FINEID - S2

VRK (PRC) CA-model and certificate contents

v3.0

Population Register Centre (VRK)

Certification Authority Services

P.O. Box 123

FIN-00531 Helsinki

Finland

http://www.fineid.fi

CERTIFIED BY inspecta

ISO 9001

Authors

Name	Initials	Organization	E-mail
Antti Partanen	AP	VRK	antti.partanen@vrk.fi
Markku Antikainen	MA	VRK	markku.antikainen@vrk.fi
Mika Pohjolainen	MP	VRK	mika.pohjolainen@vrk.fi
Sauli Toriseva	ST	VRK	sauli.toriseva@vrk.fi
Teemu Tukiainen	TT	VRK	teemu.tukiainen@vrk.fi

Document history

Version	Date	Editor	Changes	Status
3.0	28.12.2016	AP	Editorial changes and corrections	Accepted
3.0	22.12.2016	MA, AP	Editorial changes Updated references to other FINEID documents and IETF RFCs, cerfificate examples updated	Draft
2.5	13.01.2016	TT	Information regarding new CAs: *Service Providers - G3*, *VRK CA for Social Welfare and Health Care Service Providers*, *VRK CA for Test Purposes - G3* and *VRK TEST CA for Social Welfare and Health Care Service Providers*. Updated examples: CA Certificate, Service Certificate and CRL. New examples: OCSP Responder Certificate and Time Stamping Certificate. Some notes about sha256withRSA – algorithm. CA certificate MD5 hashes removed.	Accepted
2.4	18.12.2013	AP	Editorial corrections	Accepted
2.4	3.12.2013	AP	Information about new 'G2' CAs added, Netscape Certificate Extensions deprecated, Root and intermediate CA's CDP and AIA reference table added (chapter 9.2), notes added concerning sha256 hash algorithm, updated new postal address of VRK, Information about VRK Gov. CA for Multiplatform Citizen Qualified Certificates CA removed	Draft
2.3	7.6.2011	ST, AP	Information about new CAs added, transition from teletext to utf8 encoding, pseudonym attribute description added, ldap-CDP syntax modified, description of serialNumber and UPN attribute content	Accepted
2.2	31.08.2007	MP	Definition of subjectAltName extension's Principal Name updated.	Accepted
2.1	05.07.2005	AP, MP	Information about VRK Gov. CA for Multiplatform Citizen Qualified Certificates CA added, implementation of qcStatements extension updated, minor editorial corrections and updates	Accepted
2.0	24.03.2003	AP		Accepted
0.9	18.03.2003	AP		Draft
0.1	28.10.2002	AP		Initial draft

Contents

0.1. Introduction

This document describes VRK (PRC) CA-model and certificate contents.

0.2. About FINEID specifications in general

The FINEID specifications are publicly available documents describing how to implement a public key infrastructure (PKI) using certificates (and smart cards).

The corresponding documents are listed in the table below:

FINEID document	FINEID comments	Based on
FINEID S1	Framework for the Electronic ID application in the smart card.	ISO/IEC 7816-series
FINEID S2	CA-model and content of certificates published and administrated by Population Register Centre (VRK)	IETF RFC 5280 and ETSI 319 412-5 v2.1.1: certificate profile, QCStatements
FINEID S4-1	Implementation profile 1 of the FINEID S1 specification.	ISO/IEC 7816-15, PKCS#15 v1.1, FINEID S1 and FINEID S2
FINEID S4-2	Implementation profile 2 of the FINEID S1 specification.	FINEID S4-1
FINEID S5	Certificate Directory specification	IETF RFC 4510, LDAP

FINEID S4 series contains an implementation profile specifying how the FINEID S1 specification should be put into practice in FINEID context. FINEID S2 is mainly based on IETF RFC 5280 (Certificate and CRL Profile). FINEID S4-1 and S4-2 are based on International Standard ISO/IEC 7816-15 and RSA Data Security Inc. Public-Key Cryptography Standard #15 version 1.1.

Related FINEID specifications are listed below:

- FINEID S1 - Electronic Identity Application, v3.0
- FINEID S4-1 - Implementation Profile 1 for Finnish Electronic ID Card v3.0
- FINEID S4-2 - Implementation Profile 2 for Organizational Usage, v2.1A
- FINEID S5 – Directory Specification, v3.0

FINEID documentation is available at

- **http://www.fineid.fi**

IETF PKIX documentation and RFC’s are available at

- **http://www.ietf.org/rfc**

ETSI Qualified Certificate profile specification is available at

- **http://portal.etsi.org**

Netscape Certificate Extension documentation is available at

- **http://web.archive.org/web/20080514222552/http://wp.netscape.com/eng/security/comm4-cert-exts.html**

Microsoft Guidelines for Enabling Smart Card Logon with Third-Party Certification Authorities

- **http://support.microsoft.com/?kbid=281245**

RSA-based Cryptographic Schemes and Public-Key Cryptography Standards

- **http://www.emc.com/emc-plus/rsa-labs/standards-initiatives/public-key-cryptography-standards.htm**

Secure Hash Standard (SHS) FIPS PUB 180-4 (SHA-1, SHA-224, SHA-256, SHA-384, SHA-512)

- **http://csrc.nist.gov/publications/fips/**

References:

- RFC 3161: Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP). C. Adams Entrust, P. Cain BBN, D. Pinkas Integris, R. Zuccherato Entrust. August 2001.
- RFC 5280: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, D. Cooper, NIST et al., May 2008
- RFC 5480: Elliptic Curve Cryptography Subject Public Key Information, S. Turner, IECA et al., March 2009
- RFC 3739: Internet X.509 Public Key Infrastructure Qualified Certificates Profile, S. Santesson Microsoft, M. Nystrom RSA Security, T. Polk NIST, March 2004
- RFC 6960: X.509 Internet Public Key Infrastructure Online Certificate Status Protocol – OCSP. S. Santesson 3xA Security, M. Myers TraceRoute Security, R. Ankney, A. Malpani CA Technologies, S. Galperin A9, C. Adams University of Ottawa. June 2013.
- ETSI EN 319 412-2 V2.1.1, Certificate Profiles; Part 2: Certificate profile for certificates issued to natural persons, ETSI, February 2016
- ETSI EN 319 412-5 V2.1.1, Certificate Profiles; Part 5: QCStatements, ETSI, February 2016
- Netscape Certificate Extensions, Communicator 4.0 Version, Netscape Communications Corporation

- Microsoft Guidelines for Enabling Smart Card Logon with Third-Party Certification Authorities, Microsoft Knowledge Base article 281245, Revision 3.3, Microsoft Corporation, May 2005
- ITU-T Recommendation X.509 (1997) | ISO/IEC 9594-8:1998, Information technology - Open Systems Interconnection - The Directory: Authentication framework

1. FINEID S2

FINEID S2 specifies the contents of Root, CA and end entity certificates issued by Väestörekisterikeskus (VRK) - Population Register Centre (PRC). FINEID S2 describes also VRK's CA-hierarchy and contents of Certificate Revocation Lists. This specification describes also optional Qualified Certificate Profile extensions for non-repudiation certificates.

Nature of this document, like other FINEID specifications as well, is technical. Basic understanding of certificates and smart cards is needed for full benefit of FINEID documentation. It is not necessary for end users to fully understand technical details of smart cards they use.

In addition to FINEID specifications, software vendors, developers and service providers can also order test cards (with test certificates) from VRK.

The FINEID S2 certificate implementation is based heavily on the IETF RFC 5280. Some additional extensions are extracted from ETSI Qualified Certificate profile and from specifications by Netscape Communications and Microsoft Corporation.

RSA algorithm and Public-Key Cryptography Standards (PKCS) are developed and published by RSA Laboratories, http://www.emc.com/emc-plus/rsa-labs/standards-initiatives/public-key-cryptography-standards.htm

SHA-1 and SHA2 algorithm documentation (FIPS PUB 180-4) is published by NIST, http://csrc.nist.gov/publications/

Note: Not all certificates contain all attributes and extensions described in this specification. Optional attributes are marked as optional. Criticality of extension is also marked.

2. About VRK's certificates

All certificates are issued and administrated by Population Register Centre's Certification Authority services unit, later VRK.

VRK issues two basic types of certificates: User certificates and service certificates. All user certificates are stored in tokens. Smart cards contain Root, CA and two end entity certificates: One for authentication and encryption, and another for non-repudiation digital signatures. All non-repudiation certificates issued by VRK are Qualified Certificates. Private keys associated with non-repudiation certificates are generated inside tokens (smart cards) and there are no copies of those keys.

VRK issues two types of service certificates. Server certificates and system signing certificates are issued based on PKCS#10 Certificate Request and private keys generated by service provider. Service certificate for email usage is a PKCS#12 format file that contains the certificate and corresponding private and public key. It is service providers duty to keep private keys secured using Hardware Security Module, encryption, passwords or by other means.

Certificate Revocation Lists contain information about those certificates which are not valid for some reason. Most common reason is that certificate is not needed anymore or token containing private keys is lost or stolen. Service providers and software products MUST always check validity of certificate against valid CRL before trusting a single certificate. Certificate expiration is not a reason to add certificate into CRL. Also, digital signatures and other transactions occurred BEFORE certificate revocation, are still valid despite of certificate been revoked. For this reason CRL contain exact time when revocation was made.

Authority Revocation Lists contain information about those intermediate CA certificates, which are not valid for some reason. Most common reason is that CA certificate is not needed anymore. This also provides mechanism for Root CA to revoke CA certificate if its private key is exposed. Service providers and software products SHOULD always check validity of intermediate CA certificate against valid ARL before trusting a CA certificate. Certificate expiration is not a reason to add CA certificate into ARL. All certificates and CRLs issued by revoked intermediate CA are not valid after that CA's revocation. This is always fatal situation for CA.

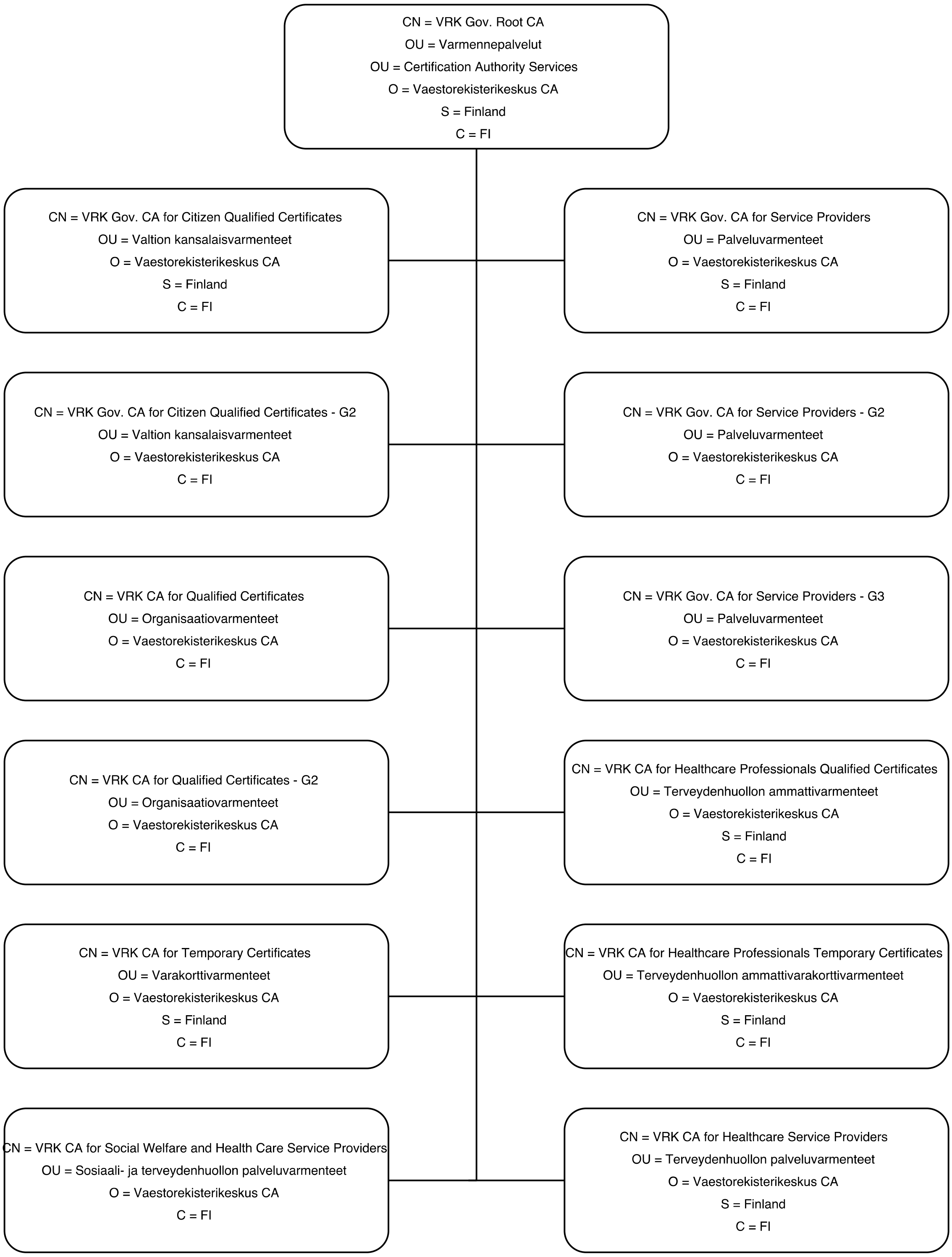
More detailed information is available in Certificate Policies (CP) and Certificate Practice Statements (CPS) available at http://www.fineid.fi/cps

When handling certificates and/or digitally signed data, software products and network services SHOULD perform Basic Path Validation as described in RFC 5280, section 6.1. More specific needs can be fulfilled comparing CPS/policy ID numbers extracted from certificates and make trust decisions based on those.

All certificates, including root, CA and end entity certificates, are published into public certificate directory, which is freely available at ldap.fineid.fi. Same directory also contains CRLs and ARLs. VRK's public certificate directory is documented in FINEID S5 directory specification.

3. Root CA model

Since 18.12.2002 CA model has based on a common Root CA where Root Certificate is self-signed and other VRK's intermediate CAs are signed by VRK Root CA.



CN = VRK Gov. Root CA
OU = Varmennepalvelut
OU = Certification Authority Services
O = Vaestorekisterikeskus CA
S = Finland
C = FI

CN = VRK Gov. CA for Citizen Qualified Certificates
OU = Valtion kansalaisvarmenteet
O = Vaestorekisterikeskus CA
S = Finland
C = FI

CN = VRK Gov. CA for Service Providers
OU = Palveluvarmenteet
O = Vaestorekisterikeskus CA
S = Finland
C = FI

CN = VRK Gov. CA for Citizen Qualified Certificates - G2
OU = Valtion kansalaisvarmenteet
O = Vaestorekisterikeskus CA
C = FI

CN = VRK Gov. CA for Service Providers - G2
OU = Palveluvarmenteet
O = Vaestorekisterikeskus CA
C = FI

CN = VRK CA for Qualified Certificates
OU = Organisaatiovarmenteet
O = Vaestorekisterikeskus CA
C = FI

CN = VRK Gov. CA for Service Providers - G3
OU = Palveluvarmenteet
O = Vaestorekisterikeskus CA
C = FI

CN = VRK CA for Qualified Certificates - G2
OU = Organisaatiovarmenteet
O = Vaestorekisterikeskus CA
C = FI

CN = VRK CA for Healthcare Professionals Qualified Certificates
OU = Terveydenhuollon ammattivarmenteet
O = Vaestorekisterikeskus CA
S = Finland
C = FI

CN = VRK CA for Temporary Certificates
OU = Varakorttivarmenteet
O = Vaestorekisterikeskus CA
S = Finland
C = FI

CN = VRK CA for Healthcare Professionals Temporary Certificates
OU = Terveydenhuollon ammattivarakorttivarmenteet
O = Vaestorekisterikeskus CA
S = Finland
C = FI

CN = VRK CA for Social Welfare and Health Care Service Providers
OU = Sosiaali- ja terveydenhuollon palveluvarmenteet
O = Vaestorekisterikeskus CA
C = FI

CN = VRK CA for Healthcare Service Providers
OU = Terveydenhuollon palveluvarmenteet
O = Vaestorekisterikeskus CA
S = Finland
C = FI

It is easy to build complete PKI enabled solutions where end users and services can share common trust point. Trust decision is made based on VRK's reputation as

Certification Authority (CA). Of course, it is possible to build services where certificates issued by only certain intermediate CA are accepted. Basic trust is however still present across all VRK intermediate CAs and end entity certificates.

4. Root certificate

Root CA Certificate	**Public key length**	**Signed by**
'VRK Gov. Root CA'	2048 bit	Self-signed

The Root certificate shall look like an ordinary end user X.509v3 certificate with following exceptions:

- **subject** equals **issuer** in self signed Root certificate
- key usages **keyCertSign** and **cRLSign** is used in the **keyUsage** extension in Root certificate
- the **basicConstraints** extension is mandatory and the value for the **cA** element shall be set to **TRUE**
- **netscape-cert-type** extension shall be set according to cert usage

Root certificate is introduced in a public directory. It is also available at various web sites. Cardholder's trusted Root certificate is typically stored in the smart card. If in any doubt, it is possible to compare Root and intermediate certificates from different sources to make sure that the Root certificate is valid and issued by VRK. It is also trivial task to test VRK intermediate certificate signature against suspicious Root certificate.

VRK Root Certificate "fingerprints" (hashes) are also listed in **section 9.**

Complete description of Root certificate content is in section 6.

5. Intermediate CA certificates

The intermediate CA certificates shall look like an ordinary end user X.509v3 certificate with following exceptions:

- key usages **keyCertSign** and **cRLSign** shall be used in the **keyUsage** extension in CA certificates
- in the **basicConstraints** extension shall be mandatory and the value for the **cA** element is set to **TRUE. MaxPathLen** attribute in CA certificates is 0 for security reasons.
- **certificatePolicies** extension is not mandatory but it is used.
- **netscape-cert-type** extension is deprecated and not used in 2nd or 3rd Generation ('G2'/'G3') intermediate CA certificates.
- **http-uri pointing to VRK Gov. Root CA authorityRevocationList**

CA certificates and possible cross-certificates are introduced in a public directory. They are also available at various web sites. Cardholder's trusted CA certificate will also reside in the smart card, except in smart cards containing mobile citizen certificates. If in any doubt, it is possible to compare those certificates in different locations to make sure that the CA certificate is valid and issued by VRK. It is also trivial task to test end entity certificate signature against suspicious intermediate CA certificate.

VRK CA Certificates “fingerprints” (hashes) are also listed in **section 9.**

Complete description of CA certificate content is in section 6.

5.1. CA certificates

VRK Root certificate and one intermediate CA shall be stored into the FINEID application into token. These can be used as starting points of trust for the cardholder.

Intermediate CA Certificates	**Public key length**	**Signed by**
‘VRK Gov. CA for Citizen Qualified Certificates’	2048 bit	‘VRK Gov. Root CA’
‘VRK Gov. CA for Citizen Qualified Certificates – G2’	4096 bit	‘VRK Gov. Root CA’
‘VRK CA for Qualified Certificates’	2048 bit	‘VRK Gov. Root CA’
‘VRK CA for Qualified Certificates – G2’	4096 bit	‘VRK Gov. Root CA’
‘VRK CA for Service Providers’	2048 bit	‘VRK Gov. Root CA’
‘VRK CA for Service Providers – G2’	4096 bit	‘VRK Gov. Root CA’
‘VRK CA for Service Providers – G3’	4096 bit	‘VRK Gov. Root CA’
‘VRK CA for Temporary Certificates’	2048 bit	‘VRK Gov. Root CA’
Social Welfare and Health Care CA Certificates		
‘VRK CA for Healthcare Professionals Qualified Certificates’	2048 bit	‘VRK Gov. Root CA’
‘VRK CA for Healthcare Service Providers’	2048 bit	‘VRK Gov. Root CA’
‘VRK CA for Healthcare Professionals Temporary Certificates’	2048 bit	‘VRK Gov. Root CA’
‘VRK CA for Social Welfare and Health Care Service Providers’	4096 bit	‘VRK Gov. Root CA’

Note: ‘VRK CA for Service Providers’, ‘VRK CA for Service Providers – G2’, ‘VRK CA for Service Providers – G3’, ‘VRK CA for Healthcare Service Providers’ and ‘VRK CA for Social Welfare and Health Care Service Providers’ CA certificates are NOT stored into tokens.

