Утверждены решением заседания
технического комитета по стандартизации
«Криптографическая защита информации»
(Протокол № 10 от 27.11.2012 г.)

Методические рекомендации

технического комитета по стандартизации
«Криптографическая защита информации»
(ТК 26)

Транспортный ключевой контейнер

Дополнения к PKCS#8 и PKCS#12

Версия 1.0

Оглавление

1. Общие положения

В данном документе представлены расширения стандартов PKCS#8 и PKCS#12 для использования в качестве транспортного ключевого контейнера ключей (далее - ТКК) ГОСТ Р 34.10–2001.

Передача закрытого ключа должна осуществляться в защищенном формате.

В ТКК могут помещаться также не защищаемые данные: сертификат открытого ключа, цепочки сертификатов открытых ключей, идентификаторы объектов (OIDs), другие элементы, необходимые для использования закрытого ключа, считываемого с ТКК.

Должна быть обеспечена аутентификация источника закрытого ключа, записанного в ТКК.

При передаче ТКК по каналу связи должна быть обеспечена взаимная аутентификация отправителя и принимающего.

Вся ответственность за защиту закрытого ключа в ТКК от компрометации и его тиражирования лежит на пользователе.

2. Представление ключей ГОСТ Р 34.10–2001

Для обеспечения защиты закрытых ключей от утечек по побочным каналам при считывании и проведении операций с ключами, целесообразно использование маскированных ключей. Для хранения маскированных ключей и наборов масок предлагаются следующие принципы.

- Алгоритм наложения маски определяется базовой операцией криптографического преобразования того или иного алгоритма. Для ключей из ГОСТ 34.10–2001 – умножение в поле $\boldsymbol{F}_q$, для ключей из ГОСТ 28147–89 – сложение по модулю 2^{32}.
- Пусть задана последовательность из k масок $M_1,M_2,\ldots,M_k$. Через $m_i(\cdot)$ обозначим операцию наложения i-ой маски, а через $m^{-1}{}_i(\cdot)$ – операцию снятия i-ой маски, $1 \le i \le k$. Имеется ключ K. Маскированный ключ K_m получается в результате k-кратного применения операции наложения маски, а именно, $K_m = m_k(\ldots(m_2(m_1(K))\ldots)$. Демаскирование выполняется при помощи k-кратного применения операции снятия маски, но в обратном порядке, а именно, $K=m^{-1}{}_1(\ldots(m^{-1}{}_{k-1}(m^{-1}{}_k(K_m))\ldots)$. Согласно установленным правилам, в памяти маскированный ключ представляется как последовательность $I=K_m \| M_1 \| M_2 \| \ldots \| M_k$, где «$\|$» – операция конкатенации. Предположим, ключ K имеет n двоичных разрядов. Будем исходить из предположения, что разрядность ключа и масок совпадает. Тогда для представления в памяти последовательности I понадобится $(k+1)n$ двоичных разрядов или $k+1$ n-разрядных блоков.

Поскольку размерность типа **INTEGER** может изменяться при наличии нулевых значений в старших разрядах числа, использование такого типа для представления закрытого ключа создает неудобства при вычислении размерности.

Предлагается использовать для хранения значений ключа OCTET STRING, то есть:

GostR3410-2001-KeyValueMask ::= OCTET STRING { K_M\|M_1\|M_2\|....\|M_k }

или

```
GostR3410-2001-KeyValueInfo :: = SEQUENCE{
        GostR3410-2001-KeyValueMask,
        GostR3410-2001-PublicKey }

GostR3410-2001-PublicKey :: = OCTET STRING {PubKeyX|PubKeyY}
```

Возможно использование немаскированного закрытого ключа (т.е. k = 0, $\mathbf{K_M} = \mathbf{K}$).

Для обеспечения унификации между представлениями ключей в PFX и сертификатах формата X.509, как секретный, так и открытый ключи представляются в формате little-endian (старший байт справа).

Операцией наложения маски является умножение ключа на число обратное маске: $\mathbf{K_M = K * M^{-1} \bmod q}$, где значение **q** взято из параметров ключа. Соответственно, операцией снятия маски является умножение маскированного ключа на маску: $\mathbf{K = K_M * M \bmod q}$.

3. Портфель ключевой информации (KeyBag)

В соответствии с PKCS#12 [2] и PKCS#8 [1] портфель ключевой информации (KeyBag) представляется в следующем виде:

```
PrivateKeyInfo ::= SEQUENCE {
  version Version,
  privateKeyAlgorithm PrivateKeyAlgorithmIdentifier,
  privateKey PrivateKey,
  attributes [0] IMPLICIT Attributes OPTIONAL }
```

Version ::= INTEGER

Версия структуры, на данный момент 0

PrivateKeyAlgorithmIdentifier ::= AlgorithmIdentifier

Для закрытых ключей ГОСТ Р 34.10–2001 используются идентификаторы соответствующих открытых ключей, представленные в разделе 2.3.2 RFC 4491 [5].

PrivateKey ::= OCTET STRING

Содержимым данного типа является значение закрытого ключа, закодированное в соответствии с типом **GostR3410-2001-PrivateKey**:

```
GostR3410-2001-PrivateKey ::= CHOICE {
        GostR3410-2001-KeyValueMask,
        GostR3410-2001-KeyValueInfo }
```

Attributes ::= SET OF Attribute

Атрибуты могут содержать дополнительную необходимую информацию о ключе .

Ключ в зашифрованном виде представляется в виде структуры:

```
EncryptedPrivateKeyInfo ::= SEQUENCE {
  encryptionAlgorithm EncryptionAlgorithmIdentifier,
  encryptedData EncryptedData }
```

EncryptionAlgorithmIdentifier ::= AlgorithmIdentifier

При шифровании должен использоваться алгоритм PBES2 с использованием ГОСТ Р 34.11–94 и ГОСТ 28147–89. Алгоритм и параметры шифрования **EncryptionAlgorithmIdentifier** указываются в соответствии с дополнением к PKCS#5 [7].

EncryptedData ::= OCTET STRING

Результат зашифрования кодированной структуры **PrivateKeyInfo.**

4. Данные, защищаемые в ТКК

В соответствии с п. 4.1 PKCS#12 [2], каждый раздел ТКК, содержащий конфиденциальные сведения, должен быть зашифрован в рамках **ContentInfo** типа **AuthenticatedSafe.** Портфель сертификата (**CertBag**) [2, п. 4.2.3], соответствующий присутствующему в ТКК портфелю закрытого ключа (**KeyBag**), несет в себе информацию, облегчающую потенциальному злоумышленнику задачу первичного анализа перехваченного зашифрованного сообщения, в частности по данным содержащимся в сертификате, можно получить информацию о владельце секретного ключа. В этой связи содержащийся в ТКК **CertBag** также может быть зашифрован. Шифрование различных **ContentInfo** и портфелей закрытого ключа **KeyBag** должно быть осуществлено обязательно с использованием уникальных параметров операции выработки ключа из пароля, исключающих повторное использование одного и того же секретного ключа для шифрования каких-либо различных шифруемых разделов ТКК.

5. Обеспечение целостности и конфиденциальности ключей

Для обеспечения целостности и конфиденциальности ключей используется транспортный ключ (KEK), выработанный одним из следующих способов:

5.1. Парольная защита

У отправителя и принимающего имеется предварительно согласованный пароль . Отправитель с помощью алгоритма диверсификации вырабатывает парольный ключ в соответствии с PKCS#5 и дополнением [7] и использует его для шифрования передаваемого закрытого ключа. Принимающий независимо вырабатывает парольный ключ и использует его для извлечения закрытого ключа из ТКК.

Целостность ТКК обеспечивается с использованием алгоритма HMAC_GOSTR3411 в соответствии с разделом 6 RFC 4490 [6].

Ввиду простоты реализации, этот способ является наиболее приемлемым для большинства практических приложений. При этом в качестве алгоритма диверсификации допускается использование только алгоритма PBKDF2 [7].

Для шифрования различных разделов ТКК используется один и тот же пароль, но различные случайные значения параметра S длиной от 8 до 32 байт [7, раздел 2].

Пароль Psw должен быть представлен в формате UTF-8 без завершающего нуля и подан на вход алгоритма PBKDF2 в качестве параметра P.

При проверки целостности ТКК с помощью алгоритма HMAC_GOSTR3411 ключ для данного алгоритма также вырабатывается по алгоритму PBKDF2 с тем же самым значением параметра P и случайным вектором S длиной от 8 до 32 байт. Ключом алгоритма HMAC_GOSTR3411 должны быть последние 32 байта 96-байтовой последовательности, вырабатываемой с помощью PBKDF2.

Используется идентификатор алгоритма:

id-GostR3411-94 OBJECT IDENTIFIER ::= {iso(1) member-body(2) ru(643) rans(2) cryptopro(2) gostR3411(9)}

Функция HMAC_GOSTR3411 вычисляется от содержимого поля **content** структуры **authSafe** (см. п. 6).

5.2. Выработка ключа по протоколу Диффи-Хелмана

В случае, если у источника имеются соответственно ключевая пара (Priv1,1) с сертификатом открытого ключа 1, а у приемника ключевая пара (Priv2,2) с сертификатом открытого ключа 2, используется первичная пересылка от приемника к источнику ТКК, содержащий сертификат открытого ключа, данный пункт может быть опущен, если у отправителя имеется сертификат открытого ключа 2. Отправитель после проверки легитимности сертификата открытого ключа вырабатывает ключ KEK как по паре ключей (Priv1,2) на базе протокола Диффи-Хеллмана. В ТКК, передаваемый принимающему, отправитель записывает сертификат своего открытого ключа 1. Принимающий, проверив легитимность этого сертификата, аналогично вырабатывает ключ KEK по паре ключей (Priv2,1).

Целостность контейнера в этом случае обеспечивается ЭЦП на ключе отправителя.

