PKIX Working Group Serguei Leontiev, CRYPTO-PRO
INTERNET-DRAFT Dennis Shefanovskij, DEMOS Co Ltd
Expires December 15, 2003 June 15, 2003
Intended Category: Informational

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

<draft-leontiev-cryptopro-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 [RFC 3279], "Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile" and, correspondingly, [RFC 3280], "Internet X.509 Public Key Infrastructure: Certificate and Certificate Revocation List (CRL) Profile". All realizations of this

specification also MUST correspond [RFC 3280].

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 [RFC 2119].

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 [RFC 3279], "Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile" and, correspondingly, [RFC 3280], "Internet X.509 Public Key Infrastructure: Certificate and Certificate Revocation List (CRL) Profile". All realizations of this specification MUST also correspond [RFC 3280].

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

This document defines one-way hash-function GOST R 34.11-94 [GOST3411] for use in the generation of 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 algorithms based on GOST R 34.10-94 public keys [CPALGS];
* Key establishment algorithms based on GOST R 34.10-2001 public keys [CPALGS].

2 Algorithm Support

This section is review of cryptographic algorithms, which may be used within the Internet X.509 certificates and CRL profile [RFC 3280]. The one-way hash functions and digital signature algorithms, which may be used to sign certificates and CRLs, and identifies object identifiers (OIDs) for public keys contained in a certificate are also described in this section.

The appropriate CA and/or applications MUST support digital signatures and public keys fully for one of the specified algorithms. Hence when using any of the algorithms identified in this specification in CA and/or applications MUST support them as described below.

2.1 One-way Hash Functions

This section identifies one-way, collision free hash function GOST R 34.11-94 - the only one could 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 [RFC 3280].

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 could be found in [GOSTR3411] in Russian, [Schneier95] ch. 18.11, p. 454. contain the brief technical description in English.

The initial value (IV) and S-box are optional for algorithm parameters (Algorithm Parameters part in [GOST28147] in Russian, description in English see in [Schneier95] ch. 14.1, p. 331). The Standard [GOST3411] does not define hash function algorithm parameters, which ought to be set by OID in software code. The Parameters for OID prescribed below are included in appendix.

2.2 Signature Algorithms

In according to [RFC 3280] the Certificates and CRL may be signed with either GOST R 34.10-94 or with GOST R 34.10-2001 signature algorithms. The signatureAlgorithm field of Certificates or CertificateRevocationList for certificate or CRL indicates the algorithm ID used for signature and associated parameters, essential as OID. In case of omitted parameters, these ones are generated hereditarily from top of issuers. This section also defines algorithm identifiers and parameters that MUST be used in the signatureAlgorithm field in a Certificate or CertificateRevocationList.

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. The contents of the parameters component for each algorithm may vary and details are provided below for each algorithm separately.

The data to be signed (per example, the one-way hash function output

value) is preformatted for the signature algorithm to be used. Then, a private key operation is performed to generate the signature value. This signature value is then ASN.1 encoded as a BIT STRING and included in the Certificate or CertificateRevocationList in the signatureValue field.

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 CertificateRevocationList is the next:

```
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 } }
```

See chapter Algorithms Parameters for further details.

When the id-GostR3411-94-with-GostR3410-94 algorithm identifier appears in an AlgorithmIdentifier and parameters are omitted, the software MUST use the parameters from the signer's public key.

Signature algorithm GOST R 34.10-94 generates digital signature in the form of a binary 512-bit vector (<r'>256||<s>256). This vector is encoded as two data blocks, being given after decoding at the signature verification algorithm input. At first, <r'>256 block then <s>256 block. signatureValue field BIT STRING type:

```
GostR3410-94-SignatureValue ::= BIT STRING
```

At that, least-significant of the first octet (GostR3410EncryptedDigest[0]) corresponds to least-significant (1-st) of vector <r'>256||<s>256 (s1 = (GostR3410EncryptedDigest[0] & 1)). Whereas most-significant of 64-th octet (GostData[31]) corresponds to most-significant (512-d) of vector <r'>256||<s>256 (r'256 = ((GostR3410EncryptedDigest[63] & 0x80)>>7)).

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 CertificateRevocationList 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 } }
```

See chapter Algorithms Parameters for further details.

When the id-GostR3411-94-with-GostR3410-2001 algorithm identifier appears in an AlgorithmIdentifier and parameters are omitted, the MUST use the parameters concerned with the public key of the issuer of this certificate or CRL.

Signature algorithm GOST R 34.10-2001 generates digital signature in the form of a binary 512-bit vector (<r'>256||<s>256). This vector encoded as two data blocks, piping after encoding to input of signature verification algorithm in according to GOST R 34.10-2001, first block <r>256, then block <s>256. The field signatureValue has type BIT STRING:

```
GostR3410-2001-CertificateSignature ::= BIT STRING
```

At that, least-significant of the first octet (GostR3410EncryptedDigest[0]) corresponds to least-significant (1-st)

of vector <r>256||<s>256 (s1 = (GostR3410EncryptedDigest[0] & 1)). Whereas most-significant of 64-th octet (GostData[31]) corresponds to most-significant (512-d) of vector <r>256||<s>256 (r256 = ((GostR3410EncryptedDigest[63] & 0x80)>>7)).

2.3 Subject Public Key Algorithms

In according to [RFC 3280] 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 parameters encoding scheme for public key including into certificate. Public key GOST R 34.10-94 could be used for digital signature verification by GOST R 34.10-94 [GOSTR341094] algorithm and for key exchange algorithm based on GOST R 34.10-94 [CPALGS].

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) }
```

An assumed cryptographic key usage could be pointed in keyUsage field [RFC 3280]. The usage the same key for signature and for key establishment is NOT RECOMMENDED, but possible.

