ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

РЕКОМЕНДАЦИИ ПО СТАНДАРТИЗАЦИИ

Р 1323565.1.023—2018

Информационная технология

КРИПТОГРАФИЧЕСКАЯ ЗАЩИТА ИНФОРМАЦИИ

Использование алгоритмов ГОСТ Р 34.10—2012, ГОСТ Р 34.11—2012 в сертификате, списке аннулированных сертификатов (CRL) и запросе на сертификат PKCS #10 инфраструктуры открытых ключей X.509

Издание официальное

Москва
Стандартинформ
2018

gost.e-ecolog.ru

Предисловие

1 РАЗРАБОТАНЫ Открытым акционерным обществом «Информационные технологии и коммуникационные системы (ОАО «ИнфоТеКС»)

2 ВНЕСЕНЫ Техническим комитетом по стандартизации ТК 26 «Криптографическая защита информации»

3 УТВЕРЖДЕНЫ И ВВЕДЕНЫ В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 26 декабря 2018 г. № 1155-ст

4 ВВЕДЕНЫ ВПЕРВЫЕ

*Правила применения настоящих рекомендаций установлены в статье 26 Федерального закона от 29 июня 2015 г. № 162-ФЗ «О стандартизации в Российской Федерации». Информация об изменениях к настоящим рекомендациям публикуется в ежегодном (по состоянию на 1 января текущего года) информационном указателе «Национальные стандарты», а официальный текст изменений и поправок — в ежемесячном информационном указателе «Национальные стандарты». В случае пересмотра (замены) или отмены настоящих рекомендаций соответствующее уведомление будет опубликовано в ближайшем выпуске ежемесячного информационного указателя «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет (*gost.e-ecolog.ru

© Стандартинформ, оформление, 2018

Настоящие рекомендации не могут быть полностью или частично воспроизведены, тиражированы и распространены в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

Содержание

1 Область применения 1
2 Нормативные ссылки 1
3 Термины и определения 2
4 Используемые структуры 2
4.1 Запрос на сертификат 2
4.2 Сертификат 3
4.3 Список отозванных сертификатов 4
5 Порядок использования российских алгоритмов 5
5.1 Подпись ГОСТ Р 34.10—2012 5
5.2 Открытый ключ и его параметры (структура SubjectPublicKeyInfo) 6
5.3 Область использования ключа 7
Приложение А (справочное) Контрольные примеры 8
Библиография 17

Введение

Настоящие рекомендации содержат описание форматов кодирования, идентификаторов и форматов параметров для алгоритмов ГОСТ Р 34.10—2012 и ГОСТ Р 34.11—2012 при их использовании на территории Российской Федерации, в учреждениях Российской Федерации за рубежом и в находящихся за рубежом обособленных подразделениях юридических лиц, образованных в соответствии с законодательством Российской Федерации.

Необходимость разработки новой версии настоящих рекомендаций вызвана потребностью в обеспечении совместимости использования российских алгоритмов подписи ГОСТ Р 34.10—2012, алгоритмов функции хэширования ГОСТ Р 34.11—2012, алгоритмов согласования ключей в инфраструктуре открытых ключей (PKI), а также принятием алгоритмов шифрования ГОСТ Р 34.12—2015 и режимов шифрования ГОСТ Р 34.13—2015.

РЕКОМЕНДАЦИИ ПО СТАНДАРТИЗАЦИИ

Информационная технология

КРИПТОГРАФИЧЕСКАЯ ЗАЩИТА ИНФОРМАЦИИ

Использование алгоритмов ГОСТ Р 34.10—2012, ГОСТ Р 34.11—2012 в сертификате, списке аннулированных сертификатов (CRL) и запросе на сертификат PKCS #10 инфраструктуры открытых ключей X.509

Information technology. Cryptographic information security. Usage of GOST R 34.10—2012 and GOST R 34.11—2012 algorithms in certificate, CRL and PKCS#10 certificate request in X.509 public key infrastructure

Дата введения — 2019—06—01

1 Область применения

Настоящие рекомендации являются дополнением к ГОСТ Р ИСО/МЭК 9594-8. В рекомендациях описываются правила использования алгоритма подписи ГОСТ Р 34.10 и функции хэширования ГОСТ Р 34.11—2012 в инфраструктуре открытых ключей (PKI) X.509.

Для ключей, соответствующих алгоритму подписи ГОСТ Р 34.10—2012, определены идентификаторы алгоритмов и соответствующих им параметров. Также в документе указаны идентификаторы алгоритмов функции хэширования ГОСТ Р 34.11—2012 с алгоритмом подписи ГОСТ Р 34.10—2012.

В документе определено содержимое полей `signature`, `signatureAlgorithm` и `subjectPublicKey` в сертификатах X.509, списках аннулированных сертификатов и запросах на сертификаты PKCS #10.

2 Нормативные ссылки

В настоящих рекомендациях использованы нормативные ссылки на следующие стандарты:

ГОСТ Р 34.10—2012 Информационная технология. Криптографическая защита информации. Процессы формирования и проверки электронной цифровой подписи

ГОСТ Р 34.11—2012 Информационная технология. Криптографическая защита информации. Функция хэширования

ГОСТ Р ИСО/МЭК 8824-1 Информационная технология. Абстрактная синтаксическая нотация версии один (АСН.1). Часть 1. Спецификация основной нотации

ГОСТ Р ИСО/МЭК 8825-1—2003 Информационная технология. Правила кодирования АСН.1. Часть 1. Спецификация базовых (BER), канонических (CER) и отличительных (DER) правил кодирования

ГОСТ Р ИСО/МЭК 9594-8—98 Информационная технология. Взаимосвязь открытых систем. Справочник. Часть 8. Основы аутентификации

Р 50.1.113—2016 Информационная технология. Криптографическая защита информации. Криптографические алгоритмы, сопутствующие применению алгоритмов электронной цифровой подписи и функции хэширования

Р 50.1.114—2016 Информационная технология. Криптографическая защита информации. Параметры эллиптических кривых для криптографических алгоритмов и протоколов

Примечание — При пользовании настоящими рекомендациями целесообразно проверить действие ссылочных стандартов в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет или по ежегодному информационному указателю «Национальные стандарты», который опубликован по состоянию на 1 января текущего года, и по выпускам ежемесячного информационного указателя «Национальные стандарты» за текущий год. Если заменен ссы-

Издание официальное

лочный стандарт, на который дана недатированная ссылка, то рекомендуется использовать действующую версию этого стандарта с учетом всех внесенных в данную версию изменений. Если заменен ссылочный стандарт, на который дана датированная ссылка, то рекомендуется использовать версию этого стандарта с указанным выше годом утверждения (принятия). Если после утверждения настоящих рекомендаций в ссылочный стандарт, на который дана датированная ссылка, внесено изменение, затрагивающее положение, на которое дана ссылка, то это положение рекомендуется применять без учета данного изменения. Если ссылочный стандарт отменен без замены, то положение, в котором дана ссылка на него, применяется в части, не затрагивающей эту ссылку.

3 Термины и определения

В настоящих рекомендациях применены следующие термины с соответствующими определениями.

3.1 **ACH.1:** Абстрактная синтаксическая нотация, определенная в ГОСТ Р ИСО/МЭК 8824-1.

3.2 **DER:** Правило кодирования структур данных, созданных в соответствии с ACH.1, описанное в [1].

Примечание — В настоящих рекомендациях термины «ключ проверки подписи» и «ключ подписи», введенные в ГОСТ Р 34.10—2012, являются синонимами терминов «открытый ключ» и «закрытый ключ» соответственно.

Отечественные наборы параметров и алгоритмы, используемые в рекомендациях, определены в Р 50.1.114—2016 и Р 50.1.113—2016.

