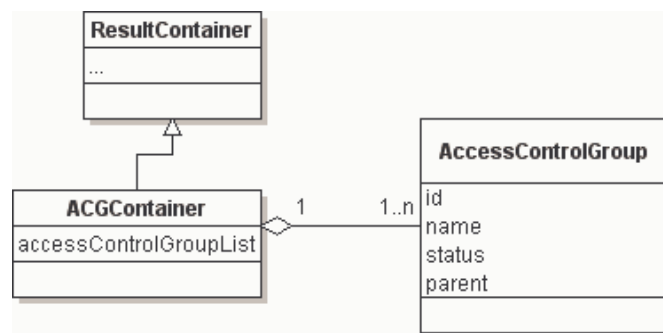


Figure 9 Return value diagram for the retrieveAccessControlGroups function

Name	Type	Description
id	Integer	The id of the access control group.
name	String	Name of the access control group.
status	Integer	Status (see Table 28: Access control group status for possible values).
parent	AccessControlGroup	Parent of the access control group.

Table 6 Return value details for the retrieveAccessControlGroups function - AccessControlGroup

1.3.3 retrieveCertificateProfiles (certificate definitions)

REQUIREMENTS

- retrieve all available certificate profiles (section 1.1.3)

PARAMETERS

- none.

RETURN VALUE

- CertificateProfileContainer – the search result container for certificate profiles

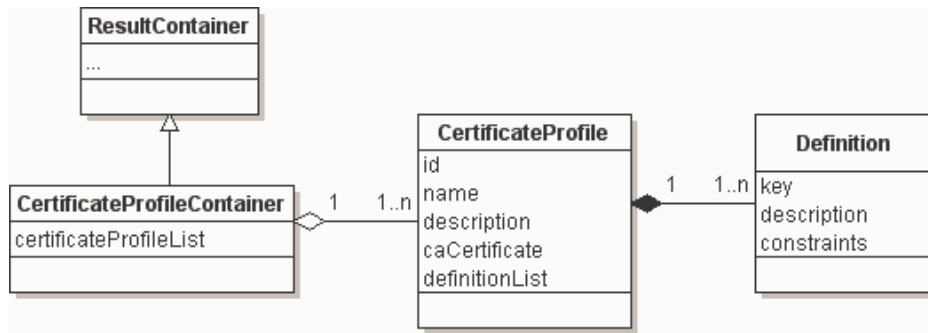


Figure 10 Return value diagram for the retrieveAccessControlGroups function

Name	Type	Description
id	Integer	The id of the certificate profile.
name	String	Name of the certificate profile.
description	String	Description of the profile.
caCertificate	CaCertificate	CA certificate for issuing certificates (the issuer). See Table 19 for more details.
definitionList	Array	Array with the defined certificate template keys.
.key	String	Key to be used as identification data key when requesting a certificate with this profile.
.description	String	Description of this key (usage).
.constraints	String	Constraints for the values of this key.

Table 7 Return value details for the retrieveAccessControlGroups function - CertificateProfil

1.3.4 retrieveRequests (check request status)

REQUIREMENTS

- retrieve details (including status) for one or more requests

PARAMETERS

- RequestSearchFilter - contains criteria to search for requests (also see Search functions)

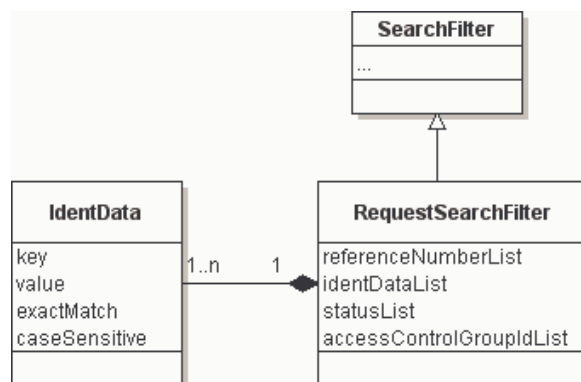


Figure 11 Parameter diagram for the retrieveRequests function

Name	Type	Required	Description
referenceNumberList	Array	No	Array containing request reference numbers. Only requests with these reference numbers will be retrieved.
statusList	Array	No	Array containing request statuses (see Table 27: Certificate request status for possible values). Only requests having these statuses will be retrieved.
identDataList	Array	No	Array containing request identification data. Only requests possessing matching identification data will be retrieved.
.key	String	Yes	Key for an identification value.
.value	String	Yes	Identification value.
.exactMatch	Boolean	No	Defines if the search should be exact.
.caseSensitive	Boolean	No	Defines if the search should be case sensitive.
accessControlGroupIdList	Array	No	Array containing access control group ids. Only requests that belong to these groups will be retrieved.

