Network Working Group
Request for Comments: 4491
Updates: 3279
Category: Standards Track

S. Leontiev, Ed.
CRYPTO-PRO
D. Shefanovski, Ed.
Mobile TeleSystems OJSC

May 2006

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

Использование алгоритмов GOST R 34.10-94, GOST R 34.10-2001 и GOST R 34.11-94 с сертификатами Internet X.509 PKI и профилем CRL

Статус документа

В этом документе приведена спецификация проекта стандартного протокола Internet. Документ служит приглашением к дискуссии в целях развития и совершенствования протокола. Текущий статус стандартизации протокола можно узнать из текущей версии документа Internet Official Protocol Standards (STD 1). Документ может распространяться без ограничений.

Аннотация

Этот документ дополняет RFC 3279, описывая форматы представления, идентификаторы и форматы параметров для алгоритмов GOST R 34.10-94, GOST R 34.10-2001 и GOST R 34.11-94 для их применения в инфраструктуре открытых ключей Internet X.509 (PKI[1]).

Оглавление

1. Введение

Этот документ дополняет RFC 3279 [PKALGS] и описывает соглашения по использованию алгоритмов подписи GOST R 34.10-94 [GOST3431095, GOSTR341094] и GOST R 34.10-2001 [GOST3431004, GOSTR341001], алгоритмов создания производных ключей VKO GOST R 34.10-94 и VKO GOST R 34.10-2001, а также необратимых хэш-функций GOST R 34.11-94 [GOST3431195, GOSTR341194] в инфраструктуре открытых ключей Internet X.509 PKI [PROFILE].

Данный документ обеспечивает дополнительную информацию и спецификации, требуемые российским «Соглашением о совместимости СКЗИ».

Указаны идентификаторы алгоритмов и связанные параметры для субъектов открытых ключей, которые используют алгоритмы GOST R 34.10-94 [GOSTR341094]/VKO GOST R 34.10-94 [CPALGS] и GOST R 34.10-2001 [GOSTR341001]/VKO GOST R 34.10-2001 [CPALGS], а также формат представления для подписей, создаваемых этими алгоритмами. Указаны также идентификаторы алгоритмов для использования необратимой хэш-функции GOST R 34.11-94 с алгоритмами подписи GOST R 34.10-94 и GOST R 34.10-2001.

Данная спецификация определяет содержимое полей signatureAlgorithm, signatureValue, signature и subjectPublicKeyInfo в сертификатах и списках отзыва (CRL) X.509. Для каждого алгоритма представлены подходящие варианты расширения сертификата keyUsage.

Модули ASN.1, включающие все использованные в этом документе определения, можно найти в [CPALGS].

[1]Public Key Infrastructure.

1.1. Уровни требований

Ключевые слова **необходимо** (MUST), **недопустимо** (MUST NOT), **требуется** (REQUIRED), **нужно** (SHALL), **не нужно** (SHALL NOT), **следует** (SHOULD), **не следует** (SHOULD NOT), **рекомендуется** (RECOMMENDED), **возможно** (MAY), **необязательно** (OPTIONAL) в данном документе интерпретируются в соответствии с [RFC2119].

2. Поддержка алгоритмов

В этом разделе приведён обзор криптографических алгоритмов, которые могут использоваться с сертификатами Internet X.509 и профилями CRL [PROFILE]. Описаны необратимые хэш-функции и алгоритмы цифровой подписи, которые могут применяться для подписывания сертификатов и CRL, а также приведены идентификаторы объектов (OID) и представления ASN.1 для открытых ключей, содержащихся в сертификатах.

Удостоверяющие центры (CA[1]) и/или приложения, соответствующие этому стандарту, **должны** поддерживать хотя бы один из указанных алгоритмов открытых ключей и подписи.

2.1. Необратимая хэш-функция

В этом разделе описано использование необратимой и не имеющей коллизий хэш-функции GOST R 34.11-94, единственной, которая может применяться с алгоритмами цифровой подписи GOST R 34.10-94/2001. Данные, которые хэшируются для подписания сертификатов и CRL, полностью описаны в RFC 3280 [PROFILE].

2.1.1. Необратимая хэш-функция GOST R 34.11-94

Хэш-функция GOST R 34.11-94 разработана Главным управлением безопасности связи Федерального агентства правительственной связи и информации, а также Всероссийским НИИ Стандартизации. Алгоритм GOST R 34.11-94 даёт на выходе 256-битовое хэш-значение для произвольного конечного размера входных данных. Данный документ не включает полной спецификации GOST R 34.11-94, которую можно найти в [GOSTR341194] на русском языке. В работе [Schneier95], параграф 18.11, стр. 454 приведено краткое техническое описание алгоритма на английском языке.