6. Формат PFX контейнера

В соответствии с PKCS#12 контейнер имеет вид:

```
PFX ::= SEQUENCE {
        version             INTEGER {v3(3)}(v3,...),
        authSafe    ContentInfo,
        macData     MacData OPTIONAL
}

MacData ::= SEQUENCE {
        mac         DigestInfo,
        macSalt     OCTET STRING,
        iterations  INTEGER DEFAULT 1
}
```

authSafe - в зависимости от метода контроля целостности представляется либо как тип **data** - при использовании парольной защиты, либо как **signedData** - в случае использования ЭЦП отправителя для защиты целостности контейнера, в соответствии с PKCS#7 [3].

В случае использования парольной защиты для контроля целостности поле **macData** должно содержать информацию об алгоритме и параметрах выработки парольного ключа в соответствии с [7]. Контроль целостности обеспечивается с использованием алгоритма HMAC_GOSTR3411.

Данные в **authSafe** представляют собой результат кодирования списка объектов:

```
AuthenticatedSafe ::= SEQUENCE OF ContentInfo
        -- Data if unencrypted
        -- EncryptedData if password-encrypted
        -- EnvelopedData if public key-encrypted
```

При использовании алгоритма Диффи-Хелмана для выработки ключа обмена, данные представляются в виде **EnvelopedData** в соответствии с RFC 5652 [3] и RFC 4490 [6].

AuthenticatedSafe - может содержать объекты различного типа - ключи, сертификаты, списки отозванных сертификатов. В соответствии с PKCS#12 [2] :

```
SafeBag ::= SEQUENCE {
        bagId           BAG-TYPE.&id ({PKCS12BagSet})
        bagValue        [0] EXPLICIT BAG-
TYPE.&Type({PKCS12BagSet}{@bagId}),
        bagAttributes       SET OF PKCS12Attribute OPTIONAL
}
```

В случае представления данных в виде **EnvelopedData** секретный ключ может быть представлен в виде:

```
keyBag          BAG-TYPE ::=
        {KeyBag IDENTIFIED BY {bagtypes 1}}
```

bagValue в этом случае содержит ключ и информацию о нем в виде **PrivateKeyInfo.**

При использовании парольной защиты для хранения ключа должен использоваться тип:

```
pkcs8ShroudedKeyBag BAG-TYPE ::=
        {PKCS8ShroudedKeyBag IDENTIFIED BY {bagtypes 2}}
```

bagValue в этом случае содержит ключ и информацию о нем в зашифрованном виде **EncryptedPrivateKeyInfo.**

7. Контрольные примеры

В данном разделе приведен пример контейнера, зашифрованного на пароле «Пароль для PFX» в соответствии с изложенной в данном документе схемой

7.1. Сертификаты

В данном разделе представлены используемые в контрольных примерах сертификаты и соответствующие им закрытые ключи

7.1.1. Корневой сертификат

Значение корневого сертификата закодированное с соответствии с алгоритмом BASE64:

```
MIIBfTCCASqgAwIBAgIBADAKBgYqhQMCAgMFADA3MQswCQYDVQQGEwJSVTEoMCYG
A1UEAwwfQ0EgY2VydGlmaWNhdGUgKFBLQ1MxMiBleGFtcGxlKTAeFw0xMjA2MTQx
MzEyMzFaFw0xNzA2MTQxMzEyMzFaMDcxCzAJBgNVBAYTAlJVMSgwJgYDVQQDDB9D
QSBjZXJ0aWZpY2F0ZSAoUEtDUzEyIGV4YW1wbGUpMGMwHAYGKoUDAgITMBIGByqF
AwICIwEGByqFAwICHgEDQwAEQHxF7QOkGNDlnKWiBwdD80gToowegPcHR1Y1r2ZR
RQqB610f3uEWN/EikI7exYVRR0dmCyILLMmgRxX+KU4qmgejHTAbMAwGA1UdEwQF
MAMBAf8wCwYDVR0PBAQDAgEGMAoGBiqFAwICAwUAA0EAFnvKPRo2tQkI/iqu/CkP
YQJPW43KnRMqkmB/NnGOC5+wdivIA5yJaGbT2sQ1r+n6qyJnG32yV44DrSe7b2DV
OA==
```

Представление ASN.1 корневого сертификата:

```
0  381: SEQUENCE {
   4  298:   SEQUENCE {
   8    3:     [0] {
  10    1:       INTEGER 2
         :       }
  13    1:     INTEGER 0
  16   10:     SEQUENCE {
  18    6:       OBJECT IDENTIFIER '1 2 643 2 2 3'
  26    0:       NULL
         :       }
  28   55:     SEQUENCE {
  30   11:       SET {
  32    9:         SEQUENCE {
  34    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
  39    2:           PrintableString 'RU'
         :           }
         :         }
  43   40:       SET {
  45   38:         SEQUENCE {
  47    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
  52   31:           UTF8String 'CA certificate (PKCS12 example)'
         :           }
         :         }
         :       }
  85   30:     SEQUENCE {
  87   13:       UTCTime 14/06/2012 13:12:31 GMT
 102   13:       UTCTime 14/06/2017 13:12:31 GMT
         :       }
 117   55:     SEQUENCE {
 119   11:       SET {
 121    9:         SEQUENCE {
 123    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
 128    2:           PrintableString 'RU'
         :           }
         :         }
 132   40:       SET {
 134   38:         SEQUENCE {
 136    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
 141   31:           UTF8String 'CA certificate (PKCS12 example)'
         :           }
         :         }
         :       }
 174   99:     SEQUENCE {
 176   28:       SEQUENCE {
 178    6:         OBJECT IDENTIFIER '1 2 643 2 2 19'
 186   18:         SEQUENCE {
 188    7:           OBJECT IDENTIFIER '1 2 643 2 2 35 1'
 197    7:           OBJECT IDENTIFIER '1 2 643 2 2 30 1'
         :           }
         :         }
 206   67:       BIT STRING, encapsulates {
 209   64:         OCTET STRING
         :           7C 45 ED 03 A4 18 D0 E5 9C A5 A2 07 07 43 F3 48
         :           13 A2 8C 1E 80 F7 07 47 56 35 AF 66 51 45 0A 81
         :           EB 5D 1F DE E1 16 37 F1 22 90 8E DE C5 85 51 47
         :           47 66 0B 22 0B 2C C9 A0 47 15 FE 29 4E 2A 9A 07
         :         }
         :       }
 275   29:     [3] {
 277   27:       SEQUENCE {
 279   12:         SEQUENCE {
 281    3:           OBJECT IDENTIFIER basicConstraints (2 5 29 19)
 286    5:           OCTET STRING, encapsulates {
 288    3:             SEQUENCE {
 290    1:               BOOLEAN TRUE
         :               }
         :             }
         :           }
 293   11:         SEQUENCE {
 295    3:           OBJECT IDENTIFIER keyUsage (2 5 29 15)
 300    4:           OCTET STRING, encapsulates {
 302    2:             BIT STRING 1 unused bit
         :               '1100000'B
         :             }
         :           }
         :         }
         :       }
         :     }
 306   10:   SEQUENCE {
 308    6:     OBJECT IDENTIFIER '1 2 643 2 2 3'
```

```
 316    0:     NULL
        :     }
 318   65:   BIT STRING
        :     16 7B CA 3D 1A 36 B5 09 08 FE 2A AE FC 29 0F 61
        :     02 4F 5B 8D CA 9D 13 2A 92 60 7F 36 71 8E 0B 9F
        :     B0 76 2B C8 03 9C 89 68 66 D3 DA C4 35 AF E9 FA
        :     AB 22 67 1B 7D B2 57 8E 03 AD 27 BB 6F 60 D5 38
        :   }
```

7.1.2. Тестовый сертификат

Тестовый сертификат присутствует во всех приведенных контрольных примерах в качестве сертификата, подлежащего упаковке в контейнер. Значение тестового сертификата, закодированное с соответствии с алгоритмом BASE64:

```
MIIBfTCCASqgAwIBAgICASIwCgYGKoUDAgIDBQAwNzELMAkGA1UEBhMCUlUxKDAm
BgNVBAMMH0NBIGNlcnRpZmljYXRlIChQS0NTMTIgZXhhbXBsZSkwHhcNMTIwNjE0
MTkzMDE5WhcNMTcwNjE0MTkzMDE5WjA5MQswCQYDVQQGEwJSVTEqMCgGA1UEAwwh
VGVzdCBjZXJ0aWZpY2F0ZSAoUEtDUzEyIGV4YW1wbGUpMGMwHAYGKoUDAgITMBIG
ByqFAwICIwEGByqFAwICHgEDQwAEQHlSMCMAJZgoHUCIFmn+eqOgYlQy8h7SfjZ2
kMkJ4xTae8jtZYxHLq3P+qJeismsHdmqdqSBbOGGdOaJNNPbAZKjGjAYMAkGA1Ud
EwQCMAAwCwYDVR0PBAQDAgWgMAoGBiqFAwICAwUAA0EArslfUeqhW9eFkspn89+C
OQEJX6JoghiOjFYlky0XmaaDl3D6EcbID+B6cBEmcXF21xxIEeYJIAqGzOEnMXdT
cg==
```

