УДК 004.056.53(045)

ПРАКТИЧНІ СХЕМИ РЕАЛІЗАЦІЇ АЛГОРИТМІВ ЕЛЕКТРОННОГО ЦИФРОВОГО ПІДПИСУ

Анна Чунарьова
Національний авіаційний університет

Анотація: **Проведено порівняльний аналіз асиметричних схем формування ЕЦП, які засновані на проблемі дискретного логарифмування над скінченним полем та еліптичними кривими. В результаті складена порівняльна таблиця оцінки ефективності використання даних алгоритмів.**
Summary: **A comparative analysis of schemes of asymmetric digital signature based on the discrete logarithm problem over finite fields and elliptic curves. As a result, compiled a comparative table of assessing the efficiency of these algorithms.**
Ключові слова: **Інформаційна безпека, електронний цифровий підпис, еліптична крива, дискретне логарифмування, хеш-функція.**

I Вступ

Розвиток інформаційних технологій привів до появи електронного документообігу. Проте використання електронного документообігу пов'язане з збереженням документів від несанкціонованого копіювання, модифікації і підробки. Для вирішення проблеми захисту інформації від несанкціонованого доступу (НСД) необхідно використовувати сучасні специфічні засоби і методи захисту. Одним із поширених методів такого захисту є використання електронного цифрового підпису (ЕЦП). Під поняттям електронного цифрового підпису будемо розуміти реквізит електронного документа, отриманий в результаті криптографічного перетворення інформації з використанням закритого ключа електронного цифрового підпису, що дає змогу ідентифікувати власника електронного документа. На даний час ЕЦП виконує функції контролю цілісності електронного документу та підтвердження його авторства [4 – 6].

II Постановка задачі

ЕЦП може бути сформована за допомогою двох схем: 1) симетрична схема; передбачає наявність в системі третьої особи, яка користується довірою обох сторін; авторизацією документа в даній схемі є сам факт зашифрування електронного документа секретним ключем і передача його третій особі; 2) асиметрична схема; дана схема відноситься до криптосистем з відкритим ключем та не потребує наявності третьої особи.

Метою даною статті є аналіз сучасних асиметричних схем реалізації ЕЦП.

III Основна частина

Сімейство алгоритмів цифрового підпису, що засновані на проблемі дискретного логарифму

Спочатку дамо короткий опис двох фундаментальних алгоритмів цифрового підпису, що засновують ціле сімейство алгоритмів, базованих на складності вирішення задачі дискретного логарифмування у скінченному полі. Це добре відомі у світі алгоритми ElGamal та DSA.

Отже у алгоритмі ElGamal обирається велике просте число p (мінімального розміру – 1024 біта) і число q, що дорівнює p–1 або є великим простим множником p–1; q має мінімальний розмір – 160 біт. Далі обираємо число $g \in F_p$, таке, що його мультиплікативний порядок за модулем $p = q$. Ці параметри є звичайними параметрами системи, що здійснює підпис за алгоритмом ElGamal. p та g називаються системними параметрами.

Для генерації ключів алгоритму користувач-підписувач генератором псевдовипадкових чисел генерує число $x \in Z_q$ і обраховує відкритий ключ за формулою

$$y = g^x \bmod p. \tag{1}$$

Для кожного підписуваного повідомлення генерується секретне псевдовипадкове число $k \in Z_q$, що потім використовується при генеруванні підпису. Підпис складається з двох чисел (r, s), що вираховуються наступним чином:

$$r = g^k \bmod p,\ s = k^{-1}(m - rx) \bmod (p-1), \tag{2}$$

де m представляє підписуване повідомлення (зауваження: замість відкритого значення m остаточний

варіант схеми підпису використовує однонаправлену хеш-функцію $H(m)$).