Данная функция **должна** всегда применяться с набором параметров, идентифицируемым id-GostR3411-94-CryptoProParamSet (см. параграф 8.2 [CPALGS]).

2.2. Алгоритмы подписи

Соответствующие данной спецификации УЦ могут использовать алгоритмы подписи GOST R 34.10-94 и GOST R 34.10-2001 для подписывания сертификатов и CRL.

Эти алгоритмы подписи во всех случаях **должны** применяться с необратимой хэш-функцией GOST R 34.11-94 как указано в [GOSTR341094] b [GOSTR341001].

В этом разделе определены идентификаторы и параметры алгоритмов для использования в поле signatureAlgorithm структур Certificate и CertificateList.

2.2.1. Алгоритм подписи GOST R 34.10-94

Алгоритм GOST R 34.10-94 разработан Главным управлением безопасности связи Федерального агентства правительственной связи и информации, а также Всероссийским НИИ Стандартизации. В данном документе не приводится полной спецификации алгоритма GOST R 34.10-94, которая дана в [GOSTR341094] на русском языке, а краткое описание на английском языке содержится в работе [Schneier95] (глава 20.3, стр. 495).

Идентификатор объекта ASN.1 для этого алгоритма цифровой подписи приведён ниже.

```
id-GostR3411-94-with-GostR3410-94 OBJECT IDENTIFIER ::=
   { iso(1) member-body(2) ru(643) rans(2) cryptopro(2) gostR3411-94-with-gostR3410-94(4) }
```

Если идентификатор алгоритма id-GostR3411-94-with-GostR3410-94 указан в поле algorithm структуры AlgorithmIdentifier, поле parameters **нужно** опускать. Т. е., в качестве AlgorithmIdentifier **нужно** указывать последовательность (SEQUENCE) из одной компоненты OBJECT IDENTIFIER id-GostR3411-94-with-GostR3410-94.

Алгоритм электронной подписи GOST R 34.10-94 создаёт цифровую подпись в форме двух 256-битовых значений r' и s. Их представление строками октетов будет занимать 64 октета, из которых первые 32 содержат значение s в представлении big-endian, а вторые 32 октета содержат значение r' в том же представлении.

Это определение значения подписи напрямую применимо для CMS [CMS], где такие значения представляются в виде строк октетов. Однако значения подписей в сертификатах и CRL [PROFILE] представляются в форме битовых строк и представление в виде строк октетов должно конвертироваться.

Для преобразования строки октетов в битовую строку старший бит первого октета строки **нужно** поместить в первый бито строки битов и т. д. до младшего бита последнего октета значения подписи, который **нужно** поместить в последний бит строки битов.

2.2.2. Алгоритм подписи GOST R 34.10-2001

Алгоритм GOST R 34.10-2001 разработан Главным управлением безопасности связи Федерального агентства правительственной связи и информации, а также Всероссийским НИИ Стандартизации. Данный документ не содержит полной спецификации GOST R 34.10-2001, она приведена в [GOSTR341001] (на русском языке).

Идентификатор объекта ASN.1 для этого алгоритма цифровой подписи приведён ниже.

```
id-GostR3411-94-with-GostR3410-2001 OBJECT IDENTIFIER ::=
  { iso(1) member-body(2) ru(643) rans(2) cryptopro(2) gostR3411-94-with-gostR3410-2001(3) }
```

Если идентификатор алгоритма id-GostR3411-94-with-GostR3410-2001 указан в поле algorithm структуры AlgorithmIdentifier, поле parameters **нужно** опускать. Т. е., в качестве AlgorithmIdentifier **нужно** указывать последовательность (SEQUENCE) из одной компоненты OBJECT IDENTIFIER id-GostR3411-94-with-GostR3410-2001.

[1]Certification authority.

Алгоритм электронной подписи GOST R 34.10-2001 создаёт цифровую подпись в форме двух 256-битовых значений r и s. Их представление строками октетов будет занимать 64 октета, из которых первые 32 содержат значение s в представлении big-endian, а вторые 32 октета содержат значение r в том же представлении.

Для преобразования строки октетов в битовую строку при использовании в сертификатах и CRL **должен** использоваться процесс, описанный выше (параграф 2.2.1).

2.3. Алгоритмы открытых ключей субъектов

В этом разделе определены идентификаторы OID и параметры открытых ключей для ключей, использующих алгоритмы GOST R 34.10-94 [GOSTR341094]/VKO GOST R 34.10-94 [CPALGS] и GOST R 34.10-2001 [GOSTR341001]/VKO GOST R 34.10-2001 [CPALGS].

Использование одного ключа для подписи и производного ключа **не рекомендуется**. Предусмотренное применение ключа **может** быть указано расширением сертификата keyUsage (см. [PROFILE], параграф 4.2.1.3).