4 Используемые структуры

Механизм информационного обмена при использовании сертификатов описывается следующим образом:

1 Субъект формирует запрос на сертификат в соответствии с 4.1 и посылает данный запрос удостоверяющему центру (УЦ);

2 УЦ формирует на основании данного запроса сертификат с открытым ключом в формате X.509 в соответствии с 4.2 и посылает его субъекту;

3 Для отзыва сертификатов до истечения срока их действия УЦ формирует список отозванных сертификатов в соответствии с 4.3

4.1 Запрос на сертификат

Запрос на сертификат представляет собой структуру `CertificationRequest` в формате ACH.1 (см. [2]):

```
CertificationRequest ::= SEQUENCE
{
      certificationRequestInfo            CertificationRequestInfo,
      signatureAlgorithm
      AlgorithmIdentifier{{SignatureAlgorithms}},
      signature                      BIT STRING
}
```

Поля структуры имеют следующие значения:

`certificationRequestInfo` — информация запроса на сертификат, содержащая, в частности, открытый ключ субъекта, сформированная в соответствии с 4.1.1;

`signatureAlgorithm` — информация об алгоритме подписи, который использовался при формировании подписи данных поля `certificationRequestInfo` структуры `CertificationRequest`, сформированная в соответствии с 4.1.2;

`signature` — подпись данных поля `certificationRequestInfo` структуры `CertificationRequest`, сформированная в соответствии с 4.1.3.

4.1.1 Поле certificationRequestInfo структуры CertificationRequest

Поле `certificationRequestInfo` структуры `CertificationRequest` задается структурой `CertificationRequestInfo` в формате ACH.1:

```
CertificationRequestInfo ::= SEQUENCE
{
      version            INTEGER { v1(0) } (v1,...),
      subject            Name,
```

```
    subjectPKInfo      SubjectPublicKeyInfo{{PKInfoAlgorithms}},
    attributes[0]      Attributes{{CRIAttributes}}
}
```

Поля структуры имеют следующие значения:

version — номер версии стандарта. Для стандарта PKCS#10 v1.7 данное поле должно принимать значение 0;

subject — имя субъекта, чей открытый ключ используется для формирования сертификата;

subjectPKInfo — набор элементов, содержащих информацию об открытом ключе сертификата и сам ключ;

attributes — набор атрибутов.

Более подробное описание полей структуры CertificationRequestInfo можно найти в [2]. Поле subjectPKInfo структуры CertificationRequestInfo имеет структуру SubjectPublicKeyInfo и задается в соответствии с 5.2.

4.1.2 Поле signatureAlgorithm структуры CertificationRequest

Поле signatureAlgorithm структуры CertificationRequest задается структурой AlgorithmIdentifier, описанной в 5.1.1, и содержит информацию об алгоритме подписи, который использовался при формировании поля signature структуры CertificationRequest (см. 4.1.3). При этом используемый алгоритм подписи должен соответствовать открытому ключу субъекта, указанному в поле subjectPublicKey структуры SubjectPublicKeyInfo (см. 5.2.2).

4.1.3 Поле signature структуры CertificationRequest

Поле signature структуры CertificationRequest содержит в себе подпись значения поля certificationRequestInfo структуры CertificationRequest, закодированного в формате DER. Значение подписи формируется в соответствии с алгоритмом, указанным в поле signatureAlgorithm структуры CertificationRequest на закрытом ключе субъекта, соответствующем открытому ключу субъекта, указанному в поле subjectPublicKey структуры SubjectPublicKeyInfo (см. 5.2.2), и формируется в соответствии с 5.1.2.

4.2 Сертификат

Сертификат формата x.509 задается структурой Certificate в формате АСН.1 (см. [3]):

```
Certificate ::= SEQUENCE
{
    tbsCertificate          TBSCertificate,
    signatureAlgorithm          AlgorithmIdentifier,
    signatureValue          BIT STRING
}
```

Поля структуры имеют следующие значения:

tbsCertificate — набор параметров сертификата, которые должны быть подписаны;

signatureAlgorithm — информация об алгоритме подписи, который использовался при формировании подписи данных поля tbsCertificate структуры Certificate;

signatureValue — значение подписи, сформированное от данных поля tbsCertificate структуры Certificate в соответствии с алгоритмом, указанным в поле signatureAlgorithm структуры Certificate.

4.2.1 Поле tsbCertificate структуры Certificate

Поле tbsCertificate структуры Certificate содержит в себе информацию о сертификате, которая должна быть подписана, и задается структурой TBSCertificate, представляющейся в формате АСН.1 следующим образом:

```
TBSCertificate ::= SEQUENCE
{
    version[0]              EXPLICIT Version DEFAULT v1,
    serialNumber                CertificateSerialNumber,
    signature               AlgorithmIdentifier,
    issuer                      Name,
    validity                Validity,
    subject                 Name,
    subjectPublicKeyInfo    SubjectPublicKeyInfo,
    issuerUniqueID[1]       IMPLICIT UniqueIdentifier OPTIONAL,
```

```
      subjectUniqueID[2]            IMPLICIT UniqueIdentifier OPTIONAL,
      extensions[3]           EXPLICIT Extensions OPTIONAL
}
```

Более подробное описание всех полей структуры TBSCertificate содержится в [3].

Поле signature структуры TBSCertificate задается структурой AlgorithmIdentifier в соответствии с 5.1.1 и должно совпадать со значением поля signatureAlgorithm структуры Certificate (см.4.2.2).

Поле subjectPublicKeyInfo структуры TBSCertificate задается структурой SubjectPublicKeyInfo в соотвествии с 5.2.1 и содержит алгоритм открытого ключа и сам ключ.

4.2.2 Поле signatureAlgorithm структуры Certificate

Поле signatureAlgorithm структуры Certificate задается структурой AlgorithmIdentifier, описанной в 5.1.1, и содержит информацию об алгоритме подписи, который использовался при формировании поля signature структуры Certificate (см. 4.2.3). При этом используемый алгоритм подписи должен соответствовать открытому ключу издателя сертификата, указанному в поле issuer структуры TBSCertificate (см. 4.2.1).

4.2.3 Поле signatureValue структуры Certificate

Поле signatureValue структуры Certificate содержит в себе подпись значения поля tbsCertificate структуры Certificate, закодированного в формате DER. Значение подписи формируется в соответствии с 5.1.2 по алгоритму, указанному в поле signatureAlgorithm структуры Certificate, на закрытом ключе издателя сертификата, имя которого указано в поле issuer структуры TBSCertificate (см. 4.2.1).

4.3 Список отозванных сертификатов

Список отозванных сертификатов задается структурой CertificateList в формате АСН.1 (см. [3]):

```
CertificateList ::= SEQUENCE
{
      tbsCertList             TBSCertList,
      signatureAlgorithm            AlgorithmIdentifier,
      signatureValue          BIT STRING
}
```

Поля структуры имеют следующие значения:

tbsCertList — набор параметров списка отозванных сертификатов, которые должны быть подписаны;

signatureAlgorithm — информация об алгоритме подписи, который использовался при формировании подписи данных поля tbsCertList структуры CertificateList;

signatureValue — значение подписи, сформированное от данных поля tbsCertList структуры CertificateList в соответствии с алгоритмом, указанным в поле signatureAlgorithm структуры CertificateList.

4.3.1 Поле tbsCertList структуры CertificateList

Поле tbsCertList структуры CertificateList содержит в себе информацию о списке отозванных сертификатов, которая должна быть подписана, и задается структурой TBSCertList, представляющейся в формате АСН.1 следующим образом:

```
TBSCertList ::= SEQUENCE
{
      version                 Version OPTIONAL,
      signature               AlgorithmIdentifier,
      issuer                        Name,
      thisUpdate              Time,
      nextUpdate              Time OPTIONAL,
      revokedCertificates           SEQUENCE OF SEQUENCE
      {
            userCertificate         CertificateSerialNumber,
            revocationDate          Time,
            crlEntryExtensions            Extensions OPTIONAL
```

```
    } OPTIONAL,
    crlExtensions[0]        EXPLICIT Extensions OPTIONAL
}
```

Более подробное описание всех полей структуры TBSCertList содержится в [3].

