ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЗАТВЕРДЖЕНИЙ
ЄААД.21115-13-ЛЗ

Підп. та дата

Інв. № дубл

Взам. інв. №

Підп. та дата

Інв. № оригі.

Програмний комплекс користувача ЦСК
Бібліотека користувача ЦСК “ДПА (файли, COM)“

Версія 1.3.1.2

Опис програми
Настанова програміста
Настанова системного програміста

ЄААД.21115-13 13/32/33 01-1

Харків 2013 р.

АНОТАЦІЯ

Даний документ містить опис програми та настанови програмісту і системному програмісту на програмний компонент (бібліотеку) користувача ЦСК "ДПА (файли, COM)"

ЗМІСТ

Пор. № зміни	*Підпис відпов. особи*	*Дата внесення*

Пор. № зміни	*Підпис відпов. особи*	*Дата внесення*

1 ЗАГАЛЬНІ ВІДОМОСТІ

1.1 Найменування: програмний компонент “ІІТ Користувач ЦСК-1.3. Бібліотека “ДПА (файли, COM)” (далі – бібліотека).

1.2 Призначення: бібліотека призначена для виконання функцій пов’язаних із:

- роботою з носіями ключової інформації (НКІ);
- роботою з файловими сховищами сертифікатів та списків відкликаних сертифікатів (СВС);
- роботою з LDAP-каталогом;
- перевіркою статусу сертифікатів через протокол OCSP;
- формування та перевірку захищених файлів при інформаційній взаємодії платників податків і податкових органів в електронному вигляді телекомунікаційними каналами зв’язку з використанням електронного цифрового підпису
- отримуванням позначок часу тощо.

1.3 Бібліотека виконана у вигляді файлу бібліотеки динамічного компонування EUTaxServiceFile.dll що реєструється в ОС в якості COM-об’єкту.

1.4 Бібліотека функціонує на базі ПЕОМ з процесором типу Intel із тактовою частотою не менше 100 МГц з встановленою ОС Microsoft Windows XP/2003 Server/Vista/2008 Server/7/8/2012 Server.

1.5 Для функціонування бібліотеки необхідні компоненти, які наведені у табл. 1.1.

Таблиця 1.1 – Компоненти, які необхідні для функціонування бібліотеки.

Найменування	Примітки
CAConnectors.dll – бібліотека з’єднань	файл бібліотеки динамічного компонування для ОС Microsoft Windows XP/2003 Server/ Vista/2008 Server/7/8/2012 Server
CAGUI.dll – бібліотека графічного інтерфейсу	файл бібліотеки динамічного компонування для ОС Microsoft Windows XP/2003 Server/ Vista/2008 Server/7/8/2012 Server
CSPBase.dll – бібліотека криптографічних перетворень	файл бібліотеки динамічного компонування для ОС Microsoft Windows XP/2003 Server/ Vista/2008 Server/7/8/2012 Server
CSPExtension.dll – бібліотека розширення криптографічних перетворень	файл бібліотеки динамічного компонування для ОС Microsoft Windows XP/2003 Server/ Vista/2008 Server/7/8/2012 Server
KM.dll – базова бібліотека роботи з носіями ключової інформації	файл бібліотеки динамічного компонування для ОС Microsoft Windows XP/2003 Server/ Vista/2008 Server/7/8/2012 Server
KM.*.dll – бібліотеки роботи з носіями ключової інформації	файли бібліотек динамічного компонування для ОС Microsoft Windows XP/2003 Server/ Vista/2008 Server/7/8/2012 Server
LDAPClient.dll – бібліотека роботи з LDAP-сервером	файл бібліотеки динамічного компонування для ОС Microsoft Windows XP/2003 Server/ Vista/2008 Server/7/8/2012 Server
PKIFormats.dll – бібліотека роботи з форматами даних	файл бібліотеки динамічного компонування для ОС Microsoft Windows XP/2003 Server/ Vista/2008 Server/7/8/2012 Server
RF.dll – бібліотека роботи з файлами конфігурації	файл бібліотеки динамічного компонування для ОС Microsoft Windows 2000/XP/2003 Server/ Vista/2008 Server/7/8/2012 Server

2 ФУНКЦІОНАЛЬНЕ ПРИЗНАЧЕННЯ

2.1 Бібліотека виконує наступні функції:

1) роботу з НКІ:
   - зчитування особистого ключа з НКІ;
   - зміну паролю захисту особистого ключа на НКІ;
   - знищення особистого ключа на НКІ;
   - резервне копіювання особистого ключа на НКІ;
2) роботу із файловими сховищами сертифікатів та СВС, що складається з:
   - зчитування сертифікатів та СВС із файлового сховища;
   - перелічення сертифікатів у сховищі;
   - визначення статусу сертифікату за допомогою СВС;
   - отримання інформації про сертифікат;
   - завантаження СВС з HTTP-сервера;
3) роботу з позначками часу:
   - отримання позначок часу від TSP-сервера;
   - виділення та перевірку позначок часу у підписаних або зашифрованих даних;
   - отримання інформації про позначки часу;
4) перевірку статусу сертифікату за протоколом OCSP;
5) пошук та отримання сертифікату з LDAP-каталогу;
6) формування та перевірку захищених файлів при інформаційній взаємодії платників податків і податкових органів в електронному вигляді телекомунікаційними каналами зв'язку з використанням електронного цифрового підпису

3 РОБОТА З БІБЛІОТЕКОЮ

3.1 Інсталяція бібліотеки

Для інсталяції програми(бібліотеки) необхідно запустити програму інсталяції (майстер інсталяції) EUTaxServiceFile.exe з інсталяційного носія (оптичного диску чи ін.) чи завантажити її з web-сторінки ЦСК.

Після запуску програми інсталяції на першій сторінці (рис. 3.1) виводиться інформація про початок інсталяції. Для продовження інсталяції необхідно натиснути кнопку “Далі”, а для завершення – “Відміна”.

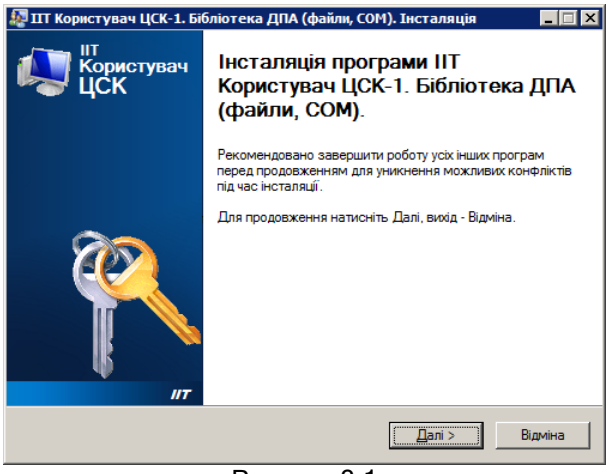

Рисунок 3.1

На наступній сторінці майстра (рис. 3.2) за необхідністю можна вказати каталог на диску до якого буде встановлено програму. Для продовження інсталяції необхідно натиснути кнопку “Далі”.

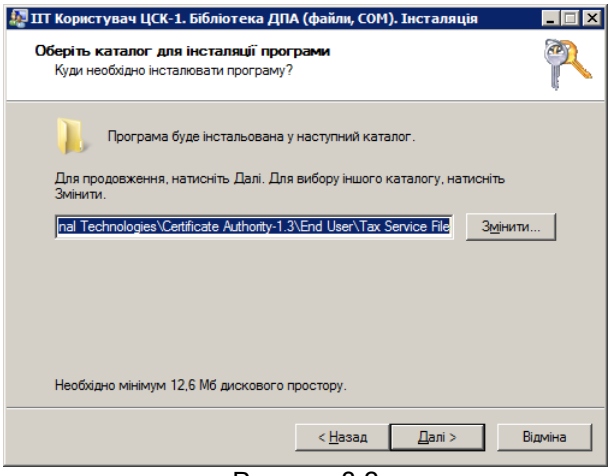

Рисунок 3.2

Після інсталяції програми, майстер завершує свою роботу.

3.2 Завантаження бібліотеки

Бібліотека EUTaxServiceFile.dll завантажується прикладними засобами ОС.

3.3 Параметри роботи бібліотеки

Параметри роботи бібліотеки зберігаються у системному реєстрі ОС в ключі HKEY_CURRENT_USER\Software\Institute of Informational Technologies\Certificate Authority-1.3\End User. До складу параметрів входять наступні параметри, які наведені у табл. 3.1.

Таблиця 3.1 – Параметри роботи бібліотеки у системному реєстрі ОС.

Гілка реєстру та назва параметра	Опис та можливі значення
FileStore\\	Параметри файлового сховища сертифікатів та СВС
Path	Каталог, в якому розміщуються сертифікати та СВС

CheckCRLs	Признак необхідності використання СВС при визначенні статусу сертифікату (1 – використовувати СВС, 0 – не використовувати)
AutoRefresh	Признак необхідності автоматичного виявлення змін у файловому сховищі при записі, модифікації чи видаленні сертифікатів та СВС (1 – контролювати зміни, 0 – не контролювати)
OnlyOwnCRLs	Признак необхідності використання СВС тільки власного ЦСК користувача (1 – використовувати СВС тільки власного ЦСК, 0 – використовувати всі СВС). Не має значення, якщо признак CheckCRLs не встановлено
FullAndDeltaCRLs	Признак необхідності перевірки наявності двох діючих СВС – повного та часткового (1 – перевіряти наявність двох діючих СВС, 0 – перевіряти наявність хоча б одного діючого СВС). Не має значення, якщо признак CheckCRLs не встановлено
SaveLoadedCerts	Признак необхідності автоматичного збереження сертифікатів отриманих з LDAP-сервера чи за протоколом OCSP у файлове сховище (1 – зберігати, 0 – не зберігати)
ExpireTime	Час зберігання стану перевіреного сертифікату (у секундах)
AutoDownloadCRLs	Признак необхідності автоматичного завантаження СВС (1 – завантажувати автоматично, 0 – не завантажувати). Не має значення, якщо признак CheckCRLs не встановлено
OCSP\\	Параметри протоколу OCSP
Use	Признак необхідності використання механізму визначення статусу сертифікатів за допомогою протоколу OCSP. Інші параметри підрозділу не мають значення якщо даний признак не встановлено
BeforeFStore	Признак черговості перевірки статусу сертифікату (1 - статус сертифікату перевіряється спочатку за допомогою OCSP-протоколу, потім за допомогою файлового сховища, 0 – перевірка здійснюється спочатку за допомогою файлового сховища, а потім (якщо статус не визначено за файловим сховищем) - за допомогою OCSP-протоколу)
Address	IP-адреса або DNS-ім'я OCSP-сервера
Port	TCP-порт OCSP-сервера
Proxy\\	Параметри proxy-сервера
Use	Признак необхідності підключення до ЦСК через proxy-сервер (1 – підключатися через proxy-сервер, 0 – не підключатися). Інші параметри підрозділу не мають значення якщо даний признак не встановлено
Anonymous	Признак анонімного proxy-сервера. (1 – використовується анонімний proxy-сервер. 0 – для доступу до proxy-сервера потрібні ім'я користувача та пароль доступу).
Address	IP-адреса або DNS-ім'я proxy-сервера
Port	TCP-порт proxy-сервера
User	Ім'я користувача proxy-сервера
SavePassword	Признак зберігання пароля доступу до proxy-сервера у системному реєстрі (1 – пароль зберігається у параметрі Password, 0 – не зберігається)
Password	Пароль доступу користувача до proxy-сервера. Не має значення, якщо признак SavePassword не встановлено
TSP\\	Параметри протоколу TSP
GetStamps	Признак необхідності отримувати позначки часу під час формування підпису (1 – отримувати позначки часу, 0 – не отримувати) . Інші параметри підрозділу не мають значення якщо даний признак не встановлено
Address	IP-адреса або DNS-ім'я TSP-сервера
Port	TCP-порт TSP-сервера
LDAP\\	Параметри LDAP-сервера
Use	Признак необхідності використання LDAP-сервера (1 – використовувати, 0 – не використовувати). Інші параметри підрозділу не мають значення якщо даний признак не встановлено

