S/MIME Working Group
INTERNET-DRAFT
Expires December 15, 2003
Intended Category: Informational

Serguei Leontiev, CRYPTO-PRO
Vladimir Popov, CRYPTO-PRO
June 15, 2003

Cryptographic Message Syntax (CMS) algorithms for GOST 28147-89, GOST R 34.10-94, GOST R 34.10-2001, GOST R 34.11-94.

<draft-leontiev-cryptopro-cpcms-00.txt>

Status of this Memo

Abstract

This document describes the conventions for using cryptographic algorithms GOST 28147-89, GOST R 34.10-94, GOST R 34.10-2001, GOST R 34.11-94, along with Cryptographic Message Syntax (CMS). The CMS is used for digital signature, digest, authentication and encryption arbitrary message contents.

Table of Contents

1 Introduction 2
1.2 Terminology. 3
2 Message Digest Algorithms. 3

2.1 Message Digest Algorithm GOST R 34.11-94 3
3 Signature Algorithms 4
3.1 Signature Algorithm GOST R 34.10-94. 4
3.2 Signature Algorithm GOST R 34.10-2001. 4
4 Key Management Algorithms. 5
4.1 Key Agreement Algorithms 5
4.1.1 Key Agreement Algorithm Based on GOST R 34.10-94 Public Keys. 5
4.1.1 Key Agreement Algorithm Based on GOST R 34.10-2001 Public Keys 6
4.2 Key Transport Algorithms. 7
4.2.1 Key Transport Algorithm Based on GOST R 34.10-94 Public Keys 7
4.2.2 Key Transport Algorithm Based on GOST R 34.10-2001 Public Keys 8
5 Content Encryption Algorithms. 8
5.1 Key-Encryption Key Algorithm GOST 28147-89 9
6 MAC Algorithms 11
6.1 HMAC with GOST R 34.11-94. 11
7 Using with S/MIME. 11
7.1 Parameter micalg 11
7.2 Atribute SMIMECapabilities 11
8 Security Considerations. 11
9 Appendix ASN.1 Modules 12
9.1 Gost28147-89-EncryptionSyntax. 12
9.2 Gost28147-89-ParamSetSyntax. 14
9.3 GostR3410-94-EncryptionSyntax. 21
9.4 GostR3410-94-SignatureSyntax 23
9.5 GostR3410-2001-EncryptionSyntax. 24
9.6 GostR3410-2001-SignatureSyntax 26
10 References 27
11 Acknowledgments. 29
Author's Address. 29
Full Copyright Statement. 30

1 Introduction

The Cryptographic Message Syntax (CMS) [CMS] is used for digital signature, digest, authentication and encryption arbitrary message contents. This companion specification describes the usage of cryptographic algorithms GOST 28147-89, GOST R 34.10-94, GOST R 34.10-2001 and hash algorithm GOST R 34.11-94 in CMS, proposed by CRYPTO-PRO Company for "Russian Cryptographic Software Compatibility

Agreement" community. This document does not describe those cryptographic algorithms; they are defined in corresponding national standards.

The CMS values are generated using ASN.1 [X.208-88], using BER-

encoding [X.209-88]. Algorithm identifiers (which include ASN.1 object identifiers) identify cryptographic algorithms, and some algorithms require additional parameters. When needed, parameters are specified with an ASN.1 structure. The algorithm identifier for each algorithm is specified, and when needed, the parameter structure is specified. The fields in the CMS employed by each algorithm are identified.

1.2 Terminology

In this document, the key words MUST, MUST NOT, REQUIRED, SHOULD, SHOULD NOT, RECOMMENDED, and MAY are to be interpreted as described in [RFC 2119].

2 Message Digest Algorithms

This section specifies the conventions for using digest algorithm GOST R 34.11-94 employed by CMS.

Digest values are located in the DigestedData digest field and the Message Digest authenticated attribute. In addition, digest values are input to signature algorithms.

2.1 Message Digest Algorithm GOST R 34.11-94

Hash function 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 object identifiers (OID) in software code.

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

id-CryptoPro-algorithms OBJECT IDENTIFIER ::=
       { id-CryptoPro }
```

The hash algorithm GOST R 34.11-94 has the following identifier:


```
id-GostR3411-94 OBJECT IDENTIFIER ::=
       { id-CryptoPro-algorithms gostr3411(9) }
```

The following structure contains digest in little-endian representation:

```
GostR3411-94-Digest ::= OCTET STRING (SIZE (32))
```

3 Signature Algorithms

This section specifies the CMS procedures for GOST R 34.10-94 and GOST R 34.10-2001 signature algorithms.

Signature algorithm identifiers are located in the SignerInfo signatureAlgorithm field of SignedData. Also, signature algorithm identifiers are located in the SignerInfo signatureAlgorithm field of countersignature attributes.

Signature values are located in the SignerInfo signature field of SignedData. Also, signature values are located in the SignerInfo signature field of countersignature attributes.

3.1 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 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.

For a signature algorithm identifier, public key OID is used:

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

Signature algorithm GOST R 34.10-94 generates digital signature in the form of a binary 512-bit vector (<r'>256||<s>256). signatureValue contains its little endian representation.

```
GostR3410-94-Signature ::= OCTET STRING
```

3.2 Signature Algorithm GOST R 34.10-2001

GOST R 34.10-2001 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 GOST R 34.11-94. This document does not contain GOST R 34.10-2001 standard description, which is fully described in [GOSTR34102001].

For a signature algorithm identifier, public key OID is used:

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

Signature algorithm GOST R 34.10-2001 generates digital signature in the form of a binary 512-bit vector (<r'>256||<s>256). signatureValue contains its little endian representation.

