PKIX Working Group
Internet Draft
Expires August 5, 2005
Intended Category: Informational

Serguei Leontiev, CRYPTO-PRO
Dennis Shefanovskij, DEMOS Co Ltd
February 5, 2005

Using the GOST R 34.10-94, GOST R 34.10-2001 and GOST R 34.11-94 algorithms with the Internet X.509 Public Key Infrastructure Certificate and CRL Profile.

<draft-ietf-pkix-gost-cppk-02.txt>

Status of this Memo

Abstract

This document describes identifiers and appropriate parameters for the algorithms GOST R 34.10-94, GOST R 34.10-2001, GOST R 34.11-94, and also ASN.1 encoding scheme for digital signatures and public keys, used in Internet X.509 Public Key Infrastructure (PKI). This specification extends [RFC3279], "Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile" and, correspondingly, [RFC3280], "Internet X.509 Public Key Infrastructure: Certificate and Certificate Revocation List (CRL) Profile". All implementations of

this specification MUST also satisfy the requirements of [RFC3280].

Table of Contents

1 Introduction

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

This document defines identifiers and corresponding algorithm parameters and attributes proposed by CRYPTO-PRO Company within "Russian Cryptographic Software Compatibility Agreement" community for the algorithms GOST R 34.10-94, GOST R 34.10-2001, GOST R 34.11-94, key derivation algorithms based on GOST R 34.10-94 public keys, key derivation algorithms based on GOST R 34.10-2001 public keys, and also ASN.1 encoding [X.660] for digital signatures and public keys, used in Internet X.509 Public Key Infrastructure (PKI).

This specification extends [RFC3279], "Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile" and, correspondingly, [RFC3280], "Internet X.509 Public Key Infrastructure: Certificate and Certificate Revocation List (CRL) Profile". All implementations of

this specification MUST also satisfy the requirements of [RFC3280].

This specification defines the content of the signatureAlgorithm, signatureValue, signature, and subjectPublicKeyInfo fields within Internet X.509 certificates and CRLs.

This document defines the use of one-way hash-function GOST R 34.11-94 [GOST3411] with digital signatures. This algorithm is used in conjunction with digital signature algorithms.

This specification describes the encoding of digital signatures, generated with the following cryptographic algorithms:

* GOST R 34.10-94;
* GOST R 34.10-2001.

This document also defines the contents of the subjectPublicKeyInfo field for Internet X.509 certificates. For each algorithm, the appropriate alternatives for the keyUsage extension are provided. This specification describes encoding formats for public keys used with the following cryptographic algorithms:

* GOST R 34.10-94 [GOST341094];
* GOST R 34.10-2001 [GOST34102001];
* Key derivation algorithm VKO GOST R 34.10-94 [CPALGS];
* Key derivation algorithm VKO GOST R 34.10-2001 [CPALGS];

ASN.1 modules, including all the definitions used in this document can be found in [CPALGS].

2 Algorithm Support

This section is an overview of cryptographic algorithms, that may be used within the Internet X.509 certificates and CRL profile [RFC3280]. It describes one-way hash functions and digital signature algorithms, that may be used to sign certificates and CRLs, and identifies OIDs and ASN.1 encoding for public keys contained in a certificate.

The conforming CAs and/or applications MUST fully support digital signatures and public keys for at least one of the specified algorithms.

2.1 One-way Hash Function

This section identifies the use of one-way, collision free hash function GOST R 34.11-94 - the only one that can be used in digital signature algorithms GOST R 34.10-94/2001. The data that is hashed for certificates and CRL signing is fully described in [RFC3280].

2.1.1 One-way Hash Function GOST R 34.11-94

GOST R 34.11-94 has been developed by "GUBS of Federal Agency Government Communication and Information" and "All-Russian Scientific

and Research Institute of Standardization". The algorithm GOST R 34.11-94 produces a 256-bit hash value of the arbitrary finite bit length input. This document does not contain GOST R 34.11-94 full specification, which can be found in [GOSTR3411] in Russian. It's brief technical description in english can be found in [Schneier95], ch. 18.11, p. 454.

This function is always used with default parameter set gostR3411CryptoProParamSetAI (see section 8.2 of [CPALGS]).

