

ПРОЕКТ

ЗАТВЕРДЖЕНО

Наказ Міністерства юстиції України,
Адміністрації Державної служби
спеціального зв'язку та захисту
інформації України

№ _____/_____

ВИМОГИ

**до алгоритмів формування ключів шифрування ключів та захисту особистих
ключів електронного цифрового підпису та особистих ключів шифрування**

I. Загальні положення

1. Ці Вимоги визначають алгоритми формування ключів шифрування ключів на основі паролльної інформації та захисту особистих ключів електронного цифрового підпису та особистих ключів шифрування з використанням алгоритмів ДСТУ ГОСТ 28147:2009 «Системы обработки информации. Защита криптографическая. Алгоритмы криптографического преобразования», затвердженого наказом Державного комітету України з питань технічного регулювання та споживчої політики від 22 грудня 2008 року № 495 (далі – ГОСТ 28147:2009) та ГОСТ 34.311-95 «Информационная технология. Криптографическая защита информации. Функция хэширования», затвердженого наказом Державного комітету України з питань технічного регулювання та споживчої політики від 21 жовтня 1997 року № 640 (далі – ГОСТ 34.311-95).

2. Формати даних представлено у нотації ASN.1, визначеній у міжнародному стандарті ISO/IEC 8824 «Information technology – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1)» ДСТУ ISO/IEC 8824-3:2008 «Інформаційні технології. Нотація абстрактного синтаксису 1 (ASN.1)». Частина 3. Специфікація обмежень (ISO/IEC 8824-3:2002, IDT), затвердженому наказом Державного комітету України з питань технічного регулювання та споживчої політики від 26 грудня 2008 року № 508 (із змінами).

3. Усі структури даних кодують за правилами DER згідно з міжнародним стандартом ISO/IEC 8825-1:2002 «Information technology – ASN.1 encoding Rules – Part 1: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)» & AMD1: 2004 «Support for EXTENDED-XER».

4. Ці Вимоги засновані на національних стандартах України ДСТУ ГОСТ 28147:2009, та ГОСТ 34.311-95 та міжнародних рекомендаціях RFC 2898 «Password-Based Cryptography Specification (PKCS#5)», September 2000 (далі – RFC 2898), RFC 2104 «HMAC: Keyed-Hashing for Message Authentication», February 1997 (далі – RFC 2104).

5. Ці Вимоги не дублюють стандарти ДСТУ ГОСТ 28147:2009, ГОСТ 34.311-95 та міжнародні рекомендації RFC 2898, а описують положення цих стандартів і рекомендацій та алгоритми формування ключів шифрування ключів на основі паролльної інформації та захисту особистих ключів електронного цифрового підпису та особистих ключів шифрування з використанням алгоритмів, визначених ДСТУ ГОСТ 28147:2009 та ГОСТ 34.311-95. У разі виникнення розбіжностей між положеннями зазначених стандартів і рекомендацій та положеннями цих Вимог, застосовуються положення цих Вимог.

6. Положення цих Вимог є обов'язковими для засобів криптографічного захисту конфіденційної інформації, державних інформаційних ресурсів або інформації, вимога щодо захисту якої встановлена законом (далі – засоби шифрування), а також надійних засобів електронного цифрового підпису. Правильність реалізації алгоритмів формування ключів шифрування ключів на основі паролльної інформації та захисту особистих ключів електронного цифрового підпису та особистих ключів шифрування з використанням алгоритмів ДСТУ ГОСТ 28147:2009 та ГОСТ 34.311-95 у надійних засобах електронного цифрового підпису та у засобах шифрування підтверджується сертифікатом відповідності або позитивним експертним висновком за результатами державної експертизи у сфері криптографічного захисту інформації, виданими спеціально уповноваженим центральним органом виконавчої влади з питань організації спеціального зв'язку та захисту інформації.

II. Алгоритм формування ключа шифрування ключа на основі паролльної інформації (PBKDF-функція)

1. Алгоритм формування ключа шифрування ключа на основі паролльної інформації (PBKDF-функція) призначений для формування симетричного ключа (DK) на основі паролльної інформації, випадкових даних та числа ітерацій.

2. Процес реалізації PBKDF-функції передбачає здійснення таких дій:
формування ключового матеріалу (KM) з використанням паролльної інформації, випадкових даних та числа ітерацій;

створення симетричного ключа для заданого алгоритму шифрування або контролю цілісності даних на основі ключового матеріалу.

3. Ці Вимоги визначають функцію формування ключа на основі функції PBKDF2 з використанням псевдовипадкової функції (PRF-функції), що базується на алгоритмі гешування за ГОСТ 34.311-95 та RFC 2104 (далі – HMAC_GOST34311):

$DK = \text{PBKDF}(P, S, c, dkLen)$,

де P – пароль, символний рядок у кодуванні Unicode UTF-8,

S – випадкові дані,

c – число ітерацій алгоритму,

$dkLen$ – необхідна довжина вихідної послідовності в байтах.

4. Під час формування симетричного ключа на основі паролльної інформації вчиняються такі дії:

виконується перевірка умови $dkLen > (2^{32} - 1) * hLen$,

де $hLen$ – довжина вихідного значення функції HMAC_GOST34311.

У разі виконання умови подальші дії не виконуються, у зв'язку з недопустимим значенням довжини ключа;

обчислюються значення:

$n = V(dkLen/hLen)$,

де V – функція округлення аргументу функції до найменшого натурального числа, більшого за аргумент функції,

$T_1 = F(P, S, c, 1)$,

$T_2 = F(P, S, c, 2)$,

...

$T_n = F(P, S, c, n)$,

де $F(P, S, c, i) = U_1 \oplus U_2 \oplus \dots \oplus U_c$,

$U_1 = \text{HMAC_GOST34311}(P, S \parallel \text{INT}(i))$,

$U_2 = \text{HMAC_GOST34311}(P, U_1)$,

...

$U_c = \text{HMAC_GOST34311}(P, U_{c-1})$,

\oplus – порозрядне додавання за модулем 2,

$\text{INT}(i)$ – представлення цілого числа « i » чотирма байтами зі старшим байтом зліва;

в результаті конкатенації $\{T_i\}$ з урізанням T_n до необхідної довжини $dkLen$ формується Ключ DK :

$DK = T_1 \parallel T_2 \parallel \dots \parallel T_n$.

5. Формування ключа шифрування ключа на основі паролльної інформації здійснюється згідно з додатком до цих Вимог.

III. Алгоритми захисту особистих ключів електронного цифрового підпису та особистих ключів шифрування (далі – даних) на основі парольної інформації

1. Алгоритм шифрування даних на основі парольної інформації (PBES-функція)

1. Алгоритм шифрування даних на основі парольної інформації (PBES-функція) призначений для шифрування даних на основі парольної інформації.

2. Ці Вимоги визначають функцію шифрування даних на основі функції PBES2 відповідно до RFC 2898 із використанням алгоритму ДСТУ ГОСТ 28147:2009 у режимі гамування зі зворотним зв'язком.

3. Під час здійснення процесу зашифрування даних на основі парольної інформації вчиняються такі дії:

встановлюється довгостроковий ключовий елемент «dke», що відповідає вимогам Інструкції про порядок постачання і використання ключів до засобів криптографічного захисту інформації, затвердженої наказом Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 12 червня 2007 року № 114, зареєстрованої в Міністерстві юстиції України 25 червня 2007 року за № 729/13996 (далі – Інструкція);

генерується випадкове значення «S» розміром від 8 до 32 байтів. Рекомендований розмір значення «S» – 32 байти;

встановлюється число ітерацій «с» в залежності від умов застосування. Мінімально допустиме значення параметра – 1000 ітерацій, рекомендоване – 10000 ітерацій;

встановлюється значення параметру «dkLen» = 32;

обчислюється значення ключа «DK» = PBKDF (P, S, с, 32);

генеруються випадкові 8 байтів як вектор ініціалізації IV (синхропосилка);

шифруються дані за алгоритмом ДСТУ ГОСТ 28147:2009 у режимі гамування зі зворотним зв'язком, використовуючи «dke», ключ «DK» та вектор ініціалізації «IV», отримані.

Параметри «S», «с», «dke» та «IV» повинні бути збережені разом із зашифрованими даними для їх розшифрування.

4. Під час здійснення процесу розшифрування даних на основі парольної інформації вчиняються такі дії:

встановлюється довгостроковий ключовий елемент «dke», що відповідає вимогам Інструкції;

отримується випадкове значення «S» та число ітерацій с;

встановлюється значення параметру «dkLen» = 32;

обчислюється значення ключа «DK» = PBKDF (P, S, с, 32);

отримується вектор ініціалізації «IV»;

розшифровуються дані за алгоритмом ДСТУ ГОСТ 28147:2009 у режимі гамування зі зворотним зв'язком, використовуючи «dke», ключ «DK» та вектор ініціалізації «IV».

2. Алгоритм контролю цілісності даних на основі парольної інформації (PBMAC-функція)

1. Алгоритм контролю цілісності даних на основі парольної інформації (PBMAC-функція) призначений для контролю цілісності даних на основі парольної інформації.

2. Ці Вимоги визначають функцію контролю цілісності даних на основі функції PBMAC1 відповідно до RFC 2898 із використанням алгоритму ДСТУ ГОСТ 28147-2009 у режимі вироблення імітовставки.

3. Під час здійснення процесу вироблення імітовставки на основі парольної інформації вчиняються такі дії:

встановлюється довгостроковий ключовий елемент «dke», що відповідає вимогам Інструкції;

генерується випадкове значення «S» розміром від 8 до 32 байтів. Рекомендований розмір значення «S» = 32 байти;

встановлюється число ітерацій «с» в залежності від умов застосування. Мінімально допустиме значення параметра – 1000 ітерацій, рекомендоване – 10000 ітерацій;

встановлюється значення параметру «dkLen» = 32;

обчислюється значення ключа «DK» = PBKDF (P, S, с, 32);

обчислюється імітовставка згідно з розділом 5 ДСТУ ГОСТ 28147:2009, використовуючи «dke» та ключ «DK».

Параметри «S», «с» та «dke» повинні бути збережені разом із обчисленою імітовставкою для її перевірки.

4. Під час здійснення процесу перевірки імітовставки на основі парольної інформації вчиняються такі дії:

встановлюється довгостроковий ключовий елемент «dke», що відповідає вимогам Інструкції;

отримується випадкове значення «S» та число ітерацій «с»;

встановлюється значення параметру «dkLen» = 32;

обчислюється значення ключа «DK» = PBKDF (P, S, с, 32);

обчислюється імітовставка згідно з ДСТУ ГОСТ 28147-2009, використовуючи «dke» та ключ «DK»;

порівнюється значення імітовставки, отриманої за результатами виконання цих обчислень із значенням імітовставки, яка перевіряється.

У разі нерівності сум значень імітовставок, подальше оброблення даних припиняється у зв'язку з їх пошкодженням.

IV. Параметри алгоритмів захисту даних на основі парольної інформації

1. Параметри алгоритму формування ключа шифрування ключа на основі парольної інформації

1. Об'єктний ідентифікатор функції PBKDF2 позначається як:
id-PBKDF2 OBJECT IDENTIFIER ::= { iso(1) member-body(2) us(840)
rsadsi(113549) pkcs(1) pkcs-5(5) 12 }.