6. Certificate contents

This section describes contents of all certificate types issued by VRK.

For complete description of certificate content, syntax and other PKI aspects, see IETF RFC 5280, X.509v3 and other reference documentation mentioned in reference list.

Section 10 contains examples of decoded certificates and CRL.

6.1. Basic certificate fields

The X.509 v3 certificate basic syntax is as follows.

```
Certificate  ::=  SEQUENCE  {
     tbsCertificate       TBSCertificate,
     signatureAlgorithm   AlgorithmIdentifier,
     signatureValue       BIT STRING  }

TBSCertificate  ::=  SEQUENCE  {
     version         [0]  EXPLICIT Version DEFAULT v1,
     serialNumber         CertificateSerialNumber,
     signature            AlgorithmIdentifier,
     issuer               Name,
     validity             Validity,
     subject              Name,
     subjectPublicKeyInfo SubjectPublicKeyInfo,
     issuerUniqueID  [1]  IMPLICIT UniqueIdentifier OPTIONAL,
                          -- If present, version MUST be v2 or v3
     subjectUniqueID [2]  IMPLICIT UniqueIdentifier OPTIONAL,
                          -- If present, version MUST be v2 or v3
     extensions      [3]  EXPLICIT Extensions OPTIONAL
                          -- If present, version MUST be v3
     }

Version  ::=  INTEGER  {  v1(0), v2(1), v3(2)  }

CertificateSerialNumber  ::=  INTEGER

Validity ::= SEQUENCE {
     notBefore      Time,
     notAfter       Time }

Time ::= CHOICE {
     utcTime        UTCTime,
     generalTime    GeneralizedTime }

UniqueIdentifier  ::=  BIT STRING

SubjectPublicKeyInfo  ::=  SEQUENCE  {
     algorithm            AlgorithmIdentifier,
     subjectPublicKey     BIT STRING  }
```

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

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

The following items describe the X.509 v3 certificate for use in the FINEID context.

6.2. Certificate Fields

The Certificate is a SEQUENCE of three required fields. The fields are described in detail in the following subsections.

6.2.1. tbsCertificate

The field contains the names of the subject and issuer, a public key associated with the subject, a validity period, and other associated information. The tbsCertificate includes extensions.

6.2.2. signatureAlgorithm

The signatureAlgorithm field contains the identifier for the cryptographic algorithm used by the CA to sign this certificate.

One of the following algorithms SHALL be used:

```
1.2.840.113549.1.1.5 -- sha1WithRSAEncryption
1.2.840.113549.1.1.11 – sha256WithRSAEncryption
```

SHA-1 algorithm was commonly used with certificates issued by VRK in the past. With all the new certificates, sha256 algorithm SHALL be used.

This field MUST contain the same algorithm identifier as the signature field in the sequence tbsCertificate.

6.2.3. signatureValue

The signatureValue field contains a digital signature computed upon the ASN.1 DER encoded tbsCertificate. The ASN.1 DER encoded tbsCertificate is used as the input to the signature function. This signature value is encoded as a BIT STRING and included in the signature field.

By generating this signature, a CA certifies the validity of the information in the tbsCertificate field. In particular, the CA certifies the binding between the public key material and the subject of the certificate.

6.3. TBSCertificate

The sequence TBSCertificate contains information associated with the subject of the certificate and the CA who issued it. Every TBSCertificate contains the names of the subject and issuer, a public key associated with the subject, a validity period, a version number, and a serial number; some MAY contain optional unique identifier fields. The remainder of this section describes the syntax and semantics of these fields. A TBSCertificate includes extensions. Extensions for the FINEID implementation are described in Section 6.3.8.

6.3.1. version

RFC 5280 defines **Version** type as follows:

```
  Version  ::=  INTEGER  {  v1(0), v2(1), v3(2)  }
```

Only version 3 certificates shall be used (the integer value is 2).

6.3.2. serialNumber

RFC 5280 defines **CertificateSerialNumber** type as follows:

```
  CertificateSerialNumber  ::=  INTEGER
```

All certificates issued by one CA must have unique serial numbers (max. 20 octets).

6.3.3. signature

RFC 5280 defines **AlgorithmIdentifier** type as follows:

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

One of the following algorithms SHALL be used:

```
  1.2.840.113549.1.1.5 -- sha1WithRSAEncryption
  1.2.840.113549.1.1.11 – sha256WithRSAEncryption
```

SHA-1 algorithm was commonly used with certificates issued by VRK in the past. With all the new certificates, sha256 algorithm SHALL be used.

6.3.4. issuer

The issuer field identifies the entity that has signed and issued the certificate. The issuer field is defined as the X.501 type Name. Name type is defined by RFC 5280 as follows:

```
Name ::= CHOICE { -- only one possibility for now --
     rdnSequence  RDNSequence }

   RDNSequence ::= SEQUENCE OF RelativeDistinguishedName

   RelativeDistinguishedName ::=
     SET SIZE (1..MAX) OF AttributeTypeAndValue

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

   AttributeType ::= OBJECT IDENTIFIER

   AttributeValue ::= ANY -- DEFINED BY AttributeType

   DirectoryString ::= CHOICE {
         teletexString           TeletexString (SIZE (1..MAX)),
         printableString         PrintableString (SIZE (1..MAX)),
         universalString         UniversalString (SIZE (1..MAX)),
         utf8String              UTF8String (SIZE (1..MAX)),
         bmpString               BMPString (SIZE (1..MAX)) }
```

The DirectoryString shall be coded as UTF8String with ISO 8859-1 (ISO Latin-1) characters.

The issuer identity is represented by at least the following attributes:

Attribute	OID	Description	ASN.1 type	Example
commonName	{ id-at 3 }	An informative unique (inside organization) name of the CA	PrintableString	'VRK Gov. CA for Citizen Qualified Certificates'
organizationName	{ id-at 10 }	An informative unique name of the issuing organization	PrintableString	'Vaestorekisterikeskus CA'
organizationalUnitName	{ id-at 11 }	An informative name of the issuing organizationUnit. At FINEID context it is used as additional certificate type description in Finnish	PrintableString	'Valtion kansalaisvarmenteet'
stateOrProvinceName	{ id-at 8 }	Name of state. At FINEID context it is used as long form of issuers country name	PrintableString	'Finland'
countryName	{ id-at 6 }	Abbreviation for country	PrintableString	'FI'

Additional attributes may be used.

All VRK’s CA certificates have same issuer:

```
o=Vaestorekisterikeskus CA
c=FI
```

Note: Population Register Centre’s official Finnish name is “Väestörekisterikeskus” (VRK). For compatibility reasons word ‘Väestörekisterikeskus’ is written in certificates without diereses (‘Vaestorekisterikeskus’). Letters ‘CA’ are also added to issuer organization name (o=‘Vaestorekisterikeskus’ vs. ‘o=Vaestorekisterikeskus CA’). This method keeps VRK as normal organization apart from VRK as Certification Authority.

6.3.5. validity

The certificate validity period is the time interval during which the CA warrants that it will maintain information about the status of the certificate. The field is represented as a SEQUENCE of two dates: the date on which the certificate validity period begins (notBefore) and the date on which the certificate validity period ends (notAfter).

RFC 5280 defines the **Validity** type as follows:

```
Validity ::= SEQUENCE {
     notBefore      Time,
     notAfter       Time  }

Time ::= CHOICE {
     utcTime        UTCTime,
     generalTime    GeneralizedTime }
```

The certificate validity period is the time interval during which the CA warrants that it will maintain information about the status of the certificate. The field is represented as a SEQUENCE of two dates: the date on which the certificate validity period begins (notBefore) and the date on which the certificate validity period ends (notAfter).

CAs conforming to this profile MUST always encode certificate validity dates through the year 2049 as UTCTime; certificate validity dates in 2050 or later MUST be encoded as GeneralizedTime.

The validity period for a certificate is the period of time from notBefore through notAfter, inclusive.

UTCTime values shall be expressed in Greenwich Mean Time (GMT) and they shall include seconds as follows:

YYMMDDhhmmssZ

YY	**two least significant digits of the year**
MM	**month (01-12)**
DD	**day (01-31)**
hh	**hour (00-23)**
mm	**minutes (00-59)**
ss	**seconds (00-59)**
Z	**indicates that the time is in GMT**

Where YY is greater than or equal to 50, the year SHALL be interpreted as 19YY; and

Where YY is less than 50, the year SHALL be interpreted as 20YY.

Example: the time **18:57:20** on February 20, 2016, in Finland shall be represented as: **“**160220**165720**Z**“**

Certificate’s notBefore time expresses the moment when corresponding CRL-service is available. Validity period starts from that point.

Validity period shall be set according to the certificate policy.

6.3.6. subject

The subject field identifies the entity associated with the public key stored in the subject public key field. The subject name MAY be carried in the subject field and/or the subjectAltName extension.

The subject field shall be coded with the same rules as the issuer field.

6.3.6.1. Citizen certificates

Certificates issued as citizen certificates may contain the following attributes:

Attribute	OID	Description	ASN.1 type	Example
commonName (mandatory)	{ id-at 3 }	Combination of subject's surname givenName and serialNumber	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'Törmänen Päivi 12345678N' (UTF8String)
surname (mandatory)	{ id-at 4 }	Family name of subject	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'Virtanen' (PrintableString) 'Törmänen' (UTF8String)
givenName (mandatory)	{ id-at 42 }	One of the first names of subject	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'Hilkka' (PrintableString) 'Päivi' (UTF8String)
serialNumber (mandatory)	{ id-at 5 }	Unique identifier of subject in Finland (FINUID)	PrintableString	'12345678N' (PrintableString)
countryName (mandatory)	{ id-at 6 }	Abbreviation for country	PrintableString	'FI' (PrintableString)

SubjectAltName extension MAY contain subject's email address (rfc822Name).

SerialNumber attribute contains a unique identifier (8 digits + checksum character) for a person that within Finland identifies the subject of certification from other persons having exactly the same name. The combination of serialNumber and other attributes of the subject name shall form a unique name for the subject within the CA. Common name is formed from surname, givenName and serialNumber.

6.3.6.2. User certificates for organizational usage

Certificates issued to persons for organizational usage may contain the following additional attributes:

Attribute	OID	Description	ASN.1 type	Example
title (optional)	{ id-at 12 }	Title of subject	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'Projektisihteeri' (PrintableString) 'Osastopäällikkö' (UTF8String)
organizationalUnit Name (optional)	{ id-at 11 }	An informative unique name of subject's organizational unit	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'Hallinto' (PrintableString) 'Henkilöstö-osasto' (UTF8String)
serialNumber (mandatory)	{ id-at 5 }	Unique identifier of subject within CA	PrintableString	'23456789L' (PrintableString)
organizationName (mandatory)	{ id-at 10 }	An informative unique name of subject's organization	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'Yritys Oyj' (PrintableString) 'Kehittämis-ministeriö' (UTF8String)

SubjectAltName extension MAY contain subject's email address (rfc822Name) and user principal name (UPN), for more details see section 6.3.8.5.

Additional attributes MAY be used.

SerialNumber attribute contains a unique identifier (8 digits + checksum character) that identifies the subject of certification from other persons having exactly the same name. In some contexts (e.g. employee certificates issued by a company) the serialNumber may not be by itself unique. However, the combination of serialNumber and other attributes of the subject name shall form a unique name for the subject within the CA. Common name is formed from surname, givenName and serialNumber.

6.3.6.3. User certificates for Healthcare Professional usage

Non-repudiation Digital Signature Certificates contain the following additional attributes:

Attribute	OID	Description	ASN.1 type	Example
title	{ id-at 12 }	Occupation title of subject in Finnish and in Swedish	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'001 lääkäri, läkare' (UTF8String) '005 farmaseutti, farmaceut' (PrintableString)
pseudonym (optional)	{ id-at 65 }	Identification code of subject	PrintableString	'123455' (PrintableString)

6.3.6.4. Service certificates

VRK issues three types of service certificates: server certificates, system signature certificates and PKCS#12 based certificates for email services:

6.3.6.4.1. Server certificates

Server certificates may contain the following attributes:

Attribute	OID	Description	ASN.1 type	Example
commonName (mandatory)	{ id-at 3 }	Server name (URL or IP address)	DirectoryString: PrintableString	'www.fineid.fi' (PrintableString)
organizationalUnit Name (optional)	{ id-at 11 }	An informative unique name of subject's organizational unit	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'Tietohallinto' (PrintableString) 'Pääkonttori' (UTF8String)
organizationName (mandatory)	{ id-at 10 }	An informative unique name of subject's organization	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'Yritys Oyj' (PrintableString) 'Väestörekisterikeskus' (UTF8String)
serialNumber (optional)	{ id-at 5 }	An identity code issued for example to companies, municipalities and natural persons engaged in business activities.	PrintableString	'0245437-2' (PrintableString) 'FI02454372' (PrintableString) '1.2.246.10.2454372' (PrintableString)
localityName (optional)	{ id-at 7 }	An informative name of city, county or other geographic region.	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'Helsinki' (PrintableString)
stateOrProvinceName (optional)	{ id-at 8 }	Name of state. At FINEID context it is used as long form of subject's country name.	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'Finland' (PrintableString)
countryName (mandatory)	{ id-at 6 }	Abbreviation for country.	PrintableString	'FI' (PrintableString)

SubjectAltName extension MAY contain subject's email address (rfc822Name) and SHALL contain subject's DNS name (dNSName).

Additional attributes MAY be used.

6.3.6.4.2. System signature certificates

System signature certificates may contain the following attributes:

Attribute	OID	Description	ASN.1 type	Example
commonName (mandatory)	{ id-at 3 }	Service name	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'Reseptikeskus' (PrintableString) 'Sanoman välityspalvelu' (UTF8String)
organizationalUnitName (optional)	{ id-at 11 }	An informative unique name of subject's organizational unit	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'Tietohallinto' (PrintableString) 'Pääkonttori' (UTF8String)
organizationName (mandatory)	{ id-at 10 }	An informative unique name of subject's organization	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'Yritys Oyj' (PrintableString) 'Väestörekisterikeskus' (UTF8String)
serialNumber (mandatory)	{ id-at 5 }	An identity code issued for example to companies, municipalities and natural persons engaged in business activities.	PrintableString	'0245437-2' (PrintableString) 'FI02454372' (PrintableString) '1.2.246.10.2454372' (PrintableString)
localityName (optional)	{ id-at 7 }	An informative name of city, county or other geographic region.	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'Helsinki' (PrintableString)
stateOrProvinceName (optional)	{ id-at 8 }	Name of state. At FINEID context it is used as long form of subject's country name.	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	'Finland' (PrintableString)
countryName (mandatory)	{ id-at 6 }	Abbreviation for country.	PrintableString	'FI' (PrintableString)

SubjectAltName extension MAY contain subject's email address (rfc822Name).

Additional attributes MAY be used.

KeyUsage for system signature certificates is digitalSignature and nonRepudiation (0xC0)

6.3.6.4.3. Service certificates for email usage

Service certificates for email usage may contain the following attributes:

Attribute	OID	Description	ASN.1 type	Example
commonName (mandatory)	{ id-at 3 }	Service name (name of the email account holder).	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	‘Yritys Oyj’ (PrintableString) ‘Maija Meikäläinen’ (UTF8String)
organizationName (mandatory)	{ id-at 10 }	An informative unique name of subject’s organization.	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	‘Yritys Oyj’ (PrintableString) ‘Kehittämis-ministeriö’ (UTF8String)
organizationalUnit Name (optional)	{ id-at 11 }	An informative unique name of subject’s organizational unit.	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	‘Tietohallinto’ (UTF8String)
serialNumber (mandatory)	{ id-at 5 }	An identity code issued for example to companies, municipalities and natural persons engaged in business activities. A code, consisting of a consecutive number and a control number, given to each party liable to register and by which the party can be identified; issued by NBPR (PRH).	PrintableString	‘0245437-2’ (PrintableString)
localityName (optional)	{ id-at 7 }	An informative name of city, county or other geographic region	DirectoryString: UTF8String. (including ISO Latin 8859-1 characters).	‘Tampere’
stateOrProvinceName (optional)	{ id-at 8 }	Name of state. At FINEID context it is used as long form of subject’s country name	PrintableString	‘Finland’
countryName (mandatory)	{ id-at 6 }	Abbreviation for country	PrintableString	‘FI’ (PrintableString)

SubjectAltName extension SHALL contain subject’s email address (rfc822Name).

KeyUsage for email service certificates is digitalSignature, keyEncipherment, dataEncipherment (0xB0)

6.3.7. subjectPublicKeyInfo

This field is used to carry the public key and identify the algorithm with which the key is used (e.g. RSA or ECC).

RFC 5280 defines the **SubjectPublicKeyInfo** type as follows:

```
SubjectPublicKeyInfo  ::=  SEQUENCE  {
     algorithm            AlgorithmIdentifier,
     subjectPublicKey     BIT STRING  }
```

Following algorithm shall be used:

```
1.2.840.113549.1.1.1 – rsaEncryption
1.2.840.10045.2.1 - id-ecPublicKey
```

In case of RSA, the value for the subjectPublicKey BIT STRING shall be the DER-encoding of the ASN.1 type **RSAPublicKey** defined in PKCS #1 v1.5:

```
RSAPublicKey ::= SEQUENCE {
   modulus            INTEGER,
   publicExponent     INTEGER
}
```

It should be noticed that if the most significant bit of the INTEGER value is set to 1, the value shall be interpreted as negative. If the modulus or public exponent should have the MSbit set to 1, an additional zero byte 00h shall be inserted as the most significant byte of the INTEGER value.

In case of ECC keys, the following curve shall be used:

```
1.2.840.10045.3.1.7 - secp256r1
```

6.3.8. Certificate extensions

This field is a SEQUENCE of one or more certificate extensions. The extensions defined for X.509 v3 certificates provide methods for associating additional attributes with users or public keys and for managing a certification hierarchy. The X.509 v3 certificate format also allows communities to define private extensions to carry information unique to those communities. Each extension in a certificate is designated as either critical or non-critical. A certificate using system MUST reject the certificate if it encounters a critical extension it does not recognize; however, a non-critical extension MAY be ignored if it is not recognized. The following sections present recommended extensions used within FINEID certificates and standard locations for information.

RFC 5280 defines the **Extensions** type as follows:

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

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

This FINEID S2 profile specifies some mandatory extensions in the table below. In addition, the criticality of each extension is also defined. The extensions that are not mandatory can be used with issuer's discretion (i.e. they are optional).

Extension name	FINEID S2		Used in VRK-FINEID environment
	Presence	Criticality	
Standard extensions			
authorityKeyIdentifier	**mandatory**	non-critical	**used**
subjectKeyIdentifier	**mandatory**	non-critical	**used**
keyUsage	**mandatory**	**critical**	**used**
certificatePolicies	**mandatory**	non-critical	**used**
subjectAltName	optional	non-critical	**used**
basicConstraints	**mandatory**	**critical**	**used**
cRLDistributionPoints	**mandatory**	non-critical	**used**
extKeyUsage	optional	non-critical	**used**
Private extensions			
authorityInformationAccess	**mandatory**	non-critical	**used**
netscape-cert-type	optional	non-critical	**used, deprecated and not used in "G2/G3" certificates**
qcStatements	**mandatory**	non-critical	**used** in non-repudiation qualified certificates

Additional extensions not listed above may be used, but they shall not be marked critical.

Mandatory and optional extensions of FINEID S2 are described in more detail below.

6.3.8.1. authorityKeyIdentifier

The authority key identifier extension provides a means of identifying the public key corresponding to the private key used to sign a certificate.