Поле signature структуры TBSCertList задается структурой AlgorithmIdentifier в соответствии с 5.1.1 и должно совпадать со значением поля signatureAlgorithm структуры CertificateList.

4.3.2 Поле signatureAlgorithm структуры CertificateList

Поле signatureAlgorithm структуры CertificateList задается структурой AlgorithmIdentifier, описанной в 5.1.1, и содержит информацию об алгоритме подписи, который использовался при формировании поля signature структуры CertificateList (см. 4.3.3). При этом используемый алгоритм подписи должен соответствовать открытому ключу издателя списка отозванных сертификатов, имя которого указано в поле issuer структуры TBSCertList (см. 4.3.1).

4.3.3 Поле signatureValue структуры CertificateList

Поле signatureValue структуры CertificateList содержит в себе подпись значения поля tbsCertList структуры CertificateList, закодированного в формате DER. Значение подписи формируется в соответствии с 5.1.2 по алгоритму, указанному в поле signatureAlgorithm структуры CertificateList, на закрытом ключе издателя списка отозванных сертификатов, имя которого указано в поле issuer структуры TBSCertList (см. 4.3.1).

5 Порядок использования российских алгоритмов

5.1 Подпись ГОСТ Р 34.10—2012

5.1.1 Идентификатор и параметры (структура AlgorithmIdentifier)

Поле signatureAlgorithm структуры CertificationRequest (см. 4.1.2), поле signature структуры TBSCertificate (см. 4.2.1), поле signatureAlgorithm структуры Certificate (см. 4.2.2), поле signature структуры TBSCertList (см. 4.3.1), поле signatureAlgorithm структуры CertificateList (см. 4.3.2), задается структурой AlgorithmIdentifier, (см. раздел 8 ГОСТ Р ИСО/МЭК 9594-8—98 и [2]):

```
AlgorithmIdentifier ::= SEQUENCE
{
    algorithm   OBJECT IDENTIFIER
    parameters  ANY DEFINED BY algorithm OPTIONAL
}
```

Поля структуры имеют следующие значения:

algorithm — идентификатор алгоритма подписи, задающийся в соответствии с 5.1.1.1;

parameters — параметры открытого ключа алгоритма подписи, задающиеся в соответствии с 5.1.1.2.

5.1.1.1 Поле algorithm структуры AlgorithmIdentifier

Поле algorithm структуры AlgorithmIdentifier содержит идентификатор алгоритма подписи и связанных с ним параметров, который используется при формировании подписи данных:

- Для алгоритма электронной подписи ГОСТ Р 34.10—2012 с длиной ключа 256 бит и алгоритма хэширования ГОСТ Р 34.11—2012 с длиной выхода 256 бит идентификатор в формате АСН.1 задается следующим образом:

id-tc26-signwithdigest-gost3410-12-256, «1.2.643.7.1.1.3.2»;

- Для алгоритма электронной подписи ГОСТ Р 34.10—2012 с длиной ключа 512 бит и алгоритма хэширования ГОСТ Р 34.11—2012 с длиной выхода 512 бит идентификатор в формате АСН.1 задается следующим образом:

id-tc26-signwithdigest-gost3410-12-512, «1.2.643.7.1.1.3.3».

5.1.1.2 Поле parameters структуры AlgorithmIdentifier

Поле parameters должно отсутствовать. Значение параметров наследуется из соответствующих полей сертификата издателя.

5.1.2 Значение подписи

Поле signature структуры CertificationRequest (см. 4.1.3), поле signatureValue структуры Certificate (см. 4.2.3), поле signatureValue структуры CertificateList (см. 4.3.3) содержит

в себе подпись данных, закодированных в формате DER. При этом результатом работы алгоритма подписи ГОСТ Р 34.10—2012 с длиной ключа 256 бит и 512 бит являются два числа *r* и *s*, определенные в ГОСТ Р 34.10—2012 и имеющие длину 256 бит и 512 бит каждый соответственно.

При использовании алгоритма ГОСТ Р 34.10—2012 с длиной ключа 256 бит поле `signature` задается битовой строкой (BIT STRING) длины 512 бит; при этом первые 256 бит содержат число *s* в представлении big-endian (старший бит записывается первым), а вторые 256 бит содержат число *r* в представлении big-endian.

При использовании алгоритма ГОСТ Р 34.10—2012 с длиной ключа 512 бит поле `signature` задается битовой строкой (BIT STRING) длины 1024 бита; при этом первые 512 бит содержат число *s* в представлении big-endian, а вторые 512 бит содержат число *r* в представлении big-endian.

5.2 Открытый ключ и его параметры (структура SubjectPublicKeyInfo)

Структура `SubjectPublicKeyInfo` (см. ГОСТ Р ИСО/МЭК 9594-8—98, раздел 8]) имеет следующий вид:

```
SubjectPublicKeyInfo ::= SEQUENCE
{
algorithm          AlgorithmIdentifier
subjectKey  BIT STRING
}
```

Поля структуры имеют следующие значения:

`algorithm` — алгоритм и набор параметров открытого ключа, задающиеся в соответствии с 5.2.1;

subjectPublicKey — открытый ключ, задающиеся в соответствии с 5.2.2.

5.2.1 Поле algorithm структуры SubjectPublicKeyInfo

Поле `algorithm` структуры `SubjectPublicKeyInfo` задается структурой `AlgorithmIdentifier`:

```
AlgorithmIdentifier ::= SEQUENCE
{
      algorithm   OBJECT IDENTIFIER
      parameters  ANY DEFINED BY algorithm OPTIONAL
}
```

Поля структуры имеют следующие значения:

`algorithm` — идентификатор алгоритма подписи, задающийся в соответствии с 5.2.1.1;

`parameters` — параметры открытого ключа алгоритма подписи, задающиеся в соответствии с 5.2.1.2.

5.2.1.1 Поле `algorithm` структуры `AlgorithmIdentifier`

Поле `algorithm` структуры `AlgorithmIdentifier` должно содержать следующий идентификатор алгоритма:

- Для алгоритма электронной подписи ГОСТ Р 34.10—2012 с длиной ключа 256 бит идентификатор в формате АСН.1 задается следующим образом:

`id-tc26-gost3410-12-256, «1.2.643.7.1.1.1.1»;`

- Для алгоритма электронной подписи ГОСТ Р 34.10—2012 с длиной ключа 512 бит идентификатор в формате АСН.1 задается следующим образом:

`id-tc26-gost3410-12-512, «1.2.643.7.1.1.1.2».`

5.2.1.2 Поле parameters структуры `AlgorithmIdentifier`

Поле `parameters` структуры `AlgorithmIdentifier` имеет структуру `GostR3410-2012-PublicKeyParameters` в формате АСН.1:

```
GostR3410-2012-PublicKeyParameters ::= SEQUENCE
{
publicKeyParamSet        OBJECT IDENTIFIER,
digestParamSet           OBJECT IDENTIFIER OPTIONAL
}
```

Поля структуры имеют следующие значения:

`publicKeyParamSet` — идентификатор параметров открытого ключа, соответствующего алгоритму подписи ГОСТ Р 34.10—2012;

`digestParamSet` — идентификатор функции хэширования ГОСТ Р 34.11—2012. Данное поле является опциональным:

Данное поле не рекомендуется включать, если используется алгоритм подписи ГОСТ Р 34.10—2012 с длиной ключа 512 бит;

Данное поле должно присутствовать, если используется алгоритм подписи ГОСТ Р 34.10—2012 с длиной ключа 256 бит при следующих значениях поля publicKeyParamSet:

id-GostR3410-2001-CryptoPro-A-ParamSet, «1.2.643.2.2.35.1»;
id-GostR3410-2001-CryptoPro-B-ParamSet, «1.2.643.2.2.35.2»;
id-GostR3410-2001-CryptoPro-C-ParamSet, «1.2.643.2.2.35.3»;
id-GostR3410-2001-CryptoPro-XchA-ParamSet, «1.2.643.2.2.36.0»;
id-GostR3410-2001-CryptoPro-XchB-ParamSet, «1.2.643.2.2.36.1».