2.3.1. Ключи GOST R 34.10-94

Открытые ключи GOST R 34.10-94 могут применяться с алгоритмом подписи GOST R 34.10-94 [GOSTR341094] и при создании производных ключей с помощью алгоритма VKO GOST R 34.10-94 [CPALGS].

Открытые ключи GOST R 34.10-94 указываются идентификатором OID, приведённым ниже.

```
id-GostR3410-94 OBJECT IDENTIFIER ::=
    { iso(1) member-body(2) ru(643) rans(2) cryptopro(2) gostR3410-94(20) }
```

Поле SubjectPublicKeyInfo.algorithm.algorithm (см. RFC 3280 [PROFILE]) для ключей GOST R 34.10-94 **должно** иметь значение id-GostR3410-94.

Когда идентификатор алгоритма id-GostR3410-94 указан в поле algorithm структуры AlgorithmIdentifier, поле parameters **может** быть опущено или иметь значение NULL. В остальных случаях это поле **должно** иметь приведённую ниже структуру.

```
GostR3410-94-PublicKeyParameters ::=
    SEQUENCE {
        publicKeyParamSet
            OBJECT IDENTIFIER,
        digestParamSet
            OBJECT IDENTIFIER,
        encryptionParamSet
            OBJECT IDENTIFIER DEFAULT
                id-Gost28147-89-CryptoPro-A-ParamSet
    }
```

где:

- publicKeyParamSet — идентификатор параметров открытого ключа для GOST R 34.10-94 (см. параграф 8.3 в [CPALGS]);
- digestParamSet — идентификатор параметров для GOST R 34.11-94 (см. параграф 8.2 в [CPALGS]);
- encryptionParamSet — идентификатор параметров для GOST 28147-89 [GOST28147] (см. параграф 8.1 в [CPALGS])

Отсутствие параметров **нужно** обрабатывать в соответствии с параграфом 6.1 RFC 3280 [PROFILE], т. е., параметры наследуются из сертификата эмитента. Когда переменная working_public_key_parameters имеет значение null, сертификат и все подтверждаемые им подписи **нужно** отвергать.

Открытые ключи GOST R 34.10-94 **должны** кодироваться в формате ASN.1 DER как OCTET STRING, это представление нужно использовать в качестве содержимого (т. е., значения) компоненты subjectPublicKey (BIT STRING) элемента данных SubjectPublicKeyInfo.

```
GostR3410-94-PublicKey ::= OCTET STRING — открытый ключ, Y
```

Поле GostR3410-94-PublicKey **должно** содержать 128 октетов (в представлении little-endian) открытого ключа Y = a^x (mod p), где a и p являются параметрами открытого ключа, а x — секретный ключ.

Некоторые приложения, содержащие ошибки, отбрасывают нулевые биты в конце битовой строки (BIT STRING), содержащей открытый ключ. **Рекомендуется** дополнять строку битов нулями до размера 1048 битов (131 октет) при декодировании с целью обеспечения возможности корректного преобразования инкапсулированной строки октетов (OCTET STRING).

При наличии в сертификате конечного элемента с открытым ключом GOST R 34.10-94 расширения keyUsage **могут** присутствовать следующие значения:

digitalSignature;

nonRepudiation;

keyEncipherment;

keyAgreement.

Если в сертификате открытого ключа GOST R 34.10-94 имеется расширение keyAgreement или keyEnchiperment, **могут** присутствовать также следующие значения:

encipherOnly;

decipherOnly.

В расширении keyUsage **недопустимо** заявлять одновременно encipherOnly и decipherOnly.

При наличии расширения keyUsage в сертификате подписавшего CA или CRL, содержащем открытый ключ GOST R 34.10-94, **могут** присутствовать также значения:

digitalSignature;

nonRepudiation;

keyCertSign;

cRLSign.

2.3.2. Ключи GOST R 34.10-2001

Открытые ключи GOST R 34.10-2001 могут использоваться с алгоритмом подписи GOST R 34.10-2001 [GOSTR341001] и алгоритмом создания производных ключей VKO GOST R 34.10-2001 [CPALGS].

Открытый ключ GOST R 34.10-2001 идентифицируется значением OID, показанным ниже.

```
id-GostR3410-2001 OBJECT IDENTIFIER ::=
    { iso(1) member-body(2) ru(643) rans(2) cryptopro(2) gostR3410-2001(19) }
```

Поле SubjectPublicKeyInfo.algorithm.algorithm (см. RFC 3280 [PROFILE]) для ключей GOST R 34.10-2001 **должно** иметь значение id-GostR3410-2001.

