УДК 538.3+537.86

АНАЛІЗ ВЛАСТИВОСТЕЙ ТА ОБЛАСТЕЙ ЗАСТОСУВАННЯ ЦИФРОВИХ ПІДПИСІВ СТАНДАРТУ ISO/IEC 9796 – 3 :2006(ISO/IEC 15946-4:2003)

Ю.І. ГОРБЕНКО, А.А. ШЕВЧУК

Розглядаються алгоритми стандарту ЕЦП ISO/IEC 9796 – 3:2006 з відновленням повідомлення, робиться їх порівняльний аналіз, а також аналіз алгоритмів з точки особливостей застосування.

The paper considers algorithms of the standard of the electronic digital signature of ISO/IEC 9796 – 3:2006 with message update, performs their comparative analysis as well as analysis of algorithms from the point of view of usage peculiarities.

ВСТУП

У сучасних автоматизованих системах управління, комп'ютерних системах і мережах, інформаційних та телекомунікаційних системах, висуваються високі вимоги до забезпечення цілісності, автентичності (справжності) та доступності інформації на всіх етапах її життєвого циклу, а також надання послуг неспростовності [1-5]. Досвід застосування і проведені дослідження підтвердили, що ці високі вимоги, особливо з реалізації функції причетності (неспростовності), можуть бути забезпечені тільки за рахунок застосування (електронного) цифрового підпису (ЕЦП). Цифровий підпис (ЕЦП), по суті, являє собою додані до інформації дані, обчислені за допомогою криптографічного перетворення інформації, яка захищається, і спирається на параметри, при наявності яких можна упевнитися в цілісності інформації і справжності інформації та її джерела, а так само забезпечити захист від підробки з боку отримувача.

На нинішній час широке розповсюдження знайшли ЕЦП з додаванням та відновленням повідомлення [10, 11], що грунтуються на використанні асиметричних криптографічних перетворень. В ЕЦП з відновленням частина або повне повідомлення можуть бути відновленими з цифрового підпису, тобто для перевірки цифрового підпису необхідно знати тільки цифровий підпис та, можливо, сертифікат відкритого ключа. В ЕЦП з додаванням – цифровий підпис приєднується до повідомлення та зберігається і передається з ним, а для перевірки ЕЦП потрібно обов'язково мати сертифікат відкритого ключа.

Теоретичне обґрунтування та практичні дослідження ЕЦП з відновленням повідомлення були виконані, у порівнянні з ЕЦП з доповненням, пізніше. В значній мірі вони появились, коли виникла необхідність в ЕЦП для коротких повідомлень. Це напрям був успішно розвинутий в роботах [6-11]. Як наслідок в 2003 році був прийнятий міжнародний стандарт ISO/IEC 15946-4 [10]. В нього було включено 5 незалежних алгоритмів ЕЦП з відновленням повідомлення, криптографічні перетворення в якому базуються на еліптичних кривих. В подальшому цей стандарт було удосконалено і він був прийнятий в 2006 році як ISO/IEC 9796-3 [11] на заміну існуючому. Додатково в нього був включений алгоритм ЕЦП, що грунтується на перетворенні в полі Галуа.

З прийняттям стандарту в практичному аспекті виникла проблема оцінки криптографічної стійкості та практичного застосування, перше за порівняльного аналізу.

Метою цієї статі є аналіз алгоритмів стандарнту ЕЦП як ISO/IEC 9796 – 3 з відновленням повідомлення, їх порівняння, оцінка захищенності та колізійної стійкості, а також аналіз алгоритмів з точки зору застосування.

Особливістю схеми підпису із відновленням повідомлення є те, що в ній висувають правила використання функції формування доповнення. Для повної перевірки, абонент цифрового підпису має мати повну та неушкоджену збитковість повідомлення. Також схеми з відновленням повідомлення не висувають обмежень у використанні функції формування збитковості. Наприклад, частка повідомлення, що відновлюються, могла б мати чітко визначений розмір у 80 бітів, але в цьому випадку нівелюються усі переваги схеми. До того ж типові повідомлення належать до якоїсь групи значень, тобто мають природну збитковість тощо.

Таким чином, схеми із відновленням повідомлення доцільно використовувати у інформаційних системах та протоколах з чітко визначеними повідомленнями. Це є принциповою особливістю з точки зору їх застосувань.

1. ЗАГАЛЬНА МОДЕЛЬ ЕЦП З ВІДНОВЛЕННЯМ ПОВІДОМЛЕНЬ

У стандарті ISO/IEC 9796–3:2006 визначено 6 різних методів і на їх основі механізмів цифрових підписів з відновленням повідомлення. Механізми цифрового підпису, що представлені в стандарті, мають назву «механізм підпису з відновленням». Вони забезпечують повне або часткове відновлення повідомлення. В подальшому ЕЦП з відновленням повідомлення будемо розглядати через специфікацію процесів обчислення(генерації) параметрів, обчислення (генерації) підпису та перевіряння підпису.

Процес обчислення параметрів. Параметри можна розділити на доменні параметри та параметри користувача. Доменні параметри складаються з параметрів для визначення скінченного

поля, параметрів для визначення еліптичної кривої над скінченним полем та іншої відкритої інформації, що є спільною, відомою та доступною для усіх об'єктів у межах домену.

Мають бути визначені такі параметри:

– ідентифікатор схеми цифрового підпису, що використовується;

– геш-функція (Hash);

– процедури генерації параметрів користувача.

Кожен об'єкт має свої власні відкриті та особисті параметри. До параметрів користувача об'єкту A відносяться:

– особистий ключ підпису x_A;

– відкритий ключ перевірки Y_A;

– іншу інформацію (не обов'язково), що є специфічною для об'єкту A, яка використовується у процесі генерації підпису і/або у процесі перевіряння.

Параметри і ключі, що застосовуються, повинні бути дійсними.

Процес обчислення підпису. До обчислення ЕЦП повинні бути введеними:

– доменні параметри;

– параметри користувача підписувача A, зокрема особистий ключ підпису x_A;

– повідомлення M, що повинне бути підписане.

Незалежно від схеми, процес обчислення підпису складається з таких етапів (процедур):

– розщеплення повідомлення на складові;

– обчислення надлишковості, або обчислення геш-значення повідомлення (за вибором користувача);

– обчислення в групі точок еліптичної кривої;

– обчислення за модулем порядку групи базової точки G;

– форматування підписаного повідомлення.

Вихідними даними процесу генерації підпису є пара цілих чисел (r,s), що є цифровим підписом повідомлення M відповідного об'єкту, наприклад A.

Процес перевіряння підпису.

Для перевіряння підпису потрібні такі параметри та ключі:

– дійсні доменні параметри;

– відкритий ключ перевірки ЕЦП Y_A користувача А;

– повідомлення M'_{clr}, що не відновлюється (якщо воно є);

– ЕЦП для повідомлення M, що представлений двома цілими числами - r' та s'.