Address	IP-адреса або DNS-ім'я LDAP-сервера
Port	TCP-порт LDAP-сервера
Anonimous	Признак анонімного доступу до LDAP-сервера (1 – анонімний доступ, 0 – доступ на основі імені користувача User та пароля Password)
User	Ім'я користувача LDAP –сервера Не має значення, якщо признак Anonimous встановлено
Password	Пароль доступу користувача до LDAP-сервера. Не має значення, якщо признак Anonimous встановлено
LookupCert	Признак необхідності пошуку сертифікатів у LDAP-каталозі (1 – шукати, 0 – не шукати)
CMP\\	Параметри CMP-сервера
Use	Признак необхідності використовувати CMP-сервер(1 – використовувати, 0 – не використовувати) . Інші параметри підрозділу не мають значення якщо даний признак не встановлено
Address	IP-адреса або DNS-ім'я CMP -сервера
Port	CMP -порт CMP -сервера
CommonName	Загальне ім'я CMP-сервера
Mode\\	Режими роботи
Offline	Признак роботи в off-line режимі (1 – off-line режим, 0 – on-line режим)

Параметри можуть бути встановлені шляхом безпосереднього запису у реєстр або шляхом виклику функції `SetSettings` із складу програмного інтерфейсу бібліотеки та введення параметрів у діалоговому вікні (див. дод. А).

3.4 Константи

3.4.1 EUKeyType

```
// Типи особистого ключа, що можуть бути встановлені та використані в бібліотеці
typedef enum EUKeyType
{
	euKeyTypeAccountant = 1,		// Особистий ключ бухгалтера
	euKeyTypeDirector = 2,			// Особистий ключ директора
	euKeyTypeDigitalStamp = 3		// Особистий ключ цифрової печатки
} EUKeyType;
```

3.5 Типи даних

Бібліотека використовує стандартні типи даних мови MIDL (Microsoft Interface Definition Language), а також такі типи, як BSTR (строка у вигляді масиву Unicode символів, що зберігає розмір), VARIANT (об'єкт цього типу дозволяє зберігати будь-який об'єкт іншого типу, ініціалізувати та змінювати його тип під час виконання програми), SAFEARRAY (об'єкт цього типу дозволяє роботу з динамічними одномірними та багатомірними масивами об'єктів).

Таблиця 3.2 – Відповідність типів бібліотеки.

MIDL	C#	JavaScript
VARIANT_BOOL	bool	bool (true, false)
LONG	int	int (знакове ціле)
BSTR	string	String (строковий тип)
SAFEARRAY	Array	Array (масив об'єктів)
VARIANT	Object (Array, string, string[], byte[],byte[][])	Object (Array, String, String [], Byte[],Byte[][])

3.5.1 IResultInfo

Опис методів з інтерфейсу об'єкта надається у вигляді IDL-опису, а також у вигляді опису за допомогою мови MIDL.

```
// Інтерфейс об'єкта, для отримання результату виконання операцій
```

```
[
	object,
	uuid(BDA88A0F-0AB4-41D3-B682-812AF2C2F23E),
	dual,
	nonextensible,
	pointer_default(unique)
]
```

3.5.1.1 Методи отримання інформації про помилку

– IsSuccessfull

```
// Метод отримання інформації про признак, того що операції завершилася успішно
// successfull            -      параметр, що повертається. Містить інформацію про
//                               признак того, що операції завершилася успішно

// Опис в файлі IDL:
[id(1)]
HRESULT IsSuccessfull(
	[out, retval] VARIANT_BOOL	*successfull);

// Опис мовою MIDL:
VARIANT_BOOL IsSuccessfull();
```

– GetStatusDescription

```
// Метод отримання опису статусу операції, що завершилася
// status                 -      параметр, що повертається. Містить інформацію про
//                               статус операції

// Опис в файлі IDL:
[id(2)]
HRESULT GetStatusDescription(
	[out, retval] BSTR		*status);

// Опис мовою MIDL:
BSTR GetStatusDescription();
```

3.5.1.2 Методи отримання результату обробки

– GetInputFileName

```
// Метод отримання імені файлу, що оброблявся
// fileName               -      параметр, що повертається. Містить інформацію про
//                               ім'я вхідного файлу

// Опис в файлі IDL:
[id(3)]
HRESULT GetInputFileName(
	[out, retval] BSTR		*fileName);

// Опис мовою MIDL:
BSTR GetInputFileName();
```

– GetOutputFileName

```
// Метод отримання імені файлу з результатом обробки
// fileName               -      параметр, що повертається. Містить інформацію про
//                               ім'я вихідного файлу

// Опис в файлі IDL:
[id(4)]
HRESULT GetOutputFileName(
	[out, retval] BSTR		*fileName);

// Опис мовою MIDL:
BSTR GetOutputFileName();
```

Пор. № зміни	*Підпис відпов. особи*	*Дата внесення*

– GetOutputData

```
// Метод отримання оброблених даних
// data                    -      параметр, що повертається. Містить дані, що
//                                було оброблені у вигляді масиву байт (SAFEARRAY)

// Опис в файлі IDL:
[id(29)]
HRESULT GetOutputData(
       [out, retval] VARIANT            *data);

// Опис мовою MIDL:
VARIANT GetOutputData();
```

3.5.1.3 Методи, що використовують графічний інтерфейс бібліотеки

– ShowInitiatorCertInfo

```
// Метод відображення інформації про сертифікати відправників файлу
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправників файлу

// Опис в файлі IDL:
[id(6)]
HRESULT ShowInitiatorCertInfo(
       [in] LONG                        initiatorIndex);

// Опис мовою MIDL:
void ShowInitiatorCertInfo(
       long                             initiatorIndex);
```

3.5.1.4 Методи отримання інформації про захищений файл

– GetInitiatorsCount

```
// Метод отримання інформації про кількість відправників файлу
// count                   -      параметр, що повертається. Містить інформацію про
//                                кількість відправників файлу

// Опис в файлі IDL:
[id(5)]
HRESULT GetInitiatorsCount(
       [out, retval] LONG               *count);

// Опис мовою MIDL:
long GetInitiatorsCount();
```

– IsTimeAvailable

```
// Метод отримання інформації про признак наявності часу підпису або мітки часу
// в отриманому від відправника файлі
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// avail                   -      параметр, що повертається. Містить інформацію про
//                                признак наявності часу підпису або мітки часу
//                                в отриманому від відправника файлі

// Опис в файлі IDL:
[id(24)]
HRESULT IsTimeAvailable(
       [in] LONG                        initiatorIndex,
       [out, retval] VARIANT_BOOL       *avail);

// Опис мовою MIDL:
VARIANT_BOOL IsTimeAvailable(
       long                             initiatorIndex);
```

– IsTimeStampExist

Пор. № зміни	Підпис відпов. особи	Дата внесення

```
// Метод отримання інформації про признак наявності мітки часу підпису в отриманому
// від відправника файлі
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// exist                   -      параметр, що повертається. Містить інформацію про
//                                признак наявності мітки часу підпису в отриманому
//                                від відправника файлі

// Опис в файлі IDL:
[id(25)]
HRESULT IsTimeStampExist(
       [in] LONG                        initiatorIndex,
       [out, retval] VARIANT_BOOL       *exist);

// Опис мовою MIDL:
VARIANT_BOOL IsTimeStampExist(
       long                             initiatorIndex);
```

– GetTimeStamp

```
// Метод отримання інформації про час підпису або мітки часу з отриманого від
// відправника файлу
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// timeStamp               -      параметр, що повертається. Містить інформацію про
//                                час підпису або мітки часу з отриманого від
//                                відправника файлу

// Опис в файлі IDL:
[id(26)]
HRESULT GetTimeStamp(
       [in] LONG                        initiatorIndex,
       [out, retval] VARIANT            *timeStamp);

// Опис мовою MIDL:
VARIANT GetTimeStamp(
       long                             initiatorIndex);
```

– IsSigned

```
// Метод отримання інформації про признак захисту отриманого файлу за допомогою ЕЦП
// відправника файлу
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// digitalSigned           -      параметр, що повертається. Містить інформацію про
//                                признак захисту отриманого файлу за допомогою ЕЦП
//                                відправника файлу

// Опис в файлі IDL:
[id(27)]
HRESULT IsSigned(
       [in] LONG                        initiatorIndex,
       [out, retval] VARIANT_BOOL       *digitalSigned);

// Опис мовою MIDL:
VARIANT_BOOL IsSigned(
       long                             initiatorIndex);
```

– IsEnveloped

```
// Метод отримання інформації про признак захисту отриманого файлу за допомогою
// направленого шифрування з використанням особистого ключа відправника файлу
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// enveloped               -      параметр, що повертається. Містить інформацію про
//                                признак захисту отриманого файлу за допомогою
//                                направленого шифрування з використанням особистого
//                                ключа відправника файлу
```

```
// Опис в файлі IDL:
[id(28)]
HRESULT IsEnveloped(
       [in] LONG                        initiatorIndex,
       [out, retval] VARIANT_BOOL       *enveloped);

// Опис мовою MIDL:
VARIANT_BOOL IsEnveloped(
       long                             initiatorIndex);
```

3.5.1.5 Методи отримання інформації про відправника файла

– GetInitiatorCertIssuer

```
// Метод отримання інформації про ім'я ЦСК, що видав сертифікат відправника файлу
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// issuer                  -      параметр, що повертається. Містить інформацію про
//                                ім'я ЦСК, що видав сертифікат відправника файлу

// Опис в файлі IDL:
[id(7)]
HRESULT GetInitiatorCertIssuer(
       [in] LONG                        initiatorIndex,
       [out, retval] BSTR               *issuer);

// Опис мовою MIDL:
BSTR GetInitiatorCertIssuer(
       long                             initiatorIndex);
```

– GetInitiatorCertIssuerCN

```
// Метод отримання інформації про реквізити ЦСК, що видав сертифікат відправника файлу
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// issuerCN                -      параметр, що повертається. Містить інформацію про
//                                реквізити ЦСК, що видав сертифікат відправника
//                                файлу

// Опис в файлі IDL:
[id(8)]
HRESULT GetInitiatorCertIssuerCN(
       [in] LONG                        initiatorIndex,
       [out, retval] BSTR               *issuerCN);

// Опис мовою MIDL:
BSTR GetInitiatorCertIssuerCN(
       long                             initiatorIndex);
```

– GetInitiatorCertSerial

```
// Метод отримання інформації про серійний номер сертифікату відправника файлу
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// serial                  -      параметр, що повертається. Містить інформацію про
//                                серійний номер сертифікату відправника файлу

// Опис в файлі IDL:
[id(9)]
HRESULT GetInitiatorCertSerial(
       [in] LONG                        initiatorIndex,
       [out, retval] BSTR               *serial);

// Опис мовою MIDL:
BSTR GetInitiatorCertSerial(
       long                             initiatorIndex);
```

– GetInitiatorCertSubject

```
// Метод отримання інформації про ім’я власника сертифікату, який відправив файл
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// subject                 -      параметр, що повертається. Містить інформацію про
//                                ім’я власника сертифікату, який відправив файл

// Опис в файлі IDL:
[id(10)]
HRESULT GetInitiatorCertSubject(
       [in] LONG                        initiatorIndex,
       [out, retval] BSTR               *subject);

// Опис мовою MIDL:
BSTR GetInitiatorCertSubject(
       long                             initiatorIndex);
```