Если в поле algorithm структуры AlgorithmIdentifier указан идентификатор алгоритма id-GostR3410-2001, при кодировании поле parameters **может** быть опущено или установлено в NULL. В остальных случаях это поле **должно** иметь приведённую ниже структуру.

```
GostR3410-2001-PublicKeyParameters ::=
    SEQUENCE {
        publicKeyParamSet
            OBJECT IDENTIFIER,
        digestParamSet
            OBJECT IDENTIFIER,
        encryptionParamSet
            OBJECT IDENTIFIER DEFAULT
                id-Gost28147-89-CryptoPro-A-ParamSet
    }
```

где:

- publicKeyParamSet — идентификатор параметров открытого ключа для GOST R 34.10-2001 (см. параграф 8.4 в [CPALGS])
- digestParamSet — идентификатор параметров для GOST R 34.11-94 (см. параграф 8.2 в [CPALGS])
- encryptionParamSet — идентификатор параметров для GOST 28147-89 [GOST28147] (см. параграф 8.1 в [CPALGS])

Отсутствие параметров **нужно** обрабатывать в соответствии с параграфом 6.1 RFC 3280 [PROFILE], т. е., параметры наследуются из сертификата эмитента. Когда переменная working_public_key_parameters имеет значение null, сертификат и все подтверждаемые им подписи **нужно** отвергать.

Открытые ключи GOST R 34. 10-2001 **должны** кодироваться в формате ASN.1 DER как OCTET STRING, это представление нужно использовать в качестве содержимого (т. е., значения) компоненты subjectPublicKey (BIT STRING) элемента данных SubjectPublicKeyInfo.

```
GostR3410-2001-PublicKey ::= OCTET STRING — вектор открытого ключа, Q
```

Согласно [GOSTR341001], открытый ключ является точкой эллиптической кривой Q = (x,y).

Строка GostR3410-2001-PublicKey **должна** содержать 64 октета, из которых первые 32 являются представлением little-endian для значения x, а оставшиеся 32 октета — представлением little-endian для y. Это соответствует двоичному представлению (<y>256||<x>256) из [GOSTR341001] (параграф 5.3).

Некоторые приложения, содержащие ошибки, отбрасывают нулевые биты в конце битовой строки (BIT STRING), содержащей открытый ключ. **Рекомендуется** дополнять строку битов нулями до размера 528 битов (66 октетов) при декодировании с целью обеспечения возможности корректного преобразования инкапсулированной строки октетов (OCTET STRING).

Ограничения на keyUsage для ключей GOST R 34.10-2001 совпадают с ограничениями, описанными в параграфе 2.3.1 для ключей GOST R 34.10-94.

3. Вопросы безопасности

Приложениям **рекомендуется** проверять значения подписей и открытых ключей на предмет соответствия стандартам [GOSTR341001, GOSTR341094] до использования этих значений.

При использовании сертификатов для поддержки цифровых подписей в качестве эквивалента рукописных подписей в контексте российского Федерального закона «Об электронной цифровой подписи» [RFEDSL] сертификат **должен** включать расширение keyUsage, это **должно** быть критичным и в keyUsage **недопустимо** включать keyEncipherment и keyAgreement.

Удостоверяющим центрам (CA) и приложениям **рекомендуется** обеспечивать уверенность в том, что секретный ключ для создания подписей не используется в течение периода, превосходящего разрешенный (обычно 15 месяцев для алгоритмов GOST R 34.10-94 и GOST R 34.10-2001).

Обсуждение вопросов безопасности, связанных с использованием параметров алгоритма, приведено в разделе «Вопросы безопасности» [CPALGS].