2.2 Signature Algorithms

Conforming CAs may use GOST R 34.10-94 or GOST R 34.10-2001 signature algorithms to sign certificates and CRLs. The signatureAlgorithm field of Certificate or CertificateList indicates the signature algorithm ID, and associated parameters. This section also defines algorithm identifiers and parameters that MUST be used in the signatureAlgorithm field in a Certificate or CertificateList.

Signature algorithms are always used conjointly with a one-way hash function GOST R 34.11-94 as indicated in [GOSTR341094] and [GOSTR34102001].

This section identifies OIDs for GOST R 34.10-94 and GOST R 34.10-2001 algorithms. The contents of the parameters component for each algorithm may vary and details are provided below for each algorithm separately.

2.2.1 Signature Algorithm GOST R 34.10-94

GOST R 34.10-94 has been developed by "GUBS of Federal Agency Government Communication and Information" and "All-Russian Scientific and Research Institute of Standardization". This signature algorithm MUST be used conjointly with one-way, collision free hash function GOST R 34.11-94. This document does not contain GOST R 34.10-94 standard description, which is fully described in [GOSTR341094] in Russian, and brief description in English could be found in [Schneier95] ch. 20.3, p. 495.

The ASN.1 OID used to identify GOST R 34.10-94 signature algorithm in fields signatureAlgorithm in Certificate and CertificateList is:

```
id-CryptoPro-algorithms OBJECT IDENTIFIER ::=
      { iso(1) member-body(2) ru(643) rans(2) cryptopro(2) }

id-GostR3411-94-with-GostR3410-94 OBJECT IDENTIFIER ::=
      { id-CryptoPro-algorithms gostR3411-94-with-gostR3410-94(4)}
```



```
GostR3410-94-CertificateSignatureAlgorithms
   ALGORITHM-IDENTIFIER ::= {
      { NULL IDENTIFIED BY
                  id-GostR3411-94-with-GostR3410-94 } |
      { GostR3410-94-PublicKeyParameters IDENTIFIED BY
                  id-GostR3411-94-with-GostR3410-94 } }
```

GostR3410-94-PublicKeyParameters are defined in section 2.3.1.

When the id-GostR3411-94-with-GostR3410-94 algorithm identifier appears in an AlgorithmIdentifier and parameters are omitted, the parameters from the public key of the signer's certificate MUST be used. If the parameters from the public key of the signer's certificate are also omited, and it's issuer's certificate has the same public key algorithm, parameters from the public key of the issuer's certificate MUST be used, and so on.

Signature algorithm GOST R 34.10-94 generates digital signature in the form of a binary 512-bit vector (<r'>256||<s>256). That is, the least-significant (1-st) bit of signatureValue BIT STRING contains the least-significant (1-st) bit of <s>, and the most-significant (512th) bit of signatureValue contains the most-significant (256th)

bit of <r'>.

2.2.2 Signature Algorithm GOST R 34.10-2001

GOST R 34.10-2001 was developed by "GUBS of Federal Agency Government Communication and Information" and "All-Russian Scientific and Research Institute of Standardization". This signature algorithm MUST be used conjointly with one-way, collision free hash function GOST R 34.11-94. This document does not contain GOST R 34.10-2001 standard description, which is fully described in [GOSTR34102001].

The ASN.1 OID used to identify GOST R 34.10-2001 signature algorithm in fields signatureAlgorithm of Certificate and CertificateList is:

```
id-GostR3411-94-with-GostR3410-2001 OBJECT IDENTIFIER ::=
      { id-CryptoPro-algorithms gostR3411-94-with-gostR3410-2001(3) }

GostR3410-2001-CertificateSignatureAlgorithms
  ALGORITHM-IDENTIFIER ::= {
    { NULL IDENTIFIED BY
                    id-GostR3411-94-with-GostR3410-2001 } |
    { GostR3410-2001-PublicKeyParameters IDENTIFIED BY
                    id-GostR3411-94-with-GostR3410-2001 } }
```

GostR3410-2001-PublicKeyParameters are defined in section 2.3.2.