Представление ASN.1 тестового сертификата:

```
0  381: SEQUENCE {
   4  298:   SEQUENCE {
   8    3:     [0] {
  10    1:       INTEGER 2
        :       }
  13    2:     INTEGER 290
  17   10:     SEQUENCE {
  19    6:       OBJECT IDENTIFIER '1 2 643 2 2 3'
  27    0:       NULL
        :       }
  29   55:     SEQUENCE {
  31   11:       SET {
  33    9:         SEQUENCE {
  35    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
  40    2:           PrintableString 'RU'
        :           }
        :         }
  44   40:       SET {
  46   38:         SEQUENCE {
  48    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
  53   31:           UTF8String 'CA certificate (PKCS12 example)'
        :           }
        :         }
        :       }
  86   30:     SEQUENCE {
  88   13:       UTCTime 14/06/2012 19:30:19 GMT
 103   13:       UTCTime 14/06/2017 19:30:19 GMT
        :       }
 118   57:     SEQUENCE {
 120   11:       SET {
 122    9:         SEQUENCE {
 124    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
 129    2:           PrintableString 'RU'
        :           }
        :         }
 133   42:       SET {
 135   40:         SEQUENCE {
 137    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
 142   33:           UTF8String 'Test certificate (PKCS12 example)'
        :           }
        :         }
        :       }
 177   99:     SEQUENCE {
 179   28:       SEQUENCE {
 181    6:         OBJECT IDENTIFIER '1 2 643 2 2 19'
 189   18:         SEQUENCE {
 191    7:           OBJECT IDENTIFIER '1 2 643 2 2 35 1'
 200    7:           OBJECT IDENTIFIER '1 2 643 2 2 30 1'
```

```
        :           }
        :         }
 209   67:       BIT STRING, encapsulates {
 212   64:         OCTET STRING
        :           79 52 30 23 00 25 98 28 1D 40 88 16 69 FE 7A A3
        :           A0 62 54 32 F2 1E D2 7E 36 76 90 C9 09 E3 14 DA
        :           7B C8 ED 65 8C 47 2E AD CF FA A2 5E 8A C9 AC 1D
        :           D9 AA 76 A4 81 6C E1 86 74 E6 89 34 D3 DB 01 92
        :         }
        :       }
 278   26:     [3] {
 280   24:       SEQUENCE {
 282    9:         SEQUENCE {
 284    3:           OBJECT IDENTIFIER basicConstraints (2 5 29 19)
 289    2:           OCTET STRING, encapsulates {
 291    0:             SEQUENCE {}
        :             }
        :           }
 293   11:         SEQUENCE {
 295    3:           OBJECT IDENTIFIER keyUsage (2 5 29 15)
 300    4:           OCTET STRING, encapsulates {
 302    2:             BIT STRING 5 unused bits
        :               '101'B
        :             }
        :           }
        :         }
        :       }
        :     }
 306   10:   SEQUENCE {
 308    6:     OBJECT IDENTIFIER '1 2 643 2 2 3'
 316    0:     NULL
        :     }
 318   65:   BIT STRING
        :     AE C9 5F 51 EA A1 5B D7 85 92 CA 67 F3 DF 82 39
        :     01 09 5F A2 68 82 18 8E 8C 56 25 93 2D 17 99 A6
        :     83 97 70 FA 11 C6 C8 0F E0 7A 70 11 26 71 71 76
        :     D7 1C 48 11 E6 09 20 0A 86 CC E1 27 31 77 53 72
        :   }
```

Значение закрытого ключа, соответствующего тестовому сертификату, в представлении big-endian (старший байт слева):

```
C9 0D 4A 60 74 4B 6E F9 DD B1 F1 D5 E2 34 F0 6C
EF 73 74 52 2D 03 91 89 D9 2E 82 DD CF 41 14 16
```

7.1.3. Сертификат отправителя

Сертификат отправителя присутствует во втором контрольном примере в качестве сертификата подписателя контейнера. Значение сертификата отправителя, закодированное с соответствии с алгоритмом BASE64:

```
MIIBgzCCATCgAwIBAgICAR8wCgYGKoUDAgIDBQAwNzELMAkGA1UEBhMCUlUxKDAm
BgNVBAMMH0NBIGNlcnRpZmljYXRlIChQS0NTMTIgZXhhbXBsZSkwHhcNMTIwNjE0
MTMyMzI1WhcNMTcwNjE0MTMyMzI1WjA/MQswCQYDVQQGEwJSVTEwMC4GA1UEAwwn
UEZYLWlzc3VlciBjZXJ0aWZpY2F0ZSAoUEtDUzEyIGV4YW1wbGUpMGMwHAYGKoUD
AgITMBIGByqFAwICIwEGByqFAwICHgEDQwAEQMABvWY7xsXHe+Rwc7aTKlNRJ0Iq
CP5DYfKvrH6LRqoU3iqqAoJEDPWSkXKP6/e5eecEZE1L5h+3EC/sZb3+4NejGjAY
MAkGA1UdEwQCMAAwCwYDVR0PBAQDAgWgMAoGBiqFAwICAwUAA0EAWu7Pmmq0Mcs7
wG0I7C4UavPifwjeQiZKyLCagvl7I7agJE2gEkY2JS56mtpAZXI+pvsCc4neYNwL
soqhSf9ycQ==
```

Представление ASN.1 сертификата отправителя:

```
0  387: SEQUENCE {
   4  304:   SEQUENCE {
   8    3:     [0] {
  10    1:       INTEGER 2
        :       }
  13    2:     INTEGER 287
  17   10:     SEQUENCE {
  19    6:       OBJECT IDENTIFIER '1 2 643 2 2 3'
  27    0:       NULL
        :       }
  29   55:     SEQUENCE {
```

```
 31   11:       SET {
 33    9:         SEQUENCE {
 35    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
 40    2:           PrintableString 'RU'
       :           }
       :         }
 44   40:       SET {
 46   38:         SEQUENCE {
 48    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
 53   31:           UTF8String 'CA certificate (PKCS12 example)'
       :           }
       :         }
       :       }
 86   30:     SEQUENCE {
 88   13:       UTCTime 14/06/2012 13:23:25 GMT
103   13:       UTCTime 14/06/2017 13:23:25 GMT
       :       }
118   63:     SEQUENCE {
120   11:       SET {
122    9:         SEQUENCE {
124    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
129    2:           PrintableString 'RU'
       :           }
       :         }
133   48:       SET {
135   46:         SEQUENCE {
137    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
142   39:           UTF8String 'PFX-issuer certificate (PKCS12 example)'
       :           }
       :         }
       :       }
183   99:     SEQUENCE {
185   28:       SEQUENCE {
187    6:         OBJECT IDENTIFIER '1 2 643 2 2 19'
195   18:         SEQUENCE {
197    7:           OBJECT IDENTIFIER '1 2 643 2 2 35 1'
206    7:           OBJECT IDENTIFIER '1 2 643 2 2 30 1'
       :           }
       :         }
215   67:       BIT STRING, encapsulates {
218   64:         OCTET STRING
       :           C0 01 BD 66 3B C6 C5 C7 7B E4 70 73 B6 93 2A 53
       :           51 27 42 2A 08 FE 43 61 F2 AF AC 7E 8B 46 AA 14
       :           DE 2A AA 02 82 44 0C F5 92 91 72 8F EB F7 B9 79
       :           E7 04 64 4D 4B E6 1F B7 10 2F EC 65 BD FE E0 D7
       :         }
       :       }
284   26:     [3] {
286   24:       SEQUENCE {
288    9:         SEQUENCE {
290    3:           OBJECT IDENTIFIER basicConstraints (2 5 29 19)
295    2:           OCTET STRING, encapsulates {
297    0:             SEQUENCE {}
       :             }
       :           }
299   11:         SEQUENCE {
301    3:           OBJECT IDENTIFIER keyUsage (2 5 29 15)
306    4:           OCTET STRING, encapsulates {
308    2:             BIT STRING 5 unused bits
       :               '101'B
       :             }
       :           }
       :         }
       :       }
       :     }
312   10:   SEQUENCE {
314    6:     OBJECT IDENTIFIER '1 2 643 2 2 3'
322    0:     NULL
       :     }
324   65:   BIT STRING
       :     5A EE CF 9A 6A B4 31 CB 3B C0 6D 08 EC 2E 14 6A
       :     F3 E2 7F 08 DE 42 26 4A C8 B0 9A 82 F9 7B 23 B6
       :     A0 24 4D A0 12 46 36 25 2E 7A 9A DA 40 65 72 3E
       :     A6 FB 02 73 89 DE 60 DC 0B B2 8A A1 49 FF 72 71
       :   }
```

Значение закрытого ключа, соответствующего сертификату отправителя, в представлении big-endian (старший байт слева):

```
F9 FC 71 08 D1 4A A6 2B F0 13 C4 9E 6C AE F7 98
CA 0C 1D EC FA 73 19 00 1F B2 F3 74 A3 3C CA 50
```

7.1.4. Сертификат получателя

Сертификат получателя присутствует во втором контрольном примере в качестве сертификата, с помощью которого производится шифрование контейнера. Значение сертификата получателя, закодированное с соответствии с алгоритмом BASE64:

```
MIIBhTCCATKgAwIBAgICASAwCgYGKoUDAgIDBQAwNzELMAkGA1UEBhMCUlUxKDAm
BgNVBAMMH0NBIGNlcnRpZmljYXRlIChQS0NTMTIgZXhhbXBsZSkwHhcNMTIwNjE0
MTMyNzA0WhcNMTcwNjE0MTMyNzA0WjBBMQswCQYDVQQGEwJSVTEyMDAGA1UEAwwp
UEZYLXJlY2VpdmVyIGNlcnRpZmljYXRlIChQS0NTMTIgZXhhbXBsZSkwYzAcBgYq
hQMCAhMwEgYHKoUDAgIkAAYHKoUDAgIeAQNDAARA2/9UcsvhgzpC4rlrJ9DNe5P5
gCjFLbxRUdZOCaGjgvQH8nPk7ueKfavqyzEHRykmiVdsY6cAsEiQdH/1TB/j7aMa
MBgwCQYDVR0TBAIwADALBgNVHQ8EBAMCBaAwCgYGKoUDAgIDBQADQQA2cxOriZ+m
RN8F66WxqonkJv7P7u86BOdMg7qGESqZIOOwoXJ4QMkDysM7jLElQ8ofR1zEPL4K
lJbyu48wZWpZ
```