4. Примеры

4.1. Сертификат GOST R 34.10-94

```
-----BEGIN CERTIFICATE-----
MIICCzCCAboCECMO42BGlSTOxwvklBgufuswCAYGKoUDAgIEMGkxHTAbBgNVBAMM
FEdvc3RSMzQxMC05NCBleGFtcGxlMRIwEAYDVQQKDAlDcnlwdG9Qcm8xCzAJBgNV
BAYTAlJVMScwJQYJKoZIhvcNAQkBFhhHb3N0UjM0MTAtOTRAZXhhbXBsZS5jb20w
HhcNMDUwODE2MTIzMjUwWhcNMTUwODE2MTIzMjUwWjBpMR0wGwYDVQQDDBRHb3N0
UjM0MTAtOTQgZXhhbXBsZTESMBAGA1UECgwJQ3J5cHRvUHJvMQswCQYDVQQGEwJS
VTEnMCUGCSqGSIb3DQEJARYYR29zdFIzNDEwLTk0QGV4YW1wbGUuY29tMIGlMBwG
BiqFAwICFDASBgcqhQMCAiACBgcqhQMCAh4BA4GEAASBgLuEZuF5nls02CyAfxOo
GWZxV/6MVCUhR28wCyd3RpjG+0dVvrey85NsObVCNyaE4g0QiiQOHwxCTSs7ESuo
v2Y5MlyUi8Go/htjEvYJJYfMdRv05YmKCYJo01x3pg+2kBATjeM+fJyR1qwNCCw+
eMG1wra3Gqgqi0WBkzIydvp7MAgGBiqFAwICBANBABHHCH4S3ALxAiMpR3aPRyqB
g1DjB8zy5DEjiULIc+HeIveF81W9lOxGkZxnrFjXBSqnjLeFKgF1hffXOAP7zUM=
-----END CERTIFICATE-----

  0 30  523: SEQUENCE {
  4 30  442:  SEQUENCE {
  8 02   16:   INTEGER
            :    23 0E E3 60 46 95 24 CE C7 0B E4 94 18 2E 7E EB
 26 30    8:   SEQUENCE {
 28 06    6:    OBJECT IDENTIFIER
            :     id-GostR3411-94-with-GostR3410-94 (1 2 643 2 2 4)
            :    }
 36 30  105:   SEQUENCE {
 38 31   29:    SET {
 40 30   27:     SEQUENCE {
 42 06    3:      OBJECT IDENTIFIER commonName (2 5 4 3)
 47 0C   20:      UTF8String 'GostR3410-94 example'
            :      }
            :     }
 69 31   18:    SET {
 71 30   16:     SEQUENCE {
 73 06    3:      OBJECT IDENTIFIER organizationName (2 5 4 10)
 78 0C    9:      UTF8String 'CryptoPro'
            :      }
            :     }
 89 31   11:    SET {
 91 30    9:     SEQUENCE {
 93 06    3:      OBJECT IDENTIFIER countryName (2 5 4 6)
 98 13    2:      PrintableString 'RU'
            :      }
            :     }
102 31   39:    SET {
104 30   37:     SEQUENCE {
106 06    9:      OBJECT IDENTIFIER emailAddress (1 2 840 113549 1 9 1)
117 16   24:      IA5String 'GostR3410-94@example.com'
            :      }
            :     }
            :    }
143 30   30:   SEQUENCE {
145 17   13:    UTCTime '050816123250Z'
160 17   13:    UTCTime '150816123250Z'
            :    }
175 30  105:   SEQUENCE {
177 31   29:    SET {
179 30   27:     SEQUENCE {
181 06    3:      OBJECT IDENTIFIER commonName (2 5 4 3)
186 0C   20:      UTF8String 'GostR3410-94 example'
            :      }
            :     }
208 31   18:    SET {
210 30   16:     SEQUENCE {
212 06    3:      OBJECT IDENTIFIER organizationName (2 5 4 10)
217 0C    9:      UTF8String 'CryptoPro'
            :      }
            :     }
228 31   11:    SET {
230 30    9:     SEQUENCE {
232 06    3:      OBJECT IDENTIFIER countryName (2 5 4 6)
237 13    2:      PrintableString 'RU'
            :      }
            :     }
241 31   39:    SET {
243 30   37:     SEQUENCE {
245 06    9:      OBJECT IDENTIFIER emailAddress (1 2 840 113549 1 9 1)
256 16   24:      IA5String 'GostR3410-94@example.com'
            :      }
            :     }
            :    }
282 30  165:   SEQUENCE {
285 30   28:    SEQUENCE {
287 06    6:     OBJECT IDENTIFIER id-GostR3410-94 (1 2 643 2 2 20)
```

```
 295 30   18:     SEQUENCE {
 297 06    7:      OBJECT IDENTIFIER
           :       id-GostR3410-94-CryptoPro-A-ParamSet
           :         (1 2 643 2 2 32 2)
 306 06    7:      OBJECT IDENTIFIER
           :       id-GostR3411-94-CryptoProParamSet
           :         (1 2 643 2 2 30 1)
           :      }
           :     }
 315 03  132:    BIT STRING 0 unused bits, encapsulates {
 319 04  128:     OCTET STRING
           :      BB 84 66 E1 79 9E 5B 34 D8 2C 80 7F 13 A8 19 66
           :      71 57 FE 8C 54 25 21 47 6F 30 0B 27 77 46 98 C6
           :      FB 47 55 BE B7 B2 F3 93 6C 39 B5 42 37 26 84 E2
           :      0D 10 8A 24 0E 1F 0C 42 4D 2B 3B 11 2B A8 BF 66
           :      39 32 5C 94 8B C1 A8 FE 1B 63 12 F6 09 25 87 CC
           :      75 1B F4 E5 89 8A 09 82 68 D3 5C 77 A6 0F B6 90
           :      10 13 8D E3 3E 7C 9C 91 D6 AC 0D 08 2C 3E 78 C1
           :      B5 C2 B6 B7 1A A8 2A 8B 45 81 93 32 32 76 FA 7B
           :     }
           :    }
           :   }
 450 30    8:  SEQUENCE {
 452 06    6:   OBJECT IDENTIFIER
           :    id-GostR3411-94-with-GostR3410-94 (1 2 643 2 2 4)
           :   }
 460 03   65:  BIT STRING 0 unused bits
           :   11 C7 08 7E 12 DC 02 F1 02 23 29 47 76 8F 47 2A
           :   81 83 50 E3 07 CC F2 E4 31 23 89 42 C8 73 E1 DE
           :   22 F7 85 F3 55 BD 94 EC 46 91 9C 67 AC 58 D7 05
           :   2A A7 8C B7 85 2A 01 75 85 F7 D7 38 03 FB CD 43
           :  }
```