– Для всіх схем, що представлені в стандарті, процес перевіряння підпису складається з окремих або всіх наступних етапів (процедур):

– перевіряння розміру підпису;

– відновлення попереднього підпису та вхідних даних;

– відновлення повідомлення;

– перевірка надлишковості, або обчислення геш-значення повідомлення (з правом вибору);

– обчислення за модулем порядку групи базової точки G;

– обчислення в групі точок еліптичної кривої;

– перевірка підпису.

Якщо усі процедури пройшли успішно, перевірник приймає підпис, інакше підпис відхиляється.

Для реалізації кожного з механізмів цифрового підпису, що визначені у стандарті, повинні бути обраними такі доменні параметри конкретної схеми цифрового підпису:

– скінченне поле Галуа $F(q)$;

– еліптичну криву E над полем $F(q)$, яка має унікальну циклічну підгрупу простого порядку n;

– точку G на еліптичній кривій E простого порядку n.

У якості надлишковості користувачі повинні обрати одну з таких надлишковостей:.

– природна надлишковість;

– додана надлишковість;

– обидва типи надлишковості.

Повідомлення з природною надлишковістю означає, що повідомлення за своєю природою включає надлишковість або те, що надлишковість повідомлення може бути побічно перевірена деякими застосуваннями.

Повідомлення з доданою надлишковістю може бути сформовано із застосуванням повідомлення або повідомлення, що відновлюється. Природна або додана надлишковість можуть бути чимось таким, що узгоджено взаємодіючими сторонами і може бути перевірено ними.

Повна надлишковість, яка складається з природної надлишковості і доданої надлишковості, має бути більше деякого мінімального значення, що визначено застосуванням у відповідному додатку.

Якщо використовується додана надлишковість, то типи надлишковості мають бути віднесені до таких фіксованих значень:

– коротка надлишковість;

– довга надлишковість.

Коротка надлишковість має використовуватися у випадках, коли з підпису можна відновити усе повідомлення.

Довга надлишковість має використовуватися у випадках, коли з підпису можна відновити лише частину повідомлення.

Довжина короткої надлишковості і довгої надлишковості, len_1 і len_2 повинна бути зафіксована. Причому якщо довжина (у бітах) повідомлення не більша ніж $len\_n - len\_1 - 1$, то з підпису можна відновити усе повідомлення і використовується коротка надлишковість. Якщо довжина (у бітах) повідомлення більша ніж $len\_n - len\_1 - 1$, то частина повідомлення, що відновлюється не може бути більшою ніж $len\_n - len\_2 - 1$ бітів, а в цілому використовується довга надлишковість.

Типовими значеннями len_1 є 64 або 80. Типові значення len_2 змінюються у діапазоні від

136 до 168. Також припускається встановлювати $len\_1 = len\_2$.

Перелік функцій та процедур

Схеми підпису, що розглядаються, дозволяють відновлювати повідомлення у змісті відновлення деяких з даних, що використовуються при обчисленні підпису. Вони є частиною процедури перевіряння підпису.

Схема підпису складається з таких складових (етапів):

– генерування (обчислення) доменних параметрів;

– генерування (обчислення) асиметричної ключової пари ЦП;

– формування ключа сеансу і попереднього підпису;

– обчислення першої частини підпису;

– обчислення другої частини підпису;

– відновлення попереднього підпису;

– відновлення вхідних даних.

Генерування (обчислення) асиметричної ключової пари ЦП

Для обчислення асиметричної ключової пари ЦП може застосовуватись один з двох наступних методів.

Метод генерація ключа I

Для заданого дійсного набору доменних параметрів еліптичної кривої асиметрична пара, тобто особистий ключ підпису x_A та відповідний йому відкритий ключ Y_A, повинні генеруватись таким чином.

1. Обрати (згенерувати) з множини $[2, n-2]$ випадкове або псевдовипадкове ціле число x_A, тобто особистий ключ. Ключ x_A має бути захищеним від несанкціонованого розкриття і бути непередбачуваним.

2. Обчислити відкритий ключ як точку еліптичної кривої $Y_A = x_A G$.

3. Асиметричною ключовою парою є пара (Y_A, x_A).

В подальшому для уніфікації позначення та одноманітного представлення приймемо позначення $P := G$ і $Q := Y_A$.

Метод генерації ключа II

Для другого методу та заданого дійсного набору доменних параметрів еліптичної кривої асиметрична ключова пара, тобто особистий ключ підпису x_A та відповідний йому відкритий ключ Y_A, повинні генеруватись таким чином.

1. Обрати (згенерувати) з множини $[2, n-2]$ випадкове або псевдовипадкове ціле число e та обчислити ціле число x_A у інтервалі $[2, n-2]$ таке, що

$$x_A e = 1 \bmod n. \qquad (1)$$

Обидва цілі числа x_A і e мають бути захищеними від несанкціонованого розкриття та непередбачуваними.

2. Обчислити точку еліптичної кривої $Y_A = eG$.

3. Ключовою парою є (Y_A, x_A), де як і раніше – Y_A відкритий ключ перевірки, а x_A – особистий ключ ЦП.

Перед використанням відкритого ключа перевірки перевірник повинен мати гарантію його дійсності та володіння.

Генерування (обчислення) ключа сеансу і попереднього ЦП.

Перед кожним обчисленням підпису об'єкт, що підписує, повинен мати доступ до нового, захищеного значення ключа сеансу та його використати.

Ключем сеансу є ціле число k таке, що $1 < k < n-1$. Реалізація схеми підпису має забезпечувати задоволення двох наступних вимог:

– ключі сеансу k, що використані, не повинні ніколи розкриватися, і одразу після використання, кожен ключ повинен бути знищеним;

– ключі сеансу повинні генеруватися таким чином, щоб імовірність використання одного і того ж ключа сеансу при формуванні підписів для двох різних повідомлень була зневажливо малою.

Попередній підпис обчислюється як функція ключа сеансу. Необхідно мати на увазі, що розкриття ключа сеансу (після його використання) може призвести до компрометації особистого ключа підпису. Але, оскільки кожен із ключів сеансу, що вже використаний, більше не використовуються ні підписувачем, ні перевірником, то він може(повинен) бути знищеним відразу після обчислення підпису.

Обчислення першої та другої частин підпису

Перша частина підпису обчислюється як функція попереднього підпису П і вхідних даних D, які є цілим числом, що залежить від повідомлення, причому $0 \le D < n$. Вона є цілим числом r таким, що $0 < r < n$.

Друга частина підпису виконується за допомогою особистого ключа підпису x_A, першої частини підпису r і ключа сеансу k. Вона є цілим числом s таким, що $0 \le s < n$, де n є порядок базової точки.

Відновлення попереднього підпису та вхідних даних

Відновлення попереднього підпису П здійснюється з використання відкритого ключа підписувача, наприклад, Y_A та самого підпису (r, s).

Відновлення вхідних даних d виконується з використанням заданої першої частини r підпису і відновленого попереднього підпису П'.

Обчислення цифрового підпису

Алгоритм обчислення цифрового підпису складається із наступних кроків:

– формування ключа сеансу та попереднього підпису;

– розщеплювання повідомлення;

– формування вхідних даних;

– обчислення цифрового підпису;

– форматування підписаного повідомлення.