– GetInitiatorCertSubjectCN

```
// Метод отримання інформації про реквізити власника сертифікату, який відправив файл
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// subjectCN               -      параметр, що повертається. Містить інформацію про
//                                реквізити власника сертифікату, який відправив
//                                файл

// Опис в файлі IDL:
[id(11)]
HRESULT GetInitiatorCertSubjectCN(
       [in] LONG                        initiatorIndex,
       [out, retval] BSTR               *subjectCN);

// Опис мовою MIDL:
BSTR GetInitiatorCertSubjectCN(
       long                             initiatorIndex);
```

– GetInitiatorCertSubjectOrg

```
// Метод отримання інформації про організацію до якої належить власник сертифікату,
// який відправив файл
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// subjectOrg              -      параметр, що повертається. Містить інформацію про
//                                організацію до якої належить власник сертифікату,
//                                який відправив файл

// Опис в файлі IDL:
[id(12)]
HRESULT GetInitiatorCertSubjectOrg(
       [in] LONG                        initiatorIndex,
       [out, retval] BSTR               *subjectOrg);

// Опис мовою MIDL:
BSTR GetInitiatorCertSubjectOrg(
       long                             initiatorIndex);
```

– GetInitiatorCertSubjectOrgUnit

```
// Метод отримання інформації про підрозділ організації до якої належить власник
// сертифікату, який відправив файл
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// subjectOrg              -      параметр, що повертається. Містить інформацію про
//                                підрозділ організації до якої належить власник
//                                сертифікату, який відправив файл

// Опис в файлі IDL:
[id(13)]
HRESULT GetInitiatorCertSubjectOrgUnit(
       [in] LONG                        initiatorIndex,
```

```
      [out, retval] BSTR                *subjectOrgUnit);

// Опис мовою MIDL:
BSTR GetInitiatorCertSubjectOrgUnit(
      long                              initiatorIndex);
```

– GetInitiatorCertSubjectTitle

```
// Метод отримання інформації про посаду власника сертифікату, який відправив файл
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// subjectOrgUnit          -      параметр, що повертається. Містить інформацію про
//                                посаду власника сертифікату, який відправив файл

// Опис в файлі IDL:
[id(14)]
HRESULT GetInitiatorCertSubjectTitle(
      [in] LONG                         initiatorIndex,
      [out, retval] BSTR                *subjectTitle);

// Опис мовою MIDL:
BSTR GetInitiatorCertSubjectTitle(
      long                              initiatorIndex);
```

– GetInitiatorCertSubjectState

```
// Метод отримання інформації про державу до якої належить власник сертифікату, який
// відправив файл
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// subjectTitle            -      параметр, що повертається. Містить інформацію про
//                                державу до якої належить власник сертифікату, який
//                                відправив файл

// Опис в файлі IDL:
[id(15)]
HRESULT GetInitiatorCertSubjectState(
      [in] LONG                         initiatorIndex,
      [out, retval] BSTR                *subjectState);

// Опис мовою MIDL:
BSTR GetInitiatorCertSubjectState(
      long                              initiatorIndex);
```

– GetInitiatorCertSubjectLocality

```
// Метод отримання інформації про населений пункт до якого належить власник
// сертифікату, який відправив файл
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// subjectState            -      параметр, що повертається. Містить інформацію про
//                                населений пункт до якого належить власник сертифікату,
//                                який відправив файл

// Опис в файлі IDL:
[id(16)]
HRESULT GetInitiatorCertSubjectLocality(
      [in] LONG                         initiatorIndex,
      [out, retval] BSTR                *subjectLocality);

// Опис мовою MIDL:
BSTR GetInitiatorCertSubjectLocality(
      long                              initiatorIndex);
```

– GetInitiatorCertSubjectFullName

```
// Метод отримання інформації про повне ім'я власника сертифікату, який відправив файл
// initiatorIndex          -      параметр, що передається. Містить інформацію про
```

Пор. № зміни	Підпис відпов. особи	Дата внесення

```
//                                  індекс відправника файлу
// subjectFullName          -      параметр, що повертається. Містить інформацію про
//                                  повне ім'я власника сертифікату, який відправив файл

// Опис в файлі IDL:
[id(17)]
HRESULT GetInitiatorCertSubjectFullName(
       [in] LONG                          initiatorIndex,
       [out, retval] BSTR                 *subjectFullName);

// Опис мовою MIDL:
BSTR GetInitiatorCertSubjectFullName(
       long                               initiatorIndex);
```

– GetInitiatorCertSubjectAddress

```
// Метод отримання інформації про адресу власника сертифікату, який відправив файл
// initiatorIndex           -      параметр, що передається. Містить інформацію про
//                                  індекс відправника файлу
// subjectAddress           -      параметр, що повертається. Містить інформацію про
//                                  адресу власника сертифікату, який відправив файл

// Опис в файлі IDL:
[id(18)]
HRESULT GetInitiatorCertSubjectAddress(
       [in] LONG                          initiatorIndex,
       [out, retval] BSTR                 *subjectAddress);

// Опис мовою MIDL:
BSTR GetInitiatorCertSubjectAddress(
       long                               initiatorIndex);
```

– GetInitiatorCertSubjectPhone

```
// Метод отримання інформації про телефон власника сертифікату, який відправив файл
// initiatorIndex           -      параметр, що передається. Містить інформацію про
//                                  індекс відправника файлу
// subjectPhone             -      параметр, що повертається. Містить інформацію про
//                                  телефон власника сертифікату, який відправив файл

// Опис в файлі IDL:
[id(19)]
HRESULT GetInitiatorCertSubjectPhone(
       [in] LONG                          initiatorIndex,
       [out, retval] BSTR                 *subjectPhone);

// Опис мовою MIDL:
BSTR GetInitiatorCertSubjectPhone(
       long                               initiatorIndex);
```

– GetInitiatorCertSubjectEMail

```
// Метод отримання інформації про адресу електронної пошти власника сертифікату,
// який відправив файл
// initiatorIndex           -      параметр, що передається. Містить інформацію про
//                                  індекс відправника файлу
// subjectEMail             -      параметр, що повертається. Містить інформацію про
//                                  адресу електронної пошти власника сертифікату, який
//                                  відправив файл

// Опис в файлі IDL:
[id(20)]
HRESULT GetInitiatorCertSubjectEMail(
       [in] LONG                          initiatorIndex,
       [out, retval] BSTR                 *subjectEMail);

// Опис мовою MIDL:
BSTR GetInitiatorCertSubjectEMail(
       long                               initiatorIndex);
```

– GetInitiatorCertSubjectDNS

```
// Метод отримання інформації про DNS-ім`я чи інше технічного засобу, що відправив файл
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// subjectDNS              -      параметр, що повертається. Містить інформацію про
//                                DNS-ім`я чи інше технічного засобу, що відправив файл

// Опис в файлі IDL:
[id(21)]
HRESULT GetInitiatorCertSubjectDNS(
       [in] LONG                        initiatorIndex,
       [out, retval] BSTR               *subjectDNS);

// Опис мовою MIDL:
BSTR GetInitiatorCertSubjectDNS(
       long                             initiatorIndex);
```

– GetInitiatorCertSubjectEDRPOUCode

```
// Метод отримання інформації про ЄДРПОУ код власника сертифікату, який відправив файл
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// subjectEDRPOUCode       -      параметр, що повертається. Містить інформацію про
//                                ЄДРПОУ код власника сертифікату, який відправив файл

// Опис в файлі IDL:
[id(22)]
HRESULT GetInitiatorCertSubjectEDRPOUCode(
       [in] LONG                        initiatorIndex,
       [out, retval] BSTR               *subjectEDRPOUCode);

// Опис мовою MIDL:
BSTR GetInitiatorCertSubjectEDRPOUCode(
       long                             initiatorIndex);
```

– GetInitiatorCertSubjectDRFOCode

```
// Метод отримання інформації про ДРФО код власника сертифікату, який відправив файл
// initiatorIndex          -      параметр, що передається. Містить інформацію про
//                                індекс відправника файлу
// subjectDRFOCode         -      параметр, що повертається. Містить інформацію про
//                                ДРФО код власника сертифікату, який відправив файл

// Опис в файлі IDL:
[id(23)]
HRESULT GetInitiatorCertSubjectDRFOCode(
       [in] LONG                        initiatorIndex,
       [out, retval] BSTR               *subjectDRFOCode);

// Опис мовою MIDL:
BSTR GetInitiatorCertSubjectDRFOCode(
       long                             initiatorIndex);
```

3.5.2 IExpiredPrivateKeyNotify

Опис методів з інтерфейсу об’єкта надається у вигляді IDL-опису, а також у вигляді опису за допомогою мови MIDL.

```
// Інтерфейс повідомлення про закінчення строку дії особистого ключа
[
       uuid(6AB24638-1261-4463-9C85-92A012C43811)
]
```

– ExpiredPrivateKeyNotify

```
// Метод отримання інформації про признак використання особистого ключа у разі
// закінчення його строку дії
```

Пор. № зміни	*Підпис відпов. особи*	*Дата внесення*

```
// use                      -      параметр, що повертається. Містить інформації про
//                                 признак використання особистого ключа у разі
//                                 закінчення його строку дії

// Опис в файлі IDL:
[id(1)]
HRESULT ExpiredPrivateKeyNotify(
       [out, retval] VARIANT_BOOL       *use);

// Опис мовою MIDL:
VARIANT_BOOL ExpiredPrivateKeyNotify();
```

3.6 Функції бібліотеки

Опис методів з інтерфейсу об'єкта надається у вигляді IDL-опису, а також у вигляді опису за допомогою мови MIDL.

```
// Інтерфейс криптографічної бібліотеки
[
       object,
       uuid(D0551CE3-42E8-491A-B785-00658F143B6C),
       dual,
       nonextensible,
       pointer_default(unique)
]
```

3.6.1 Функції загального призначення

– Initialize

```
// Метод ініціалізації криптографічної бібліотеки
// caType                   -      вхідний параметр. Містить інформацію про
//                                 символьний ідентифікатор, привласнений АЦСК,
//                                 з шлюзом якого буде працювати бібліотека.
//                                 Не важливий, якщо не додаються криптозаголовки
// initialized              -      параметр, що повертається. Містить інформацію про
//                                 признак ініціалізації бібліотеки

// Опис в файлі IDL:
[id(1)]
HRESULT Initialize(
       [in] BSTR                        caType,
       [out, retval] VARIANT_BOOL       *initialized);

// Опис мовою MIDL:
VARIANT_BOOL Initialize(
       BSTR                             caType);
```

– Finalize

```
// Метод завершення роботи з криптографічною бібліотекою

// Опис в файлі IDL:
[id(27)]
HRESULT Finalize();

// Опис мовою MIDL:
void Finalize();
```

– Close

```
// Метод знищення об`єкта бібліотеки. Визиває метод Release() COM-об'єкту

// Опис в файлі IDL:
[id(28)]
HRESULT Close();

// Опис мовою MIDL:
void Close();
```

– IsLibraryInitialized

```
// Метод отримання інформації про стан криптографічної бібліотеки
// initialized            -      параметр, що повертається. Містить інформацію про
//                               признак ініціалізації бібліотеки

// Опис в файлі IDL:
[id(2)]
HRESULT IsLibraryInitialized(
      [out, retval] VARIANT_BOOL       *initialized);

// Опис мовою MIDL:
VARIANT_BOOL IsLibraryInitialized();
```