Перевірка підпису ElGamal проходить наступним чином: спочатку перевіряється умова $0 < r < p; 0 < s < p-1$ та перевіряється рівняння $g^m = y^r r^s \bmod p$. Якщо рівняння вірне, то підпис визнається дійсним. Коректність схеми доводиться таким чином:

$$g^m \equiv g^{xr} g^{ks} \equiv (g^x)^r (g^k)^s \equiv (y)^r (r)^s \bmod p.$$

Тепер опишемо підпис за схемою DSA. Тут також обирається велике просте *p* 1024-бітової довжини (в початковому варіанті було 512 біт), множник $p-1$ та просте число q 160-бітової довжини. Породжуючий елемент скінченної групи обраховується за формулою:

$$g = h^{(p-1)q} \bmod p$$

де h – будь-яке число, меньше за $p-1$, для якого $h^{(p-1)q} \bmod p > 1$.

Секретний ключ вибирається випадково, а відкритий обчислюється за (1). В алгоритмі також використовується однонаправлена хеш-функція: $H(m)$. Стандарт визначає використання як хеш-функції SHA (Secure Hash Algorithm). Перші три параметри: *p, q* і *g* – відкриті і можуть бути спільними для користувачів мережі. Щоб підписати повідомлення m: підписувач генерирує випадкове число $k \in Z_q$ та обраховує

$$r = (g^k \bmod p) \bmod q \text{ та } s = k^{-1}(H(m) + rx) \bmod q.$$

Пара чисел (r, s) слугує підписом.

При перевірці обчислюються наступні значення:

$$w = s^{-1} \bmod q,$$

$$u1 = (H(m) \bullet w) \bmod q,\ u2 = (r \bullet w) \bmod q,$$

і перевіряється умова

$$v = (g^{u1} \bullet y^{u2}) \bmod p \bmod q. \quad (3)$$

Якщо $v = r$, то підпис дійсний.

Отже з опису цих 2 алгоритмів можна одразу зробити висновок, що вони є дуже схожими: еквівалентне генерування ключової інформації, принципи побудови схем підпису-верифікації. Безпека обох схем залежить від можливості знаходження секретного ключа *x*, тобто вирішення проблеми дискретного логарифмування в полі цілих чисел та ймовірності знаходження колізій до хеш-функції $H(m)$, що використовується в підписі. Обидві проблеми вважаються практично неможливими для вирішення.

Існує безліч можливих варіацій DSA-ElGamal-подібних схем. Деякі з них спрощують обчислення завдяки видаленню операції знаходження оберненого елементу до елементу мультиплікативної групи при генеруванні підпису, інші – мають аналогічну властивість, але для операції верифікації підпису. Існують модифікації для пакетних перевірок підписів та підписів з відновленням повідомлень. Деякі з цих схем є безпечними, безпечність інших поки що не доведена [1].

Вдалу спробу пояснити той факт, що DSA і ElGamal є різними реалізаціями однієї і тієї ж спільної схеми, зробили швейцарські спеціалісти з безпеки Ніберг та Рюппель у своїй фундаментальній статті «Message Recovery for Signature Schemes Based on the Discrete Logarithm Problem» [2]. Автори, виходячи зі схожості описаних схем, виробили узагальнені рівняння алгоритмів: узагальнене рівняння підпису набуває вигляд

$$ak = (b + cx) \bmod q.$$

де *a, b* і *c* є цілими числами.

У табл. 1 наведені можливі варіанти перестановок *a, b* і *c* без врахування ефектів $\pm$. Кожен рядок надає шість можливостей. Перевіряючи підпис, одержувач повинен переконатися, що

$$r^a = g^b y^c \bmod p.$$

Це шість різних схем цифрових підписів. Додавання мінуса збільшує їх кількість до 24. При використанні всіх можливих значеннь *a, b* і *c* число схем доходить 120.

Ці схеми є ElGamal – подібними, проте їх просто можна зробити DSA – подібними. Для цього лише потрібно визначити r як $r = (g^k \bmod p) \bmod q$.

