Розділ 7

СТАН СТВОРЕННЯ ТА ЗАСТОСУВАННЯ ІНФРАСТРУКТУР З ВІДКРИТИМИ КЛЮЧАМИ

7.1. ПОНЯТТЯ СЕРТИФІКАТА ВІДКРИТОГО КЛЮЧА

Існує значне число асиметричних криптографічних систем. Їх принциповою особливістю є те, що в них при виконанні криптографічних перетворень використовується одна або декілька асиметричних пар ключів. Наприклад, у RSA [7–10, 46] для ЕЦП та НШ використовуються різні асиметричні ключові пари (E_k, D_k). Кожна з них обчислюється випадково на основі, наприклад, вирішення ключового рівняння:

$$E_k \cdot D_k \equiv 1 (\mathrm{mod}\, \varphi(N)), \tag{7.1}$$

де $\varphi(N)$ – функція Ейлера;

$$\varphi(N) = \varphi(P \cdot Q) = (P-1)(Q-1). \tag{7.2}$$

Для криптографічних перетворень у полі $GF(p)$ [7–10, 40, 43, 46] кожна асиметрична ключова пара (x_A, Y_A) також породжується випадково. У цьому випадку x_A – випадкове число, а Y_A обчислюється як елемент поля:

$$Y_A = \Theta_0^{x_A} (\mathrm{mod}\, p). \tag{7.3}$$

Для криптографічних перетворень у групі точок еліптичної кривої [7–10, 15–16, 29–31, 35] кожна асиметрична ключова пара (d_A, Q_A), де $1 \le d_A < n$ є випадкове число, а Q_A – точка на еліптичній кривій обчислюється способом використання скалярного множення:

$$Q_A = d_A \cdot G (\mathrm{mod}\, q), \tag{7.4}$$

де G – базова точка на еліптичній кривій порядку, q – модуль перетворення.

У наведених криптосистемах, наприклад в RSA, ключ E_k ЕЦП, що призначений для підписування, будемо вважати особистим, у двох останніх криптографічних перетвореннях x_A та d_A також є особисті ключі ЕЦП – випадкові числа. Згідно з концепцією асиметричних систем щодо застосування особистих ключів

повинні безумовно бути виконаними вимоги забезпечення їх конфіденційності, цілісності, справжності й доступності. Указані вимоги можуть бути забезпечені кожним із користувачів, оскільки особистий ключ доступний тільки його власнику, і він повинен і може зберігати його в таємниці. Необхідно відзначити. що не обов'язково E_k вибирати як особистий ключ, можна вибрати і D_k, але його після вибору треба використовувати із забезпеченням конфіденційності, цілісності, справжності та доступності, причому конфіденційності безумовно.

Більш складними є задачі захисту відкритих ключів, у нашому випадку це D_k, Y_A та Q_A. Справа в тому, що вони повинні бути доступними всім користувачам, що виконують, наприклад, перевірку підписаних електронних документів, даних тощо. І за таких умов необхідно забезпечити їх цілісність, справжність і доступність. Основною концепцією вирішення цієї задачі є використання сертифікатів відкритих ключів, причому для різних застосувань – направленого шифрування, ЕЦП, криптографічного протоколу тощо.

Як у практичному, так і в теоретичному плані вирішення вказаних задач безпосередньо пов'язане з рекомендаціями X.509 [13, 14] Міжнародного союзу телекомунікації (ITU — International Telecommunication Union). Ці рекомендації є частиною рекомендацій серії X.500, що визначають стандарт служби каталогів. Каталог, по суті, є сервером або розподіленою системою серверів, що підтримують базу даних з інформацією про користувачів [13]. У цій інформації міститься відповідність імен користувачів та їхніх мережних адрес, а також інші атрибути користувачів.

У цілому, документ X.509 визначає каркас схеми надання послуг автентифікації каталогом X.500 своїм користувачам. Цей каталог може служити сховищем сертифікатів відкритих ключів, що обговорювались. Кожен сертифікат містить відкритий ключ користувача й підписується за допомогою секретного ключа надійного центру сертифікації. Окрім того, X.509 визначає альтернативні протоколи автентифікації, що будуються на використанні сертифікатів відкритих ключів. Стандарт X.509 виявляється важливим через те, що структура сертифікатів і протоколів автентифікації, обумовлених у X.509, використовується в багатьох випадках. Наприклад, формат сертифіката X.509 прийнятий у протоколах S/MIME, IP Security, SET [7, 51, 63] тощо.

Стандарт X.509 з'явився в 1988 році. Пізніше він був переглянутий, і в ньому були виправлені деякі недоліки захисту, що відображено в [51, 7]; виправлені рекомендації були опубліковані в 1993 році. Проект третьої версії з'явився в 1995 році. Нині чиною, у тому числі в Україні, є версія ДСТУ ITU-T Rec. X.509 | ISO/IEC 9594-8»Основні положення сертифікації ключів та сертифікації атрибутів». Рекомендації цього стандарту (у подальшому рекомендації X.509) базуються на використанні методів криптографії з відкритим ключем і цифровими підписами. Стандарт не змушує використовувати конкретний алгоритм. Схема цифрового підпису припускає використання функції гешування. Знову ж таки, стандарт не обумовлює наявності конкретного алгоритму гешування. Рекомендації 1988 року включали опис алгоритму гешування, що рекомендується, але згодом з'ясувалося, що цей алгоритм ненадійний, і тому в рекомендації 1993 року він не ввійшов. З 1 квітня 2007 року ISO/IEC 9594-8 | ITU-T Rec. X.509 прийнятий в Україні як національний стандарт та визначений як ДСТУ ISO/IEC

9594-8:2006»Інформаційні технології – Взаємодія відкритих систем – Каталог. Частина 8. Основні положення щодо сертифікації відкритих ключів та атрибутів». Наказ Держспоживстандарту від 26.12.2006 № 372.

Сертифікати

Головним елементом схеми X.509 є сертифікати відкритих ключів, що зв'язуються з кожним користувачем. Передбачається, що ці сертифікати користувача видаються деяким надійним центром сертифікації (CA — Certification Authority) і розміщаються в каталозі або центром сертифікації, або користувачем. Сервер каталогів безпосередньо не відповідає за створення відкритих ключів або функції сертифікації; а просто надає легко доступне користувачам місце одержання сертифікатів.

Згідно ДСТУ ITU-T Rec. X.509 | ISO/IEC 9594-8 [13, 14] допускається існування трьох типів сертифікатів – безпосередньо *сертифіката*, *сертифіката дворазового застосування* та *сертифіката шифрування*. Розглянемо їх визначення згідно з наведеним у [13].

Сертифікат являє собою цифрове зображення інформації, яке щонайменше:

– ідентифікує уповноважений орган видачі сертифіката;

– містить імена або ідентифікатори його абонента;

– містить відкритий ключ ЕЦП абонента;

– вказує на його операційний період, тобто період дійсності;

– у цифровій формі підписується центром сертифікації ключів, що видає його, з використанням особистого ключа цього центру.

Сертифікат дворазового використання – сертифікат, призначений для застосування як при наданні послуги ЕЦП, так і послуги направленого шифрування даних. Такий сертифікат містить два відкритих ключі – перевірки ЕЦП та направленого зашифрування.

Сертифікат шифрування – сертифікат, що містить відкритий ключ, який використовується для направленого шифрування електронних повідомлень, файлів, документів або передачі даних, або для встановлення чи обміну ключів сеансу для таких самих цілей.

7.2. ЗАГАЛЬНА ХАРАКТЕРИСТИКА СТАНДАРТУ ДСТУ ITU-T REC. X.509 | ISO/IEC 9594-8

Стандарт ДСТУ ITU-T Rec. X.509 | ISO/IEC 9594-8 «Основні положення сертифікації ключів та сертифікації атрибутів» визначає три таких основних положення [13, 14]:

1) сертифікації відкритого ключа;

2) сертифікації атрибутів;

3) послуги автентифікації.

Основні положення сертифікації відкритого ключа, визначені в стандарті, містять визначення інформаційних об'єктів для інфраструктури відкритого ключа (ІВК), у тому числі сертифікати відкритого ключа та списки скасування сертифікатів (ССС).

Основні положення сертифікації атрибутів містять визначення інформаційних об'єктів для інфраструктури управління повноваженнями (ІУП), у тому числі сертифікати атрибутів і списки скасування сертифікатів атрибутів (ССА). Окрім того, у стандарті також визначаються основні положення для випуску, управління, використання і скасування сертифікатів.

Для обох типів сертифікатів і для всіх схем списків скасування також включається механізм розширення. Національний стандарт містить також набір стандартних розширень для всіх типів сертифікатів та схем списків скасування, що вважаються корисними у багатьох застосуваннях ІВК та ІУП. Стандарт також містить компоненти схеми, у тому числі класи об'єктів, типи атрибутів і правила зіставлення для збереження об'єктів ІВК та ІУП у Каталозі. Інші елементи ІВК та ІУП, такі як протоколи управління ключами та сертифікатами, операційні протоколи, додаткові розширення сертифіката та ССС, виходять за рамки цих основних положень.

Каталог дозволяє використовувати сертифікати відкритого ключа та сертифікати атрибутів. У стандарті визначено основи для використання Каталогом таких засобів. Технологія відкритого ключа, що включає сертифікати відкритого ключа, використовується Каталогом для виконання суворої автентифікації, операцій підпису та/або зашифровування і для збереження підписаних та/або зашифрованих даних у Каталозі. Сертифікати атрибутів можуть використовуватися Каталогом для здійснення управління доступом згідно з чинною базою правил. Основні положення для цього визначаються в специфікації стандарту.

У версії стандарту ДСТУ ITU-T Rec. X.509 | ISO/IEC 9594-8, що прийнята як національний стандарт, з точки зору надання послуг автентифікації, міститься:

– визначення форми подання інформації автентифікації, яка зберігається Каталогом;

– опис того, яким чином інформація автентифікації може бути отримана з Каталогу;

– опис припущень, які зроблено щодо способу формування та розміщення інформації автентифікації в Каталозі;

– визначення трьох способів використання додатками інформації автентифікації та опис того, як шляхом автентифікації можуть підтримуватися інші послуги;

– інформація про те, що схема автентифікації, визначена в стандарті, є універсальною і може бути застосована в інших додатках і середовищах.

