PKIX Working Group Serguei Leontiev, CRYPTO-PRO
Internet Draft Dennis Shefanovskij, DEMOS Co Ltd
Expires August 9, 2004 February 9, 2004
Intended Category: Informational

Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, corresponding to the algorithms GOST R 34.10-94, GOST R 34.10-2001, GOST R 34.11-94

<draft-ietf-pkix-gost-cppk-00.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 establishment algorithms based on GOST R 34.10-94 public keys, key establishment 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 establishment algorithm VKO GOST R 34.10-94 [CPALGS];
* Key establishment algorithm VKO GOST R 34.10-2001 [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.

Parameters for this function are defined in section 6.2 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 establishment 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 establishment 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 6.3 of [CPALGS])
 * digestParamSet - parameters identifier for GOST R 34.11-94 (see section 6.2 of [CPALGS])
 * encryptionParamSet - optional parameters identifier for GOST 28147-89 (see section 6.1 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 establishment 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 establishment 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 6.4 of [CPALGS])
 * digestParamSet - parameters identifier for GOST R 34.11-94 (see section 6.2 of [CPALGS])
 * encryptionParamSet - optional parameters identifier for GOST 28147-89 (see section 6.1 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

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.

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

4 Appendix ASN.1 Moduls

4.1 Cryptographic-Gost-Useful-Definitions

```
Cryptographic-Gost-Useful-Definitions
    { iso(1) member-body(2) ru(643) rans(2) cryptopro(2)
      other(1) modules(1) cryptographic-Gost-Useful-Definitions(0)
1 }
DEFINITIONS ::=
BEGIN
-- EXPORTS All --
-- The types and values defined in this module are exported for
-- use in the other ASN.1 modules contained within the Russian
-- Cryptography "GOST" & "GOST R" Specifications, and for the use
-- of other applications which will use them to access Russian
-- Cryptography services. Other applications may use them for
-- their own purposes, but this will not constrain extensions and
-- modifications needed to maintain or improve the Russian
-- Cryptography service.
  -- Crypto-Pro OID branch
    id-CryptoPro OBJECT IDENTIFIER ::=
        { iso(1) member-body(2) ru(643) rans(2) cryptopro(2) }
    id-CryptoPro-algorithms OBJECT IDENTIFIER ::=
        id-CryptoPro
    id-CryptoPro-modules OBJECT IDENTIFIER ::=
        { id-CryptoPro other(1) modules(1) }
    id-CryptoPro-hashes OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms hashes(30) }
    id-CryptoPro-encrypts OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms encrypts(31) }
    id-CryptoPro-signs OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms signs(32) }
```

```
    id-CryptoPro-exchanges OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms exchanges(33) }
    id-CryptoPro-extensions OBJECT IDENTIFIER ::=
            { id-CryptoPro extensions(34) }
    id-CryptoPro-ecc-signs OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms ecc-signs(35) }
    id-CryptoPro-ecc-exchanges OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms ecc-exchanges(36) }
    id-CryptoPro-private-keys OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms private-keys(37) }