2. Параметри алгоритму PBKDF2 визначаються як:

```
PBKDF2-params ::= SEQUENCE {  
    salt CHOICE {  
        specified OCTET STRING,  
        otherSource AlgorithmIdentifier {{PBKDF2-  
SaltSources}}  
    },  
    iterationCount INTEGER (1..MAX),  
    keyLength INTEGER (1..MAX) OPTIONAL,  
    prf AlgorithmIdentifier {{PBKDF2-PRFs}}},
```

де salt – випадкове значення розміром від 8 до 32 байт, що подається у вигляді OCTET STRING;

iterationCount – кількість ітерацій, яка визначається умовами застосування;

keyLength – розмір ключа у байтах. Поле "keyLength" у випадку застосування PBES-функції повинно бути відсутнім;

prf – ідентифікатор алгоритму HMAC_GOST34311:

id-hmacGost34311 OBJECT IDENTIFIER ::= { iso(1) member-body(2)
Ukraine(804) root(2) security(1) cryptography(1) ua-pki (1) alg (1) sym (1) 2 };

Параметри алгоритму повинні бути NULL (ASN.1 NULL).

2. Параметри алгоритму шифрування даних на основі парольної інформації

1. Об'єктний ідентифікатор функції PBES2 позначається як:
id-PBES2 OBJECT IDENTIFIER ::= { iso(1) member-body(2) us(840)

rsadsi(113549) pkcs(1) pkcs-5(5) 13 }.

2. Параметри алгоритму PBES2 визначаються як:

```
PBES2-params ::= SEQUENCE {  
    keyDerivationFunc AlgorithmIdentifier {{PBES2-KDFs}},  
    encryptionScheme AlgorithmIdentifier {{PBES2-Encs}}},
```

де keyDerivationFunc – ідентифікатор та параметри PBKDF-функції відповідно до пункту 1 цієї глави.

encryptionScheme – алгоритм ДСТУ ГОСТ 28147:2009 у режим гамування зі зворотним зв'язком.

id-gost28147-cfb OBJECT IDENTIFIER ::= { iso(1) member-body(2) Ukraine(804) root(2) security(1) cryptography(1) ua-pki (1) alg (1) sym (1) gost28147(1) cfb(3) }.

```
GOST28147Parameters ::= SEQUENCE {  
  iv  OCTET STRING (SIZE (8)),  
  dke  OCTET STRING (SIZE (64)) },
```

де iv – вектор ініціалізації, що обирається випадково;

dke – довгостроковий ключовий елемент для ДСТУ ГОСТ 28147:2009, що відповідає вимогам Інструкції.

**Начальник Управління
функціонування центрального
засвідчувального органу
Міністерства юстиції України**

Д.В. Журавльов

**Директор Департаменту криптографічного
захисту інформації Адміністрації
Державної служби спеціального зв'язку
та захисту інформації України**

А.І. Пушкарьов

Додаток
до Вимог до алгоритмів формування
ключів шифрування ключів та
захисту особистих ключів елек-
тронного цифрового підпису та
особистих ключів шифрування
(пункт 5 розділу II)

Формування ключа шифрування ключа на основі паролльної інформації

Приклад 1.

PBKDF = PBKDF2

PRF = HMAC_GOST34311

sBox = SBOX-1

P = "password"

S = "salt"

C = 1

dkLen = 32

DK = 39 46 85 33 E7 8B 12 34 32 0F 2B F9 76 C4 E1 4B 10 B0 2C 70 86 10 07
79 50 4C 1C 07 2F B5 D7 3E

Приклад 2.

PBKDF = PBKDF2

PRF = HMAC_GOST34311

sBox = SBOX-1

P = "password"

S = "salt"

c = 2

dkLen = 32

DK = 13 65 54 93 83 AF FF 5B 1D F2 BF C8 F5 02 70 C6 86 57 5E 4E A6 C3
F3 D1 9C C7 7C 69 49 BB BD B3

Приклад 3.

PBKDF = PBKDF2

PRF = HMAC_GOST34311

sBox = SBOX-1

P = "password"

S = "salt"

c = 4096

dkLen = 32

DK = C7 9F 38 77 CF A7 26 4A 68 F3 E8 AA 6C 1E AF C7 98 52 51 36 8B DB
54 13 67 2F BE 0A AF 99 32 72

Приклад 4.

PBKDF = PBKDF2

PRF = HMAC_GOST34311

sBox = SBOX-1

P = "password"

S = "salt"

c = 10000

dkLen = 32

DK = 50 EA 98 2B 64 1A A7 43 DF 1B AF 65 1F C1 7A A3 D6 D0 77 F7 AD
52 E4 33 F1 7F B7 FD 0C 86 3E 45

Приклад 5.

PBKDF = PBKDF2

PRF = HMAC_GOST34311

sBox = SBOX-1

P = "passwordPASSWORDpassword"

S = "saltSALTsaltSALTsaltSALTsaltSALTsalt"

c = 4096

dkLen = 32

DK = B0 62 4C FD BE A4 89 0E 16 3E CC 24 98 81 65 42 4C B3 8F 9C F2 F3
E6 B9 B7 1E D3 47 34 8E 29 8A

Приклад 6.

PBKDF = PBKDF2

PRF = HMAC_GOST34311

sBox = SBOX-1

P = "pass\0word"

S = "sa\0lt"

c = 4096

dkLen = 32

DK = 8B 3E 73 F8 88 1C 02 9D 93 6B 68 1B 85 C2 76 3B 2F BF 30 58 56 B1
B9 7C 6D 6D 78 C9 BF A7 70 34

ЗАТВЕРДЖЕНО

Наказ Міністерства юстиції України,
Адміністрації Державної служби
спеціального зв'язку та захисту
інформації України

_____ № _____/_____

ВИМОГИ **до інтерфейсів засобів криптографічного захисту інформації**

I. Загальні положення

1. Вимоги до інтерфейсів засобів криптографічного захисту інформації визначають вимоги до інтерфейсів засобів криптографічного захисту інформації, що реалізують алгоритми ДСТУ ГОСТ 28147:2009 «Системы обработки информации. Защита криптографическая. Алгоритмы криптографического преобразования», затвердженого наказом Державного комітету України з питань технічного регулювання та споживчої політики від 22 грудня 2008 року № 495 (далі – ДСТУ ГОСТ 28147:2009), ГОСТ 34.311-95 «Информационная технология. Криптографическая защита информации. Функция хэширования», затвердженого наказом Державного комітету України з питань технічного регулювання та споживчої політики від 21 жовтня 1997 року № 640 (далі – ГОСТ 34.311-95), ДСТУ 4145-2002 «Інформаційна технологія. Криптографічний захист інформації. Електронний цифровий підпис, що ґрунтується на еліптичних кривих», затвердженого наказом Державного комітету України з питань технічного регулювання та споживчої політики від 28 грудня 2002 року № 31 (далі – ДСТУ 4145-2002).

2. Формати даних представлено у нотації ASN.1, визначеній у міжнародному стандарті ISO/IEC 8824 «Information technology – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1)» / ДСТУ

ISO/IEC 8824-3:2008 “Інформаційні технології. Нотація абстрактного синтаксису 1 (ASN.1)” – Частина 3. Специфікація обмежень (ISO/IEC 8824-3:2002, IDT), затвердженому наказом Державного комітету України з питань технічного регулювання та споживчої політики від 26 грудня 2008 року № 508 (із змінами).

3. Усі структури даних кодують за правилами DER згідно з міжнародним стандартом ISO/IEC 8825-1:2002 «Information technology – ASN.1 encoding Rules – Part 1: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)» & AMD1:2004 «Support for EXTENDED-XER».

4. Ці Вимоги засновані на національних стандартах ДСТУ ГОСТ 28147:2009, ГОСТ 34.311-95, ДСТУ 4145-2002 та стандарті «Public Key Cryptography Standard #11 v2.30: Cryptographic Token Interface Standard» (далі - PKCS#11).

5. Ці Вимоги не дублюють стандарти ДСТУ ГОСТ 28147:2009, ГОСТ 34.311-95, ДСТУ 4145-2002, PKCS#11, а описують положення цих стандартів та інтерфейси засобів криптографічного захисту інформації. У разі виникнення розбіжностей між положеннями зазначених стандартів та положеннями цих Вимог, застосовуються положення цих Вимог.

6. Положення цих Вимог є обов’язковими для засобів криптографічного захисту конфіденційної інформації, державних інформаційних ресурсів або інформації, вимога щодо захисту якої встановлена законом (далі – засоби шифрування), а також надійних засобів електронного цифрового підпису. Правильність реалізації вимог до інтерфейсів засобів криптографічного захисту інформації у надійних засобах електронного цифрового підпису та у засобах шифрування підтверджується сертифікатом відповідності або позитивним експертним висновком за результатами державної експертизи у сфері криптографічного захисту інформації, виданими спеціально уповноваженим центральним органом виконавчої влади з питань організації спеціального зв’язку та захисту інформації.

II. Використання національних криптоалгоритмів

1. Шифрування/розшифрування за ДСТУ ГОСТ 28147:2009

1. Ключі згенеровані для шифрування/розшифрування за ДСТУ ГОСТ 28147:2009 (далі – ключі за ДСТУ ГОСТ 28147:2009) – об’єкти типу «СКО_SECRET_KEY», їх атрибут «СКА_KEY_TYPE» повинен мати значення «СКК_GOST28147»:

`СКК_GOST28147 = 0x80420111;`

Окрім атрибутів типу «СКО_SECRET_KEY», ключі за ДСТУ ГОСТ 28147:2009 повинні мати атрибути, що наведені у таблиці 1 цих Вимог.

Атрибути ключів за ДСТУ ГОСТ 28147:2009

Атрибут	Тип даних	Значення
СКА_VALUE	Byte array	Значення ключа (32 байта), LSB-порядок байт
СКА_GOST_SBOXES	Byte array	DER-кодоване представлення довгострокового ключового елемента (далі – ДКЕ)

Атрибут «СКА_GOST_SBOXES» визначається константою:

СКА_GOST_SBOXES = 0x80420311;

Атрибут «СКА_GOST_SBOXES» містить довгостроковий ключовий елемент (далі – ДКЕ) для алгоритму ДСТУ ГОСТ 28147:2009, який використовується разом із заданим ключем. Значення ДКЕ кодується як «OCTET STRING» або як «OBJECT IDENTIFIER».

Якщо значення закодовано як «OCTET STRING», атрибут містить ДКЕ аналогічно представленню поля «dke» структури «DSTU4145Params», як це визначено у підпункті 3.11.1 пункту 3.11 розділу III Вимог до формату посиленого сертифіката відкритого ключа, затверджених наказом Міністерства юстиції України, Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 20 серпня 2012 року № 1236/5/453, зареєстрованим у Міністерстві юстиції України 20 серпня 2012 року за №1398/21710 (далі – Вимоги до формату посиленого сертифіката відкритого ключа).

Якщо значення закодовано як «OBJECT IDENTIFIER», атрибут містить ідентифікатор об'єкту криптографічних параметрів типу «СКО_DOMAIN_PARAMETERS» (тип ключа «СКК_GOST28147»), який визначає ДКЕ для використання із заданим ключем. Криптографічні параметри алгоритму шифрування ДСТУ ГОСТ 28147:2009 наведено в пункті 2 цієї глави.

Атрибут «СКА_GOST_SBOXES» не може змінюватися користувачем.

2. Криптографічні параметри алгоритму шифрування за ДСТУ ГОСТ 28147:2009 зберігаються у об'єктах типу «СКО_DOMAIN_PARAMETERS» (тип ключа «СКК_GOST28147»). Криптографічні параметри, окрім атрибуту типу «СКО_DOMAIN_PARAMETERS», можуть мати атрибути, приведені у таблиці 2 цих Вимог.

Атрибути криптографічних параметрів за ДСТУ ГОСТ 28147:2009

Атрибут	Тип даних	Значення
СКА_GOST_SBOXES	Byte array	DER-кодоване представлення ДКЕ
СКА_ID	Byte array	Ідентифікатор об'єкту криптографічних параметрів

Для об'єктів типу «СКО_DOMAIN_PARAMETERS» атрибут «СКА_GOST_SBOXES» у якості значення приймає DER-кодоване представлення ДКЕ відповідно до поля «dke» структури «DSTU4145Params», яка визначена у підпункті 3.11.1 пункту 3.11 розділу III Вимог до формату посиленого сертифіката відкритого ключа.