Table 8 Parameter description for the retrieveRequests function - RequestSearchFilter

RETURN VALUE

- CertificateRequestContainer – the search result container for certificate requests.

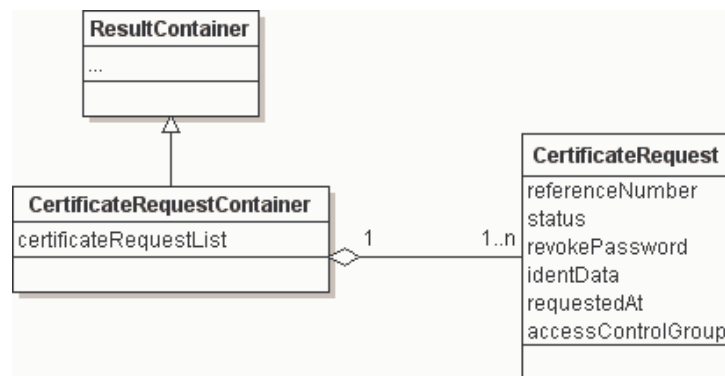


Figure 12 Return value diagram for the retrieveRequests function

Name	Type	Description
referenceNumber	Integer	The reference number of the request.
status	Integer	Status of the request. Table 27 for possible values.
requestedAt	Date	Date when the request was made.
revokePassword	String	Revocation password for the request.
identData	Array	Identification data.
accessControlGroup	AccessControlGroup	Access control group assigned to the request.

Table 9 Return value details for the retrieveRequests function - CertificateRequest

EXAMPLE

```

<ns1:retrieveRequests>
  <filter xsi:type="ns2:RequestSearchFilter">
    <referenceNumberList soapenc:arrayType="soapenc:long[1]" xsi:type="soapenc:Array">
      <referenceNumberList xsi:type="soapenc:long">111</referenceNumberList>
    </referenceNumberList>
  </filter>
</ns1:retrieveRequests>

```

Table 10 Example for the retrieveRequests function

1.3.5 retrieveCertificates (download certificates)

REQUIREMENTS

- retrieve issued certificates
- specify return format for the certificates
- retrieve private keys

PARAMETERS

- CertificateFilter - contains criteria to search for certificates (also see section 1.2.2).
- DownloadFormat - optional definition for download format.