SubjectPublicKeyInfo sintax of primary certificate context for id-GostR3410-94 algorithm produced in following ASN.1 structure:

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

GostR3410-94-PublicKeyAlgorithms  ALGORITHM-IDENTIFIER ::= {
    { GostR3410-94-PublicKeyParameters IDENTIFIED BY
```

```
                    id-GostR3410-94 } }

   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;
* digestParamSet - parameters identifier for GOST R 34.11-94;
* encryptionParamSet - optional parameters identifier for GOST 28147-89 MAY be presented anytime and MUST be presented if keyUsage is keyAgreement or keyEnchiperment.

AlgorithmIdentifier within subjectPublicKeyInfo could takes place only if a certificate contains these parameters. 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, then certificate users MUST reject it.

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 parameters encoding for public key including in certificate. Public key GOST R 34.10-2001 could be used for digital signature generation by GOST R 34.10-2001 [GOSTR34102001] algorithm and for key exchange algorithm based on GOST R 34.10-2001 [CPALGS]. Public key OID for GOST R 34.10-2001 is:

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

Assumed cryptographic key usage could be pointed in keyUsage field [RFC 3280]. The usage of the same key for digital signature and key establishment is NOT RECOMMENDED, but possible.

SubjectPublicKeyInfo syntax of primary certificate context for id-GostR3410-94 algorithm produced in following ASN.1 structure:

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

GostR3410-2001-PublicKeyAlgorithms  ALGORITHM-IDENTIFIER ::= {
```

```
    { GostR3410-2001-PublicKeyParameters IDENTIFIED BY
                    id-GostR3410-2001 } }

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

* publicKeyParamSet - public key parameters identifier for GOST R 34.10-2001;
* digestParamSet - parameters identifier for GOST R 34.11-94;
* encryptionParamSet - optional parameters identifier for GOST 28147-89 MAY be presented anytime and MUST be presented if keyUsage differs from digitalSignature, nonRepudiation, keyCertSign and cRLSign.

AlgorithmIdentifier within subjectPublicKeyInfo takes place only if a certificate could contain parameters. 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, then certificate users MUST reject it.

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 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 Algorithm Parameters

Cryptographic algorithm parameters in certificates and CRL are indicated by appropriate OID. Algorithm parameters OID, proposed hereby, are the next.

For hash function parameters GOST R 34.11-94:

id-CryptoPro-hashes OBJECT IDENTIFIER ::=

```
      { iso(1) member-body(2) ru(643) rans(2) cryptopro(2) hashes(30) }

   id-GostR3411-94-TestParamSet OBJECT IDENTIFIER ::=
      { id-CryptoPro-hashes test(0) }

   id-GostR3411-94-CryptoProParamSet OBJECT IDENTIFIER ::=
      { id-CryptoPro-hashes cryptopro(1) }
```

For public key parameters GOST R 34.10-94:

```
   id-CryptoPro-signs OBJECT IDENTIFIER ::=
      { iso(1) member-body(2) ru(643) rans(2) cryptopro(2) signs(32) }

   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) }
```

For public key parameters GOST R 34.10-2001:

```
   id-CryptoPro-ecc-signs OBJECT IDENTIFIER ::= { iso(1) member-body(2)
   ru(643) rans(2) cryptopro(2) ecc-signs(35) }

   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) }
```

Specific further parameter values are presented in Appendix.

If any organization needs to assign their own parameters, they should

be described and MUST be published in a way of appendix to this document or in other way in following forms:

3.1 GOST R 34.11-94 Parameters

Hash function parameters GOST R 34.11-94 according to [GOSTR3411]:

```
GostR3411-94-ParamSetParameters ::=
   SEQUENCE {
       hUZ Gost28147-89-UZ,
       h0  GostR3411-94-Digest
   }
```

hUZ - S-box [GOSTR3411], [GOST28147].

h0 - initializing value(IV) [GOSTR3411].

3.2 GOST R 34.10-94 Parameters

algorithm parameters GOST R 34.10-94 according to [GOSTR341094]:

```
GostR3410-94-ParamSetParameters ::=
   SEQUENCE {
       p       INTEGER,
       q       INTEGER,
       a       INTEGER,
       validationAlgorithm
      AlgorithmIdentifier {{
         GostR3410-94-ValidationAlgorithms
         }} OPTIONAL
   }
```

p - modulus, prime number, 2^1023<p<2^1024;
q - order of cyclic group, prime number, 2^254<q<2^256, q is a factor of p-1;
a - generator, integer, 1<a<p-1, at that aq (mod p) = 1;
validationAlgorithm - constant p, q and a calculating algorithm.

```
GostR3410-94-ValidationParameters ::=
      SEQUENCE {
          t       INTEGER,
          x0      INTEGER,
          c       INTEGER,
          d       INTEGER OPTIONAL
      }
```

t - bit length of p;
x0 - seed;

c - used for p and q generation;
d - used for a generation.

3.3 GOST R 34.10-2001 Parameters

Public key algorithm parameters GOST R 34.11-2001 according to [GOSTR34102001]:

```
GostR3410-2001-ParamSetParameters ::=
   SEQUENCE {
       abj CHOICE {
           ab SEQUENCE {
               a       INTEGER,
               b       INTEGER,
           },
           j       INTEGER,
       },
       p       INTEGER ,
       q       INTEGER ,
       x       INTEGER ,
       y       INTEGER
   }
```

a, b - coefficients a and b of the elliptic curve E;
j - invariant;
p - prime number - elliptic curve modulus;
q - prime number - order of cyclic group;
x, y - base point p coordinates.