У випадку використання ДКЕ, що не підтримується токеном, під час відкриття сесії створюються об'єкти з параметрами, які ним підтримуються. Для знаходження цих об'єктів забезпечується можливість їх пошуку за шаблоном:

```
СКА_CLASS      = СКО_DOMAIN_PARAMETERS
СКА_KEY_TYPE   = СКК_GOST28147
СКА_TOKEN      = TRUE
```

Якщо значення ДКЕ не підлягає розголошенню, об'єкт типу «СКО_DOMAIN_PARAMETERS» у відповіді повертається тільки код помилки «СКР_ATTRIBUTE_SENSITIVE».

3. Алгоритм шифрування за ДСТУ ГОСТ 28147:2009 визначає перший (ECB), другий (CNT) та третій (CFB) режими роботи шифратора, а також режим вироблення імітовставки (MAC) та режим шифрування ключів (WRAP), що використовується при формуванні криптографічних повідомлень, (далі – механізми шифрування), які задаються наступними ідентифікаторами:

```
СКМ_GOST28147_ECB  = 0x80420011;
СКМ_GOST28147_CNT  = 0x80420012;
СКМ_GOST28147_CFB  = 0x80420013;
СКМ_GOST28147_MAC   = 0x80420014;
СКМ_GOST28147_WRAP  = 0x80420016;
```

Інформація про механізми шифрування, що повертається у відповіді структурою «СК_MECHANISM_INFO», повинна бути наступною:

для механізмів шифрування «СКМ_GOST28147_ECB», «СКМ_GOST28147_CNT» та «СКМ_GOST28147_CFB»

```
ulMinKeySize  = 32;
ulMaxKeySize  = 32;
Flags         = SKF_ENCRYPT | SKF_DECRYPT;
```

для механізму шифрування «СКМ_GOST28147_MAC»:

```
ulMinKeySize    = 32;  
ulMaxKeySize    = 32;  
Flags           = CKF_SIGN | CKF_VERIFY;
```

для механізму шифрування «СКМ_GOST28147_WRAP»:

```
ulMinKeySize    = 32;  
ulMaxKeySize    = 32;  
Flags           = CKF_WRAP | CKF_UNWRAP;
```

При використанні можливостей апаратної підтримки у полі «Flags» повинна бути встановлена додатково відмітка «CKF_HW».

Механізм шифрування «СКМ_GOST28147_ECB» не має параметрів.

Параметри механізмів шифрування «СКМ_GOST28147_CNT» та «СКМ_GOST28147_CFB» задаються такою структурою:

```
typedef struct CK_GOST28147_PARAMS {  
    CK_BYTE synchro[8];  
} CK_GOST28147_PARAMS;
```

```
typedef CK_GOST28147_PARAMS* CK_GOST28147_PARAMS_PTR;
```

Поле «synchro» містить значення синхропосилки (LSB-порядок байтів) для алгоритму ДСТУ ГОСТ 28147:2009.

Механізм шифрування «СКМ_GOST28147_MAC» не має параметрів і обчислює імітовставку довжиною у 4 байти.

Для реалізацій алгоритму ДСТУ ГОСТ 28147:2009 передбачено нульове значення синхропосилки, як параметр за умовчанням. Параметри за умовчанням використовуються при відсутності явно визначених параметрів алгоритму.

4. Пристрої, що підтримують шифрування та вироблення імітовставки за ДСТУ ГОСТ 28147:2009, у сформованій відповіді на запит, щодо підтримуваних пристроєм механізмів шифрування (функція «C_GetMechanismList()») повинні вказати список механізмів шифрування, які підтримуються, із зазначенням їх ідентифікаторів.

При використанні можливостей апаратної підтримки у структурі «CK_MECHANISM_INFO» в полі «Flags» повинна бути встановлена додатково відмітка «CKF_HW».

При шифруванні (розшифруванні) використовується функція «C_EncryptInit()», якій у якості механізму шифрування, що використовується, вказується один із механізмів шифрування «СКМ_GOST28147_ECB», «СКМ_GOST28147_CNT» чи «СКМ_GOST28147_CFB», а у якості параметрів – покажчик на структуру

«СК_GOST28147_PARAMS». У разі, коли необхідно використовувати параметри за умовчанням, застосовується «NULL». Ключ, який використовується функцією ініціалізації, повинен мати тип «СКК_GOST28147».

Для обчислення (перевірки) імітовставки використовується функція «C_SignInit()», якій у якості механізму шифрування, що використовується, вказується механізм «СКМ_GOST28147_MAC», а у якості параметрів – покажчик на структуру «СК_GOST28147_PARAMS». У разі, коли необхідно використовувати параметри за умовчанням, застосовується «NULL».

Подальше обчислення імітовставки здійснюється за алгоритмом, визначеним ДСТУ ГОСТ 28147:2009, з використанням функцій «C_Sign()» та «C_SignUpdate()» / «C_SignFinal()», а перевірка імітовставки – з використанням функцій «C_Verify()» та «C_VerifyUpdate()» / «C_VerifyFinal()».

В обох випадках у якості ключа у функцію ініціалізації, необхідно передавати ключ типу «СКК_GOST28147».

2. Гешування за ГОСТ 34.311-95

1. Алгоритм гешування за ГОСТ 34.311-95, (далі – механізм гешування), задається таким ідентифікатором:

```
СКМ_GOST34311 = 0x80420021;
```

Інформація про механізм гешування, що повертається у відповіді структурою «СК_MECHANISM_INFO», повинна бути наступною:

```
ulMinKeySize    = 0;  
ulMaxKeySize    = 0;  
Flags           = CKF_DIGEST;
```

Параметри алгоритму задаються DER-кодованою структурою «Gost34311Params».

```
Gost34311Params ::= CHOICE {  
    dke Dke,  
    params SEQUENCE {  
        dke Dke,  
        iv OCTET STRING (SIZE(8)) OPTIONAL } };
```

Поле «dke» визначає ДКЕ, який використовується при гешуванні.

```
Dke ::= CHOICE {  
    dkeValue OCTET STRING (SIZE(64)),  
    dkeId OBJECT IDENTIFIER }
```

Поле «dkeValue» містить ДКЕ для функції гешування в упакованому форматі, описаному в підпункті 3.12.1 пункту 3.12 розділу III Вимог до формату посиленого сертифіката відкритого ключа.

Поле «dkeId» містить об'єктний ідентифікатор, що ідентифікує ДКЕ для функції гешування.

Поле «iv» визначає стартовий вектор гешування (LSB-порядок байт).

Для реалізації алгоритму гешування за ГОСТ 34.311-95 передбачені параметри за умовчанням:

ДКЕ № 1 Додатку 1 до пункту 2.2 Інструкції про порядок постачання і використання ключів до засобів криптографічного захисту інформації, затвердженої наказом Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 12 червня 2007 року № 114, зареєстрованої у Міністерстві юстиції України 25 червня 2007 року за № 729/13996 (далі – Інструкція);

нульовий стартовий вектор гешування.

Параметри за умовчанням застосовуються за відсутності явно вказаних параметрів алгоритму гешування за ГОСТ 34.311-95.

2. Пристрої, що підтримують гешування за ГОСТ 34.311-95, при формуванні відповіді повинні включати до неї список механізмів (функція «C_GetMechanismList()») із зазначенням ідентифікатору цього алгоритму.

При використанні можливостей апаратної підтримки у структурі «СК_МЕCHANISM_INFO» в полі «Flags» встановлена додатково відмітка «СКF_HW».

Для ініціалізації обчислення геш-функції за ГОСТ 34.311-95 необхідно застосовувати функцію «C_DigestInit()», вказавши її у якості механізму гешування, що використовується, механізм хешування «СКМ_GOST34311», а у якості параметрів – покажчик на структуру «СК_GOST34311_PARAMS» або «NULL», якщо необхідно використовувати параметри за умовчанням.

3. Формування та перевірки підпису за ДСТУ 4145-2002

1. Ключами, що генеруються за алгоритмом ДСТУ 4145-2002 є ключова пара (особистий та відповідний йому відкритий ключ) об'єктів, особистий ключ має тип «СКО_PRIVATE_KEY», відповідний йому відкритий ключ має тип «СКО_PUBLIC_KEY». Атрибут «СКА_KEY_TYPE» особистого та відповідного йому відкритого ключа повинен мати значення «СКК_DSTU4145»:

СКК_DSTU4145 = 0x80420131;

Особистий ключ, окрім атрибутів типу «СКО_PRIVATE_KEY», може мати атрибути, приведені у таблиці 3 цих Вимог.

Таблиця 3

**Атрибути особистого ключа, ключової пари, що генеруються за алгоритмом
ДСТУ 4145-2002**

Атрибут	Тип даних	Значення
СКА_EC_PARAMS	Byte array	DER-кодоване значення криптографічних параметрів – структури «DSTU4145Params» (відповідно до пункту 3.11 розділу III Вимог до формату посиленого сертифіката відкритого ключа, елементи базового поля зберігаються в LSB-форматі)
СКА_VALUE	Big integer	Власне значення особистого ключа (довге ціле відповідно до ДСТУ 4145-2002)

Відкритий ключ, окрім атрибутів типу «СКО_PUBLIC_KEY», може мати атрибути, наведені у таблиці 4 цих Вимог.

Таблиця 4

**Атрибути відкритого ключа, ключової пари, що генеруються за алгоритмом
ДСТУ 4145-2002**

Атрибут	Тип даних	Значення
СКА_EC_PARAMS	Byte array	DER-кодоване значення криптографічних параметрів – структури DSTU4145Params (відповідно до пункту 3.11 розділу III Вимог до формату посиленого сертифіката відкритого ключа, елементи базового поля зберігаються в LSB-форматі)
СКА_EC_POINT	Byte array	DER-кодоване представлення відкритого ключа – OCTET STRING, яке містить стиснене представлення точки еліптичної кривої в LSB-форматі (тип PublicKey відповідно до пункту 3.11 розділу III Вимог до формату посиленого сертифіката відкритого ключа)

2. Криптографічні параметри ключової пари, що генеруються за алгоритмом ДСТУ 4145-2002 зберігаються у об'єктах типу «СКО_DOMAIN_PARAMETERS» (тип ключа – «СКК_DSTU4145»). Криптографічні параметри, окрім атрибутів типу «СКО_DOMAIN_PARAMETERS», можуть мати атрибути, приведені у таблиці 5 цих Вимог.