Представление ASN.1 сертификата получателя:

```
0  389: SEQUENCE {
   4  306:   SEQUENCE {
   8    3:     [0] {
  10    1:       INTEGER 2
        :       }
  13    2:     INTEGER 288
  17   10:     SEQUENCE {
  19    6:       OBJECT IDENTIFIER '1 2 643 2 2 3'
  27    0:       NULL
        :       }
  29   55:     SEQUENCE {
  31   11:       SET {
  33    9:         SEQUENCE {
  35    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
  40    2:           PrintableString 'RU'
        :           }
        :         }
  44   40:       SET {
  46   38:         SEQUENCE {
  48    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
  53   31:           UTF8String 'CA certificate (PKCS12 example)'
        :           }
        :         }
        :       }
  86   30:     SEQUENCE {
  88   13:       UTCTime 14/06/2012 13:27:04 GMT
 103   13:       UTCTime 14/06/2017 13:27:04 GMT
        :       }
 118   65:     SEQUENCE {
 120   11:       SET {
 122    9:         SEQUENCE {
 124    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
 129    2:           PrintableString 'RU'
        :           }
        :         }
 133   50:       SET {
 135   48:         SEQUENCE {
 137    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
 142   41:           UTF8String
        :             'PFX-receiver certificate (PKCS12 example)'
        :           }
        :         }
        :       }
 185   99:     SEQUENCE {
 187   28:       SEQUENCE {
 189    6:         OBJECT IDENTIFIER '1 2 643 2 2 19'
 197   18:         SEQUENCE {
 199    7:           OBJECT IDENTIFIER '1 2 643 2 2 36 0'
 208    7:           OBJECT IDENTIFIER '1 2 643 2 2 30 1'
        :           }
        :         }
 217   67:       BIT STRING, encapsulates {
 220   64:         OCTET STRING
        :           DB FF 54 72 CB E1 83 3A 42 E2 B9 6B 27 D0 CD 7B
        :           93 F9 80 28 C5 2D BC 51 51 D6 4E 09 A1 A3 82 F4
        :           07 F2 73 E4 EE E7 8A 7D AB EA CB 31 07 47 29 26
```

```
        :           89 57 6C 63 A7 00 B0 48 90 74 7F F5 4C 1F E3 ED
        :         }
        :       }
 286   26:     [3] {
 288   24:       SEQUENCE {
 290    9:         SEQUENCE {
 292    3:           OBJECT IDENTIFIER basicConstraints (2 5 29 19)
 297    2:           OCTET STRING, encapsulates {
 299    0:             SEQUENCE {}
        :             }
        :           }
 301   11:         SEQUENCE {
 303    3:           OBJECT IDENTIFIER keyUsage (2 5 29 15)
 308    4:           OCTET STRING, encapsulates {
 310    2:             BIT STRING 5 unused bits
        :               '101'B
        :             }
        :           }
        :         }
        :       }
        :     }
 314   10:   SEQUENCE {
 316    6:     OBJECT IDENTIFIER '1 2 643 2 2 3'
 324    0:     NULL
        :     }
 326   65:   BIT STRING
        :     36 73 13 AB 89 9F A6 44 DF 05 EB A5 B1 AA 89 E4
        :     26 FE CF EE EF 3A 04 E7 4C 83 BA 86 11 2A 99 20
        :     E3 B0 A1 72 78 40 C9 03 CA C3 3B 8C B1 25 43 CA
        :     1F 47 5C C4 3C BE 0A 94 96 F2 BB 8F 30 65 6A 59
        :   }
```

Значение закрытого ключа, соответствующего сертификату получателя, в представлении big-endian (старший байт слева):

```
0F F5 46 D3 53 5C 04 5F ED 56 2B 0C F9 D1 10 E2
9E 5D 64 21 16 E0 1E A0 0E E0 2A F3 D4 88 70 F3
```

7.2. Контрольный пример 1

В данном разделе приведен пример контейнера, зашифрованного на пароле «Пароль для PFX» в соответствии с изложенной в данном документе схемой применения парольной защиты.

Значение контейнера, закодированное с соответствии с алгоритмом BASE64:

```
MIIEEwIBAzCCA7QGCSqGSIb3DQEHAaCCA6UEggOhMIIDnTCCARgGCSqGSIb3DQEH
AaCCAQkEggEFMIIBATCB/gYLKoZIhvcNAQwKAQKggbswgbgwbQYJKoZIhvcNAQUN
MGAwPwYJKoZIhvcNAQUMMDIEILpuGiK13MTTYcljCu+3p9L4lD9QUcidf0uomBH8
p0X4AgIH0DAKBgYqhQMCAgoFADAdBgYqhQMCAhUwEwQIlipveMPIIywGByqFAwIC
HwEER3OFKLu/W8vuxRmlxBKaanNMKP5OCR9/i4unqEeUdzbfEk4Xp1w+9WVc+2JF
SY2mXnoGq5VGjsZvS/HqkouBHx4k2ZtZ2ga0MTEwLwYJKoZIhvcNAQkVMSIEIOGL
70sd3FoK8jrEQ9mK7YwjSJy0AHk2O4rltiNzDd4yMIICfQYJKoZIhvcNAQcGoIIC
bjCCAmoCAQAwggJjBgkqhkiG9w0BBwEwbQYJKoZIhvcNAQUNMGAwPwYJKoZIhvcN
AQUMMDIEIL++UYUsz7L3OZRulyPkAhjE+CKEuZrPVhSI/j+3qftJAgIH0DAKBgYq
hQMCAgoFADAdBgYqhQMCAhUwEwQIa1W2aiLTw1IGByqFAwICHwGAggHlVy8Z1gHf
3syYkTYwJcYvvX0/kUlSRlJtvgzJGrQkRUVAisIaL2eNmoGvb3tBuGFQyVT3XBJR
mSJU8m7JM/Ywm6ISZ86FvNKCpuEQdhWqqaTeSW6PCwYblT3597p+wBr8o78rs9XU
RVkiBFGKhdxFBzX67WiZE8suKAEPbEcQcqKRhc6rOCq30KZoohxzm5egLf04ADfS
LCp7aLZoL+ZOPXHi4zDvguJVYaSRZAx8H9Du/pfdxWYUAgBYWF7j3w1Rr7e4R2pl
0yb8Ssb18FqqAg0i5y8y6e4QdhHNsxkhAJpDnWU9GySjz/QtmrODq6S7vVLbvxN3
lBEOnbqeKObfeMfTMtw1QnQq4To+18DZD/x215ks9hxbyDAwAJ0gBcHzk59e2Mll
Nmd1rX7h7F5RAOahj6afVb36UgYZzjrzQvxWbasNobTKnetpjqyE6wk8IVJH1JDa
/lV8njzst+7qWxaJQY4ZHb/XAZ1D3iLQAbNz9vxV2ZgQiEd4wYAJH5NyQu4yAtLI
pyUQd0E5RRG68746cRsH7Y0JBc1tdeelSynqU7CJBlixe6ltD4eieD9cNxQNgsJ+
3ixH9p2fHXhJ6X8cKv5vxR9twGv/gp+CD7hqb+OYms21Bq/DBeMmwHpkSqAJG+Aw
VjAuMAoGBiqFAwICCQUABCDY1/8wm16eIez/P047eu7pkmfxjC7v5l7w9poiOrvJ
mQQgyCOpfWCBX3nkoEVUA1uMXFC5vcUaf/mLm05x/S7fyTACAgfQ
```

Представление ASN.1:

```
0 1043: SEQUENCE {
```

```
  4    1:   INTEGER 3
  7  948:   SEQUENCE {
 11    9:     OBJECT IDENTIFIER data (1 2 840 113549 1 7 1)
 22  933:     [0] {
 26  929:       OCTET STRING, encapsulates {
 30  925:         SEQUENCE {
 34  280:           SEQUENCE {
 38    9:             OBJECT IDENTIFIER data (1 2 840 113549 1 7 1)
 49  265:             [0] {
 53  261:               OCTET STRING, encapsulates {
 57  257:                 SEQUENCE {
 61  254:                   SEQUENCE {
 64   11:                     OBJECT IDENTIFIER
      :                       pkcs-12-pkcs-8ShroudedKeyBag (1 2 840 113549 1 12 10 1 2)
 77  187:                     [0] {
 80  184:                       SEQUENCE {
 83  109:                         SEQUENCE {
 85    9:                           OBJECT IDENTIFIER
      :                             pkcs5PBES2 (1 2 840 113549 1 5 13)
 96   96:                           SEQUENCE {
 98   63:                             SEQUENCE {
100    9:                               OBJECT IDENTIFIER
      :                                 pkcs5PBKDF2 (1 2 840 113549 1 5 12)
111   50:                               SEQUENCE {
113   32:                                 OCTET STRING
      :                   BA 6E 1A 22 B5 DC C4 D3 61 C9 63 0A EF B7 A7 D2
      :                   F8 94 3F 50 51 C8 9D 7F 4B A8 98 11 FC A7 45 F8
147    2:                                 INTEGER 2000
151   10:                                 SEQUENCE {
153    6:                                   OBJECT IDENTIFIER '1 2 643 2 2 10'
161    0:                                   NULL
      :                                   }
      :                                 }
      :                               }
163   29:                             SEQUENCE {
165    6:                               OBJECT IDENTIFIER '1 2 643 2 2 21'
173   19:                               SEQUENCE {
175    8:                                 OCTET STRING 96 2A 6F 78 C3 C8 23 2C
185    7:                                 OBJECT IDENTIFIER '1 2 643 2 2 31 1'
      :                                 }
      :                               }
      :                             }
      :                           }
194   71:                         OCTET STRING
      :                   73 85 28 BB BF 5B CB EE C5 19 A5 C4 12 9A 6A 73
      :                   4C 28 FE 4E 09 1F 7F 8B 8B A7 A8 47 94 77 36 DF
      :                   12 4E 17 A7 5C 3E F5 65 5C FB 62 45 49 8D A6 5E
      :                   7A 06 AB 95 46 8E C6 6F 4B F1 EA 92 8B 81 1F 1E
      :                   24 D9 9B 59 DA 06 B4
      :                         }
      :                       }
267   49:                     SET {
269   47:                       SEQUENCE {
271    9:                         OBJECT IDENTIFIER
      :                           localKeyID (for PKCS #12) (1 2 840 113549 1 9 21)
282   34:                         SET {
284   32:                           OCTET STRING
      :                   E1 8B EF 4B 1D DC 5A 0A F2 3A C4 43 D9 8A ED 8C
      :                   23 48 9C B4 00 79 36 3B 8A E5 B6 23 73 0D DE 32
      :                         }
      :                       }
      :                     }
      :                   }
      :                 }
      :               }
      :             }
318  637:           SEQUENCE {
322    9:             OBJECT IDENTIFIER encryptedData (1 2 840 113549 1 7 6)
333  622:             [0] {
337  618:               SEQUENCE {
341    1:                 INTEGER 0
344  611:                 SEQUENCE {
348    9:                   OBJECT IDENTIFIER data (1 2 840 113549 1 7 1)
359  109:                   SEQUENCE {
361    9:                     OBJECT IDENTIFIER
      :                       pkcs5PBES2 (1 2 840 113549 1 5 13)
372   96:                     SEQUENCE {
374   63:                       SEQUENCE {
376    9:                         OBJECT IDENTIFIER
```

```
        :                           pkcs5PBKDF2 (1 2 840 113549 1 5 12)
 387   50:                         SEQUENCE {
 389   32:                           OCTET STRING
        :                     BF BE 51 85 2C CF B2 F7 39 94 6E 97 23 E4 02 18
        :                     C4 F8 22 84 B9 9A CF 56 14 88 FE 3F B7 A9 FB 49
 423    2:                           INTEGER 2000
 427   10:                           SEQUENCE {
 429    6:                             OBJECT IDENTIFIER '1 2 643 2 2 10'
 437    0:                             NULL
        :                             }
        :                           }
        :                         }
 439   29:                       SEQUENCE {
 441    6:                         OBJECT IDENTIFIER '1 2 643 2 2 21'
 449   19:                         SEQUENCE {
 451    8:                           OCTET STRING 6B 55 B6 6A 22 D3 C3 52
 461    7:                           OBJECT IDENTIFIER '1 2 643 2 2 31 1'
        :                           }
        :                         }
        :                       }
        :                     }
 470  485:                     [0]
        :                     57 2F 19 D6 01 DF DE CC 98 91 36 30 25 C6 2F BD
        :                     7D 3F 91 49 52 46 52 6D BE 0C C9 1A B4 24 45 45
        :                     40 8A C2 1A 2F 67 8D 9A 81 AF 6F 7B 41 B8 61 50
        :                     C9 54 F7 5C 12 51 99 22 54 F2 6E C9 33 F6 30 9B
        :                     A2 12 67 CE 85 BC D2 82 A6 E1 10 76 15 AA A9 A4
        :                     DE 49 6E 8F 0B 06 1B 95 3D F9 F7 BA 7E C0 1A FC
        :                     A3 BF 2B B3 D5 D4 45 59 22 04 51 8A 85 DC 45 07
        :                     35 FA ED 68 99 13 CB 2E 28 01 0F 6C 47 10 72 A2
        :                             [ Another 357 bytes skipped ]
        :                     }
        :                   }
        :                 }
        :               }
        :             }
        :           }
        :         }
        :       }
 959   86:   SEQUENCE {
 961   46:     SEQUENCE {
 963   10:       SEQUENCE {
 965    6:         OBJECT IDENTIFIER '1 2 643 2 2 9'
 973    0:         NULL
        :         }
 975   32:       OCTET STRING
        :         D8 D7 FF 30 9B 5E 9E 21 EC FF 3F 4E 3B 7A EE E9
        :         92 67 F1 8C 2E EF E6 5E F0 F6 9A 22 3A BB C9 99
        :       }
1009   32:     OCTET STRING
        :       C8 23 A9 7D 60 81 5F 79 E4 A0 45 54 03 5B 8C 5C
        :       50 B9 BD C5 1A 7F F9 8B 9B 4E 71 FD 2E DF C9 30
1043    2:     INTEGER 2000
        :     }
        :   }
```