Таблиця 1 – Варіації схеми цифрового підпису з використанням дискретних логарифмів

Рівняння підпису	**Рівняння перевірки**
$r'k=(s+mx) \bmod q$	$r^{r'}=(g^s y^m) \bmod p$
$r'k=(m+sx) \bmod q$	$r^{r'}=(g^m y^s) \bmod p$
$sk=(r'+mx) \bmod q$	$r^s=(g^{r'} y^m) \bmod p$
$sk=(m+r'x) \bmod q$	$r^s=(g^m y^{r'}) \bmod p$
$mk=(s+r'x) \bmod q$	$r^m=(g^s y^{r'}) \bmod p$
$mk=(r'+sx) \bmod q$	$r^m=(g^{r'} y^s) \bmod p$

Це ще збільшує кількість схем. Додаткові узагальнення та зміни призводять більше, ніж до 13000 варіантів (не всі з них досить ефективні) [1].

Метою ж статті є пошук ефективного алгоритму цифрового підпису на дискретному логарифмі та з властивістю відновлення повідомлення. Як автори зазначили у статті: «схеми ElGamal та DSA мають один недолік порівняно з RSA – RSA надає послугу відновлення повідомлення: повідомлення може міститися всередині підпису і може бути отримане на стороні приймальника. Повідомлення не потребує попереднього хешування а підпис економить полосу пропускання та ємність пристроїв зберігання» [2]. Ніберг та Рюппель виокремлюють особливості, які повинна мати така схема:

- у формулі обчислення елементу r має бути присутня операція множення породжую чого елементу (або оберненого до нього) на повідомлення m (або m^{-1});
- m повинно бути замінено на 1 у рівнянні обчислення s ;
- повинно бути перебудуване рівняння верифікації підпису таким чином, щоб повідомлення m могло бути вираховане з відомих (r, s).

З урахуванням даних модифікацій схема ElGamal (3) перетворюється наступну:

$$r = mg^{-1} \bmod p \text{ та } s = k^{-1}(1 + rx) \bmod q$$

при генеруванні підпису та

$$m = g^{s^{-1}} y^{rs^{-1}} r \bmod p$$

при перевірці.

Далі авторами аналізуються можливості щодо комбінації параметрів схем ElGamal та DSA для вироблення найефективнішої. Ніберг та Рюппель поставили завдання знайти з усіх цих варіацій схему, що задовольняє наступним умовам:

- підпис може обчислюватися без інверсій по модулю, що забирають багато часу;
- верифікація може проводитися без інверсій;
- схема може надавати можливість відновлення повідомлення.

І така схема була знайдена. Вона називається схемою p-NEW. Підпис обчислюється за формулами

$$r = mg^{-k} \bmod p \text{ та } s = k - rx \bmod q.$$

Повідомлення відновлюється та підпис перевіряється з рівняння перевірки: $m = g^s y^r r \bmod p$. Коректність доводиться так:

$$g^s y^r r \bmod p \equiv (mg^{-k})(g^x)^r g^s \equiv mg^{-k+xr+s} \equiv m \bmod p.$$

З усього цього можна зробити закономірний висновок: DSA не є похідним від ElGamal.

У роботі [3] розглядаються недоліки безпеки схеми p-NEW. Зокрема, стверджується, що ця схема є схильна до атаки екзистенціальної підробки: порушник може обрати $r \in [1, p-1]$ та $s \in [1, q-1]$ довільно такі, що вони будуть підписувати повідомлення, отримане, використовуючи процедуру верифікації схеми p-NEW. Не зважаючи на те, що повідомлення не може бути обране порушником наперед, схема є небезпечною в цьому сенсі. Один із шляхів подолання цієї проблеми – використанням функції, що додає надлишковості повідомленню, що передається, для того щоб завдяки цій надлишковості встановити коректність та дійсність повідомлення. Це досягається використанняv функції, що здійснює симетричне відображення вихідного повідомлення у набір символів, що містяться у досить вузькому діапазоні. Проте вимоги до цієї функції роблять складною її реалізацію, тобто цей шлях практично неприйнятний або потребує значних математичних обчислень.