    id-CryptoPro-policyQt OBJECT IDENTIFIER ::=
            { id-CryptoPro policyQt(39) }
    id-CryptoPro-policyIds OBJECT IDENTIFIER ::=
      { id-CryptoPro policyIds(38) }
    id-CryptoPro-attributes OBJECT IDENTIFIER ::=
      { id-CryptoPro-algorithms attributes(38) }
```



```
    id-CryptoPro-pkixcmp-infos OBJECT IDENTIFIER ::=
      { id-CryptoPro-algorithms pkixcmp-infos(41) }
  -- ASN.1 modules of Russian Cryptography "GOST" & "GOST R"
  -- Specifications
    cryptographic-Gost-Useful-Definitions OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules
          cryptographic-Gost-Useful-Definitions(0) 1 }
  -- GOST R 34.11-94

    gostR3411-94-DigestSyntax OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules gostR3411-94-DigestSyntax(1) 1 }
    gostR3411-94-ParamSetSyntax OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules gostR3411-94-ParamSetSyntax(7) 1 }
  -- GOST R 34.10-94

    gostR3410-94-PKISyntax OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules  gostR3410-94-PKISyntax(2) 1 }
    gostR3410-94-SignatureSyntax OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules gostR3410-94-SignatureSyntax(3) 1 }
    gostR3410-94-EncryptionSyntax OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules gostR3410-94-EncryptionSyntax(5) 2 }
    gostR3410-94-ParamSetSyntax OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules  gostR3410-94-ParamSetSyntax(8) 1 }
  -- GOST R 34.10-2001
```

```
    gostR3410-2001-PKISyntax OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules  gostR3410-2001-PKISyntax(9) 1 }
    gostR3410-2001-SignatureSyntax OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules
          gostR3410-2001-SignatureSyntax(10) 1 }
    gostR3410-2001-EncryptionSyntax OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules
          gostR3410-2001-EncryptionSyntax(11) 2 }
    gostR3410-2001-ParamSetSyntax OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules
          gostR3410-2001-ParamSetSyntax(12) 1 }
  -- GOST 28147-89

    gost28147-89-EncryptionSyntax OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules gost28147-89-EncryptionSyntax(4) 1 }
    gost28147-89-ParamSetSyntax OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules gost28147-89-ParamSetSyntax(6) 1 }
  -- Extended Key Usage for Crypto-Pro

    gost-CryptoPro-ExtendedKeyUsage OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules
          gost-CryptoPro-ExtendedKeyUsage(13) 1 }
  -- Crypto-Pro Private keys
```



```
    gost-CryptoPro-PrivateKey OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules gost-CryptoPro-PrivateKey(14) 1 }
  -- Crypto-Pro Policy
    gost-CryptoPro-Policy OBJECT IDENTIFIER ::=
        { id-CryptoPro-modules gost-CryptoPro-Policy(15) 1 }
  -- Crypto-Pro PKIXCMP structures

    gost-CryptoPro-PKIXCMP OBJECT IDENTIFIER ::=
      { id-CryptoPro-modules gost-CryptoPro-PKIXCMP(16) 1 }

  -- External ASN.1 modules for Russian Cryptography
    id-external-PKIX1Explicit93 OBJECT IDENTIFIER ::=
        { iso(1) identified-organization(3)
          dod(6) internet(1) security(5) mechanisms(5) pkix(7)
          id-mod(0) id-pkix1-explicit-93(3)
        }
  -- Useful types
    ALGORITHM-IDENTIFIER ::= TYPE-IDENTIFIER
```

```
    AlgorithmIdentifier { ALGORITHM-IDENTIFIER:InfoObjectSet } ::=
        SEQUENCE {
            algorithm
          ALGORITHM-IDENTIFIER.&id({InfoObjectSet}),
            parameters
          ALGORITHM-IDENTIFIER.&Type({InfoObjectSet} {@algorithm})
              OPTIONAL
        }
END -- Cryptographic-Gost-Useful-Definitions
```

