

РУКОВОДЯЩИЙ ДОКУМЕНТ РЕСПУБЛИКИ БЕЛАРУСЬ

**Банковские технологии
ФОРМАТ СЕРТИФИКАТОВ ОТКРЫТЫХ КЛЮЧЕЙ И
СПИСКОВ ОТОЗВАННЫХ СЕРТИФИКАТОВ**

**Банкаўскія тэхналогіі
ФАРМАТ СЕРТЫФІКАТАЎ АДКРЫТЫХ КЛЮЧОЎ І
СПІСКАЎ АДАЗВАННЫХ СЕРТЫФІКАТАЎ**

Издание официальное

УДК

Ключевые слова: ключ проверки подписи открытый, подпись электронная цифровая, протокол формирования общего ключа, сертификат открытого ключа

ОКС 35.240.40

Предисловие

1 РАЗРАБОТАН Государственным центром безопасности информации при Президенте Республики Беларусь, ЗАО «Авест»

2 ВНЕСЕН Управлением безопасности и защиты информации Национального банка Республики Беларусь

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ постановлением Совета директоров Национального банка Республики Беларусь от « » 2004 г. №

4 ВВЕДЕН ВПЕРВЫЕ

Настоящий руководящий документ не может быть тиражирован и распространен без разрешения Национального банка Республики Беларусь

Издан на русском языке

Содержание

1	Область применения	1
2	Нормативные ссылки	1
3	Термины и определения	1
4	Обозначения и сокращения	1
5	Способ описания форматов	2
6	Формат СОК	2
6.1	Представление СОК	2
6.2	Общая часть СОК	2
6.3	Особенная часть СОК.....	5
7	Формат СОС	6
7.1	Представление СОС	6
7.2	Общая часть СОС	6
7.3	Особенная часть СОС.....	7
8	Требования к дополнениям	7
8.1	Виды дополнений	7
8.2	Стандартные дополнения СОК	7
8.3	Стандартные дополнения СОС	8

РУКОВОДЯЩИЙ ДОКУМЕНТ РЕСПУБЛИКИ БЕЛАРУСЬ

**Банковские технологии
ФОРМАТ СЕРТИФИКАТОВ ОТКРЫТЫХ КЛЮЧЕЙ И СПИСКОВ
ОТОЗВАННЫХ СЕРТИФИКАТОВ****Банкаўскія тэхналогіі
ФАРМАТ СЕРТЫФІКАТАЎ АДКРЫТЫХ КЛЮЧОЎ І СПІСКАЎ
АДАЗВАННЫХ СЕРТЫФІКАТАЎ**

Дата введения 2004–09–01

1 Область применения

Настоящий руководящий документ распространяется на сертификаты открытых ключей и списки отозванных сертификатов, используемые при применении средств криптографической защиты информации, в том числе в технологии электронной цифровой подписи.

Настоящий руководящий документ устанавливает требования к формату сертификатов открытых ключей, сгенерированных в соответствии с криптографическими алгоритмами, установленными в нормативных документах Республики Беларусь, и списков отозванных сертификатов.

Настоящий руководящий документ применяется при создании сертификатов открытых ключей и списков отозванных сертификатов с использованием средств криптографической защиты информации, в том числе средств электронной цифровой подписи. Сертификаты, соответствующие требованиям настоящего документа, удовлетворяют требованиям нормативного документа ITU-T Recommendation X.509 (2000 E) (ISO/IEC 9594-8:2001(E)).

2 Нормативные ссылки

В настоящем стандарте использованы ссылки на следующие нормативные документы:

СТБ 1176.1-99 Информационная технология. Защита информации. Функция хэширования

СТБ 1176.2-99 Информационная технология. Защита информации. Процедуры выработки и проверки электронной цифровой подписи

ГОСТ 34.973-91 (ИСО 8824-87) Взаимосвязь открытых систем. Спецификация абстрактно-синтаксической нотации версии 1 (АСН.1)

ГОСТ 34.974 -91 (ИСО 8825-87) Взаимосвязь открытых систем. Описание базовых правил кодирования для абстрактно-синтаксической нотации версии 1 (АСН.1)

ITU-T Recommendation X.509 (2000 E) (ISO/IEC 9594-8:2001(E)) Series X: Data networks and open system communications. Directory (Information technology – Open systems interconnection – The Directory: Public-key and attribute certificate frameworks

3 Термины и определения

В настоящем руководящем документе применяют следующие термины с соответствующими определениями:

Идентификатор объекта – значение (отличающееся от других подобных значений), которое связывается с информационным объектом (по ГОСТ 34.973).