Атрибути криптографічних параметрів ключової пари, що генеруються за алгоритмом ДСТУ 4145-2002

Атрибут	Тип даних	Значення
СКА_EC_PARAMS	Byte array	DER-кодоване значення криптографічних параметрів – структури DSTU4145Params (відповідно до пункту 3.11 розділу III Вимог до формату посиленого сертифіката відкритого ключа, елементи базового поля зберігаються в LSB-форматі)

У випадку використання криптографічних параметрів ключової пари, що генеруються за алгоритмом ДСТУ 4145-2002, що не підтримується токеном, під час відкриття сесії створюються об'єкти типу «СКО_DOMAIN_PARAMETERS» з параметрами, які ним підтримуються. У цьому випадку створені об'єкти типу «СКО_DOMAIN_PARAMETERS» використовують наступні параметри:

```

СКА_CLASS      = СКО_DOMAIN_PARAMETERS
СКА_KEY_TYPE   = СКК_DSTU4145
СКА_TOKEN      = TRUE

```

3. Алгоритм генерування ключової пари за ДСТУ 4145-2002 (далі – механізм генерування) задається таким ідентифікатором:

```
СКМ_DSTU4145   = 0x80420031;
```

Інформація про механізм генерування («СК_MECHANISM_INFO») має бути наступною:

```

ulMinKeySize   = 163;
ulMaxKeySize   = 509;
Flags          = CKF_SIGN | CKF_VERIFY | CKF_EC_F_2M |
                 CKF_EC_ECPARAMETERS | CKF_EC_NAMEDCURVE |
                 CKF_EC_COMPRESS;

```

Алгоритм генерування ключової пари за ДСТУ 4145-2002 з використанням алгоритму гешування за ГОСТ 34.311-95 як механізм генерування задається таким ідентифікатором:

```
СКМ_GOST34311_DSTU4145   = 0x80420032;
```

Інформація про механізм генерування («СК_MECHANISM_INFO») має бути наступною:

```
ulMinKeySize   = 163;
```

```
ulMaxKeySize    = 509;  
Flags            = CKF_SIGN | CKF_VERIFY | CKF_EC_F_2M |  
                  CKF_EC_ECPARAMETERS | CKF_EC_NAMEDCURVE |  
                  CKF_EC_COMPRESS;
```

Механізм генерування використовує тільки точки еліптичної кривої у стислому зображенні, відповідно до пункту 3.11 розділу III Вимог до формату посиленого сертифіката відкритого ключа.

Даний механізм генерування не має параметрів, всі необхідні параметри мають зберігатися у атрибуті «СКА_EC_PARAMS» особистого та відповідного йому відкритого ключів.

4. Застосування механізму генерування.

Пристрої, що підтримують формування та перевірку підпису за ДСТУ 4145-2002, при формуванні відповіді повинні включати до неї список механізмів (функція «C_GetMechanismList()») із зазначенням ідентифікаторів наявних алгоритмів.

При використанні можливостей апаратної підтримки у структурі «CK_MECHANISM_INFO» в полі «Flags» встановлена додатково відмітка «CKF_HW».

Для формування (перевірки) підпису необхідно застосовувати функцію «C_SignInit()», вказавши її у якості механізму генерування, що використовується, механізм генерування «СКМ_GOST34311_DSTU4145», а у полі параметрів зазначити «NULL».

Подальше формування підпису здійснюється з використанням функцій «C_Sign()» та «C_SignUpdate()» / «C_SignFinal()», а перевірка підпису – з використанням функцій «C_Verify()» та «C_VerifyUpdate()» / «C_VerifyFinal()».

Для формування (перевірки) підпису до обчисленого значення функції гешування необхідно застосувати функцію «C_SignInit()», вказавши її у якості механізму генерування, що використовується, механізм «СКМ_DSTU4145», а у полі параметрів зазначити «NULL».

Подальше формування підпису здійснюється за алгоритмом ДСТУ 4145-2002. У якості даних передається обчислене значення функції гешування (LSB-порядок байт).

Під час формування підпису необхідно вказувати особистий ключ, а під час перевірки – відповідний йому відкритий ключ. Ключі повинні мати тип «СКК_DSTU4145».

III. Механізми генерації ключів

1. Механізми генерації ключів ДСТУ ГОСТ 28147:2009

1. Механізм «СКМ_GOST28147_KEY_GEN» здійснює генерацію ключа шифрування згідно ДСТУ ГОСТ 28147:2009 за допомогою генератора (псевдо)випадкових послідовностей.

Механізм визначається таким ідентифікатором:

СКМ_GOST28147_KEY_GEN = 0x80420041;

Інформація про механізм («СК_MECHANISM_INFO») має бути наступною:

ulMinKeySize = 32;
ulMaxKeySize = 32;
Flags = CKF_GENERATE;

При використанні можливостей апаратної підтримки у полі «Flags» повинна бути проставлена відмітка «СКF_HW».

Даний механізм не має параметрів.

Ключі шифрування створюються функцією «C_GenerateKey()» за допомогою даного механізму. Окрім стандартних атрибутів класу, у шаблоні ключа, що створюється, можуть бути присутні атрибути, що наведені у таблиці 6 цих Вимог. Значення за умовчанням для деяких атрибутів наведені в таблиці 7 цих Вимог.

Таблиця 6

Атрибути шаблону ключів шифрування, які створюються механізмом
«СКМ_GOST28147_KEY_GEN»

Атрибут	Значення
СКА_CLASS	якщо присутній, повинен бути «СКО_SECRET_KEY»
СКА_KEY_TYPE	якщо присутній, повинен бути «СКК_GOST28147»
СКА_GOST_SBOXES	ДКЕ або ідентифікатор об'єкту з ДКЕ, за замовчанням ДКЕ №1 згідно з Інструкції

Таблиця 7

Значення атрибутів, які уточнюють сферу використання ключа,
що приймаються за умовчанням

Атрибут	Значення
СКА_ENCRYPT	«TRUE» або «FALSE» (за замовчанням – «TRUE»)
СКА_DECRYPT	«TRUE» або «FALSE» (за замовчанням – «TRUE»)
СКА_SIGN	«TRUE» або «FALSE» (за замовчанням – «TRUE»)

CKA_VERIFY	«TRUE» або «FALSE» (за замовчанням – «TRUE»)
CKA_WRAP	«TRUE» або «FALSE» (за замовчанням – «TRUE»)
CKA_UNWRAP	«TRUE» або «FALSE» (за замовчанням – «TRUE»)

2. Механізм «CKM_DSTUDH_DERIVE» призначено для отримання ключа шифрування ключа (далі – КШК) на основі протоколу Діффі-Геллмана для еліптичних кривих, як це визначено в розділі V Вимог до форматів криптографічних повідомлень, затверджених наказом Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 18 грудня 2012 року № 739, зареєстрованих у Міністерстві юстиції України 14 січня 2013 року за № 108/22640 (далі – Вимог до форматів криптографічних повідомлень) («id-dhSinglePass-stdDH-gost34311kdf-scheme»).

Механізм визначається таким ідентифікатором:

CKM_DSTUDH_DERIVE = 0x80420043;

Інформація про механізм («CK_MECHANISM_INFO») має бути наступною:

```
ulMinKeySize    = 163;
ulMaxKeySize    = 509;
Flags           = CKF_DERIVE | CKF_EC_F_2M | CKF_EC_COMPRESS;
```

При використанні можливостей апаратної підтримки у полі «Flags» повинна бути проставлена відмітка «CKF_HW».

Параметри алгоритму задаються стандартною структурою «CK_ECDH1_DERIVE_PARAMS»:

```
typedef struct CK_ECDH1_DERIVE_PARAMS {
    CK_EC_KDF_TYPE    kdf;
    CK_ULONG          ulSharedDataLen;
    CK_BYTE_PTR       pSharedData;
    CK_ULONG          ulPublicDataLen;
    CK_BYTE_PTR       pPublicData;
} CK_ECDH1_DERIVE_PARAMS;
```

де поля мають такі значення:

«kdf» – ідентифікатор функції формування ключа, повинен мати значення «CKD_GOST34311_KDF»;

«ulSharedDataLen» – довжина одноразових даних, які використовуються під час генерації КШК, згідно розділу V Вимог до форматів криптографічних повідомлень, повинна дорівнювати 0 або 64;

«pSharedData» – одноразові дані, які використовуються при генерації КШК;

«ulPublicDataLen» – довжина в байтах відкритого ключа іншої сторони;

«pPublicData» – відкритий ключ іншої сторони, формат співпадає з форматом атрибута «СКА_EC_POINT».

КШК формується функцією «C_DeriveKey()» за допомогою даного механізму. Параметр функції «hBaseKey» повинен вказувати на особистий ключ.

У шаблоні КШК, який створюється, вказуються атрибути, що зазначені у таблиці 8 цих Вимог.

Таблиця 8

Атрибути, що зазначаються у шаблоні КШК

Атрибут	Значення
СКА_CLASS	«СКО_SECRET_KEY»
СКА_KEY_TYPE	«СКК_GOST28147», атрибут повинен бути присутнім
СКА_WRAP	«TRUE»
СКА_UNWRAP	«TRUE»

Об'єкт «hBaseKey» повинен бути або діючим на даний час особистим ключем користувача, або спеціально згенерованим для даного протоколу на боці відправника «віртуальним» особистим ключем. У будь-якому випадку, значення атрибутів «СКА_EC_PARAMS» об'єктів «pPublicData» та «hBaseKey» повинні співпадати.

Для обчислення геш-значення за ГОСТ 34.311-95, що використовується під час вироблення КШК, застосовується ДКЕ, який взято з атрибуту «СКА_EC_PARAMS» об'єкту «hBaseKey».

Створений ключ алгоритму ДСТУ ГОСТ 28147:2009 отримує значення ДКЕ (атрибут «СКА_GOST_SBOXES») з атрибуту «СКА_EC_PARAMS» об'єкту «hBaseKey».

3. Механізм «СКМ_DSTUDH_COFACTOR_DERIVE»

Даний механізм призначено для отримання КШК на основі протоколу Діффі-Геллмана для еліптичних кривих, як це визначено у розділі V Вимог до форматів криптографічних повідомлень, («id-dhSinglePass-cofactorDH-gost34311kdf-scheme»).

Механізм визначається таким ідентифікатором:

СКМ_DSTUDH_COFACTOR_DERIVE = 0x80420044;

Інформація про механізм («СК_МЕCHANISM_INFO») має бути наступною:

ulMinKeySize = 163;
ulMaxKeySize = 509;
Flags = CKF_DERIVE | CKF_EC_F_2M | CKF_EC_COMPRESS;

При використанні можливостей апаратної підтримки у полі «Flags» повинна бути проставлена відмітка «СКФ_HW».

Параметри алгоритму та порядок його використання визначаються так, як і у механізмі «СКМ_DSTUDH_DERIVE».

4. Механізм «СКМ_GOST_WRAP»

Даний механізм призначений для шифрування та розшифрування особистих ключів з використанням алгоритму, як це визначено у розділі VI Вимог до форматів криптографічних повідомлень. Механізм визначається таким ідентифікатором:

```
СКМ_GOST_WRAP = 0x80420016;
```

Інформація про механізм («СК_MECHANISM_INFO») повинна бути наступною:

```
ulMinKeySize = 32;  
ulMaxKeySize = 32;  
Flags        = CKF_WRAP | CKF_UNWRAP;
```

При використанні можливостей апаратної підтримки у полі «Flags» повинна бути проставлена відмітка «СКФ_HW».

Механізм не має параметрів.

Шифрування ключів здійснюється функцією «C_WrapKey()». КШК, який передається у функцію, має задовольняти таким вимогам:

```
атрибут СКА_CLASS = СКО_SECRET_KEY;  
атрибут СКА_KEY_TYPE = СКК_GOST28147;  
атрибут СКА_WRAP = TRUE.
```

Розшифрування ключів здійснюється функцією «C_UnwrapKey()». КШК, який передається у функцію шифрування, має задовольняти таким вимогам:

```
атрибут СКА_CLASS = СКО_SECRET_KEY;  
атрибут СКА_KEY_TYPE = СКК_GOST28147;  
атрибут СКА_UNWRAP = TRUE;
```

У шаблоні атрибутів КШК, який буде вміщувати розшифрований ключ, повинен бути присутнім атрибут «СКА_KEY_TYPE». Також можуть бути присутніми стандартні атрибути класу та атрибути, що наведені у таблиці 6 цих Вимог.