RFC 5280 defines **authorityKeyIdentifier** extension as follows:

```
id-ce-authorityKeyIdentifier OBJECT IDENTIFIER ::=  { id-ce 35 }

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

According to RFC 5280 this field is used to identify the public key to be used to verify the signature on this certificate or CRL. It enables distinct keys used by the same CA to be distinguished (e.g., as key updating occurs).

Only the **keyIdentifier** element shall be used.

This is a **non-critical** extension.

6.3.8.2. subjectKeyIdentifier

The subject key identifier extension provides a means of identifying certificates that contain a particular public key.

To facilitate certification path construction, this extension MUST appear in all conforming CA certificates, that is, all certificates including the basic constraints extension where the value of cA is TRUE. The value of the subject key identifier MUST be the value placed in the key identifier field of the Authority Key Identifier extension of certificates issued by the subject of this certificate.

For end entity certificates, the subject key identifier extension provides a means for identifying certificates containing the particular public key used in an application. Where an end entity has obtained multiple certificates, especially from multiple CAs, the subject key identifier provides a means to quickly identify the set of certificates containing a particular public key. To assist applications in identifying the appropriate end entity certificate, this extension SHOULD be included in all end entity certificates.

RFC 5280 defines **subjectKeyIdentifier** extension as follows:

```
KeyIdentifier ::= OCTET STRING

id-ce-subjectKeyIdentifier OBJECT IDENTIFIER ::=  { id-ce 14 }

SubjectKeyIdentifier ::= KeyIdentifier
```

According to RFC 5280 this field is used to identify the public key being certified. It enables distinct keys used by the same subject to be differentiated (e.g., as key updating occurs.).

This is a **non-critical** extension.

6.3.8.3. keyUsage

The key usage extension defines the purpose (e.g., encipherment, signature, certificate signing) of the key contained in the certificate. The usage restriction might be employed when a key that could be used for more than one operation is to be restricted. For example, when an RSA key should be used only to verify signatures on objects other than public key certificates and CRLs, the digitalSignature or nonRepudiation bits would be asserted.

Likewise, when an RSA key should be used only for key management, the keyEncipherment bit would be asserted.

This extension MUST appear in certificates that contain public keys that are used to validate digital signatures on other public key certificates or CRLs.

RFC 5280 defines the **keyUsage** extension as follows:

```
id-ce-keyUsage OBJECT IDENTIFIER ::=  { id-ce 15 }

      KeyUsage ::= BIT STRING {
           digitalSignature        (0),
           nonRepudiation          (1),
           keyEncipherment         (2),
           dataEncipherment        (3),
           keyAgreement            (4),
           keyCertSign             (5),
           cRLSign                 (6),
           encipherOnly            (7),
           decipherOnly            (8) }
```

According to RFC 5280 this field indicates the purpose for which the certified public key is used.

The following key usages may be used for end entity certificates:

- **digitalSignature** When digital signatures are used but no non-repudiation services are required.
- **nonRepudiation** The public key shall be used to verify digital signatures used to provide a non-repudiation service. This bit shall not be combined with other bits.
- **keyEncipherment** The public key is used for key transport.
- **dataEncipherment** The public key is used for encrypting other user data than keys.

This is a **critical** extension.

6.3.8.4. certificatePolicies

The certificate policies extension contains a sequence of one or more policy information terms, each of which consists of an object identifier (OID) and optional qualifiers.

Applications with specific policy requirements are expected to have a list of those policies, which they will accept, and to compare the policy OIDs in the certificate to that list.

RFC 5280 defines **certificatePolicies** extension as follows:

```
id-ce-certificatePolicies OBJECT IDENTIFIER ::=  { id-ce 32 }

   anyPolicy OBJECT IDENTIFIER ::= { id-ce-certificate-policies 0 }

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

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

CertPolicyId ::= OBJECT IDENTIFIER

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

-- policyQualifierIds for Internet policy qualifiers

id-qt          OBJECT IDENTIFIER ::=  { id-pkix 2 }
id-qt-cps      OBJECT IDENTIFIER ::=  { id-qt 1 }
id-qt-unotice  OBJECT IDENTIFIER ::=  { id-qt 2 }

PolicyQualifierId ::=
     OBJECT IDENTIFIER ( id-qt-cps | id-qt-unotice )

Qualifier ::= CHOICE {
     cPSuri           CPSuri,
     userNotice       UserNotice }

CPSuri ::= IA5String

UserNotice ::= SEQUENCE {
     noticeRef        NoticeReference OPTIONAL,
     explicitText     DisplayText OPTIONAL}

NoticeReference ::= SEQUENCE {
     organization     DisplayText,
     noticeNumbers    SEQUENCE OF INTEGER }

DisplayText ::= CHOICE {
     ia5String        IA5String      (SIZE (1..200)),
     visibleString    VisibleString  (SIZE (1..200)),
     bmpString        BMPString      (SIZE (1..200)),
     utf8String       UTF8String     (SIZE (1..200)) }
```

In an end entity certificate, these policy information terms indicate the policy under which the certificate has been issued and the purposes for which the certificate may be used. In a CA certificate, these policy information terms limit the set of policies for certification paths which include this certificate.

This specification defines two policy qualifier types for use by certificate policy writers and certificate issuers. The qualifier types are the CPS Pointer and User Notice qualifiers.

The CPS Pointer qualifier contains a pointer to a Certification Practice Statement (CPS) published by the CA. The pointer is in the form of a URI.

User notice is intended for display to a relying party when a certificate is used. The application software SHOULD display all user notices in all certificates of the certification path used, except that if a notice is duplicated only one copy needs to be displayed.

FINEID:

The certificate policy of the CA defines whether this extension is single or multivalued.

This is a **non-critical** extension.

6.3.8.5. subjectAltName

The subject alternative names extension allows additional identities to be bound to the subject of the certificate. Defined options include an Internet electronic mail address, a DNS name, an IP address, and a uniform resource identifier (URI).

When the subjectAltName extension contains an Internet mail address, the address MUST be included as an rfc822Name. The format of an rfc822Name is an "addr-spec" as defined in RFC 822.

RFC 5280 defines **subjectAltName** extension as follows:

```
id-ce-subjectAltName OBJECT IDENTIFIER ::=  { id-ce 17 }

   SubjectAltName ::= GeneralNames

   GeneralNames ::= SEQUENCE SIZE (1..MAX) OF GeneralName

   GeneralName ::= CHOICE {
        otherName                       [0]     OtherName,
        rfc822Name                      [1]     IA5String,
        dNSName                         [2]     IA5String,
        x400Address                     [3]     ORAddress,
        directoryName                   [4]     Name,
        ediPartyName                    [5]     EDIPartyName,
        uniformResourceIdentifier       [6]     IA5String,
        iPAddress                       [7]     OCTET STRING,
        registeredID                    [8]     OBJECT IDENTIFIER }

   OtherName ::= SEQUENCE {
        type-id    OBJECT IDENTIFIER,
        value      [0] EXPLICIT ANY DEFINED BY type-id }

   EDIPartyName ::= SEQUENCE {
        nameAssigner            [0]     DirectoryString OPTIONAL,
        partyName               [1]     DirectoryString }
```

To support proprietary Microsoft smart card logon functionality, authentication and encryption, certificate for organizational and healthcare professional usage contains also:

Subject Alternative Name = Other Name: Principal Name = (UPN)

The UPN OtherName OID is : "1.3.6.1.4.1.311.20.2.3"
The UPN OtherName value: Must be ASN1-encoded UTF8 string

Principal Name may be same as rfc822Name (certificate holder's valid email address) but it may also be another name form of the certificate holder that is used to identify users in Active Directory.

For example:

UPN = user1@name.com
UPN = 1234567890@teonet.fi

Note: non-repudiation certificates do NOT contain Principal Name field.

This is a **non-critical** extension.

6.3.8.6. Basic Constraints

The basic constraints extension identifies whether the subject of the certificate is a CA and the maximum depth of valid certification paths that include this certificate.

The cA boolean indicates whether the certified public key belongs to a CA.

The pathLenConstraint field is meaningful only if the cA boolean is asserted. In this case, it gives the maximum number of non-self-issued intermediate certificates that may follow this certificate in a valid certification path.

RFC 5280 defines **basicConstraints** extension as follows:

```
id-ce-basicConstraints OBJECT IDENTIFIER ::=  { id-ce 19 }

   BasicConstraints ::= SEQUENCE {
        cA                      BOOLEAN DEFAULT FALSE,
        pathLenConstraint       INTEGER (0..MAX) OPTIONAL }
```

This extension appears in all VRK's Root, intermediate CA and end entity certificates marked as **critical**.

6.3.8.7. extendedKeyUsage

This extension indicates one or more purposes for which the certified public key may be used, in addition to or in place of the basic purposes indicated in the key usage extension. In general, this extension will appear only in end entity certificates.

This extension is included into FINEID specification for software compatibility reasons only. Usage of this extension in software products is discouraged.

RFC 5280 defines **extendedKeyUsage** extension as follows:

```
id-ce-extKeyUsage OBJECT IDENTIFIER ::= { id-ce 37 }

   ExtKeyUsageSyntax ::= SEQUENCE SIZE (1..MAX) OF KeyPurposeId

   KeyPurposeId ::= OBJECT IDENTIFIER
```

The following key usage purposes are defined:

```
id-kp-serverAuth             OBJECT IDENTIFIER ::= { id-kp 1 }
   -- TLS WWW server authentication

id-kp-clientAuth             OBJECT IDENTIFIER ::= { id-kp 2 }
   -- TLS WWW client authentication

id-kp-codeSigning            OBJECT IDENTIFIER ::= { id-kp 3 }
   -- Signing of downloadable executable code

id-kp-emailProtection        OBJECT IDENTIFIER ::= { id-kp 4 }
   -- E-mail protection

id-kp-timeStamping           OBJECT IDENTIFIER ::= { id-kp 8 }
   -- Binding the hash of an object to a time

id-kp-OCSPSigning            OBJECT IDENTIFIER ::= { id-kp 9 }
   -- Signing OCSP responses


Smart Card Logon OBJECT IDENTIFIER ::= { 1.3.6.1.4.1.311.20.2.2 }
   -- Smart Card logon
```

This is a **non-critical** extension.

6.3.8.8. cRLDistributionPoints

The CRL distribution points extension identifies how CRL information is obtained. The cRLDistributionPoints extension is a SEQUENCE of DistributionPoint.

If the DistributionPointName contains multiple values, each name describes a different mechanism to obtain the same CRL. For example, the same CRL could be available for retrieval through both LDAP and HTTP.

Further discussion of CRL management is contained in section 7.

RFC 5280 defines **cRLDistributionPoints** extension as follows:

```
id-ce-cRLDistributionPoints OBJECT IDENTIFIER ::=  { id-ce 31 }

   CRLDistributionPoints ::= SEQUENCE SIZE (1..MAX) OF
                                            DistributionPoint

   DistributionPoint ::= SEQUENCE {
        distributionPoint       [0]     DistributionPointName
                                                    OPTIONAL,
        reasons                 [1]     ReasonFlags OPTIONAL,
        cRLIssuer               [2]     GeneralNames OPTIONAL }

   DistributionPointName ::= CHOICE {
        fullName                [0]     GeneralNames,
        nameRelativeToCRLIssuer [1]     RelativeDistinguishedName }

   ReasonFlags ::= BIT STRING {
        unused                  (0),
        keyCompromise           (1),
        cACompromise            (2),
        affiliationChanged      (3),
        superseded              (4),
        cessationOfOperation    (5),
        certificateHold         (6),
        privilegeWithdrawn      (7),
        aACompromise            (8) }
```

This field identifies how CRL information is obtained. It is anticipated that the distributionPoint element of DistributionPoint SEQUENCE will contain a uniformResourceIdentifier (URI, element [6] of GeneralName CHOICE) pointing to the appropriate CRL for this certificate.

FINEID:

Examples of the URI containing a LDAP query pointing to the CRL:

- http://proxy.fineid.fi/crl/vrktpc.crl
- ldap://ldap.fineid.fi:389/cn%3dVRK%20CA%20for%20Test%20Purposes,ou%3dTestivarmenteet,o%3dVaestorekisterikeskus%20TEST,dmdName%3dFINEID,c%3dFI?certificateRevocationList

This is a **non-critical** extension.

6.3.9. Private extensions

This section defines extension for use in the Internet Public Key Infrastructure. This extension may be used to direct applications to on-line information about the issuing CA or the subject. As the information may be available in multiple forms, each extension is a sequence of IA5String values, each of which represents a URI. The URI implicitly specifies the location and format of the information and the method for obtaining the information.

An object identifier is defined for the private extension. The object identifier associated with the private extension is defined under the arc id-pe within the arc id-pkix. Any future extensions defined for the Internet PKI are also expected to be defined under the arc id-pe.

```
id-pkix  OBJECT IDENTIFIER  ::=
               { iso(1) identified-organization(3) dod(6)
                 internet(1) security(5) mechanisms(5) pkix(7) }

      id-pe  OBJECT IDENTIFIER  ::=  { id-pkix 1 }
```

6.3.9.1. authorityInfoAccess

The authority information access extension indicates how to access CA information and services for the issuer of the certificate in which the extension appears.

This profile defines two accessMethod OIDs: id-ad-caIssuers and id-ad-ocsp.

RFC 5280 defines **authorityInfoAccess** extension as follows:

```
id-pe-authorityInfoAccess OBJECT IDENTIFIER ::= { id-pe 1 }

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

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

   id-ad OBJECT IDENTIFIER ::= { id-pkix 48 }

   id-ad-caIssuers OBJECT IDENTIFIER ::= { id-ad 2 }

   id-ad-ocsp OBJECT IDENTIFIER ::= { id-ad 1 }
```

The id-ad-caIssuers OID is used when the additional information lists CAs that have issued certificates superior to the CA that issued the certificate containing this extension. The referenced CA issuer's description is intended to help certificate users in the selection of a certification path that terminates at a point trusted by the certificate user.

This is a **non-critical** extension.

6.3.9.2. netscape-cert-type

This is proprietary and deprecated Netscape extension and was used in FINEID context to increase compatibility with software products. Usage of this extension in software products is discouraged and is not used in "G2/G3"-certificates.

If the extension exists in a certificate, it will limit the usage of the certificate to those specified. If the extension is not present, the certificate can be used for all applications ***except Object Signing***.

The value is a bit-string, where the individual bit positions are defined as:

bit-0	SSL client - this cert is certified for SSL client authentication use
bit-1	SSL server - this cert is certified for SSL server authentication use
bit-2	S/MIME - this cert is certified for use by clients (New in PR3)
bit-3	Object Signing - this cert is certified for signing objects such as Java applets and plugins (New in PR3)
bit-4	Reserved - this bit is reserved for future use
bit-5	SSL CA - this cert is certified for issuing certs for SSL use
bit-6	S/MIME CA - this cert is certified for issuing certs for S/MIME use (New in PR3)
bit-7	Object Signing CA - this cert is certified for issuing certs for Object Signing (New in PR3)

This is a **non-critical** extension.

6.3.9.3. qcStatements

Qualified Certificates Profile (IETF RFC 3739) defines qcStatements extension as follows:

```
qcStatements  EXTENSION ::= {
         SYNTAX            QCStatements
         IDENTIFIED BY     id-pe-qcStatements }

     id-pe-qcStatements     OBJECT IDENTIFIER ::= { id-pe 3 }

     QCStatements ::= SEQUENCE OF QCStatement
```

```
QCStatement ::= SEQUENCE {
    statementId   QC-STATEMENT.&Id({SupportedStatements}),
    statementInfo QC-STATEMENT.&Type
    ({SupportedStatements}{@statementId}) OPTIONAL }

SupportedStatements QC-STATEMENT ::= { qcStatement-1,...}
```

According to Qualified Certificates Profile this section defines an extension for inclusion of predefined statements related to Qualified Certificates.

For example, a statement by the issuer that the certificate is issued as a Qualified Certificate is suitable for this extension. Other suitable statements for this extension are statements related to applicable legal jurisdiction within which the certificate is issued (e.g. a maximum reliance limit for the certificate indicating restrictions on CA's liability).

This extension is implemented in all non-repudiation certificates. VRK uses the following ETSI defined statement:

```
id-etsi-qcs-QcCompliance OBJECT IDENTIFIER ::= { id-etsi-qcs 1 }
```

VRK encourages software developers to add support for qualified certificate statements in future software releases.

More information about Qualified certificate profile and Qualified certificate policy statements can be found from ETSI documents:

- **ETSI EN 319 412-5 V2.1.1, Certificate Profiles; Part 5: QCStatements**

This is a **non-critical** extension.

7. Certificate and Authority Revocation Lists

For complete description of CRL content and syntax, see IETF RFC 5280.

Those parts of RFC 5280 that are implemented by VRK are listed here.

The X.509 v2 CRL syntax is as follows. For signature calculation, the data that is to be signed is ASN.1 DER encoded. ASN.1 DER encoding is a tag, length, value encoding system for each element.

```
CertificateList  ::=  SEQUENCE  {
     tbsCertList          TBSCertList,
     signatureAlgorithm   AlgorithmIdentifier,
     signatureValue       BIT STRING  }

TBSCertList  ::=  SEQUENCE  {
     version                 Version OPTIONAL,
                                  -- if present, MUST be v2
     signature               AlgorithmIdentifier,
     issuer                  Name,
     thisUpdate              Time,
     nextUpdate              Time OPTIONAL,
     revokedCertificates     SEQUENCE OF SEQUENCE  {
          userCertificate         CertificateSerialNumber,
          revocationDate          Time,
          crlEntryExtensions      Extensions OPTIONAL
                                        -- if present, MUST be v2
                               }  OPTIONAL,
     crlExtensions           [0]  EXPLICIT Extensions OPTIONAL
                                        -- if present, MUST be v2
                               }

-- Version, Time, CertificateSerialNumber, and Extensions
-- are all defined in section 7

-- AlgorithmIdentifier is defined in section 7
```

7.1. CertificateList Fields

The CertificateList is a SEQUENCE of three required fields. The fields are described in detail in the following subsections.

7.1.1. tbsCertList

The first field in the sequence is the tbsCertList. This field is itself a sequence containing the name of the issuer, issue date, issue date of the next list, the optional list of revoked certificates, and optional CRL extensions. When there are no revoked certificates, the revoked certificates list is absent. When one or more certificates are revoked, each entry on the revoked certificate list is defined by a sequence of user certificate serial number, revocation date, and optional CRL entry extensions.

7.1.2. signatureAlgorithm

The signatureAlgorithm field contains the algorithm identifier for the algorithm used by the CRL issuer to sign the CertificateList.

This field MUST contain the same algorithm identifier as the signature field in the sequence tbsCertList.

One of the following algorithms SHALL be used:

```
  1.2.840.113549.1.1.5 -- sha1WithRSAEncryption
  1.2.840.113549.1.1.11 – sha256WithRSAEncryption
```

SHA-1 algorithm was commonly used by VRK in the past. Currently sha256 algorithm SHALL be used to sign the CertificateList.

7.1.3. signatureValue

The signatureValue field contains a digital signature computed upon the ASN.1 DER encoded tbsCertList. The ASN.1 DER encoded tbsCertList is used as the input to the signature function. This signature value is encoded as a BIT STRING and included in the CRL signatureValue field.

7.2. Certificate List "To Be Signed"

The certificate list to be signed, or TBSCertList, is a sequence of required and optional fields. The required fields identify the CRL issuer, the algorithm used to sign the CRL, the date and time the CRL was issued, and the date and time by which the CRL issuer will issue the next CRL.

Optional fields include lists of revoked certificates and CRL extensions. The revoked certificate list is optional to support the case where a CA has not revoked any unexpired certificates that it has issued. The profile requires conforming CRL issuers to use the CRL number and authority key identifier CRL extensions in all CRLs issued.

7.2.1. Version

This optional field describes the version of the encoded CRL. This field MUST be present and MUST specify version 2 (the integer value is 1).

7.2.2. Signature

This field contains the algorithm identifier for the algorithm used to sign the CRL.

This field MUST contain the same algorithm identifier as the signatureAlgorithm field in the sequence CertificateList.

One of the following algorithms SHALL be used:

```
  1.2.840.113549.1.1.5 -- sha1WithRSAEncryption
  1.2.840.113549.1.1.11 – sha256WithRSAEncryption
```

SHA-1 algorithm was commonly used by VRK in the past. Currently sha256 algorithm SHALL be used to sign CertificateList.

7.2.3. Issuer Name

The issuer name identifies the entity that has signed and issued the CRL. The issuer identity is carried in the issuer name field.

7.2.4. This Update

This field indicates the issue date of this CRL. ThisUpdate may be encoded as UTCTime or GeneralizedTime.

CRL issuers conforming to this profile MUST encode thisUpdate as UTCTime for dates through the year 2049. CRL issuers conforming to this profile MUST encode thisUpdate as GeneralizedTime for dates in the year 2050 or later.

7.2.5. Next Update

This field indicates the date by which the next CRL will be issued. The next CRL could be issued before the indicated date, but it will not be issued any later than the indicated date. CRL issuers SHOULD issue CRLs with a nextUpdate time equal to or later than all previous CRLs. nextUpdate may be encoded as UTCTime or GeneralizedTime.

CRL issuers conforming to this profile MUST encode nextUpdate as UTCTime for dates through the year 2049. CRL issuers conforming to this profile MUST encode nextUpdate as GeneralizedTime for dates in the year 2050 or later.

7.2.6. Revoked Certificates

When there are no revoked certificates, the revoked certificates list MUST be absent. Otherwise, revoked certificates are listed by their serial numbers. Certificates revoked by the CA are uniquely identified by the certificate serial number. The date on which the revocation occurred is specified. The time for revocationDate MUST be expressed. Additional information may be supplied in CRL entry extensions.

7.3. Extensions

This field is a sequence of one or more CRL extensions.

7.3.1. CRL Extensions

The extensions defined by ITU-T for X.509 v2 CRLs provide methods for associating additional attributes with CRLs. The X.509 v2 CRL format also allows communities to define private extensions to carry information unique to those communities. Each extension in a CRL may be designated as critical or non-critical. A CRL validation MUST fail if it encounters a critical extension which it does not know how to process. However, an unrecognized non-critical extension may be ignored. The following subsections present those extensions used within VRK CRLs.

7.3.1.1. Authority Key Identifier

The authority key identifier extension provides a means of identifying the public key corresponding to the private key used to sign a CRL. The identification can be based on either the key identifier (the subject key identifier in the CRL signer's certificate) or on the issuer name and serial number. This extension is especially useful where an issuer has more than one signing key, either due to multiple concurrent key pairs or due to changeover.

7.3.1.2. CRL Number

The CRL number is a non-critical CRL extension which conveys a monotonically increasing sequence number for a given CRL scope and CRL issuer. This extension allows users to easily determine when a particular CRL supersedes another CRL. CRL numbers also support the identification of complementary complete CRLs and delta CRLs.

Given the requirements above, CRL numbers can be expected to contain long integers. CRL verifiers MUST be able to handle CRLNumber values up to 20 octets.