Информационный объект – некоторая точно определенная совокупность информации, определение или спецификация, использование которой в конкретном взаимодействии идентифицируется соответствующим именем (по ГОСТ 34.973).

4 Обозначения и сокращения

В настоящем руководящем документе применяют следующие обозначения и сокращения:

ИО – идентификатор объекта;

ПФОК – протокол формирования общего ключа;

СОК – сертификат открытого ключа;

СОС – список отозванных сертификатов;

УЦ – удостоверяющий центр;

ЭЦП – электронная цифровая подпись;

d – переменная, определенная в пункте 7.3.2 СТБ 1176.2;

d_0, \dots, d_t ; r_0, \dots, r_s – переменные, определенные в пункте 7.2.3 СТБ 1176.2;

h – функция хэширования, значения которой вычисляются в соответствии с алгоритмом, установленным в СТБ 1176.1 со следующими значениями параметров:

$H=4E4E9C9C\ 9C9C4E4E\ 9C9C4E4E\ 4E4E9C9C\ 9C9C4E4E\ 4E4E9C9C\ 4E4E9C9C\ 9C9C4E4E$ (в шестнадцатеричной системе счисления), $L=256$;

H – параметр, определенный в разделе 3 СТБ 1176.1;

p, q, a, l, r – параметры, определенные в разделах 4 и 7 СТБ 1176.2;

z_1, \dots, z_{31} – инициализирующее значение датчика случайных чисел, определенное в пункте 7.2.1 СТБ 1176.2.

5 Способ описания форматов

СОК и СОС являются значениями некоторых типов данных. Для определения этих типов в настоящем документе используется спецификация абстрактно-синтаксической нотации версии 1 (ASN.1), установленная в ГОСТ 34.973. Правила кодирования этих типов данных устанавливается в соответствии с ГОСТ 34.974 при соблюдении следующих ограничений:

- для представления длины значения должен использоваться явный формат с применением минимально возможного количества октетов;
- все биты двоичного представления значения «BOOLEAN TRUE» должны быть установлены в «1»;
- неиспользуемые биты в значении «BIT STRING» должны быть установлены в «0»;
- значения, совпадающие со значениями по умолчанию, не должны включаться в закодированное представление составных значений;
- значения в составе «SET OF» должны быть представлены в порядке увеличения значений их закодированных представлений. Более короткие значения перед сравнением справа должны дополняться нулями;
- представления значений «GeneralizedTime» обязательно должны завершаться символом «Z» и обязано содержать значение секунд. Дробные части секунд не должны включаться в закодированное представление.

6 Формат СОК

6.1 Представление СОК

СОК является значением типа «Certificate», определяемого следующим образом:

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

Значение «tbsCertificate» определяет общую часть СОК, как электронного документа. Значения «signatureAlgorithm» и «signatureValue» определяют особенную часть СОК, как электронного документа.

6.2 Общая часть СОК

6.2.1 Значение «tbsCertificate» типа «TBSCertificate» определяется следующим образом:

```

TBSCertificate ::= SEQUENCE {
    version          [0] EXPLICIT INTEGER {v3(2)},
    serialNumber     INTEGER,
    signature        SignAlgorithmIdentifier,
    issuer           Name,
    validity         Validity,
    subject          Name,
    subjectPublicKeyInfo SubjectPublicKeyInfo,
    extensions       [3] Extensions
}

```

Значение «tbsCertificate» должно содержать данные о физическом или юридическом лице и принадлежащем ему открытом ключе. Данные, составляющие это значение, подписываются УЦ.