У цій версії національного стандарту описано два рівні автентифікації:

1) *проста автентифікація*, у якій для перевірки особи, що надіслала запит на автентифікацію, використовується пароль;

2) *сувора автентифікація*, у якій для перевірки особи, що надіслала запит на автентифікацію, використовується ключ та відповідне криптографічне перетворення.

Оскільки проста автентифікація надає лише обмежений захист від несанкціонованого доступу, то як основу для забезпечення послуг автентифікації рекомендується використовувати виключно сувору автентифікацію. Стандарт, що розглядається, не вимагає використання для автентифікації тільки суворих схем, але рекомендується використовувати їх у додатках, де вони конче потрібні.

Більш детальну характеристику стандарту дає зміст стандарту, у тому числі й щодо обсягу матеріалу, який наведено нижче [14].

7.3. ФОРМАТ СЕРТИФІКАТА ВЕРСІЇ 3(V3) СТАНДАРТУ ДСТУ ITU-T REC. X.509 | ISO/IEC 9594-8

Щодо ІВК в стандарті використовуються дві основні структури даних: сертифікат відкритого ключа та список скасування сертифікатів. У цьому параграфі згідно із стандартом наводяться формати сертифіката відкритого ключа та списку скасування сертифікатів [13, 14].

7.3.1. Формат сертифіката відкритого ключа

При використанні відкритого ключа користувачі потребують упевненості в тому, що особистий ключ, асоційований з даним відкритим ключем, є власністю саме того суб'єкта (особистості чи системи), який використовує його для цифрового підпису чи шифрування. Указане досягається шляхом використання сертифікатів відкритого ключа. Сертифікат відкритого ключа являє собою структуру даних, що зв'язує значення відкритого ключа з суб'єктом і третьою довірчою стороною – уповноваженим на сертифікацію, якою є, скажімо, ЦСК. Цей зв'язок підтверджується цифровим підписом, який виробляється уповноваженим на сертифікацію з використанням особистого ключа. Також у зв'язку з тим, що підпис сертифіката та його строк чинності можуть бути незалежно перевірені кінцевим об'єктом, який використовує сертифікат, сертифікат може розповсюджуватись через ненадійні канали зв'язку та серверні системи, а також може зберігатись у незахищеному сховищі в системі, яка використовує сертифікат. Сертифікат має обмежений строк дії.

ДСТУ ITU-T Rec. X.509 | ISO/IEC 9594-8 визначає стандартний формат сертифіката. Формат сертифіката в стандарті 1988-го року називається форматом 1-ї версії (v1). При перегляді стандарту в 1993 році додали два поля, і цей формат став форматом версії 2 (v2). При останньому перегляді стандарту ISO/IEC, ITU-T та ANSI X9 розробили X.509-ту версію сертифіката, що називається версією 3 (v3).

Сертифікат має однозначно визначену певну структуру зі стандартним набором полів. Усі ці поля будуть описані нижче. Усі дані, які потрібно підписати, кодуються за правилами розпізнаного кодування ASN.1 (DER) [219].

Сертифікат являє собою сукупність полів, у які записується певна інформація. Приклад полів сертифіката v3 наведено в таблиці 7.1.

Таблиця 7.1. **Поля сертифіката з розширеннями**

Поля	Призначення поля
1	2
version	Описує версію сертифіката (v1, v2, v3)
SerialNumber	Унікально ідентифікує сертифікат
signature algorithm	Містить ідентифікатор для алгоритму, який використовується для підпису сертифіката
parameters	Необов'язкові параметри алгоритму

Закінчення табл. 7.2

1	2
issuer	Указує на об'єкт, який підписав і випустив сертифікат
validity notBefore	Період чинності сертифіката: Дата та час початку дії
notAfter	Дата та час закінчення дії
subject	Ідентифікує об'єкт, зв'язаний з відкритим ключем, що зберігається в полі «відкритий ключ суб'єкта»
SubjectPublicKeyInfo algorithm	Ідентифікує алгоритм, з яким використовується ключ
SubjectPublicKey	Містить відкритий ключ
UniqueIdentifiers	Служать для виключення можливості повторного використання імен емітента та суб'єкта
AuthorityKeyIdentifier	Ідентифікатор ключа уповноваженого
KeyIdentifier	Ієнтифікатор ключа
AuthorityCertIssuer	Назва уповноваженого
AuthorityCertSerialNumber	Серійний номер сертифіката уповноваженого
SubjectKeyIdentifier	Ідентифікатор ключа суб'єкта
KeyUsage	Використання ключа, бітовий рядок, де кожен біт вказує призначення ключа
PrivateKeyUsagePeriod	Строк дії таємного ключа підпису
CertificatePolicies	Містить послідовність з одного чи більше термів інформації політики
PolicyMappings IssuerDomainPolicy SubjectDomainPolicy	Використовується тільки для сертифіката уповноваженого
SubjectAltName	Надає альтернативне ім'я суб'єкту
IssuerAltName	Надає альтернативне ім'я емітенту
BasicConstraints	Відрізняє ключ уповноваженого від ключів кінцевих об'єктів
NameConstraints	Визначає сертифікацію домену по відношенню до підпорядкованого
SubjectDirectoryAttributes	Атрибути суб'єкта
CRLDistributionPoints	Пункти розповсюдження ССС

Основні поля сертифіката

Поле ***version (версія)*** – описує версію сертифіката. Якщо використовуються розширення сертифіката, то версія має бути 3 (значення 2). Якщо немає розширень, але присутній UniqueIdentifier, то версія має бути 2 (значення 1). Якщо присутні тільки базові поля, то версія має бути 1 (значення опускається). Реалізації, що застосовуються, повинні бути готові розпізнавати й підтримувати будь-яку версію сертифіката. Але щонайменше реалізації повинні розпізнавати версію 3.

Поле ***SerialNumber (серійний номер)*** – має бути невід'ємним цілим числом, яке назначається УС кожному сертифікату. Воно має бути унікальним для кожного сертифіката, що випущений даним УС, оскільки ім'я емітента і серійний номер забезпечують унікальність сертифіката. Серійний номер може бути великим цілим числом. Користувачі сертифіката мають бути спроможними використовувати значення SerialNumber довжиною до 20 октетів. УС не повинні використовувати значення довше, ніж 20 октетів.

Поле ***signature (підпис)*** – містить ідентифікатор алгоритму, що використовується УС для підпису сертифіката. Це поле повинно містити те саме значення, що й поле SignatureAlgorithm у послідовності Certificate. Вміст поля необов'язкових параметрів може змінюватись залежно від алгоритму. Можуть використовуватись як стандартні, так і інші ідентифікатори алгоритму.

Поле ***«Емітент» (issuer)*** – вказує на об'єкт, який підписав і випустив сертифікат. Воно повинно містити непусте розпізнавальне ім'я (DN). Поле емітента має тип Name з X.501. Цей тип складається з атрибутів. Стандартний набір атрибутів визначений у X.520. У полях «емітент» і «суб'єкт» можуть міститися такі атрибути:

– країна;
– організація;
– підрозділ організації;
– специфікатор розпізнавального імені;
– назва штату чи області;
– загальне ім'я;
– серійний номер.

Як додаток реалізації повинні бути готові отримати такі атрибути:

– місцевість;
– звання;
– прізвище;
– дане ім'я;
– ініціали;
– псевдонім;
– специфікатор номера версії.

Поле ***validity (дійсність)*** – сертифіката вказує інтервал часу, протягом якого УС гарантує, що буде підтримувати інформацію про статус сертифіката. Поле визначається як послідовність із двох дат: дати початку періоду дійсності (notBefore) і дати закінчення цього періоду (notAfter). Обидві дати можуть бути представлені у форматі UTCTime чи GeneralizedTime [57, 59].

Поле ***Subject (суб'єкт)*** – ідентифікує об'єкт, зв'язаний з відкритим ключем, який зберігається в компоненті SubjectPublicKey поля «Інформація про відкритий ключ суб'єкта». Ім'я суб'єкта може передаватися в полі суб'єкта та/чи у розширені SubjectAltName.

Якщо поле суб'єкта не пусте, то воно має містити розпізнавальне ім'я X.500 (DN). Це ім'я має бути унікальним для кожного суб'єкта, сертифікованого одним УС. УС може випустити більше ніж один сертифікат з одним і тим самим DN для одного й того самого суб'єкта. Суб'єктом може виступати як УС, так і кінцевий об'єкт.

Поле ***SubjectPublicKeyInfo (Інформація про відкритий ключ суб'єкта)*** використовується для розміщення відкритого ключа та ідентифікації алгоритму, з яким використовується ключ. Алгоритм ідентифікується за допомогою структури Algorithm Identifier.

Поля ***UniqueIdentifiers (Унікальні ідентифікатори)*** є тільки в сертифікатах версії 2 або 3. Унікальні ідентифікатори емітента та суб'єкта служать для захисту від повторного використання імен емітента та суб'єкта.

Розширення сертифіката

З метою забезпечення гнучкості та масштабованості стандарт визначає механізми розширення сертифіката. У них міститься важлива інформація. Так, розширення дозволяють включати до сертифіката інформацію, яка не може бути представлена в основних полях.

Розширення, що затверджені цим стандартом, можна поділити на дві категорії: обмежувальні та інформаційні [13, 14]. *Обмежувальні розширення* обмежують область застосування ключа або самого сертифіката. *Інформаційні розширення* містять додаткову інформацію, яка може бути використана в прикладному програмному забезпеченні користувача сертифіката.

Обмежувальні розширення сертифіката

Базові обмеження (BasicConstraints). Розширення BasicConstraints дозволяє розрізняти суб'єкти сертифіката, оскільки основні поля сертифіката не розділяються залежно від типу користувача. Окрім цього, розширення визначає максимальну глибину дійсного шляху сертифікації, який включає цей сертифікат. Розширення є критичним.