```
id-ce-cRLNumber OBJECT IDENTIFIER ::= { id-ce 20 }

CRLNumber ::= INTEGER (0..MAX)
```

7.3.1.3. Issuing Distribution Point

The issuing distribution point is a critical CRL extension that identifies the CRL distribution point and scope for a particular CRL, and it indicates whether the CRL covers revocation for end entity certificates only, CA certificates only, attribute certificates only, or a limited set of reason codes. Although this extension is critical, conforming implementations are not required to support this extension.

If the distributionPoint field is absent, the CRL MUST contain entries for all revoked unexpired certificates issued by the CRL issuer, if any, within the scope of the CRL.

```
id-ce-issuingDistributionPoint OBJECT IDENTIFIER ::= { id-ce 28 }

issuingDistributionPoint ::= SEQUENCE {
     distributionPoint          [0] DistributionPointName OPTIONAL,
     onlyContainsUserCerts      [1] BOOLEAN DEFAULT FALSE,
     onlyContainsCACerts        [2] BOOLEAN DEFAULT FALSE,
     onlySomeReasons            [3] ReasonFlags OPTIONAL,
     indirectCRL                [4] BOOLEAN DEFAULT FALSE,
     onlyContainsAttributeCerts [5] BOOLEAN DEFAULT FALSE }
```

7.3.2. CRL Entry Extensions

The CRL entry extensions defined by ITU-T for X.509 v2 CRLs provide methods for associating additional attributes with CRL entries. Each extension in a CRL entry may be designated as critical or non-critical. A CRL validation MUST fail if it encounters a critical CRL entry extension which it does not know how to process. However, an unrecognized non-critical CRL entry extension may be ignored.

All CRL entry extensions used in this specification are non-critical. Support for these extensions is optional for conforming CRL issuers and applications. However, CRL issuers SHOULD include reason codes and invalidity dates whenever this information is available.

7.3.2.1. Reason Code

The reasonCode is a non-critical CRL entry extension that identifies the reason for the certificate revocation. CRL issuers are strongly encouraged to include meaningful reason codes in CRL entries.

```
   id-ce-cRLReason OBJECT IDENTIFIER ::= { id-ce 21 }

   -- reasonCode ::= { CRLReason }

   CRLReason ::= ENUMERATED {
        unspecified             (0),
        keyCompromise           (1),
        cACompromise            (2),
        affiliationChanged      (3),
        superseded              (4),
        cessationOfOperation    (5),
        certificateHold         (6),
        removeFromCRL           (8),
        privilegeWithdrawn      (9),
        aACompromise           (10) }
```

7.3.2.2. Invalidity Date

The invalidity date is a non-critical CRL entry extension that provides the date on which it is known or suspected that the private key was compromised or that the certificate otherwise became invalid. This date may be earlier than the revocation date in the CRL entry, which is the date at which the CA processed the revocation. When a revocation is first posted by a CRL issuer in a CRL, the invalidity date may precede the date of issue of earlier CRLs, but the revocation date SHOULD NOT precede the date of issue of earlier CRLs.

The GeneralizedTime values included in this field MUST be expressed in Greenwich Mean Time (Zulu).

```
id-ce-invalidityDate OBJECT IDENTIFIER ::= { id-ce 24 }

invalidityDate ::=  GeneralizedTime
```

8. Summary Tables

8.1. Common subject and issuer attributes

Detailed information can be found in RFCs 5280, 4512, 4519, 4523, 4524, and FINEID S5 specifications.

Contents of the attribute types are encoded as Printable Strings or UTF8 Strings using ISO Latin-1 (8859.1) character set.

For backward compatibility reasons software implementations SHALL support Latin-1 character set encoded as Teletext/T.61 and UTF8 string.

Software implementations SHALL recognize the following attributes.

```
id-at OBJECT IDENTIFIER ::= { joint-iso-ccitt(2) ds(5) 4 }
id-ce OBJECT IDENTIFIER  ::= {joint-iso-ccitt(2) ds(5) 29}

id-at-commonName        AttributeType ::= { id-at 3 }
id-at-surname           AttributeType ::= { id-at 4 }
id-at-givenName         AttributeType ::= { id-at 42 }
id-at-serialNumber      AttributeType ::= { id-at 5 }
id-at-title             AttributeType ::= { id-at 12 }
id-at-pseudonym         AttributeType ::= { id-at 65 }
id-at-organizationalUnitName AttributeType ::= { id-at 11 }
id-at-organizationName  AttributeType ::= { id-at 10 }
id-at-stateOrProvinceName AttributeType ::= { id-at 8 }
id-at-localityName      AttributeType ::= { id-at 7 }
id-at-countryName       AttributeType ::= { id-at 6 }
id-at-dmdName           AttributeType ::= { id-at 54 }
id-ce-subjectAltName OBJECT IDENTIFIER ::=  { id-ce 17 }
    -- for email addresses
```

Other attributes might be used.

9. Certificate information summary

Root and CA certificates:

Issuer Name	Certificate type	Signed by	Valid from	Valid until	Key length
VRK Gov. Root CA	Root certificate	VRK Gov. Root CA	18.12.2002	18.12.2023	2048
VRK Gov. CA for Citizen Qualified Certificates	CA certificate	VRK Gov. Root CA	10.01.2003	09.01.2019	2048
VRK Gov. CA for Citizen Qualified Certificates – G2	CA certificate	VRK Gov. Root CA	21.11.2013	17.12.2023	4096
VRK CA for Qualified Certificates	CA certificate	VRK Gov. Root CA	14.01.2003	13.01.2019	2048
VRK CA for Qualified Certificates – G2	CA certificate	VRK Gov. Root CA	21.11.2013	17.12.2023	4096
VRK CA for Service Providers	CA certificate	VRK Gov. Root CA	28.01.2003	27.01.2019	2048
VRK CA for Service Providers – G2	CA certificate	VRK Gov. Root CA	12.11.2013	17.12.2023	4096
VRK CA for Service Providers – G3	CA certificate	VRK Gov. Root CA	07.12.2015	18.12.2023	4096
VRK CA for Temporary Certificates	CA certificate	VRK Gov. Root CA	24.06.2008	13.01.2019	2048
VRK CA for Healthcare Service Providers	CA certificate	VRK Gov. Root CA	27.10.2010	17.12.2023	2048
VRK CA for Social Welfare and Health Care Service Providers	CA certificate	VRK Gov. Root CA	07.12.2015	18.12.2023	4096
VRK CA for Healthcare Professionals Qualified Certificates	CA certificate	VRK Gov. Root CA	27.10.2010	17.12.2023	2048
VRK CA for Healthcare Professionals Temporary Certificates	CA certificate	VRK Gov. Root CA	27.10.2010	17.12.2023	2048

End entity certificates:

Issuer Name	Certificate type	Signed by	Validity period	Key length
VRK Gov. CA for Citizen Qualified Certificates	Personal Citizen certificate	VRK Gov. CA for Citizen Qualified Certificates	5 years	2048
VRK Gov. CA for Citizen Qualified Certificates – G2	Personal Citizen certificate	VRK Gov. CA for Citizen Qualified Certificates – G2	5 years	2048
VRK CA for Qualified Certificates	Personal certificate	VRK CA for Qualified Certificates	max. 5 years	2048
VRK CA for Qualified Certificates – G2	Personal certificate	VRK CA for Qualified Certificates – G2	max. 5 years	2048
VRK CA for Service Providers	Service certificate	VRK CA for Service Providers	1,2 or 5 years	2048/ 4096
VRK CA for Service Providers – G2	Service certificate	VRK CA for Service Providers – G2	1,2 or 5 years	2048/ 4096
VRK CA for Service Providers – G3	Service certificate	VRK CA for Service Providers – G3	1,2 or 5 years	2048/ 4096
VRK CA for Temporary Certificates	Personal certificate	VRK CA for Temporary Certificates	max. 3 months	2048
VRK CA for Healthcare Service Providers	Service certificate	VRK CA for Healthcare Service Providers	max. 5 years	2048/ 4096
VRK CA for Social Welfare and Health Care Service Providers	Service certificate	VRK CA for Social Welfare and Health Care Service Providers	max. 5 years	2048/ 4096
VRK CA for Healthcare Professionals Qualified Certificates	Personal certificate	VRK CA for Healthcare Professionals Qualified Certificates	max. 5 years	2048
VRK CA for Healthcare Professionals Temporary Certificates	Personal certificate	VRK CA for Healthcare Professionals Temporary Certificates	max. 3 months	2048

Test Root and CA certificates:

Issuer Name	Certificate type	Signed by	Valid from	Valid until	Key length
VRK TEST Root CA	TEST Root certificate	VRK TEST Root CA	17.12.2002	17.12.2023	2048
VRK CA for Test Purposes	TEST CA certificate	VRK TEST Root CA	13.01.2003	12.01.2019	2048
VRK CA for Test Purposes – G2	TEST CA certificate	VRK TEST Root CA	08.11.2013	16.12.2023	4096
VRK CA for Test Purposes – G3	TEST CA certificate	VRK TEST Root CA	13.11.2015	17.12.2023	4096
VRK TEST CA for Temporary Certificates	TEST CA certificate	VRK TEST Root CA	23.01.2008	17.12.2023	2048
VRK TEST CA for Healthcare Professionals	TEST CA certificate	VRK TEST Root CA	22.06.2010	22.06.2023	2048
VRK TEST CA for Healthcare Service Providers	TEST CA certificate	VRK TEST Root CA	22.06.2010	22.06.2023	2048
VRK TEST CA for Social Welfare and Health Care Service Providers	TEST CA certificate	VRK TEST Root CA	13.11.2015	17.12.2023	4096

Test end entity certificates:

Issuer Name	Certificate type	Signed by	Validity period	Key length
VRK CA for Test Purposes	Test personal certificate	VRK CA for Test Purposes	max. 5 years	2048
VRK CA for Test Purposes – G2	Test personal certificate	VRK CA for Test Purposes – G2	max. 5 years	2048
VRK CA for Test Purposes – G3	Test personal certificate	VRK CA for Test Purposes – G3	max. 5 years	2048
VRK CA for Test Purposes	Test service certificate	VRK CA for Test Purposes	max. 5 years	2048/ 4096
VRK CA for Test Purposes – G2	Test service certificate	VRK CA for Test Purposes – G2	max. 5 years	2048/ 4096
VRK TEST CA for Healthcare Professionals	Test personal certificate	VRK TEST CA for Healthcare Professionals	max. 5 years	2048
VRK TEST CA for Healthcare Service Providers	Test service certificate	VRK TEST CA for Healthcare Service Providers	max. 5 years	2048/ 4096
VRK TEST CA for Social Welfare and Health Care Service Providers	Test service certificate	VRK TEST CA for Social Welfare and Health Care Service Providers	max. 5 years	2048/ 4096

9.1. Root and CA Certificate Fingerprints (signature hashes)

Root and CA certificates:

	SHA-1 (160 bit)	SHA-256 (256 bit)
VRK Gov. Root CA	FA:A7:D9:FB:31:B7:46:F2:00:A8 5E:65:79:76:13:D8:16:E0:63:B5	F0:08:73:3E:C5:00:DC:49:87:63 CC:92:64:C6:FC:EA:40:EC:22:00 0E:92:7D:05:3C:E9:C9:0B:FA:04 6C:B2
VRK Gov. CA for Citizen Qualified Certificates	40:D5:DE:E1:9E:C9:44:63:81:96 78:A1 FD:34:6B:A6:56:ED 00:0D	36:C5:EB:DE:1D:6A:9A:0E:D4:E6 6B:61:4A:0C:7A:21:7A:07:7D:C6 38:EE:70:4F:BF:66:68:44:3B:59 54:B1
VRK Gov. CA for Citizen Qualified Certificates – G2	6F:D1:63:94:96:99:75:0D:69:51 D5:28:F8:57:B7:ED:95:50:F1:13	82:63:C4:EC:2B:EF:EB:26:BD:D8 33:F7:78:73:CC:8E:C5:9D:93:4A 16:28:2C:2B:08:48:CD:A2:AF:DA 8E:B5
VRK CA for Qualified Certificates	C2:4E:F0:75:7B:90:A9:29:78:2E 58:FF AC:3E:1E:D2:B8:05:94:8B	AE:75:B2:00:65:5E:16:AF:C6:35 01:2C:99:FE:50:3D:C0:70:EA:CA 88:A3:62:E8:2F:D7:40:EA:3C:F3 68:5E
VRK CA for Qualified Certificates – G2	CE:29:25:26:56:70:B3:E4:BD:FA 52:B6:2A:65:D4:C9:43:46:E8:41	A5:DF:C5:AD:09:B2:F6:6A:68:EF 0F:B3:B4:BD:AA:42:AA:3A:DD:6F 3E:7B:BF:CB:DF:D2:7F:C7:CC:D9 F3:D8
VRK CA for Service Providers	57:93:F4:65:15:73:01:0B:C1:86 22:07:FC:90:83:17:4A:9C:8E:38	CA:B0:74:E6:1D:87:6A:0F:06:76 5B:70:8C:CE:6B:80:CF:3E:9A:73 0F:C3:2B:08:17:AC:B0:D2:93:8C 2C:67
VRK CA for Service Providers – G2	56:E6:18:58:21:84:CC:CB:4C:44 8C:F1:4C:76:B1:6F:E8:25:E7:59	41:D2:E6:F4:67:8D:24:28:04:50 83:AC:07:4A:2A:D6:0E:F2:F8:11 BA:06:E1:96:3F:E3:19:B2:FF:67 C1:37
VRK CA for Service Providers – G3	FD:00:6B:D7:5C:5D:C1:CE:24:0F 3C:AD:EB:86:1C:35:8D:D6:9C:42	B7:50:A1:25:F5:FB:28:5B:FE:5C C1:FF:2D:97:13:88:75:DB:CE:C7 5A:B6:78:12:4C:27:0E:26:D8:C1 9A:8E
VRK CA for Temporary Certificates	1C:04:7C:5E:05:CC:DF:99:F0:3D 6A:8D:80:B8:8D:66:E7:04:D9:E2	8B:89:D1:26:D2:51:9D:E2:7D:FD D0:72:FC:58:7F:39:B3:10:11:2E EE:17:9D:C0:00:2C:FD:36:C2:CF 74:8A
VRK CA for Healthcare Professionals Qualified Certificates	7B:03:65:FC:D3:1E:56:F9:43:2B 11:67:A6:D6:FB:EA:8E:9D:EE:09	C2:73:FA:D0:43:93:0E:B4:F5:F5 20:2F:5A:92:EA:D9:35:15:7F:DD 73:E2:9F:80:7A:95:B0:E4:2C:16 0F:99
VRK CA for Healthcare Professionals Temporary Certificates	8D:EF:EB:1B:AA:F9:BC:FF:D2:D9 C7:C8:41:46:2D:52:0C:92:35:C3	60:EF:56:2D:61:9F:A7:4B:9B:2B 80:F8:6D:CD:9B:4D:36:AC:D6:FD 74:B1:9A:61:6C:ED:F0:81:0F:9F 66:79
VRK CA for Healthcare Service Providers	C2:07:4F:87:AF:48:F7:02:63:E7 8B:91:10:C8:7D:51:E0:CE:3E:14	DC:A6:93:F3:A4:1D:AA:9C:75:C0 09:0A:A6:DE:70:16:73:CF:32:13 CD:60:8B:1E:75:4C:AF:EF:F5:16 87:A6
VRK CA for Social Welfare and Health Care Service Providers	F0:82:3B:E0:8C:B8:DE:C9:72:B7 EB:C0:5A:84:91:B7:12:EC:BF:AB	7D:CA:92:5E:39:E2:A4:5C:8E:61 FA:EB:7D:9A:87:01:DB:A0:D2:9F FE:E9:79:74:65:20:7C:FA:0F:CB 4B:01

Current software products use typically SHA-1 or SHA-256 fingerprints.

Older software products might still use MD5 fingerprints but use of MD5 is discouraged, thus MD5 fingerprints are not anymore provided here.

Test Root and CA certificates:

	SHA-1 (160 bit)	SHA-256 (256 bit)
VRK TEST Root CA	EC:BB:5F:DD:DC:BF:00:71:D6:AB F9:12 67:4C:6D:B2:20:46:7A:20	0E:11:39:48:50:45:37:D4:1D:77 9A:4E:28:11:5D:D5:3B:8A:7B:E9 00:56:A7:4E:97:15:12:1B:2A:2E 2B:07
VRK CA for Test Purposes	AA:DC:E8:CE:A0:7D:AF:35:4D:50 23:99 D0:ED:CA:E1:0D:88 F7:78	43:6D:82:C1:C4:CC:B5:A1:8D:58 CB:CB:E9:9C:AE:7F:48:B0:6A:92 FC:60:25:B0:D6:66:E8:35:7B:AB E2:E1
VRK CA for Test Purposes – G2	6F:63:43:EB:F2:8B:90:4A:53:69 4A:5B:6E:C7:60:41:AF:43:9C:C5	82:15:1B:D1:6F:10:C2:DA:80:32 80:40:A6:A6:9D:8E:82:61:CB:1B 3B:32:B3:B8:84:4E:AB:3C:EF:8C 43:20
VRK CA for Test Purposes – G3	45:BB:BD:2E:F2:F7:59:79:9A:EF 57:39:D7:C9:E0:79:96:EA:75:54	4A:2C:23:B4:1E:29:65:2B:4B:D5 8D:CB:E9:42:13:D3:C0:48:65:87 58:FC:E6:35:97:30:0A:1B:6B:4E FC:2C
VRK TEST CA for Healthcare Professionals	08:08:3D:01:2B:4B:4C:EF:2B:D1 92:59:EA:0E:6B:F2:53:F3:3F:70	01:AE:5C:60:03:AA:DE:7D:09:A1 F6:75:8E:E3:54:11:FE:8A:55:31 58:30:89:38:51:26:83:19:D6:9D 71:9B
VRK TEST CA for Healthcare Service Providers	12:4A:C1:4E:AD:41:80:3D:62:FE 03:91:8E:FD:AB:16:CA:51:19:36	DD:A9:6B:64:B6:E1:C1:F5:80:EB ED:B4:E5:FC:5D:5A:7D:30:77:C5 0B:58:FB:02:E6:26:1A:A2:19:1B E4:E8
VRK TEST CA for Social Welfare and Health Care Service Providers	41:80:0E:6F:5E:92:32:82:CC:BC 5A:E5:F7:50:83:A5:D3:D5:AF:87	BB:D9:7D:8B:49:FF:E9:0F:65:95 08:07:2E:FC:80:72:33:02:27:FB 53:78:90:43:2C:84:06:04:FD:F8 80:A8

Current software products use typically SHA-1 or SHA-256 fingerprints.

Older software products might still use MD5 fingerprints but use of MD5 is discouraged, thus MD5 fingerprints are not anymore provided here.

9.2. Root and CA Certificate AIA and CDP uris

Root and CA certificates:

CA	Authority Information Access-calssuers
VRK Gov. Root CA	http://proxy.fineid.fi/ca/vrkrootc.crt
VRK Gov. CA for Citizen Qualified Certificates	http://proxy.fineid.fi/ca/vrkcqc.crt
VRK Gov. CA for Citizen Qualified Certificates – G2	http://proxy.fineid.fi/ca/vrkcqc2.crt
VRK CA for Qualified Certificates	http://proxy.fineid.fi/ca/vrkcqc.crt
VRK CA for Qualified Certificates – G2	http://proxy.fineid.fi/ca/vrkcqc2.crt
VRK CA for Service Providers	http://proxy.fineid.fi/ca/vrksp.crt
VRK CA for Service Providers – G2	http://proxy.fineid.fi/ca/vrksp2.crt
VRK CA for Service Providers – G3	http://proxy.fineid.fi/ca/vrksp3.crt
VRK CA for Temporary Certificates	http://proxy.fineid.fi/ca/vrktc.crt
VRK CA for Healthcare Professionals Qualified Certificates	http://proxy.fineid.fi/ca/vrkhcp.crt
VRK CA for Healthcare Professionals Temporary Certificates	http://proxy.fineid.fi/ca/vrkhctc.crt
VRK CA for Healthcare Service Providers	http://proxy.fineid.fi/ca/vrkhcsp.crt
VRK CA for Social Welfare and Health Care Service Providers	http://proxy.fineid.fi/ca/vrkshsp.crt

Test Root and CA certificates:

CA	Authority Information Access-calssuers
VRK TEST Root CA	http://proxy.fineid.fi/ca/vrktestc.crt
VRK CA for Test Purposes	http://proxy.fineid.fi/ca/vrktp.crt
VRK CA for Test Purposes – G2	http://proxy.fineid.fi/ca/vrktp2.crt
VRK CA for Test Purposes – G3	http://proxy.fineid.fi/ca/vrktp3.crt
VRK TEST CA for Healthcare Professionals	http://proxy.fineid.fi/ca/vrkthcp.crt
VRK TEST CA for Healthcare Service Providers	http://proxy.fineid.fi/ca/vrkthsp.crt
VRK TEST CA for Social Welfare and Health Care Service Providers	http://proxy.fineid.fi/ca/vrktshsp.crt

Root and CA certificates:

CA	ARL/CRL distribution points
VRK Gov. Root CA	http://proxy.fineid.fi/arl/vrkroota.crl
VRK Gov. CA for Citizen Qualified Certificates	http://proxy.fineid.fi/crl/vrkcqcc.crl
VRK Gov. CA for Citizen Qualified Certificates – G2	http://proxy.fineid.fi/crl/vrkcqc2c.crl
VRK CA for Qualified Certificates	http://proxy.fineid.fi/crl/vrkqcc.crl
VRK CA for Qualified Certificates – G2	http://proxy.fineid.fi/crl/vrkqc2c.crl
VRK CA for Service Providers	http://proxy.fineid.fi/crl/vrkspc.crl
VRK CA for Service Providers – G2	http://proxy.fineid.fi/crl/vrksp2c.crl
VRK CA for Service Providers – G3	http://proxy.fineid.fi/crl/vrksp3c.crl
VRK CA for Temporary Certificates	http://proxy.fineid.fi/crl/vrktcc.crl
VRK CA for Healthcare Professionals Qualified Certificates	http://proxy.fineid.fi/crl/vrkhcpc.crl
VRK CA for Healthcare Professionals Temporary Certificates	http://proxy.fineid.fi/crl/vrkhctcc.crl
VRK CA for Healthcare Service Providers	http://proxy.fineid.fi/crl/vrkhcspc.crl
VRK CA for Social Welfare and Health Care Service Providers	http://proxy.fineid.fi/crl/vrkshspc.crl

Test Root and CA certificates:

CA	ARL/CRL distribution points
VRK TEST Root CA	http://proxy.fineid.fi/arl/vrktesta.crl
VRK CA for Test Purposes	http://proxy.fineid.fi/crl/vrktpc.crl
VRK CA for Test Purposes – G2	http://proxy.fineid.fi/crl/vrktp2c.crl
VRK CA for Test Purposes – G3	http://proxy.fineid.fi/crl/vrktp3c.crl
VRK TEST CA for Healthcare Professionals	http://proxy.fineid.fi/crl/vrkthcpc.crl
VRK TEST CA for Healthcare Service Providers	http://proxy.fineid.fi/crl/vrkthspc.crl
VRK TEST CA for Social Welfare and Health Care Service Providers	http://proxy.fineid.fi/crl/vrktshspc.crl

9.3. CA Certificate OCSP URLs

CA	Authority Information Access-calssuers
VRK Gov. CA for Citizen Qualified Certificates	http://ocsp.fineid.fi/vrkcqc
VRK Gov. CA for Citizen Qualified Certificates – G2	http://ocsp.fineid.fi/vrkcqc2
VRK CA for Qualified Certificates	http://ocsp.fineid.fi/vrkcqc
VRK CA for Qualified Certificates – G2	http://ocsp.fineid.fi/vrkcqc2
VRK CA for Service Providers	http://ocsp.fineid.fi/vrksp
VRK CA for Service Providers – G2	http://ocsp.fineid.fi/vrksp2
VRK CA for Service Providers – G3	http://ocsp.fineid.fi/vrksp3
VRK CA for Temporary Certificates	http://ocsp.fineid.fi/vrktc
VRK CA for Healthcare Professionals Qualified Certificates	http://ocsp.fineid.fi/vrkhcp
VRK CA for Healthcare Professionals Temporary Certificates	http://ocsp.fineid.fi/vrkhctc
VRK CA for Healthcare Service Providers	http://ocsp.fineid.fi/vrkhcsp
VRK CA for Social Welfare and Health Care Service Providers	http://ocsp.fineid.fi/vrkshsp

Test CA certificates:

CA	Authority Information Access-calssuers
VRK CA for Test Purposes	http://ocsptest.fineid.fi/vrktp
VRK CA for Test Purposes – G2	http://ocsptest.fineid.fi/vrktp2
VRK CA for Test Purposes – G3	http://ocsptest.fineid.fi/vrktp3
VRK TEST CA for Healthcare Professionals	http://ocsptest.fineid.fi/vrkthcp
VRK TEST CA for Healthcare Service Providers	http://ocsptest.fineid.fi/vrkthsp
VRK TEST CA for Social Welfare and Health Care Service Providers	http://ocsptest.fineid.fi/vrktshsp

10. Root, CA and End Entity Certificate examples and example of Certificate Revocation List

Some examples of different types of certificates are provided here for a reference. Please note that even though in the examples of Citizen certificate and User certificate for Organizational Use SHA1WithRSA signatures are present, all of the newly created certificates use SHA256WithRSA and no new certificates with SHA1WithRSA are issued anymore.

10.1. Root Certificate