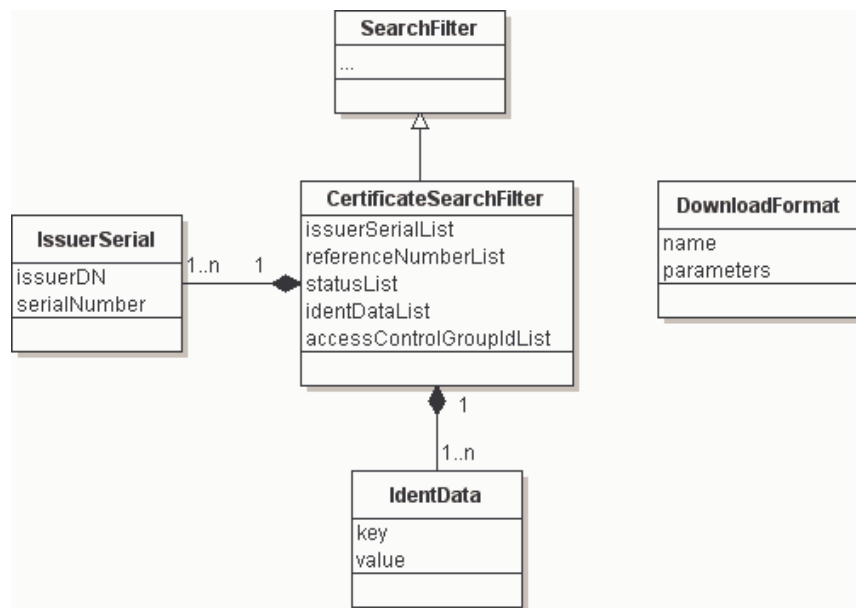


Figure 13 Parameter diagram for the retrieveCertificates function

Name	Type	Required	Description
issuerSerialList	Array	No	Array containing combinations of the certificate issuer and the serial number. Only certificates having exactly these combinations will be retrieved.
.issuerDN	String	Yes	issuerDN of the CA-certificate used to generate this certificate.
.serialNumber	Long	Yes	Serial number of a certificate (unique for a given issuer).
referenceNumberList	Array	No	Array containing request reference numbers. Only certificates with these reference numbers will be retrieved.
statusList	Array	No	Array containing certificate statuses (see Table 30 for possible values). Only requests having these statuses will be retrieved.
identDataList	Array	No	Array containing request identification data. Only certificates possessing the matching identification data will be retrieved.
.key	String	Yes	Key for an identification value.
.value	String	Yes	Identification value.
accessControlGroupIdList	Array	No	Array containing access control group ids. Only requests that belong to these groups will be retrieved.

Table 11 Parameter description for the retrieveCertificates function – CertificateSearchFilter

Name	Type	Required	Description
name	String	No	Name of the format (see Table 29 for more details).
parameters	Array	No	Parameters required for the specified format (see Table 29 for more details).
.name	String	Yes	Parameter name.
.value	String	Yes	Parameter value.

Table 12 Parameter description for the retrieveCertificates function – DownloadFormat

RETURN VALUE

- CertificateRequestContainer – the search result container for certificate requests. The certificates have no search result container because a CertificateRequest may have more than one certificate and acts like a container for certificates. Therefore the search method for certificates will also return a request container.

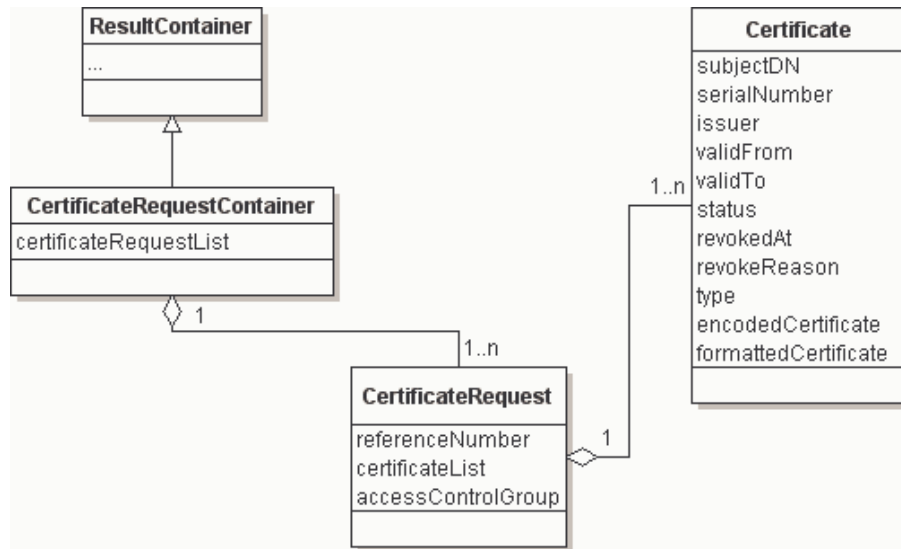


Figure 14 Return value diagram for the retrieveCertificates function

Name	Type	Description
subjectDN	String	Subject distinguished name of the certificate.
serialNumber	Integer	Serial number of the certificate.
issuer	CaCertificate	CA certificate that issued this certificate. See Table 19 for more details.
validFrom	Date	Begin of the validity period.
validTo	Date	End of the validity period.
status	Integer	Status of the certificate. See Table 30 for possible values.
revokedAt	Date	Revocation date of the certificate (if certificate is revoked).
revokeReason	Integer	Revocation reason. See Table 32 for possible values.
type	Integer	Type of the certificate (x.509 or CV).
encodedCertificate	String	Certificate in binary format (Base64 encoded).
formattedCertificate	String	Certificate in specified download format (Base64 encoded).