4 Security Considerations

Parameter values for using cryptographic algorithms affect rigidity of information protection system. It is RECCOMENDED, that software applications verify signature values, subject public keys and algorithm parameters to conform to [GOSTR34102001], [GOSTR341094] standards prior to their use.

The algorithm parameters proposed hereby and described in this document, have been analyzed by special certification laboratory of Scientific and Technical Centre "ATLAS" and by Centre of Certificational Investigations in appropriate levels of target_of_evaluation (TOE).

In case of different parameters usage, it is RECCOMENDED that they are to be examined by authorized agency with an approved methods of cryptographic analysis.

5 Appendix ASN.1 Moduls

5.1 Cryptographic-Gost-Useful-Definitions

```
-- Copyright(C) CRYPTO-PRO Company
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(39) }
  -- 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
```

5.2 GostR3411-94-DigestSyntax

```
-- Copyright(C) CRYPTO-PRO Company
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
```

5.3 GostR3411-94-ParamSetSyntax

```
-- Copyright(C) CRYPTO-PRO Company
GostR3411-94-ParamSetSyntax
    { iso(1) member-body(2) ru(643) rans(2) cryptopro(2)
      other(1) modules(1) gostR3411-94-ParamSetSyntax(7) 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,
        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 }
        Gost28147-89-UZ
        FROM Gost28147-89-EncryptionSyntax
             gost28147-89-EncryptionSyntax
        id-GostR3411-94-TestParamSet,
        id-GostR3411-94-CryptoProParamSet,
        GostR3411-94-Digest
        FROM GostR3411-94-DigestSyntax gostR3411-94-DigestSyntax
        AlgorithmIdentifier, ALGORITHM-IDENTIFIER
        FROM Cryptographic-Gost-Useful-Definitions
             cryptographic-Gost-Useful-Definitions
    ;
  -- GOST R 34.11-94 Cryptographic Parameters Set:
  -- algorithm & parameters
  -- OID for Parameters Set imported from GostR3411-94-DigestSyntax
    GostR3411-94-ParamSetParameters ::=
        SEQUENCE {
            hUZ Gost28147-89-UZ,    -- S-Box for digest
            h0  GostR3411-94-Digest -- start digest value
        }
    GostR3411-94-ParamSetAlgorithms ALGORITHM-IDENTIFIER ::= {
        { GostR3411-94-ParamSetParameters IDENTIFIED BY
          id-GostR3411-94-TestParamSet
        } |
        { GostR3411-94-ParamSetParameters IDENTIFIED BY
          id-GostR3411-94-CryptoProParamSet
        }
    }
    -- GOST R 34.11-94 Tests parameters set
    -- (GOST R 34.11-94 Annex A. Test vector)
    gostR3411TestParamSetAI AlgorithmIdentifier
     {{ GostR3411-94-ParamSetAlgorithms }} ::=
        {
            algorithm
```

```
          id-GostR3411-94-TestParamSet,
            parameters
          GostR3411-94-ParamSetParameters:{
              hUZ
                 -- pi1 pi2 pi3 pi4 pi5 pi6 pi7 pi8
                 -- 4   E   5   7   6   4   D   1
                 -- A   B   8   D   C   B   B   F
                 -- 9   4   1   A   7   A   4   D
                 -- 2   C   D   1   1   0   1   0
                 -- D   6   A   0   5   7   3   5
                 -- 8   D   3   8   F   2   F   7
                 -- 0   F   4   9   D   1   5   A
                 -- E   A   2   F   8   D   9   4
                 -- 6   2   E   E   4   3   0   9
                 -- B   3   F   4   A   6   A   2
                 -- 1   8   C   6   9   8   E   3
                 -- C   1   7   C   E   5   7   E
                 -- 7   0   6   B   0   9   6   6
                 -- F   7   0   2   3   C   8   B
                 -- 5   5   9   5   B   F   2   8
                 -- 3   9   B   3   2   E   C   C
                   '4E5764D1AB8DCBBF941A7A4D2CD11010D6A057358D38F2F7
0F49D15AEA2F8D9462EE4309B3F4A6A218C698E3C17CE57E706B0966F7023C8B559
5BF2839B32ECC'H,
              h0   '0000000000000000000000000000000000000000000000000
0000000000000000'H
          }
        }
    -- CryptoPro parameters
    gostR3411CryptoProParamSetAI AlgorithmIdentifier
    {{ GostR3411-94-ParamSetAlgorithms }} ::=
        {
            algorithm
          id-GostR3411-94-CryptoProParamSet,
            parameters
          GostR3411-94-ParamSetParameters:{
              hUZ
                 -- pi1 pi2 pi3 pi4 pi5 pi6 pi7 pi8
                 -- A   5   7   4   7   7   D   1
                 -- 4   F   F   A   6   6   E   3
                 -- 5   4   C   7   4   2   4   A
                 -- 6   0   E   C   B   4   1   9
                 -- 8   2   9   0   9   D   7   5
                 -- 1   D   4   F   C   9   0   B
                 -- 3   B   1   2   2   F   5   4
                 -- 7   9   0   8   A   0   A   F
                 -- D   1   3   E   1   A   3   8
                 -- C   7   B   1   8   1   C   6
```

```
                -- E   6   5   6   0   5   8   7
                -- 0   3   2   5   E   B   F   E
                -- 9   C   6   D   F   8   6   D
                -- 2   E   A   B   D   E   2   0
                -- B   A   8   9   3   C   9   2
                -- F   8   D   3   5   3   B   C
               'A57477D14FFA66E354C7424A60ECB41982909D751D4FC90B3B122F5
47908A0AFD13E1A38C7B181C6E65605870325EBFE9C6DF86D2EABDE20BA893C92F8
D353BC'H,
             h0   '0000000000000000000000000000000000000000000000000000
0000000000000000'H
          }
        }
END -- GostR3411-94-ParamSetSyntax
```