When the id-GostR3411-94-with-GostR3410-2001 algorithm identifier appears in an AlgorithmIdentifier and parameters are omitted, the parameters from the public key of the signer's certificate MUST be used. If the parameters from the public key of the signer's certificate are also omited, and it's issuer's certificate has the same public key algorithm, parameters from the public key of the issuer's certificate MUST be used, and so on.

Signature algorithm GOST R 34.10-2001 generates digital signature in the form of a binary 512-bit vector (<r'>256||<s>256). That is, the least-significant (1-st) bit of signatureValue BIT STRING contains the least-significant (1-st) bit of <s>, and the most-significant (512th) bit of signatureValue contains the most-significant (256th) bit of <r'>.

2.3 Subject Public Key Algorithms

In according to [RFC3280] the certificates may contain a public key for any algorithm. Within the framework of this specification the only GOST R 34.10-94 and GOST R 34.10-2001 public key algorithms defined. The algorithm and associated parameters are definable as OID in certificate through ASN.1 structure AlgorithmIdentifier.

This section identifies defines OID and public key parameters for the GOST R 34.10-94 and GOST R 34.10-2001 algorithms. The appropriate CA MUST use the predefined OID issuing certificates containing public keys for these algorithms. The appropriate applications supporting any of these algorithms MUST fully recognize the OID identified in this section

2.3.1 GOST R 34.10-94 Keys

This section defines OID and parameter encoding for inclusion of GOST R 34.10-94 public key in certificate. Such public key can be used for digital signature validation algorithm GOST R 34.10-94 [GOSTR341094], and for key derivation algorithm VKO GOST R 34.10-94 [CPALGS].

An assumed cryptographic key usage MAY be specified by keyUsage extension [RFC3280]. The usage of the same key for signature and key derivation is NOT RECOMMENDED, but possible.

Public key OID for GOST R 34.10-94 declared in this document is:

```
id-GostR3410-94 OBJECT IDENTIFIER ::=
    { id-CryptoPro-algorithms gostR3410-94(20) }
```

SubjectPublicKeyInfo.algorithm.algorithm field (see [RFC3280]) for

GOST R 34.10-94 keys MUST be id-GostR3410-94;

SubjectPublicKeyInfo.algorithm.parameters in this case MUST have the following structure:

```
 GostR3410-94-PublicKeyParameters ::=
     SEQUENCE {
         publicKeyParamSet
```

```
            OBJECT IDENTIFIER,
        digestParamSet
            OBJECT IDENTIFIER,
        encryptionParamSet
            OBJECT IDENTIFIER OPTIONAL
    }
```

where:
 * publicKeyParamSet - public key parameters identifier for GOST R 34.10-94 (see section 8.3 of [CPALGS])
 * digestParamSet - parameters identifier for GOST R 34.11-94 (see section 8.2 of [CPALGS])
 * encryptionParamSet - optional parameters identifier for GOST 28147-89 (see section 8.1 of [CPALGS]) MAY be present in any certificate and MUST be present if keyUsage includes keyAgreement or keyEnchiperment.

If GOST R 34.10-94 algorithm parameters are omitted in subjectPublicKeyInfo, and CA signs subject certificate using GOST R 34.10-94, then GOST R 34.10-94 parameters taken from subjectPublicKeyInfo field of issuer certificate are applicable to public key of GOST R 34.10-94 subject. That is, cryptographic parameters inheritance takes place. If subjectPublicKeyInfo AlgorithmIdentifier field contain no parameters, but CA sign certificate using signature algorithm different from GOST R 34.10-94, such certificate MUST be rejected by conforming applications.

Public key GOST R 34.10-94 MUST be ASN.1 encoded in following way.

In GOST R 34.10-94 public key is a number y = a^x (mod p), where a and p - parameters, and y is a bit-vector (<y>1024), at that encoding should present <y>1024 (BIT STRING) as a vector holding data in a little-endian. At first, a key is presented as an OCTET STRING, and then, being DER-encoded, presented as a BIT STRING.