Позиция 194: Закрытый ключ, зашифрованный ключом:

```
74 D7 93 BA 1B 2B A1 41 DA 96 97 A6 87 32 D3 8A
55 1F E3 C7 BC BA 16 7C BE DF D5 60 B5 F4 81 DF
```

Значение закрытого ключа, закодированное с соответствии с алгоритмом BASE64:

```
MEUCAQAwHAYGKoUDAgITMBIGByqFAwICIwEGByqFAwICHgEEIgQgFhRBz92CLtmJ
kQMtUnRz72zwNOLV8bHd+W5LdGBKDck=
```

Представление ASN.1 закрытого ключа:

```
0   69: SEQUENCE {
  2    1:   INTEGER 0
  5   28:   SEQUENCE {
  7    6:     OBJECT IDENTIFIER '1 2 643 2 2 19'
 15   18:     SEQUENCE {
 17    7:       OBJECT IDENTIFIER '1 2 643 2 2 35 1'
 26    7:       OBJECT IDENTIFIER '1 2 643 2 2 30 1'
        :       }
        :     }
 35   34:   OCTET STRING, encapsulates {
```

```
 37   32:     OCTET STRING
       :       16 14 41 CF DD 82 2E D9 89 91 03 2D 52 74 73 EF
       :       6C F0 34 E2 D5 F1 B1 DD F9 6E 4B 74 60 4A 0D C9
       :     }
       :   }
```

Позиция 470: Зашифрованный набор портфелей, включающий в себя один портфель сертификата (CertBag). Ключ шифрования:

```
A4 F9 2E 16 9D 35 C5 7F F6 A8 28 37 5D 6C 93 E6
5A 95 24 74 CA C5 3E E5 D7 AC BB BE 5E 56 42 09
```

Значение расшифрованного текста, закодированное с соответствии с алгоритмом BASE64:

```
MIIB4TCCAd0GCyqGSIb3DQEMCgEDoIIBmTCCAZUGCiqGSIb3DQEJFgGgggGFBIIB
gTCCAX0wggEqoAMCAQICAgEiMAoGBiqFAwICAwUAMDcxCzAJBgNVBAYTAlJVMSgw
JgYDVQQDDB9DQSBjZXJ0aWZpY2F0ZSAoUEtDUzEyIGV4YW1wbGUpMB4XDTEyMDYx
NDE5MzAxOVoXDTE3MDYxNDE5MzAxOVowOTELMAkGA1UEBhMCUlUxKjAoBgNVBAMM
IVRlc3QgY2VydGlmaWNhdGUgKFBLQ1MxMiBleGFtcGxlKTBjMBwGBiqFAwICEzAS
BgcqhQMCAiMBBgcqhQMCAh4BA0MABEB5UjAjACWYKB1AiBZp/nqjoGJUMvIe0n42
dpDJCeMU2nvI7WWMRy6tz/qiXorJrB3ZqnakgWzhhnTmiTTT2wGSoxowGDAJBgNV
HRMEAjAAMAsGA1UdDwQEAwIFoDAKBgYqhQMCAgMFAANBAK7JX1HqoVvXhZLKZ/Pf
gjkBCV+iaIIYjoxWJZMtF5mmg5dw+hHGyA/genARJnFxdtccSBHmCSAKhszhJzF3
U3IxMTAvBgkqhkiG9w0BCRUxIgQg4YvvSx3cWgryOsRD2YrtjCNInLQAeTY7iuW2
I3MN3jI=
```

Представление ASN.1 расшифрованного текста:

```
0  481: SEQUENCE {
   4  477:   SEQUENCE {
   8   11:     OBJECT IDENTIFIER pkcs-12-certBag (1 2 840 113549 1 12 10 1 3)
  21  409:     [0] {
  25  405:       SEQUENCE {
  29   10:         OBJECT IDENTIFIER
       :           x509Certificate (for PKCS #12) (1 2 840 113549 1 9 22 1)
  41  389:         [0] {
  45  385:           OCTET STRING, encapsulates {
  49  381:             SEQUENCE {
  53  298:               SEQUENCE {
  57    3:                 [0] {
  59    1:                   INTEGER 2
       :                   }
  62    2:                 INTEGER 290
  66   10:                 SEQUENCE {
  68    6:                   OBJECT IDENTIFIER '1 2 643 2 2 3'
  76    0:                   NULL
       :                   }
  78   55:                 SEQUENCE {
  80   11:                   SET {
  82    9:                     SEQUENCE {
  84    3:                       OBJECT IDENTIFIER countryName (2 5 4 6)
  89    2:                       PrintableString 'RU'
       :                       }
       :                     }
  93   40:                   SET {
  95   38:                     SEQUENCE {
  97    3:                       OBJECT IDENTIFIER commonName (2 5 4 3)
 102   31:                       UTF8String 'CA certificate (PKCS12 example)'
       :                       }
       :                     }
       :                   }
 135   30:                 SEQUENCE {
 137   13:                   UTCTime 14/06/2012 19:30:19 GMT
 152   13:                   UTCTime 14/06/2017 19:30:19 GMT
       :                   }
 167   57:                 SEQUENCE {
 169   11:                   SET {
 171    9:                     SEQUENCE {
 173    3:                       OBJECT IDENTIFIER countryName (2 5 4 6)
 178    2:                       PrintableString 'RU'
       :                       }
       :                     }
 182   42:                   SET {
 184   40:                     SEQUENCE {
 186    3:                       OBJECT IDENTIFIER commonName (2 5 4 3)
 191   33:                       UTF8String 'Test certificate (PKCS12 example)'
```

```
        :                       }
        :                     }
        :                   }
 226   99:                 SEQUENCE {
 228   28:                   SEQUENCE {
 230    6:                     OBJECT IDENTIFIER '1 2 643 2 2 19'
 238   18:                     SEQUENCE {
 240    7:                       OBJECT IDENTIFIER '1 2 643 2 2 35 1'
 249    7:                       OBJECT IDENTIFIER '1 2 643 2 2 30 1'
        :                       }
        :                     }
 258   67:                   BIT STRING, encapsulates {
 261   64:                     OCTET STRING
        :                     79 52 30 23 00 25 98 28 1D 40 88 16 69 FE 7A A3
        :                     A0 62 54 32 F2 1E D2 7E 36 76 90 C9 09 E3 14 DA
        :                     7B C8 ED 65 8C 47 2E AD CF FA A2 5E 8A C9 AC 1D
        :                     D9 AA 76 A4 81 6C E1 86 74 E6 89 34 D3 DB 01 92
        :                       }
        :                     }
 327   26:                   [3] {
 329   24:                     SEQUENCE {
 331    9:                       SEQUENCE {
 333    3:                         OBJECT IDENTIFIER basicConstraints (2 5 29 19)
 338    2:                         OCTET STRING, encapsulates {
 340    0:                           SEQUENCE {}
        :                           }
        :                         }
 342   11:                       SEQUENCE {
 344    3:                         OBJECT IDENTIFIER keyUsage (2 5 29 15)
 349    4:                         OCTET STRING, encapsulates {
 351    2:                           BIT STRING 5 unused bits
        :                             '101'B
        :                           }
        :                         }
        :                       }
        :                     }
        :                   }
 355   10:                 SEQUENCE {
 357    6:                   OBJECT IDENTIFIER '1 2 643 2 2 3'
 365    0:                   NULL
        :                   }
 367   65:                 BIT STRING
        :                   AE C9 5F 51 EA A1 5B D7 85 92 CA 67 F3 DF 82 39
        :                   01 09 5F A2 68 82 18 8E 8C 56 25 93 2D 17 99 A6
        :                   83 97 70 FA 11 C6 C8 0F E0 7A 70 11 26 71 71 76
        :                   D7 1C 48 11 E6 09 20 0A 86 CC E1 27 31 77 53 72
        :                 }
        :               }
        :             }
        :           }
        :         }
 434   49:       SET {
 436   47:         SEQUENCE {
 438    9:           OBJECT IDENTIFIER
        :             localKeyID (for PKCS #12) (1 2 840 113549 1 9 21)
 449   34:           SET {
 451   32:             OCTET STRING
        :               E1 8B EF 4B 1D DC 5A 0A F2 3A C4 43 D9 8A ED 8C
        :               23 48 9C B4 00 79 36 3B 8A E5 B6 23 73 0D DE 32
        :             }
        :           }
        :         }
        :       }
        :     }
        :   }
```