2. Механізми генерації ключів ДСТУ 4145-2002

1. Механізм «СКМ_DSTU4145_KEY_PAIR_GEN»

Механізм генерації «СКМ_DSTU4145_KEY_GEN» описує генерацію ключової пари відповідно розділу 9 ДСТУ 4145-2002. Механізм визначається таким ідентифікатором:

```
СКМ_DSTU4145_KEY_PAIR_GEN      = 0x80420042;
```

Інформація про механізм («СК_MECHANISM_INFO») має бути такою:

```
ulMinKeySize    = 163;  
ulMaxKeySize    = 509;  
Flags           = CKF_GENERATE_KEY_PAIR | CKF_EC_F_2M |  
                  CKF_EC_ECPARAMETERS |  
                  CKF_EC_NAMEDCURVE |  
                  CKF_EC_COMPRESS;
```

При використанні можливостей апаратної підтримки у полі «Flags» повинна бути проставлена відмітка «CKF_HW».

Даний механізм не має параметрів.

Ключові пари створюються функцією «C_GenerateKeyPair()» за допомогою даного механізму.

У шаблоні атрибутів особистого ключа, який створюється, повинні бути присутні обов'язкові атрибути, які наведені у таблиці 9 цих Вимог.

У шаблоні атрибутів відкритого ключа, який створюється, повинні бути присутні обов'язкові атрибути, які наведені у таблиці 10 цих Вимог.

В обох шаблонах дозволяється вказувати стандартні атрибути, що впливають на місце розміщення ключів та доступ до них: «СКА_TOKEN», «СКА_PRIVATE», «СКА_SENSITIVE», «СКА_EXTRACTABLE».

Атрибути «СКА_SIGN_RECOVER», «СКА_VERIFY_RECOVER», «СКА_WRAP», «СКА_UNWRAP», «СКА_ENCRYPT» та «СКА_DECRYPT» із значенням «TRUE» у шаблонах не вказуються, оскільки ці операції не підтримуються базовим алгоритмом.

Якщо у шаблоні присутні перераховані атрибути із значенням «TRUE», функція повинна у відповіді повернути інформацію про помилку: «СКR_TEMPLATE_INCONSISTENT».

Обов'язково необхідно вказати значення атрибуту «СКА_DERIVE».

Якщо атрибути «СКА_EC_PARAMS» визначені для обох ключів, їх значення повинні співпадати. Достатньо вказати атрибут «СКА_EC_PARAMS» для одного з ключів.

Деякі токени можуть підтримувати конкретний набір криптографічних параметрів. При неможливості використовувати визначені параметри, функція генерування ключів у відповіді поверне помилку «СКR_DOMAIN_PARAMS_INVALID». У таких випадках використовуються тільки ті параметри, що підтримуються. Такі параметри повинні бути відомими заздалегідь або одержуватися з токена через об'єкти типу «СКО_DOMAIN_PARAMETERS», як зазначено у пункті 2 глави 3 розділу II цих Вимог.

Токени можуть підтримувати значення криптографічних параметрів за умовчанням, але при цьому необхідно вказувати значення атрибуту «СКА_EC_PARAMS».

Таблиця 9

Атрибути шаблону особистих ключів, які створюються механізмом
«СКМ_DSTU4145_KEY_PAIR_GEN»

Атрибут	Значення
СКА_CLASS	якщо присутній, повинен бути «СКО_PRIVATE_KEY»
СКА_KEY_TYPE	якщо присутній, повинен бути «СКК_DSTU4145»
СКА_EC_PARAMS	DER-кодоване значення криптографічних параметрів – структури DSTU4145Params (відповідно до пункту 3.11 розділу III Вимог до формату посиленого сертифіката відкритого ключа, елементи базового поля зберігаються в LSB-форматі)
СКА_DERIVE	TRUE або FALSE (за замовчанням – FALSE)

Таблиця 10

Атрибути шаблону відкритих ключів, які створюються механізмом
«СКМ_DSTU4145_KEY_PAIR_GEN»

Атрибут	Значення
СКА_CLASS	якщо присутній, повинен бути СКО_PUBLIC_KEY
СКА_KEY_TYPE	якщо присутній, повинен бути СКК_DSTU4145
СКА_EC_PARAMS	DER-кодоване значення криптографічних параметрів – структури DSTU4145Params (відповідно до пункту 3.11 розділу III Вимог до формату посиленого сертифіката відкритого ключа, елементи базового поля зберігаються в LSB-форматі)

**Начальник Управління функціонування
центрального засвідчувального органу
Міністерства юстиції України**

Д.В. Журавльов

**Директор Департаменту криптографічного
захисту інформації Адміністрації
Державної служби спеціального зв'язку
та захисту інформації України**

А.І. Пушкарьов

ЗАТВЕРДЖЕНО

Наказ Міністерства юстиції України,
Адміністрації Державної служби
спеціального зв'язку та захисту
інформації України

_____ № _____ / _____

ВИМОГИ

**до форматів транспортних контейнерів особистих ключів електронного
цифрового підпису та особистих ключів шифрування**

I. Загальні положення

8.1. Ці Вимоги визначають формати транспортних контейнерів та захищених транспортних контейнерів особистих ключів електронного цифрового підпису та особистих ключів шифрування з використанням алгоритмів ДСТУ 4145-2002 «Інформаційні технології. Криптографічний захист інформації. Цифровий підпис, що ґрунтується на еліптичних кривих. Формування та перевіряння» (далі – ДСТУ 4145-2002), ДСТУ ГОСТ 28147:2009 «Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования», (далі – ДСТУ ГОСТ 28147:2009), ГОСТ 34.311-95 «Информационная технология. Криптографическая защита информации. Функция хеширования» (далі – ГОСТ 34.311-95)

8.2. Формати даних представлено у нотації ASN.1, визначеній у міжнародному стандарті ISO/IEC 8824 «Information technology – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1)» / ДСТУ ISO/IEC 8824-3:2008 «Інформаційні технології. Нотація абстрактного синтаксису 1 (ASN.1)». – Частина 3. Специфікація обмежень (ISO/IEC 8824-3:2002, IDT), затвердженому наказом Державного комітету України з питань технічного регулювання та споживчої політики від 26 грудня 2008 року № 508 (із змінами).

8.3. Усі структури даних кодують за правилами DER згідно з міжнародним стандартом ISO/IEC 8825-1:2002 «Information technology – ASN.1 encoding Rules – Part 1: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)» & AMD1:2004 «Support for EXTENDED-XER».

4. Ці Вимоги засновані на національних стандартах України ДСТУ 4145-2002, ДСТУ ГОСТ 28147:2009 та ГОСТ 34.311-95 «Информационная технология. Криптографическая защита информации. Функция хеширования» (далі – ГОСТ 34.311-95) та міжнародних рекомендаціях RFC 5208 «Private-Key Information Syntax Specification (PKCS#8)», May 2008 (далі – RFC 5208) та RFC 2898 «Password-Based Cryptography Specification (PKCS#5)», September 2000 (далі – RFC 2898).

5. Ці Вимоги не дублюють стандарти ДСТУ 4145-2002, ДСТУ ГОСТ 28147:2009, ГОСТ 34.311-95 та міжнародні рекомендації RFC 5208, RFC 2898, а описують положення цих стандартів і рекомендацій та формати транспортних контейнерів та захищених транспортних контейнерів особистих ключів електронного цифрового підпису та особистих ключів шифрування. У разі виникнення розбіжностей між положеннями зазначених стандартів і рекомендацій та положеннями цих Вимог, застосовуються положення цих Вимог.

6. Положення цих Вимог є обов'язковими для засобів криптографічного захисту конфіденційної інформації, державних інформаційних ресурсів або інформації, вимога щодо захисту якої встановлена законом (далі – засоби шифрування), а також надійних засобів електронного цифрового підпису. Правильність реалізації форматів транспортних контейнерів та захищених транспортних контейнерів особистих ключів електронного цифрового підпису та особистих ключів шифрування у надійних засобах електронного цифрового підпису та у засобах шифрування підтверджується сертифікатом відповідності або позитивним експертним висновком за результатами державної експертизи у сфері криптографічного захисту інформації, виданими спеціально уповноваженим центральним органом виконавчої влади з питань організації спеціального зв'язку та захисту інформації.

II. Порядок формування транспортних контейнерів особистих ключів електронного цифрового підпису та особистих ключів шифрування

1. При здійсненні процедури формування контейнерів особистих ключів електронного цифрового підпису або особистих ключів шифрування виконуються такі дії:

формується формат транспортного контейнеру особистого ключа електронного цифрового підпису або особистого ключа шифрування та отримується DER-кодований рядок байтів;

шифрується DER-кодований рядок байтів та формується захищений контейнер особистого ключа.

2. Захищений транспортний контейнер особистого ключа електронного цифрового підпису або особистого ключа шифрування може бути збережений у

файловій системі. Кожний захищений контейнер повинен бути поданий у вигляді DER-кодованих байтів та міститися у окремому файлі із розширенням «.pk8».

III. Формат транспортного контейнера особистого ключа електронного цифрового підпису та особистого ключа шифрування

1. Формат транспортного контейнера особистого ключа електронного цифрового підпису або особистого ключа шифрування має такий вигляд:

```
PrivateKeyInfo ::= SEQUENCE {  
    version                [0] Version,  
    privateKeyAlgorithm    PrivateKeyAlgorithmIdentifier,  
    privateKey             PrivateKey,  
    attributes              [0] IMPLICIT Attributes OPTIONAL }  
Version ::= INTEGER  
PrivateKeyAlgorithmIdentifier ::= AlgorithmIdentifier  
PrivateKey ::= OCTET STRING  
Attributes ::= SET OF Attribute
```

2. Поле «version» містить версію формату типу «PrivateKeyInfo». Це поле повинно мати значення «0».

3. Ідентифікатор криптоалгоритму у полі «AlgorithmIdentifier» містить об'єктний ідентифікатор та відповідні параметри криптоалгоритму. Об'єктний ідентифікатор повинен вказувати на криптоалгоритм ДСТУ 4145-2002 відповідно до Вимог до формату посиленого сертифіката відкритого ключа, затверджених наказом Міністерства юстиції України, Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 20 серпня 2012 року № 1236/5/453, зареєстрованих у Міністерстві юстиції України 20 серпня 2012 року за № 1398/21710 (далі – Вимоги до формату посиленого сертифіката відкритого ключа). Параметри криптоалгоритму повинні бути представлені структурою «DSTU4145Params» як наведено у Вимогах до формату посиленого сертифіката відкритого ключа.

4. Поле «privateKey» містить особистий ключ алгоритму ДСТУ 4145-2002 у форматі Little-Endian, який кодується як OCTET STRING.

5. Поле “attributes” є необов'язковим та містить набір атрибутів.

6. Транспортний контейнер особистого ключа визначається відповідно до прикладу 1 Прикладів ASN.1 структур транспортних контейнерів особистого ключа електронного цифрового підпису або особистого ключа шифрування, що додаються до цих Вимог.

IV. Формат захищеного транспортного контейнера особистого ключа електронного цифрового підпису та особистого ключа шифрування

1. Формат захищеного транспортного контейнера особистого ключа електронного цифрового підпису або особистого ключа шифрування має такий вигляд:

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

2. Поле «encryptionAlgorithm» містить об'єктний ідентифікатор та відповідні параметри алгоритму шифрування контейнера особистого ключа «PrivateKeyInfo». Об'єктний ідентифікатор повинен вказувати на криптоалгоритм PBES2 відповідно до RFC 2898. Параметри криптоалгоритму повинні бути представлені структурою «PBES2-params» відповідно до Вимог до алгоритмів формування ключів шифрування ключів та захисту особистих ключів електронного цифрового підпису та особистих ключів шифрування (далі – Вимоги до алгоритмів формування ключів шифрування ключів).

3. Поле «encryptedData» містить дані, які зашифровані відповідно до Вимог до алгоритмів формування ключів шифрування ключів.

4. Захищений транспортний контейнер особистого ключа. (Пароль – «password») визначається відповідно до прикладу 2 Прикладів ASN.1 структур транспортних контейнерів особистого ключа електронного цифрового підпису або особистого ключа шифрування, що додаються до цих Вимог.