При этом идентификатор digestParamSet должен быть равен идентификатору алгоритма хэширования ГОСТ Р 34.11—2012 с длиной выхода 256 бит: id-tc26-gost3411-12-256, «1.2.643.7.1.1.2.2»;

Данное поле не рекомендуется включать, если используется алгоритм подписи ГОСТ Р 34.10—2012 с длиной ключа 256 бит при следующем значении поля publicKeyParamSet:

id-tc26-gost-3410-2012-256-paramSetA, «1.2.643.7.1.2.1.1.1»;

Данное поле должно отсутствовать, если используется алгоритм подписи ГОСТ Р 34.10—2012 с длиной ключа 256 бит при следующих значениях поля publicKeyParamSet:

id-tc26-gost-3410-2012-256-paramSetB, «1.2.643.7.1.2.1.1.2»;
id-tc26-gost-3410-2012-256-paramSetC, «1.2.643.7.1.2.1.1.3»;
id-tc26-gost-3410-2012-256-paramSetD, «1.2.643.7.1.2.1.1.4».

5.2.2 Поле subjectPublicKey структуры SubjectPublicKeyInfo

Поле subjectPublicKey структуры SubjectPublicKeyInfo содержит открытый ключ, который задается парой координат (*x*, *y*), определенной в ГОСТ Р 34.10—2012, представленных в следующем формате:

- Открытый ключ, соответствующий алгоритму ГОСТ Р 34.10—2012 с длиной ключа 256 бит, имеет представление GostR3410-2012-256-PublicKey, которое задается байтовой строкой длины 64 байта, где первые 32 байта содержат представление координаты *x* в формате little-endian, а последние 32 байта содержат представление координаты *y* в формате little-endian.
- Открытый ключ, соответствующий алгоритму ГОСТ Р 34.10—2012 с длиной ключа 512 бит, имеет представление GostR3410-2012-512-PublicKey, которое задается байтовой строкой длины 128 байт, где первые 64 байта содержат представление координаты *x* в формате little-endian, а последние 64 байта содержат представление координаты *y* в формате little-endian.

Ключи проверки подписи GostR3410-2012-256-PublicKey и GostR3410-2012-512-PublicKey должны быть представлены в DER-кодировке в виде строки октетов (OCTET STRING) в соответствии с ГОСТ Р ИСО/МЭК 8825-1—2003 (раздел 8.6):

```
GostR3410-2012-256-PublicKey ::= OCTET STRING (64),
GostR3410-2012-512-PublicKey ::= OCTET STRING (128).
```

5.3 Область использования ключа

Расширение KeyUsage (см. [3], раздел 4.2.1.3) является опциональным полем структуры TBSCertificate (см. 4.2.1). Для ключей, соответствующих алгоритму подписи ГОСТ Р 34.10—2012, данное расширение может иметь следующие флаги:

```
KeyUsage ::= BIT STRING
{
      digitalSignature        (0);
      contentCommitment       (1);
      keyAgreement            (4);
      keyCertSign             (5);
      cRLSign                 (6);
      encipherOnly            (7);
      decipherOnly            (8).
}
```

При этом если ключ может использоваться для формирования ключей согласования, то флаг keyAgreement обязательно должен присутствовать в расширении KeyUsage, а флаги encipherOnly и decipherOnly могут присутствовать опционально. При этом флаги encipherOnly и decipherOnly не могут присутствовать в расширении KeyUsage одновременно.

Приложение А
(справочное)

Контрольные примеры

В данном разделе приводятся примеры запроса на сертификат PKCS #10, сертификата и списка аннулированных сертификатов с параметром алгоритма подписи ГОСТ Р 34.10—2012 с длиной ключа 256 бит.

А.1 Пример 1

А.1.1 Значения параметров

В данном примере используются следующие параметры, определенные в ГОСТ Р 34.10—2012, приложение А.1:

- Модуль эллиптической кривой;
- Коэффициенты эллиптической кривой;
- Порядок группы точек эллиптической кривой;
- Порядок циклической подгруппы точек эллиптической кривой;
- Координаты базовой точки эллиптической кривой.

В качестве идентификатора приведенных выше параметров используется значение «1.2.643.2.2.35.0». В качестве ключа подписи используется ключ, определенный в ГОСТ Р 34.10—2012, приложение А.1.1.6:

$d = 7A929ADE789BB9BE10ED359DD39A72C11B60961F49397EEE1D19CE9891EC3B28_{16}$

В качестве открытого ключа используется ключ проверки подписи, определенный в ГОСТ Р 34.10—2012, приложение А.1.1.7:

$X_q = 7F2B49E270DB6D90D8595BEC458B50C58585BA1D4E9B788F6689DBD8E56FD80B_{16}$

$Y_q = 26F1B489D6701DD185C8413A977B3CBBAF64D1C593D26627DFFB101A87FF77DA_{16}$

В качестве случайного числа для подписи используется *k*, определенный в ГОСТ Р 34.10—2012, приложение А.1.2:

$k = 77105C9B20BCD3122823C8CF6FCC7B956DE33814E95B7FE64FED924594DCEAB316$

А.1.2 Запрос на сертификат

А.1.2.1 Запрос на сертификат в кодировке BASE64

```
MIHTMIGBAgEAMBIxEDAOBgNVBAMTB0V4YW1wbGUwZjAfBggqhQMHAQEBATATBgcq
hQMCAiMABggqhQMHAQECAgNDAARAC9hv5djbiWaPeJtOHbqFhcVQi0XsW1nYkG3b
cOJJK3/ad/+HGhD73ydm0pPF0WSvuzx7lzpByIXRHXDWibTxJqAAMAoGCCqFAwcB
AQMCA0EAaqqzjjXUqqUXlAMBeZEi2FVIT1efTLuW1jzf3zrMQypBqijS8asUgoDN
ntVv7aQZdAU1VKQnZ7g60EP9OdwEkw==
```

А.1.2.2 ACH.1 представление запроса на сертификат

```
  0  211: SEQUENCE {
  3  129:   SEQUENCE {
  5    1:     INTEGER 0
  8   18:     SEQUENCE {
 10   16:       SET {
 12   14:         SEQUENCE {
 14    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
 19    7:           PrintableString 'Example'
       :           }
       :         }
       :       }
 28  102:     SEQUENCE {
 30   31:       SEQUENCE {
 32    8:         OBJECT IDENTIFIER
       :           id-tc26-gost3410-12-256 (1.2.643.7.1.1.1.1)
 42   19:         SEQUENCE {
 44    7:           OBJECT IDENTIFIER
       :             id-GostR3410-2001-TestParamSet (1 2 643 2 2 35 0)
 53    8:           OBJECT IDENTIFIER
       :             id-tc26-gost3411-12-256 (1.2.643.7.1.1.2.2)
       :           }
       :         }
 63   67:       BIT STRING
 66   64:         OCTET STRING
       :           0B D8 6F E5 D8 DB 89 66 8F 78 9B 4E 1D BA 85 85
       :           C5 50 8B 45 EC 5B 59 D8 90 6D DB 70 E2 49 2B 7F
```

```
         :           DA 77 FF 87 1A 10 FB DF 27 66 D2 93 C5 D1 64 AF
         :           BB 3C 7B 97 3A 41 C8 85 D1 1D 70 D6 89 B4 F1 26
         :         }
 133      0:       CONTEXT SPECIFIC (0){
          :         }
         :       }
 135    10:     SEQUENCE {
 137     8:       OBJECT IDENTIFIER
         :         id-tc26-signwithdigest-gost3410-12-256 (1.2.643.7.1.1.3.2)
         :       }
 147    65:     BIT STRING
         :       6A AA B3 8E 35 D4 AA A5 17 94 03 01 79 91 22 D8
         :       55 48 4F 57 9F 4C BB 96 D6 3C DF DF 3A CC 43 2A
         :       41 AA 28 D2 F1 AB 14 82 80 CD 9E D5 6F ED A4 19
         :       74 05 35 54 A4 27 67 B8 3A D0 43 FD 39 DC 04 93
         :     }
```