4.2 GostR3411-94-DigestSyntax

```
GostR3411-94-DigestSyntax
    { iso(1) member-body(2) ru(643) rans(2) cryptopro(2)
      other(1) modules(1) gostR3411-94-DigestSyntax(1) 1 }
DEFINITIONS ::=
BEGIN
-- EXPORTS All --
-- The types and values defined in this module are exported for
-- use in the other ASN.1 modules contained within the Russian
-- Cryptography "GOST" & "GOST R" Specifications, and for the use
-- of other applications which will use them to access Russian
-- Cryptography services. Other applications may use them for
-- their own purposes, but this will not constrain extensions and
-- modifications needed to maintain or improve the Russian
-- Cryptography service.
    IMPORTS
        id-CryptoPro-algorithms, id-CryptoPro-hashes,
        gost28147-89-EncryptionSyntax,
        cryptographic-Gost-Useful-Definitions
```



```
        FROM Cryptographic-Gost-Useful-Definitions
            { iso(1) member-body(2) ru(643) rans(2)
              cryptopro(2) other(1) modules(1)
              cryptographic-Gost-Useful-Definitions(0) 1 }
        Gost28147-89-Data, Gost28147-89-UZ
        FROM Gost28147-89-EncryptionSyntax
             gost28147-89-EncryptionSyntax
        AlgorithmIdentifier, ALGORITHM-IDENTIFIER
        FROM Cryptographic-Gost-Useful-Definitions
             cryptographic-Gost-Useful-Definitions
    ;
```

```
  -- GOST R 34.11-94 OID
    id-GostR3411-94 OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms gostR3411-94(9) }
  -- GOST R 34.11-94 Cryptographic Parameters Set OIDs
    id-GostR3411-94-TestParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-hashes test(0) }
    id-GostR3411-94-CryptoProParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-hashes cryptopro(1) }
  -- GOST R 34.11-94 Data Types
    GostR3411-94-Data ::= Gost28147-89-Data
    GostR3411-94-Digest ::= OCTET STRING (SIZE (32))
  -- GOST R 34.11-94 Digest Parameters & Algorithms
    GostR3411-94-DigestParameters ::=
        OBJECT IDENTIFIER (
          id-GostR3411-94-TestParamSet |     -- Only for tests use
          id-GostR3411-94-CryptoProParamSet
        )
    GostR3411-94-DigestAlgorithms ALGORITHM-IDENTIFIER ::= {
        { NULL IDENTIFIED BY id-GostR3411-94 } |
          -- Assume id-GostR3411-94-CryptoProParamSet
        { GostR3411-94-DigestParameters
          IDENTIFIED BY id-GostR3411-94 }
    }
END -- GostR3411-94-DigestSyntax
```