Формування ключа сеансу та попереднього підпису

Попередній підпис є проміжним елементом даних, що виробляється впродовж процесу обчислення підпису в будь-якому захищеному від

вгадування механізмі підпису. Спершу у відповідності з вимогами, що викладені вище, формується ключ сеансу k. Значення ключа сеансу має бути доступним тільки процесу генерації підпису. Попередній підпис є відкритим елементом даних, тоді як значення ключа сеансу має бути доступне тільки процесу обчислення підпису та бути конфіденційним. Необхідно також відмітити, що ключі сеансу можуть формуватися, а відповідні попередні підписи можуть обчислюватися, автономно та зберігатися безпечним чином для використання при наступних підписах.

Розщеплення повідомлення

Повідомлення M розщеплюється на частину, що відновлюється M_{rec}, та частину повідомлення, що не відновлюється M_{clr}, при цьому відповідно len_rec і len_clr визначені як довжини частини, що відновлюється M_{rec} і частини, що не відновлюється M_{clr}. При чому додана надлишковість повідомлення M може бути розщеплена таким чином.

Якщо довжина (у бітах) len_M повідомлення M задовольняє умові $len\_M \le len\_n - len\_1 - 1$, то з підпису можна відновити усе повідомлення M. У цьому випадку вірними є рівняння: $M_{rec} = M$ і $len\_rec = len\_M$. Крім того, $len\_clr = 0$ і $len\_h = len\_1$.

Якщо довжина (у бітах) len_M повідомлення M задовольняє умові $len\_M > len\_n - len\_1 - 1$, то len_rec визначається як ціле число, таке що $len\_rec \le len\_n - len\_2 - 1$. Крайні ліві len_rec біти M складають частину повідомлення, що відновлюється M_{rec}. А крайні праві $len\_M - len\_rec$ біти складають частину повідомлення M, що не відновлюється M_{clr}, та вірними є рівняння $len\_rec = len\_M - len\_clr$ та $len\_h = len\_2$.

Формування вхідних даних

Вхідними даними функції введення даних є:

– len_rec, len_clr та попередній підпис – П;

– Частина повідомлення, що не відновлюється M_{clr} (необов'язково) і геш-токен, повідомлення, що відновлюється;

– M_{rec} з доданою надлишковістю, або повідомлення, що відновлюється, M_{rec} з природною надлишковістю.

Геш-токен є або самим геш-значенням або є геш-значенням, до якого праворуч приєднано ідентифікатор геш-функції, якщо геш-значення обчислюється засобом гешування повідомлення. Вибір щодо включення до геш-токену ідентифікатора геш-функції має обумовлюватися доменними параметрами. Виходом функції введення даних є значення D, яке після перетворення на ціле число знаходиться у діапазоні $0 \le D < n$.

Обчислення цифрового підпису

Цифрові підписи, що формуються, мають дві частини – r та s. Перша частина r обчислюється як функція вхідних даних D і попереднього підпису П. Друга частина s обчислюється як функція від першої частини підпису r, ключа сеансу k та особистого ключа підпису x_A (див. кожну схему нижче).

Форматування підписаного повідомлення

Для успішного відкриття і перевірки підписаного повідомлення необхідне знання довжини частини повідомлення, що відновлюється, та довжини частини повідомлення, що не відновлюється. Якщо ця інформація не міститься у доменних параметрах, то вона повинна бути включена до підписаного повідомлення.

Підписане повідомлення складається з таких елементів даних:

– частина повідомлення, що не відновлюється, та довжина повідомлення;

– перша частина r підпису;

– друга частина s підпису.

Перевіряння цифрового підпису

Алгоритм перевіряння підпису складається з таких кроків:

– відкриття підписаного повідомлення;

– перевіряння розміру підпису;

– відновлення попереднього підпису або вхідних даних;

– відновлення вхідних даних або повідомлення;

– повторне обчислення геш-токену (необов'язкове);

– зіставлення підпису та прийняття рішення.

Зіставлення підпису складається з :

– порівняння відновлених і повторно обчислених (обрізаних) геш-токенів;

– перевіряння надлишковості.

Відкриття підписаного повідомлення

На початку цього кроку перевірник повинен мати доступ до такої інформації:

– довжини різних частин повідомлення, що містяться у підписаному повідомленні;

– значень параметрів len_n, len_1 та len_2.

– Перевірник вибирає різні частини підписаного повідомлення у такому порядку:

– частина повідомлення, що не відновлюється;

– перша частина r' підпису;

– друга частина s' підпису.

Перевіряння розміру підпису

Перевірник має перевірити розмір частин підпису, тобто що $0 < r < n$ та що $0 \le s < n$, де n є порядок базової точки.

Відновлення попереднього підпису та вхідних даних

На початку цього кроку перевірник повинен мати доступ до такої інформації:

– відкриті параметри, які визначають схему підпису, що використовується;

– відкритий ключ перевірки Y_A об'єкту, що підписує.

Обчислення на цьому кроці залежать від схеми підпису, що використовується. Попередній підпис і вхідні дані відновлюються з підпису. Відновленими вхідними даними є D'.

Відновлення вхідних даних або повідомлення

Відновлені вхідні дані D' перетворюються на рядок бітів. Вхідними даними D' є:

– геш-токен;

– повідомлення, що відновлюється, з доданою надлишковістю;

– повідомлення, що відновлюється, з природною надлишковістю.

Повторне обчислення геш-значення

Перед обчисленням геш-значення спершу ідентифікується геш-функція, що використовувалась об'єктом при обчисленні підпису, наприклад, на основі ідентифікатора геш-функції з відновленого геш-значення. Потім геш-значення повторно обчислюється шляхом гешування повідомлення, що мається у перевірника.

Повторно обчислене геш-значення використовується для отримання повторно обчисленого геш - токену шляхом необов'язкового поєднання з ідентифікатором геш-функції.

Зіставлення підпису

Зіставлення підпису складається з:

– порівняння відновленого і повторно обчисленого (обрізаного) геш-значення;

– перевіряння надлишковості.

– Процедура порівняння складається з порівняння повторно обчисленого (обрізаного) геш-значення h'' та відновленого (обрізаного) геш-значення h'. Підпис має відхилятися, якщо ці два значення не рівні.

Процедура перевіряння надлишковості призначена для перевірки додаткової та/або природної надлишковості відновленого повідомлення. Підпис повинен відхилятися, якщо надлишковість не підтверджено.

В подальшому будемо застосовувати наступні функції перетворення та генерації маски.

BS2IP – примітив перетворення бітових рядків в цілі числа.

BS2OSP – примітив перетворення бітових рядків в октетові рядки.

EC2OSP – примітив перетворення еліптичної кривої в октетові рядки.

FE2IP – примітив перетворення елементів кінцевого поля в цілі числа.

FE2OSP – примітив перетворення елементів кінцнвого поля в октетові рядки.

I2BSP – примітив перетворення цілих чисел в бітові рядки.