6.2.2 Значение «serialNumber» должно устанавливаться для каждого сертификата УЦ, выпустившим данный сертификат. Серийный номер СОК должен быть уникальным для каждого сертификата, выпущенного данным УЦ. Значения, определяющие идентификатор УЦ, выпустившего СОК, и его серийный номер, образуют уникальную пару, однозначно идентифицирующую СОК.

6.2.3 Значение «signature» типа «SignAlgorithmIdentifier» определяется следующим образом:

```

SignAlgorithmIdentifier ::= SEQUENCE {
    algorithm        OBJECT IDENTIFIER,
    parameters       NULL
}

```

Значение «algorithm» должно устанавливаться равным значению ИО алгоритма ЭЦП, в соответствии с которым подписан данный СОК.

6.2.4 Значения «issuer» и «subject» типа «Name» определяется следующим образом:

```

Name ::= SEQUENCE OF RelativeDistinguishedName
RelativeDistinguishedName ::= SET OF AttributeTypeAndValue
AttributeTypeAndValue ::= SEQUENCE {
    type            OBJECT IDENTIFIER,
    value           AttributeValue
}
AttributeValue ::= CHOICE {
    printableString PrintableString,
    ia5String        IA5String,
    bmpString        BMPString
}

```

Значение «issuer» должно содержать данные, идентифицирующие УЦ, выпустивший этот СОК. При этом должны быть выдержаны следующие ограничения:

- если, в качестве одного из значений, необходимых для идентификации УЦ, выпустившего СОК, выступает адрес электронной почты, то его ИО должен быть равен значению «1.2.840.113549.1.9.1», а значение адреса электронной почты должно иметь тип «IA5String»;
- не допускается использование более одного значения типа «AttributeTypeAndValue» в составе одного значения типа «RelativeDistinguishedName»;
- не допускается использование двух значений типа «AttributeTypeAndValue» с одинаковыми значениями «type».

Значение «subject» должно содержать данные, идентифицирующие физическое или юридическое лицо, которому принадлежит открытый ключ.

6.2.5 Значение «validity» типа «Validity» определяется следующим образом:

```

Validity ::= SEQUENCE {
    notBefore        GeneralizedTime,
    notAfter         GeneralizedTime
}

```

Значение «validity» должно содержать дату и время начала действия сертификата (в значении «notBefore») и дату и время окончания срока действия сертификата (в значении «notAfter»). Данное значение определяет срок действия сертификата открытого ключа.

6.2.6 Значение «subjectPublicKeyInfo» типа «SubjectPublicKeyInfo» определяется следующим образом:

```

SubjectPublicKeyInfo ::= SEQUENCE {
    algorithm      PublicKeyAlgorithmIdentifier,
    subjectPublicKey BIT STRING
}
PublicKeyAlgorithmIdentifier ::= SEQUENCE {
    algorithmId      OBJECT IDENTIFIER,
    params           PublicKeyParameters
}
PublicKeyParameters ::= CHOICE {
    bdsParams      [0] EXPLICIT BDSParams
    cskParams      [1] EXPLICIT CSKParams
}
BDSParams ::= CHOICE {
    bdsParamsReference OBJECT IDENTIFIER,
    bdsParamsList      BDSParamsListType
}
CSKParams ::= CHOICE {
    cskParamsReference OBJECT IDENTIFIER,
    cskParamsList      CSKParamsListType
}

```

Значение «subjectPublicKey» должно содержать последовательность бит, определяющую открытый ключ, представленный в двоичной системе счисления, дополненный до требуемой длины слева незначащими нулевыми битами. Первый бит последовательности должен представлять старший бит двоичного представления открытого ключа. Последний бит последовательности должен представлять младший бит двоичного представления открытого ключа.

Значение «algorithmId» должно устанавливаться равным значению ИО алгоритма, в соответствии с которым сгенерирован приведенный в данном сертификате открытый ключ.

Значение «params» должно определять набор значений параметров алгоритма ЭЦП или ПФОК, в соответствии с которыми вычислен открытый ключ. Это значение определяется либо при помощи ссылки на множество значений параметров с использованием значений «bdsParamsReference» (для параметров алгоритма ЭЦП) и «cskParamsReference» (для параметров алгоритма ПФОК), либо при помощи явного указания значений параметров с использованием значений «bdsParamsList» (для параметров алгоритма ЭЦП) и «cskParamsList» (для параметров алгоритма ПФОК).