4.3 GostR3410-94-PKISyntax

```
GostR3410-94-PKISyntax
    { iso(1) member-body(2) ru(643) rans(2) cryptopro(2)
      other(1) modules(1) gostR3410-94-PKISyntax(2) 1 }
DEFINITIONS ::=
BEGIN
-- EXPORTS All --
-- The types and values defined in this module are exported for
-- use in the other ASN.1 modules contained within the Russian
-- Cryptography "GOST" & "GOST R" Specifications, and for the use
-- of other applications which will use them to access Russian
```



```
-- Cryptography services. Other applications may use them for
-- their own purposes, but this will not constrain extensions and
-- modifications needed to maintain or improve the Russian
-- Cryptography service.
```

```
    IMPORTS
        id-CryptoPro-algorithms,
        id-CryptoPro-signs, id-CryptoPro-exchanges,
        gost28147-89-EncryptionSyntax,
        gostR3411-94-DigestSyntax,
        cryptographic-Gost-Useful-Definitions
        FROM Cryptographic-Gost-Useful-Definitions
            { iso(1) member-body(2) ru(643) rans(2)
              cryptopro(2) other(1) modules(1)
              cryptographic-Gost-Useful-Definitions(0) 1 }
        id-Gost28147-89-TestParamSet,
        id-Gost28147-89-CryptoPro-A-ParamSet,
        id-Gost28147-89-CryptoPro-B-ParamSet,
        id-Gost28147-89-CryptoPro-C-ParamSet,
        id-Gost28147-89-CryptoPro-D-ParamSet,
        id-Gost28147-89-CryptoPro-Simple-A-ParamSet,
        id-Gost28147-89-CryptoPro-Simple-B-ParamSet,
        id-Gost28147-89-CryptoPro-Simple-C-ParamSet,
        id-Gost28147-89-CryptoPro-Simple-D-ParamSet
        FROM Gost28147-89-EncryptionSyntax
             gost28147-89-EncryptionSyntax
        id-GostR3411-94-TestParamSet,
        id-GostR3411-94-CryptoProParamSet
        FROM GostR3411-94-DigestSyntax gostR3411-94-DigestSyntax
        AlgorithmIdentifier, ALGORITHM-IDENTIFIER
        FROM Cryptographic-Gost-Useful-Definitions
             cryptographic-Gost-Useful-Definitions
    ;
  -- GOST R 34.10-94 OIDs
    id-GostR3410-94 OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms gostR3410-94(20) }
    id-GostR3411-94-with-GostR3410-94 OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms
          gostR3411-94-with-gostR3410-94(4) }
  -- GOST R 34.10-94 Public Key Cryptographic Parameters Set OIDs
    id-GostR3410-94-TestParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-signs test(0) }
    id-GostR3410-94-CryptoPro-A-ParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-signs cryptopro-A(2) }
    id-GostR3410-94-CryptoPro-B-ParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-signs cryptopro-B(3) }
    id-GostR3410-94-CryptoPro-C-ParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-signs cryptopro-C(4) }
    id-GostR3410-94-CryptoPro-D-ParamSet OBJECT IDENTIFIER ::=
```

```
        { id-CryptoPro-signs cryptopro-D(5) }
    id-GostR3410-94-CryptoPro-XchA-ParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-exchanges cryptopro-XchA(1) }
    id-GostR3410-94-CryptoPro-XchB-ParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-exchanges cryptopro-XchB(2) }
    id-GostR3410-94-CryptoPro-XchC-ParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-exchanges cryptopro-XchC(3) }
  -- GOST R 34.10-94 Data Types
    GostR3410-94-CertificateSignature ::=
        BIT STRING ( SIZE(256..512) )
    GostR3410-94-PublicKeyOctetString ::=
        OCTET STRING ( SIZE(
               64 | -- Only for tests use
               128
               ) )
    GostR3410-94-PublicKey ::=
        BIT STRING ( SIZE(16..1048) )
               -- Container for GostR3410-94-PublicKeyOctetString
    GostR3410-94-PublicKeyParameters ::=
            SEQUENCE {
            publicKeyParamSet
          OBJECT IDENTIFIER (
              id-GostR3410-94-TestParamSet | -- Only for tests use
              id-GostR3410-94-CryptoPro-A-ParamSet |
              id-GostR3410-94-CryptoPro-B-ParamSet |
              id-GostR3410-94-CryptoPro-C-ParamSet |
              id-GostR3410-94-CryptoPro-D-ParamSet |
              id-GostR3410-94-CryptoPro-XchA-ParamSet |
              id-GostR3410-94-CryptoPro-XchB-ParamSet |
              id-GostR3410-94-CryptoPro-XchC-ParamSet
          ),
            digestParamSet
          OBJECT IDENTIFIER (
              id-GostR3411-94-TestParamSet | -- Only for tests use
              id-GostR3411-94-CryptoProParamSet
          ),
            encryptionParamSet
          OBJECT IDENTIFIER (
              id-Gost28147-89-TestParamSet | -- Only for tests use
              id-Gost28147-89-CryptoPro-A-ParamSet |
              id-Gost28147-89-CryptoPro-B-ParamSet |
              id-Gost28147-89-CryptoPro-C-ParamSet |
              id-Gost28147-89-CryptoPro-D-ParamSet |
              id-Gost28147-89-CryptoPro-Simple-A-ParamSet |
              id-Gost28147-89-CryptoPro-Simple-B-ParamSet |
              id-Gost28147-89-CryptoPro-Simple-C-ParamSet |
              id-Gost28147-89-CryptoPro-Simple-D-ParamSet
```

```
          ) OPTIONAL
```



```
        }
    GostR3410-94-PublicKeyAlgorithms  ALGORITHM-IDENTIFIER ::= {
        { GostR3410-94-PublicKeyParameters IDENTIFIED BY
               id-GostR3410-94 }
    }
    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 }
        }
END -- GostR3410-94-PKISyntax
```