```
GostR3410-2001-Signature ::= OCTET STRING
```

4 Key Management Algorithms

This chapter describes the key agreement and key transport algorithms, always supposing that key enciphering usage is GOST 28147-89 algorithm only.

4.1 Key Agreement Algorithms

This part describes the key agreement algorithms based on both GOST R 34.10-94 and GOST R 34.10-2001 public keys.

Key agreement algorithm identifiers are located in the EnvelopedData RecipientInfos KeyAgreeRecipientInfo keyEncryptionAlgorithm and AuthenticatedData RecipientInfos KeyAgreeRecipientInfo keyEncryptionAlgorithm fields.

Key wrap algorithm identifiers are located in the KeyWrapAlgorithm parameters within the EnvelopedData RecipientInfos KeyAgreeRecipientInfo keyEncryptionAlgorithm and AuthenticatedData RecipientInfos KeyAgreeRecipientInfo keyEncryptionAlgorithm fields.

Wrapped content-encryption keys are located in the EnvelopedData RecipientInfos KeyAgreeRecipientInfo RecipientEncryptedKeys encryptedKey field. Wrapped message-authentication keys are located in the AuthenticatedData RecipientInfos KeyAgreeRecipientInfo RecipientEncryptedKeys encryptedKey field.

4.1.1 Key Agreement Algorithm Based on GOST R 34.10-94 Public Keys

The key agreement algorithm based on GOST R 34.10-94 public keys is described in [CPALGS]. When using this algorithm, the EnvelopedData RecipientInfos KeyAgreeRecipientInfo field is used as follows:

version MUST be 3.

originator MUST be the originatorKey alternative. The originatorKey algorithm field MUST contain the object identifier id-GostR3410-94 with necessary parameters (see [CPALGS]). The originatorKey publicKey field MUST contain the sender's public key.

keyEncryptionAlgorithm MUST be the id-GostR3410-94 algorithm identifier. It's parameters encapsulate GostR3410-94-TransportParameters, containing GOST 28147-89 algorithm parameters used for key encryption, and UKM. ephemeralPublicKey MUST NOT be present.

GostR3410-94-TransportParameters ::=

```
   SEQUENCE {
       encryptionParamSet   OBJECT IDENTIFIER,
       ephemeralPublicKey   SubjectPublicKeyInfo OPTIONAL,
       ukm                  OCTET STRING
   }
```

encryptedKey encapsulates Gost28147-89-EncryptedKey, which contains encrypted session key and it's MAC.

```
Gost28147-89-EncryptedKey ::=   SEQUENCE {
       encryptedKey         Gost28147-89-Key,
       macKey               Gost28147-89-MAC
   }
```

4.1.2 Key Agreement Algorithm Based on GOST R 34.10-2001 Public Keys

The key agreement algorithm based on GOST R 34.10-2001 public keys is described in [CPALGS]. When using this algorithm, the EnvelopedData RecipientInfos KeyAgreeRecipientInfo field is used as follows:

Version MUST be 3.

originator MUST be the originatorKey alternative. The originatorKey algorithm field MUST contain the object identifier id-GostR3410-2001 with necessary parameters (see [CPALGS]). The originatorKey publicKey field MUST contain the sender's public key.

keyEncryptionAlgorithm MUST be the id-GostR3410-2001 algorithm identifier. It's parameters encapsulate GostR3410-2001-TransportParameters, containing GOST 28147-89 algorithm parameters used for key encryption, and UKM. ephemeralPublicKey MUST NOT be present.


```
GostR3410-2001-TransportParameters ::=
   SEQUENCE {
       encryptionParamSet   OBJECT IDENTIFIER,
       ephemeralPublicKey   SubjectPublicKeyInfo OPTIONAL,
       ukm                  OCTET STRING
   }
```

encryptedKey encapsulates Gost28147-89-EncryptedKey, which

contains encrypted session key and it's MAC.

```
Gost28147-89-EncryptedKey ::=   SEQUENCE {
       encryptedKey         Gost28147-89-Key,
       macKey               Gost28147-89-MAC
   }
```

4.2 Key Transport Algorithms

This part describes the key transport algorithms based on both GOST R 34.10-94 and GOST R 34.10-2001 public keys.

Key transport algorithm identifiers are located in the EnvelopedData RecipientInfos KeyTransRecipientInfo keyEncryptionAlgorithm field.

Key transport encrypted content-encryption keys are located in the EnvelopedData RecipientInfos KeyTransRecipientInfo encryptedKey field.

4.2.1 Key Transport Algorithm Based on GOST R 34.10-94 Public Keys

The key transport algorithm based on GOST R 34.10-94 public keys is described in [CPALGS]. When using this algorithm, the EnvelopedData RecipientInfos KeyTransRecipientInfo field is used as follows:

version MUST be 0 or 3.

keyEncryptionAlgorithm MUST be identical to the recipient public key algorithm identifier.

encryptedKey encapsulates GostR3410-94-KeyTransportEncryptedKeyOctetString, which contains encrypted session key, it's MAC, GOST 28147-89 algorithm parameters used for key encryption, sender's ephemeral public key, and UKM.

ephemeralPublicKey MUST be present, and its parameters, if present, MUST be equal to the recipient public key parameters;