GostR3410-94-PublicKey ::= BIT STRING

GostR3410-94-PublicKeyOctetString ::= OCTET STRING

If the keyUsage extension is present in an end-entity certificate,

which contains a GOST R 34.10-94 public key, the following values MAY

be present:

   digitalSignature;
   nonRepudiation.
   keyEncipherment;
   keyAgreement.

If the keyAgreement or keyEnchiperment extension is present in a certificate GOST R 34.10-94 public key, the following values MAY be present as well:

   encipherOnly;
   decipherOnly.

The keyUsage extension MUST NOT assert both encipherOnly and decipherOnly.

If the keyUsage extension is present in an CA or CRL signer certificate which contain a GOST R 34.10-94 public key, the following values MAY be present:

   digitalSignature;
   nonRepudiation;
   keyCertSign;
   cRLSign.

2.3.2 GOST R 34.10-2001 Keys

This section defines OID and parameter encoding for inclusion of GOST R 34.10-2001 public key in certificate. Such public key can be used for digital signature validation algorithm GOST R 34.10-2001 [GOSTR34102001], and for key derivation algorithm VKO GOST R 34.10-2001 [CPALGS].

An assumed cryptographic key usage MAY be specified by keyUsage extension [RFC3280]. The usage of the same key for signature and key derivation is NOT RECOMMENDED, but possible.

Public key OID for GOST R 34.10-2001 declared in this document is:

```
id-GostR3410-2001 OBJECT IDENTIFIER ::=
    { id-CryptoPro-algorithms gostR3410-2001(19) }
```

SubjectPublicKeyInfo.algorithm.algorithm field (see [RFC3280]) for GOST R 34.10-2001 keys MUST be id-GostR3410-2001;

SubjectPublicKeyInfo.algorithm.parameters in this case MUST have the

following structure:

```
 GostR3410-2001-PublicKeyParameters ::=
     SEQUENCE {
         publicKeyParamSet
             OBJECT IDENTIFIER,
         digestParamSet
             OBJECT IDENTIFIER,
         encryptionParamSet
             OBJECT IDENTIFIER OPTIONAL
     }
```

where:
 * publicKeyParamSet - public key parameters identifier for GOST R 34.10-2001 (see section 8.4 of [CPALGS])
 * digestParamSet - parameters identifier for GOST R 34.11-94 (see section 8.2 of [CPALGS])
 * encryptionParamSet - optional parameters identifier for GOST 28147-89 (see section 8.1 of [CPALGS]) MAY be present in any certificate and MUST be present if keyUsage includes keyAgreement or keyEnchiperment.

If GOST R 34.10-2001 algorithm parameters are omitted in subjectPublicKeyInfo, and CA signs subject certificate using GOST R 34.10-2001, then GOST R 34.10-2001 parameters taken from subjectPublicKeyInfo field of issuer certificate are applicable to public key of GOST R 34.10-2001 subject. That is, cryptographic parameters inheritance takes place. If subjectPublicKeyInfo AlgorithmIdentifier field contain no parameters, but CA sign certificate using signature algorithm different from GOST R 34.10-2001, such certificate MUST be rejected by conforming applications.

GOST R 34.10-2001 public key MUST be ASN.1 encoded in a following way. GOST R 34.10-2001 specifies that public key is a point on the elliptic curve Q = dP, where d is a private key, P is a base point, and Q presents in a way of 512-bit vector (<Xq>256||<Yq>256). This vector is DER-encoded as two data blocks. At first, <Xq>256 block, then <Yq>256 block. subjectPublicKey field BIT STRING type is presented as a taken up object GostR3410-2001-PublicKeyOctetString.

At that, least-significant of the first octet (GostR3410-2001-PublicKeyOctetString[0]) corresponds to least-significant (1-st) of vector <Xq>256||<Yq>256 (Yq1 = (GostR3410-2001-PublicKeyOctetString[0] & 1)).

Whereas most-significant of 64-th octet (GostR3410-2001-PublicKeyOctetString[63]) corresponds to most-

significant (512-d) of vector <Xq>256||<Yq>256 (Xq256 = ((GostR3410-2001-PublicKeyOctetString[63] & 0x80)>>7)).

In other words, <Xq>256||<Yq>256 vector is stored in little-endian, that correspond binary vector form and their concatenation in GOST R 34.10-2001 ch. 5.3. At first, key is placed in OCTET STRING, than is DER-encoded and placed in BIT STRING.