Позиция 975: Значение функции HMAC_GOSTR3411, вычисленное на ключе:

```
A0 E5 61 A9 45 9D 89 2B 1B 17 DC 77 1D 0A 7D A9
F7 6A C7 86 A9 31 E1 D3 D1 4D 22 23 E1 CD 16 9D
```

Ключ получен взятием последних 32 байт из ключевого материала длиной 96 байт, сгенерированного функцией PBKDF2:

```
63 73 4E B2 CA 03 8A F5 8D A1 C0 78 FD C7 87 DB
ED D9 B3 D3 6E 9C 6E B3 66 2E 2F 35 F3 3B 01 23
69 4D AA 59 BE A6 F1 3A 8D 54 5D 6B 51 DE 9A A8
FE F5 DA B2 1D 50 C2 A6 F5 38 60 8C AB 5F 53 19
A0 E5 61 A9 45 9D 89 2B 1B 17 DC 77 1D 0A 7D A9
```

```
F7 6A C7 86 A9 31 E1 D3 D1 4D 22 23 E1 CD 16 9D
```

7.3. Контрольный пример 2

В данном разделе приведен пример контейнера, зашифрованного на выработанном с помощью сертификата получателя ключе и подписанного ключом отправителя в соответствии с изложенной в данном документе схемой защиты на ключе, выработанном по протоколу Диффи-Хеллмана.

Значение контейнера, закодированное с соответствии с алгоритмом BASE64:

```
MIIG9wIBAzCCBvAGCSqGSIb3DQEHAqCCBuEwggbdAgEBMQwwCgYGKoUDAgIJBQAw
ggROBgkqhkiG9w0BBwGgggQ/BIIEOzCCBDcwggQzBgkqhkiG9w0BBwOgggQkMIIE
IAIBADGCAQ4wggEKAgEAMD0wNzELMAkGA1UEBhMCUlUxKDAmBgNVBAMMH0NBIGNl
cnRpZmljYXRlIChQS0NTMTIgZXhhbXBsZSkCAgEgMBwGBiqFAwICEzASBgcqhQMC
AiQABgcqhQMCAh4BBIGnMIGkMCgEIAGMe6IJ3eXR1pyr+xM15mY82dhNVBQUrJuE
ZbFnqnV6BARFro2ZoHgGByqFAwICHwGgYzAcBgYqhQMCAhMwEgYHKoUDAgIkAAYH
KoUDAgIeAQNDAARAAMjagM5tZaUi0eIW9Fzy5KOOFLXxaOiYvDl3hVQgyqtZaNdL
JMVYCcp6BSsEwJzQdKtnVQsYucsrJkI4USM6twQILd4b1Osjv1YwggMHBgkqhkiG
9w0BBwEwHQYGKoUDAgIVMBMECO5jVyCHNqshBgcqhQMCAh8BgIIC2fsgXMb2pLM/
bARkShmJgKENv9rlfvhAfxDF+zrVpoX6zBJTmOENQR+yGcsSpMWS8iByLIaIs47I
V/ZC/1tYxSaAi3gchNiS4GlnEQI1Qe/raYF3j8XuMT/hPfxijMT68xB92s/oAqH0
Mf/vrBj6fonF6Qjm8KkL3xz0sQKt+WFY/iI9HWYXm9zmJVxQ2SN4S5qacYm+SaNY
DE+Vu+gZtfT0tPBRMC1GdB1MjKzvPA9ETfE0Cby99E46gYrUYRQXiUs6slH7En+M
BVK0G8R8Q7lbo1H/NGXce2WS0Qll09ACz485TY85EHw8I9xCAFKWwk0wIjl+DuaF
46YKA/DmbrhMPCOl53thXYt4lzttcBsyRKeDsEBVolBxNmkXlSQa3Wyq3WPefNth
6cTbSK7Xybrm5QPHtxVQaWLYq4b6qsMt9aQC6s4tw7EoInJ/lEQTRF0Xgrbogl4l
aTdaoiwV/x5kF33+yffgnVR1GNbIz6J94CQwSE9Y/XWJ/gibCdyahXkKn3q+88L/
7ov9WGcdCZ59oTCmhOcyqSTYIKhPndO2lbIXCacKtAygAJVskTauGHZ2WlT4nRx0
PdXfd0AqHkLjRRK3Veu0pNQtW+itHyxJJgECPEKwBdV40I1HQoR7qxo6YmLCK4Ey
awNro32rw19iyyDb3MUTVg4ks2QtkEwBFCqsC3LTN2TViUsRJKxo7D9K517q0mZS
T1LVnUr0DglwYEE2riDPMhBrzOWGPHO83HUfATr3lxKSbQOEpumQTuAgp4w8X0tY
g/QZe501YL/v2VwJmAWNYaGFy6/o9Txyhy3KGtTqmWKIy4kChPYYM0CA1Bi4zm1X
yWI7dSKhdBSfImWGEiXpV99AnK3v05Q+UGSPwM9ZrPNzyklfsd2cWVzJn2d7jz/D
1RNRFnm1tHOkb7TDLXXZOUmtCftV2EDDyoNawzzyMj3Dax9WqibdU4TF1XQS0Mjz
IqCCAYcwggGDMIIBMKADAgECAgIBHzAKBgYqhQMCAgMFADA3MQswCQYDVQQGEwJS
VTEoMCYGA1UEAwwfQ0EgY2VydGlmaWNhdGUgKFBLQ1MxMiBleGFtcGxlKTAeFw0x
MjA2MTQxMzIzMjVaFw0xNzA2MTQxMzIzMjVaMD8xCzAJBgNVBAYTAlJVMTAwLgYD
VQQDDCdQRlgtaXNzdWVyIGNlcnRpZmljYXRlIChQS0NTMTIgZXhhbXBsZSkwYzAc
BgYqhQMCAhMwEgYHKoUDAgIjAQYHKoUDAgIeAQNDAARAwAG9ZjvGxcd75HBztpMq
U1EnQioI/kNh8q+sfotGqhTeKqoCgkQM9ZKRco/r97l55wRkTUvmH7cQL+xlvf7g
16MaMBgwCQYDVR0TBAIwADALBgNVHQ8EBAMCBaAwCgYGKoUDAgIDBQADQQBa7s+a
arQxyzvAbQjsLhRq8+J/CN5CJkrIsJqC+XsjtqAkTaASRjYlLnqa2kBlcj6m+wJz
id5g3AuyiqFJ/3JxMYHsMIHpAgEBMD0wNzELMAkGA1UEBhMCUlUxKDAmBgNVBAMM
H0NBIGNlcnRpZmljYXRlIChQS0NTMTIgZXhhbXBsZSkCAgEfMAoGBiqFAwICCQUA
oEswGAYJKoZIhvcNAQkDMQsGCSqGSIb3DQEHATAvBgkqhkiG9w0BCQQxIgQg54zs
FimpJAw5HrFrkRk33mtnqSIefxWYpqF1/a6jDJMwCgYGKoUDAgITBQAEQC40g80b
nTSh3xEjk45Ml30Es2QlZemwrpLgM/35NRZ5WFii0QBwZopHdmhwXnnZftVnxCFN
G3iM4SEARe4NoGY=
```