```
GostR3410-94-KeyTransportEncryptedKeyOctetString ::= SEQUENCE {
```



```
       sessionEncryptedKey   Gost28147-89-EncryptedKey,
```

```
            transportParameters   GostR3410-94-TransportParameters
        }

    GostR3410-94-TransportParameters ::=
        SEQUENCE {
            encryptionParamSet   OBJECT IDENTIFIER,
            ephemeralPublicKey   SubjectPublicKeyInfo OPTIONAL,
            ukm                  OCTET STRING
        }
```

4.2.2 Key Transport Algorithm Based on GOST R 34.10-2001 Public Keys

The key transport algorithm based on GOST R 34.10-2001 public keys is described in [CPALGS]. When using this algorithm, the EnvelopedData RecipientInfos KeyTransRecipientInfo field is used as follows:

version MUST be 0 or 3.

keyEncryptionAlgorithm MUST be identical to the recipient public key algorithm identifier.

encryptedKey encapsulates GostR3410-2001-KeyTransportEncryptedKeyOctetString, which contains encrypted session key, it's MAC, GOST 28147-89 algorithm parameters used for key encryption, sender's ephemeral public key, and UKM.

ephemeralPublicKey MUST be present, and it's parameters, if present, MUST be equal to the recipient public key parameters;

```
    GostR3410-2001-KeyTransportEncryptedKeyOctetString ::= SEQUENCE {
            sessionEncryptedKey   Gost28147-89-EncryptedKey,
            transportParameters   GostR3410-2001-TransportParameters
        }

    GostR3410-2001-TransportParameters ::=
        SEQUENCE {
            encryptionParamSet   OBJECT IDENTIFIER,
            ephemeralPublicKey   SubjectPublicKeyInfo OPTIONAL,
            ukm                  OCTET STRING
        }
```

5 Content Encryption Algorithms

This section specifies the conventions employed by CMS implementations that support content encryption using GOST 28147-89.

Content encryption algorithm identifiers are located in the EnvelopedData EncryptedContentInfo contentEncryptionAlgorithm and the EncryptedData EncryptedContentInfo contentEncryptionAlgorithm fields.

Content encryption algorithms are used to encipher the content located in the EnvelopedData EncryptedContentInfo encryptedContent field and the EncryptedData EncryptedContentInfo encryptedContent field.

5.1 Content Encryption Algorithm GOST 28147-89

This section specifies the use of GOST 28147-89 algorithm for data encipherment.

GOST 28147-89 is fully described in [GOST28147] (in Russian).

This document specifies the following OID for this algorithm:

```
id-Gost28147-89 OBJECT IDENTIFIER ::=
       { id-CryptoPro-algorithms gost28147-89(21) }
```

Algorithm parameters MUST be present and have the following structure:

```
    Gost28147-89-Parameters ::=      SEQUENCE {
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           ),          iv
Gost28147-89-IV      }
```

encryptionParamSet specify the set of corresponding Gost28147-89-ParamSetParameters.

```
Gost28147-89-ParamSetParameters ::= SEQUENCE {
   eUZ   Gost28147-89-UZ,
   mode  INTEGER {
            gost28147-89-OFB(0),
            gost28147-89-CFB(1),
            cryptoPro-CBC(2)
```

```
        },
    shiftBits   INTEGER { gost28147-89-block(64) },
```



```
    keyWrap  AlgorithmIdentifier {{
                Gost28147-89-KeyWrapAlgorithms
             }},
    keyMix   AlgorithmIdentifier {{
             Gost28147-89-KeyMixAlgorithms
             }} }
```

where
- iv - initializsation vector;
- eUZ - S-box;
- mode - cipher mode;
- shiftBits - cipher parameter;
- keyWrap - key export algorithm identifier;
- keyMix - key meshing algorithm.

The following values for encryptionParamSet are already defined:

```
    id-Gost28147-89-TestParamSet OBJECT IDENTIFIER ::=      { id-
CryptoPro-encrypts test(0) }

    id-Gost28147-89-CryptoPro-A-ParamSet OBJECT IDENTIFIER ::=      {
id-CryptoPro-encrypts cryptopro-A(1) }

    id-Gost28147-89-CryptoPro-B-ParamSet OBJECT IDENTIFIER ::=      {
id-CryptoPro-encrypts cryptopro-B(2) }

    id-Gost28147-89-CryptoPro-C-ParamSet OBJECT IDENTIFIER ::=      {
id-CryptoPro-encrypts cryptopro-C(3) }

    id-Gost28147-89-CryptoPro-D-ParamSet OBJECT IDENTIFIER ::=      {
id-CryptoPro-encrypts cryptopro-D(4) }

    id-Gost28147-89-CryptoPro-Simple-A-ParamSet      OBJECT
IDENTIFIER ::=         { id-CryptoPro-encrypts cryptopro-Simple-A(6)
}

    id-Gost28147-89-CryptoPro-Simple-B-ParamSet      OBJECT
IDENTIFIER ::=         { id-CryptoPro-encrypts cryptopro-Simple-B(7)
}
```

```
    id-Gost28147-89-CryptoPro-Simple-C-ParamSet      OBJECT
IDENTIFIER ::=         { id-CryptoPro-encrypts cryptopro-Simple-C(8)
}

    id-Gost28147-89-CryptoPro-Simple-D-ParamSet      OBJECT
IDENTIFIER ::=         { id-CryptoPro-encrypts cryptopro-Simple-D(9)
}
```


6 MAC Algorithms

This section specifies the conventions employed by CMS implementations that support the message authentication code (MAC) based on GOST R 34.11-94 HMAC. This MAC can also be used as a pseudo-random function with 256 bits (32 bytes) internal state size, which can be used to derive keys.

MAC algorithm identifiers are located in the AuthenticatedData macAlgorithm field.

MAC values are located in the AuthenticatedData mac field

6.1 HMAC with GOST R 34.11-94

GOSTR3411_HMAC (K,text) function is based on hash function GOST R 34.11-94, as defined in [HMAC], with the following parameter values: B = 32, L = 32.

OID for GOSTR3411_HMAC, defined by this document:

```
id-HMACGostR3411-94 OBJECT IDENTIFIER ::=
       { id-CryptoPro-algorithms hmacgostr3411(10) }
```

This algorithm has the same parameters, as GOST R 34.11-94 digest algorithm, and uses the same OIDs for their identification (see [CPPK]).

7 Using with S/MIME

This section defines use of the algorithms defined in this document together with S/MIME [RFC 2633].

7.1 Parameter micalg

When using the algorithms defined in this document, micalg parameter should be set to 'unknown', according to [RFCC 2633].

7.2 Attribute SMIMECapabilities

S/MIME message, which uses the algorithms defined in this document, should contain the list of algorithm identifiers for digest and encryption algorithms, defined in this document, with their parameters, in it's SMIMECapabilities attribute.

8 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 Center "ATLAS" and by Center 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 approved methods of cryptographic analysis.

9 Appendix ASN.1 Modules

9.1 Gost28147-89-EncryptionSyntax

```
-- Copyright(C) CRYPTO-PRO Company
Gost28147-89-EncryptionSyntax
    { iso(1) member-body(2) ru(643) rans(2) cryptopro(2)
      other(1) modules(1) gost28147-89-EncryptionSyntax(4) 1 }
DEFINITIONS EXPLICIT TAGS ::=
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-encrypts,
        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 }
        AlgorithmIdentifier, ALGORITHM-IDENTIFIER
        FROM Cryptographic-Gost-Useful-Definitions
             cryptographic-Gost-Useful-Definitions
    ;
  -- GOST 28147-89 OID
    id-Gost28147-89 OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms gost28147-89(21) }
```



```
  -- GOST 28147-89 Cryptographic Parameter Sets OIDs
    id-Gost28147-89-TestParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-encrypts test(0) }
    id-Gost28147-89-CryptoPro-A-ParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-encrypts cryptopro-A(1) }
    id-Gost28147-89-CryptoPro-B-ParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-encrypts cryptopro-B(2) }
    id-Gost28147-89-CryptoPro-C-ParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-encrypts cryptopro-C(3) }
    id-Gost28147-89-CryptoPro-D-ParamSet OBJECT IDENTIFIER ::=
        { id-CryptoPro-encrypts cryptopro-D(4) }
    id-Gost28147-89-CryptoPro-Simple-A-ParamSet
        OBJECT IDENTIFIER ::=
           { id-CryptoPro-encrypts cryptopro-Simple-A(6) }
    id-Gost28147-89-CryptoPro-Simple-B-ParamSet
        OBJECT IDENTIFIER ::=
           { id-CryptoPro-encrypts cryptopro-Simple-B(7) }
    id-Gost28147-89-CryptoPro-Simple-C-ParamSet
```

```
        OBJECT IDENTIFIER ::=
           { id-CryptoPro-encrypts cryptopro-Simple-C(8) }
    id-Gost28147-89-CryptoPro-Simple-D-ParamSet
        OBJECT IDENTIFIER ::=
           { id-CryptoPro-encrypts cryptopro-Simple-D(9) }
  -- GOST 28147-89 Types
    Gost28147-89-Data ::= OCTET STRING (SIZE (0..4294967294))
    Gost28147-89-EncryptedData ::=
        OCTET STRING (SIZE (0..4294967294))
    Gost28147-89-UZ ::= OCTET STRING (SIZE (64))
    Gost28147-89-IV ::= OCTET STRING (SIZE (8))
    Gost28147-89-Key ::= OCTET STRING (SIZE (32))
    Gost28147-89-MAC ::= OCTET STRING (SIZE (1..4))
    Gost28147-89-EncryptedKey ::=
        SEQUENCE {
            encryptedKey         Gost28147-89-Key,
            macKey               Gost28147-89-MAC (SIZE (4))
        }
  -- GOST 28147-89 encryption algorithm parameters
    Gost28147-89-Parameters ::=
        SEQUENCE {
            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
          ),
            iv                   Gost28147-89-IV
        }
    Gost28147-89-Algorithms ALGORITHM-IDENTIFIER ::= {
        { Gost28147-89-Parameters IDENTIFIED BY
               id-Gost28147-89 }
    }
END -- Gost28147-89-EncryptionSyntax
```

9.2 Gost28147-89-ParamSetSyntax

```
-- Copyright(C) CRYPTO-PRO Company
Gost28147-89-ParamSetSyntax
    { iso(1) member-body(2) ru(643) rans(2) cryptopro(2)
      other(1) modules(1) gost28147-89-ParamSetSyntax(6) 1 }
DEFINITIONS EXPLICIT TAGS ::=
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-encrypts,
        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-UZ,
        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
```



```
        AlgorithmIdentifier, ALGORITHM-IDENTIFIER
        FROM Cryptographic-Gost-Useful-Definitions
             cryptographic-Gost-Useful-Definitions
    ;
```