V. Порядок формування ідентифікатора відкритого ключа

1. Для забезпечення перевірки відповідності особистого ключа електронного цифрового підпису або особистого ключа шифрування відкритому ключу, що міститься в сертифікаті відкритого ключа підписувача або відправника, здійснюється формування ідентифікатора відкритого ключа із виконанням таких дій:

з поля «privateKey» вилучається особистий ключ алгоритму ДСТУ 4145-2002;

з поля «AlgorithmIdentifier» вилучаються загальні параметри, що визначаються алгоритмом ДСТУ 4145-2002;

як значення функції від особистого ключа та загальних параметрів відповідно до ДСТУ 4145-20025 обчислюється відкритий ключ;

отримується стиснене зображення відкритого ключа;

обчислюється значення геш-функції за ГОСТ 34.311-95 від отриманої на попередньому етапі послідовності байтів. Як стартовий вектор геш-функції використовується нульовий вектор.

2. Якщо параметри криптоалгоритму у полі «AlgorithmIdentifier» містять таблицю заповнення вузлів заміни блоку підстановки (довгостроковий ключовий елемент – ДКЕ), то при обчисленні геш-функції використовується саме цей ДКЕ, інакше використовується ДКЕ № 1, що наведений у додатку 1 до Інструкції про порядок постачання і використання ключів до засобів криптографічного захисту інформації, затвердженої наказом Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 12 червня 2007 року № 114, зареєстрованої в Міністерстві юстиції України 25 червня 2007 року за № 729/13996.

**Начальник Управління функціонування
центрального засвідчувального органу
Міністерства юстиції України**

Д.В. Журавльов

**Директор Департаменту криптографічного
захисту інформації Адміністрації
Державної служби спеціального зв'язку
та захисту інформації України**

А.І. Пушкарьов

Додаток
до Вимог до форматів транс-
портних контейнерів особистих
ключів електронного цифро-
вого підпису та особистих ключів
шифрування
(пункт 6 розділу III,
пункт 4 розділу IV цих Вимог)

ПРИКЛАДИ

ASN.1 структур транспортних контейнерів особистого ключа електронного цифрового підпису або особистого ключа шифрування

Приклад 1. Транспортний контейнер особистого ключа.

```
SEQUENCE : privateKeyInfo  
    INTEGER : version 0  
SEQUENCE : privateKeyAlgorithm  
    OBJECT IDENTIFIER : id-dstu4145PB [1.2.804.2.1.1.1.1.3.1.1]  
SEQUENCE : DSTU4145Params  
    SEQUENCE : ecbinary  
        SEQUENCE : f  
            INTEGER : m 257  
            INTEGER : trinomial 12  
INTEGER : a 0  
OCTET STRING : b  
    10BEE3DB6AEA9E1F86578C45C12594FF942394A7D738F  
    9187E6515017294F4CE01  
INTEGER : n  
    00800000000000000000000000000000006759213AF18  
    2E987D3E17714907D470D  
OCTET STRING : bp  
    B60FD2D8DCE8A93423C6101BCA91C47A007E6C300B26C  
    D556C9B0E7D20EF292A00  
OCTET STRING : dke  
    A9D6EB45F13C708280C4967B231F5EADF658EBA4C037291D  
    38D96BF025CA4E17F8E9720DC615B43A28975F0BC1DEA364  
    38B564EA2C179FD0123E6DB8FAC57904  
OCTET STRING : privateKey  
    5FDB9C6030A36861080C8CE90EE448C29BDBF07EE0BC78A6C2ECA2  
    5CEB24012C
```

Приклад 2. Захищений транспортний контейнер особистого ключа. (Пароль – “password”)

```

SEQUENCE : encryptedPrivateKeyInfo
  SEQUENCE : encryptionAlgorithm
    OBJECT IDENTIFIER : id-PBES2 [1.2.840.113549.1.5.13]
    SEQUENCE : PBES2-params
      SEQUENCE : keyDerivationFunc
        OBJECT IDENTIFIER : id-PBKDF2 [1.2.840.113549.1.5.12]
        SEQUENCE : PBKDF2-params
          OCTET STRING : salt
            31A58DC1462981189CF6C701E276C7553A5AB5F6E3
            6D8418E4AA40C930CF3876
          INTEGER : iterationCount
            10000
        SEQUENCE : prf
          OBJECT IDENTIFIER : id-hmacGost34311 [1.2.804.2.1.1.1.1.2]
          NULL : "
      SEQUENCE : encryptionScheme
        OBJECT IDENTIFIER : id-gost28147-cfb [1.2.804.2.1.1.1.1.1.3]
        SEQUENCE : GOST28147Params
          OCTET STRING : iv
            4BB10F5C2945D49E
          OCTET STRING : dke
            A9D6EB45F13C708280C4967B231F5EADF658EBA4C0
            37291D38D96BF025CA4E17F8E9720DC615B43A2897
            5F0BC1DEA36438B564EA2C179FD0123E6DB8FAC579
            04
        OCTET STRING : encryptedData
          29A22E2951E632E1E444AE38F521C890FF6377FC0539113A66720B
          FC4E9107C566A07E3EAB9AE67F337ED9C66C021363E79508A9FDFA
          09E78877DFBE76543160DC83195427A9C7FF2F6F40D8D0FEA26583
          C72EF6E5E2045DA9512A61FBC2B9573E8B0BDC8F034D8CDA3ACA63
          B78C9877FA75C228756BE76083A235247A094C1EF2996FFBFCB45E
          6D14807B38E26A86261035131DEC63B37307B44EF2C0EAFE51392C
          D8A2B8B50FC6F8BC8B1A62EFD276D4E81BB358F4931BAAA3660C0C
          0B5DF52E5233D90D1F4EF5203C40F036CF5912914660BF28212C9B
          3FD9141CB89B93C13522DEB33085A25CC102B5B7DBA377078A645E
          88

```


ЗАТВЕРДЖЕНО

Наказ Міністерства юстиції України,
Адміністрації Державної служби
спеціального зв'язку та захисту
інформації України

_____ № _____/_____

ВИМОГИ

**до форматів контейнерів зберігання особистих ключів електронного
цифрового підпису, особистих ключів шифрування та сертифікатів
відкритих ключів**

I. Загальні положення

1. Ці Вимоги визначають формати контейнерів зберігання особистих ключів електронного цифрового підпису, особистих ключів шифрування, сертифікатів відкритих ключів та іншої інформації, з використанням алгоритмів ДСТУ 4145-2002 «Інформаційні технології. Криптографічний захист інформації. Цифровий підпис, що ґрунтується на еліптичних кривих. Формування та перевіряння» (далі – ДСТУ 4145-2002); ДСТУ ГОСТ 28147:2009 «Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования» (далі – ДСТУ ГОСТ 28147:2009); ГОСТ 34.311-95 «Информационная технология. Криптографическая защита информации. Функция хеширования» (далі – ГОСТ 34.311-95).

2. Формати даних представлено у нотації ASN.1, визначеній у міжнародному стандарті ISO/IEC 8824 «Information technology – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1)» / ДСТУ ISO/IEC 8824-3:2008 «Інформаційні технології. Нотація абстрактного синтаксису 1 (ASN.1)» – частина 3. Специфікація обмежень (ISO/IEC 8824-3:2002, IDT), затвердженому наказом Державного комітету України з питань технічного регулювання та споживчої політики від 26 грудня 2008 року № 508 (із змінами).

3. Усі структури даних кодують за правилами DER згідно з міжнародними стандартами ISO/IEC 8825-1:2002 «Information technology – ASN.1 encoding Rules – Part 1: Specification of Basic Encoding Rules (BER), Canonical Encoding

Rules (CER) and Distinguished Encoding Rules (DER)» та AMD1:2004 «Support for EXTENDED-XER».

4. Ці Вимоги засновані на національних стандартах України ДСТУ 4145-2002, ДСТУ ГОСТ 28147:2009, ГОСТ 34.311-95 з урахуванням Інструкції про порядок постачання і використання ключів до засобів криптографічного захисту інформації, затвердженої наказом Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 12 червня 2007 року № 114, зареєстрованої в Міністерстві юстиції України 25 червня 2007 року за № 729/13996 (далі – Інструкція); Вимог до формату посиленого сертифіката відкритого ключа, затверджених наказом Міністерства юстиції України, Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 20 серпня 2012 року № 1236/5/453, зареєстрованих у Міністерстві юстиції України 20 серпня 2012 року за № 1398/21710 (далі – Вимоги до формату посиленого сертифіката відкритого ключа); Вимог до форматів криптографічних повідомлень, затверджених наказом Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 18 грудня 2012 року № 739, зареєстрованих у Міністерстві юстиції України 14 січня 2013 року за №108/22610 (далі – Вимог до форматів криптографічних повідомлень); Вимог до алгоритмів формування ключів шифрування ключів та захисту особистих ключів електронного цифрового підпису та особистих ключів шифрування (далі – Вимог до алгоритмів формування ключів шифрування ключів) та Вимог до форматів транспортних контейнерів особистих ключів електронного цифрового підпису та особистих ключів шифрування (далі – Вимог до форматів транспортних контейнерів особистих ключів); міжнародних рекомендаціях RFC 5208 «Private-Key Information Syntax Specification (PKCS#8)», May 2008 (далі – RFC 5208), RFC 2898 «Password-Based Cryptography Specification (PKCS#5)», September 2000 (далі – RFC 2898) та стандарту PKCS#12 «Personal Information Exchange Syntax», October 2012 (далі – PKCS#12).

5. Ці Вимоги не дублюють стандарти ДСТУ 4145-2002, ДСТУ ГОСТ 28147:2009, ГОСТ 34.311-95, міжнародні рекомендації RFC 5208, RFC 2898 та стандарт PKCS#12, а описують положення цих стандартів і рекомендацій та формати контейнерів зберігання особистих ключів електронного цифрового підпису, особистих ключів шифрування, сертифікатів відкритих ключів та іншої інформації. У разі виникнення розбіжностей між положеннями зазначених стандартів і рекомендацій та положеннями цих Вимог, застосовуються положення цих Вимог.

6. Положення цих Вимог є обов'язковими для засобів криптографічного захисту конфіденційної інформації, державних інформаційних ресурсів або інформації, вимога щодо захисту якої встановлена законом (далі – засоби шифрування), а також надійних засобів електронного цифрового підпису.

Правильність реалізації форматів контейнерів зберігання особистих ключів електронного цифрового підпису, особистих ключів шифрування, сертифікатів

відкритих ключів та іншої інформації у надійних засобах електронного цифрового підпису та у засобах шифрування підтверджується сертифікатом відповідності або позитивним експертним висновком за результатами державної експертизи у сфері криптографічного захисту інформації, виданими спеціально уповноваженим центральним органом виконавчої влади з питань організації спеціального зв'язку та захисту інформації.

II. Особливості формування контейнерів зберігання особистих ключів електронного цифрового підпису, особистих ключів шифрування та сертифікатів відкритих ключів

1. Контейнер зберігання особистих ключів електронного цифрового підпису, особистих ключів шифрування та сертифікатів відкритих ключів може містити особисті ключі підписувачів або відправників, сертифікати відкритих ключів підписувачів або відправників, акредитованих центрів сертифікації ключів, засвідчувальних центрів і центрального засвідчувального органу (ланцюжок сертифікатів), списки відкликаних сертифікатів (CVC) акредитованих центрів сертифікації ключів, засвідчувальних центрів і центрального засвідчувального органу та повинен забезпечувати конфіденційність та цілісність зазначених даних.

2. Відповідно до PKCS#12 для забезпечення конфіденційності даних контейнера зберігання особистих ключів та сертифікатів повинен використовуватися режим захисту даних на основі паролі інформації (password privacy mode).