I2OSP – примітив перетворення цілих чисел в октетові рядки.

MGF1, MGF2 – функції генерації маски.

OS2BSP – примітив перетвоення октетових рядків в бітові рядки.

OS2ECP – примітив перетворення октетових рядків в еліптичну криву.

OS2FEP – примітив перетворення октетових рядків в елементи кінцевого поля.

OS2IP – примітив перетворення октетових рядків в цілі числа.

2. ПІДПИС НІБЕРГА-РЮПЕЛЯ У СКІНЧЕННОМУ ПОЛІ (NYRBERG-RUEPPEL MESSAGE RECOVERY SIGNATURE)

У 1993 році Ніберг та Рюпель запропонували схему ЕЦП із відновленням повідомлення, що була заснована на проблемі дискретного логарифмування в полі Галуа [1, 9]. Вона дає перевагу при застосуванні із повідомленнями невеликого розміру. ЕЦП, розроблений за такою схемою, може ефективно використовуватися у інфраструктурах з відкритими ключами, у протоколах з малим розміром повідомлення, наприклад електронних магазинах тощо.

Основними складовими підпису Ніберга-Рюпеля є передпідпис, підпис та перевірка підпису.

Передпідпис формується шляхом піднесення до ступеню первісного елементу (доменного параметру) p

$$k = rand([1, n-1])$$
$$R = P^k$$
$$\Pi = FE2OSP_f(R)$$

Підпис виконується згідно до загальної схеми

$$\delta = OS2IP(d) \qquad \delta \in [0, n-1]$$
$$\pi = OS2IP(\Pi) \bmod n$$
$$\tilde{r} = (\delta + \pi) \bmod n$$
$$s = (k - x_A \tilde{r}) \bmod n$$
$$r = I2OSP(\tilde{r}, L(n))$$

Для перевірки підпису проводиться відновлення передпідпису та відновлення частини, що відновлюється, з r компоненти підпису.

Рішення щодо дійсності підпису приймається після аналізу збитковості відновленої частини.

$$\tilde{r}' = OS2IP(r')$$
$$R' = P^{s'} Q^{\tilde{r}'}$$
$$\Pi' = FE2OSP_F(R')$$
$$\pi' = OS2IP(\Pi') \bmod n$$
$$\delta' = (\tilde{r}' - \pi') \bmod n$$
$$d' = I2OSP(\delta', L_{dat})$$

3. ПІДПИС НІБЕРГА-РЮПЕЛЯ У ГРУПІ ТОЧОК ЕЛІПТИЧНИХ КРИВИХ (ELLIPTIC CURVE NYRBERG-RUEPPEL MESSAGE RECOVERY SIGNATURE)

Схема підпису

Підпис є класичною схемою ЕЦП Ніберга-Рюпеля у групі точок ЕК[10].

Передпідпис формується шляхом скалярного множення ключа сеансу k на базову точку (доменний параметр) P

$$k = rand([1, n-1])$$
$$R = kP$$
$$\Pi = EC2OSP_E(R, compressed)$$

Формування підпису проводиться згідно до класичної схеми Ніберга-Рюпеля, із використанням операції додавання у якості функції маскування.

$$\delta = OS2IP(d) \qquad \delta \in [0, n-1]$$
$$\pi = OS2IP(\Pi) \bmod n$$
$$\tilde{r} = (\delta + \pi) \bmod n$$
$$s = (k - x_A \tilde{r}) \bmod n$$
$$r = I2OSP(\tilde{r}, L(n))$$

Для перевірки підпису проводиться відновлення передпідпису та відновлення частини, що відновлюється, з r компоненти підпису.

Рішення щодо дійсності підпису приймається після аналізу збитковості відновленої частини.

$$\tilde{r}' = OS2IP(r')$$
$$R' = s'P + \tilde{r}'Q$$
$$\Pi = EC2OSP(R', compressed)$$
$$\pi' = OS2IP(\Pi') \bmod n$$
$$\delta' = (r' - \pi') \bmod n$$
$$d' = I2OSP(\delta', L_{dat})$$

4. ПІДПИС МІЯДЖІ У ГРУПІ ТОЧОК ЕЛІПТИЧНИХ КРИВИХ (ELLIPTIC CURVE MIYAJI MESSAGE RECOVERY SIGNATURE)

Спеціальні функції

Схема використовує функції маскування та гешування із однаковою довжиною та n бітовою довжиною результату.

$$Mask: \{0,1\}^{8^*} \to \{0,1\}^{8L(n)}$$
$$Hash: \{0,1\}^{8^*} \to \{0,1\}^{8L_{Hash}}$$

Передпідпис формується скалярним множенням ключа сеансу k на базову точку P.

$$k = rand([1, n-1])$$
$$R = kP$$
$$\Pi = Mask(EC2OSP_E(R, compressed))$$

При підписі повідомлення, що відновлюється, маскується передпідписом за допомогою складання за модулем 2.

$$r = d \oplus \Pi$$
$$s = (OS2IP(r)k - OS2IP(r) - 1/(x_A + 1)) \bmod n$$

Для перевірки підпису проводиться відновлення передпідпису та відновлення частини, що відновлюється, з r компоненти підпису.

Рішення щодо дійсності підпису приймається після аналізу збитковості відновленої частини.

$$R' = ((1 + OS2IP(r') + s')/OS2IP(r'))P + (s'/OS2IP(r'))Q$$
$$\Pi' = Mask(EC2OSP_E(R', uncomressed))$$
$$d' = r' \oplus \Pi'$$

5. ПІДПИС АБЕ-ОКАМОТО У ГРУПІ ТОЧОК ЕЛІПТИЧНИХ КРИВИХ (ELLIPTIC CURVE ABE-OKAMOTO MESSAGE RECOVERY SIGNATURE)

Спеціальні функції

Схема [2,10] використовує дві функції гешування $Hash_1$ та $Hash_2$, бітова довжина яких відповідно дорівнює довжині збитковості, та довжині останньої частини повідомлення, що буде відновлено. Використовується також функція формування гамми, що визначена стандартом як $MGF1$.

$$Hash_1: \{0,1\}^{8^*} \to \{0,1\}^{8L_{red}}$$
$$Hash_2: \{0,1\}^{8^*} \to \{0,1\}^{8(L_f + 1 - L_{red})}$$
$$MGF: \{0,1\}^{8^*} \to \{0,1\}^{8(L(n)+K)}$$
$$MGF(x) = MGF1(x, L(n) + K), x \in \{0,1\}^{8^*}$$

Алгоритм формування передпідпису є стандартним

$$k = rand([1, n-1])$$
$$R = kP$$
$$\Pi = EC2OSP_E(R, compressed)$$

Формування повідомлення

Схема ECAO визначає алгоритм формування частини повідомлення, що відновлюється. Функція ділить повідомлення на частину, що включена до підпису (M_{rec}), та частину, що передається відкритою (M_{clr}). Частина M_{rec} подвійно гешується: перший геш використовується у незмінному вигляді, друга частина використовується як збитковість для першої, та додається до підпису. Таким чином збільшується корисний простір для відновлення.