```
  -- GOST 28147-89 Cryptographic Parameters Set:
  -- algorithm & parameters
  -- OID for Parameters Set imported from
  -- Gost28147-89-EncryptionSyntax
    Gost28147-89-ParamSetParameters ::=
            SEQUENCE {
            eUZ           Gost28147-89-UZ,
            mode          INTEGER {
                   gost28147-89-OFB(0),
                   gost28147-89-CFB(1),
                        cryptoPro-CBC(2)
               },
            shiftBits     INTEGER { gost28147-89-block(64) },
            keyWrap       AlgorithmIdentifier {{
                   Gost28147-89-KeyWrapAlgorithms
               }},
            keyMix        AlgorithmIdentifier {{
                   Gost28147-89-KeyMixAlgorithms
               }}
        }
    Gost28147-89-ParamSetAlgorithms ALGORITHM-IDENTIFIER ::= {
        { Gost28147-89-ParamSetParameters IDENTIFIED BY
               id-Gost28147-89-TestParamSet  } |
        { Gost28147-89-ParamSetParameters IDENTIFIED BY
               id-Gost28147-89-CryptoPro-A-ParamSet  } |
        { Gost28147-89-ParamSetParameters IDENTIFIED BY
               id-Gost28147-89-CryptoPro-B-ParamSet  } |
        { Gost28147-89-ParamSetParameters IDENTIFIED BY
               id-Gost28147-89-CryptoPro-C-ParamSet  } |
        { Gost28147-89-ParamSetParameters IDENTIFIED BY
               id-Gost28147-89-CryptoPro-D-ParamSet  } |
        { Gost28147-89-ParamSetParameters IDENTIFIED BY
               id-Gost28147-89-CryptoPro-Simple-A-ParamSet  } |
        { Gost28147-89-ParamSetParameters IDENTIFIED BY
               id-Gost28147-89-CryptoPro-Simple-B-ParamSet  } |
        { Gost28147-89-ParamSetParameters IDENTIFIED BY
               id-Gost28147-89-CryptoPro-Simple-C-ParamSet  } |
        { Gost28147-89-ParamSetParameters IDENTIFIED BY
               id-Gost28147-89-CryptoPro-Simple-D-ParamSet  }
    }
    id-Gost28147-89-CryptoPro-KeyWrap OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms keyWrap(13) cryptoPro(1) }
    id-Gost28147-89-None-KeyWrap OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms keyWrap(13) none(0) }
```

```
    Gost28147-89-KeyWrapAlgorithms  ALGORITHM-IDENTIFIER ::= {
        { NULL IDENTIFIED BY id-Gost28147-89-CryptoPro-KeyWrap } |
        { NULL IDENTIFIED BY id-Gost28147-89-None-KeyWrap }
    }
    id-Gost28147-89-CryptoPro-KeyMix  OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms keyMix(14) cryptoPro(1) }
    id-Gost28147-89-None-KeyMix OBJECT IDENTIFIER ::=
        { id-CryptoPro-algorithms keyMix(14) none(0) }
    Gost28147-89-KeyMixAlgorithms  ALGORITHM-IDENTIFIER ::= {
        { NULL IDENTIFIED BY id-Gost28147-89-CryptoPro-KeyMix } |
        { NULL IDENTIFIED BY id-Gost28147-89-None-KeyMix }
    }
  -- GOST 28147-89 Cryptographic Parameters Set: values
    -- Test Parameters Set
    gost28147-89-TestParamSetAI
        AlgorithmIdentifier {{
            Gost28147-89-ParamSetAlgorithms
        }} ::=
        {
            algorithm
          id-Gost28147-89-TestParamSet,
            parameters
          Gost28147-89-ParamSetParameters:{
              eUZ   '4CDE389C2989EFB6FFEB56C55EC29B029875613B113F896
003970C798AA1D55DE210AD43375DB38EB42C77E7CD46CAFAD66A201F70F41EA4AB
03F22165B844D8'H,
                   mode  gost28147-89-OFB,
              shiftBits  64,
                   keyWrap
               { algorithm id-Gost28147-89-None-KeyWrap },
              keyMix
               { algorithm id-Gost28147-89-None-KeyMix }
          }
        }
    -- CryptoPro Parameters Sets
    gost28147-89-UZ-CryptoPro-A Gost28147-89-UZ ::=
      -- K1 K2 K3 K4 K5 K6 K7 K8
      -- 9  3  E  E  B  3  1  B
      -- 6  7  4  7  5  A  D  A
      -- 3  E  6  A  1  D  2  F
      -- 2  9  2  C  9  C  9  5
      -- 8  8  B  D  8  1  7  0
      -- B  A  3  1  D  2  A  C
      -- 1  F  D  3  F  0  6  E
      -- 7  0  8  9  0  B  0  8
      -- A  5  C  0  E  7  8  6
      -- 4  2  F  2  4  5  C  2
```

```
      -- E  6  5  B  2  9  4  3
```



```
      -- F  C  A  4  3  4  5  9
      -- C  B  0  F  C  8  F  1
      -- 0  4  7  8  7  F  3  7
      -- D  D  1  5  A  E  B  D
      -- 5  1  9  6  6  6  E  4
        '93EEB31B67475ADA3E6A1D2F292C9C9588BD8170BA31D2AC1FD3F06E70
890B08A5C0E78642F245C2E65B2943FCA43459CB0FC8F104787F37DD15AEBD51966
6E4'H
    gost28147-89-CryptoPro-A-ParamSetAI
        AlgorithmIdentifier {{
            Gost28147-89-ParamSetAlgorithms
        }} ::=
        {
            algorithm
          id-Gost28147-89-CryptoPro-A-ParamSet,
            parameters
          Gost28147-89-ParamSetParameters:{
              eUZ   gost28147-89-UZ-CryptoPro-A,
                   mode  gost28147-89-CFB,
                   shiftBits  64,
                   keyWrap
                { algorithm id-Gost28147-89-CryptoPro-KeyWrap },
              keyMix
                { algorithm id-Gost28147-89-CryptoPro-KeyMix }
          }
        }
    --
    gost28147-89-UZ-CryptoPro-B Gost28147-89-UZ ::=
      -- K1 K2 K3 K4 K5 K6 K7 K8
      -- 8  0  E  7  2  8  5  0
      -- 4  1  C  5  7  3  2  4
      -- B  2  0  0  C  2  A  B
      -- 1  A  A  D  F  6  B  E
      -- 3  4  9  B  9  4  9  8
      -- 5  D  2  6  5  D  1  3
      -- 0  5  D  1  A  E  C  7
      -- 9  C  B  2  B  B  3  1
      -- 2  9  7  3  1  C  7  A
      -- E  7  5  A  4  1  4  2
      -- A  3  8  C  0  7  D  9
```