Автори у цій статті надають варіант, що обходить використання такої надлишкової функції, але цей

варіант втрачає можливість відновлення повідомлення, яку нам дає схема p-NEW. Для підпису повідомлення за цією схемою обирається ще один породжуючий елемент g_1 (адже цих елементів у групі може бути декілька) та підпис обчислюється наступним чином:

$$r = mg^k \bmod p \text{ та } s = -k - xp(r),$$

де як функція $\rho(r)$ використовується функція, що відображає елементи групи з породжуючим елементом g_1, що може бути групою точок еліптичної кривої, в групу точок скінченного поля.

Відповідно змінюється і рівняння перевірки:

$$g_1^{\ m} = ry^{\rho(r)} g^s.$$

Далі доводиться безпечність цієї схеми та наголошується, що вона є більш ефективною за схему подвійного підпису p-NEW, що є еквівалентно безпечною, але жертвує можливістю відновлення порівняно з оригінальною схемою.

Ще один варіант надається у [4], що модифікує послідовника p-NEW – схему Канга, яка в рівняння підпису додає ще один параметр, в результаті чого підписане повідомлення є триплетом (*r*,*s*,*t*). Ця схема робить схему Канга більш ефективною для довгих повідомлень.

Алгоритми цифрового підпису на еліптичних кривих. Цифровий підпис з відновленням повідомлення

Вище були описані варіанти схем, що базуються на проблемі дискретного логарифму в скінченних полях. Проте ці алгоритми легко адаптуються під еліптичні криві. Отже тепер зосередимося на пошуку схеми цифрового підпису з відновленням повідомлення, що використовує передові здобутки новітньої криптографії, а саме обчислення у групі точок еліптичної кривої.

Криптографічні алгоритми на еліптичних кривих будуються цілком аналогічно алгоритмам на простих скінченних полях. Фактично треба тільки піднесення до степеня за великим модулем (основне перетворення, що визначає стійкість криптографічного алгоритму) замінити на скалярний добуток точки еліптичної кривої на ціле число.

У загальному випадку алгоритм переводитися на аналогічній, визначений на еліптичних кривих за принципами, поданими в таблиці.

Таблиця 2 – Відповідність між параметрами криптографічних алгоритмів в простому скінченному полі та алгоритмами на еліптичній кривій

Криптографічний алгоритм в простому кінцевому полі	**Криптографічний алгоритм на еліптичній кривій над полем характеристики 2**
Циклічна підгрупа мультиплікативної (циклічної) групи простого скінченного поля $GF(p)$ порядку u (u - дільник порядку мультиплікативної групи поля $p-1$).	Циклічна підгрупа порядку *u* групи (не обов'язково циклічної) точок еліптичної кривої над полем $GF(2^m)$ (u - дільник порядку кривої N).
Породжуючий елемент циклічної підгрупи – ціле число g порядку u.	Породжуюча точка P циклічної підгрупи порядку u.
Секретний ключ - ціле число *t* (з інтервалу (1, *u* – 1))	Секретний ключ - ціле число *t* (з інтервалу (1, *u* – 1))
Відкритим ключем виступає ціле число $y=g^t$	Відкритим ключем виступає точка еліптичної кривої $Q=tP$

Розглянемо алгоритм ECDSA, що є аналогічним до DSA, але визначений не над кінцевим полем, а над групами точок еліптичних кривих.

Визначемо над полем F_p еліптичну криву E і точку P, що лежить на цій кривій $E(F_p)$. Точка має простий порядок *q* (це є дуже важливою умовою). Крива E і точка P є системними параметрами. Користувач генерує секретний та відкритий ключі наступним чином: обирає випадкове або псевдовипадкове ціле число x з інтервалу $[1, q-1]$, що буде секретним ключем. Потім обчислюється добуток (кратне) $Q = xP$, в результаті чого отримаємо ще одну точку, що лежить на цій кривій $E(F_p)$ та входить у циклічну підгрупу, утворену породжуючою точкою P.