Використання ключа (Key Usage). РозширенняKeyUsage визначає область використання ключа, що міститься в сертифікаті. Це розширення має вигляд бітового рядка, де кожен біт означає призначення ключа: цифровий підпис (0), причетність (1), шифрування ключа (2), шифрування даних (3), узгодження ключів (4), перевірка підпису сертифіката (5), підпис CCC (6), тільки шифрування (7), тільки розшифрування (8). Для визначення призначення потрібний біт встановлюється в 1. Можливе спільне використання бітів.

Розширене використання ключів (ExtKeyUsage). Це розширення визначає цілі, для яких може використовуватись сертифікований відкритий ключ, додатково до цілей, указаних у розширенні використання ключа. Звичайно, це розширення використовується для сертифікатів кінцевих об'єктів.

Політики застосування сертифікатів. До цієї групи можна віднести обмежувальні розширення CertificatePolicies, PolicyMappings та PolicyConstraints.

Розширення CertificatePolicies містить інформацію про правила застосування сертифіката. Організації можуть підтримувати досить велике коло додатків. Деякі сертифікати бувають більш надійними залежно від процедур

їх випуску або типів криптографічних модулів (наприклад, посилені сертифікати, які створені надійними засобами цифрового підпису). Різні організації та відомства використовують різні політики застосування сертифікатів. При цьому користувачі не завжди спроможні розрізняти ці політики. Для вирішення цієї проблеми використовується розширення CertificatePolicies. Це розширення містить унікальний ідентифікатор, який характеризує політику застосування сертифікатів.

Розширення PolicyMappings використовується УС. За допомогою розширення УС може фіксувати відповідність деяких своїх політик застосування сертифіката політикам застосування сертифікатів іншого УС. Розширення містить одну чи більше пар об'єктних ідентифікаторів. Кожна пара включає IssuerDomainPolicy та SubjectDomainPolicy. Це вказує на те, що УС, який випустив сертифікат, вважає свою політику такою, що відповідає політиці суб'єкта.

За допомогою ***розширення PolicyConstraints*** надається можливість визнати неправочинним розширення PolicyMappings у разі, якщо сертифікація виходить за межі домену. Обмеження впроваджується двома способами: використанням заборони відображення політики або вимогою, щоб кожний сертифікат на шляху містив ідентифікатор доступної політики.

Обмеження імені (NameConstraints). РозширенняNameConstraints вказує на те, у якому просторі імен мають знаходитись імена. Ці обмеження накладаються на розпізнавальні імена та на альтернативні імена суб'єкта. Таким чином, розширення використовується для визначення множини припустимих або неприпустимих імен та є механізмом підтвердження надійності сертифікатів.

Заборона будь-якої політики (InhibitAnyPolicy). Розширення Inhibit AnyPolicy обмежує використання будь-якої політики. Воно вказує на те, що спеціальне значення *any-policy* не задається у явному вигляді для інших політик застосування сертифікації.

Інформаційні обмеження сертифіката

Ідентифікатори ключів. Стандарт для ідентифікації ключів визначає два розширення: AuthorityKeyIdentifier та SubjectKeyIdentifier. Розширення AuthorityKeyIdentifier надає засоби ідентифікації відкритого ключа УС, який узгоджений з особистим ключем, що був використаний для підпису сертифіката. Це розширення використовується у разі, коли емітент має декілька ключів для підпису.

Користувачі також можуть володіти великою кількістю ключів або декількома сертифікатами для одного ключа. Розширення SubjectKeyIdentifier використовується для того, щоб розпізнавати ключі підпису одного й того самого користувача.

Альтернативні імена. З метою розширення границь ідентифікації суб'єктів сертифіката вводяться розширення SubjectAltName та IssuerAltName. Розширення дозволяють використовувати альтернативні імена, наприклад DNS-ім'я, IP-адреси, URL, e-mail тощо. Альтернативне ім'я повинно перевірятися у відповідності до регламенту УС. Окрім цього, підтримується можливість використовувати свої власні імена.

Період використання особистого ключа. Поле PrivateKeyUsagePeriod визначає період дійсності для особистого ключа в тому випадку, коли цей період відрізняється від періоду дійсності сертифіката. Поле складається з двох компонент: час початку і закінчення дії особистого ключа.

Атрибути каталогу суб'єкта. Розширення SubjectDirectoryAttributes використовується для передачі ідентифікаційних атрибутів суб'єкта. Розширення подається як послідовність одного чи більше атрибутів.

Пункти розповсюдження списку скасування сертифікатів (ССС). Розширення CRLDistributionPoints задає унікальний ідентифікатор ресурсу – URL для вказівки місцезнаходження списку скасування сертифікатів, а також ідентифікує способи отримання інформації.

Найновіший ССС. Розширення freshestCRL показує, як може бути отримана інформація відносно дельта-ССС. Воно має бути некритичним.

7.3.2. Список скасування сертифікатів та його розширення

Згідно з [13, 14] для скасування сертифікатів можна використовувати декілька механізмів. Базовим механізмом є використання Списку Скасування Сертифікатів (ССС). ССС являє собою структуру даних, у якій міститься послідовність інформації, що ідентифікує конкретні скасовані сертифікати та містить додаткову інформацію про скасування. ССС також є структурованим записом у форматі ASN.1. ССС підписується емітентом, яким є УС. Але стандарт не забороняє можливості делегування повноважень на випуск ССС іншим органам.

Формат ССС визначається міжнародним стандартом (табл. 7.2). Основним форматом на сьогодні є формат версії 2 (v2). Розрізняють декілька типів ССС – основний ССС та дельта-ССС. Список може також бути прямим і непрямим.

Основний (базовий) ССС – список, який містить усі скасовані сертифікати.

Дельта-ССС – список, який містить тільки зміни, які з'явилися за певний час в основному ССС, тобто містить список лише тих сертифікатів, статус яких змінився з часу випуску основного ССС.

Прямий список – список, який випускає УС, що випустив сертифікати.

Непрямий список – список, що випускає орган, уповноважений УС на випуск списку.

Поля ССС. У таблиці 7.2 наданий перелік полів сертифіката, їх пояснення подано нижче.

Таблиця 7.2. **Поля ССС та розширення**

Поле	Призначення поля
1	2
version	описує версію закодованого сертифіката
signature algorithm	містить ідентифікатор для алгоритму, який використовується УС для підпису ССС

Закінчення табл. 7.2

1	2
parameters	необов'язкові параметри алгоритму
issuer	вказує на об'єкт, який підписав і випустив ССС
thisUpdate	вказує на дату випуску цього ССС
nextUpdate	вказує на дату наступного випуску ССС
revokedCertificates	містить послідовність даних про скасовані сертифікати, присутнє необов'язково
userCertificate	містить серійний номер скасованого сертифіката
revocationDate	містить дату скасування
crlEntryExtensions	набір розширень запису
AuthorityKeyIdentifier	ідентифікатор ключа уповноваженого
keyIdentifier	ідентифікатор ключа
authorityCertIssuer	назва уповноваженого
authorityCertSerialNumber	серійний номер сертифіката уповноваженого
CRLNumber	некритичне розширення, яке передає номер, що послідовно й монотонно збільшується
crlScope	використовується для подання формулювань різноманітних типів ССС
deltaCRLIndicator	для ідентифікації ССС як дельта-ССС
IssuingDistributionPoint	ідентифікує пункт розповсюдження ССС і призначення для конкретного ССС
ffreshestCRL	вказує на спосіб отримання дельта-ССС для даного повного ССС
baseUpdateTime	використовується для визначення дати/ часу, після якої цей різницевий список забезпечує оновлення для статусу скасування
delta Info	використовується для того, щоб показати сторонам, що довіряють, що дельта-ССС доступні для ССС
ordered List	показує, що послідовність скасованих сертифікатів у полі revokedCertificates списку скасування сертифікатів сформована у зростаючому порядку
cRLStreamIdentifier	використовується для ідентифікації контексту, у межах якого номер ССС є унікальним
status Referrals	засіб для передачі інформації користувачам про повідомлення скасування сертифіката

Поле ***«Версія» (version)***. Це необов'язкове поле описує версію закодованого сертифіката. Якщо використовуються розширення, то версія повинна дорівнювати 2.

Поле ***«Підпис» (signature)***. Це поле містить ідентифікатор алгоритму, за допомогою якого емітент ССС підписав список сертифікатів (CertificateList).

Поле ***«Ім'я емітента» (issuer)***. Поле імені емітента вказує на об'єкт, який підписав і випустив ССС. Воно повинно містити непусте розпізнавальне ім'я (DN).

Поле ***«Дата випуску» (thisUpdate)***. Це поле вказує на дату випуску цього ССС. Воно може бути подано у форматі UTCTime чи GeneralizedTime.

Поле ***«Наступне оновлення» (nextUpdate)***. Це поле вказує на дату наступного випуску ССС. Він може бути виданий раніше, ніж указано, але не пізніше. Поле може бути подано у форматі UTCTime чи GeneralizedTime.

Поле ***«Скасовані сертифікати»*** (revokedCertificates). Коли немає скасованих сертифікатів, то цей список має бути відсутнім. В іншому випадку, скасовані сертифікати повинні розміщатися в списку за їхніми серійними номерами. Також тут повинна міститись дата скасування кожного сертифіката.

Розширення ССС (таблиця 7.3). Це поле забезпечує зв'язок додаткових атрибутів з ССС. Розширення визначаються такими документами: ANSI X9, ISO/IEC та ITU-T для X.509 v2 ССС. Як і в сертифікаті, розширення ССС можуть бути *критичними* або *некритичними*.