– GetLastErrorDescription

```
// Метод отримання інформації про останню помилку
// errorDescription       -      параметр, що повертається. Містить інформацію про
//                               останню помилку

// Опис в файлі IDL:
[id(14)]
HRESULT GetLastErrorDescription(
      [out, retval] BSTR               *errorDescription);

// Опис мовою MIDL:
BSTR GetLastErrorDescription();
```

3.6.2 Функції отримання та встановлення параметрів роботи бібліотеки

– SetSettings

```
// Метод встановлення налаштувань криптографічної бібліотеки за допомогою
// графічного інтерфейсу бібліотеки

// Опис в файлі IDL:
[id(3)]
HRESULT SetSettings();

// Опис мовою MIDL:
void SetSettings();
```

– SetUIMode

```
// Метод встановлення графічного режиму бібліотеки:
// Якщо uiMode = TRUE:
// - особистий ключ, буде зчитуватися в кожній функції, що потребує особистого ключа
//   використовуючи графічний інтерфейс бібліотеки, в іншому випадку необхідно
//   попередньо встановити особисті ключі за допомоги функцій SetPrivateKey,
//   SetPrivateKeyFile;
// - будуть відображатися діалоги з помилками та поточною операцією
// Значення цього параметра не впливає на функції: SetSettings, ViewCerts, ViewCRLs,
// SelectServerCert
// uiMode                 -      параметр, що передається. Містить інформацію про
//                               признак використання графічного режиму роботи
//                               бібліотеки
// result                 -      параметр, що повертається. Містить інформацію про
//                               признак успішності завершення

// Опис в файлі IDL:
[id(15)]
HRESULT SetUIMode(
      [in] VARIANT_BOOL                uiMode,
      [out, retval] VARIANT_BOOL       *result);

// Опис мовою MIDL:
VARIANT_BOOL SetUIMode(
```

```
      VARIANT_BOOL                    uiMode);
```

– SetFilesOptions

```
// Метод встановлення необхідності використання транспортного та криптозаголовків
// Розповсюджується на всі функції. Якщо встановлений в FALSE ігнорує параметри
// використання транспортного та криптозаголовків, що передаються в функцію
// useHeaders              -      параметр, що передається. Містить інформацію про
//                                признак необхідності використання транспортного та
//                                криптозаголовків

// Опис в файлі IDL:
[id(24)]
HRESULT SetFilesOptions(
      [in] VARIANT_BOOL               useHeaders);

// Опис мовою MIDL:
void SetFilesOptions(
      VARIANT_BOOL                    useHeaders);
```

3.6.3 Функції роботи з сховищем сертифікатів та СВС

– ViewCerts

```
// Метод перегляду сертифікатів, що знаходяться в файловому сховищі, за допомогою
// графічного інтерфейсу бібліотеки

// Опис в файлі IDL:
[id(4)]
HRESULT ViewCerts();

// Опис мовою MIDL:
void ViewCerts();
```

– ViewCRLs

```
// Метод перегляду СВС, що знаходяться в файловому сховищі, за допомогою графічного
// інтерфейсу бібліотеки

// Опис в файлі IDL:
[id(5)]
HRESULT ViewCRLs();

// Опис мовою MIDL:
void ViewCRLs();
```

– SelectServerCert

```
// Метод отримання інформації про сертифікат сервера за допомогою графічного
// інтерфейсу бібліотеки. Перевіряє відповідність обраного сертифікату сервера
// сертифікату зчитаного ключа для направленого шифрування
// serverCertID            -      параметр, що повертається. Містить інформацію про
//                                ідентифікатор сертифікату сервера

// Опис в файлі IDL:
[id(6)]
HRESULT SelectServerCert(
      [out, retval] BSTR              *serverCertID);

// Опис мовою MIDL:
BSTR SelectServerCert();
```

– FindServerCert

```
// Метод пошуку сертифікату сервера за кодом ЄДРПУ
// serverCertEDRPOUCode    -      параметр, що передається. Містить інформацію про
//                                код ЄДРПОУ сертифікату сервера
// digitalStamp            -      параметр, що передається. Містить інформацію про
```

```
//                              признак, що сертифікат сервера має бути
//                              електронною печаткою
// serverCertID           -     параметр, що повертається. Містить інформацію про
//                              ідентифікатор сертифікату сервера

// Опис в файлі IDL:
[id(7)]
HRESULT FindServerCert(
       [in] BSTR                        serverCertEDRPOUCode,
       [in] VARIANT_BOOL                digitalStamp,
       [out, retval] BSTR               *serverCertID);

// Опис мовою MIDL:
BSTR FindServerCert(
       BSTR                             serverCertEDRPOUCode,
       VARIANT_BOOL                     digitalStamp);
```

3.6.4 Функції роботи з особистим ключем та носієм ключової інформації

Для здійснення операцій захисту податкової звітності передбачено використання особистих ключів:
- бухгалтера (здійснюється підпис файлу зі звітом), є не обов'язковим, якщо посада на підприємстві відсутня;
- директора (здійснюється підпис файлу зі звітом, його направлене шифрування та підпис отриманого шифрованого файлу), направлене шифрування та підпис отриманого шифрованого повідомлення здійснюється у випадку коли підпис директора використовується в якості цифрової печатки;
- цифрова печатка (здійснюється підпис, його направлене шифрування та підпис отриманого шифрованого файлу), не використовується, коли особистий ключ директора використовується в якості цифрової печатки;

Можливі типи особистого ключа (keyType) описуються в розділі Константи (EUKeyType).

Необхідні ключі для роботи повинні бути встановлені до виклику функцій, які їх використовують, якщо встановлено режим роботи бібліотеки без використання графічного режиму (SetUIMode(FALSE)).

У випадку, коли особистий ключ директора використовується як електронна печатка, необхідно при встановленні ключа директора встановити useDirectorAsDigitalStamp = TRUE. При цьому встановлювати особистий ключ цифрової печатки непотрібно.

– SetPrivateKey

```
// Метод встановлення особистого ключа для роботи бібліотеки без використання
// графічного режиму SetUIMode(FALSE)
// keyType                -     параметр, що передається. Містить інформацію про
//                              тип особистого ключа
// privKey                -     параметр, що передається. Містить інформацію про
//                              особистий ключ
// privKeyPassword        -     параметр, що передається. Містить інформацію про
//                              пароль доступу до особистого ключа
// useDirectorAsDigitalStamp -  параметр, що передається. Містить інформацію
//                              про признак використання особистого ключа
//                              (використовується тільки у випадку
//                              keyType = euKeyTypeDirector)
// result                 -     параметр, що повертається. Містить інформацію про
//                              признак успішності завершення

// Опис в файлі IDL:
HRESULT SetPrivateKey(
       [in] EUKeyType                   keyType,
       [in] VARIANT                     privKey,
       [in] BSTR                        privKeyPassword,
       [in] VARIANT_BOOL                useDirectorAsDigitalStamp,
       [out, retval] VARIANT_BOOL       *result);

// Опис мовою MIDL:
VARIANT_BOOL SetPrivateKey(
       EUKeyType                        keyType,
       VARIANT                          privKey,
       BSTR                             privKeyPassword,
       VARIANT_BOOL                     useDirectorAsDigitalStamp);
```

– SetPrivateKeyFile

```
// Метод встановлення особистого ключа з файлу для роботи бібліотеки без використання
// графічного режиму SetUIMode(FALSE)
// keyType                  -      параметр, що передається. Містить інформацію про
//                                 тип особистого ключа
// privKeyFileName          -      параметр, що передається. Містить інформацію про
//                                 ім`я файла з особистим ключем
// privKeyPassword          -      параметр, що передається. Містить інформацію про
//                                 пароль доступу до особистого ключа
// useDirectorAsDigitalStamp -     параметр, що передається. Містить інформацію
//                                 про признак використання особистого ключа
//                                 (використовується тільки у випадку
//                                 keyType = euKeyTypeDirector)
// result                   -      параметр, що повертається. Містить інформацію про
//                                 признак успішності завершення

// Опис в файлі IDL:
[id(26)]
HRESULT SetPrivateKeyFile(
       [in] EUKeyType                   keyType,
       [in] BSTR                        privKeyFileName,
       [in] BSTR                        privKeyPassword,
       [in] VARIANT_BOOL                useDirectorAsDigitalStamp,
       [out, retval] VARIANT_BOOL       *result);

// Опис мовою MIDL:
VARIANT_BOOL SetPrivateKeyFile(
       EUKeyType                        keyType,
       BSTR                             privKeyFileName,
       BSTR                             privKeyPassword,
       VARIANT_BOOL                     useDirectorAsDigitalStamp);
```

– ResetPrivateKey

```
// Метод знищення в пам`яті особистого ключа при роботі бібліотеки без використання
// графічного режиму SetUIMode(FALSE).
// keyType                  -      параметр, що передається. Містить інформацію про
//                                 тип особистого ключа
// result                   -      параметр, що повертається. Містить інформацію про
//                                 признак успішності завершення

// Опис в файлі IDL:
[id(17)]
HRESULT ResetPrivateKey(
       [in] EUKeyType                   keyType,
       [out, retval] VARIANT_BOOL       *result);

// Опис мовою MIDL:
VARIANT_BOOL ResetPrivateKey(
       EUKeyType                        keyType);
```

– GetPrivateKeyInfo

```
// Метод отримання інформації про власника особистого ключа. Якщо особистий ключ
// вказаного типу не зчитаний, метод GetInitiatorsCount об'єкта типу IResultInfo
// поверне 0, або 1 - в іншому випадку.
// keyType                  -      параметр, що передається. Містить інформацію про
//                                 тип особистого ключа
// result                   -      параметр, що повертається. Містить об'єкт
//                                 типу IResultInfo з результатом виконання операції

// Опис в файлі IDL:
[id(29)]
HRESULT GetPrivateKeyInfo(
       [in] EUKeyType                   keyType,
       [out, retval] VARIANT            *result);
```

Пор. № зміни	Підпис відпов. особи	Дата внесення

```
// Опис мовою MIDL:
VARIANT GetPrivateKeyInfo (
       EUKeyType                        keyType);
```

3.6.4 Функції ЕЦП

Функції ЕЦП при підписі файлу або даних з файлу додають до підписаних даних криптографічний заголовок, у випадку якщо функцією SetFilesOptions не встановлено значення FALSE або не вказано явно в параметрах функції. Якщо встановлено режим роботи бібліотеки без використання графічного режиму необхідні особисті ключі повинні бути попередньо встановлені функціями SetPrivateKey або SetPrivateKeyFile, в іншому випадку функцією буде повернено помилку.