Представление ASN.1:

```
0 1783: SEQUENCE {
   4    1:   INTEGER 3
   7 1776:   SEQUENCE {
  11    9:     OBJECT IDENTIFIER signedData (1 2 840 113549 1 7 2)
  22 1761:     [0] {
  26 1757:       SEQUENCE {
  30    1:         INTEGER 1
  33   12:         SET {
  35   10:           SEQUENCE {
  37    6:             OBJECT IDENTIFIER '1 2 643 2 2 9'
  45    0:             NULL
        :             }
        :           }
  47 1102:         SEQUENCE {
  51    9:           OBJECT IDENTIFIER data (1 2 840 113549 1 7 1)
  62 1087:           [0] {
  66 1083:             OCTET STRING, encapsulates {
  70 1079:               SEQUENCE {
  74 1075:                 SEQUENCE {
  78    9:                   OBJECT IDENTIFIER
        :                     envelopedData (1 2 840 113549 1 7 3)
```

```
 89 1060:                  [0] {
 93 1056:                    SEQUENCE {
 97    1:                      INTEGER 0
100  270:                      SET {
104  266:                        SEQUENCE {
108    1:                          INTEGER 0
111   61:                          SEQUENCE {
113   55:                            SEQUENCE {
115   11:                              SET {
117    9:                                SEQUENCE {
119    3:                                  OBJECT IDENTIFIER
       :                                    countryName (2 5 4 6)
124    2:                                  PrintableString 'RU'
       :                                  }
       :                                }
128   40:                              SET {
130   38:                                SEQUENCE {
132    3:                                  OBJECT IDENTIFIER
       :                                    commonName (2 5 4 3)
137   31:                                  UTF8String 'CA certificate (PKCS12 example)'
       :                                  }
       :                                }
       :                              }
170    2:                            INTEGER 288
       :                            }
174   28:                          SEQUENCE {
176    6:                            OBJECT IDENTIFIER '1 2 643 2 2 19'
184   18:                            SEQUENCE {
186    7:                              OBJECT IDENTIFIER '1 2 643 2 2 36 0'
195    7:                              OBJECT IDENTIFIER '1 2 643 2 2 30 1'
       :                              }
       :                            }
204  167:                          OCTET STRING, encapsulates {
207  164:                            SEQUENCE {
210   40:                              SEQUENCE {
212   32:                                OCTET STRING
       :                  01 8C 7B A2 09 DD E5 D1 D6 9C AB FB 13 35 E6 66
       :                  3C D9 D8 4D 54 14 14 AC 9B 84 65 B1 67 AA 75 7A
246    4:                                OCTET STRING 45 AE 8D 99
       :                                }
252  120:                              [0] {
254    7:                                OBJECT IDENTIFIER '1 2 643 2 2 31 1'
263   99:                                [0] {
265   28:                                  SEQUENCE {
267    6:                                    OBJECT IDENTIFIER '1 2 643 2 2 19'
275   18:                                    SEQUENCE {
277    7:                                      OBJECT IDENTIFIER '1 2 643 2 2 36 0'
286    7:                                      OBJECT IDENTIFIER '1 2 643 2 2 30 1'
       :                                      }
       :                                    }
295   67:                                  BIT STRING, encapsulates {
298   64:                                    OCTET STRING
       :                  00 C8 DA 80 CE 6D 65 A5 22 D1 E2 16 F4 5C F2 E4
       :                  A3 8E 14 B5 F1 68 E8 98 BC 39 77 85 54 20 CA AB
       :                  59 68 D7 4B 24 C5 58 09 CA 7A 05 2B 04 C0 9C D0
       :                  74 AB 67 55 0B 18 B9 CB 2B 26 42 38 51 23 3A B7
       :                                    }
       :                                  }
364    8:                                OCTET STRING 2D DE 1B D4 EB 23 BF 56
       :                                }
       :                              }
       :                            }
       :                          }
       :                        }
374  775:                      SEQUENCE {
378    9:                        OBJECT IDENTIFIER data (1 2 840 113549 1 7 1)
389   29:                        SEQUENCE {
391    6:                          OBJECT IDENTIFIER '1 2 643 2 2 21'
399   19:                          SEQUENCE {
401    8:                            OCTET STRING EE 63 57 20 87 36 AB 21
411    7:                            OBJECT IDENTIFIER '1 2 643 2 2 31 1'
       :                            }
       :                          }
420  729:                        [0]
       :                  FB 20 5C C6 F6 A4 B3 3F 6C 04 64 4A 19 89 80 A1
       :                  0D BF DA E5 7E F8 40 7F 10 C5 FB 3A D5 A6 85 FA
       :                  CC 12 53 98 E1 0D 41 1F B2 19 CB 12 A4 C5 92 F2
       :                  20 72 2C 86 88 B3 8E C8 57 F6 42 FF 5B 58 C5 26
       :                  80 8B 78 1C 84 D8 92 E0 69 67 11 02 35 41 EF EB
       :                  69 81 77 8F C5 EE 31 3F E1 3D FC 62 8C C4 FA F3
```

```
         :                   10 7D DA CF E8 02 A1 F4 31 FF EF AC 18 FA 7E 89
         :                   C5 E9 08 E6 F0 A9 0B DF 1C F4 B1 02 AD F9 61 58
         :                           [ Another 601 bytes skipped ]
         :                         }
         :                       }
         :                     }
         :                   }
         :                 }
         :               }
         :             }
         :           }
1153  391:         [0] {
1157  387:           SEQUENCE {
1161  304:             SEQUENCE {
1165    3:               [0] {
1167    1:                 INTEGER 2
         :                 }
1170    2:               INTEGER 287
1174   10:               SEQUENCE {
1176    6:                 OBJECT IDENTIFIER '1 2 643 2 2 3'
1184    0:                 NULL
         :                 }
1186   55:               SEQUENCE {
1188   11:                 SET {
1190    9:                   SEQUENCE {
1192    3:                     OBJECT IDENTIFIER countryName (2 5 4 6)
1197    2:                     PrintableString 'RU'
         :                     }
         :                   }
1201   40:                 SET {
1203   38:                   SEQUENCE {
1205    3:                     OBJECT IDENTIFIER commonName (2 5 4 3)
1210   31:                     UTF8String 'CA certificate (PKCS12 example)'
         :                     }
         :                   }
         :                 }
1243   30:               SEQUENCE {
1245   13:                 UTCTime 14/06/2012 13:23:25 GMT
1260   13:                 UTCTime 14/06/2017 13:23:25 GMT
         :                 }
1275   63:               SEQUENCE {
1277   11:                 SET {
1279    9:                   SEQUENCE {
1281    3:                     OBJECT IDENTIFIER countryName (2 5 4 6)
1286    2:                     PrintableString 'RU'
         :                     }
         :                   }
1290   48:                 SET {
1292   46:                   SEQUENCE {
1294    3:                     OBJECT IDENTIFIER commonName (2 5 4 3)
1299   39:                     UTF8String 'PFX-issuer certificate (PKCS12 example)'
         :                     }
         :                   }
         :                 }
1340   99:               SEQUENCE {
1342   28:                 SEQUENCE {
1344    6:                   OBJECT IDENTIFIER '1 2 643 2 2 19'
1352   18:                   SEQUENCE {
1354    7:                     OBJECT IDENTIFIER '1 2 643 2 2 35 1'
1363    7:                     OBJECT IDENTIFIER '1 2 643 2 2 30 1'
         :                     }
         :                   }
1372   67:                 BIT STRING, encapsulates {
1375   64:                   OCTET STRING
         :                   C0 01 BD 66 3B C6 C5 C7 7B E4 70 73 B6 93 2A 53
         :                   51 27 42 2A 08 FE 43 61 F2 AF AC 7E 8B 46 AA 14
         :                   DE 2A AA 02 82 44 0C F5 92 91 72 8F EB F7 B9 79
         :                   E7 04 64 4D 4B E6 1F B7 10 2F EC 65 BD FE E0 D7
         :                   }
         :                 }
1441   26:               [3] {
1443   24:                 SEQUENCE {
1445    9:                   SEQUENCE {
1447    3:                     OBJECT IDENTIFIER basicConstraints (2 5 29 19)
1452    2:                     OCTET STRING, encapsulates {
1454    0:                       SEQUENCE {}
         :                       }
         :                     }
1456   11:                   SEQUENCE {
1458    3:                     OBJECT IDENTIFIER keyUsage (2 5 29 15)
```

```
1463    4:                     OCTET STRING, encapsulates {
1465    2:                       BIT STRING 5 unused bits
          :                         '101'B
          :                       }
          :                     }
          :                   }
          :                 }
          :               }
1469   10:             SEQUENCE {
1471    6:               OBJECT IDENTIFIER '1 2 643 2 2 3'
1479    0:               NULL
          :               }
1481   65:             BIT STRING
          :               5A EE CF 9A 6A B4 31 CB 3B C0 6D 08 EC 2E 14 6A
          :               F3 E2 7F 08 DE 42 26 4A C8 B0 9A 82 F9 7B 23 B6
          :               A0 24 4D A0 12 46 36 25 2E 7A 9A DA 40 65 72 3E
          :               A6 FB 02 73 89 DE 60 DC 0B B2 8A A1 49 FF 72 71
          :             }
          :           }
1548  236:         SET {
1551  233:           SEQUENCE {
1554    1:             INTEGER 1
1557   61:             SEQUENCE {
1559   55:               SEQUENCE {
1561   11:                 SET {
1563    9:                   SEQUENCE {
1565    3:                     OBJECT IDENTIFIER countryName (2 5 4 6)
1570    2:                     PrintableString 'RU'
          :                     }
          :                   }
1574   40:                 SET {
1576   38:                   SEQUENCE {
1578    3:                     OBJECT IDENTIFIER commonName (2 5 4 3)
1583   31:                     UTF8String 'CA certificate (PKCS12 example)'
          :                     }
          :                   }
          :                 }
1616    2:               INTEGER 287
          :               }
1620   10:             SEQUENCE {
1622    6:               OBJECT IDENTIFIER '1 2 643 2 2 9'
1630    0:               NULL
          :               }
1632   75:             [0] {
1634   24:               SEQUENCE {
1636    9:                 OBJECT IDENTIFIER contentType (1 2 840 113549 1 9 3)
1647   11:                 SET {
1649    9:                   OBJECT IDENTIFIER data (1 2 840 113549 1 7 1)
          :                   }
          :                 }
1660   47:               SEQUENCE {
1662    9:                 OBJECT IDENTIFIER
          :                   messageDigest (1 2 840 113549 1 9 4)
1673   34:                 SET {
1675   32:                   OCTET STRING
          :                   E7 8C EC 16 29 A9 24 0C 39 1E B1 6B 91 19 37 DE
          :                   6B 67 A9 22 1E 7F 15 98 A6 A1 75 FD AE A3 0C 93
          :                   }
          :                 }
          :               }
1709   10:             SEQUENCE {
1711    6:               OBJECT IDENTIFIER '1 2 643 2 2 19'
1719    0:               NULL
          :               }
1721   64:             OCTET STRING
          :               2E 34 83 CD 1B 9D 34 A1 DF 11 23 93 8E 4C 97 7D
          :               04 B3 64 25 65 E9 B0 AE 92 E0 33 FD F9 35 16 79
          :               58 58 A2 D1 00 70 66 8A 47 76 68 70 5E 79 D9 7E
          :               D5 67 C4 21 4D 1B 78 8C E1 21 00 45 EE 0D A0 66
          :             }
          :           }
          :         }
          :       }
          :     }
          :   }
```

Позиция 470: Зашифрованный набор портфелей, включающий в себя портфель сертификата (CertBag) и портфель закрытого ключа (KeyBag). Ключ шифрования:

```
CD 7C 2F C8 2E 69 D2 8E E5 98 11 0E 56 FF 67 20
13 82 D4 16 6E 16 5D 10 1C BA 50 FC E0 62 EF A2
```