Таблиця 7.3. **Розширення запису ССС**

Поле	Призначення поля
cRLReason	причина скасування конкретного сертифіката
holdInstructionCode	містить зареєстрований ідентифікатор команди, який вказує на дію, яку треба виконати, коли зустрічається заблокований сертифікат
invalidityDate	вказує на дату, коли відомо, чи очікувалось, що особистий ключ був скомпрометований
certificateIssuer	вказує на емітента сертифіката

Розширення ***«Ідентифікатор ключа уповноваженого органу» (AuthorityKeyIdentifier)***. Розширення «Ідентифікатор ключа уповноваженого органу» дозволяє ідентифікувати відкритий ключ, що відповідає особистому ключу, яким був підписаний ССС. Ця ідентифікація ґрунтується або на використанні ідентифікатора ключа або на використанні імені емітента та серійного номера.

Розширення ***«Номер ССС» (cRLNumber)***. Розширення cRLNumber – це некритичне розширення, яке передає номер, що послідовно й монотонно збільшується, для заданого призначення ССС та його емітента. Це розширення дає змогу користувачам легко дізнатись, коли конкретний ССС замінюється іншим.

Якщо емітент ССС випускає дельта-ССС додатково до повного, з тим самим призначенням, то вони (дельта і повний ССС) повинні розділяти одну й ту саму послідовність номерів. Якщо вони видаються одночасно, то їхній номер ССС має бути однаковим.

Розширення ***«Вказівник дельта-ССС» (deltaCRLIndicator)***. Розширення «Вказівник дельта-ССС» – це критичне розширення ССС для ідентифікації ССС як дельта-ССС. Воно повинно включатись, якщо емітент ССС використовує дельта-ССС у своїй роботі. Дельта-ССС має містити ті самі причини скасування і той самий набір сертифікатів, що й основний ССС, на який вказує цей частковий ССС.

Розширення ***«Випускний пункт розповсюдження» (Issuing Distribution Point)***. Це розширення є критичним та ідентифікує пункт розповсюдження ССС і призначення для конкретного ССС. Також воно визначає, чи покриває даний ССС скасування тільки для сертифікатів кінцевих об'єктів, сертифікатів УС, сертифікатів атрибутів, чи обмежені набори кодів причин скасування. Якщо це розширення відсутнє, то в ССС повинні міститись усі скасовані сертифікати. Також потрібно відзначити, що пункти розповсюдження ССС не мають своїх власних ключових пар.

Розширення ***«Найновіший ССС» (freshestCRL)***. Це розширення вказує на спосіб отримання дельта-ССС для даного повного ССС. Воно не використовується в дельта-ССС і має бути некритичним.

Розширення ***«Оновлення бази» (baseUpdateTime)***. Поле «Оновлення бази» призначене для використання в дельта-ССС та використовується для визначення дати/ часу, після якої цей різницевий список забезпечує оновлення для статусу скасування. Це розширення повинне використовуватися тільки в дельта-ССС, які містять розширення deltaCRLIndicator.

Розширення ***«Дельта-інформація» (delta Info)***. Це розширення призначене для використання в тих ССС, які не є дельта-ССС. Воно використовується для того, щоб показати сторонам, що довіряють, що дельта-ССС доступні для ССС, які містять це розширення. Розширення вказує адресу, за якою можна знайти зв'язані дельта-ССС, час випуску наступного дельта-ССС (не обов'язково).

Розширення ***«Упорядкований список» (ordered List)***. Розширення «Упорядкований список» показує, що послідовність скасованих сертифікатів у полі revokedCertificates списку скасування сертифікатів сформована у зростаючому порядку за допомогою серійного номера сертифіката або дати скасування.

Розширення ***«Ідентифікатор потоку ССС» (cRLStreamIdentifier)***. Поле «Ідентифікатор потоку ССС» використовується для ідентифікації контексту, у межах якого номер ССС є унікальним. Кожне значення цього розширення для одного й того самого уповноваженого органу має бути унікальним.

Розширення ***«Передача на розгляд статусу» (status Referrals)***. Це розширення ССС призначене для використання в межах структури ССС як засіб для передачі інформації користувачам про повідомлення скасування сертифіката.

Розширення ***«Область дії ССС» (crlScope)***. Область дії ССС указується в межах самого ССС. Для запобігання атаки типу заміни ССС з боку додатка, що не підтримує розширення області дії, розширення «Область дії ССС», якщо воно присутнє, повинне позначатися як критичне.

Розширення запису ССС. Усі розширення, рекомендовані для використання в даному полі, є некритичними.

Код причини скасування (reason Code). У полі цього розширення вказується причина скасування конкретного сертифіката. У розширенні рекомендується

розміщувати змістовні коди причини скасування. Якщо причина не визначена, то воно має бути відсутнім.

Код команди блокування (holdInstructionCode). У полі «Код команди блокування» міститься зареєстрований ідентифікатор команди, що вказує на дію, яку треба виконати, коли зустрічається заблокований сертифікат.

Дата недійсності (invalidity Date). Це поле вказує на дату, коли достовірно відомо, чи вважається, що особистий ключ був скомпрометований. Ця дата завжди менша за дату скасування в записі ССС.

Емітент сертифіката (certificate Issuer). Це розширення запису ССС вказує на емітента сертифіката, асоційованого із записом в непрямому ССС.

Аналіз підходів щодо використання структур даних і механізмів розширень сертифікатів в ІВК США та Європейському Союзі. З метою визначення підходів щодо використання стандартних структур даних, які визначені в міжнародному стандарті, в Україні доцільно здійснити аналіз відповідних підходів в інших країнах.

7.4. ОСОБЛИВОСТІ СТРУКТУРИ СЕРТИФІКАТІВ ІВК США ТА ЄВРОПЕЙСЬКОГО СОЮЗУ

Аналіз стану та концептуальних положень розроблення й застосування ІВК в США показує, що провідні компанії та федеральні органи використовують як основну структуру сертифіката, що визначена в ISO/IEC 9594-8 | ITU-T Rec. X.509. Але оскільки сам названий стандарт недостатньо конкретизує поля сертифікатів, то в США було вирішено уточнити та прийняти свій федеральний профіль стандарта. Прийнятий профіль практично співпадає із сертифікатом, що міститься в стандарті, окрім двох полів – subjectUniqueID та issuerUniqueID. Ці поля включені в сертифікат вказаного профілю додатково і не можуть розпізнаватися клієнтами ІВК, тобто згідно ISO/IEC 9594-8 | ITU-T Rec. X.509. Також для внутрішніх потреб використовується механізм розширень, що дозволений стандартом. Розширення складаються з імені, поля критичності та самого значення. Причому якщо розширення критичне, то користувач повинен обробляти його, якщо ж він не розпізнає таке розширення, то воно ігнорується. У таблиці 7.4 наведені розширення сертифіката, перелік об'єктів ІВК, що їх використовують, їх призначення, а також встановлюється критичність або некритичність розширення. У таблиці 7.5 наведені дані щодо підтримки розширень сертифіката згідно з «Minimum Interoperability Specification for PKI Components, Version 1 (MISPC)». Необхідно відзначити, що в таблиці 7.5 наведені дані щодо підтримки розширень клієнтами, а також наявності та особливостей розширень у різних видах сертифікатів.

Для скасування сертифікатів у Федеральній ІВК США застосовується механізм *списків скасування сертифікатів* (ССС, версії 2). Для підтримки необхідної додаткової інформації використовуються відповідні розширення. Особливістю профілів федеральних стандартів є спроби скоротити деякі поля сертифікатів. Указане зумовлено необхідністю задоволення вимог щодо зменшення часової складності обробки сертифікатів у прикладних системах.

Таблиця 7.4. **Розширення сертифіката**

Розширення	Використовується	Особливості використання	Критичність
1	2	3	4
Інформація про ключ та політику			
keyIdentifier	усіма	ідентифікує ключ, що застосований при виготовленні сертифіката	Ні
AuthorityKeyIdentifier	усіма	унікальний ідентифікатор ключа авторизації для кожного УС	
authorityCertIssuer	усіма	альтернатива відносно ідентифікатора ключа	
authorityCertSerialNumber	усіма	використовується разом з authorityCertIssuer	
SubjectKeyIdentifier	усіма	визначає відмінності між різними ключами суб'єкта	Ні
keyUsage	усіма	визначає призначення ключа	Так*
extendedKeyUsage	усіма	визначає розширене призначення ключа	Ні*
privateKeyUsagePeriod	усіма	строк використання таємного ключа, тільки для ключів цифрового підпису	Ні*
CertificatePolicies	усіма	ідентифікатори політики, кожен з яких визначає конкретну політику, що стосується сертифіката	Ні*
policyIdentifiers	усіма	ідентифікатор політики	
policyQualifiers	усіма	додаткова інформація про політику	
policyMappings	УС	називає еквівалентні політики	Ні
Атрибути емітента та суб'єкта сертифіката			
SubjectAltName	усіма	містить список альтернативних імен суб'єкта (наприклад, rfc822 ім'я, X.400 адресу, IP-адресу тощо)	Ні*
issuerAltName	усіма	містить список альтернативних імен емітента	Ні*
subjectDirectoryAttributes	усіма	атрибути об'єктів (наприклад, алгоритм, який підтримується)	Ні

Закінчення табл. 7.4

1	2	3	4
Обмеження шляху сертифікації			
basicConstraints	усіма	базові обмеження на роль суб'єкта та довжину ключа	Так *
CA	усіма	відрізняє сертифікат УС від сертифіката кінцевого об'єкта	
pathLenConstraint	УС	Максимальне число УС на шляху сертифікації, причому 0 показує, що УС тільки випускає сертифікати кінцевого об'єкта	
nameConstraints	УС	обмежує простір імен сертифіката УС	Так *
permittedSubtrees	УС	називає зовнішні піддерева та зв'язки	
excludedSubtrees	УС	показує недозволені піддерева	
policyConstraints	усіма	обмежує випущені сертифікати	Так *
requireExplicitPolicy	усіма	усі сертифікати, що містяться на шляху сертифікації, повинні містити в собі прийнятний ідентифікатор політики	
inhibitPolicyMapping	усіма	забороняє відображення політик у наступних сертифікатах	
Ідентифікація ССС			
crlDistributionPoints	усіма	ділить ССС на короткі списки	Ні*
distributionPoint	усіма	місце, де можна отримати ССС	
reasons	усіма	причина за якої сертифікат міститься в ССС	
CRL Issued	усіма	ім'я того, хто випустив ССС	

Примітка

* Стандарт встановлює критичність (Так*) або некритичність (Ні*).