```
SEQUENCE {
   SEQUENCE {
      CONTEXT-SPECIFIC [ 0 ] {
         INTEGER { 2 }  -- x509v3 certificate
      },
      INTEGER { 120000 },  -- Certificate serial number
      SEQUENCE {
         OBJECT IDENTIFIER { "1.2.840.113549.1.1.5" },  -- SHA-1 with RSA Encryption
         NULL { "NULL" }
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT IDENTIFIER { "2.5.4.6" },  -- id-at-countryName
               PrintableString { "FI" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT IDENTIFIER { "2.5.4.8" },  -- id-at-stateOrProvinceName
               PrintableString { "Finland" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT IDENTIFIER { "2.5.4.10" },  -- id-at-organizationName
               PrintableString { "Vaestorekisterikeskus TEST" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT IDENTIFIER { "2.5.4.11" },  -- id-at-organizationalUnitName
               PrintableString { "Certification Authority Services" }
            }
         },
         SET {
            SEQUENCE {
```

```
         OBJECT IDENTIFIER { "2.5.4.11" },  -- id-at-organizationalUnitName
         PrintableString { "Varmennepalvelut" }
      }
   },
   SET {
      SEQUENCE {
         OBJECT IDENTIFIER { "2.5.4.3" },  -- id-at-commonName
         PrintableString { "VRK TEST Root CA" }
      }
   }
},
SEQUENCE {
   UTCTime { "021217190044Z" },  -- not before
   UTCTime { "231217185850Z" }   -- not after
},
SEQUENCE {
   SET {
      SEQUENCE {
         OBJECT IDENTIFIER { "2.5.4.6" },  -- id-at-countryName
         PrintableString { "FI" }
      }
   },
   SET {
      SEQUENCE {
         OBJECT IDENTIFIER { "2.5.4.8" },  -- id-at-stateOrProvinceName
         PrintableString { "Finland" }
      }
   },
   SET {
      SEQUENCE {
         OBJECT IDENTIFIER { "2.5.4.10" },  -- id-at-organizationName
         PrintableString { "Vaestorekisterikeskus TEST" }
      }
   },
   SET {
      SEQUENCE {
         OBJECT IDENTIFIER { "2.5.4.11" },  -- id-at-organizationalUnitName
         PrintableString { "Certification Authority Services" }
      }
   },
   SET {
      SEQUENCE {
         OBJECT IDENTIFIER { "2.5.4.11" },  -- id-at-organizationalUnitName
         PrintableString { "Varmennepalvelut" }
      }
   },
   SET {
      SEQUENCE {
         OBJECT IDENTIFIER { "2.5.4.3" },  -- id-at-commonName
         PrintableString { "VRK TEST Root CA" }
      }
   }
},
SEQUENCE {
   SEQUENCE {
      OBJECT IDENTIFIER { "1.2.840.113549.1.1.1" },  -- RSA encryption
      NULL { "NULL" }
   },
   BIT STRING [ "PRIMITIVE" ] {
      #00,
      SEQUENCE {
         INTEGER {
            #00D4C5A379A4D050AA1B409BF5DE557FAA6C5D882F0880BA692DEBC03C
            25CACA3B13AE51E5DCDDFE9258AFE7FB2D3064EFF1AFF73889B1252537EF
            AFED90DF70B96788074B637026D2BEFDF6E2C2B67AEBB1D49022B5D6A7A1
            602E1D15E47B2B4D7128C1A8B03BA3A7F9ECE9EB25760062D3FDF418D885
            54AB3B29FD7D42303EBC6C0C56D0DD35505D8892ED631C55EB4832987BC3
            F3627083550576F975076124479CAB35DCC135B79E44617A70111B2C952F
            AC790D079CFCAE7836AAACFAAE6B9205697B4AA0A1411BFFA8FF3C533FA7
            65A6A5DD30A383E21A19BA14461D68756FAEFC9C101952F30AEB6356ACA2
            A8C30FC2EEB9354083BE8127C5E215A8F24B
         },
         INTEGER { 65537 }  -- exponent
      }
   }
},
CONTEXT-SPECIFIC [ 3 ] {
   SEQUENCE {
      SEQUENCE {
         OBJECT IDENTIFIER { "2.5.29.19" },  -- Basic Constraints
         BOOLEAN {
            #FF  -- critical
         },
         OCTET STRING [ "PRIMITIVE" ] {
            SEQUENCE {
               BOOLEAN {
                  #FF  -- CA Certificate=True
               }
            }
         }
      },
      SEQUENCE {
         OBJECT IDENTIFIER { "2.16.840.1.113730.1.1" },  -- Netscape certificate type
         OCTET STRING [ "PRIMITIVE" ] {
            BIT STRING {
               #00,
                     #07  -- Object signing CA, S/MIME CA, SSL CA
                  }
               }
            },
            SEQUENCE {
               OBJECT IDENTIFIER { "2.5.29.15" },  -- Key Usage
               BOOLEAN {
                  #FF  -- critical
               },
               OCTET STRING [ "PRIMITIVE" ] {
                  BIT STRING {
                     #01,
                     #C6  -- digitalSignature, nonRepudiation, keyCertSign, cRLSign
                  }
               }
            },
            SEQUENCE {
               OBJECT IDENTIFIER { "2.5.29.14" },  -- Subject Key Identifier
               OCTET STRING [ "PRIMITIVE" ] {
                  OCTET STRING {
                     #DEDE500B1767C337A9C998883448C93179EB6DA1
                  }
               }
            }
         }
      }
   },
   SEQUENCE {
      OBJECT IDENTIFIER { "1.2.840.113549.1.1.5" },  -- SHA-1 with RSA Encryption
      NULL { "NULL" }
   },
   BIT STRING {
      #00,
      #54A87EDD1EA0DFF5F579CB1A158A2E0BDE47F79391A257EDAA7ADDF7DF
      B6795E932F5EC5848016E9F9B3C470EFF05FF162DFB4C6BF62F1A97D5EB8
      6B372ACDAB36CA8DC728ABD33C739E8096E6E6D312DEA562F0F19FEA3C58
      70F78015730383D4807B1E862F1239BA85DF959DD38838FAE02BC24B11CF
      093C1EDB26E9AD5AD1B62E8BC023B58843931A3BF1B944B156144A732C19
      5209F5A1599C8A6EB702FBDF0DC0FC62DC7CC5394F3E789B4E4433970696
      30217E9038AD92C15B90E51D59856858591BE2F8ACFC161CE99CA317559B
      D7211836F55F2F8FF5EE48999D91B12CD6CCE18826DAF50467CB9A944B57
      7C509FBFD0C0F0C7952147DADA94C7E939
   }
}
```

10.2. CA Certificate

```
SEQUENCE {
   SEQUENCE {
      CONTEXT_SPECIFIC [ 0 ] {
        INTEGER =  2; -- x509v3 certificate
      },
      INTEGER =  127666; -- Certificate serial number
      SEQUENCE {
         OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" }, -- SHA-256 with RSA Encryption
         NULL =  "NULL";
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.8";   -- id-at-stateOrProvinceName
               PrintableString { "Finland" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               PrintableString { "Vaestorekisterikeskus TEST" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Certification Authority Services" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Varmennepalvelut" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               PrintableString { "VRK TEST Root CA" }
            }
         }
      },
      SEQUENCE {
         UTCTime { "151113103903Z" }, -- not before
         UTCTime { "231217185850Z" }  -- not after
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               PrintableString { "Vaestorekisterikeskus TEST" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Testivarmenteet" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               PrintableString { "VRK CA for Test Purposes - G3" }
            }
         }
      },
      SEQUENCE {
         SEQUENCE {
            OBJECT_IDENTIFIER { "1.2.840.113549.1.1.1" },   -- RSA encryption
            NULL =  "NULL";
         },
         BIT_STRING [ PRIMITIVE ] {
            #00,
            SEQUENCE {
               INTEGER {
                  #00BA386FFCF493A01D94F76463720D2DFF8075FEE1BB06E30E981A39FA
                  EAEBE6C4043D27A89539617A985F486117CD3030E4D6D35F91D5DDCBA252
                  7C945B06A11D1564C62BB364C4C170DB112131B019EF1B67A930A0F60C9B
                  F7CB361651046DB81DCD16981FD6E90C3515DC4277D96AF90C7684633FFC
                  E49E613273438991C0667C7776C836A2098A6D1405E243E63DEFD5FBC58D
                  6D14CAB2668FFC0D31FA21202D778FB9E98A1402809E85570FC34B4A5DB9
```

```
                  77F894F4084A7426739FBBDFDF03CADEC892ECE1380828568FE4A8957E0A
                  8894F99D9F74449DBE2E1B06D0578A35C5355A6F26BE1452FA2BF2956D53
                  B0DE55B5798B36502EBA2A12FE69F6F5E9E67D3BA226DFE9CAE881440851
                  AD7E11F9F98207A70ECA14257AA3E22A37BBF7E06663C1CCF1091E59A810
                  C4CE76895914F8CAA011AEB7F7B56084AA2C2F46EB4BDEAD2F55F120172E
                  A8C5BD62349D1069AAFF3FDF9DB4D1FFCD3EAB2C250490D0CD849207B8A1
                  E4129CC3FDAD58CF5CF85130CA682F3944A62B2B3FB841F5A2CB3804E972
                  B1A824C65970E800D89A91AC53BB8B87C6CEBE5CF88B57B4BAA27FD75988
                  5E345126993C09AC3680B570C7F6CF587419E83FEE1F74BFB32114C01CD6
                  EBB0EF089AA5A0C1BCBE3EB572FA19B6B3469573892101467A64350ACAE8
                  B0B0CBEC6EAED1CD1B0178CFC658AF49F7457E989C37051884C41949B15F
                  21B55187
               },
               INTEGER =  65537; -- exponent
            }
            }
         }
      },
      CONTEXT_SPECIFIC [ 3 ] {
         SEQUENCE {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.35";   -- Authority Key Identifier
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     CONTEXT_SPECIFIC [ 0, "IMPLICIT" ] {
#DEDE500B1767C337A9C998883448C93179EB6DA1 }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.14";   -- Subject Key Identifier
               OCTET_STRING [ PRIMITIVE ] {
                  OCTET_STRING {
                     #5BCE869CC75343E602B9FB716C8C6DA320E5B1F8
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.15";   -- Key Usage
               BOOLEAN =  #FF; -- critical
               OCTET_STRING [ PRIMITIVE ] {
                  BIT_STRING {
                    #01,
                    #06 -- keyCertSign, cRLSign
                 }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.32";   -- Certificate Policies
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.2.246.517.99.10.31.1" }, -- VRK Test G3 CPS
                        SEQUENCE {
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.2" },  -- User notice
                              SEQUENCE {
                                 VisibleString {
                                    "Varmennepolitiikka on saatavilla - Certifikat policy
finns - Certificate policy is available http://www.fineid.fi/cps99"
                                 }
                              }
                           },
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.1" }, -- CPS
                              IA5String { "http://www.fineid.fi/cps99/" }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.19";   -- Basic Constraints
               BOOLEAN =  #FF; -- critical
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     BOOLEAN =  #FF; -- CA Certificate=True
                     INTEGER =  0;   -- pathLenConstraint
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.31";   -- CRL Distribution Points
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        CONTEXT_SPECIFIC [ 0 ] {
                           CONTEXT_SPECIFIC [ 0 ] {
                              CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://proxy.fineid.fi/arl/vrktesta.crl" }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.1.1" }, -- Authority Info Access
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.48.2" }, -- CA Issuers
                        CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://proxy.fineid.fi/ca/vrktestc.crt" }
                     }
                  }
               }
            }
         }
      }
   },
   SEQUENCE {
      OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" }, -- SHA-256 with RSA Encryption
      NULL =  "NULL";
   },
   BIT_STRING {
      #00,
      #08577D12B4677D2E7ED0C9F9712E375565E6F8E5D7ECEBC30E7514718C
      37ABDA5D0DBAF51499CB5B5A0112438ADB94595908AA3A973A77FB61BD6F
      B72BA1DFD8727C8FD34FC909FF4FAABB17654F130F5A7A09E3F311B25BC4
      91C253DBC3D5B847B4903BAAFDCC83FE8C046949304CFB5F103829759182
      04972E6CCC16C6D2FB3B8C318688F8D5C425CECE2C71766104D1BF6424BD
      5220600EEDB57AF6400324A4A3CEB7144C5A21E23421934ED60EA61BAA30
      17B9782CDF25CFFD41003D2AABAC0991A4AD69A57677E393B0931BA0EAA9
      65DD98498A56EB845B449541823861547E827CD43226674839E1158FEE49
      68417986B20C176DD2393C5ED05942A055
   }
      }
```

10.3. Citizen Certificate - Authentication & Encryption