А.1.3 Сертификат

А.1.3.1 Сертификат в кодировке BASE64

```
MIIBGDCBxqADAgECAgEKMAoGCCqFAwcBAQMCMBIxEDAOBgNVBAMTB0V4YW1wbGUwIBcNMDEwMTAxMDAwMDAwWhgPM
jA1MDEyMzEwMDAwMDBaMBIxEDAOBgNVBAMTB0V4YW1wbGUwZjAfBggqhQMHAQEBATATBgcqhQMCAiMABggqhQMHAQ
ECAgNDAARAC9hv5djbiWaPeJtOHbqFhcVQi0XsW1nYkG3bcOJJK3/ad/+HGhD73ydm0pPF0WSvuzx7lzpByIXRHXD
WibTxJjAKBggqhQMHAQEDAgNBAEOGDlxBQFcTPHxIEpISzpCT8mass1FbiDokJqzGC2u+Qaoo0vGrFIKAzZ7Vb+2k
GXQFNVSkJ2e4OtBD/TncBJM=
```

А.1.3.2 Сертификат в АСН.1 представлении

```
   0  280: SEQUENCE {
   4  198:   SEQUENCE {
   7    3:     [0] {
   9    1:       INTEGER 2
        :       }
  12    1:     INTEGER 10
  15   10:     SEQUENCE {
  17    8:       OBJECT IDENTIFIER
        :         id-tc26-signwithdigest-gost3410-12-256 (1.2.643.7.1.1.3.2)
        :       }
  27   18:     SEQUENCE {
  29   16:       SET {
  31   14:         SEQUENCE {
  33    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
  38    7:           PrintableString 'Example'
        :           }
        :         }
        :       }
  47   32:     SEQUENCE {
  49   13:       UTCTime 01/01/2001 00:00:00 GMT
  64   15:       GeneralizedTime 31/12/2050 00:00:00 GMT
        :       }
  81   18:     SEQUENCE {
  83   16:       SET {
  85   14:         SEQUENCE {
  87    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
  92    7:           PrintableString 'Example'
        :           }
        :         }
        :       }
 101  102:     SEQUENCE {
 103   31:       SEQUENCE {
 105    8:         OBJECT IDENTIFIER
        :           id-tc26-gost3410-12-256 (1.2.643.7.1.1.1.1)
 115   19:         SEQUENCE {
 117    7:           OBJECT IDENTIFIER
        :             id-GostR3410-2001-TestParamSet (1 2 643 2 2 35 0)
```

```
126    8:           OBJECT IDENTIFIER
       :             id-tc26-gost3411-12-256 (1.2.643.7.1.1.2.2)
       :           }
       :         }
136   67:       BIT STRING
139   64:         OCTET STRING
       :           0B D8 6F E5 D8 DB 89 66 8F 78 9B 4E 1D BA 85 85
       :           C5 50 8B 45 EC 5B 59 D8 90 6D DB 70 E2 49 2B 7F
       :           DA 77 FF 87 1A 10 FB DF 27 66 D2 93 C5 D1 64 AF
       :           BB 3C 7B 97 3A 41 C8 85 D1 1D 70 D6 89 B4 F1 26
       :       }
       :     }
205   10:   SEQUENCE {
207    8:     OBJECT IDENTIFIER
       :       id-tc26-signwithdigest-gost3410-12-256 (1.2.643.7.1.1.3.2)
       :     }
217   65:   BIT STRING
       :     0F D2 33 A6 C5 10 4A 94 BA 81 58 0C D0 C1 52 77
       :     39 45 E6 DA 1B EA 06 84 3B 02 29 C7 2D 08 B6 7F
       :     41 AA 28 D2 F1 AB 14 82 80 CD 9E D5 6F ED A4 19
       :     74 05 35 54 A4 27 67 B8 3A D0 43 FD 39 DC 04 93
       :   }
```

A.1.4 Список аннулированных сертификатов

A.1.4.1 Список аннулированных сертификатов в кодировке BASE64

```
MIGSMEECAQEwCgYIKoUDBwEBAwIwEjEQMA4GA1UEAxMHRXhhbXBsZRcNMTQwMTAxMDAwMDAwWhcNMTQwMTAyMDAwM
DAwWjAKBggqhQMHAQEDAgNBAEK/OSoU0+vpV68+RstQv19CIaADrT0XJ1PJSpw3ox0gQaoo0vGrFIKAzZ7Vb+2kGX
QFNVSkJ2e4OtBD/TncBJM=
```

A.1.4.2 ACH.1 представление списка аннулированных сертификатов

```
  0  146: SEQUENCE {
  3   65:   SEQUENCE {
  5    1:     INTEGER 1
  8   10:     SEQUENCE {
 10    8:       OBJECT IDENTIFIER
       :         id-tc26-signwithdigest-gost3410-12-256 (1.2.643.7.1.1.3.2)
       :       }
 20   18:     SEQUENCE {
 22   16:       SET {
 24   14:         SEQUENCE {
 26    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
 31    7:           PrintableString 'Example'
       :           }
       :         }
       :       }
 40   13:     UTCTime 01/01/2014 00:00:00 GMT
 55   13:     UTCTime 02/01/2014 00:00:00 GMT
       :     }
 70   10:   SEQUENCE {
 72    8:     OBJECT IDENTIFIER
       :       id-tc26-signwithdigest-gost3410-12-256 (1.2.643.7.1.1.3.2)
       :     }
 82   65:   BIT STRING
       :       42 BF 39 2A 14 D3 EB E9 57 AF 3E 46 CB 50 BF 5F
       :       42 21 A0 03 AD 3D 17 27 53 C9 4A 9C 37 A3 1D 20
       :       41 AA 28 D2 F1 AB 14 82 80 CD 9E D5 6F ED A4 19
       :       74 05 35 54 A4 27 67 B8 3A D0 43 FD 39 DC 04 93
       :   }
```

A.2 Пример 2

В данном разделе приводятся примеры запроса на сертификат PKCS #10, сертификата и списка аннулированных сертификатов с параметром алгоритма подписи ГОСТ Р 34.10—2012 «1.2.643.7.1.2.1.1.1» с ключом подписи длины 256 бит для скрученной эллиптической кривой Эдвардса.

А.2.1 Значения параметров

В данном примере используются следующие параметры:

- Модуль эллиптической кривой;
- Коэффициенты эллиптической кривой;
- Порядок группы точек эллиптической кривой;
- Порядок циклической подгруппы точек эллиптической кривой;
- Координаты базовой точки эллиптической кривой.