Значение расшифрованного текста, закодированное с соответствии с алгоритмом BASE64:

```
MIIC1TCB8QYLKoZIhvcNAQwKAQGgga4wgasCAQAwHAYGKoUDAgITMBIGByqFAwIC
IwEGByqFAwICHgEEgYcwgYQEQMLb6fp64gRp1N+OIF1DPv8lAWGimKuNtCX53IBv
o8bav1UKWxDTZBQH/HjECxd7rd1etk4d4JoVdg3T2wCzrnEEQHlSMCMAJZgoHUCI
Fmn+eqOgYlQy8h7SfjZ2kMkJ4xTae8jtZYxHLq3P+qJeismsHdmqdqSBbOGGdOaJ
NNPbAZIxMTAvBgkqhkiG9w0BCRUxIgQg4YvvSx3cWgryOsRD2YrtjCNInLQAeTY7
iuW2I3MN3jIwggHdBgsqhkiG9w0BDAoBA6CCAZkwggGVBgoqhkiG9w0BCRYBoIIB
hQSCAYEwggF9MIIBKqADAgECAgIBIjAKBgYqhQMCAgMFADA3MQswCQYDVQQGEwJS
VTEoMCYGA1UEAwwfQ0EgY2VydGlmaWNhdGUgKFBLQ1MxMiBleGFtcGxlKTAeFw0x
MjA2MTQxOTMwMTlaFw0xNzA2MTQxOTMwMTlaMDkxCzAJBgNVBAYTAlJVMSowKAYD
VQQDDCFUZXN0IGNlcnRpZmljYXRlIChQS0NTMTIgZXhhbXBsZSkwYzAcBgYqhQMC
AhMwEgYHKoUDAgIjAQYHKoUDAgIeAQNDAARAeVIwIwAlmCgdQIgWaf56o6BiVDLy
HtJ+NnaQyQnjFNp7yO1ljEcurc/6ol6Kyawd2ap2pIFs4YZ05ok009sBkqMaMBgw
CQYDVR0TBAIwADALBgNVHQ8EBAMCBaAwCgYGKoUDAgIDBQADQQCuyV9R6qFb14WS
ymfz34I5AQlfomiCGI6MViWTLReZpoOXcPoRxsgP4HpwESZxcXbXHEgR5gkgCobM
4Scxd1NyMTEwLwYJKoZIhvcNAQkVMSIEIOGL70sd3FoK8jrEQ9mK7YwjSJy0AHk2
O4rltiNzDd4y
```

Представление ASN.1 расшифрованного текста:

```
0  725: SEQUENCE {
   4  241:   SEQUENCE {
   7   11:     OBJECT IDENTIFIER pkcs-12-keyBag (1 2 840 113549 1 12 10 1 1)
  20  174:     [0] {
  23  171:       SEQUENCE {
  26    1:         INTEGER 0
  29   28:         SEQUENCE {
  31    6:           OBJECT IDENTIFIER '1 2 643 2 2 19'
  39   18:           SEQUENCE {
  41    7:             OBJECT IDENTIFIER '1 2 643 2 2 35 1'
  50    7:             OBJECT IDENTIFIER '1 2 643 2 2 30 1'
        :             }
        :           }
  59  135:         OCTET STRING, encapsulates {
  62  132:           SEQUENCE {
  65   64:             OCTET STRING
        :               C2 DB E9 FA 7A E2 04 69 D4 DF 8E 20 5D 43 3E FF
        :               25 01 61 A2 98 AB 8D B4 25 F9 DC 80 6F A3 C6 DA
        :               BF 55 0A 5B 10 D3 64 14 07 FC 78 C4 0B 17 7B AD
        :               DD 5E B6 4E 1D E0 9A 15 76 0D D3 DB 00 B3 AE 71
 131   64:             OCTET STRING
        :               79 52 30 23 00 25 98 28 1D 40 88 16 69 FE 7A A3
        :               A0 62 54 32 F2 1E D2 7E 36 76 90 C9 09 E3 14 DA
        :               7B C8 ED 65 8C 47 2E AD CF FA A2 5E 8A C9 AC 1D
        :               D9 AA 76 A4 81 6C E1 86 74 E6 89 34 D3 DB 01 92
        :             }
        :           }
        :         }
        :       }
 197   49:     SET {
 199   47:       SEQUENCE {
 201    9:         OBJECT IDENTIFIER
        :           localKeyID (for PKCS #12) (1 2 840 113549 1 9 21)
 212   34:         SET {
 214   32:           OCTET STRING
        :             E1 8B EF 4B 1D DC 5A 0A F2 3A C4 43 D9 8A ED 8C
        :             23 48 9C B4 00 79 36 3B 8A E5 B6 23 73 0D DE 32
        :           }
        :         }
        :       }
        :     }
 248  477:   SEQUENCE {
 252   11:     OBJECT IDENTIFIER pkcs-12-certBag (1 2 840 113549 1 12 10 1 3)
 265  409:     [0] {
 269  405:       SEQUENCE {
 273   10:         OBJECT IDENTIFIER
        :           x509Certificate (for PKCS #12) (1 2 840 113549 1 9 22 1)
 285  389:         [0] {
 289  385:           OCTET STRING, encapsulates {
 293  381:             SEQUENCE {
 297  298:               SEQUENCE {
 301    3:                 [0] {
 303    1:                   INTEGER 2
```

```
       :                 }
306    2:               INTEGER 290
310   10:               SEQUENCE {
312    6:                 OBJECT IDENTIFIER '1 2 643 2 2 3'
320    0:                 NULL
       :                 }
322   55:               SEQUENCE {
324   11:                 SET {
326    9:                   SEQUENCE {
328    3:                     OBJECT IDENTIFIER countryName (2 5 4 6)
333    2:                     PrintableString 'RU'
       :                     }
       :                   }
337   40:                 SET {
339   38:                   SEQUENCE {
341    3:                     OBJECT IDENTIFIER commonName (2 5 4 3)
346   31:                     UTF8String 'CA certificate (PKCS12 example)'
       :                     }
       :                   }
       :                 }
379   30:               SEQUENCE {
381   13:                 UTCTime 14/06/2012 19:30:19 GMT
396   13:                 UTCTime 14/06/2017 19:30:19 GMT
       :                 }
411   57:               SEQUENCE {
413   11:                 SET {
415    9:                   SEQUENCE {
417    3:                     OBJECT IDENTIFIER countryName (2 5 4 6)
422    2:                     PrintableString 'RU'
       :                     }
       :                   }
426   42:                 SET {
428   40:                   SEQUENCE {
430    3:                     OBJECT IDENTIFIER commonName (2 5 4 3)
435   33:                     UTF8String 'Test certificate (PKCS12 example)'
       :                     }
       :                   }
       :                 }
470   99:               SEQUENCE {
472   28:                 SEQUENCE {
474    6:                   OBJECT IDENTIFIER '1 2 643 2 2 19'
482   18:                   SEQUENCE {
484    7:                     OBJECT IDENTIFIER '1 2 643 2 2 35 1'
493    7:                     OBJECT IDENTIFIER '1 2 643 2 2 30 1'
       :                     }
       :                   }
502   67:                 BIT STRING, encapsulates {
505   64:                   OCTET STRING
       :                   79 52 30 23 00 25 98 28 1D 40 88 16 69 FE 7A A3
       :                   A0 62 54 32 F2 1E D2 7E 36 76 90 C9 09 E3 14 DA
       :                   7B C8 ED 65 8C 47 2E AD CF FA A2 5E 8A C9 AC 1D
       :                   D9 AA 76 A4 81 6C E1 86 74 E6 89 34 D3 DB 01 92
       :                   }
       :                 }
571   26:               [3] {
573   24:                 SEQUENCE {
575    9:                   SEQUENCE {
577    3:                     OBJECT IDENTIFIER basicConstraints (2 5 29 19)
582    2:                     OCTET STRING, encapsulates {
584    0:                       SEQUENCE {}
       :                       }
       :                     }
586   11:                   SEQUENCE {
588    3:                     OBJECT IDENTIFIER keyUsage (2 5 29 15)
593    4:                     OCTET STRING, encapsulates {
595    2:                       BIT STRING 5 unused bits
       :                         '101'B
       :                       }
       :                     }
       :                   }
       :                 }
       :               }
599   10:             SEQUENCE {
601    6:               OBJECT IDENTIFIER '1 2 643 2 2 3'
609    0:               NULL
       :               }
611   65:             BIT STRING
       :               AE C9 5F 51 EA A1 5B D7 85 92 CA 67 F3 DF 82 39
       :               01 09 5F A2 68 82 18 8E 8C 56 25 93 2D 17 99 A6
       :               83 97 70 FA 11 C6 C8 0F E0 7A 70 11 26 71 71 76
```

```
        :                 D7 1C 48 11 E6 09 20 0A 86 CC E1 27 31 77 53 72
        :               }
        :             }
        :           }
        :         }
        :       }
678   49:     SET {
680   47:       SEQUENCE {
682    9:         OBJECT IDENTIFIER
        :           localKeyID (for PKCS #12) (1 2 840 113549 1 9 21)
693   34:         SET {
695   32:           OCTET STRING
        :             E1 8B EF 4B 1D DC 5A 0A F2 3A C4 43 D9 8A ED 8C
        :             23 48 9C B4 00 79 36 3B 8A E5 B6 23 73 0D DE 32
        :           }
        :         }
        :       }
        :     }
        :   }
```

Литература

1. PKCS #8 v.1.2 Private-Key Information Syntax Standard, 1993 г.
2. PKCS#12 v.1.0 Personal Information Exchange Syntax, 1999 г.
3. RFC5652: Cryptographic Message Syntax.
4. RFC4357: 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.
5. RFC4491: 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.
6. RFC4490: Using the GOST 28147–89, GOST R 34.11–94, GOST R 34.10–94, and GOST R 34.10–2001 Algorithms with Cryptographic Message Syntax (CMS).
7. Парольная защита с использованием алгоритмов ГОСТ. Дополнения к PKCS#5 (версия 1.0).