В подписи приведённого выше сертификата r' имеет значение

```
0x22F785F355BD94EC46919C67AC58D7052AA78CB7852A017585F7D73803FBCD43
```

s имеет значение

```
0x11C7087E12DC02F102232947768F472A818350E307CCF2E431238942C873E1DE
```

4.2. Сертификат GOST R 34.10-2001

```
-----BEGIN CERTIFICATE-----
MIIB0DCCAX8CECv1xh7CEb0Xx9zUYma0LiEwCAYGKoUDAgIDMG0xHzAdBgNVBAMM
Fkdvc3RSMzQxMC0yMDAxIGV4YW1wbGUxEjAQBgNVBAoMCUNyeXB0b1BybzELMAkG
A1UEBhMCUlUxKTAnBgkqhkiG9w0BCQEWGkdvc3RSMzQxMC0yMDAxQGV4YW1wbGUu
Y29tMB4XDTA1MDgxNjE0MTgyMFoXDTE1MDgxNjE0MTgyMFowbTEfMB0GA1UEAwwW
R29zdFIzNDEwLTIwMDEgZXhhbXBsZTESMBAGA1UECgwJQ3J5cHRvUHJvMQswCQYD
VQQGEwJSVTEpMCcGCSqGSIb3DQEJARYaR29zdFIzNDEwLTIwMDFAZXhhbXBsZS5j
b20wYzAcBgYqhQMCAhMwEgYHKoUDAgIkAAYHKoUDAgIeAQNDAARAhJVodWACGkB1
CM0TjDGJLP3lBQN6Q1z0bSsP508yfleP68wWuZWIA9CafIWuD+SN6qa7flbHy7Df
D2a8yuoaYDAIBgYqhQMCAgMDQQA8L8kJRLcnqeyn1en7U23Sw6pkfEQu3u0xFkVP
vFQ/3cHeF26NG+xxtZPz3TaTVXdoiYkXYiD02rEx1bUcM97i
-----END CERTIFICATE-----

  0 30  464: SEQUENCE {
  4 30  383:  SEQUENCE {
  8 02   16:   INTEGER
           :    2B F5 C6 1E C2 11 BD 17 C7 DC D4 62 66 B4 2E 21
 26 30    8:   SEQUENCE {
 28 06    6:    OBJECT IDENTIFIER
           :     id-GostR3411-94-with-GostR3410-2001 (1 2 643 2 2 3)
           :    }
 36 30  109:   SEQUENCE {
 38 31   31:    SET {
 40 30   29:     SEQUENCE {
 42 06    3:      OBJECT IDENTIFIER commonName (2 5 4 3)
 47 0C   22:      UTF8String 'GostR3410-2001 example'
           :      }
           :     }
 71 31   18:    SET {
 73 30   16:     SEQUENCE {
 75 06    3:      OBJECT IDENTIFIER organizationName (2 5 4 10)
 80 0C    9:      UTF8String 'CryptoPro'
           :      }
           :     }
 91 31   11:    SET {
 93 30    9:     SEQUENCE {
 95 06    3:      OBJECT IDENTIFIER countryName (2 5 4 6)
100 13    2:      PrintableString 'RU'
           :      }
           :     }
104 31   41:    SET {
106 30   39:     SEQUENCE {
108 06    9:      OBJECT IDENTIFIER emailAddress (1 2 840 113549 1 9 1)
119 16   26:      IA5String 'GostR3410-2001@example.com'
           :      }
```

```
            :     }
            :    }
 147 30   30:   SEQUENCE {
 149 17   13:    UTCTime '050816141820Z'
 164 17   13:    UTCTime '150816141820Z'
            :    }
 179 30  109:   SEQUENCE {
 181 31   31:    SET {
 183 30   29:     SEQUENCE {
 185 06    3:      OBJECT IDENTIFIER commonName (2 5 4 3)
 190 0C   22:      UTF8String 'GostR3410-2001 example'
            :     }
            :    }
 214 31   18:    SET {
 216 30   16:     SEQUENCE {
 218 06    3:      OBJECT IDENTIFIER organizationName (2 5 4 10)
 223 0C    9:      UTF8String 'CryptoPro'
            :     }
            :    }
 234 31   11:    SET {
 236 30    9:     SEQUENCE {
 238 06    3:      OBJECT IDENTIFIER countryName (2 5 4 6)
 243 13    2:      PrintableString 'RU'
            :     }
            :    }
 247 31   41:    SET {
 249 30   39:     SEQUENCE {
 251 06    9:      OBJECT IDENTIFIER emailAddress (1 2 840 113549 1 9 1)
 262 16   26:      IA5String 'GostR3410-2001@example.com'
            :     }
            :    }
            :   }
 290 30   99:   SEQUENCE {
 292 30   28:    SEQUENCE {
 294 06    6:     OBJECT IDENTIFIER id-GostR3410-2001 (1 2 643 2 2 19)
 302 30   18:     SEQUENCE {
 304 06    7:      OBJECT IDENTIFIER
            :       id-GostR3410-2001-CryptoPro-XchA-ParamSet
            :       (1 2 643 2 2 36 0)
 313 06    7:      OBJECT IDENTIFIER
            :       id-GostR3411-94-CryptoProParamSet
            :       (1 2 643 2 2 30 1)
            :     }
            :    }
 322 03   67:    BIT STRING 0 unused bits, encapsulates {
 325 04   64:     OCTET STRING
            :      84 95 68 75 60 02 1A 40 75 08 CD 13 8C 31 89 2C
            :      FD E5 05 03 7A 43 5C F4 6D 2B 0F E7 4F 32 7E 57
            :      8F EB CC 16 B9 95 88 03 D0 9A 7C 85 AE 0F E4 8D
            :      EA A6 BB 7E 56 C7 CB B0 DF 0F 66 BC CA EA 1A 60
            :     }
            :    }
            :   }
 391 30    8:  SEQUENCE {
 393 06    6:   OBJECT IDENTIFIER
            :    id-GostR3411-94-with-GostR3410-2001 (1 2 643 2 2 3)
            :   }
 401 03   65:  BIT STRING 0 unused bits
            :   3C 2F C9 09 44 B7 27 A9 EC A7 D5 E9 FB 53 6D D2
            :   C3 AA 64 7C 44 2E DE ED 31 16 45 4F BC 54 3F DD
            :   C1 DE 17 6E 8D 1B EC 71 B5 93 F3 DD 36 93 55 77
            :   68 89 89 17 62 20 F4 DA B1 31 D5 B5 1C 33 DE E2
            :  }
```