Table 13 Return value details for the retrieveCertificates function - Certificate

EXAMPLE

```

<ns1:retrieveCertificates >
  <filter xsi:type="ns2:CertificateSearchFilter" >
    <referenceNumberList soapenc:arrayType="soapenc:long[1]" xsi:type="soapenc:Array">
      <referenceNumberList xsi:type="soapenc:long">111</referenceNumberList>
    </referenceNumberList>
  </filter>
</ns1:retrieveCertificates >
  
```

Table 14 Example for the retrieveCertificates function

1.3.6 revokeCertificates (certificate revocation)

REQUIREMENTS

- revoke certificates; all certificates connected to a certificate request reference number will be synchronously revoked.

PARAMETERS

- CertificateSearchFilter - contains criteria that can uniquely specify the certificates to be revoked
- RevocationData - object containing the revocation reason and description.

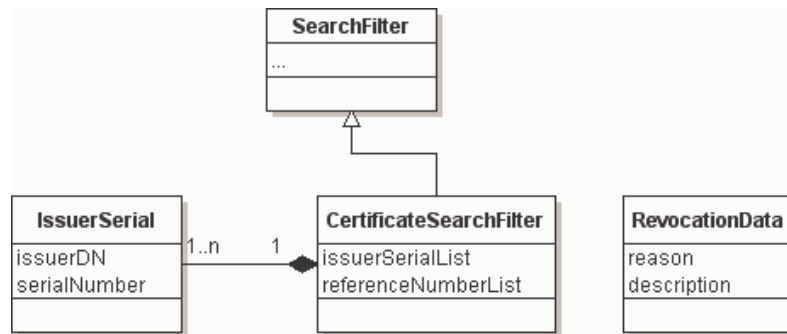


Figure 15: Parameter diagram for the revokeCertificates function

Name	Type	Required	Description
issuerSerialList	Array	Yes/No	Array containing combinations of the certificate issuer and the serial number. Only certificates having possessing exactly these combinations will be revoked (either the issuerSerialList or the referenceNumberList must be specified).
.issuerDN	String	Yes	IssuerDN of the CA-certificate used to generate this certificate.
.serialNumber	Long	Yes	Serial number of a certificate (unique for the same issuer).
referenceNumberList	Array	Yes/No	Array containing request reference numbers. Only certificates with these reference numbers will be revoked (either the issuerSerialList or the referenceNumberList must be specified)

Table 15 Parameter description for the revokeCertificates function – CertificateSearchFilter

Name	Type	Required	Description
reason	Integer	Yes	Revocation reason. See Table 32 for possible values.
description	String	No	Optional description for the revocation.

Table 16 Parameter description for the revokeCertificates function – RevocationData

RETURN VALUE

- CertificateRequestContainer – the container with all certificate requests for which certificates have been revoked. See Figure 12 and Table 13 for more details.

EXAMPLE

```

<ns1:revokeCertificates>
  <filter xsi:type="ns2:CertificateSearchFilter">
    <referenceNumberList soapenc:arrayType="soapenc:long[1]" xsi:type="soapenc:Array">
      <referenceNumberList xsi:type="soapenc:long">111</referenceNumberList>
    </referenceNumberList>
  </filter>
  <revocationData xsi:type="ns3:RevocationData">
    <description xsi:type="soapenc:string">detailed description</description>
    <reason xsi:type="soapenc:long">7</reason>
  </revocationData>
</ns1:revokeCertificates>
    
```

Table 17 Example for the revokeCertificates function

1.3.7 retrieveCaCertificates (provide CA certificate hierarchies)

REQUIREMENTS

- retrieve all available CA certificates

PARAMETERS

- CaCertificateSearchFilter – contains criteria to search for CA certificates.

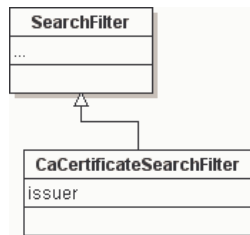


Figure 16 Parameter diagram for the retrieveCaCertificates function

Name	Type	Required	Description
issuer	Long	No	Id of CA certificate used to issue other CA certificates. Only CA certificate having this issuer will be retrieved.