$$M = M_{rec} \| M_{clr}$$
$$L(M_{rec}) \le L_{\max}$$
$$pad = I2OSP(1, L_{\max} + 1 - L(M_{rec}))$$
$$\tilde{M}_{rec} = pad \| M_{rec}$$
$$h = Hash_1(\tilde{M}_{rec})$$
$$d = h \| (Hash_2(h) \oplus \tilde{M}_{rec})$$

Підпис приховує частину повідомлення, що відновлюється, у r компоненті підпису, а частини в геш частини, що передається відкрито – у s компоненті.

$$r = d \oplus \Pi$$
$$u = MGF(r \| M_{clr})$$
$$t = OS2IP(u) \bmod n$$
$$s = (k - x_A t) \bmod n$$

При перевірянні підпису за допомогою відкритої частини повідомлення, спочатку отримується гамма, за допомогою якої відновлюється частина повідомлення, що приховується в підписі.

$$u' = MGF(r' \| M'_{clr})$$
$$t' = OS2IP(u') \bmod n$$
$$R' = s'P + t'Q$$
$$\Pi' = EC2OSP_E(R', compressed)$$
$$d' = r' \oplus \Pi'$$
$$h' = [d']^{8L_{red}}$$
$$\tilde{M}'_{rec} = [d']_{8(L_f+1-L_{red})} \oplus Hash_2(h')$$
$$h'' = Hash_1(\tilde{M}'_{rec})$$

6. ПІДПИС ПІНТСОВА-ВАНСТОНА У ГРУПІ ТОЧОК ЕЛІПТИЧНИХ КРИВИХ (ELLIPTIC CURVE PINTSOV-VANSTONE MESSAGE RECOVERY SIGNATURE)

Схема підпису Пінтсова-Ванстона із частковим відновленням повідомлення є варіантом схеми Шнора [9, 10, 11] та підпису Ніберга-Рюпеля. Це дає можливість формувати дуже малі підписи на повідомленнях із збитковістю. Для 80 бітів безпеки, розмір підпису міститься 20-30 байт, у залежності від кількості збитковості у повідомленні. (Для порівняння: підпис ECDSA із такими ж самими параметрами домену має розмір близько 40 байт). Схема використовує блоковий симетричний шифр. Висувається декілька кандидатів БСШ для використання із цією схемою: AES, 3DES, та також доводиться можливість використання перетворень на базі операції XOR.

Деякі схеми підпису із повним або частковим відновленням повідомлення мають обмеження на довжину. Наприклад, у схемі Nyberg-Rueppel це обмеження має такі недоліки:

• Для дуже короткого повідомлення, примусово визначена довжина підпису зобов'язує використовувати більше доповнення.

• Для повідомлення, що є більшим ніж означена довжина підпису, неможливо забезпечити повне покриття підписом.

Схема Пінтсова-Ванстона може використовуватися без таких обмежень. При використанні схеми із еліптичними кривими може бути отриманий найкращий результат з точки зору розміру підпису.

7. ПІДПИС KCDSA У ГРУПІ ТОЧОК ЕЛІПТИЧНИХ КРИВИХ (ELLIPTIC CURVE KCDSA/NURBERG-RUEPPEL MESSAGE RECOVERY SIGNATURE)

Спеціальні функції

Схема [10, 11] використовує функції маскування та гешування із однаковою довжиною та n-бітовою довжиною результату.

$$MGF : \{0,1\}^{8*} \to \{0,1\}^{8L(n)}$$

Передпідпис

$$k = rand([1, n-1])$$
$$R = kP$$
$$\Pi = MGF(EC2OSP_E(R, compressed))$$

У підписі використовуються дані сертифікату (відкритий ключ) z_a у якості додаткових даних для маскування. Таким чином, для перевірки підпису та відновлення повідомлення дійсно необхідно мати сертифікат підписувача.

$$r = d \oplus \Pi \oplus MGF(z_A \| M_{clr})$$
$$t = OS2IP(r) \bmod n$$
$$s = (k - x_a t) \bmod n$$

Перевірка

$$t' = OS2IP(r') \bmod n$$
$$R' = s'P + t'Q$$
$$\Pi' = MGF(EC2OSP_E(R', compressed))d' =$$
$$= r' \oplus \Pi' \oplus MGF(z_a \| M'_{clr})$$

8. ПРИКЛАДИ ВИКОРИСТАННЯ

Поштові марки

Поштові марки мають містити поштові дані у проміжку від 20 до 50 байт. Деякі частини поштових даних, включаючи дату та поштовий код відправника, відправляються у чистому вигляді. Інші частини, такі як серійний номер повідомлення, поштова адреса відправника включаються до частини, що буде відновлено. Мінімально ця інформація займає від 13 байт даних. Натуральна збитковість може бути на рівні 7 байт. Щоб отримати 10 байтів збитковості, 3 байти збитковості вирівнюють. Таким чином, частина, що відновлюється, буде містити 16 байтів.

Розробник рекомендує використовувати у таких випадках 20-байтову еліптичну криву (160-163 біти), SHA-1 у якості 20 байтової функції гешування та 3DES. Таким чином, інша частина підпису буде займати 20 байтів, та 3 байти буде додано до збитковості. Сумарне перебільшення складе 23 байти при рівні безпеки 2^{-80}.

Підписування дуже короткого повідомлення

Розглянемо підписування малого повідомлення довжиною у 1 байт, таке як так/ні, придбати/продати/затримати тощо. Для того щоб запобігти атакам повтору, до таких коротких повідомлень треба додати номер послідовності, у розмірі від 3-х байт. У разі такого використання, треба збільшити стійкість до підробки. Для цього додається вирівнювання у розмірі 4-х байт. Таким чином, отримуємо частку повідомлення у 8 байтів. Із DES, SHA-1 та 20 байтовою еліптичною кривою підпис буде мати 28 байтів, 24 з яких є криптографічним надлишком. Номер послідовності є необхідним елементом, та, таким чином, буде природною збитковістю. Тому надлишок складає 7 байтів, що дає 2^{-56} стійкість до екзистенційної підробки (повна стійкість складає 2^{-80}).

Підписування та відновлення повідомлень із надлишком у 20 байт

Якщо повідомлення, що має бути відновлене, довше ніж 20 байт, можна розраховувати на те, що деякі вимоги до форматування повідомлен-

ня становлять як мінімум 10 байтів натуральної збитковості. Тоді надлишок складе 20 байтів – друга частина підпису. У гіршому випадку, якщо натуральна збитковість відсутня, можна додати 10 байтів збитковості.

9. СТІЙКІСТЬ ЕЦП ECPV (ПІНТСОВА-ВАНСТОНА)

Стійкість ЕЦП з доповненням розглянемо на прикладі ECPV (Пінтсова-Ванстона). Її зумовлюють такі компоненти, як стійкість перетворень в групі точок еліптичної кривої, стійкість функції гешування, стійкість БСШ та величина збитковості. Методика порівняльного аналізу наведена в [12].