Таблиця 7.5. **Розширення ССС**

Розширення	Застосування	Критичність
1	2	3
AuthorityKeyIdentifier	ідентифікує особистий ключ УС, яким підписаний ССС	Ні
keyIdentifier	унікальний ідентифікатор ключа, що є альтернативним certIssuer & authorityCertSerialNumber	
certIssuer	ім'я емітента сертифіката УС	

Закінчення табл. 7.5

1	2	3
authorityCertSerial Number	використовується із certIssuer; причому комбінація має бути унікальною	
issuerAltName	альтернативне ім'я емітента ССС	Ні*
CRLNumber	порядковий номер ССС	Ні
issuingDistribution Point	ім'я пункту розповсюдження ССС; також указує на причини для скасування	Так
deltaCRLIndicator	вказує на різницевий ССС	Так

Примітка

* Стандарт встановлює критичність (Так*) або некритичність (Ні*).

Аналіз нормативних документів Європейського Союзу показав, що структура сертифіката визначена в документі ETSI TS 102 280 «X.509 v3 Certificate Profile for Certificates Issued to Natural Persons» [220]. Він достатньо чітко визначає поля, що мають бути присутні у сертифікаті. Крім того, в ньому також наведені рекомендовані вимоги до кожного поля, у тому числі розширення. Приклад структури сертифіката згідно з указаним документом наведено в таблиці 7.6.

Таблиця 7.6. **Структури сертифіката ЄС**

Поле	Рекомендації
1	2
Validity	До 2049 року дата чинності сертифіката кодується у форматі UTCTime, у 2050 та пізніше кодується як GeneralizedTime
Version	Немає спільних вимог
Serial namber	Серійний номер має бути невід'ємним цілим числом. Не повинен бути більше 20 октетів
Signature	Sha-1 WithRSAEncryption OBJECT IDENTIFIER :: = {iso(1) member-body(2) us(840) rsadsi(113549) pksc(1) pksc-1(1)5}
Issuer	Усі сертифікати, випущені після 31.12.2003, повинні використовувати UTF8String кодування. Поле повинно ідентифікувати організацію, яка відповідає за випуск сертифікатів. Назва має бути офіційно зареєстрованою назвою організації
Subject	Повинне обов'язково містити атрибут countryName
Subject public key info	Використовується RSA шифрування. pkcs-1 OBJECT IDENTIFIER :: = {iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)1} rsaEncryption OBJECT IDENTIFIER :: = {pkcs– 1(1)}

Закінчення табл. 7.4

Authority key identifier	Включається в усі сертифікати кінцевих обєктів. Значення цього поля отримують зі значення відкритого ключа, яким перевіряють підпис. Не повинно бути критичним
Subject key identifier	Отримується зі значення відкритого ключа за допомогою гешування 160-bit SHA-1 або значення 0100 та найменш значимих 60 бітів SHA-1. Не повинно бути критичним
KeyUsage	Це розширення є обовязковим
Private key usage period	Не повинно використовуватись разом з Internet PKI. Не повинно бути критичним
Certificate policies	Якщо УС не обмежує набір політик для шляху сертифікації, що включає даний сертифікат, то він може використати спеціальну політику anyPolicy, що має значення {2 5 29 32 0}. Може помічатися як критичне
Policy mappings	Не повинно використовуватись у сертифікатах кінцевих об'єктів
Subject alternative name	При генерації сертифікатів, які містять адресу електронної пошти, повинно відповідати rfc822Name
Issuer alternative name	Немає спеціальних вимог
Subject directory attributes	Має бути некритичним
Basic constraints	Може бути як критичним, так і некритичним
Name constraints	Не повинно включатися в сертифікати кінцевих об'єктів
Policy constraints	Не повинно включатися в сертифікати кінцевих об'єктів
Extended key usage	Немає спеціальних вимог
CRL distribution points	Немає спеціальних вимог
Inhibit any-policy	Не повинно включатися в сертифікати кінцевих об'єктів
Freshest CRL	Має бути некритичним
Authority information access	Має бути некритичним
Subject information access	Має бути некритичним
Biometric information	Не повинно бути критичним

У цілому, можна зробити висновок, що в США та ЄС використовують стандартні формати сертифіката та ССС, які визначені в ISO/IEC 9594-8 | ITU-T Rec. X.509. Для забезпечення тих чи інших вимог використовується механізм розширень.

7.5. СЕРТИФІКАТИ ВІДКРИТИХ КЛЮЧІВ АТРИБУТІВ

Міжнародний стандарт ITU-T Rec. X.509 | ISO/IEC 9594-8 визначає основні положення сертифікації атрибутів. Сертифікати атрибутів призначені для використання у тих випадках, коли потрібно мати дійсні відомості про суб'єкт. Ці відомості і можуть використовуватись у критеріях прийняття рішень про права доступу, категорії допуску, належність до певної групи або платіжну спроможність та іншу інформацію авторизації суб'єкта. Особливістю такої інформації є те, що вона має у порівнянні із сертифікатом відкритого ключа менший, або навіть суттєво менший, строк дії.

Складність або й неможливість застосування для вирішення вказаних задач сертифікатів відкритих ключів пов'язана з такими факторами:

1) сертифікат відкритого ключа, що містить інформацію авторизації, повинен бути скасованим за будь-яких змін інформації авторизації. У той же час сертифікат відкритого ключа випускається на довгий строк (рік або більше).

2) уповноважений орган, що випускає сертифікат з атрибутами суб'єкта, як правило, не має повноважень підписувати цю інформацію, а повинен узгоджувати це з джерелом інформації щодо прав доступу конкретного користувача.

3) сертифікат атрибутів зв'язує атрибути користувача з особою власника сертифіката. Крім того, сертифікат атрибутів використовується разом із сертифікатом відкритого ключа, причому кожен має свою роль.

4) автентифікація суб'єкта повинна здійснюється з використанням сертифіката відкритого ключа, у той же час зв'язування атрибутів із суб'єктом повинно здійснюватись за допомогою сертифіката атрибутів.

Зважаючи на вказане, можна зробити висновок, що сертифікат атрибутів є структурою, яка відокремлена від сертифіката відкритого ключа суб'єкта. Але, як правило, він повинен застосовуватись разом із сертифікатом відкритого ключа. Розглянемо формат сертифіката атрибутів [13, 14].

Із наведеного можна зробити висновок, що сертифікат атрибутів є структурою, яка відокремлена від сертифіката відкритого ключа суб'єкта. Це пов'язано з тим, що суб'єкт може мати безліч сертифікатів атрибутів, зв'язаних з кожним із певних сертифікатів відкритого ключа. Також не існує вимог, щоб один і той самий уповноважений орган створював для користувача і сертифікат відкритого ключа, і сертифікат(и) атрибутів. В інформаційних середовищах, де різні уповноважені органи відповідають за випуск як сертифікатів відкритого ключа, так і сертифікатів атрибутів, сертифікат(и) відкритого ключа, випущений уповноваженим на сертифікацію (УС), і сертифікати атрибутів, випущені уповноваженим з атрибутів (УА), повинні бути підписані з використанням різних особистих ключів. Якщо один об'єкт є одночасно й УС, що випускає сертифікати відкритого ключа, й УА, що випускає сертифікати атрибутів, то настійно рекомендується, щоб для підписання сертифікатів атрибутів використовувався інший ключ, ніж той, що використовується для підписання сертифікатів відкритого ключа.

Основні поля сертифіката атрибутів наведені в таблиці 7.7.

Поле «Версія» (version). Це поле встановлює розходження між різними версіями сертифіката атрибутів. Для сертифікатів атрибутів, випущених відповідно до синтаксису стандарту ITU-T Rec. X.509 | ISO/IEC 9594-8, версія повинна бути 2.

Поле «Утримувач» (holder). Це поле містить інформацію про особу утримувача сертифіката атрибутів. Воно містить реєстраційний номер базового

сертифіката (baseCertificateID), ім'я утримувача (entityName) та інформацію підпису (objectDigestInfo). Якщо присутній компонент baseCertificateID, він визначає сертифікат відкритого ключа, який повинен використовуватися для автентифікації особи утримувача, коли повноваження пред'являються за допомогою сертифіката атрибутів.

Якщо компонент entityName присутній, то він визначає одне або декілька імен для утримувача. Якщо entityName є тільки компонентом, який присутній в holder, то будь-який сертифікат відкритого ключа, що має одне з таких самих імен, як і суб'єкт, може використовуватися для автентифікації особи цього утримувача, коли повноваження пред'являються за допомогою сертифіката атрибутів. Якщо одночасно присутні й baseCertificateID, й entityName, то може бути використаним тільки сертифікат, заданий у baseCertificateID. У цьому випадку entityName включається тільки як інструмент, що допомагає перевірнику повноважень знайти указаний сертифікат відкритого ключа.

Якщо присутній компонент objectDigestInfo, то він використовується безпосередньо для автентифікації особи утримувача. Утримувач автентифікується шляхом порівняння цифрового підпису відповідної інформації, створеної перевірником повноважень тим самим алгоритмом, що визначений в objectDigestInfo, зі змістом objectDigest. Якщо вони є ідентичними, то утримувач автентифікується з метою пред'явлення повноважень за допомогою цього сертифіката атрибутів.

Поле «Емітент» (issuer). Поле емітента вказує на об'єкт, який підписав і випустив сертифікат. Воно містить ім'я емітента (issuerName), реєстраційний номер базового сертифіката (baseCertificateID) та інформацію підпису (objectDigestInfo).

Компонент issuerName (якщо присутній) визначає одне або більше імен для емітента.

Компонент baseCertificateID (якщо присутній) визначає емітента шляхом посилання на конкретний сертифікат відкритого ключа, для якого цей емітент є суб'єктом.

Компонент objectDigestInfo (якщо присутній) визначає емітента шляхом надання геш-коду від ідентифікуючої інформації для емітента.