```
      -- C  F  F  F  D  F  0  6
      -- D  B  3  4  6  A  6  F
      -- 6  8  6  E  8  0  F  D
      -- 7  6  1  9  E  9  8  5
      -- F  E  4  8  3  5  E  C
        '80E7285041C57324B200C2AB1AADF6BE349B94985D265D1305D1AEC79C
B2BB3129731C7AE75A4142A38C07D9CFFFDF06DB346A6F686E80FD7619E985FE483
5EC'H
```



```
    gost28147-89-CryptoPro-B-ParamSetAI
        AlgorithmIdentifier {{
            Gost28147-89-ParamSetAlgorithms
        }} ::=
        {
            algorithm
          id-Gost28147-89-CryptoPro-B-ParamSet,
            parameters
          Gost28147-89-ParamSetParameters:{
              eUZ   gost28147-89-UZ-CryptoPro-B,
                   mode  gost28147-89-CFB,
                   shiftBits  64,
                   keyWrap
                { algorithm id-Gost28147-89-CryptoPro-KeyWrap },
              keyMix
                { algorithm id-Gost28147-89-CryptoPro-KeyMix }
          }
        }
    --
    gost28147-89-UZ-CryptoPro-C Gost28147-89-UZ ::=
      -- K1 K2 K3 K4 K5 K6 K7 K8
      -- 1  0  8  3  8  C  A  7
      -- B  1  2  6  D  9  9  4
      -- C  7  5  0  B  B  6  0
      -- 2  D  0  1  0  1  8  5
      -- 9  B  4  5  4  8  D  A
      -- D  4  9  D  5  E  E  2
      -- 0  5  F  A  1  2  2  F
      -- F  2  A  8  2  4  0  E
      -- 4  8  3  B  9  7  F  C
      -- 5  E  7  2  3  3  3  6
      -- 8  F  C  9  C  6  5  1
      -- E  C  D  7  E  5  B  B
```

```
      -- A  9  6  E  6  A  4  D
      -- 7  A  E  F  F  0  1  9
      -- 6  6  1  C  A  F  C  3
      -- 3  3  B  4  7  D  7  8
        '10838CA7B126D994C750BB602D0101859B4548DAD49D5EE205FA122FF2
A8240E483B97FC5E7233368FC9C651ECD7E5BBA96E6A4D7AEFF019661CAFC333B47
D78'H
    gost28147-89-CryptoPro-C-ParamSetAI
        AlgorithmIdentifier {{
            Gost28147-89-ParamSetAlgorithms
        }} ::=
        {
            algorithm
          id-Gost28147-89-CryptoPro-C-ParamSet,
            parameters
```



```
          Gost28147-89-ParamSetParameters:{
              eUZ   gost28147-89-UZ-CryptoPro-C,
                    mode  gost28147-89-CFB,
                    shiftBits  64,
                    keyWrap
                     { algorithm id-Gost28147-89-CryptoPro-KeyWrap },
                   keyMix
                    { algorithm id-Gost28147-89-CryptoPro-KeyMix }
          }
        }
    --
    gost28147-89-UZ-CryptoPro-D Gost28147-89-UZ ::=
      -- K1 K2 K3 K4 K5 K6 K7 K8
      -- F  B  1  1  0  8  3  1
      -- C  6  C  5  C  0  0  A
      -- 2  3  B  E  8  F  6  6
      -- A  4  0  C  9  3  F  8
      -- 6  C  F  A  D  2  1  F
      -- 4  F  E  7  2  5  E  B
      -- 5  E  6  0  A  E  9  0
      -- 0  2  5  D  B  B  2  4
      -- 7  7  A  6  7  1  D  C
      -- 9  D  D  2  3  A  8  3
      -- E  8  4  B  6  4  C  5
      -- D  0  8  4  5  7  4  9
      -- 1  5  9  9  4  C  B  7
```

```
      -- B  A  3  3  E  9  A  D
      -- 8  9  7  F  F  D  5  2
      -- 3  1  2  8  1  6  7  E'H
        'FB110831C6C5C00A23BE8F66A40C93F86CFAD21F4FE725EB5E60AE9002
5DBB2477A671DC9DD23A83E84B64C5D084574915994CB7BA33E9AD897FFD5231281
67E'H
    gost28147-89-CryptoPro-D-ParamSetAI
        AlgorithmIdentifier {{
            Gost28147-89-ParamSetAlgorithms
        }} ::=
        {
            algorithm
          id-Gost28147-89-CryptoPro-D-ParamSet,
            parameters
          Gost28147-89-ParamSetParameters:{
              eUZ   gost28147-89-UZ-CryptoPro-D,
                   mode  gost28147-89-CFB,
                   shiftBits  64,
                   keyWrap
                    { algorithm id-Gost28147-89-CryptoPro-KeyWrap },
                  keyMix
                    { algorithm id-Gost28147-89-CryptoPro-KeyMix }
```



```
          }
        }
    --
    gost28147-89-CryptoPro-Simple-A-ParamSetAI
        AlgorithmIdentifier {{
            Gost28147-89-ParamSetAlgorithms
        }} ::=
        {
            algorithm
          id-Gost28147-89-CryptoPro-Simple-A-ParamSet,
            parameters
          Gost28147-89-ParamSetParameters:{
              eUZ   gost28147-89-UZ-CryptoPro-A,
                   mode  gost28147-89-CFB,
                   shiftBits  64,
                   keyWrap
                    { algorithm id-Gost28147-89-None-KeyWrap },
                  keyMix
                    { algorithm id-Gost28147-89-CryptoPro-KeyMix }
```