Крім того, стійкість ECPV залежить від незалежності цих чотирьох компонент. Наприклад, функція гешування не повинна бути означеною у термінах групи точок еліптичної кривої, та множина збитковості не повинна включати в себе множину усіх M_{clr}, таких що $S_{v\backslash}(M_{clr})$ для деяких фіксованих r.

Заради швидкості, деякі реалізації можуть використовувати відсікання або доповнення, замість більш безпечної функції встановлення ключа на базі функції гешування. Це не впливає на загальну безпечність підпису.

Для більш точного визначення правил використання підпису розглянемо декілька припущень.

Припустимо, що f – зловмисник, який здійснює підробку. Із значною вірогідністю ми можемо припустити, що f може опитувати $H()$ та $S()$. Ґрунтуючись на порядку виконання, ми можемо отримати декілька варіантів.

Спочатку виконується гешування $H(r \| M_{rec})$, потім $S_{v'}(s)$. Так як $S_{v'}(d) = r$, ми маємо $S^{-1}_{v'}(r) = d$. Але значення $S_{v'}(d)$ отримується випадково, таким чином, вірогідність того, що $S_{v'}(d) = r$ є незначною.

1. $H(r \| M_{clr}) \to S^{-1}_{V'}(r)$. У цьому випадку $d = S^{-1}_{V'}(r)$ повинна обиратися випадково, таким чином вірогідність того, що $s \in N$ є 2^{a-b}.

2. $S^{-1}_{V'}(r) \to H(r \| M_{clr})$ та $S^{-1}_{V'}(s) \to H(r \| M_{clr})$. Використовується лема Понтчеваля та Штерна. Випадково обирається індекс t, та f виконує дії два рази, але змінює t випадкове значення H, що було отримане f. Так як загальна кількість опитування є біноміальною, існує значний шанс, що t запитання $H()$ буде $H(r \| M_{clr})$ у обох випадках. Якщо h та h' випадкові значення, отримані від $H()$ у цьому випадку, та (r,s) і (r',s'), тоді $dG - hW = V = d'G - h'W$, тому що значення V отримано f у першому опитуванні. Так як $sG = W$ та $(h-h')W = (d-d')G$, то $s = (h-h')^{-1}(d-d') \bmod r$.

Сильна геш-функція

Нехай $H()$ буде функцією гешування. $H()$ є сильною функцією гешування, якщо не існує поліноміального за часом алгоритму $A()$, що спочатку знайде значення h або l_0, а потім випадкове вхідне значення c, деяке l таке, що $H(c \| l) = h$, або $H(c \| l) = H(c \| l_0)$ із великою вірогідністю.

$$Hash: \{0,1\}^{8^*} \to \{0,1\}^{8L_{red}-1}$$
$$MGF: \{0,1\}^{8^*} \to \{0,1\}^{8(L_{key})}$$
$$Sym: \{0,1\}^{8^*} \times \{0,1\}^{8L_{key}} \to \{0,1\}^{8^*}$$

Схема визначає свій спосіб формування частини, що відновлюється.

$$C_{red} = Oct(L_{red})$$
$$d = \tilde{C}_{red} \| M_{rec}$$

У підпису приймає участь функція $Sym()$ симетричного шифрування, у якості функції маскування у стандартній схемі Ніберга-Рюпеля.

$$r = Sym(d, \Pi)$$
$$u = Hash(r \| M_{clr})$$
$$t = OS2IP(u)$$
$$s = (k - x_a t) \bmod n$$

Перевірка

$$u' = Hash(r' \| M'_{clr})$$
$$t' = OS2IP(u')$$
$$R' = s'P + t'Q = (x', y')$$
$$\Pi' = KDF(FE2OSP_f(x'))$$
$$d' = Sym^{-1}(r', \Pi')$$
$$C_{red} = Oct(L_{red})$$
$$\tilde{C}_{red} = [d']^{8L_{red}}$$
$$M'_{rec} = [d']_{8(L(d')-L_{red})}$$

10. СТІЙКІСТЬ ЕЦП З ВІДНОВЛЕННЯМ ПОВІДОМЛЕНЬ ДО КОЛІЗІЙ

За основу досліджень колізійної стійкості, щодо захисту від підробок, було взято ідеї з [6, 14, 15]. В табл. 1 наведені загальні показники ЕЦП з відновленням повідомлень.

Таблиця 1

Показники ЕЦП

	$L(d)$	$L(H^*(M_{rec}))$	$L(H^*(M_{clr}))$	(r,s)
NR	$L(n)-1$	$d: L(d) - L(M_{rec})$		$\{0,1\}^{8L(n)} \times [1, n-1]$
ECNR	$L(n)-1$	$d: L(d) - L(M_{rec})$		$\{0,1\}^{8L(n)} \times [1, n-1]$
ECMR	$L(n)$; $L(\{0,1\}^{8(2L_+1)})$	$d: L(d) - L(M_{rec})$		$\{0,1\}^{8L(n)} \times [1, n-1]$
ECAO	$L(\{0,1\}^{8(L_F+1)})$	$L(M_{rec})_1 +$ $+[L(d) - M_{rec}]_2$	$L(n) \times [1,n]$	$\{0,1\}^{8(L_F+1)} \times [1, n-1]$

Модель компоненти захисту від підробки:

– для захисту від підробки вироблюється код виробу, що складається с коду групи товарів та серійного номеру виробу;

– будемо вважати, що серійний номер виробу є унікальним, та не містить природної надлишковості. Код групи товарів напроти має деяку природну надлишковість;

– код виробу вкладається у частині підпису, що відновлюється;

– опціонально може бути включена додаткова відкрита інформація (можливо як частина параметрів домену виробника), що буде розглядатися як частина повідомлення, що є відкритою.

Розглянемо можливість виникнення колізій компонентів означених моделей та граничну кількість одиниць товару, що можна маркувати за допомогою означеної моделі.

11. СТІЙКІСТЬ ДО КОЛІЗІЙ ЧАСТИНИ ПОВІДОМЛЕННЯ, ЩО ВІДНОВЛЮЄТЬСЯ

Вірогідність колізії для підписів із відновленням повідомлення серед ω виробів із точки зору всього підпису можна оцінити як [14]

$$P(\omega,(r,s)) = 1 - e^{-\omega\frac{\omega-1}{2(2^{8L(r)+8L(s)}-1)}}. \quad (2)$$

Але при перевірці підпису використовується також семантичний критерій остаточного вирішення питання дійсності підпису. Для зменшення вірогідності підтвердження та відновлення некоректного повідомлення використовується збільшення надлишковості повідомлення, пов'язаного із гешуванням частини повідомлення, що відновлюється.

Максимальна довжина частини, що відновлюється, має величину $L(n)-1$. Для скінченного поля у дійсний час дозволяється використання довжин $n \in \{2048,3072\}$. Для групи точок еліптичних кривих $n \in 160,163,...431,...$ Таким чином, для $n = 2048$ максимальна довжина частини, що може бути відновлена, складе 2040 біт. Так як рішення щодо дійсності підпису приймається після перевірки надлишковості, стійкість всієї схеми залежить від її розміру. Тому максимальна кількість корисної інформації, що може бути відновлена з підпису, зменшується на необхідний розмір надлишковості.