Поле «Підпис» (signature). Визначає криптографічний алгоритм, який використовується для вироблення цифрового підпису сертифіката атрибутів.

Поле «Реєстраційний номер» (serialNumber). Містить реєстраційний номер, який унікально визначає сертифікат атрибутів у межах області дії емітента такого сертифіката.

Поле «Строк чинності» (attrCertValidityPeriod). Це поле визначає інтервал часу, під час якого сертифікат атрибутів вважається чинним, він задається у форматі GeneralizedTime. Поле містить дві дати: дату початку строку чинності (notBefore) і дату закінчення цього строку (notAfter).

Поле «Атрибути» (attributes). Поле містить зв'язані з утримувачем атрибути, які сертифікуються (наприклад, повноваження).

Поле «Реєстраційний номер емітента» (issuerUniqueID). Може використовуватися для визначення емітента сертифіката атрибутів у разі якщо компонента емітента недостатньо.

Поле «Розширення» (extensions). Це поле дозволяє виконувати додавання нових полів до сертифіката атрибутів.

Таблиця 7.7. **Поля сертифіката атрибутів з розширеннями**

Поле	Призначення
1	2
version	вказує версію сертифіката атрибутів
holder baseCertificateID entityName objectDigestInfo	унікально ідентифікує сертифікат атрибутів вказує емітента та реєстраційний номер сертифіката відкритого ключа утримувача вказує ім'я об'єкта чи ролі використовується для безпосередньої автентифікації утримувача
issuer issuerName baseCertificateID objectDigestInfo	вказує на об'єкт, який підписав і випустив сертифікат атрибутів вказує ім'я емітента вказує реєстраційний номер сертифіката відкритого ключа емітента використовується для безпосередньої автентифікації емітента
signature	містить ідентифікатор алгоритму, який використовується УА для підпису сертифіката
serialNumber	число, яке унікально ідентифікує сертифікат
attrCertValidityPeriod notBefore notAfter	період чинності сертифіката: дата та час, відколи сертифікат стає чинним дата та час втрати чинності сертифікатом
Attributes	містить зв'язані із утримувачем атрибути
issuerUniqueID	може використовуватися для однозначного визначення емітента
extensions	дозволяє додавати нові поля до сертифіката атрибутів
time Specification	використовується УА для обмеження конкретних періодів часу, протягом яких повноваження можуть бути пред'явлені утримувачем повноважень
targeting Information	дозволяє виконувати адресацію сертифіката атрибутів у конкретну групу серверів/ послуг
user Notice	дозволяє УА включати повідомлення, що повинне відображатися для утримувача, при пред'явленні повноважень, і/або для перевірника повноважень при застосуванні сертифіката атрибутів, що містить це розширення
acceptablePrivilege-Policies	використовується для обмеження наданих повноважень при використанні з конкретним набором політик повноважень

Закінчення табл. 7.7

1	2
noRevAvail	УА може використовувати це розширення для зазначення того, що для цього сертифіката атрибутів інформація стану не надається
so Identifier	вказує, що суб'єкт сертифікації може діяти як джерело повноважень (ДП)
attribute Descriptor	забезпечує один механізм, що може використовуватися ДП, щоб зробити доступними визначення атрибутів повноважень і пов'язаних правил домінування, які доступні для перевірника повноважень
roleSpecCertIdentified	може використовуватися УА як покажчик на сертифікат специфікації ролі, що містить надання повноважень ролі
basicAttConstraints	вказує, чи допускається наступне делегування повноважень
delegatedNameConstraints	вказує множину імен, у межах якого мають бути розміщені всі імена утримувачів у наступних сертифікатах на шляху делегування
acceptableCertPolicies	використовується для управління допустимими політиками сертифікації
authorityAttributeIdentifier	являє собою зворотний покажчик на сертифікат, у якому емітент сертифіката, що містить розширення, надав відповідні повноваження

Розширення сертифіката атрибутів

Поряд із визначенням самих розширень визначаються також правила для типів сертифіката, у яких може бути присутнє розширення. За винятком розширення ідентифікатора ДП, кожне із розширень, що може бути включене в сертифікат відкритого ключа, повинно вводитися тільки якщо такий сертифікат відкритого ключа є сертифікатом, що привласнює повноваження своєму суб'єктові (тобто має бути присутнє розширення subjectDirectoryAttributes). Якщо яке-небудь з цих розширень присутнє в сертифікаті відкритого ключа, то таке розширення застосовується до всіх повноважень, присутніх у розширенні subjectDirectoryAttributes.

Наступні розширення сертифікатів може бути включено до сертифікатів з метою управління повноваженнями.

Розширення «Специфікація часу» (time Specification). Розширення «Специфікація часу» може використовуватись УА для обмеження конкретних періодів часу, протягом яких повноваження, встановлені в сертифікаті, що містить таке розширення, можуть бути пред'явлені утримувачем повноважень. Наприклад, УА може випускати сертифікат, що надає повноваження, яке може бути пред'явлено тільки з понеділка до п'ятниці з 9:00 до 17:00 години.

Розширення «Інформація адресації» (targeting Information). Розширення «Інформація адресації» дозволяє виконувати адресацію сертифіката атрибутів у

конкретну групу серверів/ послуг. Сертифікат атрибутів, що містить це розширення, повинен використовуватися тільки в зазначених серверах/ послугах.

Розширення «Повідомлення користувача» (user Notice). Розширення «Повідомлення користувача» дозволяє УА включати повідомлення, що повинне відображатися для утримувачів, при пред'явленні їхніх повноважень та/або для перевірника повноважень при застосуванні сертифіката атрибутів, що містить це розширення.

Розширення «Допустимі політики застосування повноважень» (acceptablePrivilegePolicies). Поле «Допустимі політики застосування повноважень» використовується для обмеження пред'явлення наданих повноважень при використанні з конкретним набором політик повноважень.

Розширення «Пункти розповсюдження ССС». Це розширення ідентифікує способи отримання інформації ССС. Воно має бути некритичним. Розширення складається з трьох вибіркових полів: пункта розповсюдження, причин та емітента ССС.

Розширення «Відсутність інформації скасування» (noRevAvail). У деяких інформаційних середовищах (наприклад, де сертифікати атрибутів випущено з дуже короткими строками чинності), може не вимагатися скасування сертифікатів. УА може використовувати це розширення для зазначення того, що для цього сертифіката атрибутів інформація стану скасування не надається.

Розширення «Ідентифікатор ДП» (so Identifier). Розширення «Ідентифікатор ДП» вказує, що суб'єкт сертифікації може діяти як ДП з метою управління повноваженнями. Суб'єкт сертифікації може визначати атрибути, що привласнюють повноваження. Також випускати сертифікати дескрипторів атрибутів для таких атрибутів і використовувати особистий ключ, який відповідає сертифікованому відкритому ключу, для випуску сертифікатів, що присвоюють повноваження утримувачам. Сертифікати, які випускаються, можуть бути сертифікатами атрибутів або сертифікатами відкритого ключа з розширенням subjectDirectoryAttributes, що містить повноваження.

У деяких середовищах не потрібно таке розширення і можуть використовуватися інші механізми для визначення об'єктів, що діють у ролі ДП. Це розширення потрібно тільки в середовищах, де необхідний суворий централізований контроль з боку УА для управління об'єктами, які діють як ДП.

Розширення «Дескриптор атрибута» (attribute Descriptor). Визначення атрибуту повноважень і правил домінування, які керують наступним делегуванням таких повноважень необхідно перевірнику повноважень для гарантії того, що санкціонування було зроблено правильно. Ці визначення й правила можуть бути надані перевірнику повноважень різними способами (наприклад, вони можуть бути локально задані).

Це розширення забезпечує механізм, що може використовуватися ДП для того, щоб зробити доступними визначення атрибутів повноважень і пов'язаних правил домінування, які доступні перевірнику повноважень. Сертифікат атрибутів, що містить це розширення, називається *сертифікатом дескрипторів атрибута* і є особливим типом сертифіката атрибутів.

Сертифікат дескрипторів атрибутів, хоча й є синтаксично ідентичним AttributeCertificate, але він також:

– містить у своєму полі attributes порожній SEQUENCE;

– є самовипущеним сертифікатом (тобто емітент і утримувач є одним і тим самим об'єктом);

– включає розширення дескриптора атрибутів.

Розширення «Ідентифікатор сертифіката специфікації ролі» (roleSpecCertIdentified). Це розширення може використатися УА як вказівник на сертифікат специфікації ролі, що містить надання повноважень ролі. Воно може бути присутнім у сертифікаті встановлення ролі (тобто сертифікаті, що містить атрибут role).

Коли перевірник повноважень має справу із сертифікатом надання ролей, він повинен одержати набір повноважень такої ролі з метою визначення, успішно чи ні пройшла перевірка. Якщо повноваження присвоєні ролі в сертифікаті специфікації ролі, то це поле може використовуватися для знаходження такого сертифіката.

Розширення «Базові обмеження атрибутів» (basicAttConstraints). Це поле вказує на те, чи допускається наступне делегування повноважень, присвоєних у сертифікаті, що містить таке розширення. Якщо це так, то також можуть бути задані обмеження на довжину шляху делегування.

Розширення «Обмеження на делеговані імена» (delegatedNameConstraints). Поле «Обмеження на делеговані імена» вказує простір імен, у межах якого мають бути розміщені всі імена утримувачів у наступних сертифікатах на шляху делегування.

Розширення «Допустимі політики застосування сертифікатів» (acceptableCertPolicies). При делегуванні за допомогою сертифікатів атрибутів використовується поле «Допустимі політики застосування сертифікатів» для управління допустимими політиками сертифікації, відповідно до яких повинні випускатися сертифікати відкритого ключа для наступних утримувачів на шляху делегування. Шляхом перерахування множини політик у цьому полі УА вимагає, щоб наступні емітенти на шляху делегування делегували тільки повноваження, які містяться в ньому, власникам, які мають сертифікати відкритого ключа, випущені відповідно до однієї або більше перерахованих політик сертифікації. Перераховані тут політики не є політиками, відповідно до яких випускається сертифікат атрибутів, а політиками, відповідно до яких повинні випускатися допустимі сертифікати відкритого ключа для наступних утримувачів.