5.4 GostR3410-94-PKISyntax

```
-- Copyright(C) CRYPTO-PRO Company
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
```

5.5 GostR3410-94-ParamSetSyntax

```
-- Copyright(C) CRYPTO-PRO Company
GostR3410-94-ParamSetSyntax
    { iso(1) member-body(2) ru(643) rans(2) cryptopro(2)
      other(1) modules(1) gostR3410-94-ParamSetSyntax(8) 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,
        gostR3410-94-PKISyntax,
        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-GostR3410-94,
        id-GostR3410-94-TestParamSet,
        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
        FROM GostR3410-94-PKISyntax gostR3410-94-PKISyntax
        AlgorithmIdentifier, ALGORITHM-IDENTIFIER
        FROM Cryptographic-Gost-Useful-Definitions
             cryptographic-Gost-Useful-Definitions
    ;
  -- GOST R 34.10-94 Public Key Cryptographic Parameters Set:
  -- algorithm & parameters
  -- OID for Parameters Set imported from GostR3410-94-PKISyntax
    GostR3410-94-ParamSetParameters ::=
        SEQUENCE {
            t    INTEGER (512 | 1024),  -- 512 - only for tests use
            p    INTEGER (
               167597599124282463744675312477573076593492072757404917221
544518046522050375919337210023428727086292846125398227331075635671 9
235351493321243304206125760513
               ..
               134078079299425970995740249982058461274793658205923933777
235614437217640300735469768018742981669034276900318581864860508537 5
388281194656994643364900608409 5
```

```
               |
               112355820928894744233081574424314045851123561183894160795
893800723582922378438101957942798326504710013200071174919620848536 7
436055090103890580296441496713277361049333905409282976888872507788 0
882465817684505312860552384417646403930092119569408801702322709406 9
177866436399967028711549822690522097706015140085 77
               ..
               179769313486231590772930519078902473361797697894230657273
430081157732675805500963132708477322407536021120113879871393357658 7
897688144166224928474306394741243777678934248654852763022196012460 9
411945308295208500576883815068234246288147391311054082723716335051 0
684586298239947245938479716304835356329624224137215
               ),   -- 2^509 < p < 2^512 or 2^1020 < p < 2^1024
            q       INTEGER (
               289480223093290488558927462521719769633174961664101410098
643960019782824099 85
               ..
               115792089237316195423570985008687907853269984665640564039
457584007913129639935
               ),   -- 2^254 < q < 2^256
            a       INTEGER (
               2
               ..
               179769313486231590772930519078902473361797697894230657273
430081157732675805500963132708477322407536021120113879871393357658 7
897688144166224928474306394741243777678934248654852763022196012460 9
411945308295208500576883815068234246288147391311054082723716335051 0
684586298239947245938479716304835356329624224137214
               ),   -- 1 < a < p-1 < 2^1024-1
            validationAlgorithm
          AlgorithmIdentifier {{
                    GostR3410-94-ValidationAlgorithms
               }} OPTIONAL
       }
   GostR3410-94-ParamSetAlgorithm ALGORITHM-IDENTIFIER ::= {
       { GostR3410-94-ParamSetParameters IDENTIFIED BY
             id-GostR3410-94-TestParamSet } |
       { GostR3410-94-ParamSetParameters IDENTIFIED BY
             id-GostR3410-94-CryptoPro-A-ParamSet  } |
       { GostR3410-94-ParamSetParameters IDENTIFIED BY
             id-GostR3410-94-CryptoPro-B-ParamSet  } |
       { GostR3410-94-ParamSetParameters IDENTIFIED BY
             id-GostR3410-94-CryptoPro-C-ParamSet  } |
       { GostR3410-94-ParamSetParameters IDENTIFIED BY
             id-GostR3410-94-CryptoPro-D-ParamSet  } |
       { GostR3410-94-ParamSetParameters IDENTIFIED BY
             id-GostR3410-94-CryptoPro-XchA-ParamSet  } |
       { GostR3410-94-ParamSetParameters IDENTIFIED BY
```

```
              id-GostR3410-94-CryptoPro-XchB-ParamSet  } |
        { GostR3410-94-ParamSetParameters IDENTIFIED BY
              id-GostR3410-94-CryptoPro-XchC-ParamSet  }
    }
  -- GOST R 34.10-94 validation/constructor
    id-GostR3410-94-a         OBJECT IDENTIFIER ::=
        { id-GostR3410-94 a(1) }
    id-GostR3410-94-aBis OBJECT IDENTIFIER ::=
        { id-GostR3410-94 aBis(2) }
    id-GostR3410-94-b         OBJECT IDENTIFIER ::=
        { id-GostR3410-94 b(3) }
    id-GostR3410-94-bBis OBJECT IDENTIFIER ::=
        { id-GostR3410-94 bBis(4) }
    GostR3410-94-ValidationParameters ::=
        SEQUENCE {
            t    INTEGER (512 | 1024),  -- 512 - only for tests use
            x0   INTEGER (0 .. 65535),
            c    INTEGER (0 .. 65535),
            d    INTEGER (
               2
               ..
               179769313486231590772930519078902473361797697894230657273
430081157732675805500963132708477322407536021120113879871393357658
897688144166224928474306394741243777678934248654852763022196012460
411945308295208500576883815068234246288147391311054082723716335051
684586298239947245938479716304835356329624224137214
               )    -- 1 < d < p-1 < 2^1024-1
               OPTIONAL
        }
    GostR3410-94-ValidationBisParameters ::=
        SEQUENCE {
            t    INTEGER (512 | 1024),  -- 512 - only for tests use
            x0   INTEGER (0 .. 4294967295),
            c    INTEGER (0 .. 4294967295),
            d    INTEGER (
               2
               ..
               179769313486231590772930519078902473361797697894230657273
430081157732675805500963132708477322407536021120113879871393357658
897688144166224928474306394741243777678934248654852763022196012460
411945308295208500576883815068234246288147391311054082723716335051
684586298239947245938479716304835356329624224137214
               )    -- 1 < d < p-1 < 2^1024-1
               OPTIONAL
        }
    GostR3410-94-ValidationAlgorithms  ALGORITHM-IDENTIFIER ::= {
        { GostR3410-94-ValidationParameters IDENTIFIED BY
              id-GostR3410-94-a } |
```

```
        { GostR3410-94-ValidationBisParameters IDENTIFIED BY
               id-GostR3410-94-aBis } |
        { GostR3410-94-ValidationParameters IDENTIFIED BY
               id-GostR3410-94-b } |
        { GostR3410-94-ValidationBisParameters IDENTIFIED BY