GostR3410-2001-PublicKey ::= BIT STRING

GostR3410-2001-PublicKeyOctetString ::= OCTET STRING

If the keyUsage extension is present in an end-entity certificate, which conveys a GOST R 34.10-2001 public key, the following values MAY be present:

digitalSignature,
nonRepudiation,
keyEncipherment,
keyAgreement.

If the keyAgreement or keyEnchiperment extension is present in a certificate, the following values MAY be present:

encipherOnly,
decipherOnly.

The keyUsage extension MUST NOT assert both encipherOnly and decipherOnly.

If the keyUsage extension is present in an CA or CRL signer certificate which contain a GOST R 34.10-2001 public key, the following values MAY be present:

digitalSignature,
nonRepudiation,
keyCertSign,
cRLSign.

3 Security Considerations

It is RECCOMENDED, that applications verify signature values and subject public keys to conform to [GOSTR34102001], [GOSTR341094] standards prior to their use.

When certificate is used as analogue to a manual signing, in the context of Russian Federal Digital Signature Law [RFDSL], certificate MUST contain keyUsage extension, it MUST be critical, and keyUsage

MUST NOT include keyEncipherment and keyAgreement.

When certificate validity period (typicaly 5 years for end entities and 7 years for CAs in Russia) is not equal to the private key validity period (typicaly 15 months in Russia) it is RECOMENDED to use private key usage period extension.

For security discussion concerning use of algorithm parameters, see section Security Considerations from [CPALGS].