Table 18 Parameter description for the retrieveCaCertificates function - CaCertificateFilter

RETURN VALUE

- CaCertificateContainer – the search result container for CA certificates

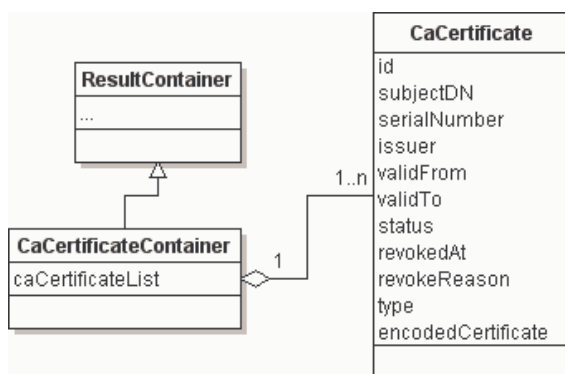


Figure 17 Return value diagram for the retrieveCaCertificates function

Name	Type	Description
id	Integer	Unique identifier for this CA certificate (analogue with the reference number of a request).
subjectDN	String	Subject distinguished name of the CA certificate.
serialNumber	Integer	Serial number of the CA certificate.
issuer	CaCertificate	CA certificate that issued this certificate.
validFrom	Date	Begin of the validity period.
validTo	Date	End of the validity period.
status	Integer	Status of the CA certificate. See Table 30 for possible values.
revokedAt	Date	Revocation date of the CA certificate.
revokeReason	Integer	Revocation reason. See Table 32 for possible values.
type	Integer	Type of the CA certificate (x.509 or CV).
encodedCertificate	String	Certificate in binary format (Base64 encoded).

Table 19 Return value details for the retrieveCaCertificates function – CaCertificate

1.3.8 retrieveCRLs (provide certificate revocation lists)

REQUIREMENTS

- retrieve certification revocation lists

PARAMETERS

- CRLSearchFilter – contains criteria to search for certificates.

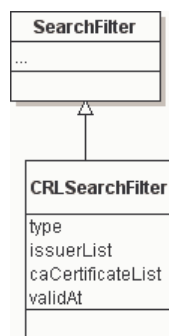


Figure 18 Parameter diagram for the retrieveCRLs function

Name	Type	Required	Description
type	Integer	No	Type of the CRL (see Table 31 for possible values). Only CRLs having this type will be retrieved.
issuerList	Array	No	Ids of the CA certificates that issued CRLs. Only CRLs having one of these issuers will be retrieved.
caCertificateList	Array	No	Ids of the CA certificates that issued the revoked certificates in the CRL. Only CRLs defined to be containing certificates issued by one of these CA certificates will be retrieved).
validAt	Date	No	Point in time when a CRL should be valid. Only CRLs valid at this point in time will be retrieved.

Table 20 Parameter description for the retrieveCRLs function - CRLSearchFilter

RETURN VALUE

- CRLContainer – the search result container for CRLs



Figure 19 Return value diagram for the retrieveCRLs function

Name	Type	Description
id	Integer	Unique identifier for this CRL (analogue with the reference number of a request).
criNumber	Integer	Number uniquely identifying the CRL for a CRL issuer.
issuer	CaCertificate	CA certificate that issued this CRL.
type	Integer	Type of the CRL (see Table 31 for possible values). Only CRLs having this type will be retrieved.
lastUpdate	Date	Last update for this CRL.
nextUpdate	Date	Next update for this CRL.
caCertificateList	Array	The certificate issuers that were searched for revoked certificates when building this CRL.

Table 21 Return value details for the retrieveCRLs function - CRL

1.4 Registration-Authority functions

1.4.1 processCertificateRequest (edit/approve/reject/postpone/activate)

REQUIREMENTS

- edit certificate request data (identification data, revocation password, etc.)
- reject a certificate request
- postpone a certificate request
- approve a certificate request and generate certificates
- activate certificate request; all certificates connected to a certificate request reference number will be synchronously activated.

PARAMETERS

- CertificateRequest – request to be processed. In case of activate, approve, reject or postpone actions, only the reference number is required. See section 1.3.1 for more details on this parameter.
- processAction – action to be executed: edit, approve, reject, postpone, activate
- comment – optional comment for the operation

RETURN VALUE

- PKIResponse – General response for methods which do not have a return value (only error codes).

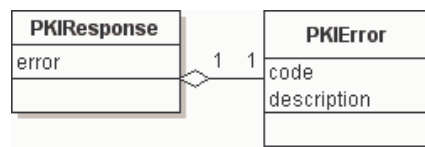


Figure 20: Return value diagram for the processCertificateRequest function

Name	Type	Required	Description
error	PKIError	Yes	See section 1.1 for more details.
.code	String	Yes	Error code.
.description	String	Yes	Detailed description of error code. Table 26 contains a detailed description of all possible error codes.