В качестве идентификатора приведенных выше параметров используется значение «1.2.643.7.1.2.1.1.1» из Р 50.1.114—2016, приложение А.3. В качестве ключа подписи используется ключ:

```
d = 7A929ADE789BB9BE10ED359DD39A72C11B60961F49397EEE1D19CE9891EC3B2816
```

В качестве ключа проверки подписи используется ключ, соответствующий выбранному ключу подписи:

```
Xq = DCB8EA3CDDF95C6D77A00BCE47E31354D156D9D0C54A2240403E9859C1F4422B16
Yq = 568C1E8197EF5270E24AB0214C6F6F9CA4FDEFD2C154DB7ACFC9A4CE3A4888AB16
```

В качестве случайного числа для подписи используется следующее значение *k*:

```
k = 27105C9B20BCD3122823C8CF6FCC7B956DE33814E95B7FE64FED924594DCEAB316
```

А.2.2 Запрос на сертификат

А.2.2.1 Запрос на сертификат в кодировке BASE64

```
MIHKMHkCAQAwEjEQMA4GA1UEAxMHRXhhbXBsZTBeMBcGCCqFAwcBAQEBMAsGCSqF
AwcBAgEBAQNDAARAK0L0wVmYPkBAIkrF0NlW0VQT40fOC6B3bVz53TzquNyriEg6
zqTJz3rbVMHS7/2knG9vTCGwSuJwUu+XgR6MVqAAMAoGCCqFAwcBAQMCA0EADrt8
U1Trnk0Fi+9PQ7w6BN/qoqc8+LPJktdNvXlwEpsdDh2lvjR8bxtSVseurCAK1krH
em9bOg4Jcxjnrm7naQ==
```

А.2.2.2 АСН.1 представление запроса на сертификат

```
   0  202: SEQUENCE {
   3  121:   SEQUENCE {
   5    1:     INTEGER 0
   8   18:     SEQUENCE {
  10   16:       SET {
  12   14:         SEQUENCE {
  14    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
  19    7:           PrintableString 'Example'
         :           }
         :         }
         :       }
  28   94:     SEQUENCE {
  30   23:       SEQUENCE {
  32    8:         OBJECT IDENTIFIER
         :           id-tc26-gost3410-12-256 (1.2.643.7.1.1.1.1)
  42   11:         SEQUENCE {
  44    9:           OBJECT IDENTIFIER
         :             id-tc26-gost-3410-2012-256-paramSetA (1.2.643.7.1.2.1.1.1)
         :           }
         :         }
  55   67:       BIT STRING
  58   64:         OCTET STRING
         :           2B 42 F4 C1 59 98 3E 40 40 22 4A C5 D0 D9 56 D1
         :           54 13 E3 47 CE 0B A0 77 6D 5C F9 DD 3C EA B8 DC
         :           AB 88 48 3A CE A4 C9 CF 7A DB 54 C1 D2 EF FD A4
         :           9C 6F 6F 4C 21 B0 4A E2 70 52 EF 97 81 1E 8C 56
         :       }
 124    0:       CONTEXT SPECIFIC (0){
         :       }
         :     }
 126   10:   SEQUENCE {
 128    8:     OBJECT IDENTIFIER
         :       id-tc26-signwithdigest-gost3410-12-256 (1.2.643.7.1.1.3.2)
         :     }
 138   65:   BIT STRING
         :     0E BB 7C 53 54 EB 9E 4D 05 8B EF 4F 43 BC 3A 04
         :     DF EA A2 A7 3C F8 B3 C9 92 D7 4D BD 79 70 12 9B
         :     1D 0E 1D A5 BE 34 7C 6F 1B 52 56 C7 AE AC 20 0A
```

```
        :     D6 4A C7 7A 6F 5B 3A 0E 09 73 18 E7 AE 6E E7 69
        :   }
```

А.2.3 Сертификат

А.2.3.1 Сертификат в кодировке BASE64

```
MIIBEDCBvqADAgECAgEKMAoGCCqFAwcBAQMCMBIxEDAOBgNVBAMTB0V4YW1wbGUwIBcNMDEwMTAxMDAwMDAwWhgPM
jA1MDEyMzEwMDAwMDBaMBIxEDAOBgNVBAMTB0V4YW1wbGUwXjAXBggqhQMHAQEBATALBgkqhQMHAQIBAQEDQwAEQA
vYb+XY2Ilmj3ibTh26hYXFUItF7FtZ2JBt23DiSSt/2nf/hxoQ+98nZtKTxdFkr7s8e5c6QciF0R1w1om08SYwCgY
IKoUDBwEBAwIDQQAdDh2lvjR8bxtSVseurCAK1krHem9bOg4Jcxjnrm7naTePiX/cZ4wlywyMeJAfhcCxsSnRTiLn
3QOJXcqW38rB
```

А.2.3.2 Сертификат в АСН.1 представлении

```
  0  272: SEQUENCE {
  4  190:   SEQUENCE {
  7    3:     [0] {
  9    1:       INTEGER 2
       :       }
 12    1:     INTEGER 10
 15   10:     SEQUENCE {
 17    8:       OBJECT IDENTIFIER
       :         id-tc26-signwithdigest-gost3410-12-256 (1.2.643.7.1.1.3.2)
       :       }
 27   18:     SEQUENCE {
 29   16:       SET {
 31   14:         SEQUENCE {
 33    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
 38    7:           PrintableString 'Example'
       :           }
       :         }
       :       }
 47   32:     SEQUENCE {
 49   13:       UTCTime 01/01/2001 00:00:00 GMT
 64   15:       GeneralizedTime 31/12/2050 00:00:00 GMT
       :       }
 81   18:     SEQUENCE {
 83   16:       SET {
 85   14:         SEQUENCE {
 87    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
 92    7:           PrintableString 'Example'
       :           }
       :         }
       :       }
101   94:     SEQUENCE {
103   23:       SEQUENCE {
105    8:         OBJECT IDENTIFIER
       :           id-tc26-gost3410-12-256 (1.2.643.7.1.1.1.1)
115   11:         SEQUENCE {
117    9:           OBJECT IDENTIFIER
       :             id-tc26-gost-3410-2012-256-paramSetA (1.2.643.7.1.2.1.1.1)
       :           }
       :         }
128   67:       BIT STRING
131   64:         OCTET STRING
       :           0B D8 6F E5 D8 D8 89 66 8F 78 9B 4E 1D BA 85 85
       :           C5 50 8B 45 EC 5B 59 D8 90 6D DB 70 E2 49 2B 7F
       :           DA 77 FF 87 1A 10 FB DF 27 66 D2 93 C5 D1 64 AF
       :           BB 3C 7B 97 3A 41 C8 85 D1 1D 70 D6 89 B4 F1 26
       :       }
       :     }
197   10:   SEQUENCE {
199    8:     OBJECT IDENTIFIER
       :       id-tc26-signwithdigest-gost3410-12-256 (1.2.643.7.1.1.3.2)
       :     }
209   65:   BIT STRING
```

```
        :      37 8F 89 7F DC 67 8C 25 CB 0C 8C 78 90 1F 85 C0
        :      B1 B1 29 D1 4E 22 E7 DD 03 89 5D CA 96 DF CA C1
        :      1D 0E 1D A5 BE 34 7C 6F 1B 52 56 C7 AE AC 20 0A
        :      D6 4A C7 7A 6F 5B 3A 0E 09 73 18 E7 AE 6E E7 69
        :    }
```

А.2.4 Список аннулированных сертификатов

А.2.4.1 Список аннулированных сертификатов в 16-ричном формате

```
MIGSMEECAQEwCgYIKoUDBwEBAwIwEjEQMA4GA1UEAxMHRXhhbXBsZRcNMTQwMTAxMDAwMDAwWhcNMTQwMTAyMDAwM
DAwWjAKBggqhQMHAQEDAgNBABS9aAh8O5A8eqKLB/6y571v4JY/VjJnNZ9c2Oq0UFmtHQ4dpb40fG8bUlbHrqwgCt
ZKx3pvWzoOCXMY565u52k=
```

А.2.4.2 ACH.1 представление списка аннулированных сертификатов

```
  0  146: SEQUENCE {
  3   65:   SEQUENCE {
  5    1:     INTEGER 1
  8   10:     SEQUENCE {
 10    8:       OBJECT IDENTIFIER
        :         id-tc26-signwithdigest-gost3410-12-256 (1.2.643.7.1.1.3.2)
        :       }
 20   18:     SEQUENCE {
 22   16:       SET {
 24   14:         SEQUENCE {
 26    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
 31    7:           PrintableString 'Example'
        :           }
        :         }
        :       }
 40   13:     UTCTime 01/01/2014 00:00:00 GMT
 55   13:     UTCTime 02/01/2014 00:00:00 GMT
        :     }
 70   10:   SEQUENCE {
 72    8:     OBJECT IDENTIFIER
        :       id-tc26-signwithdigest-gost3410-12-256 (1.2.643.7.1.1.3.2)
        :     }
 82   65:   BIT STRING
        :       14 BD 68 08 7C 3B 90 3C 7A A2 8B 07 FE B2 E7 BD
        :       6F E0 96 3F 56 32 67 35 9F 5C D8 EA B4 50 59 AD
        :       1D 0E 1D A5 BE 34 7C 6F 1B 52 56 C7 AE AC 20 0A
        :       D6 4A C7 7A 6F 5B 3A 0E 09 73 18 E7 AE 6E E7 69
        :   }
```

А.3 Пример 3

В данном разделе приводятся примеры запроса на сертификат PKCS #10, сертификата и списка аннулированных сертификатов с параметром алгоритма подписи ГОСТ Р 34.10—2012 «1.2.643.7.1.2.1.2.0» с ключом подписи длины 512 бит.