Значение «bdsParamsReference» должно устанавливаться равным значению ИО, обозначающего ссылку на набор значений параметров алгоритма ЭЦП, в соответствии с которыми сгенерирован этот открытый ключ.

Значение «bdsParamsList» должно определять набор значений параметров алгоритма ЭЦП, в соответствии с которыми сгенерирован этот открытый ключ.

Значение «cskParamsReference» должно устанавливаться равным значению ИО, обозначающего ссылку на набор значений параметров алгоритма ПФОК, в соответствии с которыми сгенерирован этот открытый ключ.

Значение «cskParamsList» должно определять набор значений параметров алгоритма ПФОК, в соответствии с которыми сгенерирован этот открытый ключ.

6.2.7 Тип «BDSPublicKeyParamsList» определяется следующим образом:

```

BDSParametersList ::= SEQUENCE {
    bdsParameterL  [0] IMPLICIT INTEGER,
    bdsParameterR  [1] IMPLICIT INTEGER,
    bdsParameterP  [2] IMPLICIT INTEGER,
    bdsParameterQ  [3] IMPLICIT INTEGER,
    bdsParameterA  [4] IMPLICIT INTEGER,
    bdsParameterH  [5] IMPLICIT INTEGER,
    bdsParametersInitData BDSParametersInitData OPTIONAL
}
BDSParametersInitData ::= SEQUENCE {
    bdsPrmsInitZSequence OCTET STRING,
    bdsPrmsInitDSequence OCTET STRING,
    bdsPrmsInitRSequence OCTET STRING,
    bdsPrmsInitDValue INTEGER
}

```

Значение «bdsParameterL» должно определять параметр *l* алгоритма ЭЦП.

Значение «bdsParameterR» должно определять параметр *r* алгоритма ЭЦП.

Значение «bdsParameterP» должно определять параметр *p* алгоритма ЭЦП.

Значение «bdsParameterQ» должно определять параметр q алгоритма ЭЦП.

Значение «bdsParameterA» должно определять параметр a алгоритма ЭЦП.

Значение «bdsParameterH» должно определять параметр H алгоритма ЭЦП.

Значение «bdsPrmsInitZSequence» должно содержать последовательность чисел z_1, \dots, z_{31} . Каждые два октета данной последовательности должны представлять один элемент последовательности чисел z_1, \dots, z_{31} . При этом первый октет должен представлять младший октет элемента, второй – старший октет.

Значение «bdsPrmsInitDSequence» должно содержать последовательность чисел d_0, \dots, d_t . Каждые два октета данной последовательности должны представлять один элемент последовательности чисел d_0, \dots, d_t . При этом первый октет должен представлять младший октет элемента, второй – старший октет.

Значение «bdsPrmsInitRSequence» должно содержать последовательностью чисел r_0, \dots, r_s . Каждые два октета данной последовательности должны представлять один элемент последовательности чисел r_0, \dots, r_s . При этом первый октет должен представлять младший октет элемента, второй – старший октет.

Значение «bdsPrmsInitDValue» должно определять значение переменной d .

6.2.8 Тип «CSKPublicKeyParamsList» определяется следующим образом:

```
CSKParametersList ::= SEQUENCE {
    cskParameterL [0] IMPLICIT INTEGER,
    cskParameterR [1] IMPLICIT INTEGER,
    cskParameterP [2] IMPLICIT INTEGER,
    cskParameterG [3] IMPLICIT INTEGER,
    cskParameterN [4] IMPLICIT INTEGER,
    cskParametersInitData CSKParametersInitData OPTIONAL
}
CSKParametersInitData ::= SEQUENCE {
    cskPrmsInitZSequence OCTET STRING
}
```

Значение «cskParameterL» должно определять параметр, устанавливающий длину в битах модуля, по которому проводятся вычисления в алгоритме ПФОК.

Значение «cskParameterR» должно определять параметр, устанавливающий длину секретных ключей алгоритма ПФОК.

Значение «cskParameterP» должно определять параметр, устанавливающий модуль, по которому проводятся вычисления в алгоритме ПФОК.