               id-GostR3410-94-bBis }
    }
  -- GOST R 34.10-94 Keys Parameters sets
    -- GOST R 34.10-94 Tests parameters set
    -- (GOST R 34.10-94 Annex A. Test vector)
    gostR3410-94-TestParamSetAI
        AlgorithmIdentifier {{
            GostR3410-94-ParamSetAlgorithm
        }} ::=
        {
            algorithm
          id-GostR3410-94-TestParamSet,
            parameters
          GostR3410-94-ParamSetParameters:{
              t       512,
              p       12491554796616397392007291845361681019980789084728846304013646795466302633346425772369277064638881858428879662416202925770315709968465491470753112581700067,
              q       690083979912374782185295287117535788574643562215565368387576361326463015887 81,
              a       830582195677962819385275050881175724488998263282184352149103571317337146852879875383174426740723070452746106232173266903443274617378695814257292977241346 8,
              validationAlgorithm {
                   algorithm
                    id-GostR3410-94-a,
                   parameters
                    GostR3410-94-ValidationParameters: {
                        t       512,
                        x0      24265,
                        c       29505,
                        d       2
                       }
                  }
          }
        }
    -- CryptoPro parameters
    gostR3410-94-CryptoPro-A-ParamSetAI
        AlgorithmIdentifier {{
            GostR3410-94-ParamSetAlgorithm
        }} ::=
        {
            algorithm
```

```
          id-GostR3410-94-CryptoPro-A-ParamSet,
            parameters
          GostR3410-94-ParamSetParameters:{
              t       1024,
              p       127021248288932417465907042777176443525787653508916535812817507265705031260985098497423188333483401180925999995120988934130659205614996724254121049274349357074920312769561451689224110579311248812610229678534638401693520013288995000362260684222750813532307004517341633685004541062586971416883686778842537820383,
              q       68363196144955700784444165611827252895102170888761442055095051287550314083023,
              a       10099790675505530477208181553592522486984108257205345787482351587557714799052927277724415285269929879648335669968284202797289605274717317548059048560713474685214192868091256150280222185647539190902656116367847270145019066794290930185446216399730872221732889830323194097355403213400972588322876850946740663962,
              validationAlgorithm {
                   algorithm
                    id-GostR3410-94-bBis,
                   parameters
                    GostR3410-94-ValidationBisParameters: {
                        t       1024,
                        x0      1376285941,
                        c       3996757427
                       }
               }
          }
        }
    --
    gostR3410-94-CryptoPro-B-ParamSetAI
        AlgorithmIdentifier {{
            GostR3410-94-ParamSetAlgorithm
        }} ::=
        {
            algorithm
          id-GostR3410-94-CryptoPro-B-ParamSet,
            parameters
          GostR3410-94-ParamSetParameters:{
              t       1024,
              p       139454871199115825601409655107690713107041707059928031797758001454375765357722984094124368522288239833039114681648076688236921220737322672160740747771700911134550432053804647694904686120113087816240740184800477047157336662926249423571248823968542221753660143391485680840520336859458494803187341288580489525163,
              q       79885141663410976897627118935756323747307951916507639758300472692338873533959,
              a       42941826148615804143873447737955502392672345968607143066798112994089471231420027060385216699563848719957657284814
```

```
898909770759462613437669456364882730370838934791080835932647976778601915343474400961034231316672578686920482194932878633360203384797092684342247621055760235016132614780652761028509445403338652341,
              validationAlgorithm {
                   algorithm
                    id-GostR3410-94-bBis,
                   parameters
                    GostR3410-94-ValidationBisParameters: {
                        t       1024,
                        x0      1536654555,
                        c       1855361757,
                        d
   1440862938614001456765549029392820565478578022414617829967020177130599747551043947399151406115284791024439062735788342744854120601660303926203867703556828005895720381811489539897659442553756127180085030 6
                      }
                  }
          }
        }
    --
    gostR3410-94-CryptoPro-C-ParamSetAI
        AlgorithmIdentifier {{
            GostR3410-94-ParamSetAlgorithm
        }} ::=
        {
            algorithm
          id-GostR3410-94-CryptoPro-C-ParamSet,
            parameters
          GostR3410-94-ParamSetParameters:{
              t       1024,
              p       11062467923351196304051895241701704024858629548198313837741963962985843959489706089561702242106285255603278638246716655439297654402921844747893079518669992827880792192992701142854655143387580637711044353429355406671265303499627709932071577435422876212836718437037091413501719450458050502917705036345178049380 1,
              q       11346886119981935056486823337887519804326794776488510997961231672532899549103,
              a       81655271797088101601789319141530034822625440513533581624682494676818766212834782128842865458440139551426222087723485023722868022275009502224827866201744494021697716482008353639820229802489262048089869933550806433231352972533220881945689510851551781002210034593705882910730711865530059621499368407371287108323,
              validationAlgorithm {
                   algorithm
                    id-GostR3410-94-bBis,
                   parameters
                    GostR3410-94-ValidationBisParameters: {
```

```
                        t       1024,
                        x0      113275885,
                        c       3037364845,
                        d       9175906676429839327
                       }
                  }
          }
        }
    --
    gostR3410-94-CryptoPro-D-ParamSetAI
        AlgorithmIdentifier {{
            GostR3410-94-ParamSetAlgorithm
        }} ::=
        {
            algorithm
          id-GostR3410-94-CryptoPro-D-ParamSet,