В открытом ключе приведённого выше сертификата x имеет значение

```
0x577E324FE70F2B6DF45C437A0305E5FD2C89318C13CD0875401A026075689584
```

y имеет значение

```
0x601AEACABC660FDFB0CBC7567EBBA6EA8DE40FAE857C9AD0038895B916CCEB8F
```

Соответствующий секретный ключ имеет значение d

```
0x0B293BE050D0082BDAE785631A6BAB68F35B42786D6DDA56AFAF169891040F77
```

В приведённом выше сертификате r имеет значение

```
0xC1DE176E8D1BEC71B593F3DD3693557768898917622 0F4DAB131D5B51C33DEE2
```

s имеет значение

```
0x3C2FC90944B727A9ECA7D5E9FB536DD2C3AA647C442EDEED3116454FBC543FDD
```

5. Благодарности

Этот документ был подготовлен в соответствии с «Соглашением о совместимости СКЗИ», подписанным ФГУП НТЦ «Атлас», ООО «КРИПТО-ПРО», ООО «Фактор-ТС», ЗАО «МО ПНИЭИ», ООО «Инфотекс», ЗАО «СПбРЦЗИ», ООО «Криптоком», ООО «Р-Альфа». Целью этого соглашения является обеспечение взаимной совместимости продукции и решений.

Авторы выражают свою признательность

представительству компании Microsoft в России за предоставление информации о продукции и решениях компании, а также технические консультации в части PKI;

представительству RSA Security в России и компании «Демос» за активное сотрудничество и неоценимую помощь в создании этого документа;

RSA Security Inc за тестирование совместимости предложенных форматов данных при встраивании в продукцию RSA Keon;

Baltimore Technology plc за тестирование совместимости предложенных форматов данных при встраивании в их продукцию UniCERT;

Peter Gutmann за полезную программу dumpasn1;

Russ Housley (Vigil Security, LLC, housley@vigilsec.com) и Василию Сахарову (DEMOS Co Ltd, svp@dol.ru) за поощрение авторов к созданию этого документа;

Григорию Чудову за помощь в прохождении процесса IETF для этого документа;

Дмитрию Приходько (VSTU, PrikhodkoDV@volgablob.ru) за неоценимую помощь в корректуре этого документа и проверку формы и содержания структур ASN.1 упомянутых и использованных в документе.