```
SEQUENCE {
   SEQUENCE {
      CONTEXT_SPECIFIC [ 0 ] {
       INTEGER =  2;  -- x509v3 certificate
      },
      INTEGER =  101001326;  -- Certificate serial number
      SEQUENCE {
         OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" },  -- SHA-256 with RSA Encryption
         NULL =  "NULL";
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               PrintableString { "Vaestorekisterikeskus TEST" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Testivarmenteet" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               PrintableString { "VRK CA for Test Purposes - G3" }
            }
         }
      },
      SEQUENCE {
         UTCTime { "161019080245Z" },  -- not before
         UTCTime { "211010205959Z" }   -- not after
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.5";   -- id-at-serialNumber
               PrintableString { "123456789" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.42";   -- id-at-givenName
               PrintableString { "IIKKA" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.4";   -- id-at-surName
               UTF8String { "SPECIMEN-BACÄNO" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               UTF8String { "SPECIMEN-BACÄNO IIKKA 123456789" }
            }
         }
      },
      SEQUENCE {
         SEQUENCE {
            OBJECT_IDENTIFIER { "1.2.840.113549.1.1.1" },   -- RSA encryption
            NULL =  "NULL";
         },
         BIT_STRING [ PRIMITIVE ] {
            #00,
            SEQUENCE {
               INTEGER {
                  #00BEF0A2C04A0D23F504E5A68AC1EC08ACF51728EFC39AA4D74847F391
                  BFE0C1C4E8CEC5DA21846FA0A6B53AEC98E990CBBD8F79697CD8388D5962
                  E454FA2BA88026709D46D34BA37CCC07D99E613A231F96147DD5654C8F51
                  379B28D80937D7F81102505B13F682D44C47A4A6D443AFEAE74456C958A9
                  7EBD293FFCF098AE386C08F51A47ADC39405B4412E87D40803088F34B2B3
                  3D65B7A5FC2E16B6871A7BE21AE9ED3F2D962939AD9DAB05DD6DD4124ED0
                  956BB1517EE2159E0E08100C5F0DE83ACC759558BDAFFB7E469A48E6D936
                  ADBF1DAAD05275782683300E25414CA1D12E5BB0D2CA92E2049BD22DE3FE
                  D77B9B661A2DF23D8385A6E1FF8D27853B57
               },
               INTEGER =  65537;
```

```
            }
         }
      },
      CONTEXT_SPECIFIC [ 3 ] {
         SEQUENCE {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.35";   -- Authority Key Identifier
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     CONTEXT_SPECIFIC [ 0, "IMPLICIT" ] {
#5BCE869CC75343E602B9FB716C8C6DA320E5B1F8 }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.14";   -- Subject Key Identifier
               OCTET_STRING [ PRIMITIVE ] { OCTET_STRING {
#C30EEE8FAE497A6132AFBD711E9763F735A70D7D } }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.15";   -- Key Usage
               BOOLEAN =  #FF;  -- critical
               OCTET_STRING [ PRIMITIVE ] {
                  BIT_STRING { #04, #B0 }  -- digitalSignature, keyEncipherment,
dataEncipherment
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.32";   -- Certificate Policies
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.2.246.517.99.10.32.1" }, -- Test Purposes G3
CPS
                        SEQUENCE {
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.1" },  -- CPS
                              IA5String { "http://www.fineid.fi/cps99/" }
                           },
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.2" },  -- User notice
                              SEQUENCE {
                                 VisibleString {
                                    "Varmennepolitiikka on saatavilla - Certifikat policy
finns - Certificate policy is available http://www.fineid.fi/cps99"
                                 }
                              }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.19";   -- Basic Constraints
               BOOLEAN =  #FF;  -- critical
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE { BOOLEAN =  #00; }  -- CA Certificate=False
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.31";   -- CRL Distribution Points
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        CONTEXT_SPECIFIC [ 0 ] {
                           CONTEXT_SPECIFIC [ 0 ] {
                              CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://proxy.fineid.fi/crl/vrktp3c.crl" }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.1.1" },  -- authorityInfoAccess
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.48.2" },  -- caIssuers
                        CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://proxy.fineid.fi/ca/vrktp3.crt" }
                     },
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.48.1" },  -- OCSP
                        CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://ocsptest.fineid.fi/vrktp3" }
                     }
                  }
               }
            }
         }
      }
   },
   SEQUENCE {
      OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" }, -- sha256withRSA
      NULL =  "NULL";
   },
   BIT_STRING {
      #00,
      #0B27944ED1F792144B0E282B92CFEC705C203A2A5C083DB5BB789C9AA3
      1F49C3F11DF06DBB129A414BA301AC8A6B477C59E178189A08B821C245BB
      5D81DDE3157E290DEADA994C1A91F8A4C59D68882112E3C8B837563F9764
      E06BBD5B8C52F43F782987D2A4BCD162EC3182F72A8A9F981754D1B72810
      13813779BE5040A76F6A9F3C79E78615FAE8323D9058C0F2AE77B0BB0B04
      A7303C57C24758D40B28E3CD2CF98986ED7F85F62F9718D50D0AC031E96A
      4C2C9A60E49FF21FEDAA74F66B650792E6804CEA68FADAA1A85F94B26819
      CAFA08183DEC99824C2CCEE9A3A9F7DCEB7620960C66D2D4C1BF27CDA5D4
      34B7311C2D29E5A827EE3D377105C7BC4DECBB93C430D2FB3F2EACC80BAF
      2A5F9A6BC42355238C905D9269D92DA9E42509C5AC62FF799BA8CC1F2F42
      EFAEABF0DF3CA8A3FDD7F1F34F3067A42843A92530288E94B7561B3416AC
      5875046CD65BC07C6B9E58F86C940F1A83DC21B829723A5CA6C9F5134336
      72D0C8AB942CC2BD67D1C6E5D3E0D83F23569271BDE00EBBA6D4C9AD5EF5
      A82A761EF3920F2E713971C36F3F267672352C03C283EFC7F1C73DED1E4A
      DB14AB469BE0429ECD68FF5DF85A162CD40CDCCD58991183FB381B0BA812
      35D457AEA9BFB186EFBA0A175B9D7FAB02037F0CED07956DED4144A87DF2
      B783C94E7C79BA3D39564FE3E25A37C18D3D51EFED47E35B9CA0391ADFB3
      1E4F03
   }
}
```

10.4. Citizen Certificate - Non Repudiation (RSA)

```
SEQUENCE {
   SEQUENCE {
      CONTEXT_SPECIFIC [ 0 ] {
       INTEGER =  2; -- x509v3 certificate
      },
      INTEGER =  101001327;  -- Certificate serial number
      SEQUENCE {
         OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" }, -- SHA-256 with RSA Encryption
         NULL =  "NULL";
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               PrintableString { "Vaestorekisterikeskus TEST" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Testivarmenteet" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               PrintableString { "VRK CA for Test Purposes - G3" }
            }
         }
      },
      SEQUENCE {
         UTCTime { "161019080248Z" },  -- not before
         UTCTime { "211010205959Z" }   -- not after
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.5";   -- id-at-serialNumber
               PrintableString { "123456789" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.42";   -- id-at-givenName
               PrintableString { "IIKKA" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.4";   -- id-at-surName
               UTF8String { "SPECIMEN-BACÄNO" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               UTF8String { "SPECIMEN-BACÄNO IIKKA 123456789" }
            }
         }
      },
      SEQUENCE {
         SEQUENCE {
            OBJECT_IDENTIFIER { "1.2.840.113549.1.1.1" },   -- RSA encryption
            NULL =  "NULL";
         },
         BIT_STRING [ PRIMITIVE ] {
            #00,
            SEQUENCE {
               INTEGER {
                  #00DA10F8C69F1BBA5B4DD36D429F04453E4333CA17995F30EAD3BEF4ED
                  78DE56FBACA952A887E18902D998D8E8FD9D15DCADEC0687AD9C201D9494
                  A8DD90C4591E86AC3A2C88FB10AC3AA103F806FEFDE4B26903A4FAB7FB67
                  0FCA9D915683BA7CA650396F932EE5A4BD0243508CF8B919837F0E4CAD0A
                  E495AF05B95C4878F98039D4C07F14E6963FEC48060466029DEE8718FDAA
                  78A05F530E8C21483E28DA759E235B46FDFEFFEE1A2E2CCF34B06C1B8569
                  D255805A21CA49F03F8079987D6ADF02653A5B4ABA916CEE112B36FC6F2D
                  A4BCA946D8F13E275E8B1282F439BC16028FB86FDC575A57A9C777662A8F
                  7D7FCF60E735DBD212A1741302F646F1A511
               },
               INTEGER =  65537;
            }
         }
      },
      CONTEXT_SPECIFIC [ 3 ] {
         SEQUENCE {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.35";   -- Authority Key Identifier
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     CONTEXT_SPECIFIC [ 0, "IMPLICIT" ] {
#5BCE869CC75343E602B9FB716C8C6DA320E5B1F8 }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.14";   -- Subject Key Identifier
               OCTET_STRING [ PRIMITIVE ] { OCTET_STRING {
#E391D694D40D15B9883D372D0E00DD2C6F7F84EB } }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.15";   -- Key Usage
               BOOLEAN =  #FF;  -- critical
               OCTET_STRING [ PRIMITIVE ] {
                  BIT_STRING { #06, #40 }  -- nonRepudiation
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.32";   -- Certificate Policies
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.2.246.517.99.10.32.1" }, -- Test Purposes G3
CPS
                        SEQUENCE {
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.1" },  -- CPS
                              IA5String { "http://www.fineid.fi/cps99/" }
                           },
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.2" }, -- User notice
                              SEQUENCE {
                                 VisibleString {
                                    "Varmennepolitiikka on saatavilla - Certifikat policy
finns - Certificate policy is available http://www.fineid.fi/cps99"
                                 }
                              }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.19";   -- Basic Constraints
               BOOLEAN =  #FF;  -- critical
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE { BOOLEAN =  #00; }  -- CA Certificate=False
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.31";   -- CRL Distribution Points
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        CONTEXT_SPECIFIC [ 0 ] {
                           CONTEXT_SPECIFIC [ 0 ] {
                              CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://proxy.fineid.fi/crl/vrktp3c.crl" }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.1.1" },  -- CA Information Access
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.48.2" },  -- caIssuers
                        CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://proxy.fineid.fi/ca/vrktp3.crt" }
                     },
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.48.1" },  -- OCSP
                        CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://ocsptest.fineid.fi/vrktp3" }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.1.3" },  -- qcStatements
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE { OBJECT_IDENTIFIER { "0.4.0.1862.1.1" } }  -- qcs-QcCompliance
                  }
               }
            }
```

```
         }
      }
   },
   SEQUENCE {
      OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" },  -- sha256WithRSAEncryption
      NULL =  "NULL";
   },
   BIT_STRING {
      #00,
      #B2B37D7B588845E7CF87823B4EAD4F4AC7040859B2DF2EA4456EEF00E2
      98C12E222F36953532C93887E57C0C68148BBA00CB381F21039C7ADA5ABE
      9793A6F4A5AB5848ACBF73C82A89EBCA3387DBCC8E1978DA63E3B0F6F4D5
      1FFBD4930F54C72782CDE5A49D890FC0940557F072AA6946B86C3ECFF51B
      DC14A1CCF13B175C635476437C47F8832C99B384F34EAF9D7FCDACD5E955
      38C8C6A9C24DAF4B271B05DF28CCFBFD7BCCB987AC59F580CCF652312DED
      7E05052968EA84E3182D078A92494DF5D0154E2236BEF7D166FDCC8F91BE
      461A4659C68F2AE0E7614207B2A26B3B3D28D7A3522E29F0AF461AD802B2
      A3D14DB4E7B72CCB299A41D4319338CB3C60E0DC81F912183DF3E6073688
      E442176505656F68DC5E7F2831CE9A5B1D6573E11E0335133DB8757E3CBF
      E4B5CF6B2C782389BE401EDFF9CB4A632947A0A1FF645F9E7D3F22F79A30
      F6E325E77688BC6CB957C39A27A108850AFBE9C539E647619BB5BE72BE77
      96955E102DC916BD1E81E14B406F4D17B0B2A57E49E43D70113F8EF21E9B
      F1FD375FB37AC377D15BF169685E9C5150F497A5129F4A489D5642B56421
      DD532F0332125339B4585EFBC1D50B89C3787BC8C0B9368E0AE28667C16D
      1898539016BDD1FE7C3CDDDF5B769AA0D4C332383CF5DA2B698FAEF15199
      8CF507F9B73B69FDCFB215F9216639613FB9C1AFEEE9F1279F2C511F48AD
      2D1830
   }
}
```

10.5. Citizen Certificate - Non Repudiation (ECC)

```
SEQUENCE {
   SEQUENCE {
      CONTEXT_SPECIFIC [ 0 ] {
        INTEGER =  2; -- x509v3 certificate
      },
      INTEGER =  101001328;  -- Certificate serial number
      SEQUENCE {
         OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" }, -- SHA-256 with RSA Encryption
         NULL =  "NULL";
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               PrintableString { "Vaestorekisterikeskus TEST" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Testivarmenteet" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               PrintableString { "VRK CA for Test Purposes - G3" }
            }
         }
      },
      SEQUENCE {
         UTCTime { "161019080252Z" },  -- not before
         UTCTime { "211010205959Z" }   -- not after
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.5";   -- id-at-serialNumber
               PrintableString { "123456789" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.42";   -- id-at-givenName
               PrintableString { "IIKKA" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.4";   -- id-at-surName
               UTF8String { "SPECIMEN-BACÄNO" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               UTF8String { "SPECIMEN-BACÄNO IIKKA 123456789" }
            }
         }
      },
      SEQUENCE {
         SEQUENCE {
            OBJECT_IDENTIFIER { "1.2.840.10045.2.1" },   -- Elliptic curve cryptography
            OBJECT_IDENTIFIER { "1.2.840.10045.3.1.7" }  -- 256bit curve szOID_ECC_CURVE_P256
         },
         BIT_STRING {
            #00,
            #04AF5CA6E95DCDF9E5EB187E4E4C7EDE66C552F19037A5000EC4C09B15
            798AC23F776BB8320AF54B5A8BFA88096E1F7F8B65B66A1A1E3187F3B6DD
            745788FBB734
         }
      },
      CONTEXT_SPECIFIC [ 3 ] {
         SEQUENCE {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.35";   -- Authority Key Identifier
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     CONTEXT_SPECIFIC [ 0, "IMPLICIT" ] {
#5BCE869CC75343E602B9FB716C8C6DA320E5B1F8 }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.14";   -- Subject Key Identifier
               OCTET_STRING [ PRIMITIVE ] { OCTET_STRING {
#6680C4916CC5BCC14C3B6C4B4A2D1200732A8C66 } }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.15";   -- Key Usage
               BOOLEAN =  #FF;  -- critical
               OCTET_STRING [ PRIMITIVE ] {
                  BIT_STRING { #06, #40 } -- nonRepudiation
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.32";   -- Certificate Policies
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.2.246.517.99.10.32.1" }, -- Test Purposes G3
CPS
                        SEQUENCE {
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.1" },  -- CPS
                              IA5String { "http://www.fineid.fi/cps99/" }
                           },
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.2" }, -- User notice
                              SEQUENCE {
                                 VisibleString {
                                    "Varmennepolitiikka on saatavilla - Certifikat policy
finns - Certificate policy is available http://www.fineid.fi/cps99"
                                 }
                              }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.19";   -- Basic Constraints
               BOOLEAN =  #FF;  -- critical
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE { BOOLEAN =  #00; }  -- CA Certificate=False
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.31";   -- CRL Distribution Points
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        CONTEXT_SPECIFIC [ 0 ] {
                           CONTEXT_SPECIFIC [ 0 ] {
                              CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://proxy.fineid.fi/crl/vrktp3c.crl" }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.1.1" },  -- CA Information Access
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.48.2" }, -- CA Issuers
                        CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://proxy.fineid.fi/ca/vrktp3.crt" }
                     },
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.48.1" },  -- OCSP
                        CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://ocsptest.fineid.fi/vrktp3" }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.1.3" },  -- qcStatements
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE { OBJECT_IDENTIFIER { "0.4.0.1862.1.1" } }  -- QcCompliance
                  }
               }
            }
         }
      }
   },
   SEQUENCE {
      OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" }, -- sha256WithRSAEncryption
      NULL =  "NULL";
   },
   BIT_STRING {
      #00,
      #6D4284098F09783CA288408C78666CFBBE24BC79E0B0B4AAD3CDADD7E2
      881ADC175DA260C47AD89A7569259A8AC996532BD09414D8DC79291D8030
      DD4D03CB3801DCB8543BF3C4DE9772C11AADFE1462C80F37A249EEDDE474
      33024AFBFB48E221E81424BCAE306364E740DABD0B5BFD52C029E728D655
      DD8D76B4F28BDD06F95CBBCEFC9ED8D081AF01A660BF2D1E4F6B19BB95B4
      43D34B70A15BC0D6779A26E117960A35E84146048525DCE2E84B7C187864
      4D5DB77321FB90AB353DD4C2F08B9DF771CD2F3B3B5AC07D44273ED12853
      848255C8A9FD4ECC859D7EBF019CD8046F6770D7FAE9F4FC324D43EBE176
      59E6E0542CEB3B7A141AA58162827E89D92298F2C22A5652CF3AC6BA3B38
      2BA3FDE54423455D7FFDAA35A0D26E894EC179779A33E19F02401E59D59F
      FF4BFE4547186C93A87F24C380D3219F15CDDDB10A96BC108FAAD8716C76
      1AB748B83A21E00F7FA78FC918EFB947142F1D60CE0BFF8141156D07EA86
      680CF1B46DA88437E9492BE7F6463E52B5EF15D2E7FCFA592CC4981DC1AA
      1BE43D52088AC5610289A260C7BA47286A72B7E64A8F6664ED17E0965343
      91C7894579AF30C9C1E95E7099A28056D43360052A8E2D873990BDB3E438
      38FD8B498EBBD01F34D0DC18DB94D71C55CEEEEDCF1105BA0842CA472E2C
      9DF8C7F9F4DC3EFD2CA356FD13E72864ABABBA66DC5DD949DE6A5191F8E4
      6ADD54
   }
}
```

10.6. User Certificate for Organizational usage - Authentication & Encryption

```
SEQUENCE {
   SEQUENCE {
      CONTEXT_SPECIFIC [ 0 ] {
        INTEGER =  2;  -- x509v3 certificate
      },
      INTEGER =  100117246;  -- Certificate serial number
      SEQUENCE {
         OBJECT_IDENTIFIER { "1.2.840.113549.1.1.5" },   -- SHA-1 with RSA Encryption
         NULL =  "NULL";
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.8";   -- id-at-stateOrProvinceName
               PrintableString { "Finland" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               PrintableString { "Vaestorekisterikeskus TEST" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Testivarmenteet" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               PrintableString { "VRK CA for Test Purposes" }
            }
         }
      },
      SEQUENCE {
         UTCTime { "110310104516Z" },  -- not before
         UTCTime { "160310215959Z" }   -- not after
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Testaus ja Kokeilu" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               PrintableString { "Oy Testi Ab" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.12";   -- id-at-title
               UTF8String { "Testauspäällikkö" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.5";   -- id-at-serialNumber
               PrintableString { "997548070" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.42";   -- id-at-givenName
               UTF8String { "2k-avaimetååö009" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.4";   -- id-at-surName
               UTF8String { "JAVA-PRÅVSTRÖM-ORG" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               UTF8String {
                  "JAVA-PRÅVSTRÖM-ORG 2k-avaimetååö009 997548070"
               }
            }
         }
      },
      SEQUENCE {
         SEQUENCE {
            OBJECT_IDENTIFIER { "1.2.840.113549.1.1.1" },   -- RSA encryption
            NULL =  "NULL";
         },
         BIT_STRING [ PRIMITIVE ] {
            #00,
            SEQUENCE {
               INTEGER {
                  #00AE1EA04708B7593EBDBB3B05B30E1E0EDE9871D959A458B8A3DF6653
                  CB25E7CF0209748D1B60E1C928598EB63961F2E1EE9F965C02F2219170F3
                  233FD4F2410A393FD0B7E1B2ED8D76AD33FDB2A3746137BC8D3115BDEAAB
                  D4DAE3F916804AB3E3D06CFB495FB99688E73F3364DC8A4C06C4BAF252E2
                  46BD5DFDE458A577482A2E7BC9A9ACB95CD3DA202BC3D273F79B02B08AFC
                  24503C9AADF80FB2BDAE0F12E285AB6591FC211441946365A396E147CC93
                  6927E36316988364BB0044FE49D83B96D4088FD6D1BBA5BA7BE3A7082B4D
                  AF5A94ABDC6674E09997ADAD73F81E0723EC8C26E83F2C969A8A99835891
                  ADAE11196F48E6DE7FA4983DD35E2D036277
               },
               INTEGER =  65537; -- exponent
            }
         }
      },
      CONTEXT_SPECIFIC [ 3 ] {
         SEQUENCE {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.19";   -- Basic Constraints
               BOOLEAN =  #FF; -- critical
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE { BOOLEAN =  #00; }  -- CA Certificate=False
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.32";   -- Certificate Policies
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.2.246.517.99.10.3.1" },-- VRK Test CPS
                        SEQUENCE {
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.2" },-- User notice
                              SEQUENCE {
```