            parameters
          GostR3410-94-ParamSetParameters:{
              t       1024,
              p       90545764962192996590429095877462531561130560 83
907389766971404812524422262512556054474620855996091570786713584955 0
236741915584185990627801066465809510095784713989819413820871596464 8
914493053407920737078890520482730623038837767710173664838239857482 8
787891286471201460474326612697849693665518073864436497893214 9,
              q       10898843579635350691237459149897219262019048 75
576195823347717353905992992115 93,
              a       75697661102170730178212875780161062808552838 03
109571158829574281419208532589041660017017859858216341400371468755 1
412794400562878935266630754392677014598582103365983119173924473251 1
225464712252386803315902707727668715343476086350472025298282727146 1
690125050616858238384366331089777463541013033926723743254833 7,
              validationAlgorithm {
                   algorithm
                    id-GostR3410-94-bBis,
                   parameters
                    GostR3410-94-ValidationBisParameters: {
                        t       1024,
                        x0      333089693,
                        c       2699681355,
                        d
   691588776390130148119174466524027889478644 38
221427558424603662432 52
                       }
                  }
          }
        }
    --
    gostR3410-94-CryptoPro-XchA-ParamSetAI
```

```
        AlgorithmIdentifier {{
            GostR3410-94-ParamSetAlgorithm
        }} ::=
        {
            algorithm
          id-GostR3410-94-CryptoPro-XchA-ParamSet,
            parameters
          GostR3410-94-ParamSetParameters:{
              t       1024,
              p       142011741597563481196368286022318089743276138395243738762872573441927459393512718973631166078467600360848946623567625795282774719212241929071046134208380636394084512691828894000571524625445295769349356752728956831541775441763139384457191755096847107846595662547942312293338483924514339614727760681880609734239,
              q       91771529896554605945588149018382750217296858393520724172743325725474374979801,
              a       13353181327272067343385951994831900121794237596784748689948235959936964252873471246159040332773182141032801252925387191478859899310331056774413619636480306472137782665689868646846327771015080940118260877020161532499046833293129492091277624113787803022435574660628397165937642683267426978088006163152816347588 7,
              validationAlgorithm {
                   algorithm
                    id-GostR3410-94-bBis,
                   parameters
                    GostR3410-94-ValidationBisParameters: {
                        t       1024,
                        x0      3495862036,
                        c       1177570399,
                        d
   354788961024091889513964706477208328196239186534141058228233456746622201867258017799725121699052644608624377641 60334831107459
                    }
                }
          }
        }
    --
    gostR3410-94-CryptoPro-XchB-ParamSetAI
        AlgorithmIdentifier {{
            GostR3410-94-ParamSetAlgorithm
        }} ::=
        {
            algorithm
          id-GostR3410-94-CryptoPro-XchB-ParamSet,
            parameters
          GostR3410-94-ParamSetParameters:{
              t       1024,
```

```
              p       102894612662499485967655207436053031521797049
9893048882484132448474923022758470167998871003604670704877377286176
1712276940986331539089568784129110109512690503345393869871295783467
2572648683417200196629860561193666752429682367397084815179752036423
5957365336895739206176985528459396504253089504608806716026943 3,
              q       9109671391802626916582318050603555673628769498
1825930883887968885281641595199,
              a       8890864727828423151699995801875757891031463338
6525791400519736593048131440685857067369829407947744496306656291505
5036082523994437900272386749145996230867832228661977543992816745254
8232986298598753575466286051738837854736167685769017780335804511440
7733371962538423532919394477873664752824509866617878992443177,
              validationAlgorithm {
                   algorithm
                    id-GostR3410-94-bBis,
                   parameters
                    GostR3410-94-ValidationBisParameters: {
                        t       1024,
                        x0      2046851076,
                        c       3541716983,
                        d
   5733266761098947605661596972889153356605878 7
3174927484418272365769042745461 46
                       }
                  }
          }
        }
    --
    gostR3410-94-CryptoPro-XchC-ParamSetAI
        AlgorithmIdentifier {{
            GostR3410-94-ParamSetAlgorithm
        }} ::=
        {
            algorithm
          id-GostR3410-94-CryptoPro-XchC-ParamSet,
            parameters
          GostR3410-94-ParamSetParameters:{
              t       1024,
              p       1246996366993477513607147265794064436203408861
3950559892172484557299870737698999651480662364723992859320868822848
7511654383509433276647222625940615560580450040947211826027729977563
5402371690630448079715771649447778447000597419032457722226253269698
3744465283535272930439374610657638334915100171593092411549954 9,
              q       6787876137336591234380295020065682527118129468
0501479431146754294748422492761,
              a       4430618464297584182473135030809859326863990650
1189417569952700748609973181426950235239623239110557450826919295792
8789387521018677047181623251027516953100431855964837602657827828194
```

```
24960556189369658653255131371944831362477736534684101187967407098 40
82549699793755607223451067047210860259793099687631930729083 34,
              validationAlgorithm {
                   algorithm
                    id-GostR3410-94-bBis,
                   parameters
                    GostR3410-94-ValidationBisParameters: {
                        t       1024,
                        x0      371898640,
                        c       2482514131,
                        d
   39341170171309491894611690922945474002657559
065001688714824159421346618645269196467699 3
                       }
                  }
          }
        }
END -- GostR3410-94-ParamSetSyntax
```