4 Appendix Examples

4.1 GOST R 34.10-94 Certificate

```
   0 30  527: SEQUENCE {
   4 30  444:   SEQUENCE {
   8 02   16:     INTEGER
          :       17 31 2A C2 1B D2 08 58 BC 04 1E 52 37 D0 74 50
  26 30   10:     SEQUENCE {
  28 06    6:       OBJECT IDENTIFIER
          :         id_GostR3411_94_with_GostR3410_94
          :           ( 1 2 643 2 2 4)
  36 05    0:       NULL
          :       }
  38 30  105:     SEQUENCE {
  40 31   29:       SET {
  42 30   27:         SEQUENCE {
  44 06    3:           OBJECT IDENTIFIER
```

```
           :             commonName (2 5 4 3)
 49 0C   20:           UTF8String 'GostR3410-94 example'
           :           }
           :         }
 71 31   18:       SET {
 73 30   16:         SEQUENCE {
 75 06    3:           OBJECT IDENTIFIER
           :             organizationName (2 5 4 10)
 80 0C    9:           UTF8String 'CryptoPro'
           :           }
           :         }
 91 31   11:       SET {
 93 30    9:         SEQUENCE {
 95 06    3:           OBJECT IDENTIFIER
           :             countryName (2 5 4 6)
100 13    2:           PrintableString 'RU'
```



```
           :           }
           :         }
104 31   39:       SET {
106 30   37:         SEQUENCE {
108 06    9:           OBJECT IDENTIFIER
           :             emailAddress (1 2 840 113549 1 9 1)
119 16   24:           IA5String 'GostR3410-94@example.com'
           :           }
           :         }
           :       }
145 30   30:     SEQUENCE {
147 17   13:       UTCTime '050203151651Z'
162 17   13:       UTCTime '150203151651Z'
           :       }
177 30  105:     SEQUENCE {
179 31   29:       SET {
181 30   27:         SEQUENCE {
183 06    3:           OBJECT IDENTIFIER
           :             commonName (2 5 4 3)
188 0C   20:           UTF8String 'GostR3410-94 example'
           :           }
           :         }
210 31   18:       SET {
212 30   16:         SEQUENCE {
214 06    3:           OBJECT IDENTIFIER
```

```
           :             organizationName (2 5 4 10)
 219 0C    9:           UTF8String 'CryptoPro'
           :           }
           :         }
 230 31   11:       SET {
 232 30    9:         SEQUENCE {
 234 06    3:           OBJECT IDENTIFIER
           :             countryName (2 5 4 6)
 239 13    2:           PrintableString 'RU'
           :           }
           :         }
 243 31   39:       SET {
 245 30   37:         SEQUENCE {
 247 06    9:           OBJECT IDENTIFIER
           :             emailAddress (1 2 840 113549 1 9 1)
 258 16   24:           IA5String 'GostR3410-94@example.com'
           :           }
           :         }
           :       }
 284 30  165:     SEQUENCE {
 287 30   28:       SEQUENCE {
 289 06    6:         OBJECT IDENTIFIER
           :           id_GostR3410_94 ( 1 2 643 2 2 20)
```



```
 297 30   18:         SEQUENCE {
 299 06    7:           OBJECT IDENTIFIER
           :             id_GostR3410_94_CryptoPro_A_ParamSet
           :               ( 1 2 643 2 2 32 2)
 308 06    7:           OBJECT IDENTIFIER
           :             id_GostR3411_94_CryptoProParamSet
           :               ( 1 2 643 2 2 30 1)
           :           }
           :         }
 317 03  132:       BIT STRING 0 unused bits, encapsulates {
 321 04  128:         OCTET STRING
           :           BB 84 66 E1 79 9E 5B 34 D8 2C 80 7F 13 A8 19 66
           :           71 57 FE 8C 54 25 21 47 6F 30 0B 27 77 46 98 C6
           :           FB 47 55 BE B7 B2 F3 93 6C 39 B5 42 37 26 84 E2
           :           0D 10 8A 24 0E 1F 0C 42 4D 2B 3B 11 2B A8 BF 66
           :           39 32 5C 94 8B C1 A8 FE 1B 63 12 F6 09 25 87 CC
           :           75 1B F4 E5 89 8A 09 82 68 D3 5C 77 A6 0F B6 90
           :           10 13 8D E3 3E 7C 9C 91 D6 AC 0D 08 2C 3E 78 C1
```

```
           :           B5 C2 B6 B7 1A A8 2A 8B 45 81 93 32 32 76 FA 7B
           :         }
           :       }
           :     }
 452 30   10:   SEQUENCE {
 454 06    6:     OBJECT IDENTIFIER
           :       id_GostR3411_94_with_GostR3410_94 ( 1 2 643 2 2 4)
 462 05    0:     NULL
           :     }
 464 03   65:   BIT STRING 0 unused bits
           :     71 28 D8 4E 9A 38 33 FE 2E 42 12 02 CE C8 AC B3
           :     F6 91 46 90 37 1A CA 6B 16 61 05 95 BF B0 99 D2
           :     94 CC F0 8C CC CE 45 01 5B 71 87 B1 48 C2 16 96
           :     A7 15 90 DF 83 6C EE 37 ED E4 4F EE BD E2 7F 41
           :   }
```