Для ЕЦП NR, ECMR, ECMR, ECKNR вірогідність колізії P кодів, вироблених із ключем у n бітів у партії з ω виробів, згідно парадоксу про день народження із збільшенням розміру корисного повідомлення $8L(M_{rec})$, може бути визначена як [15]:

$$P(M_{rec},\omega,n) = 1 - e^{-\omega\frac{\omega-1}{2(2^{8L(n)-8L(M_{rec})}-1)}}. \quad (3)$$

Формула (3) також дозволяє обчислити вірогідність колізії у відкритій частині тексту.

Хоча у підпису ECPV умовно немає залежності вірогідності колізії повідомлення,що відновлюється від параметрів ЕС, у реальних умовах вираз (3) буде мати вигляд

$$P(M_{rec},\omega,Sym_{glen}) = 1 - e^{-\omega\frac{\omega-1}{2(2^{8L(Sym_{glen})-8L(M_{rec})}-1)}}, \quad (4)$$

де Sym_{glen} – максимальний вихід блоку чи гамми симетричного шифру у октетах.

Для ЕСАО вираз (4) вірний з деякими уточненнями:

$$P(M_{rec},\omega,L_F) = 1 - e^{-\omega\frac{\omega-1}{2(2^{8L_F-8L(M_{rec})+8L(M_{rec})}-1)}} \quad (5)$$

де L_F – розмір поля, над яким будується еліптична крива.

Вирази (3), (4), (5) можуть бути трансформовані для визначення максимальної кількості товару, що можна маркувати ЕЦП із заданими параметрами:

$$\omega(P,M_{rec},n) = \sqrt{2(2^{8L(n)-8L(M_{rec})}-1)\ln\left(\frac{1}{1-P}\right)}, \quad (6)$$

$$\omega(P,M_{rec},Sym_{glen}) = \sqrt{2(2^{8L(Sym_{glen})-8L(M_{rec})}-1)\ln\left(\frac{1}{1-P}\right)}, \quad (7)$$

$$\omega(P,M_{rec},L_F) = \sqrt{2(2^{8L_F-8L(M_{rec})+8L(M_{rec})}-1)\ln\left(\frac{1}{1-P}\right)}. \quad (8)$$

Тепер підрахуємо дійсну кількість товару, яку можна маркувати із зазначеною схемою. Для того введемо додаткову умову

$$L(M_{rec}) = L(\omega).$$

Із уточненнями рівняння буде мати вигляд

$$\begin{cases} \omega(P,M_{rec},n) = \sqrt{2(2^{8L(n)-8L(M_{rec})}-1)\ln\left(\frac{1}{1-P}\right)} \\ \lfloor L(M_{rec})\rfloor = \lfloor L(\omega)\rfloor. \end{cases} \quad (9)$$

Легко побачити, що довжина $M_{rec} = 37$ забезпечить найбільш повне використання простору значень нашої моделі, коли вірогідність колізії $P = 2^{-52}$, та $n = 163$. Якщо взяти 7 бітів як код групи товарів (приблизно 100 груп, із надлишковістю), то загальна кількість товарів, що може бути випущена у межах групи, дорівнюватиме $2^{30} = 1073741824$.

Підпис ЕСАО використовує декілька гешувань частини повідомлення, що відновлюється. Одне з них включається до геш-токену, інше складається за модулем два із повідомленням. Таким чином, можна казати, що простір можливих значень об'єднується.

Для підпису NR вірогідність колізії суттєво менша, так як розмір n має розмір від 2048 бітів.

Підпис ECPV не накладає обмежень на сумарну довжину повідомлення із геш-токеном. Таким чином, можна казати, що ECPV дозволяє встановити довільну вірогідність колізії геш-токену.

12. СТІЙКІСТЬ ДО КОЛІЗІЙ ЧАСТИНИ ПОВІДОМЛЕННЯ, ЩО НЕ ВІДНОВЛЮЄТЬСЯ

ЕЦП із відновленням повідомлення гарантує цілісність та неспростовність не тільки частини повідомлення, що відновлюється, але й усього повідомлення взагалі. В [10, 11] стверджується,

що для підписів NR, ECNR, ECMR вірогідність до колізії відкритої частини повідомлення така ж сама, як і для закритої (3), (5), (4). Для ECAO, ECPV вірогідність колізії відкритої частини для ω підписів складе

$$P(\omega,n)=1-e^{-\omega\frac{\omega-1}{2(2^{8L(n)}-1)}}. \qquad (10)$$

13. АНАЛІЗ СКЛАДНОСТІ РЕАЛІЗАЦІЇ ЕЦП З ВІДНОВЛЕННЯМ ПОВІДОМЛЕННЯ

В табл. 2 наведено значення числа операцій, що необхідно виконати для алгоритмів обчислення ЕЦП, що входять до [10,11].

Таблиця 2

Число операцій, що виконуються при обчисленні підпису

	NR	ECRN	ECMR	ECAO	ECPV	ECKNR
Складання за модулем n	2	2	3	1	1	1
Множення за модулем n	1	1	2	1	1	1
Інверсія за модулем n	0	0	1	0	0	0
Скалярне множення у г.т. ЕК, або підведення до ступеню у скінченному полі	1	1	1	1	1	1
Сумма за модулем 2	0	0	1	2	0	2
Гешування	1	1	1 (або 0)	2	1	0
MGF/KDF	0	0	0 (або 1)	1	1	2
БСШ	0	0	0	0	1	0

В табл. 3 наведено значення числа операцій, що необхідно виконати для алгоритмів при перевірці ЕЦП.

Таблиця 3

Число операцій, що виконуються при перевірції підпису

	NR	ECRN	ECMR	ECAO	ECPV	ECKNR
Складання за модулем n	1	1	2	0	0	0
Множення за модулем n	0	0	2	0	0	0
Інверсія за модулем n	0	0	1	0	0	0
Складання на еліптичній кривій або множення у скінченному полі	1	1	1	1	1	1
Скалярне множення у г.т. ЕК, або підведення до ступеню у скінченному полі	2	2	2	2	2	2
Сумма за модулем 2	0	0	1	2	0	2
Гешування	1	1	1 (або 0)	2	1	0
MGF/KDF	0	0	0 (або 1)	1	1	2
БСШ	0	0	0	0	1	0

В табл. 4 наведено значення числа операцій, що необхідно виконати для алгоритмів додавання та подвоєння у різних базисах.

Таблиця 4

Складність додавання та подвоєння у різних базисах

Координати	Додавання точок	Подвоєння точок
Афінні	$t(A+A)=I+2M+S$	$t(2A)=I+2M+2S$
Проективні	$t(P+P)=12M+2S$	$t(2P)=7M+5S$
Якобіанові	$t(y+y)=12M+4S$	$t(2I)=4M+6S$
Чудновського	$t(y^c+y^c)=11M+3S$	$t(2I^c)=5M+6S$
Модифіковані якобіанові	$t(y^m+y^m)=13M+6S$	$t(2I^m)=4M+4S$

Необхідно враховувати, що швидкість функції гешування приблизно у десять разів більша, ніж інверсії.