```
                                 VisibleString {
                                    "Tutustu varmennepolitiikkaan - se certifikat policy
http://www.fineid.fi/cps99"
                                 }
                              }
                           },
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.1" },  -- CPS
                              IA5String { "http://www.fineid.fi/cps99/" }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.1.1" },   -- Authority Info Access
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.48.2" },   -- CA Issuers
                        CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://proxy.fineid.fi/ca/vrktp.crt" }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.17";   -- Subject Alternative Name
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     CONTEXT_SPECIFIC [ 1, "IMPLICIT" ] { "2kavaimet009.java-
pravstrom@testi.fi" },
                     CONTEXT_SPECIFIC [ 0 ] {
                        OBJECT_IDENTIFIER { "1.3.6.1.4.1.311.20.2.3" },   -- MS User Principal
Name (UPN)
                        CONTEXT_SPECIFIC [ 0 ] {
                           UTF8String { "2kavaimet009.java-pravstrom@testi.fi" }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER { "2.16.840.1.113730.1.1" },   -- Netscape certificate type
               OCTET_STRING [ PRIMITIVE ] {
                  BIT_STRING { #05, #A0 }   -- S/MIME client, SSL client auth
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.15";   -- Key Usage
               BOOLEAN =  #FF;   -- critical
               OCTET_STRING [ PRIMITIVE ] {
                  BIT_STRING { #04, #B0 }   -- digitalSignature, keyEncipherment, 
dataEncipherment
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.37";   -- Extended Key Usage
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.3.2" },   -- Client authentication
                     OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.3.4" },   -- eMail protection
                     OBJECT_IDENTIFIER { "1.3.6.1.4.1.311.20.2.2" }   -- MS SmartCard Logon
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.35";   -- Authority Key Identifier
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     CONTEXT_SPECIFIC [ 0, "IMPLICIT" ] { 
#7231188759FE93E79E55F9C04AE18FF34A96580E }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.31";   -- CRL Distribution Points
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        CONTEXT_SPECIFIC [ 0 ] {
                           CONTEXT_SPECIFIC [ 0 ] {
                              CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] { 
"http://proxy.fineid.fi/crl/vrktpc.crl" }
                           }
                        }
                     },
                     SEQUENCE {
                        CONTEXT_SPECIFIC [ 0 ] {
                           CONTEXT_SPECIFIC [ 0 ] {
                              CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] { 

"ldap://ldap.fineid.fi:389/cn%3dVRK%20CA%20for%20Test%20Purposes,ou%3dTestivarmenteet,o%3dVaes
torekisterikeskus%20TEST,dmdName%3dFINEID,c%3dFI?certificateRevocationList"
                              }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.14";   -- Subject Key Identifier
               OCTET_STRING [ PRIMITIVE ] { OCTET_STRING {
#75CDE567FBB276199CC32622F3A3F760129C640B } }
            }
         }
      }
   },
   SEQUENCE {
      OBJECT_IDENTIFIER { "1.2.840.113549.1.1.5" },   -- SHA-1 with RSA Encryption
      NULL =  "NULL";
   },
   BIT_STRING {
      #00,
      #47269597C8CC644F6356162CBFA42D158A4DF8DD1FA4A2A11B7C98A8F4
      771D00BFB9D4AFDE8D6E8A1F7A79CE6D4D7AC367855B5F07C6E5A0D3C9DD
      D5DF21B9608C41AA1D69F42426F868E73DE62DFAF4B6FAC833DDDEB58CB5
      AD10AFAF0F0AD14EE97528A294C58508C5073C720C1EF7069A4DAA256863
      199DA85B8F84C4EF9649DF3FFB64544CFD2EDD04004C2D44D91EB471F027
      5483B67ABD040E8EAEFD3D3B161E591027CBE389B4E9AD86BF501EFD8D0E
      0D8238F87E0914AF43D7D60A65D85AD0D308F26BCE77299A3A81732EC2C3
      ADD00EA473D34C54BB30EBD7AC782703451CE9E24E9F1F59734BBB516BC5
      D650FFD5178806BBE0F3E48EB1EC4BBE55
   }
}
```

10.7. User Certificate for Organizational usage – Non Repudiation

```
SEQUENCE {
   SEQUENCE {
      CONTEXT_SPECIFIC [ 0 ] {
       INTEGER =  2;  -- x509v3 certificate
      },
      INTEGER =  100117243;   -- Certificate serial number
      SEQUENCE {
         OBJECT_IDENTIFIER { "1.2.840.113549.1.1.5" },   -- SHA-1 with RSA Encryption
         NULL =  "NULL";
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.8";   -- id-at-stateOrProvinceName
               PrintableString { "Finland" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               PrintableString { "Vaestorekisterikeskus TEST" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Testivarmenteet" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               PrintableString { "VRK CA for Test Purposes" }
            }
         }
      },
      SEQUENCE {
         UTCTime { "110310104502Z" },   -- not before
         UTCTime { "160310215959Z" }    -- not after
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Testaus ja Kokeilu" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               PrintableString { "Oy Testi Ab" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.12";   -- id-at-title
               UTF8String { "Testauspäällikkö" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.5";   -- id-at-serialNumber
               PrintableString { "997548070" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.42";   -- id-at-givenName
               UTF8String { "2k-avaimetååö009" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.4";   -- id-at-surName
               UTF8String { "JAVA-PRÅVSTRÖM-ORG" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               UTF8String {
```

```
            "JAVA-PRÅVSTRÖM-ORG 2k-avaimetåäö009 997548070"
         }
      }
   }
},
SEQUENCE {
   SEQUENCE {
      OBJECT_IDENTIFIER { "1.2.840.113549.1.1.1" },   -- RSA encryption
      NULL =  "NULL";
   },
   BIT_STRING [ PRIMITIVE ] {
      #00,
      SEQUENCE {
         INTEGER {
            #00B1DDF366C658AEA3D904B9FD5C8B2F0BE359A371F1B123B43632694F
            01837D4F72824266055D3AA610142AF692A83F585F76E414A9032DE16BEA
            63676A45B8B51E3057B8228A35BFEAC7D69C69F4569035A6D24817E4930A
            4E1170F93825D6A47A9DA9E38EFA1F4FCDD4FB0001874D33156D090510CC
            1C21E5B72C9802D51866679C241547619361CCA508A7BFC49D5DC39CAE96
            40D87F9549B33E0F8B964CFCA85143B65229459B5D1FA9730A08658E3233
            F0BEA7D874147E164A317E949B4CB564834C3C88F7343F89AA8C9BA7E3F8
            0527C3DA7A4CE5A16A96CC2BCB2DC53D8F1AED5FC41D3460D202F9CED370
            D18D4BECA5C179A03134159E4CBEE6B2DA6B
         },
         INTEGER =  65537;   -- exponent
      }
   }
},
CONTEXT_SPECIFIC [ 3 ] {
   SEQUENCE {
      SEQUENCE {
         OBJECT_IDENTIFIER =  "2.5.29.19";   -- Basic Constraints
         BOOLEAN =  #FF;   -- critical
         OCTET_STRING [ PRIMITIVE ] {
            SEQUENCE { BOOLEAN =  #00; }  -- CA Certificate=False
         }
      },
      SEQUENCE {
         OBJECT_IDENTIFIER =  "2.5.29.32";   -- Certificate Policies
         OCTET_STRING [ PRIMITIVE ] {
            SEQUENCE {
               SEQUENCE {
                  OBJECT_IDENTIFIER { "1.2.246.517.99.10.3.1" },   -- VRK Test CPS
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.2" },   -- User notice
                        SEQUENCE {
                           VisibleString {
```

```
                                    "Tutustu varmennepolitiikkaan - se certifikat policy
http://www.fineid.fi/cps99"
                                 }
                              }
                           },
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.1" },   -- CPS
                              IA5String { "http://www.fineid.fi/cps99/" }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.1.1" },   -- Authority Info Access
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.48.2" },   -- CA Issuers
                        CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://proxy.fineid.fi/ca/vrktp.crt" }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.17";   -- Subject Alternative Name
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     CONTEXT_SPECIFIC [ 1, "IMPLICIT" ] { "2kavaimet009.java-
pravstrom@testi.fi" }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.1.3" },   id-pe-qcStatements
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE { OBJECT_IDENTIFIER { "0.4.0.1862.1.1" } }   -- QcCompliance
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.15";   -- Key Usage
               BOOLEAN =  #FF;   -- critical
               OCTET_STRING [ PRIMITIVE ] {
                  BIT_STRING { #06, #40 }   -- nonRepudiation
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.35";   -- Authority Key Identifier
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     CONTEXT_SPECIFIC [ 0, "IMPLICIT" ] {
#7231188759FE93E79E55F9C04AE18FF34A96580E }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.31";   -- CRL Distribution Points
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        CONTEXT_SPECIFIC [ 0 ] {
                           CONTEXT_SPECIFIC [ 0 ] {
                              CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://proxy.fineid.fi/crl/vrktpc.crl" }
                           }
                        }
                     },
                     SEQUENCE {
                        CONTEXT_SPECIFIC [ 0 ] {
                           CONTEXT_SPECIFIC [ 0 ] {
                              CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {

"ldap://ldap.fineid.fi:389/cn%3dVRK%20CA%20for%20Test%20Purposes,ou%3dTestivarmenteet,o%3dVaes
torekisterikeskus%20TEST,dmdName%3dFINEID,c%3dFI?certificateRevocationList"
                              }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.14";   -- Subject Key Identifier
               OCTET_STRING [ PRIMITIVE ] { OCTET_STRING {
#B9E412BA3FF517026EF91ADB67D3EF7357536F26 } }
            }
         }
      }
   },
   SEQUENCE {
      OBJECT_IDENTIFIER { "1.2.840.113549.1.1.5" },   -- SHA-1 with RSA Encryption
      NULL =  "NULL";
   },
   BIT_STRING {
      #00,
      #8B888CCA8D8D983C06896DA5B225FE1050577181E7B46204D744C4CB66
      46C320FACC4F3F0BDE3D68A7E134D2937D6AB1EF15CFE3CA34B6D726E977
      34A67AA9B0446A52BB83A4C7E627EABC6633D1EB727586C1EDFD5887438C
      612A916F70DE499B2D7571377F91B418D47334F95CC862B7763B1BCEAB7A
      8726D0FF047E67EDCEFF7066FC32BB3D5637071604F1C9EC2308EEE27662
      AB6A06CABD4E4E474E6D84930140D926C166029CC4BCF11210EC5ECC2FA3
      F0CF627C8F5E00F37F7EC2719D2120DB30D62938BE4F8B0B9857D0EC8D4C
      7940FF402667E361B1E9FFDA5E77C3F35130D2CFADB806DBE949E3F75D6E
      2C02F1055FADA2F393B806A39C1B6B2D59
  }
}
```

10.8. Service Certificate

```
SEQUENCE {
   SEQUENCE {
      CONTEXT_SPECIFIC [ 0 ] {
         INTEGER =  2;  -- x509v3 certificate
      },
      INTEGER =  200600425;  -- Certificate serial number
      SEQUENCE {
         OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" },   -- SHA-256 with RSA Encryption
         NULL =  "NULL";
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               PrintableString { "Vaestorekisterikeskus CA" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Palveluvarmenteet" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               PrintableString { "VRK CA for Service Providers - G3" }
            }
         }
      },
      SEQUENCE {
         UTCTime { "160915090718Z" },  -- not before
         UTCTime { "180915205959Z" }   -- not after
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               UTF8String { "Väestörekisterikeskus" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Varmennepalvelut" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.5";   -- id-at-serialNumber
               PrintableString { "0245437-2" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               PrintableString { "developer.fineid.fi" }
            }
         }
      },
      SEQUENCE {
         SEQUENCE {
            OBJECT_IDENTIFIER { "1.2.840.113549.1.1.1" },   -- RSA encryption
            NULL =  "NULL";
         },
         BIT_STRING [ PRIMITIVE ] {
            #00,
            SEQUENCE {
               INTEGER {
                  #00B0A48159E33DD728339F2E94CC8C4D2A4B4AD9F648A13F0377A1F352
                  702BBF0DC04F3CA7D132CEB8A7E436BEC0F9D7D579779E9A2B1156E4ED27
                  ED097C93C676D9AABE8327B9A28909AFAE731151AF5672181A91AFC53A85
                  2FD24926C7EE7591C953B245B7741550A29BA1CB09E4FFAB68917A2823B3
                  42CC60F541D3AA307FE3C222CD7482F78BBEBE8C321841A61B89E6CDB791
                  2C132D162EC4036FF69718B74C0905F00BC819B7CEF068D21887DB627F6F
                  D995F1DB859125855017BDF3C64218E80023D316BC9AAD749E0A4CDE6E7C
                  1CDE94EBDB9DAAE0D08CEBC943FF145428154C1970BF19AEC6CEE6EF3ED2
                  8DCF0CB8B1017F785A882F6C78BA28F26EFD877FA0C3437AD2406C5C46AB
                  0EEB3080397F9BB81F814213F1B8F52C601881FE469EF998F011FF41DCE6
                  92773371DE17973B7453093604B68BD083EBB2DAC19485EBE77FE3B85E83
                  66FD38108F1BC2DA091E84F68D1A8A1D516A5092303BB43123A5E70EA57E
```

```
                        E01016ADA7318487710915077F48A7D2B207D08584D142EB8F1611AD1006
                        C26D25846320766677C807E441EB2CAE28E2609120C42BE475900BD289D1
                        9D560A2598A42645DFED6E8E6EE40E5A030ABE9D960065618E411B9865B4
                        E14CD856D47C277CDE8986D2E0D3B9AEE2AF9132231480E7EF0F3CABA126
                        E25A284833A85F8694FCAEADE6A64614D7D0A1D20B40FC2CBA8F30B953DE
                        0C0EA429
                     },
                     INTEGER =  65537;
                  }
               }
            },
            CONTEXT_SPECIFIC [ 3 ] {
               SEQUENCE {
                  SEQUENCE {
                     OBJECT_IDENTIFIER =  "2.5.29.35";   -- Authority Key Identifier
                     OCTET_STRING [ PRIMITIVE ] {
                        SEQUENCE {
                           CONTEXT_SPECIFIC [ 0, "IMPLICIT" ] {
      #6504E82D92E7CB2AAB5715A8652AAAFAB71674F6 }
                        }
                     }
                  },
                  SEQUENCE {
                     OBJECT_IDENTIFIER =  "2.5.29.14";   -- Subject Key Identifier
                     OCTET_STRING [ PRIMITIVE ] { OCTET_STRING {
      #C583251C5C105A10C72911FB7DFE007E54A7105A } }
                  },
                  SEQUENCE {
                     OBJECT_IDENTIFIER =  "2.5.29.15";   -- Key Usage
                     BOOLEAN =  #FF;  -- critical
                     OCTET_STRING [ PRIMITIVE ] {
                  BIT_STRING { #04, #B0 }  -- digitalSignature, keyEncipherment,
dataEncipherment
                     }
                  },
                  SEQUENCE {
                     OBJECT_IDENTIFIER =  "2.5.29.32";   -- Certificate Policies
                     OCTET_STRING [ PRIMITIVE ] {
                        SEQUENCE {
                           SEQUENCE { OBJECT_IDENTIFIER { "0.4.0.2042.1.7" } }, -- ETSI CPS
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.2.246.517.1.10.34.1" }, -- VRK Server
      certificate CPS
                              SEQUENCE {
                                 SEQUENCE {
                                    OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.1" },  -- CPS
                                    IA5String { "http://www.fineid.fi/cps33/" }
                                 },
                                 SEQUENCE {
```

```
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.2" },  -- User notice
                              SEQUENCE {
                                 VisibleString {
                                    "Varmennepolitiikka on saatavilla - Certifikat
policy finns - Certificate policy is available http://www.fineid.fi/cps33"
                                 }
                              }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.17";   -- Subject Alternative Name
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     CONTEXT_SPECIFIC [ 2, "IMPLICIT" ] { "developer.fineid.fi" }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.19";   -- Basic Constraints
               BOOLEAN =  #FF;  -- critical
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE { BOOLEAN =  #00; }  -- CA Certificate=False
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.31";   -- CRL Distribution Points
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        CONTEXT_SPECIFIC [ 0 ] {
                           CONTEXT_SPECIFIC [ 0 ] {
                              CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://proxy.fineid.fi/crl/vrksp3c.crl" }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.37";   -- Extended Key Usage
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.3.2" }, -- Client authentication
                     OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.3.1" }  -- Server authentication
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.1.1" }, -- CA Information Access
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.48.2" },  -- CA Issuers
                        CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://proxy.fineid.fi/ca/vrksp3.crt" }
                     },
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.48.1" },  -- OCSP
                        CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://ocsp.fineid.fi/vrksp3" }
                     }
                  }
               }
            }
         }
      }
   },
   SEQUENCE {
      OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" },  -- sha256WithRSAEncryption
      NULL =  "NULL";
   },
   BIT_STRING {
      #00,
      #7B6F762D772B42EA5F71413124BB533DB362E79222C99AA7DF00D7BAC1
      9D443A470D31C593690063DCF100F60943F7768A6507FBE9189ED2F011F0
      446F0CA437925A5E52BBF86E221F0C302227EBA1472A66D417C7D3E8D74A
      BB1A734D62C105BECA4769C5570BA4B2CA26935CFDBAE76691326A22DC7D
      C21C612CD9D9638FD2BA70A1428C3E8742745623CD13DFF7ABD62DF33510
      4C30BBB24BB82D0AB729896AA7DAC2F5DFB42B31727538D49F7E6657B0FF
      5904B759BA169714E4341F0653CC0E317045473BDF1E32B5566B86C95B45
      0DEC22C317A160CCB52338EB194A4B465ADA02AFDE1696CC52E1EA60F9C3
      8F1B095EB2D7796E132D46D53437FBC1C3129DBD4B6D73E38653A9A7EB56
      DAD103198FBAF2426CE4A2522CC7A44E0D3930FD8D0FE098AD87322F12C7
      DBDEA39459BF1F9ECF4A0C91E4CFE5B011452C6973DDE6EE6A91ED47E584
      290C4F581740084BE0BEE181664404F5358718F33AC3FFCD7DBC3B39F1F5
      C8335E4F0AC0686AB25FC7714ECE13543EEC5436A9EBD9807534579CFBF0
      A9B211A0F3865F26139AE09FA64A8D3E3D0F01011A243789C1E86D08A45D
      F1C5719E8A3269A87B011BCE4ED448A3EB26C5162ECCB4EA95F98C78611C
      412BB7F74E5B24BC2853D8B6FF408FBBFFA2501BA108697E0526AA60FA70
      1E663729D5365FCB46F3AB2732562D24404DD3C0DBA09382A8D589C479C7
      C54127
```

```
    }
}
```

10.9. Certificate Revocation List

```
SEQUENCE {
   SEQUENCE {
      INTEGER =  1; -- version 2 CRL
      SEQUENCE {
         OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" },  -- SHA-256 with RSA Encryption
         NULL =  "NULL";
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               PrintableString { "Vaestorekisterikeskus TEST" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Testivarmenteet" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               PrintableString { "VRK CA for Test Purposes - G2" }
            }
         }
      },
      UTCTime { "151229075706Z" }, -- this Update
      UTCTime { "151229095600Z" }, -- next Update
      SEQUENCE {
         SEQUENCE {
            INTEGER =  100500001; -- certificate serial number
            UTCTime { "131108090947Z" } -- UTC time of certificate revocation
         },
         SEQUENCE {
            INTEGER =  100500212; -- certificate serial number
            UTCTime { "140116091109Z" } -- UTC time of certificate revocation
         },
         SEQUENCE {
            INTEGER =  100502514; -- certificate serial number
            UTCTime { "141110104727Z" } -- UTC time of certificate revocation
         },
         SEQUENCE {
            INTEGER =  100504313; -- certificate serial number
            UTCTime { "150706113006Z" }, -- UTC time of certificate revocation
            SEQUENCE {
               SEQUENCE {
                  OBJECT_IDENTIFIER =  "2.5.29.24";   -- Invalidity Date
                  OCTET_STRING [ PRIMITIVE ] {
                    GeneralizedType { "20150706112900Z" } -- time of known invalidity
                  }
               },
               SEQUENCE {
                  OBJECT_IDENTIFIER =  "2.5.29.21";   -- Reason code
                  OCTET_STRING [ PRIMITIVE ] {
                    ENUMERATED =  #06; -- certificate Hold
                  }
               }
            }
         },
         SEQUENCE {
            INTEGER =  100500208; -- certificate serial number
            UTCTime { "140116091200Z" } -- UTC time of certificate revocation
         },
         SEQUENCE {
            INTEGER =  100502912; -- certificate serial number
            UTCTime { "141205105700Z" }, -- UTC time of certificate revocation
            SEQUENCE {
               SEQUENCE {
                  OBJECT_IDENTIFIER =  "2.5.29.24";   -- Invalidity Date
                  OCTET_STRING [ PRIMITIVE ] {
                    GeneralizedType { "20141205105600Z" } -- time of known invalidity
                  }
               },
               SEQUENCE {
                  OBJECT_IDENTIFIER =  "2.5.29.21";   -- Reason code
                  OCTET_STRING [ PRIMITIVE ] {
                    ENUMERATED =  #06; -- certificate Hold
                  }
               }
            }
         },
.
.
.