3. Відповідно до PKCS#12 для забезпечення цілісності даних контейнера зберігання особистих ключів та сертифікатів повинен використовуватися режим контролю цілісності даних на основі паролі інформації (password integrity mode).

4. Для забезпечення конфіденційності та контролю цілісності даних контейнера зберігання особистих ключів та сертифікатів повинен використовуватися однаковий пароль.

5. Сертифікати відкритих ключів та CVC повинні зберігатися у структурі "AuthenticatedSafe" як дані типу «зашифровані дані».

6. Особисті ключі користувача повинні зберігатися у структурі «AuthenticatedSafe» як дані типу «дані», які містять об'єкт «PKCS8ShroudedKeyBag».

7. Контейнер зберігання особистих ключів електронного цифрового підпису, особистих ключів шифрування та сертифікатів відкритих ключів може бути збережений у файловій системі. Кожний контейнер повинен бути поданий у ви-

гляді DER-кодованих байтів та міститися у окремому файлі із розширенням «.pfx».

III. Формат контейнера зберігання особистих ключів електронного цифрового підпису, особистих ключів шифрування та сертифікатів відкритих ключів

1. Формат контейнера зберігання особистих ключів електронного цифрового підпису, особистих ключів шифрування та сертифікатів відкритих ключів має такий вигляд:

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

2. Поле «version» містить версію формату контейнера зберігання електронного цифрового підпису, особистих ключів шифрування та сертифікатів відкритих ключів. Це поле повинно мати значення «3».

3. Поле «authSafe» визначається структурою «ContentInfo», яка подана в нотації ASN.1 та визначена у ДСТУ ISO/IEC 8824-1:2009 «Інформаційні технології. Нотація абстрактного синтаксису 1 (ASN.1). Частина 1. Специфікація базової нотації», ДСТУ ISO/IEC 8824-2:2009 «Інформаційні технології. Нотація абстрактного синтаксису 1 (ASN.1). Частина 2. Специфікація інформаційного об'єкта», ДСТУ ISO/IEC 8824-3:2009 «Інформаційні технології. Нотація абстрактного синтаксису 1 (ASN.1). Частина 3. Специфікація обмежень», ДСТУ ISO/IEC 8824-4:2009 «Інформаційні технології. Нотація абстрактного синтаксису 1 (ASN.1). Частина 4. Параметризація специфікацій ASN.1» (далі – ISO/IEC 8824).

Цими Вимогами дозволяється використання даних у структурі «ContentInfo» типу «дані», що визначаються об'єктним ідентифікатором

```
id-data OBJECT IDENTIFIER ::= {iso(1) member-body(2) us(840)  
rsadsi(113549) pkcs(1) pkcs7(7) 1}
```

Поле «content» структури «authSafe» містить структуру типу «AuthenticatedSafe»:

```
AuthenticatedSafe ::= SEQUENCE OF ContentInfo.
```

4. Поле «macData» є необов'язковим та використовується для контролю цілісності структури «authSafe»:

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

Відповідно до цих Вимог поле «macData» повинно завжди бути присутнім.

IV. Типи даних структури «AuthenticatedSafe»

1. Цими Вимогами дозволяється використання даних у структурі «AuthenticatedSafe» типу «дані» або «зашифровані дані». Тип даних «дані» визначаються ідентифікатором «id-data», а тип даних «зашифровані дані» – ідентифікатором:

```
pkcs-7 OBJECT IDENTIFIER ::= {iso(1) member-body(2) US(840)
rsadsi(113549) pkcs(1) 7}
encryptedData OBJECT IDENTIFIER ::= {pkcs-7 6}.
```

2. Формат даних типу «зашифровані дані» задається структурою «EncryptedData»:

```
EncryptedData ::= SEQUENCE {
    version          CMSVersion,
    encryptedContentInfo EncryptedContentInfo,
    unprotectedAttrs [1] IMPLICIT UnprotectedAttributes OPTIONAL}.
```

2.1. Поле «version» містить версію формату «EncryptedData». У разі наявності поля «unprotectedAttrs» це поле повинно мати значення «2», в іншому випадку – значення «0».

2.2. Поле «encryptedContentInfo» визначається Вимогами до форматів криптографічних повідомлень та містить зашифровані дані. Об'єктний ідентифікатор повинен вказувати на криптоалгоритм PBES2, параметри криптоалгоритму повинні бути представлені структурою «PBES2-params» як наведено у Вимогах до алгоритмів формування ключів шифрування ключів.

2.3. Поле «unprotectedAttrs» є необов'язковим та містить набір незашифрованих атрибутів.

3. Дані у структурі «AuthenticatedSafe» типу «дані» або «зашифровані дані» містять об'єкт «SafeContents», який зберігається у відкритому вигляді або зашифрованому вигляді:

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

3.1. Поле «bagId» визначає тип даних структури «SafeBag».

3.2. Поле «bagValue» містить об'єкт даних зазначеного типу у пункті 3.1 цього розділу.

3.3. Поле «bagAttributes» є необов'язковим та містить набір атрибутів:

```
PKCS12Attribute ::= SEQUENCE {  
    attrId      ATTRIBUTE.&id ({PKCS12AttrSet}),  
    attrValues  SET OF ATTRIBUTE.&Type ({PKCS12AttrSet}{@attrId}}).
```

3.4. Контейнер зберігання особистого ключа та сертифіката визначається відповідно до Прикладу 1 Прикладів ASN.1 структури контейнеру зберігання особистих ключів та сертифікатів, що додаються до цих Вимог.

V. Типи даних структури «SafeBag»

1. Відповідно до PKCS#12 цими вимогами визначається шість типів даних, які можуть бути використані у структурі «SafeContents»:

```
PKCS12BagSet BAG-TYPE ::= {  
    keyBag |  
    pkcs8ShroudedKeyBag |  
    certBag |  
    crlBag |  
    secretBag |  
    safeContentsBag }.
```

2. Тип даних «KeyBag» визначається як

```
KeyBag ::= PrivateKeyInfo
```

та може містити лише один контейнер з особистим ключем, формат якого відповідає Вимогам до форматів транспортних контейнерів особистих ключів. На тип даних «KeyBag» вказує ідентифікатор:

```
pkcs-12 OBJECT IDENTIFIER ::= {iso(1) member-body(2) US(840)  
rsadsi(113549) pkcs(1) 12}  
bagtypes OBJECT IDENTIFIER ::= {pkcs-12 10 1}  
BAG-TYPE ::= TYPE-IDENTIFIER  
keyBag BAG-TYPE ::= {KeyBag IDENTIFIED BY {bagtypes 1}}.
```

3. Тип даних «PKCS8ShroudedKeyBag» визначається як

```
PKCS8ShroudedKeyBag ::= EncryptedPrivateKeyInfo
```

та може містити лише один захищений контейнер з особистим ключем, формат якого відповідає Вимогам до форматів транспортних контейнерів особистих ключів. На тип даних «PKCS8ShroudedKeyBag» вказує ідентифікатор:

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

4. Тип даних «CertBag» використовується для зберігання сертифіката відкритого ключа:

```
CertBag ::= SEQUENCE {  
    certId      BAG-TYPE.&id ({CertTypes}),  
    certValue   [0] EXPLICIT BAG-TYPE.&Type ({CertTypes}{@certId}})
```

```

pkcs-9 OBJECT IDENTIFIER ::= {iso(1) member-body(2) us(840)
rsadsi(113549) pkcs(1) pkcs-9(9)}
certTypes OBJECT IDENTIFIER ::= {pkcs-9 22}
x509Certificate BAG-TYPE ::= {
    OCTET STRING IDENTIFIED BY {certTypes 1}}
sdsiCertificate BAG-TYPE ::= {
    IA5String IDENTIFIED BY {certTypes 2}}
CertTypes BAG-TYPE ::= {
    x509Certificate |
    sdsiCertificate}

```

На тип даних «CertBag» вказує ідентифікатор:

```
certBag BAG-TYPE ::= {CertBag IDENTIFIED BY {bagtypes 3}}.
```

4.1. Поле «certId» містить ідентифікатор типу сертифіката. Цими Вимогами дозволяється використання сертифікату типу «x509Certificate».

4.2. Відповідно до п. 5.4.1 Поле «certValue» містить строку октетів з DER-кодованим значенням сертифіката відкритого ключа відповідно до Вимог до формату посиленого сертифіката відкритого ключа.

5. Тип даних «CRLBag» використовується для зберігання списку відкликаних сертифікатів (CBC):

```

CRLBag ::= SEQUENCE {
    crlId      BAG-TYPE.&id ({CRLTypes}),
    crlValue   [0] EXPLICIT BAG-TYPE.&Type ({CRLTypes}{@crlId})}
crlTypes OBJECT IDENTIFIER ::= {pkcs-9 23}
x509CRL BAG-TYPE ::= {OCTET STRING IDENTIFIED BY {crlTypes 1}}
CRLTypes BAG-TYPE ::= {x509CRL}

```

На тип даних «CRLBag» вказує ідентифікатор:

```
crlBag BAG-TYPE ::= {CRLBag IDENTIFIED BY {bagtypes 4}}.
```

5.1. Поле «crlId» містить ідентифікатор типу CBC.

5.2. Поле «crlValue» містить строку октетів з DER-кодованим значенням CBC відповідно до Вимог до формату списку відкликаних сертифікатів.

6. Тип даних «SecretBag» призначений для зберігання особистої інформації користувача та не є предметом цих Вимог.

7. Тип даних «SafeContents» містить структуру «SafeContents» та призначений для рекурсивного зберігання вкладених типів «SafeBag». Тип даних «SafeContents» не є предметом цих Вимог.

8. Структура «SafeContents» визначається відповідно до Прикладу 2 Прикладів ASN.1 структури контейнеру зберігання особистих ключів та сертифікатів, що додаються до цих Вимог.

VI. Параметри структури «SafeBag»

1. Цими Вимогами дозволяється використання атрибута «localKeyId» для даних типу «PKCS8ShroudedKeyBag» та «KeyBag» у полі «bagAttributes»:

```
localKeyId ATTRIBUTE ::= {  
    WITH SYNTAX OCTET STRING  
    EQUALITY MATCHING RULE octetStringMatch  
    SINGLE VALUE TRUE  
    ID pkcs-9-at-localKeyId}  
pkcs-9-at-localKeyId OBJECT IDENTIFIER ::= {pkcs-9 21}.
```

2. Атрибут «localKeyId» не є обов'язковим та містить ідентифікатор відкритого ключа відповідно до Вимог до формату посиленого сертифіката відкритого ключа.

У випадку відсутності розширення «localKeyId», отримання ідентифікатора відкритого ключа для забезпечення перевірки відповідності особистого ключа електронного цифрового підпису або особистого ключа шифрування відкритому ключу, що міститься в сертифікаті відкритого ключа підписувача або відправника, здійснюється у порядку, визначеному у розділі V Вимог до форматів транспортних контейнерів особистих ключів.