Значение «cskParameterG» должно определять параметр, устанавливающий основание для операций возведения в степень в алгоритме ПФОК.

Значение «cskParameterN» должно определять параметр, устанавливающий длину в битах общего ключа, получаемого в результате реализации ПФОК.

Значение «cskPrmsInitZSequence» должно содержать последовательность чисел, использованных при генерации параметров ПФОК. Каждые два октета данной последовательности представляют один элемент последовательности чисел. При этом первый октет представляет младший октет элемента, второй – старший октет.

6.2.9 Значение «extensions» типа «Extensions» определяется следующим образом:

```
Extensions ::= SEQUENCE Extension
Extension ::= SEQUENCE {
    extnID OBJECT IDENTIFIER,
    critical BOOLEAN DEFAULT FALSE,
    extnValue OCTET STRING
}
```

Значение «extnID» должно устанавливаться равным значению ИО дополнения.

Значение «critical» должно равняться значению «TRUE», если данное дополнение является критическим.

Значение «extnValue» должно равняться закодированному значению дополнения. Формат значения дополнения зависит от типа данного дополнения, который определяется значением его ИО.

6.3 Особенная часть СОК

6.3.1 Значение «signatureAlgorithm» типа «SignAlgorithmIdentifier» определяется следующим образом:

```

SignAlgorithmIdentifier ::= SEQUENCE {
    algorithm      OBJECT IDENTIFIER,
    parameters    NULL
}

```

Значение «algorithm» должно устанавливаться равным значению ИО алгоритма ЭЦП, в соответствии с которым подписан данный СОК.

6.3.2 Значение «signatureValue» должно содержать ЭЦП в двоичной системе счисления, выработанную от данных, составляющих закодированное представление значения «tbsCertificate». Первый бит последовательности должен представлять старший бит двоичного представления значения ЭЦП. Последний бит последовательности должен представлять младший бит двоичного представления значения ЭЦП.

7 Формат СОС

7.1 Представление СОС

СОС является значением типа «SEQUENCE», определяемого следующим образом:

```

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

```

Значение «tbsCertList» определяет общую часть СОС, как электронного документа. Значения «signatureAlgorithm» и «signatureValue» определяют особенную часть СОС, как электронного документа.

7.2 Общая часть СОС

7.2.1 Значение «tbsCertList» типа «TBSCertList» определяется следующим образом:

```

TBSCertList ::= SEQUENCE {
    version      Version INTEGER (v2(1)),
    signature    SignAlgorithmIdentifier,
    issuer       Name,
    thisUpdate   GeneralizedTime,
    nextUpdate   GeneralizedTime,
    revokedCertificates SEQUENCE OF SEQUENCE {
        serialNumber      INTEGER,
        revocationDate    GeneralizedTime,
        crlEntryExtensions Extensions OPTIONAL
    } OPTIONAL,
    crlExtensions [0] Extensions
}

```

Значение «tbsCertList» должно содержать информацию об отозванных СОК, включая информацию об УЦ создавшем этот СОС. Данные, составляющие это значение, подписываются УЦ, создавшим СОС.

7.2.2 Значение «signature» типа «SignAlgorithmIdentifier» определяется следующим образом:

```

SignAlgorithmIdentifier ::= SEQUENCE {
    algorithm      OBJECT IDENTIFIER,
    parameters    NULL
}

```

Значение «algorithm» должно устанавливаться равным значению ИО алгоритма ЭЦП, в соответствии с которым подписан данный СОС.

7.2.3 Определение типа «Name» значения «issuer» приводится в пункте 6.2.4.

Значение «issuer» должно содержать данные, идентифицирующие УЦ, выпустивший этот СОС. При этом должны быть выдержаны ограничения, приведенные в пункте 6.2.4.

7.2.4 Значение «thisUpdate» должно содержать дату и время создания данного СОС.

7.2.5 Значение «nextUpdate» должно содержать дату и время, не позднее которого УЦ, создавший данный СОС, создаст следующий СОС.

7.2.6 Значение «serialNumber» должно содержать серийный номер СОК, включенного в СОС.