5.6 GostR3410-2001-PKISyntax

```
-- Copyright(C) CRYPTO-PRO Company
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
```

5.7 GostR3410-2001-ParamSetSyntax

```
-- Copyright(C) CRYPTO-PRO Company
GostR3410-2001-ParamSetSyntax
    { iso(1) member-body(2) ru(643) rans(2) cryptopro(2)
      other(1) modules(1) gostR3410-2001-ParamSetSyntax(12) 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,
        gostR3410-2001-PKISyntax,
        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-GostR3410-2001,
        id-GostR3410-2001-TestParamSet,
        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
        FROM GostR3410-2001-PKISyntax gostR3410-2001-PKISyntax
        AlgorithmIdentifier, ALGORITHM-IDENTIFIER
        FROM Cryptographic-Gost-Useful-Definitions
             cryptographic-Gost-Useful-Definitions
    ;
    GostR3410-2001-ParamSetParameters ::=
        SEQUENCE {
            a       INTEGER (
               1
                  ..
               115792089237316195423570985008687907853269984665640564039
457584007913129639935
               ),   -- 0 < a < p < 2^256
            b       INTEGER (
               1
                  ..
               115792089237316195423570985008687907853269984665640564039
457584007913129639935
               ),   -- 0 < b < p < 2^256
            p       INTEGER (
               289480223093290488558927462521719769633174961664101410098
643960019782824099 85
               ..
               115792089237316195423570985008687907853269984665640564039
```

```
45758400791312963993 5
                ),   -- 2^254 < p < 2^256
            q       INTEGER (
                28948022309329048855892746252171976963317496166410141009 8
643960019782824099 85
                ..
                11579208923731619542357098500868790785326998466564056403 9
45758400791312963993 5
                ),   -- 2^254 < q < 2^256
            x       INTEGER (0
                  ..
                11579208923731619542357098500868790785326998466564056403 9
45758400791312963993 5
                ),   -- 0 < x < p < 2^256
            y       INTEGER (0
                  ..
                11579208923731619542357098500868790785326998466564056403 9
45758400791312963993 5
                )    -- 0 < y < p < 2^256
        }
  -- GOST R 34.10-2001 Public Key Cryptographic Parameters Set:
  -- algorithm & parameters
  -- OID for Parameters Set imported from GostR3410-2001-PKISyntax
    GostR3410-2001-ParamSetAlgorithm ALGORITHM-IDENTIFIER ::= {
        { GostR3410-2001-ParamSetParameters IDENTIFIED BY
                id-GostR3410-2001-TestParamSet } |
        { GostR3410-2001-ParamSetParameters IDENTIFIED BY
                id-GostR3410-2001-CryptoPro-A-ParamSet  } |
        { GostR3410-2001-ParamSetParameters IDENTIFIED BY
                id-GostR3410-2001-CryptoPro-B-ParamSet  } |
        { GostR3410-2001-ParamSetParameters IDENTIFIED BY
                id-GostR3410-2001-CryptoPro-C-ParamSet  } |
        { GostR3410-2001-ParamSetParameters IDENTIFIED BY
                id-GostR3410-2001-CryptoPro-XchA-ParamSet  } |
        { GostR3410-2001-ParamSetParameters IDENTIFIED BY
                id-GostR3410-2001-CryptoPro-XchB-ParamSet  }
    }
    gostR3410-2001-TestParamSet
        AlgorithmIdentifier {{
            GostR3410-2001-ParamSetAlgorithm
        }} ::=
        {
            algorithm
          id-GostR3410-2001-TestParamSet,
            parameters
          GostR3410-2001-ParamSetParameters:{
              a   7,
              b   43308876546767276905765904595650931995942111794451
```

```
039583252968842033849580414,
                        -- 5FBFF498AA938CE739B8E022FBAFEF40563F6E6A
3472FC2A514C0CE9DAE23B7E
              p   57896044618658097711785492504343953926634992332820
282019728792003956564821041,
                  -- 800000000000000000000000000000000000000000000000
00000000000000431
              q   57896044618658097711785492504343953927082934583725
450622380973592137631069619,
                        -- 8000000000000000000000000000000150FE8A18
92976154C59CFC193ACCF5B3
              x   2,
              y   40189740565903750333544942293705977563573938990554
508069097936521343156628O
                  -- 8E2A8A0E65147D4BD6316030E16D19C85C97F0A9CA26712
2B96ABBCEA7E8FC8
          }
        }
    gostR3410-2001-CryptoPro-A-ParamSet
        AlgorithmIdentifier {{
            GostR3410-2001-ParamSetAlgorithm
        }} ::=
        {
            algorithm
          id-GostR3410-2001-CryptoPro-A-ParamSet,
            parameters
          GostR3410-2001-ParamSetParameters:{
              a   11579208923731619542357098500868790785326998466564
056403945758400791312963931 6,
               -- -3 == p - 3
              b   166,
                        -- a6
              p   11579208923731619542357098500868790785326998466564
056403945758400791312963931 9,
                  -- ffffffffffffffffffffffffffffffffffffffffffffffff
ffffffffffffd97
              q   11579208923731619542357098500868790785307376290849
924322537815580507906885032 3,
                        -- ffffffffffffffffffffffffffff6c611070
995ad10045841b09b761b893
              x   1,
              y   64033881142927202683649881450433473985931760268884