Розширення «Ідентифікатор атрибута повноважень» (authorityAttributeIdentifier). При делегуванні повноважень УА, що делегує повноваження, сам повинен мати щонайменше такі ж повноваження і повноваження для делегування таких повноважень. УА, що делегує повноваження іншому УА або кінцевому об'єкту, може розміщувати таке розширення в сертифікаті, який він випускає, УА або кінцевого об'єкта. Розширення являє собою зворотний вказівник на сертифікат, у якому емітент сертифіката, що містить розширення, присвоїв відповідні повноваження. Розширення може використовуватися перевірником повноважень для гарантії того, що випускаючий УА мав досить повноважень для можливості делегування повноважень утримувачу сертифіката, який містить таке розширення.

7.6. ІНФРАСТРУКТУРА УПРАВЛІННЯ ПОВНОВАЖЕННЯМИ

Інфраструктура управління повноваженнями створюється для підтримки життєвого циклу сертифікатів атрибутів. Вона є системою, що схожа на інфраструктуру відкритих ключів. Основними схожими елементами є подання даних у вигляді сертифіката та застосування асиметричної пари ключів – особистого та відкритого.

7.6.1. Загальна модель управління повноваженнями

Згідно [13, 14] до складу інфраструктури управління повноваженнями входять три об'єкти: об'єкт, який захищають, пред'явник повноважень і перевірник повноважень. У подальшому будемо вважати, що об'єкт є ресурсом, що захищається. Першому об'єкту характерні об'єктні методи, які можуть застосовуватись відносно нього. Наприклад, об'єктом може бути брандмауер, який може виконати дію «Дозволити вхід», або об'єктом може бути файл у файловій системі, відносно якого можуть виконуватись такі операції як «Читати», «Записувати», «Виконати» тощо.

Пред'явником повноважень є об'єкт, який володіє конкретними повноваженнями і пред'являє свої повноваження для конкретного контексту їх використання.

Перевірником повноважень є об'єкт, який визначає, чи вистачає пред'явлених повноважень для заданого контексту використання інформації чи ресурсів.

Прийняття рішення відносно альтернативи – успіху/ невдачі, що виконується перевірником повноважень, залежить від таких складових:

– власних повноважень пред'явника;
– політики застосування повноважень;
– поточних змінних оточення;
– чутливості об'єктного методу до змін.

Повноваження утримувача повноважень відбивають ступінь довіри до такого утримувача, яка виражається емітентом у тому, що утримувач повноважень буде пов'язаний з тими аспектами політики, які не впроваджені технічними засобами. Ці повноваження містяться в сертифікаті(ах) атрибутів утримувача повноважень (або розширенні subjectDirectoryAttributes сертифіката відкритого ключа), який може бути представлений перевірнику повноважень у запиті на доступ або може бути розповсюджений за допомогою деяких інших засобів, таких як Каталог.

Політика застосування повноважень задає ступінь повноважень, яка вважається достатньою для заданої чутливості дії або контексту використання. Щодо політики повноважень, повинна забезпечуватися її цілісність та автентичність на етапі дійсності. У концепції управління атрибутами безпосередньо сама політика також може бути об'єктом поширення в середовищі. Взагалі існує два рішення – щоб політика взагалі не передавалася, а просто задавалася й завжди локально підтримувалася в середовищі перевірника повноважень. З іншого боку, у зв'язку з тим, що деякі політики є «універсальними» і повинні передаватися та бути відомими будь-якому об'єкту в системі.

У цілому, політика повноважень повинна задавати поріг допустимості для заданого набору повноважень, тобто визначати, коли перевірник повноважень може робити висновок, що заданий набір повноважень є «достатнім» для того, щоб він міг надавати доступ до об'єкта, ресурсів, додатків тощо.

Змінні оточення повинні містити в собі ті аспекти політики, які необхідні для визначення успіху/ невдачі (наприклад, час дня або поточний фінансовий баланс) і доступні для перевірника повноваження через деякі локальні засоби. Формування змінних оточення є повністю локальним.

Чутливість дії може відображати атрибути документа або запиту, що обробляється, наприклад, конфіденційність змісту документа.

Зв'язування повноважень з об'єктом забезпечується Уповноваженим з атрибутів через підписану з використанням особистого ключа цифрового підпису структуру даних, яка називається *сертифікатом атрибутів*, або через сертифікат відкритого ключа, який містить розширення, явно визначене для такої мети.

Немає необхідності встановлення будь-якого зв'язку між перевірником повноважень та будь-яким конкретним УА. Оскільки утримувачі повноважень можуть мати сертифікати атрибутів, випущені для них багатьма УА, то перевірники повноважень можуть використовувати сертифікати, що випущені різними УА. Вони також не повинні для надання доступу до конкретного ресурсу бути ієрархічно пов'язані один з одним.

Додаток або система, що використовує сертифікати атрибутів, повинна перевіряти дійсність кожного сертифіката атрибутів перед кожним його застосуванням.

Згідно [13, 14] Уповноважений з атрибутів (УА) і Уповноважений на сертифікацію (УС) логічно (і, у багатьох випадках, фізично) є повністю незалежними. Таким чином, уся ІВК, включаючи УС, може існувати й діяти до встановлення ІУП. УС, хоча він і є джерелом повноважень для встановлення достовірності, він не стає автоматично джерелом повноважень відносно повноважень через атрибути.

Джерело повноважень (ДП) є об'єктом, якому перевірник повноважень довіряє як об'єкту з повною відповідальністю за надання певного переліку повноважень. Ресурси можуть обмежувати повноваження ДП шляхом призначення певних ДП для конкретних функцій (наприклад, одного для повноважень читання, а іншого для повноважень запису). У той же час ДП саме є УА, тому що воно може випускати сертифікати іншим об'єктам. По суті, ДП є аналогом «кореневого УА» в ІВК у тому розумінні, що перевірник повноважень довіряє сертифікатам, які підписані ДП. У деяких критичних середовищах може також існувати для УА необхідність здійснювати жорсткий контроль над об'єктами, які можуть діяти як ДП. Таким чином, система УП є достатньо гнучкою і може задовольняти вимогам багатьох типів інформаційних середовищ.

7.6.2. Застосування інфраструктур управління повноваженнями

Згідно положень [13, 14] ІУП можуть застосовуватися для надання різноманітних послуг. Основним об'єктом, якому безпосередньо присвоюються всі повноваження, є окремий об'єкт – УА. По суті, він є єдиним джерелом повноважень. При цьому до основних повноважень необхідно віднести такі:

1) присвоєння суб'єктам і об'єктам різних ролей з певними повноваженнями;

2) підтримка механізмів делегування повноважень;

3) підтримка та узгоджена взаємодія УА з УС;

4) виконання делегування повноважень по ланцюгу взаємодії;

5) визначення та обмеження на шляхи делегування повноважень тощо.

У більшості критичних технологій власники повноважень можуть мати право виконувати функції з різними фіксованими обмеженнями та можливостями. У цьому випадку говорять про необхідність і порядок підтримки властивостей

визначених ролей. Для цього УА випускає сертифікати, які присвоюють відповідним суб'єктам і об'єктам різні ролі. Як правило, повноваження, що пов'язані з ролями, надаються таким об'єктам і суб'єктам (індивідуумам) неявно.

Іншою послугою, яку повинен підтримувати УА, є підтримка делегування повноважень. Якщо в ІУП застосовується механізм делегування повноважень, то джерело повноважень надає повноваження об'єкту, якому дозволяється діяти як УА, і він може далі делегувати свої повноваження. Причому делегування повноважень може здійснюватися через один чи декілька проміжних УА, але доти, поки вони не будуть зрештою присвоєні кінцевому об'єкту. Кінцевий об'єкт не може далі делегувати такі повноваження. При цьому кожен проміжний УА може також діяти і як пред'явник повноважень.

Щодо особливостей організаційних та організаційно-технічних структур виникає необхідність, щоб один і той самий об'єкт міг діяти одночасно як УС та УА. Така подвійна логічна роль одного й того самого фізичного об'єкта може мати місце, якщо повноваження передаються в розширенні ***subjectDirectoryAttributes*** сертифіката відкритого ключа. В інших інформаційних середовищах УС і УА можуть діяти як окремі фізичні об'єкти. У такому випадку повноваження можуть надаватись тільки з використанням сертифікатів атрибутів відкритого ключа.

У локальних та корпоративних мережах виникає потреба в делегуванні повноважень. Сутність цього механізму може бути описана у вигляді моделі делегування повноважень. У такій моделі необхідна присутність щонайменше чотирьох взаємодіючих сторін: перевірник повноважень, джерело повноважень, уповноважений атрибутів та пред'явник повноважень (рис. 7.1).