А.3.1 Значения параметров

В данном примере используются следующие параметры:

- Модуль эллиптической кривой;
- Коэффициенты эллиптической кривой;
- Порядок группы точек эллиптической кривой;
- Порядок циклической подгруппы точек эллиптической кривой;
- Координаты базовой точки эллиптической кривой.

В качестве идентификатора приведенных выше параметров используется значение «1.2.643.7.1.2.1.2.0» из ГОСТ Р 34.10—2012, приложение А.2. В качестве ключа подписи используется ключ, определенный в ГОСТ Р 34.10—2012, приложение А.2.1.6:

```
d = 0BA6048AADAE241BA40936D47756D7C93091A0E8514669700EE7508E508B1020\\
72E8123B2200A0563322DAD2827E2714A2636B7BFD18AADFC62967821FA18DD4_16
```

В качестве ключа проверки подписи используется ключ, определенный в ГОСТ Р 34.10—2012, приложение А.2.1.7:

```
X_q = 115DC5BC96760C7B48598D8AB9E740D4C4A85A65BE33C1815B5C320C854621DD\\
5A515856D13314AF69BC5B924C8B4DDFF75C45415C1D9DD9DD33612CD530EFE1_16
Y_q = 37C7C90CD40B0F5621DC3AC1B751CFA0E2634FA0503B3D52639F5D7FB72AFD61\\
EA199441D943FFE7F0C70A2759A3CDB84C114E1F9339FDF27F35ECA93677BEEC_16
```

В качестве случайного числа при подписи используется *k*, определенный в ГОСТ Р 34.10—2012, приложение А.2.2:

```
k = 0359E7F4B1410FEACC570456C6801496946312120B39D019D455986E364F3658\\
86748ED7A44B3E794434006011842286212273A6D14CF70EA3AF71BB1AE679F1
```
$_{16}$

А.3.2 Запрос на сертификат

А.3.2.1 Запрос на сертификат в кодировке **BASE64**

```
MIIBTzCBvAIBADASMRAwDgYDVQQDEwdFeGFtcGxlMIGgMBcGCCqFAwcBAQECMAsG
CSqFAwcBAgECAAOBhAAEgYDh7zDVLGEz3dmdHVxBRVz3302LTJJbvGmvFDPRVlhR
Wt0hRoUMMlxbgcEzvmVaqMTUQOe5io1ZSHsMdpa8xV0R7L53NqnsNX/y/TmTH04R
TLjNo1knCsfw5/9D2UGUGeph/Sq3f12fY1I9O1CgT2PioM9Rt8E63CFWDwvUDMnH
N6AAMAoGCCqFAwcBAQMDA4GBAEM7HWzkClHx5XN+sWqixoOCmkBbnZEn4hJg/J1q
wF2HvyTibEUnilwhkqdbqUmTq9YHTn/xvwP9L1OXr6HZRVgvhvpgoIEJGiPdeV4e
PGie5RKjyC7g3MJkPHjuqPys01SSVYSGsg8cnsGXyQaZhQJgyTvLzZxcMxfhk0Th
c642
```

А.3.2.2 ACH.1 представление запроса на сертификат

```
   0  335: SEQUENCE {
   4  188:   SEQUENCE {
   7    1:     INTEGER 0
  10   18:     SEQUENCE {
  12   16:       SET {
  14   14:         SEQUENCE {
  16    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
  21    7:           PrintableString 'Example'
       :           }
       :         }
       :       }
  30  160:     SEQUENCE {
  33   23:       SEQUENCE {
  35    8:         OBJECT IDENTIFIER
       :           id-tc26-gost3410-12-512 (1.2.643.7.1.1.1.2)
  45   11:         SEQUENCE {
  47    9:           OBJECT IDENTIFIER
       :             id-tc26-gost-3410-12-512-paramSetTest (1.2.643.7.1.2.1.2.0)
       :           }
       :         }
  58  132:       BIT STRING
  62  128:         OCTET STRING
       :           E1 EF 30 D5 2C 61 33 DD D9 9D 1D 5C 41 45 5C F7
       :           DF 4D 8B 4C 92 5B BC 69 AF 14 33 D1 56 58 51 5A
       :           DD 21 46 85 0C 32 5C 5B 81 C1 33 BE 65 5A A8 C4
       :           D4 40 E7 B9 8A 8D 59 48 7B 0C 76 96 BC C5 5D 11
       :           EC BE 77 36 A9 EC 35 7F F2 FD 39 93 1F 4E 11 4C
       :           B8 CD A3 59 27 0A C7 F0 E7 FF 43 D9 41 94 19 EA
       :           61 FD 2A B7 7F 5D 9F 63 52 3D 3B 50 A0 4F 63 E2
       :           A0 CF 51 B7 C1 3A DC 21 56 0F 0B D4 0C C9 C7 37
       :       }
 195    0:      CONTEXT SPECIFIC (0){
       :       }
       :     }
 195   10:   SEQUENCE {
 197    8:     OBJECT IDENTIFIER
       :       id-tc26-signwithdigest-gost3410-12-512 (1.2.643.7.1.1.3.3)
       :     }
 207  129:   BIT STRING
       :     43 3B 1D 6C E4 0A 51 F1 E5 73 7E B1 6A A2 C6 83
       :     82 9A 40 5B 9D 91 27 E2 12 60 FC 9D 6A C0 5D 87
       :     BF 24 E2 6C 45 27 8A 5C 21 92 A7 5B A9 49 93 AB
       :     D6 07 4E 7F F1 BF 03 FD 2F 53 97 AF A1 D9 45 58
       :     2F 86 FA 60 A0 81 09 1A 23 DD 79 5E 1E 3C 68 9E
       :     E5 12 A3 C8 2E E0 DC C2 64 3C 78 EE A8 FC AC D3
       :     54 92 55 84 86 B2 0F 1C 9E C1 97 C9 06 99 85 02
       :     60 C9 3B CB CD 9C 5C 33 17 E1 93 44 E1 73 AE 36
       :   }
```

А.3.3 Сертификат

А.3.3.1 Сертификат в кодировке **BASE64**

```
MIIBlTCCAQGgAwIBAgIBCzAKBggqhQMHAQEDAzASMRAwDgYDVQQDEwdFeGFtcGxlMCAXDTAxMDEwMTAwMDAwMFoYD
zIwNTAxMjMxMDAwMDAwWjASMRAwDgYDVQQDEwdFeGFtcGxlMIGgMBcGCCqFAwcBAQECMAsGCSqFAwcBAgECAAOBhA
AEgYDh7zDVLGEz3dmdHVxBRVz3302LTJJbvGmvFDPRVlhRWt0hRoUMMlxbgcEzvmVaqMTUQOe5io1ZSHsMdpa8xV0
R7L53NqnsNX/y/TmTH04RTLjNo1knCsfw5/9D2UGUGeph/Sq3f12fY1I9O1CgT2PioM9Rt8E63CFWDwvUDMnHNzAK
BggqhQMHAQEDAwOBgQA9Vd5th8PV+E7oRo6Oiy2W4DQS5+enYn2gg+fFdnQm22r2MVNyGRUyCj2h+aNGM7r87Q72B
PcvG+UvN6IYEpJsL4b6YKCBCRoj3XleHjxonuUSo8gu4NzCZDx47qj8rNNUklWEhrIPHJ7Bl8kGmYUCYMk7y82cXD
MX4ZNE4XOuNg==
```