7.2.7 Значение «revocationDate» должно содержать дату и время отзыва СОК.

7.2.8 Определение типа «Extensions» для значений «crlEntryExtensions» и «crlExtensions» приведено в пункте 6.2.9.

Значение «crlEntryExtensions» должно содержать дополнения, необходимые для описания элемента последовательности «revokedCertificates».

Значение «crlExtensions» должно содержать дополнения СОС.

7.3 Особенная часть СОС

7.3.1 Значение «signatureAlgorithm» типа «SignAlgorithmIdentifier» определяется следующим образом:

```
SignAlgorithmIdentifier ::= SEQUENCE {
    algorithm          OBJECT IDENTIFIER,
    parameters        NULL
}
```

Значение «algorithm» должно устанавливаться равным значению ИО алгоритма ЭЦП, в соответствии с которым подписан данный СОС.

7.3.2 Значение «signatureValue» должно содержать ЭЦП в двоичной системе счисления, выработанную от данных, составляющих значение «tbsCertList». Первый бит последовательности должен представлять старший бит двоичного представления значения ЭЦП. Последний бит последовательности должен представлять младший бит двоичного представления значения ЭЦП.

8 Требования к дополнениям

8.1 Виды дополнений

8.1.1 Дополнения СОК (СОС) могут быть критичными или некритичными. Если системе не удастся установить тип или распознать семантику критичного дополнения, то СОК (СОС) необходимо считать не действительным. Если не удастся установить тип или распознать семантику некритичного дополнения, то СОК (СОС) может быть воспринят системой без учета информации, содержащейся в дополнении. Если тип и семантика некритичного дополнения успешно распознаются, то информация, содержащаяся в дополнении, может быть обработана системой.

8.1.2 Дополнения СОК (СОС) могут быть обязательными или опциональными. Если дополнение является обязательным, то оно должно быть обязательно приведено в СОК (СОС). Если обязательное дополнение не приведено в СОК (СОС), то такой СОК (СОС) считается не действительным.

8.1.3 Дополнения СОК (СОС) могут быть стандартными и нестандартными. Если дополнение является стандартным, то его использование допускается только в соответствии с требованиями, изложенными в настоящем документе. Все стандартные дополнения должны иметь определённое значение признака критичности. Нестандартные дополнения могут допускать различные значения признака критичности, то есть одно и то же нестандартное дополнение может быть как критичным, так и некритичным.

8.2 Стандартные дополнения СОК

8.2.1 СОК может иметь следующие стандартные дополнения:

- дополнение с идентификатором открытого ключа проверки подписи УЦ;
- дополнение с идентификатором открытого ключа;
- дополнение с областью применения открытого ключа;
- дополнение с расширенной областью применения открытого ключа;
- дополнение с основными ограничениями применения открытого ключа;
- дополнение с точками распространения СОК.

8.2.2 Дополнение с идентификатором открытого ключа проверки подписи УЦ должно быть обязательным, стандартным, некритичным дополнением и иметь ИО равный «2.5.29.1». Значение данного дополнения определяется следующим образом:

```
AuthorityKeyIdentifier ::= SEQUENCE {
    keyIdentifier          [0] OCTET STRING
}
```

Значение «keyIdentifier» должно содержать результат хэширования функцией h закодированного представления значения «subjectPublicKey» из СОК УЦ.

8.2.3 Дополнение с идентификатором открытого ключа должно быть обязательным, стандартным, некритичным дополнением и иметь ИО равный «2.5.29.14». Значение данного дополнения определяется следующим образом:

SubjectKeyIdentifier ::= OCTET STRING

Значение «SubjectKeyIdentifier» должно содержать результат хэширования функцией *h* закодированного представления значения «subjectPublicKey», представленного в данном СОК.

8.2.4 Дополнение с областью применения открытого ключа должно быть обязательным, стандартным, критичным дополнением и иметь ИО равный «2.5.29.15». Значение данного дополнения определяется следующим образом:

```
KeyUsage ::= BIT STRING {
    digitalSignature      (0),
    nonRepudiation       (1),
    keyEncipherment      (2),
    dataEncipherment     (3),
    keyAgreement         (4),
    keyCertSign          (5),
    cRLSign              (6),
    encipherOnly         (7),
    decipherOnly         (8),
}
```

В этом дополнении должны описываться возможные области применения открытого ключа. Каждый бит значения дополнения определяет допустимость применения открытого ключа для определенной цели. Если бит равен «1», то открытый ключ может использоваться для соответствующей цели. Если бит равен «0», то открытый ключ не может использоваться для этой цели.