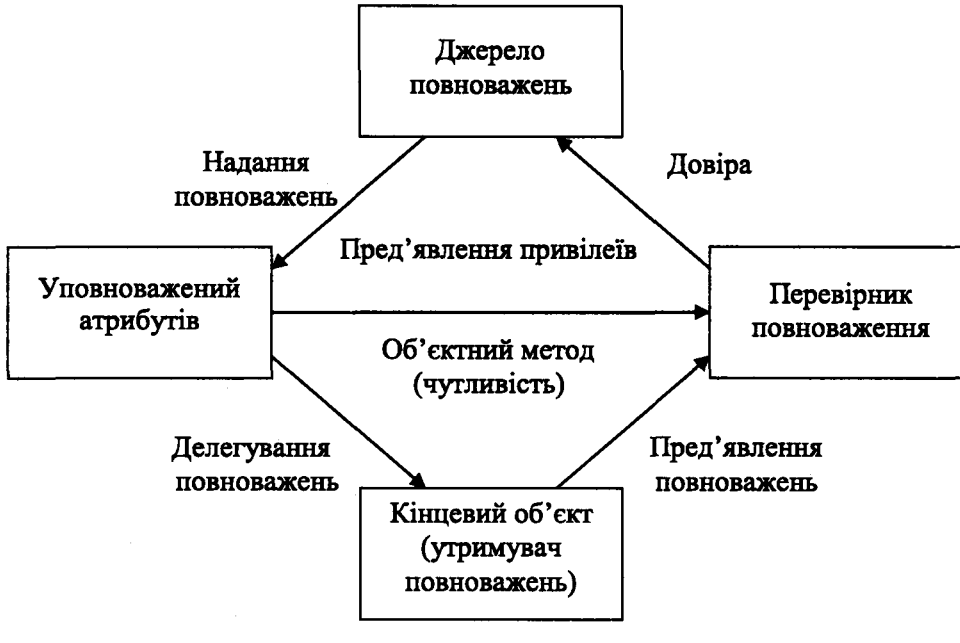

Рис. 7.1. Модель делегування повноважень

Якщо в системах не використовується делегування повноважень, то в цьому випадку джерело повноважень є початковим емітентом сертифікатів, який має право присвоювати повноваження певним утримувачам повноважень. За таких умов ДП санкціонує утримувача повноважень для дії як УА, надалі ці повноваження можуть бути делеговані іншим об'єктам способом випуску сертифікатів атрибутів. За таких умов ДП може накладати обмеження на делегування. Наприклад, може бути зроблено обмеження на довжину шляху, обмеження простору імен у межах делегування тощо. Кожний з таких проміжних УА може в сертифікатах, які випускаються ним для подальших утримувачів повноважень, санкціонувати подальше делегування, яке буде виконуватися такими утримувачами. Вони можуть також діяти як і УА. При цьому найбільш універсальне обмеження з делегування повноважень полягає в тому, що ніякий УА не може делегувати більше повноважень, ніж він має сам. Пред'явник повноважень може також додатково обмежувати повноваження УА нижніх рівнів.

При використанні делегування перевірник повноважень довіряє ДП делегувати деякі або всі свої повноваження утримувачам, деякі з яких можуть делегувати далі частину або всі свої повноваження.

У моделі, що розглядається, перевірник повноважень довіряє ДП як уповноваженому центру для заданого набору повноважень щодо інформації та ресурсів. Якщо сертифікат пред'явника повноважень не випущений таким ДП, то перевірник повноважень повинен мати можливість знайти шлях делегування сертифікатів від сертифіката пред'явника повноважень до сертифіката, який випустило ДП. Перевірка дійсності такого шляху делегування включає перевірку того, щоб кожен УА мав досить повноважень і був санкціонований для делегування таких повноважень.

У випадку, коли повноваження передаються за допомогою сертифікатів атрибутів, шлях делегування відрізняється від шляху перевірки дійсності сертифіката, який використовується для перевірки дійсності сертифікатів відкритого ключа об'єктів, що беруть участь у процесі делегування. Однак якість автентичності, що забезпечується процесом перевірки дійсності сертифіката відкритого ключа, повинна бути порівняною з чутливістю дії відносно повноважень, які захищаються.

Також шлях делегування повинен складатися або повністю із сертифікатів атрибутів, або повністю із сертифікатів відкритого ключа. Об'єкт, що отримує свої повноваження в сертифікаті атрибутів, може виконувати делегування тільки (якщо це санкціоновано) шляхом видачі наступних сертифікатів атрибутів. Аналогічно, черговий утримувач, що одержує свої повноваження в сертифікаті відкритого ключа, може делегувати повноваження (якщо це санкціоновано) тільки шляхом видачі наступних сертифікатів відкритого ключа. Але взагалі, тільки УА можуть делегувати повноваження. Кінцеві об'єкти делегувати повноваження не мають можливостей.

7.6.3. Модель ролей утримувачів

Серед концептуальних положень інфраструктури УА необхідно виділити використання способу непрямого надання суб'єктам повноважень. Присвоєння однієї або декількох ролей здійснюється через атрибут ролей, який міститься у

випущеному сертифікаті. При такому підході ті чи інші повноваження можна розглядати як спеціальні, що надаються через сертифікати специфікації ролі. Причому сертифікатами призначення ролі можуть бути як сертифікати атрибутів, так і сертифікати відкритих ключів. Сертифікатами специфікації ролі можуть бути тільки сертифікати атрибутів. Якщо сертифікати призначення ролі не використовуються, то присвоєння повноважень ролі може робитися в інший спосіб. Наприклад, присвоєння повноважень ролі можуть задаватися локально.

При присвоєнні повноважень локально допустимим є таке:

– будь-який УА може визначати будь-яке число ролей;

– сама роль і члени ролі можуть визначатися й адмініструватися окремо, навіть різними УА;

– приналежність ролі, як і будь-які інші повноваження, можуть бути делеговані;

– ролям і можливостям їх виконання можуть бути присвоєні будь-які терміни дії.

Якщо сертифікат призначення ролі є сертифікатом атрибутів, то атрибут ***role*** міститься в компоненті ***attributes*** сертифіката атрибутів. Якщо сертифікат призначення ролі є сертифікатом відкритого ключа, то атрибут ***role*** міститься у розширенні ***subjectDirectoryAttributes***. В останньому випадку будь-які додаткові повноваження, які містяться у сертифікаті відкритого ключа, є повноваженнями, які прямо надаються суб'єктові сертифіката, а не повноваженнями, що пов'язані з роллю.

Таким чином, пред'явник повноважень може представляти сертифікат призначення ролі перевірнику повноважень тільки для того, щоб продемонструвати те, що пред'явник повноважень має конкретну роль (наприклад, «менеджер», або «покупець»). Перевірник повноважень дізнається про повноваження, що пов'язані із затвердженою роллю, з метою ухвалення рішення перевірки – успіх чи невдача. Для таких цілей може бути використаний сертифікат специфікації ролі.

Сертифікат специфікації ролі не може бути делегований будь-якому іншому об'єкту. Емітент сертифіката призначення ролі, може бути незалежним від емітента сертифіката специфікації ролі, і адміністрування таких сертифікатів (завершення дії, скасування і т.д.) може виконуватися окремо. Сертифікат атрибутів або сертифікат відкритого ключа може бути сертифікатом призначення ролі, а також містити надання інших повноважень безпосередньо тому самому суб'єкту. Але завжди сертифікат специфікації ролі повинен бути окремим сертифікатом.

7.6.4. Основні способи надання повноваження у сертифікатах атрибутів

В інфраструктурі УА об'єкти можуть одержувати повноваження двома способами [13, 14].

УА може односторонньо присвоювати повноваження об'єкту способом виготовлення сертифіката атрибутів як за власною ініціативою, так і за запитом деякої

третьої сторони. Цей сертифікат може бути збережений у загальнодоступному сховищі і згодом може бути оброблений для схвалення рішення щодо санкціонування одним або декількома перевірниками повноважень. Указане може здійснюватися як без знання про це, так і без явних дій об'єкта.

Об'єкт може запитувати повноваження у певного УА. Як тільки сертифікат буде виготовлено, він може бути виданий тільки об'єкту, що його запитав. У подальшому власник явно надає його при запиті доступу до певних захищених ресурсів.

При обох способах УА для гарантії того, що об'єкту насправді присвоєні ці повноваження, повинен ретельно виконувати свої обов'язки. Для цього можуть бути використані деякі незалежні механізми, наприклад, аналогічні сертифікації автентичності зв'язаної пари ключів за допомогою УС.

ІУП на базі сертифікатів атрибутів можуть застосовуватись при виконанні таких вимог:

1) за надання конкретних повноважень утримувачу відповідає інший об'єкт, ніж той, що випускає сертифікати відкритого ключа тому ж суб'єктові;

2) існує ряд атрибутів повноважень, які надаються утримувачу різними уповноваженими;

3) строк дії повноважень відрізняється від строку чинності сертифіката відкритого ключа утримувача, зазвичай він набагато коротший;

4) повноваження є дійсними тільки протягом певних інтервалів часу, які є асинхронними зі строком дії відкритого ключа користувача або дійсністю інших повноважень.

7.6.5. Особливості повноважень, що задаються в сертифікатах відкритого ключа

У деяких інформаційних середовищах повноваження надаються за допомогою УС відкритих ключів. Такі повноваження можуть бути указані безпосередньо в сертифікатах відкритого ключа, що забезпечує багаторазове використання значної частини вже встановленої інфраструктури. У таких випадках повноваження включаються в розширення ***subjectDirectoryAttributes*** сертифіката відкритого ключа.

Цей механізм може застосовуватись в інформаційно-телекомунікаційних системах при виконанні таких умов:

1) якщо один і той самий фізичний об'єкт діє і як УС, і як УА;

2) якщо строк дії повноважень повинен бути узгоджений зі строком дії відкритого ключа, який міститься в сертифікаті;

3) якщо делегування повноважень не допускається;

4) якщо делегування допускається, але для будь-якого одного делегування всі повноваження в сертифікаті у розширенні ***subjectDirectoryAttributes*** мають ті самі параметри делегування. При цьому всі розширення, що пов'язані з делегуванням, однаково можуть застосовуватись до всіх повноважень у сертифікаті.

Висновки та рекомендації

Аналіз основних положень сертифікації атрибутів дозволяє зробити висновок щодо актуальності проблеми управління повноваженнями в системах електронного документообігу.

Враховуючи те, що в Україні створення інфраструктури системи ЕЦП тільки починається, вважаємо доцільним розглядати питання створення ІУП на базі цієї системи. Принаймні алгоритмічно та технічно закладати можливість реалізації функцій управління сертифікатами атрибутів центрами сертифікації. Тому доцільно прийняти й цю частину міжнародного стандарту.

Для інших інформаційних, автоматизованих і телекомунікаційних систем також доцільно розглядати можливість застосування технології управління повноваженнями, наприклад, для реалізації рольових моделей управління доступом. Питання використання технологій сертифікації атрибутів можуть розглядатися при проектуванні автоматизованих систем управління підприємством, технологічними процесами, у банківських інформаційних і платіжних системах тощо.