```
      }
    }
--
gost28147-89-CryptoPro-Simple-B-ParamSetAI
    AlgorithmIdentifier {{
        Gost28147-89-ParamSetAlgorithms
    }} ::=
    {
        algorithm
      id-Gost28147-89-CryptoPro-Simple-B-ParamSet,
        parameters
      Gost28147-89-ParamSetParameters:{
          eUZ   gost28147-89-UZ-CryptoPro-B,
               mode  gost28147-89-CFB,
               shiftBits  64,
               keyWrap
            { algorithm id-Gost28147-89-None-KeyWrap },
          keyMix
            { algorithm id-Gost28147-89-CryptoPro-KeyMix }
      }
    }
--
gost28147-89-CryptoPro-Simple-C-ParamSetAI
    AlgorithmIdentifier {{
        Gost28147-89-ParamSetAlgorithms
    }} ::=
    {
        algorithm
      id-Gost28147-89-CryptoPro-Simple-C-ParamSet,
```



```
        parameters
      Gost28147-89-ParamSetParameters:{
          eUZ   gost28147-89-UZ-CryptoPro-C,
               mode  gost28147-89-CFB,
               shiftBits  64,
               keyWrap
            { algorithm id-Gost28147-89-None-KeyWrap },
          keyMix
            { algorithm id-Gost28147-89-CryptoPro-KeyMix }
      }
    }
--
```

```
    gost28147-89-CryptoPro-Simple-D-ParamSetAI
        AlgorithmIdentifier {{
            Gost28147-89-ParamSetAlgorithms
        }} ::=
        {
            algorithm
          id-Gost28147-89-CryptoPro-Simple-D-ParamSet,
            parameters
          Gost28147-89-ParamSetParameters:{
              eUZ   gost28147-89-UZ-CryptoPro-D,
                   mode  gost28147-89-CFB,
                   shiftBits  64,
                   keyWrap
                    { algorithm id-Gost28147-89-None-KeyWrap },
                  keyMix
                   { algorithm id-Gost28147-89-CryptoPro-KeyMix }
          }
        }
END -- Gost28147-89-ParamSetSyntax
```

9.3 GostR3410-94-EncryptionSyntax

```
-- Copyright(C) CRYPTO-PRO Company
GostR3410-94-EncryptionSyntax
    { iso(1) member-body(2) ru(643) rans(2) cryptopro(2)
      other(1) modules(1) gostR3410-94-EncryptionSyntax(5) 2 }
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,
        gost28147-89-EncryptionSyntax,
        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,
        GostR3410-94-PublicKeyParameters,
        GostR3410-94-PublicKeyAlgorithms
        FROM GostR3410-94-PKISyntax gostR3410-94-PKISyntax
        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,
        Gost28147-89-EncryptedKey
        FROM Gost28147-89-EncryptionSyntax
             gost28147-89-EncryptionSyntax
        -- id-external-PKIX1Explicit93,
        AlgorithmIdentifier, ALGORITHM-IDENTIFIER
        FROM Cryptographic-Gost-Useful-Definitions
             cryptographic-Gost-Useful-Definitions
        -- SubjectPublicKeyInfo
        -- FROM PKIX1Explicit93 id-external-PKIX1Explicit93
        SubjectPublicKeyInfo
        FROM PKIX1Explicit88 {iso(1) identified-organization(3)
        dod(6) internet(1) security(5) mechanisms(5) pkix(7)
        id-mod(0) id-pkix1-explicit(1)}
    ;
  -- CMS/PKCS#7 Key transport OID, Algorithm & Parameters
    -- OID for CMS/PKCS#7 Key transport is id-GostR3410-94 from
    --         GostR3410-94-PKISyntax
    -- Parameters for CMS/PKCS#7 Key transport is
    --         GostR3410-94-PublicKeyParameters from
    --         GostR3410-94-PKISyntax with encryptionParameterOID
    -- Algorithm for CMS/PKCS#7 Key transport is
    --         GostR3410-94-PublicKeyAlgorithms from
    --         GostR3410-94-PKISyntax
    -- SMIMECapability for CMS/PKCS#7 Key transport is
    --         id-GostR3410-94 from GostR3410-94-PKISyntax
```

```
    id-GostR3410-94-KeyTransportSMIMECapability
        OBJECT IDENTIFIER ::= id-GostR3410-94
    GostR3410-94-KeyTransportEncryptedKeyOctetString ::=
        SEQUENCE {
            sessionEncryptedKey    Gost28147-89-EncryptedKey,
            transportParameters    GostR3410-94-TransportParameters --
 OPTIONAL
        }
    GostR3410-94-TransportParameters ::=
        SEQUENCE {
            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
          ),
            ephemeralPublicKey   SubjectPublicKeyInfo OPTIONAL,
            ukm                  OCTET STRING
        }
    GostR3410-94-KeyEncryptionAlgorithms
        ALGORITHM-IDENTIFIER ::= {
        { GostR3410-94-PublicKeyParameters IDENTIFIED BY
               id-GostR3410-94 }
        }
END -- GostR3410-94-EncryptionSyntax
```

9.4 GostR3410-94-SignatureSyntax