Замість використання E і P як глобальних системних параметрів, можна фіксувати лише поле F_p для

всіх користувачів і дозволити кожному користувачеві вибирати свою власну еліптичну криву E і точку P на $E(F_p)$. В цьому випадку певне рівняння кривої E, координати точки P, а також порядок q цієї точки P повинні бути включені у відкритий ключ користувача. Якщо поле F_p фіксоване, то апаратна і програмна складові можуть бути побудовані так, щоб оптимізувати обчислення в тому полі. У той же час є величезна кількість варіантів вибору еліптичної кривої над полем F_p.

Підпис здійснюється наступною послідовністю кроків.

1. Згенерувати (псевдо-) випадкове ціле число $k \in [1, q-1]$;
2. Обрахувати $kP = (x_1, y_1)$ та покласти $r = x_1 \bmod q$, де r отримано з цілого числа $x_1 \in [1, p-1]$ приведенням по модулю q.

Якщо $r = 0$, то потрібно заново генерувати випадкове число та обчислювати r. Обчислити $k^{-1} \bmod q$ і покласти

$$s = k^{-1}(h + xr) \bmod q,$$

де h - значення хеш-функції підписуваного повідомлення. s не повинно дорівнювати 0, в противному випадку рівняння перевірки не існує.

Підписом для повідомлення є пара цілих чисел (r,s).

При перевірці першим кроком перевіряють чи $r \in [1, q-1]$ та $s \in [1, q-1]$, в противному випадку підпис не є дійсним. Далі обчислюються наступні змінні

$$u_1 = s^{-1}h \bmod q \text{ та } u_2 = s^{-1}r \bmod q.$$

Головне рівняння перевірки має наступний вигляд (аналогічний до (3) для DSA):

$$u_1 P + u_2 Q = (x_0, y_0).$$

Якщо $x_0 \bmod q = r$, то визнати підпис дійсним.

Коректність схеми показана наступним співвідношенням:

$$u_1 P + u_2 Q = (u_1 + xu_2)P = (s^{-1}h \bmod q + xs^{-1}r \bmod q)P = s^{-1}(h + xr) \bmod qP = kP..$$

Майже аналогічним є також алгоритм ГОСТ Р 34.10-2001, що описаний у [1], що має різницю лише в значеннях, для яких беруться обернені за модулем елементи.

Окрім цієї є також схеми, еквівалентні схемі Ніберга-Рюппеля p-NEW. Ці схеми надають можливість відновлення повідомлення. Одна з таких схем описана в [5].

У цій схемі генерування ключів проходить за аналогічною до ECDSA послідовністю кроків. Але формули обчислення підпису модифіковані:

$$R = kP,\ r = m^{-1}R_x \bmod p,$$

$$r' = r \bmod q \text{ та } s = k^{-1}(1 + r'x) \bmod q.$$

Підписом є пара чисел (r, s).

Повідомлення відновлюється з рівняння перевірки:

$$m = (s^{-1}P + r's^{-1}Q)_x r^{-1} \bmod p.$$

Отже, ми бачимо, що усі схеми, визначені над скінченним полем, що спираються на проблему дискретного логарифмування, можуть бути легко перетворені на аналогічні, визначені над групою точок еліптичної кривої, надаючи їм усі властиві раніше описані переваги.

Окрім схем цифрового підпису, що здійснюють повне відновлення повідомлення, існують також алгоритми з частковим відновленням. Сфери застосування алгоритмів теж достатньо широкі: ці схеми можуть застосовуватися в додатках та відносинах, де повне приховування повідомлення у підпис є зайвим. Проте певна частина має конфіденційну інформацію, розкриття якої не повинно відбутися. Подібні алгоритми знаходять застосування, зокрема, у поштових системах, де до секретної частини відносять такі дані як адреса отримувача, розголошення якої не є доречним, тоді як загальний зміст повідомлення не є конфіденційним. Як приклад розглянемо схему з підтвердженою безпекою, що входить у стандарт P1363 – схему цифрового підпису Пінцова і Ванстоуна [6].