Table 22 Return value details for the processCertificateRequest function - PKIResponse

1.4.2 unlockCertificates (undo temporary revocation)

REQUIREMENTS

- undo temporarily revocation for certificates

PARAMETERS

- CertificateSearchFilter - contains criteria that can uniquely specify the certificates to be unlocked

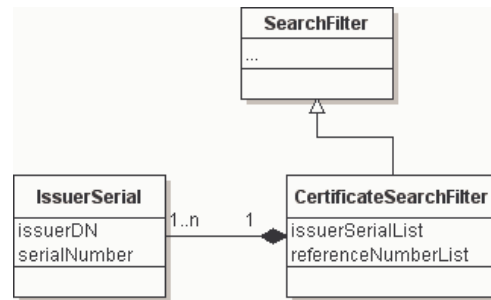


Figure 21: Parameter diagram for the unlockCertificates function

Name	Type	Required	Description
issuerSerialList	Array	Yes/No	Array containing combinations of the certificate issuer and the serial number. Only certificates possessing exactly these combinations will be unlocked (either the issuerSerialList or the referenceNumberList must be specified).
.issuerDN	String	Yes	IssuerDN of the CA-certificate used to generate this certificate.
.serialNumber	Long	Yes	Serial number of a certificate (unique for the same issuer).
referenceNumberList	Array	Yes/No	Array containing request reference numbers. Only certificates with these reference numbers will be unlocked (either the issuerSerialList or the referenceNumberList must be specified).

Table 23 Parameter description for the unlockCertificates function – CertificateSearchFilter

RETURN VALUE

- CertificateRequestContainer – the container with all certificate requests for which certificates have been unlocked. See Figure 12 and Table 13 for more details.

1.4.3 generateCRLs (certificate revocation lists generation)

REQUIREMENTS

- generate certificate revocation lists for given ca certificates (issuers)
- return the freshly generated crls

PARAMETERS

- caCertificateList – array with ids of ca certificates (see section 1.3.7 for details on CA certificates).

RETURN VALUE

- CRLContainer – the search result container for certificate revocation lists containing the freshly generated crls (see section 1.3.8 for more details).

1.4.4 retrieveStatistic (retrieve statistic data)

REQUIREMENTS

- retrieve statistic data regarding certificate lifecycle events or certificate status
- return statistic data

PARAMETERS

- StatisticQuery – defines the statistic data that should be retrieved

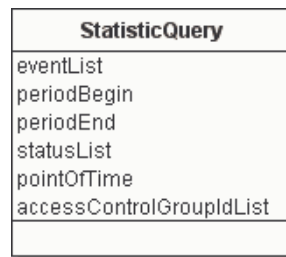


Figure 22 Parameter diagram for the retrieveStatistic function

Name	Type	Required	Description
eventList	Array	Yes/No	List with certificate lifecycle events. Required if no certificate status is provided.
periodBegin	Date	Yes/No	Begin of the period for which the statistic data will be retrieved. Required if events are provided.
periodEnd	Date	Yes/No	End of the period for which the statistic data will be retrieved. Required if events are provided.
statusList	Array	Yes/No	List with certificate statuses. Required if no events are provided.
pointOfTime	Data	Yes/No	Point of time for which the statistic data will be retrieved. Required if statuses are provided
accessControlGroupIdsList	Array	No	Array containing access control group ids. Only certificates that belong to these groups will be considered.

Table 24 Parameter description for the retrieveStatistic function - StatisticQuery

RETURN VALUE

- StatisticContainer – the result container containing statistic data grouped by access control groups.