Соответствие между номерами бит и целями применения представлено в таблице 1. Биты с не указанными в таблице 1 номерами должны равняться «0» и не должны использоваться.

Таблица 1

Номер бита	Цель применения
0	Проверка ЭЦП данных. Кроме проверки подлинности СОК и СОС
2	Шифрование ключей
4	Для выработки общего ключа в соответствии с ПФОК
5	Проверка подлинности СОК
6	Проверка подлинности СОС

8.2.5 Дополнение с основными ограничениями применения открытого ключа должно быть необязательным, стандартным, критичным дополнением и иметь ИО равный «2.5.29.19». Значение данного дополнения определяется следующим образом:

```
BasicConstraints ::= SEQUENCE {
    cA                BOOLEAN DEFAULT FALSE,
    pathLenConstraint INTEGER OPTIONAL
}
```

Значение «cA» должно равняться значению «TRUE», если приведенный в СОС открытый ключ принадлежит УЦ.

Значение «pathLenConstraint» должно содержать максимальную длину удостоверяемой цепочки СОС.

8.2.6 Дополнение с точками распространения СОС должно быть необязательным, стандартным, некритичным дополнением и иметь ИО равный «2.5.29.31». Значение данного дополнения определяется следующим образом:

```
CRLDistPointsSyntax ::= SEQUENCE OF DistributionPoint
DistributionPoint ::= SEQUENCE {
    distributionPoint [0] DistributionPointName
}
DistributionPointName ::= [0] SEQUENCE OF uniformResourceIdentifier
uniformResourceIdentifier ::= [6] IA5String
```

Значения «uniformResourceIdentifier» должны определять возможные уникальные идентификаторы ресурсов, по которым можно получить текущий СОС данного УЦ.

8.3 Стандартные дополнения СОС

8.3.1 СОС может иметь следующие стандартные дополнения:

- дополнение с идентификатором открытого ключа проверки подписи УЦ;
- дополнение с номером СОС;
- дополнение элемента СОС с причиной отзыва СОК.

8.3.2 Описание дополнения с идентификатором открытого ключа проверки подписи УЦ полностью совпадает с описанием дополнения, приведенным в пункте 8.2.2.

8.3.3 Дополнение с номером СОС должно быть обязательным, стандартным, некритичным дополнением и иметь ИО равный «2.5.29.12». Значение данного дополнения определяется следующим образом:

```
CRLNumber ::= INTEGER
```

Значение «CRLNumber» должно содержать целое число, увеличивающееся в каждом новом СОС, выпущенном данным УЦ.

8.3.4 Дополнение элемента СОС с причиной отзыва СОК должно быть необязательным, стандартным, некритичным дополнением и иметь ИО равный «2.5.29.21». Значение данного дополнения определяется следующим образом:

```
CRLReason ::= ENUMERATED {
    keyCompromise           (1),
    cACompromise            (2),
    affiliationChanged      (3),
    superseded              (4),
    cessationOfOperation    (5),
    certificateHold         (6),
    keyCompromise           (7),
    removeFromCRL          (8),
    privilegeWithdrawn      (9)
}
```

```
ENUMERATED ::= [UNIVERSAL 10] IMPLICIT INTEGER
```

Значение «CRLReason» должно содержать целое число, обозначающее причину отзыва СОК. Соответствие между значениями и причинами отзыва представлено в таблице 3.

Таблица 3

Значение	Причина отзыва
1	Ключ скомпрометирован
2	Ключ УЦ скомпрометирован
3	Изменение информации в СОК
4	Смена ключа
5	Досрочное прекращение действия ключа
6	Действие сертификата приостановлено
8	Удаление из СОС
9	Изменение полномочий

Допускается применение и других значений причины отзыва, больших чем 10.