941288852745803908878638612
                  -- 8d91e471e0989cda27df505a453f2b7635294f2ddf23e3b
122acc99c9e9f1e14
          }
        }
    gostR3410-2001-CryptoPro-B-ParamSet
```

```
        AlgorithmIdentifier {{
            GostR3410-2001-ParamSetAlgorithm
        }} ::=
        {
            algorithm
          id-GostR3410-2001-CryptoPro-B-ParamSet,
            parameters
          GostR3410-2001-ParamSetParameters:{
              a   57896044618658097711785492504343953926634992332820
282019728792003956564823190,
               -- -3 == p - 3
              b   28091019353058090096996979000309560759124368558014
865957655842872397301267595,
                  -- 3e1af419a269a5f866a7d3c25c3df80ae979259373ff2b1
82f49d4ce7e1bbc8b
              p   57896044618658097711785492504343953926634992332820
282019728792003956564823193,
                  -- 8000000000000000000000000000000000000000000000000
00000000000000c99
              q   57896044618658097711785492504343953927102133160255
826820068844496087732066703,
                        -- 8000000000000000000000000000000015f700cff
f1a624e5e497161bcc8a198f
              x   1,
              y   28792665814854611296992347458380284135028636778229
113005756334730996303888124
                  -- 3fa8124359f96680b83d1c3eb2c070e5c545c9858d03ecf
b744bf8d717717efc
          }
        }
    gostR3410-2001-CryptoPro-C-ParamSet
        AlgorithmIdentifier {{
            GostR3410-2001-ParamSetAlgorithm
        }} ::=
        {
            algorithm
          id-GostR3410-2001-CryptoPro-C-ParamSet,
            parameters
          GostR3410-2001-ParamSetParameters:{
              a   70390085352083305199547718019018437841079516630045
180471284346843705633502616,
               -- -3 == p - 3
              b   32858,
                        -- 805a
              p   70390085352083305199547718019018437841079516630045
180471284346843705633502619,
                  -- 9b9f605f5a858107ab1ec85e6b41c8aacf846e86789051d
37998f7b9022d759b
```

```
              q   703900853520833051995477180190184378409208826471640810353226014583522983966
01,
                        -- 9b9f605f5a858107ab1ec85e6b41c8aa582ca3511eddfb74f02f3a6598980bb9
              x   0,
              y   298188939177312407334712732403147699272405508123836956891464952616045659902
47
                  -- 41ece55743711a8c3cbf3783cd08c0ee4d4dc440d4641a8f366e550dfdb3bb67
          }
        }
    gostR3410-2001-CryptoPro-ExA-ParamSet
        AlgorithmIdentifier {{
            GostR3410-2001-ParamSetAlgorithm
        }} ::=
        {
            algorithm
          id-GostR3410-2001-CryptoPro-XchA-ParamSet,
            parameters
          GostR3410-2001-ParamSetParameters:{
              a   115792089237316195423570985008687907853269984665640560403945758400791312963
9316,
               -- -3 == p - 3
              b   166,
                        -- a6
              p   115792089237316195423570985008687907853269984665640560403945758400791312963
9319,
                  -- ffffffffffffffffffffffffffffffffffffffffffffffffffffffffffffffd97
              q   115792089237316195423570985008687907853073762908499243225378155805079068850323,
                        -- ffffffffffffffffffffffffffff6c611070995ad10045841b09b761b893
              x   1,
              y   640338811429272026836498814504334739859317602688849412888527458039088786386
12
                  -- 8d91e471e0989cda27df505a453f2b7635294f2ddf23e3b122acc99c9e9f1e14
          }
        }
    gostR3410-2001-CryptoPro-ExB-ParamSet
        AlgorithmIdentifier {{
            GostR3410-2001-ParamSetAlgorithm
        }} ::=
        {
            algorithm
          id-GostR3410-2001-CryptoPro-XchB-ParamSet,
            parameters
```

```
          GostR3410-2001-ParamSetParameters:{
              a   70390085352083305199547718019018437841079516630045
180471284346843705633502616,
               -- -3 == p - 3
              b   32858,
                  -- 805a
              p   70390085352083305199547718019018437841079516630045
180471284346843705633502619,
                  -- 9b9f605f5a858107ab1ec85e6b41c8aacf846e86789051d
37998f7b9022d759b
              q   70390085352083305199547718019018437840920882647164
081035322601458352298396601,
                       -- 9b9f605f5a858107ab1ec85e6b41c8aa582ca351
1eddfb74f02f3a6598980bb9
              x   0,
              y   29818893917731240733471273240314769927240550812383
695689146495261604565990247
                  -- 41ece55743711a8c3cbf3783cd08c0ee4d4dc440d4641a8
f366e550dfdb3bb67
            }
        }
END -- GostR3410-2001-ParamSetSyntax
```

6 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);

[CPALGS] Cryptographic Algorithm "CryptoPro CSP"

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

[RFC 3280] 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.

[RFC 3279] 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.

[RFC 2219] Bradner, S., "Key Words for Use in RFCs to Indi- cateRequirement 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 Dis- tinguished 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 acknowledge 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