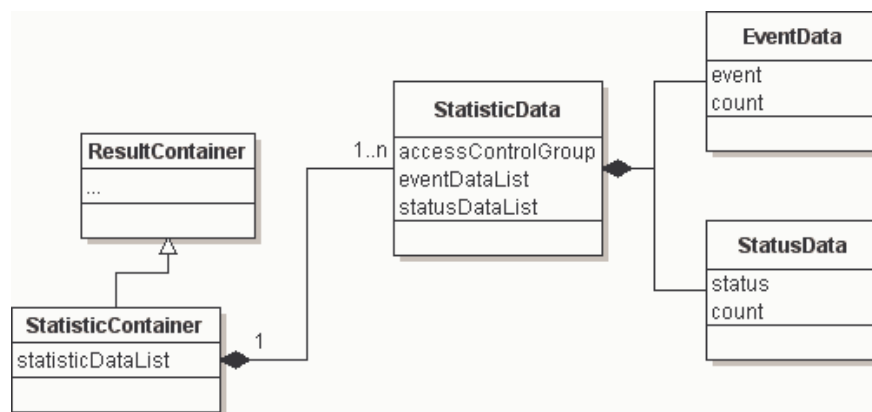


Figure 23 Return value diagram for the retrieveStatistic function

Name	Type	Description
accessControlGroup	AccessControlGroup	Access control group.
eventDataList	Array	List with statistic data grouped by queried certificate lifecycle events.
.event	Long	Certificate lifecycle event.
.count	Long	Number of certificate lifecycle events found.
statusDataList	Array	List with statistic data grouped by queried certificate statuses.
.status	Long	Certificate status.
.count	Long	Number of certificates with the given status.

Table 25 Return value details for the retrieveStatistic function - StatisticData

1.5 API – administration

The most important administrative functions are related to the management of signers and CA certificates, management and publishing of certificate revocation lists and the role management for the PKI users.

Depending on the individual requirements of specific PKI environments, the administrative functions can become very complex. Different security models can also strongly influence the way the administrative functions are being executed (four-eye principle).

For this reasons, the specification for the administrative functions is postponed to a future version of this specification.

1.6 Formats

Name	Description
0000	No error, the call was successful.
0001	Bad credentials – application or/and user credentials could not be verified.
0002	Invalid parameters.
0003	Illegal operation – the operation is not allowed for the provided combination of parameters.
0004	Entity not found – no entity was found for the specified unique identifier.
0005	Internal error.
...	...

Table 26: Error codes

Value	Description
1	New.
2	Pending.
3	Rejected.
4	Approved.

Table 27: Certificate request status

Value	Description
1	Active – all actions allowed.
2	Disabled – new certificate requests not allowed
3	Closed – no action allowed.

Table 28: Access control group status

Name	Value	Description	Required format parameters
PKCS12	1	Complete certificate chain including the private key in PKCS12 format.	password – password for the keystore alias – identifier for the certificate in the keystore
B64	3	Base64 block format.	none
PKCS7	5	Complete certificate chain in PKCS7 format.	none
PKCS8	7	Only the private key in PKCS8 format.	key – AES key to encrypt the private key

Table 29: Certificate download formats

Value	Description
1	Active.
2	Revoked.
3	Inactive.
4	On hold.

Table 30: Certificate status

Value	Description
1	CRL
2	ARL
3	MIXED

Table 31: CRL types

Value	Description
0	Unspecified.
1	Key compromise.
2	CA compromise.
3	Affiliation changed.
4	Superseded.
5	Cessation of operation.
6	Certificate hold.
7	Remove from CRL.
8	Privilege withdrawn.
9	AA compromise.

Table 32: Revocation reasons

Name	Value	Description	Required data
PKCS10	1	Public key and DN information are saved in a standardised ASN1 wrapper.	The public key.
SPKAC	2	Special key / challenge format used by Microsoft.	The public key.
GEN_KEY	3	No public key is provided by the client. The key pair must be generated by the PKI.	XML-Structure containing the generator parameters. TBD
SUBJECT_PUBLIC_KEY_INFO	4	Public key is provided as it is (no wrapper or other information).	The public key.
ATTRIBUTE_CERT	5	No public key is required.	XML-Structure containing o reference to the associated public key certificate. TBD

Table 33: Certificate request data type

REFERENCES

None.

ATTACHMENTS

This document does not have any attachments.

VERSIONSINFORMATIONS

Version	Date	Author	Corrector	Obs. (Author/Corrector)
0.1	2009-03-05	HD	CZ	Initial version.
0.2	2009-03-10	CZ	HD	Updated version with added content.
0.3	2009-05-06	HD	CZ	Implemented feedback. Added new functions.
0.4	2009-05-18	CP	CZ	Modified diagrams. Added further functions.
0.5	2009-06-25	CP		Parameter correction.
	2009-06-25	HD		Statistic, retrieveStatistic function, Admin-API.
0.6	2009-06-29	HD		Updated examples.

Version : Please use one of the following formats „x.x.x” or “x.x”.

Date : Please use the following format „YYYY-MM-DD”.