3.6.4.1 Підпис файлів

– SignFilesByAccountant

```
// Метод підпису файлів за допомогою особистого ключа бухгалтера
// fileNames               -      параметр, що передається. Містить інформацію про
//                                імена файлів для підпису у вигляді масиву строк, або
//                                однієї строки у випадку, якщо підписується один файл
// results                 -      параметр, що повертається. Містить масив об'єктів
//                                типу IResultInfo з результатом виконання операції

// Опис в файлі IDL:
[id(8)]
HRESULT SignFilesByAccountant(
       [in] VARIANT                     fileNames,
       [out, retval] VARIANT            *results);

// Опис мовою MIDL:
VARIANT SignFilesByAccountant(
       VARIANT                          fileNames);
```

– SignFilesByDirector

```
// Метод підпису файлів за допомогою особистого ключа директора
// fileNames               -      параметр, що передається. Містить інформацію про
//                                імена файлів для підпису у вигляді масиву строк, або
//                                однієї строки у випадку, якщо підписується один файл
// results                 -      параметр, що повертається. Містить масив об'єктів
//                                типу IResultInfo з результатом виконання операції

// Опис в файлі IDL:
[id(9)]
HRESULT SignFilesByDirector(
       [in] VARIANT                     fileNames,
       [out, retval] VARIANT            *results);

// Опис мовою MIDL:
VARIANT SignFilesByDirector(
       VARIANT                          fileNames);
```

– VerifyFiles

```
// Метод перевірки файлів
// fileNames               -      параметр, що передається. Містить інформацію про
//                                імена файлів для перевірки у вигляді масиву строк, або
//                                однієї строки у випадку, якщо пеервіряється один файл
// results                 -      параметр, що повертається. Містить масив об'єктів
//                                типу IResultInfo з результатом виконання операції

// Опис в файлі IDL:
[id(12)]
HRESULT VerifyFiles(
       [in] VARIANT                     fileNames,
       [out, retval] VARIANT            *results);
```

```
// Опис мовою MIDL:
VARIANT VerifyFiles(
       VARIANT                          fileNames);
```

3.6.4.2 Підпис даних

– SignFilesContentsByAccountant

```
// Метод підпису даних з файлів за допомогою особистого ключа бухгалтера
// filesContents          -      параметр, що передається. Містить інформацію про
//                               дані з файлів для підпису у вигляді масиву байтових
//                               масивів або байтовому масиві у випадку, якщо
//                               підписуються дані з одного файлу
// results                -      параметр, що повертається. Містить масив об'єктів
//                               типу IResultInfo з результатом виконання операції

// Опис в файлі IDL:
[id(18)]
HRESULT SignFilesContentsByAccountant(
       [in] VARIANT                     filesContents,
       [out, retval] VARIANT            *results);

// Опис мовою MIDL:
VARIANT SignFilesContentsByAccountant(
       VARIANT                          filesContents);
```

– SignFilesContentsByDirector

```
// Метод підпису даних з файлів за допомогою особистого ключа директора
// filesContents          -      параметр, що передається. Містить інформацію про
//                               дані з файлів для підпису у вигляді масиву байтових
//                               масивів або байтовому масиві у випадку, якщо
//                               підписуються дані з одного файлу
// results                -      параметр, що повертається. Містить масив об'єктів
//                               типу IResultInfo з результатом виконання операції

// Опис в файлі IDL:
[id(19)]
HRESULT SignFilesContentsByDirector(
       [in] VARIANT                     filesContents,
       [out, retval] VARIANT            *results);

// Опис мовою MIDL:
VARIANT SignFilesContentsByDirector(
       VARIANT                          filesContents);
```

– VerifyFilesContents

```
// Метод перевірки даних з файлів
// filesContents          -      параметр, що передається. Містить інформацію про
//                               дані з файлів для підпису у вигляді масиву байтових
//                               масивів або байтовому масиві у випадку, якщо
//                               перевіряються дані з одного файлу
// results                -      параметр, що повертається. Містить масив об'єктів
//                               типу IResultInfo з результатом виконання операції

// Опис в файлі IDL:
[id(22)]
HRESULT VerifyFilesContents(
       [in] VARIANT                     filesContents,
       [out, retval] VARIANT            *results);

// Опис мовою MIDL:
VARIANT VerifyFilesContents (
       VARIANT                          filesContents);
```

3.6.4 Функції захисту

Функції захисту при захисті файлу або даних з файлу додають до захищених даних криптографічні заголовки та транспортний заголовок, у випадку якщо функцією SetFilesOptions не встановлено значення FALSE або не вказано явно в параметрах функції. Якщо встановлено режим роботи бібліотеки без використання графічного режиму необхідні особисті ключі повинні бути попередньо встановлені функціями SetPrivateKey або SetPrivateKeyFile, в іншому випадку функцією буде повернено помилку.

3.6.4.1 Захист файлів

– ProtectFilesByDigitalStamp

```
// Метод захисту файлів за допомогою особистого ключа цифрової печатки. Захист файлів
// передбачає виконання наступних криптографічних операцій (у порядку їх виконання):
// 1) підпис за допомогою ключа цифрової печатки (якщо
//    useDirectorAsDigitalStamp = TRUE, операція не виконується);
// 2) направлене шифрування з використанням сертифіката сервера одержувача, та
//    особистого ключа цифрової печатки (якщо useDirectorAsDigitalStamp = FALSE) або
//    директора (якщо useDirectorAsDigitalStamp = TRUE)
// 3) підпис зашифрованих даних за допомогою ключа цифрової печатки (якщо
//    useDirectorAsDigitalStamp = FALSE) або директора (якщо
//    useDirectorAsDigitalStamp = TRUE)
// fileNames               -      параметр, що передається. Містить інформацію про
//                                імена файлів для захисту у вигляді масиву строк, або
//                                однієї строки у випадку, якщо захищається один файл
// useDirectorAsDigitalStamp -    параметр, що передається. Містить інформацію
//                                про признак використання особистого ключа директора
//                                в якості цифрової печатки
// emailAddress            -      параметр, що передається. Містить інформацію
//                                про адресу електронної пошти відправника.
// serverCertID            -      параметр, що передається. Містить інформацію про
//                                ідентифікатор сертифікату сервера
// results                 -      параметр, що повертається. Містить масив об'єктів
//                                типу IResultInfo з результатом виконання операції

// Опис в файлі IDL:
[id(10)]
HRESULT ProtectFilesByDigitalStamp(
       [in] VARIANT                    fileNames,
       [in] VARIANT_BOOL               useDirectorAsDigitalStamp,
       [in] BSTR                       emailAddress,
       [in] BSTR                       serverCertID,
       [out, retval] VARIANT           *results);

// Опис мовою MIDL:
VARIANT ProtectFilesByDigitalStamp(
       VARIANT                         fileNames,
       VARIANT_BOOL                    useDirectorAsDigitalStamp,
       BSTR                            emailAddress,
       BSTR                            serverCertID);
```

– ProtectFiles

```
// Метод захисту файлів. Захист файлів передбачає виконання наступних криптографічних
// операцій (у порядку їх виконання):
// 1) підпис за допомогою ключа бухгалтера(якщо useAccountant = TRUE);
// 2) підпис за допомогою ключа директора (завжди);
// 3) підпис за допомогою ключа цифрової печатки (якщо
//    useDirectorAsDigitalStamp = FALSE);
// 4) направлене шифрування з використанням сертифіката сервера одержувача, та
//    особистого ключа цифрової печатки (якщо useDirectorAsDigitalStamp = FALSE) або
//    директора (якщо useDirectorAsDigitalStamp = TRUE)
// 5) підпис зашифрованих даних за допомогою ключа цифрової печатки (якщо
//    useDirectorAsDigitalStamp = FALSE) або директора (якщо
//    useDirectorAsDigitalStamp = TRUE)
// fileNames               -      параметр, що передається. Містить інформацію про
//                                імена файлів для захисту у вигляді масиву строк або
//                                однієї строки у випадку, якщо захищається один файл
```

```
// useAccountant           -      параметр, що передається. Містить інформацію про
//                                признак необхідності підпису бухгалтером
// useDirectorAsDigitalStamp -    параметр, що передається. Містить інформацію
//                                про признак використання особистого ключа
//                                директора в якості цифрової печатки
// emailAddress            -      параметр, що передається. Містить інформацію
//                                про адресу електронної пошти відправника.
// serverCertID            -      параметр, що передається. Містить інформацію про
//                                ідентифікатор сертифікату сервера
// results                 -      параметр, що повертається. Містить масив
//                                об'єктів типу IResultInfo з результатом
//                                виконання операції

// Опис в файлі IDL:
[id(11)]
HRESULT ProtectFiles(
       [in] VARIANT                    fileNames,
       [in] VARIANT_BOOL               useAccountant,
       [in] VARIANT_BOOL               useDirectorAsDigitalStamp,
       [in] BSTR                       emailAddress,
       [in] BSTR                       serverCertID,
       [out, retval] VARIANT           *results);

// Опис мовою MIDL:
VARIANT ProtectFiles (
       VARIANT                         fileNames,
       VARIANT_BOOL                    useAccountant,
       VARIANT_BOOL                    useDirectorAsDigitalStamp,
       BSTR                            emailAddress,
       BSTR                            serverCertID);
```

– ProtectFilesEx

```
// Метод захисту файлів з додатковими параметрами. Захист файлів передбачає виконання
// наступних криптографічних операцій (у порядку їх виконання):
// 1) підпис за допомогою ключа бухгалтера(якщо useAccountant = TRUE);
// 2) підпис за допомогою ключа директора(якщо useDirector = TRUE);
// 3) підпис за допомогою ключа цифрової печатки (якщо useDigitalStamp = TRUE);
// 4) направлене шифрування з використанням сертифіката сервера одержувача, та
//    особистого ключа цифрової печатки (якщо useDigitalStamp = TRUE, значення
//    useDirector в цьому випадку ігнорується) або директора
// 5) підпис зашифрованих даних за допомогою ключа цифрової печатки (якщо
//    useDigitalStamp = TRUE, значення useDirector в цьому випадку ігнорується) або
//    директора
// Якщо функцією SetFilesOptions встановлено не використовувати криптозаголовки та
// транспортний заголовок, значення параметрів appendCryptoHeaders та
// appendTransportHeader буде проігноровано
// fileNames               -      параметр, що передається. Містить інформацію про
//                                імена файлів для захисту у вигляді масиву строк, або
//                                однієї строки у випадку, якщо захищається один файл
// useAccountant           -      параметр, що передається. Містить інформацію про
//                                признак необхідності підпису бухгалтером
// useDirector             -      параметр, що передається. Містить інформацію про
//                                признак необхідності підпису директором або
//                                підпису/шифрування/підпису, якщо цифрова печатка
//                                не використовується (useDigitalStamp = FALSE)
// useDigitalStamp         -      параметр, що передається. Містить інформацію про
//                                признак необхідності підпису/шифрування/підпису
//                                за допомоги електронної печатки
// appendCryptoHeaders     -      параметр, що передається. Містить інформацію про
//                                признак необхідності додавати криптозаголовки до
//                                після криптографічних операцій
// appendTransportHeader -        параметр, що передається. Містить інформацію про
//                                признак необхідності додавати транспортний
//                                заголовок до захищених даних
// emailAddress            -      параметр, що передається. Містить інформацію
//                                про адресу електронної пошти відправника
// serverCertIDs           -      параметр, що передається. Містить інформацію про
//                                ідентифікатори сертифікатів серверів у вигляді
```

Пор. № зміни	Підпис відпов. особи	Дата внесення

```
//                                масиву строк або строки у випадку, якщо
//                                використовується лише один сервер
// results                 -      параметр, що повертається. Містить масив
//                                об'єктів типу IResultInfo з результатом
//                                виконання операції

// Опис в файлі IDL:
[id(25)]
HRESULT ProtectFilesEx(
       [in] VARIANT                    fileNames,
       [in] VARIANT_BOOL               useAccountant,
       [in] VARIANT_BOOL               useDirector,
       [in] VARIANT_BOOL               useDigitalStamp,
       [in] VARIANT_BOOL               appendCryptoHeaders,
       [in] VARIANT_BOOL               appendTransportHeader,
       [in] BSTR                       emailAddress,
       [in] VARIANT                    serverCertIDs,
       [out, retval] VARIANT           *results);

// Опис мовою MIDL:
VARIANT ProtectFilesEx(
       VARIANT                         fileNames,
       VARIANT_BOOL                    useAccountant,
       VARIANT_BOOL                    useDirector,
       VARIANT_BOOL                    useDigitalStamp,
       VARIANT_BOOL                    appendCryptoHeaders,
       VARIANT_BOOL                    appendTransportHeader,
       BSTR                            emailAddress,
       VARIANT                         serverCertIDs);
```