[PART OF CRL REMOVED]
```

```
.
.
.
        SEQUENCE {
           INTEGER =  100500002; -- certificate serial number
           UTCTime { "131108093522Z" } -- UTC time of certificate revocation
        },
        SEQUENCE {
           INTEGER =  100500207; -- certificate serial number
           UTCTime { "150818082645Z" } -- UTC time of certificate revocation
        },
        SEQUENCE {
           INTEGER =  100504714; -- certificate serial number
           UTCTime { "150917134402Z" }, -- UTC time of certificate revocation
           SEQUENCE {
              SEQUENCE {
                 OBJECT_IDENTIFIER =  "2.5.29.24";   -- Invalidity Date
                 OCTET_STRING [ PRIMITIVE ] {
                   GeneralizedType { "20150917134300Z" } -- time of known invalidity
                 }
              },
              SEQUENCE {
                 OBJECT_IDENTIFIER =  "2.5.29.21";   -- Reason code
                 OCTET_STRING [ PRIMITIVE ] {
                   ENUMERATED =  #06; -- certificate Hold
                 }
              }
           }
        },
        SEQUENCE {
           INTEGER =  100504805; -- certificate serial number
           UTCTime { "151001113558Z" }, -- UTC time of certificate revocation
           SEQUENCE {
              SEQUENCE {
                 OBJECT_IDENTIFIER =  "2.5.29.24";   -- Invalidity Date
                 OCTET_STRING [ PRIMITIVE ] {
                   GeneralizedType { "20151001113500Z" } -- time of known invalidity
                 }
              },
              SEQUENCE {
                 OBJECT_IDENTIFIER =  "2.5.29.21";   -- Reason code
                 OCTET_STRING [ PRIMITIVE ] {
                   ENUMERATED =  #06; -- certificate Hold
                 }
              }
           }
        },
         SEQUENCE {
            INTEGER =  100501758; -- certificate serial number
            UTCTime { "151112120554Z" }, -- UTC time of certificate revocation
            SEQUENCE {
               SEQUENCE {
                  OBJECT_IDENTIFIER =  "2.5.29.24";   -- Invalidity Date
                  OCTET_STRING [ PRIMITIVE ] {
                    GeneralizedType { "20151112120500Z" } -- time of known invalidity
                  }
               },
               SEQUENCE {
                  OBJECT_IDENTIFIER =  "2.5.29.21";   -- Reason code
                  OCTET_STRING [ PRIMITIVE ] {
                    ENUMERATED =  #06; -- certificate Hold
                  }
               }
            }
         }
      },
      CONTEXT_SPECIFIC [ 0 ] {
         SEQUENCE {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.20";   -- CRL Number
               OCTET_STRING [ PRIMITIVE ] { INTEGER =  19434; }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.35";   -- Authority Key Identifier
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     CONTEXT_SPECIFIC [ 0, "IMPLICIT" ] {
#63FE681031EA955E27AB3E61495F1FA123F68972 }
                  }
               }
            }
         }
      }
   },
   SEQUENCE {
      OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" },  -- SHA-256 with RSA Encryption
      NULL =  "NULL";
   },
   BIT_STRING {
      #00,
      #896AB168DAAA6A7D2768B0143F87A5DDCA66341361A1B7A2725D22FAB4
      4487482E8053D306785DF37CDEFF542AA1DDF86DD888B2389C27D39149A2
      91CEFE3192148011C5CB72ACBFCCD0EA5C5C101930E5C4B29B395EE27BA2
      DD9E852BA7F8EDCFCCA48DA0CF483BDC19B407538381084E582884F3EA22
      9196A89496848AEE59DAD62CBE783C4472E7A8CC42CFD394F5B922EE2A91
```

```
          9F535C7C026F567E1644822544B92FDFB3DAF7C1E4F923C40E2CA4F7CF5E
          FF6BE610E12BBB9D15AA0EB1945233A3FF8EAB9190B6542F0D92650CC740
          70A77641A1AB5E19FF77059C65621BE1DC29F258AC5300E1C4FF341E5772
          514A369D941C50623E4A3B66763295EFAB0CE2DE5EBADA8FF06E7696B0DE
          C17AE6309E4AF9A397FE3D2C90B5EE71BDE91138E717109BFF1F5F4607EF
          FB6238639CE789444EAC99F978DB4BBC3A19D1896817FE7F9001B37C1B7B
          98B7F0836D2295F8274286B91DADFD4BDFB5547AD071CB595192CD0B0C07
          098E2C48BE402259528927CDA25B97D9422DE1F10467FC21610D4144DF33
          7BFB7F3DCFCF12E2B66F96EB5224629AD8453F33F580790DD69BB5CA91E9
          3AF036BEE470870DB7BFE722F0CAFEA2DA2E18021BC401BF5D4A15C9E4C9
          8ED7B143406CAA14F2A520A2F3F2E6172C2F1E29A635A7DB750DE2003A33
          BB6ED8B324E979996E1359DEE17D3F59EB503744B20C5B534822FED9A76B
          568A21
      }
  }
```

10.10. OCSP Responder Certificate

```
SEQUENCE {
   SEQUENCE {
      CONTEXT_SPECIFIC [ 0 ] {
         INTEGER =  2;  -- x509v3 certificate
      },
      INTEGER =  101001139;  -- Certificate serial number
      SEQUENCE {
         OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" },  -- SHA-256 with RSA Encryption
         NULL =  "NULL";
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               PrintableString { "Vaestorekisterikeskus TEST" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Testivarmenteet" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               PrintableString { "VRK CA for Test Purposes - G3" }
            }
         }
      },
      SEQUENCE {
         UTCTime { "161018101002Z" },  -- not before
         UTCTime { "211019235959Z" }   -- not after
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               PrintableString =  "VRK";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Varmennepalvelut" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.5";   -- id-at-serialNumber
               PrintableString { "0245437-2" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               PrintableString { "OCSP Responder" }
            }
         }
      },
      SEQUENCE {
         SEQUENCE {
            OBJECT_IDENTIFIER { "1.2.840.113549.1.1.1" },   -- RSA encryption
            NULL =  "NULL";
         },
         BIT_STRING [ PRIMITIVE ] {
            #00,
            SEQUENCE {
               INTEGER {
                  #00B1A6400DEFC17E4F55DE38E302973C11C0D6C0CA4B0B86DE36B59E10
                  0FA1B8D81375C92B094A3DDB734C9A6CA0E1B95BC86E2D1471DEBEBDDD9E
                  D3274B7AAD097AC1E272B7CE67CEEC2EF989734F89787E3599C5FA13B96F
                  8A17496AB5C026BB3E0EF45D46D2B2FA482D7DCB975D3AA66003025F8557
                  4793749A3665247A51EBE35637B1D3D821B1F22147C4E0E8C74D5706B358
                  CBB87AFCEAE6335BC14F19A4520BBD549365CAAF6032A1A39C50CD4C0F56
                  DC516A24780B86696537DC93115B85F3AF8DE8E82B7FB5E42AFB5091AFC2
                  0BA10D96B88481911EF6F9859E8C19FC49EAA45E185AE0C329FED1F46387
                  9B70566C609D208787377B42024556C95177
               },
               INTEGER =  65537;
            }
```

```
         }
      },
      CONTEXT_SPECIFIC [ 3 ] {
         SEQUENCE {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.35";   -- Authority Key Identifier
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     CONTEXT_SPECIFIC [ 0, "IMPLICIT" ] {
#5BCE869CC75343E602B9FB716C8C6DA320E5B1F8 }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.14";   -- Subject Key Identifier
               OCTET_STRING [ PRIMITIVE ] { OCTET_STRING {
#CE2C5B57BCA51AF7D54D44417F93F5852F0D6A5E } }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.15";   -- Key Usage
               BOOLEAN =  #FF;
               OCTET_STRING [ PRIMITIVE ] {
                  BIT_STRING { #05, #A0 } -- digitalSignature, keyEncipherment
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.32";   -- Certificate Policies
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.2.246.517.99.10.31.1" }, -- Test CPS: VRK
CA for test purposes G3
                        SEQUENCE {
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.1" },  -- CPS
                              IA5String { "http://www.fineid.fi/cps99/" }
                           },
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.2" },  -- User notice
                              SEQUENCE {
                                 VisibleString {
                                    "Varmennepolitiikka on saatavilla - Certifikat
policy finns - Certificate policy is available http://www.fineid.fi/cps99"
                                 }
                              }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.17";   -- Subject Alternative Name
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     CONTEXT_SPECIFIC [ 1, "IMPLICIT" ] { "vaestorekisterikeskus@vrk.fi"
}
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.19";   -- Basic Constraints
               BOOLEAN =  #FF;  -- critical
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE { BOOLEAN =  #00; }  -- CA Certificate=False
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.37";   -- Extended Key Usage
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE { OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.3.9" } } -- OCSPSigning
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.1.1" },  -- CA Information Access
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.48.2" },  -- CA Issuers
                        CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://proxy.fineid.fi/ca/vrktp3.crt" }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.48.1.5" },  -- OCSP No Check Extension
               OCTET_STRING { #0500 }
            }
         }
      }
   },
   SEQUENCE {
      OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" },  -- sha256WithRSAEncryption
      NULL =  "NULL";
   },
   BIT_STRING {
      #00,
      #6A12F9FA2FEA9542488DD411D0510C1FD8E9153BE31F4C227B99D7B261
      16973FDC68316E922DF13C3A9B18D491903A1EB20AB0A6FFE0BB3905E7E0
      E84C41ADD3B19E9BA19AA25AB9282C323175017F9189E191912A7C027814
      04FDFA4E3BE569F30364F2A936EBA24E87590D31EEAA3628494C31C64199
      7842626A0F72728A3932305BB0C36AE53798EF0531F1312B8A120241BA9B
      B64D15A64C6768F672A3C46E5EF246BF06D571D3BB8E64E9FE811E9171CE
      CD2770AA88D75535455EC993CCB3561FCC25266F603F638D55C12EC641FB
      EF766F935FD3569E933499706651D8B7E507314ED0818FA4CF676602C305
      0D1D2B689400346F2382BD4B6E7AEC24A1080119228D6CD0C178C66EFAD5
      A35260EF912CF773F3D9ADD2A5874260BA38CF484C7AC970FC61FE85DABB
      60E4C79ABD1802621A781398E04E0CC2EDA21CC9D5BB012D1C15C9FD95D0
      3AAC922FAEC354395278E3E706BD102840666ED00380D9EE67369BC8CF0C
      1C48404425F9F5F0280676D210974FFC2048BF6C8A4E2A8E3EB78227FE92
      8218AEF44CE9BDE53633936DB40602F492279B6CDD95CA3C091366F7088F
      BF59FA36A9A3CF15127A41B893F25A3D0A998E19BBFB93812789B75FEAD6
      74E86527CDACDD4F4062291CAE42F94FA591C9F2180A0FC5D555D1605376
      96673CE8FB78CEDB91A229134A2304A750DA79E1FF6504A58EA25DD94191
      006165
   }
}
```

10.11. Time Stamping Certificate

```
SEQUENCE {
   SEQUENCE {
      CONTEXT_SPECIFIC [ 0 ] {
         INTEGER =  2;  -- x509v3 certificate
       },
      INTEGER =  200508004;
      SEQUENCE {
         OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" }, -- SHA-256 with RSA Encryption
         NULL =  "NULL";
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               PrintableString { "Vaestorekisterikeskus CA" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Palveluvarmenteet" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               PrintableString { "VRK CA for Service Providers - G2" }
            }
         }
      },
      SEQUENCE {
         UTCTime { "150923130000Z" }, -- not before
         UTCTime { "160923125900Z" }  -- not after
      },
      SEQUENCE {
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.6";   -- id-at-countryName
               PrintableString =  "FI";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.10";   -- id-at-organizationName
               PrintableString =  "VRK";
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.11";   -- id-at-organizationalUnitName
               PrintableString { "Varmennepalvelut" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.5";   -- id-at-serialNumber
               PrintableString { "0245437-2" }
            }
         },
         SET {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.4.3";   -- id-at-commonName
               PrintableString { "VRK TSA1" }
            }
         }
      },
      SEQUENCE {
         SEQUENCE {
            OBJECT_IDENTIFIER { "1.2.840.113549.1.1.1" },   -- RSA encryption
            NULL =  "NULL";
         },
         BIT_STRING [ PRIMITIVE ] {
            #00,
            SEQUENCE {
               INTEGER {
                  #00CDCB396843C43F3E236A522BD1BECD85D089EB5313D0ED605E8399CF
                  5C44F08B61DCF308BE212DC2E1E3FAC3D76FB8D4B9CFADF2A174425CF11B
                  140AD53E0FC1837E3C312485BA81D9A726DEA6FFD6ADD7147C3A877E5379
                  CA387D6C9579698FF111C0710CED14F4E96FC6E810E5A2BFE947B5C9A626
                  930F2C3B71D451B3827BE60FAEB8920722C4304FB78934BBC3DAD015DDAB
                  9A309F7279FB0C48C2B3C44F8C688C43082918BB0AA96D0EF49177FC3D50
                  9938A1029404D8D854E6420766ED66F998D43746EB4EDFB60D99E65F0F8E
                  9E7717E706F873DEE78BC7BDE4217D950762B942277F84D32166187DA517
                  DB71D63D7453A2AA3513BDCAFB2870DEED1D
               },
               INTEGER =  65537; -- exponent
            }
         }
```

```
      },
      CONTEXT_SPECIFIC [ 3 ] {
         SEQUENCE {
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.19";   -- Basic Constraints
               BOOLEAN =  #FF; -- critical
               OCTET_STRING [ PRIMITIVE ] {
                   SEQUENCE { BOOLEAN =  #00; }  -- CA Certificate=False
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.32";   -- Certificate Policies
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE { OBJECT_IDENTIFIER { "0.4.0.2042.1.3" } }, -- ETSI CPS
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.2.246.517.1.10.24.1" }, -- VRK CPS Object 
Identifier
                        SEQUENCE {
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.2" }, -- User notice
                              SEQUENCE {
                                 VisibleString {
                                    "Varmennepolitiikka on saatavilla - Certifikat 
policy finns - Certificate policy is available http://www.fineid.fi/cps23"
                                 }
                              }
                           },
                           SEQUENCE {
                              OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.2.1" }, -- CPS
                              IA5String { "http://www.fineid.fi/cps23/" }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.1.1" }, -- Authority Info Access
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.48.2" }, -- CA Issuers
                        CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] { 
"http://proxy.fineid.fi/ca/vrksp2.crt" }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.17";   -- Subject Alternative Name
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     CONTEXT_SPECIFIC [ 1, "IMPLICIT" ] { "vaestorekisterikeskus@vrk.fi"
}
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.15";   -- Key Usage
               BOOLEAN =  #FF; -- critical
               OCTET_STRING [ PRIMITIVE ] {
                  BIT_STRING { #06, #C0 } -- digitalSignature, nonRepudiation
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.37";   -- Extended Key Usage
               BOOLEAN =  #FF; -- critical
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE { OBJECT_IDENTIFIER { "1.3.6.1.5.5.7.3.8" } } -- timeStamping
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.35";   -- Authority Key Identifier
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     CONTEXT_SPECIFIC [ 0, "IMPLICIT" ] {
#69997A20AE437E2597092CFB297B2C8C67D231CD }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.31";   -- CRL Distribution Points
               OCTET_STRING [ PRIMITIVE ] {
                  SEQUENCE {
                     SEQUENCE {
                        CONTEXT_SPECIFIC [ 0 ] {
                           CONTEXT_SPECIFIC [ 0 ] {
                              CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
"http://proxy.fineid.fi/crl/vrksp2c.crl" }
                           }
                        }
                     },
                     SEQUENCE {
                        CONTEXT_SPECIFIC [ 0 ] {
                           CONTEXT_SPECIFIC [ 0 ] {
                              CONTEXT_SPECIFIC [ 6, "IMPLICIT" ] {
```

```
"ldap://ldap.fineid.fi:389/cn%3dVRK%20CA%20for%20Service%20Providers%20-
%20G2,ou%3dPalveluvarmenteet,o%3dVaestorekisterikeskus%20CA,dmdName%3dFINEID,c%3dFI?cert
ificateRevocationList"
                              }
                           }
                        }
                     }
                  }
               }
            },
            SEQUENCE {
               OBJECT_IDENTIFIER =  "2.5.29.14";   -- Subject Key Identifier
               OCTET_STRING [ PRIMITIVE ] { OCTET_STRING {
#2BC429A09C1C55D4BD73BC43DA95DDFB891F441D } }
            }
         }
      }
   },
   SEQUENCE {
      OBJECT_IDENTIFIER { "1.2.840.113549.1.1.11" }, -- SHA-256 with RSA Encryption
      NULL =  "NULL";
   },
   BIT_STRING {
      #00,
      #6636E8A11B8D96B67F675BE9FF78318E102581876FB6247C0F63B6BA88
      A0AFB9F5C07FA66F2C6935AEF58B52A78DBE1F8059B99B2DBECD8DC9BF9E
      AEDD4254D38FB07FABAAC0D54EF7061448C7DB2C39E8EF58E9DEBE3049B1
      E5F6D6DD14B129A05FDCD55FE3EEBC8F4FD34178A78EE6B7E42ECFF6D808
      31D9E634256CA52665E3D71DF06BC22714DC637F546BDF0243F6932C682B
      7EA871E414418949AE208771FB1B458D5192BF1D3F75E717DD7EB026123D
      5C4680B88081884F6044729A365D03704B2D0BF099867B2C8EB10E7328D3
      380EFB4352B4BE4F34C24144C39B9222626E1CFD2D74C80DC3C3C0129A18
      D5103A896F035D0F2AED2F6607D50F04647256F9816FC953F085E0CF44FD
      4A5414DF9CC72FC45ECBD69A923BCCCF2387EAC366D3CB058053D7628C98
      DD56E272F9CE095E8879615B6FB113FEA1B29C3E4DFE73A9D9CAE47F1B35
      4E24203B47B9990D576B4FFA3B13BAB9BA398A30C7C1480242B0FD264FAA
      E7CC0184027B1EF8D914604C6E9E7CA084A0F365B9F17BD2098314DA7E8A
      95ECBCE1233628B6FDE9E216EF8E0748BA23B9193AE05AE74C6259F0D916
      AFBAFCDB929A535780EAD532C1E939D0BFB7377FC0ED1972170C81A264C0
      9B2A571BD99D8CE4A6960E9F1BD771007BCA970B962BED6914CE9D0B2B4C
      1986642BE4AC61C1FA2577D4E074D1AEBE2C90F0EF984C615AF797FC7D6B
      9FF0A1
   }
}
```