**Начальник Управління
функціонування центрального
засвідчувального органу
Міністерства юстиції України**

Д.В. Журавльов

**Директор Департаменту криптографічного
захисту інформації Адміністрації
Державної служби спеціального зв'язку
та захисту інформації України**

А.І. Пушкарьов

Додаток
до Вимог до форматів
контейнерів зберігання особистих
ключів електронного цифрового
підпису, особистих
ключів шифрування та сертифікатів
відкритих ключів
(підпункт 3.4 пункту 4
розділу IV, пункт 8 розділу V)

ПРИКЛАДИ

ASN.1 структури контейнеру зберігання особистих ключів та сертифікатів

Приклад 1. Контейнер зберігання особистого ключа та сертифіката

Пароль – «password».

```
SEQUENCE :
  INTEGER : 3
  SEQUENCE :
    OBJECT IDENTIFIER : data [1.2.840.113549.1.7.1]
    CONTEXT SPECIFIC (0) :
      OCTET STRING :
        SEQUENCE :
          SEQUENCE :
            OBJECT IDENTIFIER : data [1.2.840.113549.1.7.1]
            CONTEXT SPECIFIC (0) :
              OCTET STRING :
                SEQUENCE :
                  SEQUENCE :
                    OBJECT IDENTIFIER :
                    pkcs-12-pkcs-8ShroudedKeyBag [1.2.840.113549.1.12.10.1.2]
                    CONTEXT SPECIFIC (0) :
                      SEQUENCE :
                        SEQUENCE :
                          OBJECT IDENTIFIER :
                          id-PBES2 [1.2.840.113549.1.5.13]
                          SEQUENCE :
                            SEQUENCE :
                              OBJECT IDENTIFIER :
                              id-PBKDF2 [1.2.840.113549.1.5.12]
                              SEQUENCE :
                                OCTET STRING :
```

31A58DC1462981189CF6C701E27
6C7553A5AB5F6E36D8418E4AA40
C930CF3876

INTEGER : 10000

SEQUENCE :

OBJECT IDENTIFIER :

id-hmacGost34311 [1.2.804.2.1.1.1.1.2]

NULL :

SEQUENCE :

OBJECT IDENTIFIER :

id-gost28147-cfb [1.2.804.2.1.1.1.1.1.3]

SEQUENCE :

OCTET STRING :

4BB10F5C2945D49E

OCTET STRING :

A9D6EB45F13C708280

C4967B231F5EADF658

EBA4C037291D38D96B

F025CA4E17F8E9720D

C615B43A28975F0BC1

DEA36438B564EA2C17

9FD0123E6DB8FAC579

04

OCTET STRING :

29A22E2951E632E1E444A

E38F521C890FF6377FC05

39113A66720BFC4E9107C

566A07E3EAB9AE67F337E

D9C66C021363E79508A9F

DFA09E78877DFBE765431

60DC83195427A9C7FF2F6

F40D8D0FEA26583C72EF6

E5E2045DA9512A61FBC2B

9573E8B0BDC8F034D8CDA

3ACA63B78C9877FA75C22

8756BE76083A235247A09

4C1EF2996FFBFCB45E6D1

4807B38E26A8626103513

1DEC63B37307B44EF2C0E

AFE51392CD8A2B8B50FC6

F8BC8B1A62EFD276D4E81

BB358F4931BAAA3660C0C

0B5DF52E5233D90D1F4EF

5203C40F036CF59129146
60BF28212C9B3FD9141CB
89B93C13522DEB33085A2
5CC102B5B7DBA377078A6
45E88

SEQUENCE :

OBJECT IDENTIFIER : encryptedData [1.2.840.113549.1.7.6]

CONTEXT SPECIFIC (0) :

SEQUENCE :

INTEGER : 0

SEQUENCE :

OBJECT IDENTIFIER : data [1.2.840.113549.1.7.1]

SEQUENCE :

OBJECT IDENTIFIER : id-PBES2 [1.2.840.113549.1.5.13]

SEQUENCE :

SEQUENCE :

OBJECT IDENTIFIER :

id-PBKDF2 [1.2.840.113549.1.5.12]

SEQUENCE :

OCTET STRING :

9F93C3D9B8CB403374

434DA22DFFC397488C

A2251FEB8E9DA65E64

5E594BCEC0

INTEGER : 10000

SEQUENCE :

OBJECT IDENTIFIER :

id-hmacGost34311 [1.2.804.2.1.1.1.1.2]

NULL :

SEQUENCE :

OBJECT IDENTIFIER :

id-gost28147-cfb [1.2.804.2.1.1.1.1.1.3]

SEQUENCE :

OCTET STRING :

9F11E6430C51E266

OCTET STRING :

A9D6EB45F13C708280

C4967B231F5EADF658

EBA4C037291D38D96B

F025CA4E17F8E9720D

C615B43A28975F0BC1

DEA36438B564EA2C17

9FD0123E6DB8FAC579

04

CONTEXT SPECIFIC (0) :

3B6BAC1A6C39CC80A25616FC6987A3
1EAF44E4E0D145C7B5F15B218EDA74
FCF85D8A4FF456F91DF60F170FA10B
288040E7E29759ED8076A16ABE21B7
73DA361DC04B0650A7E17981F98C4C
D35E2DAD6FCA1DA147D0983450E4F1
43E2B1CD1E3303B10AAEF1419EE174
2EC79CD41CE771ABFDC5B0CB4ADAEC
ED2586B311154BB19A2A141E1642E2
72B9DA853D1E627A003F8562571F8E
42B05CBAF2B243C9DCDBE344DB3206
7E40600BC60E958C397599F5DF47AB
92B4B62FFC9133BD460C4D52692D03
068A6058A543E4731654752037EED7
A73947E9E83BC5A74844C067712E03
2D137200FFBD9BEFDF68D559025AA5
C717FE259D8D9597A805872BF20884
0F888831AAF4213302CFFC237609BF
7AE51BC71A24CBB6C6DB03AC1F7A59
20109D410152E74A8C27DCCAEA688C
E46FF75481E8B3C1AE90EDA6B7B663
3D3AAAFAFDCA080D96F8BC600831D1
AF6F617781400188F301D69A716B08
012FB57276B4EA5A844D39A71888A7
058F47E52C5FAAC4FAC16201CFEAF9
811535FEC0FA7439A247DAFC611891
02FD00E3B340F4D1C61A18C082BDC4
700749AC609CD5532E2E295BA0302E
7E59C2A3E12B95F9EE5D90BB9DBB66
F7A9ADD26733C26A44105678342773
6F83B53B7531CCF009499FA14931FB
3F7859684B2520636CDBC4BAF6D126
8459156BCFB912DF26CE4A8224E627
072D92F20DEC249A5F27EDA67C1A39
23F5E75CA24355388E828B04FC69D3
BF08CBC4B68DA598DAA7A1665ACADE
F470E3D712E0E13E47108F06D009B1
9CB4A6131797F741B09A899EFAFED5
3FC7344064FB17676F50AE7ACBF33C
FEF973CBD7403FA3D77BF280368F97
E5AF489BB26DDC4CFFE2B821235400

8A07CB1DA400BC6B2BD3CD098FE0E2
29D69C2C1E2103600BFFB5B459743E
C775FC87888EF945DDE5F2D33F97D9
2CA665FACB9EDCFC2B146E70DC0808
02174C2F293C27E2B2819EBA82DAE7
5F7B989E887FE9539CC5F041F1E916
94D81DFBCC3E4945004CA962270435
2E483E8238146C88E312C0E7C1EC14
7A9DBB642752330ED08115606B8EB3
1212FE29AE858BF26F8F2AEB289A0B
36CCFA405BBA8A2F067456A2B8A5BA
77D4BDBC0D0004C4DE6896A6A2D9E1
B4ACCB3799129D17F46696E2994AE5
877539761308DF7B45D2D95DB2EAC9
D7B75EC7BF11A1AC8502CDC111D004
16405404BB2DDB38DF37B3F6740DCF
6647CAEB7E9FDDEBE798A87E8A5A97
4E0730C1495E6345BE934103E6

SEQUENCE :

SEQUENCE :

SEQUENCE :

OBJECT IDENTIFIER : [1.2.804.2.1.1.1.2.1]

NULL : "

OCTET STRING :

9DD623DE32AB6A09B16C442D8F34195F02182F3ED34FD09E
76F817FB648E725A

OCTET STRING :

EAB98DB1A017DF6613BCBD87501FCE27A21EC76EF00DD8DF00D
229A53B0BB67C

INTEGER : 10000

Приклад 2. Приклад структури «SafeContents»

Пароль – «password».

SEQUENCE :

SEQUENCE :

OBJECT IDENTIFIER :

pkcs-12-pkcs-8ShroudedKeyBag [1.2.840.113549.1.12.10.1.2]

CONTEXT SPECIFIC (0) :

SEQUENCE :

SEQUENCE :

CONTEXT SPECIFIC (0) :

INTEGER : 2

INTEGER :

68E9687A597F245F01000000010000002A000000

SEQUENCE :

OBJECT IDENTIFIER : id-dstu4145PB [1.2.804.2.1.1.1.3.1.1]

SEQUENCE :

SET :

SEQUENCE :

OBJECT IDENTIFIER : organizationName [2.5.4.10]

UTF8 STRING : 'Організація'

SET :

SEQUENCE :

OBJECT IDENTIFIER : organizationalUnitName [2.5.4.11]

UTF8 STRING : 'Підрозділ'

SET :

SEQUENCE :

OBJECT IDENTIFIER : commonName [2.5.4.3]

UTF8 STRING : 'ЦСК'

SET :

SEQUENCE :

OBJECT IDENTIFIER : serialNumber [2.5.4.5]

UTF8 STRING : 'UA-01'

SET :

SEQUENCE :

OBJECT IDENTIFIER : countryName [2.5.4.6]

PRINTABLE STRING : 'UA'

SET :

SEQUENCE :

OBJECT IDENTIFIER : localityName [2.5.4.7]

UTF8 STRING : 'Київ'

SEQUENCE :

[illegible]

7D470D
OCTET STRING :
B60FD2D8DCE8A93423C6101BCA91C4
7A007E6C300B26CD556C9B0E7D20EF
292A00
OCTET STRING :
A9D6EB45F13C708280C4967B231F5EADF
658EBA4C037291D38D96BF025CA4E17F8
E9720DC615B43A28975F0BC1DEA36438B
564EA2C179FD0123E6DB8FAC57904
BIT STRING UnusedBits:0 :
OCTET STRING :
ADE7A73D54E9650575BE685700C31B31823D
2C8C131ADF24A2028F6598DD20A001
CONTEXT SPECIFIC (3) :
SEQUENCE :
SEQUENCE :
OBJECT IDENTIFIER : subjectKeyIdentifier [2.5.29.14]
OCTET STRING :
OCTET STRING :
E8E9687A597F245F666575A298F35B
276FFA09696274FB63FB6D33E874A9
F5DC
SEQUENCE :
OBJECT IDENTIFIER : authorityKeyIdentifier [2.5.29.35]
OCTET STRING :
SEQUENCE :
CONTEXT SPECIFIC (0) :
E8E9687A597F245F666575A298F
35B276FFA09696274FB63FB6D33
E874A9F5DC
SEQUENCE :
OBJECT IDENTIFIER : keyUsage [2.5.29.15]
BOOLEAN : 'y'
OCTET STRING :
BIT STRING UnusedBits:1 :
06
SEQUENCE :
OBJECT IDENTIFIER : certificatePolicies [2.5.29.32]
BOOLEAN : 'y'
OCTET STRING :
SEQUENCE :
SEQUENCE :

OBJECT IDENTIFIER : [1.2.804.2.1.1.1.2.2]
SEQUENCE :
OBJECT IDENTIFIER : basicConstraints [2.5.29.19]
BOOLEAN : 'y'
OCTET STRING :
SEQUENCE :
BOOLEAN : 'y'
INTEGER : 2
SEQUENCE :
OBJECT IDENTIFIER : [1.3.6.1.5.5.7.1.3]
BOOLEAN : 'y'
OCTET STRING :
SEQUENCE :
SEQUENCE :
OBJECT IDENTIFIER : [1.2.804.2.1.1.1.2.1]
SEQUENCE :
OBJECT IDENTIFIER : id-dstu4145PB [1.2.804.2.1.1.1.3.1.1]
BIT STRING UnusedBits:0 :
OCTET STRING :
E66B694671447222449D08A43EB5BADB8BD418639E
7CF545E5AD6FC3984FCA5AB02F1BA4A8DA875C7998
D9ACC6847E467C6CA340CB0A31A57EF6E11BD84ACB
19