– UnprotectFiles

```
// Метод зняття захисту з файлів, отриманих від серверу. Якщо при взаємодії з сервером
// в якості особистого ключа цифрової печатки використовувався ключ директора,
// необхідно параметр useDirectorAsDigitalStamp встановити як TRUE
// fileNames               -      параметр, що передається. Містить інформацію про
//                                імена файлів для зняття захисту у вигляді масиву
//                                строк, або однієї строки у випадку, якщо захист
//                                знімається з одного файлу
// useDirectorAsDigitalStamp -    параметр, що передається. Містить інформацію
//                                про признак використання особистого ключа
//                                директора в якості цифрової печатки
// results                 -      параметр, що повертається. Містить масив
//                                об'єктів типу IResultInfo з результатом
//                                виконання операції

// Опис в файлі IDL:
[id(13)]
HRESULT UnprotectFiles(
       [in] VARIANT                    fileNames,
       [in] VARIANT_BOOL               useDirectorAsDigitalStamp,
       [out, retval] VARIANT           *results);

// Опис мовою MIDL:
VARIANT UnprotectFiles(
       VARIANT                         fileNames,
       VARIANT_BOOL                    useDirectorAsDigitalStamp);
```

3.6.4.2 Захист даних

– ProtectFilesContentsByDigitalStamp

```
// Метод захисту даних з файлів за допомогою особистого ключа цифрової печатки. Захист
// даних з файлів передбачає виконання наступних криптографічних операцій (у порядку їх
// виконання):
// 1) підпис за допомогою ключа цифрової печатки (якщо
//    useDirectorAsDigitalStamp = TRUE, операція не виконується);
// 2) направлене шифрування з використанням сертифіката сервера одержувача, та
//    особистого ключа цифрової печатки (якщо useDirectorAsDigitalStamp = FALSE) або
```

```
//    директора (якщо useDirectorAsDigitalStamp = TRUE)
// 3) підпис зашифрованих даних за допомогою ключа цифрової печатки (якщо
//    useDirectorAsDigitalStamp = FALSE) або директора (якщо
//    useDirectorAsDigitalStamp = TRUE)
// fileNames               -      параметр, що передається. Містить інформацію про
//                                імена файлів для додавання їх в криптозаголовки у
//                                вигляді масиву строк, або однієї строки у випадку,
//                                якщо захищається один файл
// filesContents           -      параметр, що передається. Містить інформацію про
//                                дані з файлів для підпису у вигляді масиву
//                                байтових масивів, або байтовому масиві у випадку, якщо
//                                захищаються дані з одного файлу
// useDirectorAsDigitalStamp -    параметр, що передається. Містить інформацію
//                                про признак використання особистого ключа
//                                директора в якості цифрової печатки
// emailAddress            -      параметр, що передається. Містить інформацію
//                                про адресу електронної пошти відправника.
// serverCertID            -      параметр, що передається. Містить інформацію про
//                                ідентифікатор сертифікату сервера
// results                 -      параметр, що повертається. Містить масив
//                                об'єктів типу IResultInfo з результатом
//                                виконання операції

// Опис в файлі IDL:
[id(20)]
HRESULT ProtectFilesContentsByDigitalStamp(
       [in] VARIANT                    fileNames,
       [in] VARIANT                    filesContents,
       [in] VARIANT_BOOL               useDirectorAsDigitalStamp,
       [in] BSTR                       emailAddress,
       [in] BSTR                       serverCertID,
       [out, retval] VARIANT           *results);

// Опис мовою MIDL:
VARIANT ProtectFilesContentsByDigitalStamp(
       VARIANT                         fileNames,
       VARIANT                         filesContents,
       VARIANT_BOOL                    useDirectorAsDigitalStamp,
       BSTR                            emailAddress,
       BSTR                            serverCertID);
```

– ProtectFilesContents

```
// Метод захисту даних з файлів. Захист файлів передбачає виконання наступних
// криптографічних операцій (у порядку їх виконання):
// 1) підпис за допомогою ключа бухгалтера(якщо useAccountant = TRUE);
// 2) підпис за допомогою ключа директора (завжди);
// 3) підпис за допомогою ключа цифрової печатки (якщо
//    useDirectorAsDigitalStamp = TRUE, операція не виконується);
// 4) направлене шифрування з використанням сертифіката сервера одержувача, та
//    особистого ключа цифрової печатки (якщо useDirectorAsDigitalStamp = FALSE) або
//    директора (якщо useDirectorAsDigitalStamp = TRUE)
// 5) підпис зашифрованих даних за допомогою ключа цифрової печатки (якщо
//    useDirectorAsDigitalStamp = FALSE) або директора (якщо
//    useDirectorAsDigitalStamp = TRUE)
// fileNames               -      параметр, що передається. Містить інформацію про
//                                імена файлів для додавання їх в криптозаголовки у
//                                вигляді масиву строк, або однієї строки у випадку,
//                                якщо захищається один файл
// filesContents           -      параметр, що передається. Містить інформацію про
//                                дані з файлів для підпису у вигляді масиву
//                                байтових масивів, або байтовому масиві у випадку, якщо
//                                захищаються дані з одного файлу
// useAccountant           -      параметр, що передається. Містить інформацію про
//                                признак необхідності підпису бухгалтером
// useDirectorAsDigitalStamp -     параметр, що передається. Містить інформацію
//                                про признак використання особистого ключа
//                                директора в якості цифрової печатки
// emailAddress            -      параметр, що передається. Містить інформацію
```

Пор. № зміни	Підпис відпов. особи	Дата внесення

```
//                                  про адресу електронної пошти відправника.
// serverCertID             -       параметр, що передається. Містить інформацію про
//                                  ідентифікатор сертифікату сервера
// results                  -       параметр, що повертається. Містить масив
//                                  об'єктів типу IResultInfo з результатом
//                                  виконання операції

// Опис в файлі IDL:
[id(21)]
HRESULT ProtectFilesContents(
       [in] VARIANT                     fileNames,
       [in] VARIANT                     filesContents,
       [in] VARIANT_BOOL                useAccountant,
       [in] VARIANT_BOOL                useDirectorAsDigitalStamp,
       [in] BSTR                        emailAddress,
       [in] BSTR                        serverCertID,
       [out, retval] VARIANT            *results);

// Опис мовою MIDL:
VARIANT ProtectFilesContents(
       VARIANT                          fileNames,
       VARIANT                          filesContents,
       VARIANT_BOOL                     useAccountant,
       VARIANT_BOOL                     useDirectorAsDigitalStamp,
       BSTR                             emailAddress,
       BSTR                             serverCertID);
```

– UnprotectFilesContents

```
// Метод зняття захисту даних з файлу, отриманих від серверу. Якщо при взаємодії з
// сервером в якості особистого ключа цифрової печатки використовувався ключ директора,
// необхідно параметр useDirectorAsDigitalStamp встановити як TRUE
// protectedFilesContents -         параметр, що передається. Містить інформацію про
//                                  дані з файлів для зняття захисту у вигляді
//                                  масиву байтових масивів
// useDirectorAsDigitalStamp -      параметр, що передається. Містить інформацію
//                                  про признак використання особистого ключа
//                                  директора в якості цифрової печатки
// results                  -       параметр, що повертається. Містить масив
//                                  об'єктів типу IResultInfo з результатом
//                                  виконання операції

// Опис в файлі IDL:
[id(23)]
HRESULT UnprotectFilesContents(
       [in] VARIANT                     protectedFilesContents,
       [in] VARIANT_BOOL                useDirectorAsDigitalStamp,
       [out, retval] VARIANT            *results);

// Опис мовою MIDL:
VARIANT UnprotectFilesContents(
       VARIANT                          protectedFilesContents,
       VARIANT_BOOL                     useDirectorAsDigitalStamp);
```

Пор. № зміни	Підпис відпов. особи	Дата внесення

ДОДАТОК А

ВСТАНОВЛЕННЯ ПАРАМЕТРІВ РОБОТИ КРИПТОГРАФІЧНОЇ ПІДСИСТЕМИ

А.1 Файлове сховище

Для настроювання параметрів файлового сховища сертифікатів та СВС необхідно перейти до закладки “Файлове сховище”. Вікно “Параметри роботи” із сторінкою “Файлове сховище” наведене на рис. А.1. На цій сторінці встановлюються наступні параметри роботи програми:

- “Каталог з сертифікатуми та СВС”. Даний параметр встановлює каталог файлового сховища для зберігання сертифікатів та СВС.
  Всі сертифікати та СВС, що завантажуються не засобами програми повинні записуватися у даний каталог.
- “Автоматично перечитувати файлове сховище при виявлені змін”. Даний параметр визначає необхідність автоматичного перечитування каталогу файлового сховища програмою при внесенні будь-яких змін до цього каталогу (запису нового сертифікату чи СВС у каталог чи видалення файлу з сертифікатом або СВС).
  Якщо параметр не встановлено необхідно виконувати повторне зчитування файлового сховища після внесення змін. Для цього необхідно обрати підпункт “Зчитати сертифікати та СВС” в пункті меню “Сертифікати та СВС” або натиснути клавішу “F9” у головному вікні програми.
- “Зберігати у файлове сховище сертифікати, що отримані з OCSP- чи LDAP-серверів”. Даний параметр визначає необхідність автоматичного збереження сертифікатів, що не знайдені у файловому сховищі, а отримані з OCSP- чи LDAP-серверів у файлове сховище.
- “Час зберігання стану перевіреного сертифікату”. Даний параметр визначає час протягом якого сертифікати що вже перевірені не будуть повторно перевірятися.
  Застосування такого механізму збереження стану сертифікату протягом певного часу забезпечує зменшення ресурсів системи на перевірку сертифікату при частих звертаннях (механізм кешування статусу сертифікату).
- “Використовувати СВС”. Параметр вказує на необхідність використання СВС в якості засобу перевірки статусу сертифікатів відкритих ключів що використовуються.
- “Тільки свого ЦСК”. Даний параметр визначає необхідність використовувати при перевірці сертифікатів СВС лише свого ЦСК у ланцюжку.
  Для цього повинен бути зчитаний особистий ключ користувача, оскільки ЦСК користувача визначається за допомогою параметрів особистого ключа.
- “Повний та частковий”. Даний параметр визначає необхідність перевірки наявності двох діючих СВС (повного та часткового) при здійсненні перевірки сертифікатів.
  Якщо параметр не встановлено достатньо лише одного повного діючого СВС. Даний параметр дозволяє не виконувати постійне завантаження останнього діючого часткового СВС.
- “Завантажувати автоматично”. Даний параметр визначає можливість автоматичного завантаження СВС під час перевірки статусу сертифікатів, якщо у файловому сховищі не знайдено діючих СВС.
  Параметр має сенс якщо у сертифікатух ЦСК, або серверів ЦСК встановлено шлях отримання СВС.

Для збереження внесених змін необхідно натиснути кнопку “Застосувати”.