14. ЧИСЛО ГЕШУВАНЬ MGF

Стандарт ISO/IEC 9796-3 визначає алгоритми формування гамми MGF1 та MGF2, як:

$$MGF1(x,l)=[Hash(x\|I2OSP(0,4))\|$$
$$\|Hash(x\|I2OSP(1,4))\|\ldots$$
$$\ldots\|Hash(x\|I2OSP(k-1,4))]^{8l} \qquad (11)$$

$$MGF2(x,l)=[Hash(x\|I2OSP(1,4))\|$$
$$\|Hash(x\|I2OSP(2,4))\|\ldots$$
$$\ldots\|Hash(x\|I2OSP(k,4))]^{8l}, \qquad (12)$$

де $k=\lceil l/L_{Hash}\rceil$.

Таким чином, для кожного виконання функцій MGF потрібно обчислення k функцій гешування.

В табл. 5 наведено експериментально отримані значення швидкість підписів алгоритмів ЕЦП стандарту ISO/IEC 9796-3.

Таблиця 5

Швидкість підписів для алгоритмів стандарту ISO/IEC 9796-3

	Підпис (100/с)	Перевірка (100/с)
NR	3.78	7.78
ECNR	10.61	21.54
ECMR	11.13	20.51
ECAO	10.91	21.25
ECPV	9.61	17.54
ECKNR	12.61	20.54

В табл. 6 наведені дані відносно обсягу памяті, що необхідна для реалізації алгоритмів ISO/IEC 9796-3.

Таблиця 6

Обсяг пам'яті, що необхідний для реалізації алгоритмів ISO/IEC 9796-3

	Підпис/100	Перевірка/100
NR	170514864	190513864
ECNR	1274718104	2610783160
ECMR	1282372192	2506140400
ECAO	1284437920	2603164096
ECPV	1074718104	1610783160
ECKNR	1274718204	1810783460

ВИСНОВКИ

1. Існує два типи підписів: із відновленням повідомлення та доповненням. У дійсний час абсолютно поширеним типом є із доповненням повідомлення

2. У дійсний час в усіх сферах діяльності мають широке розповсюдження мобільні обчислювальні та комунікаційні системи, пристрої особистої ідентифікації. Такі системи характеризуються низькою вартістю виготовлення, невеликою обчислювальною здатністю та невеликими об'ємами пам'яті. Для забезпечення цілісності та достовірності інформації за допомогою цих систем доцільно використовувати спеціалізовані криптографічні алгоритми, що визначені в ISO/IEC 9796–3.

3. Підпис із відновленням повідомлення, порівняно до підпису із доповненням, надає додаткову послугу безпеки – конфіденційність. Також для невеликих обсягів повідомлення можливо сховати усю інформацію, що передається, у самому підписі.

4. Підписи із відновленням повідомлення стандартизовані у міжнародних стандартах ISO/IEC 15946-4, ISO/IEC 9796-3. Стандарт ISO/IEC 9796-3 поширює та уточнює алгоритми, що вказані у ISO/IEC 15946-4, та з 2008 року є основним стандартом підписів із відновлення повідомлення.

5. Стандарт ISO/IEC 9796-3 містить 5 підписів у групі точок ЕК, та 1 у скінченному полі. Підписи мають спільну загальну схему Ніберга-Рюпеля, але впроваджують модифікації блоку передпідпису, для оптимального використання *r*-компоненти.

6. Найбільш перспективними є підписи, ECPV та ECNR. ECNR із модифікаціями по суті є національним стандартом України ДСТУ 4145:2002. ECPV є перспективним підписом, що використовує симетричне шифрування для включення інформації до підпису, і не накладає обмежень на кількість інформації, що може бути відновлена. ECPV також є підписом із найменшою довжиною.

7. На міжнародному рівні розглядається можливість використання ECPV та ECNR у RFID чипах для захисту товарів від підробок, та для маркувань медикаментів у Індії.

Література.

[1] ISO/IEC 14888-1, Information technology — Security techniques — Digital signatures with appendix — Part 1: General.

[2] M. ABE and T. OKAMOTO, "A signature scheme with message recovery as secure as discrete logarithm," Advances in Cryptology — Asiacrypt'99, Lecture Notes in Computer Science 1716, pp. 378-389, Springer-Verlag, 1999.

[3] ANSI X9.62-1999, Public Key Cryptography For The Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA), 1999.

[4] ANSI X9.63-1999, Public Key Cryptography For The Financial Services Industry: Elliptic Curve Key Agreement and Transport Protocols, 1999.

[5] IEEE Std 1363-2000, IEEE Standard Specifications for Public-Key Cryptography.

[6] C. H. LIM and P. J. LEE, "A study on the proposed Korean digital signature algorithm," Advances in Cryptology — Asiacrypt'98, Lecture Notes in Computer Science 1514, pp. 175-186, Springer-Verlag,1998.

[7] A. J. MENEZES, P. C. VAN OORSCHOT and S. A. VANSTONE, "Handbook of applied cryptography," CRC Press, 1997.

[8] A. MIYAJI, "Another Countermeasure to Forgeries over Message Recovery Signature," IEICE Trans., Fundamentals, vol. E80-A, No.11, pp. 2192-2200, 1997.

[9] K. NYBERG and R. A. RUEPPEL, "Message recovery for signature schemes based on the discrete logarithm problem," Designs, Codes and Cryptography, 7, pp. 61-81, 1996 L. PINTSOV and S. VANSTONE, "Postal Revenue Collection in the Digital Age," Proceedings of the.

[10] ISO/IEC 15946-4 Digital signatures giving messagerecovery / ISO/IEC. — URL:http://www.iso.org/, 2004. — Жовтень.

[11] ISO/IEC 9796-3: Discrete logarithm based mechanisms / ISO/IEC. — URL:http://www.iso.org/.

[12] Методика сравнения алгоритмов ЭЦП в группе точек эллиптической кривой / Горбенко Ю. И. Денисенко Б. И. // Прикладная радиоэлектроника. — 2008.

[13] Miyaji, Atsuko. Weakness in message recovery signature schemes based on discrete logarithm problems 2. — 2002.

[14] Вступ у теорію γ-мірних колізій та її застосування / Сінаюк Л.В. Горбенко Ю.І., Фролов О.С. // Прикладная радиоэлектроника. — 2006.

Надійшла до редколегії 16.09.2009

Горбенко Юрій Іванович, кандидат технічних наук, технічний директор ЗАТ «ІІТ», науковий співробітник НІЦ «Z» каф. БІТ ХНУРЕ. Область наукових інтересів: захист інформації в інформаційно-телекомунікаційних системах.

Шевчук Олексій Анатолійович, магістр кафедри БІТ ХНУРЕ. Область наукових інтересів: захист інформації в інформаційно-телекомунікаційних системах.