```
-- Copyright(C) CRYPTO-PRO Company
GostR3410-94-SignatureSyntax
    { iso(1) member-body(2) ru(643) rans(2) cryptopro(2)
      other(1) modules(1) gostR3410-94-SignatureSyntax(3) 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
        gostR3411-94-DigestSyntax,
        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-GostR3411-94, GostR3411-94-Digest,
        GostR3411-94-DigestParameters,
        id-GostR3411-94-TestParamSet,
        id-GostR3411-94-CryptoProParamSet
        FROM GostR3411-94-DigestSyntax gostR3411-94-DigestSyntax
        id-GostR3410-94,
        GostR3410-94-PublicKeyParameters,
        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 Signature Data Type
    GostR3410-94-Signature ::=
        OCTET STRING (SIZE (64))
  -- GOST R 34.10-94 Signature Parameters & Algorithm
    GostR3410-94-CMSSignatureAlgorithms  ALGORITHM-IDENTIFIER ::= {
        { GostR3410-94-PublicKeyParameters IDENTIFIED BY
               id-GostR3410-94 }
    }

END -- GostR3410-94-SignatureSyntax
```

9.5 GostR3410-2001-EncryptionSyntax

```
-- Copyright(C) CRYPTO-PRO Company
GostR3410-2001-EncryptionSyntax
    { iso(1) member-body(2) ru(643) rans(2) cryptopro(2)
      other(1) modules(1) gostR3410-2001-EncryptionSyntax(11) 2 }
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,
        gost28147-89-EncryptionSyntax,
        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,
        GostR3410-2001-PublicKeyParameters,
        GostR3410-2001-PublicKeyAlgorithms
        FROM GostR3410-2001-PKISyntax gostR3410-2001-PKISyntax
        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,
        Gost28147-89-EncryptedKey
        FROM Gost28147-89-EncryptionSyntax
             gost28147-89-EncryptionSyntax
        -- id-external-PKIX1Explicit93,
```

```
        AlgorithmIdentifier, ALGORITHM-IDENTIFIER
        FROM Cryptographic-Gost-Useful-Definitions
             cryptographic-Gost-Useful-Definitions
        -- id-external-PKIX1Explicit93,
        SubjectPublicKeyInfo
        FROM PKIX1Explicit88 {iso(1) identified-organization(3)
        dod(6) internet(1) security(5) mechanisms(5) pkix(7)
        id-mod(0) id-pkix1-explicit(1)}
    ;
  -- CMS/PKCS#7 Key transport OID, Algorithm & Parameters
    -- OID for CMS/PKCS#7 Key transport is id-GostR3410-2001 from
    --         GostR3410-2001-PKISyntax
    -- Parameters for CMS/PKCS#7 Key transport is
```



```
    --         GostR3410-2001-PublicKeyParameters from
    --         GostR3410-2001-PKISyntax with encryptionParameterOID
    -- Algorithm for CMS/PKCS#7 Key transport is
    --         GostR3410-2001-PublicKeyAlgorithms from
    --         GostR3410-2001-PKISyntax
    -- SMIMECapability for CMS/PKCS#7 Key transport is
    --         id-GostR3410-2001 from GostR3410-2001-PKISyntax
    id-GostR3410-2001-KeyTransportSMIMECapability
        OBJECT IDENTIFIER ::= id-GostR3410-2001
    GostR3410-2001-KeyTransportEncryptedKeyOctetString ::=
        SEQUENCE {
            sessionEncryptedKey    Gost28147-89-EncryptedKey,
            transportParameters    GostR3410-2001-TransportParameters
OPTIONAL
        }
    GostR3410-2001-TransportParameters ::=
        SEQUENCE {
            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
```

```
          ),
            ephemeralPublicKey   SubjectPublicKeyInfo OPTIONAL,
            ukm                  OCTET STRING ( SIZE(8) )
        }
    GostR3410-2001-KeyEncryptionAlgorithms
        ALGORITHM-IDENTIFIER ::= {
          { GostR3410-2001-PublicKeyParameters IDENTIFIED BY
              id-GostR3410-2001 }
            }
END -- GostR3410-2001-EncryptionSyntax
```

9.6 GostR3410-2001-SignatureSyntax

```
-- Copyright(C) CRYPTO-PRO Company
GostR3410-2001-SignatureSyntax
    { iso(1) member-body(2) ru(643) rans(2) cryptopro(2)
      other(1) modules(1) gostR3410-2001-SignatureSyntax(10) 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
        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,
        GostR3410-2001-PublicKeyParameters
        FROM GostR3410-2001-PKISyntax gostR3410-2001-PKISyntax
        AlgorithmIdentifier, ALGORITHM-IDENTIFIER
        FROM Cryptographic-Gost-Useful-Definitions
             cryptographic-Gost-Useful-Definitions
```

```
    ;
  -- GOST R 34.10-2001 Signature Data Type
    GostR3410-2001-Signature ::=
        OCTET STRING (SIZE (64))
  -- GOST R 34.10-2001 Signature Parameters & Algorithm
    GostR3410-2001-CMSSignatureAlgorithms
        ALGORITHM-IDENTIFIER ::= {
          { GostR3410-2001-PublicKeyParameters IDENTIFIED BY
               id-GostR3410-2001 }
        }
END -- GostR3410-2001-SignatureSyntax
```

10 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.208-88] CCITT. Recommendation X.208: Specification of Abstract Syntax Notation One (ASN.1). 1988.

[X.209-88] CCITT. Recommendation X.209: Specification of Basic Encoding Rules for Abstract Syntax Notation One

(ASN.1). 1988..

[CPPK] "Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificates and Certifi- cate Revocation List (CRL), corresponding to the algorithms GOST R 34.10-94, GOST R 34.10-2001, GOST R

34.11-94", IETF draft, <draft-cryptopro-cppk-00.txt>,
...

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 aim 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 collaboration and critical help in creation of this document.

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 CPCMS".

Author's Addresses

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

Vladimir Popov
CRYPTO-PRO
38, Obraztsova,
Moscow, 127018, Russian Federation

EMail: vpopov@CryptoPro.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