4.4 GostR3410-2001-PKISyntax

```
GostR3410-2001-PKISyntax
    { iso(1) member-body(2) ru(643) rans(2) cryptopro(2)
      other(1) modules(1) gostR3410-2001-PKISyntax(9) 1 }
DEFINITIONS ::=
BEGIN
-- EXPORTS All --
-- The types and values defined in this module are exported for
-- use in the other ASN.1 modules contained within the Russian
-- Cryptography "GOST" & "GOST R" Specifications, and for the use
-- of other applications which will use them to access Russian
-- Cryptography services. Other applications may use them for
-- their own purposes, but this will not constrain extensions and
-- modifications needed to maintain or improve the Russian
-- Cryptography service.
    IMPORTS
        id-CryptoPro-algorithms,
        id-CryptoPro-ecc-signs, id-CryptoPro-ecc-exchanges,
        gost28147-89-EncryptionSyntax,
        gostR3411-94-DigestSyntax,
        cryptographic-Gost-Useful-Definitions
        FROM Cryptographic-Gost-Useful-Definitions
            { iso(1) member-body(2) ru(643) rans(2)
              cryptopro(2) other(1) modules(1)
              cryptographic-Gost-Useful-Definitions(0) 1 }
```

```
        id-Gost28147-89-TestParamSet,
        id-Gost28147-89-CryptoPro-A-ParamSet,
        id-Gost28147-89-CryptoPro-B-ParamSet,
        id-Gost28147-89-CryptoPro-C-ParamSet,
        id-Gost28147-89-CryptoPro-D-ParamSet,
        id-Gost28147-89-CryptoPro-Simple-A-ParamSet,
        id-Gost28147-89-CryptoPro-Simple-B-ParamSet,
        id-Gost28147-89-CryptoPro-Simple-C-ParamSet,
```



```
        id-Gost28147-89-CryptoPro-Simple-D-ParamSet
        FROM Gost28147-89-EncryptionSyntax
             gost28147-89-EncryptionSyntax
        id-GostR3411-94-TestParamSet,
        id-GostR3411-94-CryptoProParamSet
        FROM GostR3411-94-DigestSyntax gostR3411-94-DigestSyntax
        AlgorithmIdentifier, ALGORITHM-IDENTIFIER
        FROM Cryptographic-Gost-Useful-Definitions
             cryptographic-Gost-Useful-Definitions
    ;
  -- GOST R 34.10-2001 OIDs
    id-GostR3410-2001 OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms gostR3410-2001(19) }
    id-GostR3411-94-with-GostR3410-2001 OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms
          gostR3411-94-with-gostR3410-2001(3) }
  -- GOST R 34.10-2001 Public Key Cryptographic Parameters Set OIDs
    id-GostR3410-2001-TestParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-ecc-signs test(0) }
    id-GostR3410-2001-CryptoPro-A-ParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-ecc-signs cryptopro-A(1) }
    id-GostR3410-2001-CryptoPro-B-ParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-ecc-signs cryptopro-B(2) }
    id-GostR3410-2001-CryptoPro-C-ParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-ecc-signs cryptopro-C(3) }
    id-GostR3410-2001-CryptoPro-XchA-ParamSet
        OBJECT IDENTIFIER ::=
          { id-CryptoPro-ecc-exchanges cryptopro-XchA(0) }
    id-GostR3410-2001-CryptoPro-XchB-ParamSet
        OBJECT IDENTIFIER ::=
          { id-CryptoPro-ecc-exchanges cryptopro-XchB(1) }
  -- GOST R 34.10-2001 Data Types
    GostR3410-2001-CertificateSignature ::=
```

```
        BIT STRING ( SIZE(256..512) )
    GostR3410-2001-PublicKeyOctetString ::=
        OCTET STRING ( SIZE(64) )
    GostR3410-2001-PublicKey ::=
        BIT STRING ( SIZE(16..524) )
               -- Container for GostR3410-2001-PublicKeyOctetString
    GostR3410-2001-PublicKeyParameters ::=
            SEQUENCE {
            publicKeyParamSet
          OBJECT IDENTIFIER (
              id-GostR3410-2001-TestParamSet |    -- Only for tests use
              id-GostR3410-2001-CryptoPro-A-ParamSet |
              id-GostR3410-2001-CryptoPro-B-ParamSet |
              id-GostR3410-2001-CryptoPro-C-ParamSet |
              id-GostR3410-2001-CryptoPro-XchA-ParamSet |
```



```
              id-GostR3410-2001-CryptoPro-XchB-ParamSet
          ),
            digestParamSet
          OBJECT IDENTIFIER (
              id-GostR3411-94-TestParamSet | -- Only for tests use
              id-GostR3411-94-CryptoProParamSet
          ),
            encryptionParamSet
          OBJECT IDENTIFIER (
              id-Gost28147-89-TestParamSet | -- Only for tests use
              id-Gost28147-89-CryptoPro-A-ParamSet |
              id-Gost28147-89-CryptoPro-B-ParamSet |
              id-Gost28147-89-CryptoPro-C-ParamSet |
              id-Gost28147-89-CryptoPro-D-ParamSet |
              id-Gost28147-89-CryptoPro-Simple-A-ParamSet |
              id-Gost28147-89-CryptoPro-Simple-B-ParamSet |
              id-Gost28147-89-CryptoPro-Simple-C-ParamSet |
              id-Gost28147-89-CryptoPro-Simple-D-ParamSet
          ) OPTIONAL
        }
    GostR3410-2001-PublicKeyAlgorithms  ALGORITHM-IDENTIFIER ::= {
        { GostR3410-2001-PublicKeyParameters IDENTIFIED BY
               id-GostR3410-2001 }
    }
    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 }
      }
END -- GostR3410-2001-PKISyntax
```

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-crypto-pro-cpalgs-00.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, "Internet X.509 Public Key Infrastructure Certificate 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