4.2 GOST R 34.10-2001 Certificate

```
   0 30  468: SEQUENCE {
   4 30  385:   SEQUENCE {
   8 02   16:     INTEGER
           :       48 E9 54 A5 CF E9 69 F5 C9 5C F7 55 E7 83 41 AF
  26 30   10:     SEQUENCE {
  28 06    6:       OBJECT IDENTIFIER
           :         id_GostR3411_94_with_GostR3410_2001
           :           ( 1 2 643 2 2 3)
  36 05    0:       NULL
           :       }
  38 30  109:     SEQUENCE {
  40 31   31:       SET {
```



```
  42 30   29:         SEQUENCE {
  44 06    3:           OBJECT IDENTIFIER
           :             commonName (2 5 4 3)
  49 0C   22:           UTF8String 'GostR3410-2001 example'
           :           }
           :         }
  73 31   18:       SET {
  75 30   16:         SEQUENCE {
  77 06    3:           OBJECT IDENTIFIER
           :             organizationName (2 5 4 10)
  82 0C    9:           UTF8String 'CryptoPro'
```

```
            :           }
            :         }
 93 31   11:       SET {
 95 30    9:         SEQUENCE {
 97 06    3:           OBJECT IDENTIFIER
            :             countryName (2 5 4 6)
102 13    2:           PrintableString 'RU'
            :           }
            :         }
106 31   41:       SET {
108 30   39:         SEQUENCE {
110 06    9:           OBJECT IDENTIFIER
            :             emailAddress (1 2 840 113549 1 9 1)
121 16   26:           IA5String 'GostR3410-2001@example.com'
            :           }
            :         }
            :       }
149 30   30:     SEQUENCE {
151 17   13:       UTCTime '050203151646Z'
166 17   13:       UTCTime '150203151646Z'
            :       }
181 30  109:     SEQUENCE {
183 31   31:       SET {
185 30   29:         SEQUENCE {
187 06    3:           OBJECT IDENTIFIER
            :             commonName (2 5 4 3)
192 0C   22:           UTF8String 'GostR3410-2001 example'
            :           }
            :         }
216 31   18:       SET {
218 30   16:         SEQUENCE {
220 06    3:           OBJECT IDENTIFIER
            :             organizationName (2 5 4 10)
225 0C    9:           UTF8String 'CryptoPro'
            :           }
            :         }
236 31   11:       SET {
```



```
238 30    9:         SEQUENCE {
240 06    3:           OBJECT IDENTIFIER
            :             countryName (2 5 4 6)
245 13    2:           PrintableString 'RU'
```

```
          :           }
          :         }
249 31   41:       SET {
251 30   39:         SEQUENCE {
253 06    9:           OBJECT IDENTIFIER
          :             emailAddress (1 2 840 113549 1 9 1)
264 16   26:           IA5String 'GostR3410-2001@example.com'
          :           }
          :         }
          :       }
292 30   99:     SEQUENCE {
294 30   28:       SEQUENCE {
296 06    6:         OBJECT IDENTIFIER
          :           id_GostR3410_2001 ( 1 2 643 2 2 19)
304 30   18:         SEQUENCE {
306 06    7:           OBJECT IDENTIFIER
          :             id_GostR3410_2001_CryptoPro_XchA_ParamSet
          :               ( 1 2 643 2 2 36 0)
315 06    7:           OBJECT IDENTIFIER
          :             id_GostR3411_94_CryptoProParamSet
          :               ( 1 2 643 2 2 30 1)
          :           }
          :         }
324 03   67:       BIT STRING 0 unused bits, encapsulates {
327 04   64:         OCTET STRING
          :           84 95 68 75 60 02 1A 40 75 08 CD 13 8C 31 89 2C
          :           FD E5 05 03 7A 43 5C F4 6D 2B 0F E7 4F 32 7E 57
          :           8F EB CC 16 B9 95 88 03 D0 9A 7C 85 AE 0F E4 8D
          :           EA A6 BB 7E 56 C7 CB B0 DF 0F 66 BC CA EA 1A 60
          :         }
          :       }
          :     }
393 30   10:   SEQUENCE {
395 06    6:     OBJECT IDENTIFIER
          :       id_GostR3411_94_with_GostR3410_2001 ( 1 2 643 2 2 3)
403 05    0:     NULL
          :     }
405 03   65:   BIT STRING 0 unused bits
          :     1F 0E 5D C3 F6 B0 FC E8 8D BC 7C 8E 13 AE 64 BF
          :     2A 38 1E 9D 2C 7F 3D DC B0 CE 94 52 4A 75 D1 53
          :     B6 E3 BA 1F 34 92 B7 B6 C2 DB 1C E2 E3 51 AA B3
          :     79 FA E5 19 BD 75 5A 91 D8 AE F5 85 83 E1 5C 2C
          :   }
```

5 References

[GOST28147] "Cryptographic Protection for Data Processing System", GOST 28147-89, Gosudarstvennyi Standard of USSR, Government Committee of the USSR for Standards, 1989. (In Russian);

[GOSTR341094] "Information technology. Cryptographic Data Security. Produce and check procedures of Electronic Digital Signatures based on Asymmetric Cryptographic Algorithm.", GOST R 34.10-94, Gosudarstvennyi Standard of Russian Federation, Government Committee of the Russia for Standards, 1994. (In Russian);

[GOSTR34102001] "Information technology. Cryptographic data security. Signature and verification processes of [electronic] digital signature.", GOST R 34.10-2001, Gosudarstvennyi Standard of Russian Federation, Government Committee of the Russia for Standards, 2001. (In Russian);