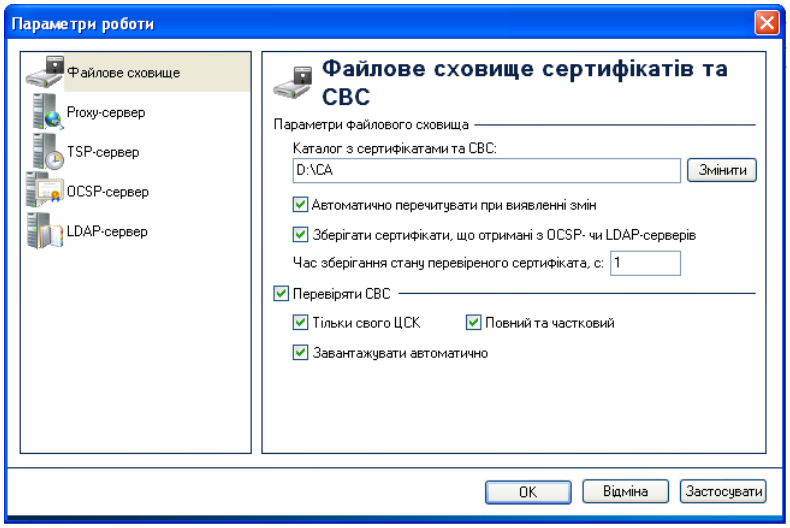

Рисунок А.1

А.2 Proxy-сервер

Для настроювання параметрів proxy-сервера необхідно перейти до закладки “Proxy-сервер” у вікні що наведене на рисунку А.1. Вікно “Параметри роботи” із сторінкою “Proxy-сервер” наведене на рис. А.2. На сторінці “Proxy-сервер” встановлюються наступні параметри роботи програми:

- “Підключатися через proxy-сервер”. Встановлює необхідність використання proxy-сервера під час підключення до серверів обробки запитів.
- “Ім’я чи IP-адреса сервера”. Даний параметр встановлює IP-адресу або DNS-ім’я proxy-сервера.
- “TCP-порт”. Даний параметр встановлює TCP-порт proxy-сервера.
- “Автентифікуватися на proxy-сервері”. Встановлює необхідність автентифікації (вводу логіну та паролю) під час підключення до proxy-сервера.
- “Ім’я користувача”. Даний параметр встановлює ім’я користувача proxy-сервера.
  Якщо proxy-сервер працює в режимі без автентифікації даний параметр може не вводитися.
- “Пароль”. Даний параметр встановлює пароль доступу користувача до proxy-сервера.
  Якщо proxy-сервер працює в режимі без автентифікації даний параметр може не вводитися.
- “Зберегти пароль”. Даний параметр встановлює необхідність зберігати пароль доступу до proxy-сервера у реєстрі ОС.
  У випадку якщо даний параметр не встановлено, введення паролю буде запрошуватися при першому підключенні до proxy-сервера у програмі.

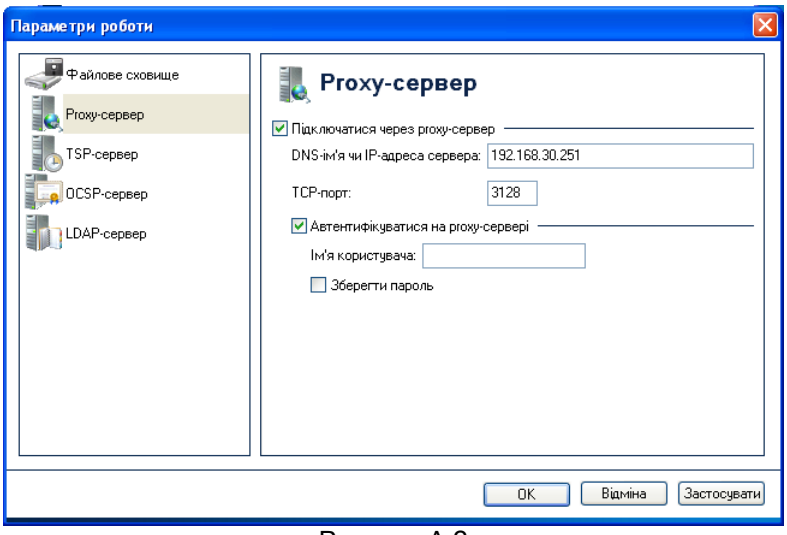

Рисунок А.2

Пор. № зміни	Підпис відпов. особи	Дата внесення

Для збереження внесених змін необхідно натиснути кнопку “Застосувати”.

A.3 TSP-сервер

Для настроювання параметрів TSP-сервера необхідно перейти до закладки “TSP-сервер” у вікні що наведене на рисунку А.1. Вікно “Параметри роботи” із сторінкою “TSP-сервер” наведене на рис. А.3. На сторінці “TSP-сервер” встановлюються наступні параметри роботи програми:

– “Ім’я чи IP-адреса сервера”. Даний параметр встановлює IP-адресу або DNS-ім’я TSP-сервера.
  Як правило це є IP-адреса або DNS-ім’я сервера взаємодії ЦСК.
– “TCP-порт”. Даний параметр встановлює TCP-порт TSP-сервера.
  Як правило це порт протоколу HTTP (80).

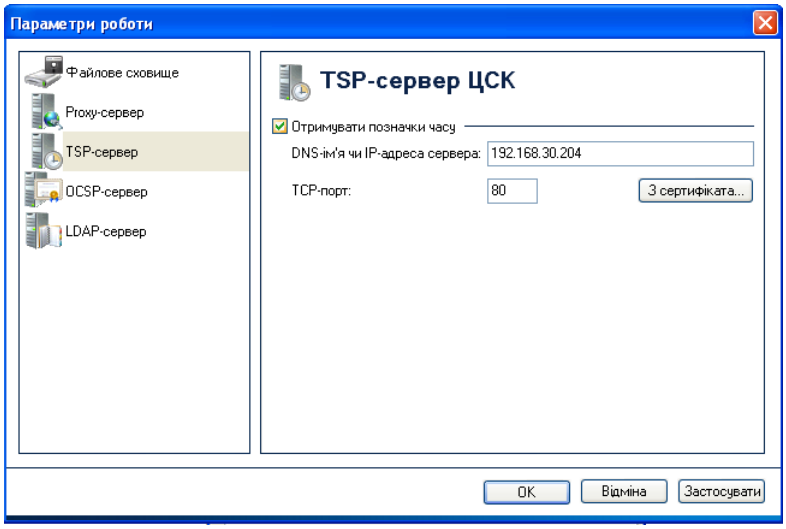

Рисунок А.3

За замовчанням встановлюються параметри TSP-сервера що вказані у відповідному сертифікаті сервера або ЦСК. Параметри TSP-сервера можна також встановити з сертифікату за допомогою кнопки “З сертифікату…”.

Для збереження внесених змін необхідно натиснути кнопку “Застосувати”.

A.4 OCSP-сервер

Для настроювання параметрів OCSP-сервера необхідно перейти до закладки “OCSP-сервер” у вікні що наведене на рисунку А.1. Вікно “Параметри роботи” із сторінкою “OCSP-сервер” наведене на рис. А.4. На сторінці “OCSP-сервер” встановлюються наступні параметри роботи програми:

– “Ім’я чи IP-адреса сервера”. Даний параметр встановлює IP-адресу або DNS-ім’я OCSP-сервера.
  Як правило це є IP-адреса або DNS-ім’я сервера взаємодії ЦСК.
– “TCP-порт”. Даний параметр встановлює TCP-порт OCSP-сервера.
  Як правило це порт протоколу HTTP (80).
– “Перевіряти статус сертифікатів через OCSP до перевірки у файловому сховищі”. Даний параметр встановлює черговість перевірки статусу сертифікату.
  Якщо параметр встановлено, статус сертифікату перевіряється спочатку за допомогою OCSP-протоколу, потім за допомогою файлового сховища.
  Якщо параметр не встановлено, перевірка здійснюється спочатку за допомогою файлового сховища, а потім (за необхідністю) за допомогою OCSP-протоколу.

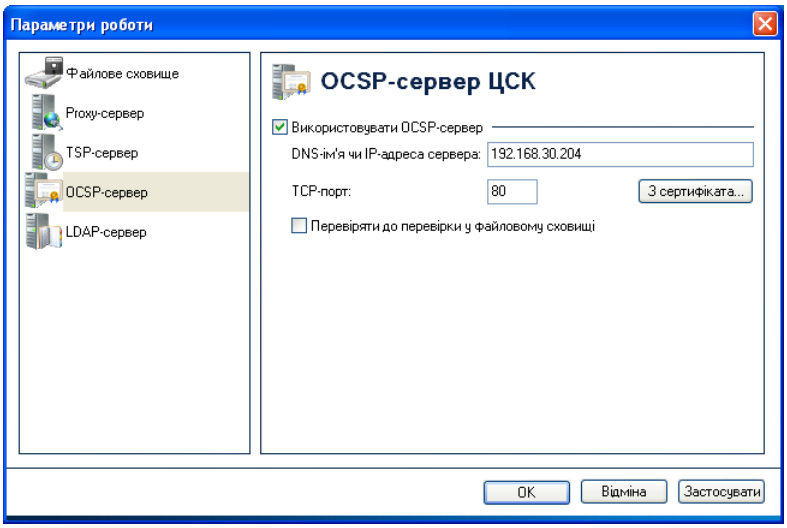

Рисунок А.4

За замовчанням встановлюються параметри OCSP-сервера що вказані у відповідному сертифікаті сервера або ЦСК. Параметри OCSP-сервера можна також встановити з сертифікату за допомогою кнопки “З сертифікату…”.

Для збереження внесених змін необхідно натиснути кнопку “Застосувати”.

А.5 LDAP-сервер

Для настроювання параметрів LDAP-сервера перейти до закладки “LDAP-сервер” у вікні що наведене на рисунку А.1. Вікно “Параметри роботи” із сторінкою “LDAP-сервер” наведене на рис. А.5. На сторінці “LDAP-сервер” встановлюються наступні параметри роботи програми:

- “Ім’я чи IP-адреса сервера”. Даний параметр встановлює IP-адресу або DNS-ім’я LDAP-сервера. Як правило це є IP-адреса або DNS-ім’я сервера взаємодії ЦСК.
- “TCP-порт”. Даний параметр встановлює TCP-порт LDAP-сервера. Як правило це порт протоколу LDAP (389).
- “Анонімний доступ”. Даний параметр встановлює застосування анонімного доступу до LDAP-сервера (без використання імені користувача та паролю).
- “Ім’я користувача”. Даний параметр використовується якщо не встановлено параметр “Анонімний доступ” та встановлює ім’я користувача LDAP-сервера.
- “Пароль доступу”. Даний параметр використовується якщо не встановлено параметр “Анонімний доступ” та встановлює пароль доступу користувача до LDAP-сервера.
- “Шукати сертифікати у LDAP-каталозі”. Даний параметр встановлює необхідність пошуку сертифікатів у LDAP-каталозі, у випадку якщо сертифікат не знайдено у файловому сховищі та за допомогою OCSP-протоколу.

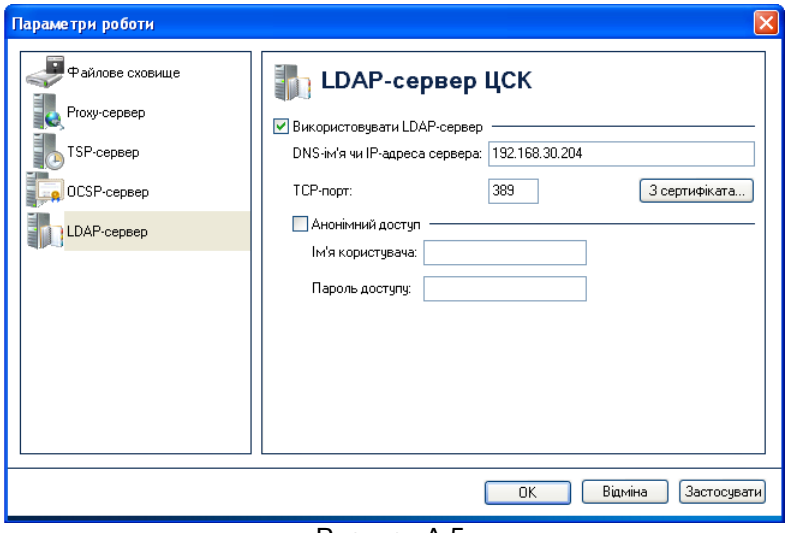

Рисунок А.5

Пор. № зміни	*Підпис відпов. особи*	*Дата внесення*

За замовчанням встановлюються параметри LDAP-сервера що вказані у відповідному сертифікаті сервера або ЦСК. Параметри LDAP-сервера можна також встановити з сертифікату за допомогою кнопки “З сертифікату…”.