6. Литература

6.1. Нормативные документы

[GOST28147] "Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования. ГОСТ 28147-89", Государственный стандарт СССР, Государственный комитет СССР по стандартизации, 1989. (на русском языке)

[GOST3431195] "Информационная технология. Криптографическая защита информации. Функция хэширования.", ГОСТ 34.311-95, Межгосударственный совет по стандартизации, метрологии и сертификации (МГС), Минск, 1995. (на русском языке)

[GOST3431095] "Информационная технология. Криптографическая защита информации. Процедуры выработки и проверки электронной цифровой подписи на базе асимметричного криптографического алгоритма.", ГОСТ 34.310-95, Межгосударственный совет по стандартизации, метрологии и сертификации (МГС), Минск, 1995. (на русском языке)

[GOST3431004] "Информационная технология. Криптографическая защита информации. Процессы формирования и проверки электронной цифровой подписи."[1], ГОСТ 34.310-2004, Межгосударственный совет по стандартизации, метрологии и сертификации (МГС), Минск, 2004. (на русском языке)

[GOSTR341094] "Информационная технология. Криптографическая защита информации. Процедуры выработки и проверки электронной цифровой подписи на базе асимметричного криптографического алгоритма.", ГОСТ Р 34.10-94, Государственный стандарт Российской Федерации, Госстандарт России, 1994. (на русском языке)

[GOSTR341001] "Информационная технология. Криптографическая защита информации. Процессы формирования и проверки электронной цифровой подписи.", ГОСТ Р 34.10-2001, Государственный стандарт Российской Федерации, Госстандарт России, 2001. (на русском языке)

[GOSTR341194] "Информационная технология. Криптографическая защита информации. Функция хэширования.", ГОСТ Р 34.11-94[2], Государственный стандарт Российской Федерации, Госстандарт России, 1994. (на русском языке)

[CPALGS] Popov, V., Kurepkin, I., and S. Leontiev, "Additional Cryptographic Algorithms for Use with GOST 28147-89, GOST R 34.10-94, GOST R 34.10-2001, and GOST R 34.11-94 Algorithms", RFC 4357, January 2006.

[PKALGS] Bassham, L., Polk, W., and R. Housley, "Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 3279, April 2002.

[PROFILE] 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.

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

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

6.2. Дополнительная литература

[Schneier95] B. Schneier, Applied Cryptography, Second Edition, John Wiley & Sons, Inc., 1995.

[RFEDSL] Федеральный закон «Об электронной цифровой подписи» от 10.01.2002 N 1-ФЗ[3]

[CMS] Housley, R., "Cryptographic Message Syntax (CMS)", RFC 3852, July 2004.

Адреса авторов

Сергей Леонтьев, редактор

[1]Название стандарта на английском языке в оригинале указано неверно (см. https://www.rfc-editor.org/errata/eid5099). *Прим. перев.*
[2]В оригинале ошибочно указано GOST R 31.10-94 (см. https://www.rfc-editor.org/errata/eid5089). *Прим. перев.*
[3]В соответствии с Федеральным законом от 6 апреля 2011 г. N 63-ФЗ утратил силу с 01.07.2013 г. *Прим. перев.*

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

Денис Стефановский, редактор
Mobile TeleSystems OJSC
4, Marksistskaya Str.,
Moscow, 109147, Russian Federation
EMail: dbs@mts.ru

Григорий Чудов
CRYPTO-PRO
38, Obraztsova,
Moscow, 127018, Russian Federation
EMail: chudov@cryptopro.ru

Александр Афанасьев
Factor-TS
office 711, 14, Presnenskij val,
Moscow, 123557, Russian Federation
EMail: afa1@factor-ts.ru

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

Болеслав Изотов
FGUE STC "Atlas"
38, Obraztsova,
Moscow, 127018, Russian Federation
EMail: izotov@nii.voskhod.ru

Елена Минаева
MD PREI
build 3, 6A, Vtoroj Troitskij per.,
Moscow, Russian Federation
EMail: evminaeva@mail.ru

Игорь Остапенко
MD PREI
Office 600, 14, B.Novodmitrovskaya,
Moscow, Russian Federation
EMail: igori@mo.msk.ru

Сергей Муругов
R-Alpha

4/1, Raspletina,

Moscow, 123060, Russian Federation

EMail: msm@top-cross.ru

Игорь Устинов

Cryptocom

office 239, 51, Leninskij prospekt,

Moscow, 119991, Russian Federation

EMail: igus@cryptocom.ru

Анатолий Еркин

SPRCIS (SPbRCZI)

1, Obrucheva,

St.Petersburg, 195220, Russian Federation

EMail: erkin@nevsky.net

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru

Полное заявление авторских прав

Интеллектуальная собственность

Подтверждение

Финансирование функций RFC Editor обеспечено IETF Administrative Support Activity (IASA).