[GOSTR341194] "Information technology. Cryptographic Data Security. Hashing function.", GOST R 34.10-94, Gosudarstvennyi Standard of Russian Federation, Government Committee of the Russia for Standards, 1994. (In Russian);

[RFDSL] Russian Federal Digital Signature Law, 10 Jan 2002 N1-FZ

[CPALGS] "Additional cryptographic algorithms for use with GOST 28147-89, GOST R 34.10-94, GOST R 34.10-2001, and GOST R 34.11-94 algorithms", V. Popov, I. Kurepkin, S. Leontiev, February 2004, draft-popov-cryptopro-cpalgs-01.txt work in progress;

[Schneier95] B. Schneier, Applied cryptography, second edition, John Wiley & Sons, Inc., 1995;

[RFC3280] Housley, R., Polk, W., Ford, W. and D. Solo,

and Certificate Revocation List (CRL) Profile", RFC 3280, April 2002.

[RFC3279] Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. L. Bassham, W. Polk, R. Housley. April 2002.

[RFC2119] Bradner, S., "Key Words for Use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[TLS] The TLS Protocol Version 1.0. T. Dierks, C. Allen. January 1999, RFC 2246.

[X.660] ITU-T Recommendation X.660 Information Technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER), 1997.

Acknowledgments

This document was created in accordance with "Russian Cryptographic Software Compatibility Agreement", signed by FGUE STC "Atlas", CRYPTO-PRO, Factor-TC, MD PREI, Infotecs GmbH, SPRCIS (SPbRCZI), Cryptocom, R-Alpha. The goal of this agreement is to achieve mutual compatibility of the products and solutions.

The authors wish to thank:

Microsoft Corporation Russia for provided information about company products and solutions, and also for technical consulting in PKI.

RSA Security Russia and Demos Co Ltd for active colaboration and critical help in creation of this document.

RSA Security Inc for compatibility testing of the proposed data formats while incorporating them into RSA Keon product.

Baltimore Technology plc for compatibility testing of the proposed data formats while incorporating them into UniCERT product.

Russ Hously (Vigil Security, LLC, housley@vigilsec.com) and

Vasilij Sakharov (DEMOS Co Ltd, svp@dol.ru) for initiative creating this document.

This document is based on a contribution of CRYPTO-PRO company. Any substantial use of the text from this document must reference CRYPTO-PRO. CRYPTO-PRO requests that all material mentioning or referencing this document identify this as "CRYPTO-PRO CPPK".

Author's Addresses

Serguei Leontiev
CRYPTO-PRO
38, Obraztsova,
Moscow, 127018, Russian Federation
EMail: lse@cryptopro.ru

Dennis Shefanovski
DEMOS Co Ltd
6/1, Ovchinnikovskaja naberezhnaya,
Moscow, 113035, Russian Federation
EMail: sdb@dol.ru

Alexandr Afanasiev
Factor-TC
office 711, 14, Presnenskij val,
Moscow, 123557, Russian Federation
EMail: aaaf@factor-ts.ru

Nikolaj Nikishin
Infotecs GmbH
p/b 35, 80-5, Leningradskij prospekt,
Moscow, 125315, Russian Federation
EMail: nikishin@infotecs.ru

Boleslav Izotov
FGUE STC "Atlas"
38, Obraztsova,
Moscow, 127018, Russian Federation
EMail: izotov@stcnet.ru

Elena Minaeva
MD PREI
build 3, 6A, Vtoroj Troitskij per.,
Moscow, Russian Federation
EMail: evminaeva@mo.msk.ru

Serguei Murugov
R-Alpha

4/1, Raspletina,
Moscow, 123060, Russian Federation
EMail: msm@office.ru

Igori Ustinov
Cryptocom
office 239, 51, Leninskij prospekt,
Moscow, 119991, Russian Federation
EMail: igus@cryptocom.ru

Anatolij Erkin
SPRCIS (SPbRCZI)
1, Obrucheva,
St.Petersburg, 195220, Russian Federation
EMail: erkin@nevsky.net

Full Copyright Statement

ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.