А.3.3.2 ACH.1 представление сертификата

```
   0  405: SEQUENCE {
   4  257:   SEQUENCE {
   8    3:     [0] {
  10    1:       INTEGER 2
        :       }
  13    1:     INTEGER 11
  16   10:     SEQUENCE {
  18    8:       OBJECT IDENTIFIER
        :         id-tc26-signwithdigest-gost3410-12-512 (1.2.643.7.1.1.3.3)
        :       }
  28   18:     SEQUENCE {
  30   16:       SET {
  32   14:         SEQUENCE {
  34    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
  39    7:           PrintableString 'Example'
        :           }
        :         }
        :       }
  48   32:     SEQUENCE {
  50   13:       UTCTime 01/01/2001 00:00:00 GMT
  65   15:       GeneralizedTime 31/12/2050 00:00:00 GMT
        :       }
  82   18:     SEQUENCE {
  84   16:       SET {
  86   14:         SEQUENCE {
  88    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
  93    7:           PrintableString 'Example'
        :           }
        :         }
        :       }
 102  160:     SEQUENCE {
 105   23:       SEQUENCE {
 107    8:         OBJECT IDENTIFIER
        :           id-tc26-gost3410-12-512 (1.2.643.7.1.1.1.2)
 117   11:         SEQUENCE {
 119    9:           OBJECT IDENTIFIER
        :             id-tc26-gost-3410-12-512-paramSetTest (1.2.643.7.1.2.1.2.0)
        :           }
        :         }
 130  132:       BIT STRING
 134  128:         OCTET STRING
        :           E1 EF 30 D5 2C 61 33 DD D9 9D 1D 5C 41 45 5C F7
        :           DF 4D 8B 4C 92 5B BC 69 AF 14 33 D1 56 58 51 5A
        :           DD 21 46 85 0C 32 5C 5B 81 C1 33 BE 65 5A A8 C4
        :           D4 40 E7 B9 8A 8D 59 48 7B 0C 76 96 BC C5 5D 11
        :           EC BE 77 36 A9 EC 35 7F F2 FD 39 93 1F 4E 11 4C
        :           B8 CD A3 59 27 0A C7 F0 E7 FF 43 D9 41 94 19 EA
        :           61 FD 2A B7 7F 5D 9F 63 52 3D 3B 50 A0 4F 63 E2
        :           A0 CF 51 B7 C1 3A DC 21 56 0F 0B D4 0C C9 C7 37
        :         }
        :       }
 265   10:   SEQUENCE {
```

```
 267    8:     OBJECT IDENTIFIER
        :       id-tc26-signwithdigest-gost3410-12-512 (1.2.643.7.1.1.3.3)
        :     }
 277  129:   BIT STRING
        :     3D 55 DE 6D 87 C3 D5 F8 4E E8 46 8E 8E 8B 2D 96
        :     E0 34 12 E7 E7 A7 62 7D A0 83 E7 C5 76 74 26 DB
        :     6A F6 31 53 72 19 15 32 0A 3D A1 F9 A3 46 33 BA
        :     FC ED 0E F6 04 F7 2F 1B E5 2F 37 A2 18 12 92 6C
        :     2F 86 FA 60 A0 81 09 1A 23 DD 79 5E 1E 3C 68 9E
        :     E5 12 A3 C8 2E E0 DC C2 64 3C 78 EE A8 FC AC D3
        :     54 92 55 84 86 B2 0F 1C 9E C1 97 C9 06 99 85 02
        :     60 C9 3B CB CD 9C 5C 33 17 E1 93 44 E1 73 AE 36
        :   }
```

А.3.4 Список аннулированных сертификатов

А.3.4.1 Список аннулированных сертификатов в кодировке BASE64

```
MIHTMEECAQEwCgYIKoUDBwEBAwMwEjEQMA4GA1UEAxMHRXhhbXBsZRcNMTQwMTAxMDAwMDAwWhcNMTQwMTAyMD
AwMDAwWjAKBggqhQMHAQEDAwOBgQA6E/t67NtVYO72E3z8XdZGkXMuv7NpCh/Ax+ik7uoIMH1kjU3AmGxGqHs/
vkx69C6jQ1nHlZVMo5/zq77ZBR9NL4b6YKCBCRoj3XleHjxonuUSo8gu4NzCZDx47qj8rNNUklWEhrIPHJ7B18kGm
YUCYMk7y82cXDMX4ZNE4XOuNg==
```

А.3.4.2 ACH.1 представление списка аннулированных сертификатов

```
   0  211: SEQUENCE {
   3   65:   SEQUENCE {
   5    1:     INTEGER 1
   8   10:     SEQUENCE {
  10    8:       OBJECT IDENTIFIER
        :         id-tc26-signwithdigest-gost3410-12-256 (1.2.643.7.1.1.3.3)
        :       }
  20   18:     SEQUENCE {
  22   16:       SET {
  24   14:         SEQUENCE {
  26    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
  31    7:           PrintableString 'Example'
        :           }
        :         }
        :       }
  40   13:     UTCTime 01/01/2014 00:00:00 GMT
  55   13:     UTCTime 02/01/2014 00:00:00 GMT
        :     }
  70   10:   SEQUENCE {
  72    8:     OBJECT IDENTIFIER
        :       id-tc26-signwithdigest-gost3410-12-256 (1.2.643.7.1.1.3.3)
        :     }
  82  129:   BIT STRING
        :       3A 13 FB 7A EC DB 55 60 EE F6 13 7C FC 5D D6 46
        :       91 73 2E BF B3 69 0A 1F C0 C7 E8 A4 EE EA 08 30
        :       7D 64 8D 4D C0 98 6C 46 A8 7B 3F BE 4C 7A F4 2E
        :       A3 43 59 C7 95 95 4C A3 9F F3 AB BE D9 05 1F 4D
        :       2F 86 FA 60 A0 81 09 1A 23 DD 79 5E 1E 3C 68 9E
        :       E5 12 A3 C8 2E E0 DC C2 64 3C 78 EE A8 FC AC D3
        :       54 92 55 84 86 B2 0F 1C 9E C1 97 C9 06 99 85 02
        :       60 C9 3B CB CD 9C 5C 33 17 E1 93 44 E1 73 AE 36
        :   }
```

Библиография

[1] ITU-T Recommendation X.690 (1997) | ISO/IEC 8825-1:1998, Information Technology — ACH.1 Encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)

[2] Nystrom M., Kaliski B., PKCS #10: Certification Request Syntax Specification, version 1.7, November 2000

[3] IETF RFC 5280 Cooper D., Santesson S., Farrell S., Boeyen S., Housley R. and W. Polk, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, IETF RFC 5280, May 2008

УДК 681.3.06:006.354 ОКС 35. 040 ОКСТУ 5002 П85

Ключевые слова: криптографические протоколы, аутентификация, пароль, ключ

БЗ 1—2019/30

Редактор *Н.А. Аргунова*
Технический редактор *В.Н. Прусакова*
Корректор *И.А. Королева*
Компьютерная верстка *Е.О. Асташина*

Сдано в набор 26.12.2018. Подписано в печать 11.01.2019. Формат 60×84$^{1}/_{8}$. Гарнитура Ариал.
Усл. печ. л. 2,79. Уч.-изд. л. 2,23.
Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

Создано в единичном исполнении ФГУП «СТАНДАРТИНФОРМ» для комплектования Федерального информационного фонда стандартов, 117418 Москва, Нахимовский пр-т, д. 31, к. 2.

gost.e-ecolog.ru gost.e-ecolog.ru

gost.e-ecolog.ru