Для збереження внесених змін необхідно натиснути кнопку “Застосувати”.

ДОДАТОК Б

ПЕРЕГЛЯД СЕРТИФІКАТІВ ТА СВС

Б.1 Зчитування сертифікатів та СВС

Програма автоматично виконує зчитування сертифікатів та СВС з файлового сховища при першій необхідності після свого запуску. При внесенні змін (запису чи видалення сертифікатів чи СВС) до файлового сховища під час роботи програми, якщо не встановлено параметр “Автоматично перечитувати файлове сховище при виявленні змін” (див п. А.1), необхідно перечитати файлове сховище. Для цього необхідно обрати підпункт “Зчитати сертифікати та СВС” в пункті меню “Сертифікати та СВС” або натиснути клавішу F9.

Б.2 Перегляд сертифікатів

Вікно із сертифікатуми наведене на рис. Б.1.

За допомогою даного вікна можна видаляти сертифікати з файлового сховища, перевіряти та переглядати сертифікати.

Сертифікати у вікні відсортовані за типами власників (тип власника обирається у верхній частині вікна у випадаючому списку):

- всі сертифікати;
- сертифікати центрів сертифікації ключів;
- сертифікати серверів ЦСК;
- сертифікати CMP-серверів;
- сертифікати TSP-Серверів
- сертифікати OCSP-Серверів
- сертифікати користувачів.

Для перегляду списку сертифікатів власника певного типу необхідно обрати відповідний тип власника у верхній частині вікна у списку що випадає.

Для перегляду сертифікату необхідно натиснути на відповідному записі про сертифікат у списку. Сертифікат буде відображено у вікні що наведене на рисунках Б.2 та Б.3.

Для видалення сертифікатів з файлового сховища необхідно виділити у списку відповідні записи про сертифікати та натиснути кнопку “Видалити”.

Для перевірки сертифікату необхідно виділити відповідний запис про сертифікат у списку та натиснути кнопку “Перевірити”. Перевірка сертифікату здійснюється відповідно до встановлених параметрів роботи (див п. А.1) - за допомогою СВС, OCSP-протоколу тощо. Результатом перевірки буде вікно що наведене на рис. Б.4. Якщо у цьому вікні натиснути “Сертифікат”, сертифікат буде відображений у вікні детального перегляду (рис. Б.3).

Для імпорту сертифікату до файлового сховища необхідно натиснути “Імпортувати”, та обрати потрібний сертифікат на будь-якому носії інформації.

Для експорту сертифікату з файлового сховища в інше місце (носій інформації тощо), необхідно натиснути “Експортувати”, та обрати інше місце розташування.

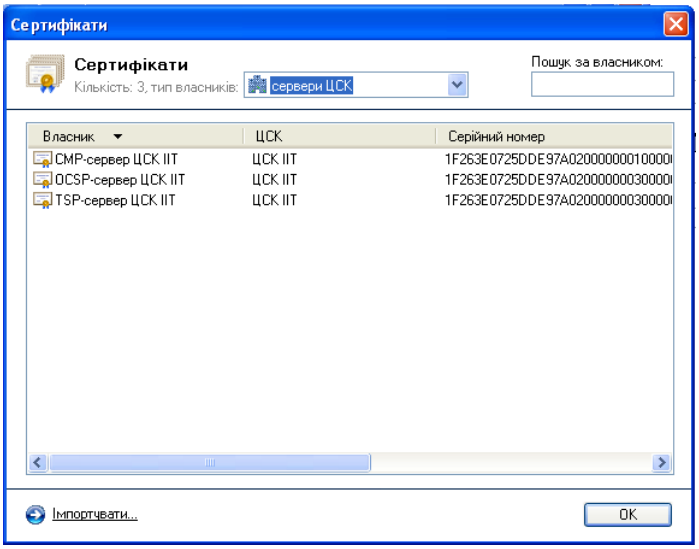

Рисунок Б.1

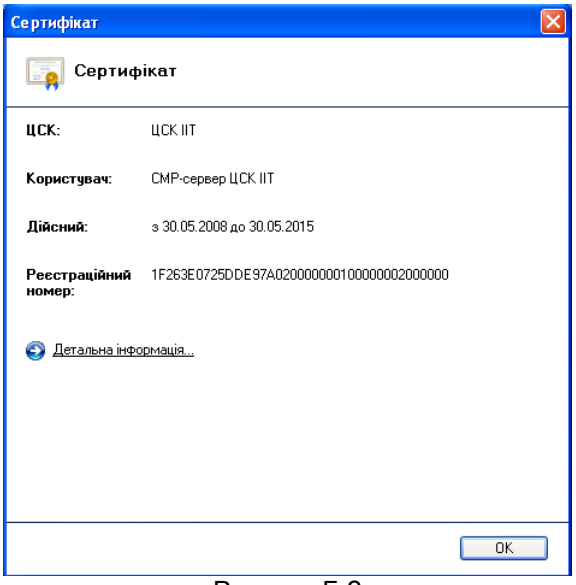

Рисунок Б.2

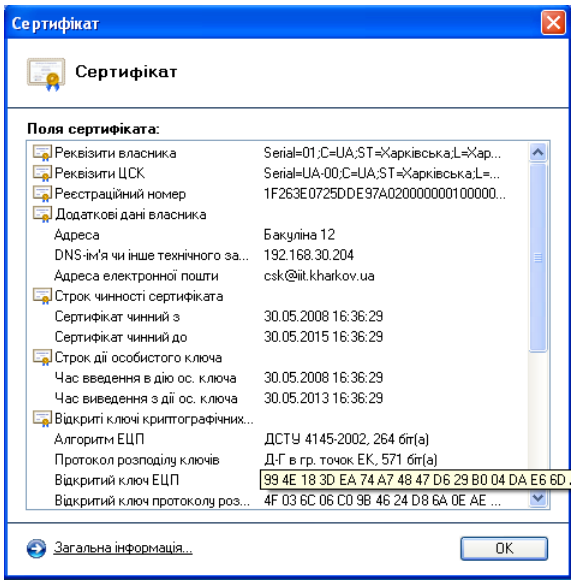

Рисунок Б.3

Пор. № зміни	Підпис відпов. особи	Дата внесення

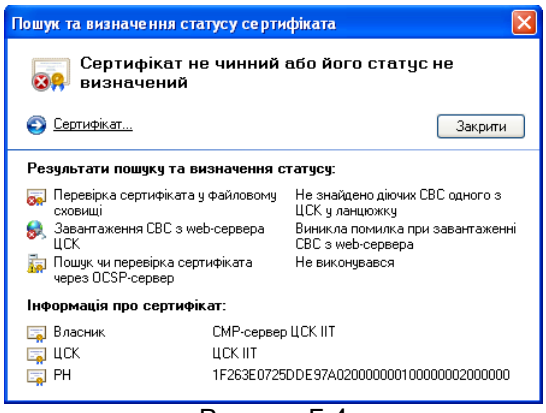

Рисунок Б.4

Б.3 Перегляд СВС

Вікно із списками відкликаних сертифікатів наведене на рис. Б.5.

Вікно перегляду СВС дозволяє видаляти СВС з файлового сховища, переглядати СВС та завантажувати СВС з web-сервера ЦСК.

Для перегляду СВС необхідно натиснути на відповідному записі про СВС у списку. СВС буде відображено у вікні що наведене на рисунках Б.6 та Б.7.

Для видалення файлу СВС з файлового сховища необхідно виділити відповідний запис про СВС у списку та натиснути кнопку “Видалити”.

Для імпорту СВС до файлового сховища необхідно натиснути “Імпортувати”, та обрати потрібний СВС на будь-якому носії інформації.

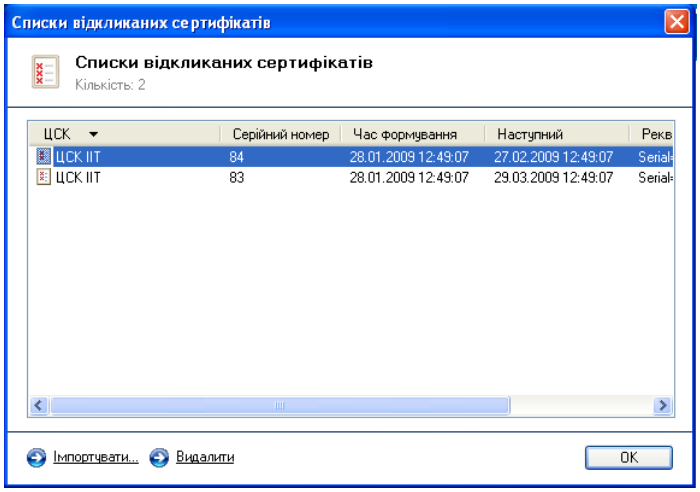

Рисунок Б.5

Пор. № зміни	Підпис відпов. особи	Дата внесення

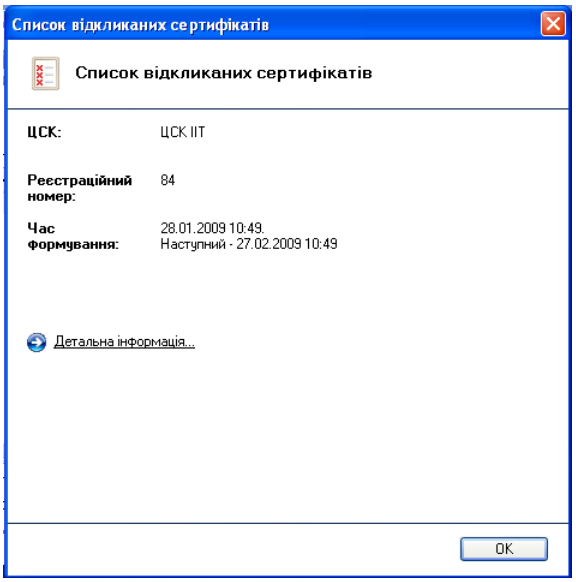

Рисунок Б.6

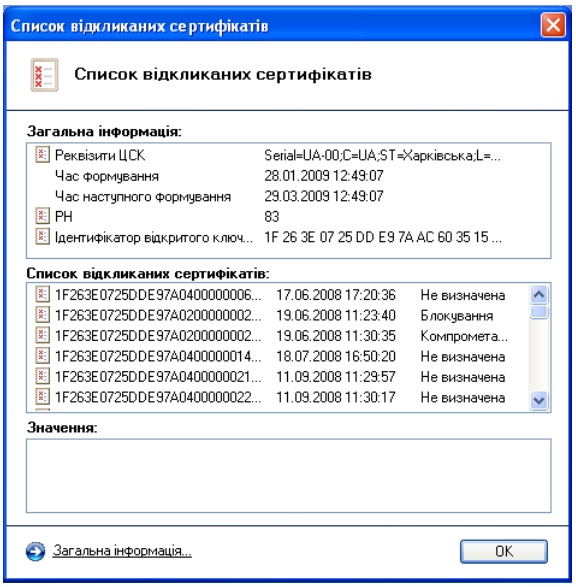

Рисунок Б.7

Б.4 Завантаження СВС

Для автоматичного завантаження списку відкликаних сертифікатів з web-сервера ЦСК необхідно відповідну позначку (“Завантажувати автоматично”) у вікні параметрів що наведене на рис. А.1.