Спочатку зазначимо критерії, за якими обирався алгоритм підпису.

- Криптостійкість, забезпечена на стандартному рівні (мінімум – 80 біт безпеки).

- Стійкість до екзистенціальної підробки: алгоритм повинний підтримувати імовірність підробки підпису на певному рівні (наприклад, 2^{40} операцій). Мінімум декілька часів, що потрібні для підробки підпису, може виявитися достатньо для відлякування потенційних порушників.
- Мінімізація розміру: поштова система може мати серйозні обмеження щодо розміру підпису. Зокрема, оптичні зчитувачі штрих-кодів більш ефективно працюють з даними невеликих об'ємів.
- Максимальна ефективність обчислення повинна бути підтримана на такому рівні, щоб мати змогу робити декілька підписів/перевірянь підпису в секунду на сучасному обладнанні.
- Підпис повинен містити в собі всю інформацію, необхідну для перевірки, не підключаючи зовнішні джерела.
- Конфіденційність: певні частини повідомлення мають бути криптографічно захищені від несанкціонованого ознайомлення.

Отже, на етапі підготовки до підпису здійснюється розбиття повідомлення, що підписується (PD) на 2 частини $C||V$, де C відображає дані, що повинні бути криптографічно захищені, V – відкриті дані. Цілісність відкритих даних також забезпечується, бо вони також підписуються. Генерація проходить за такими кроками:

1. $$R = kP,$$

де R є точкою на обраній кривій, яка повинна бути інтерпретована у бітову строку для наступної трансформації;

2. $$c=Tr_R(C), \qquad (4)$$

де Tr_R – трансформація, що параметризується раніше обрахованим значенням R і сконструйована для того, щоб знищити будь-яку алгебраїчну структуру, яку C може мати. Як трансформація може використовуватися алгоритм симетричного шифрування, такий як AES або проста операція XOR ключа та секретних даних;

3. $h=H(e||I||V)$,

де H – криптографічна хеш-функція, а I – ідентифікатор відправника;

4. $d=ah+k(\mathrm{mod}\, n)$,

де a – секретний ключ відправника;

5. пара чисел (c, d) є цифровим підписом.

Алгоритм генерації підпису стає набагато ефективнішим, якщо бітова довжина даних, що потребують конфіденціального захисту, менша або дорівнює бітовій довжині R, коли можна застосувати операцію XOR до цих даних. Якщо ж секретні дані перевищують по довжині ключ, то використовувати XOR з повторенням гами не слід для зберігання належного рівня криптостійкості.

Для перевірки підпису отримувач дістає такий пакет даних: ідентифікатор I відправника, підпис (c,d) та відкриті дані V. Для перевірки проводяться наступні обрахунки:

1. $U = sP - dQ$,

де Q – відкритий ключ відправника, а $d=H(e||I||V)$;

2. Виконується обернена до (4) трансформація $X=Tr_{U^{-1}}(c)$;

3. Перевірити надлишковість X, та якщо X має потрібну надлишковість (40 біт) прийняти підпис.

У цьому алгоритмі параметр надлишковості має велике значення. Вибір його розміру має випливати з рівня безпеки відносно атаки правдоподібної підробки. До переваг слід віднести змінну величину відновлюваної частини повідомлення.

З огляду на принципи побудови схеми криптостійкість обумовлена низкою факторів.

1. Безпекою групи точок еліптичної кривої, зокрема можливістю обчислення дискретного логарифму в цій групі.
2. Стійкістю до колізій хеш-функції.
3. Стійкістю шифру, що застосовується як трансформація у п. 2.
4. Величиною надлишкових даних, що додаються до приховуваних даних.

У роботі [7] розглянуто схожу схему, але її опис дано в термінах «спареної» криптографії, тобто криптографії, заснованої на спаруванні різних груп чисел в одному алгоритмі. Як приклад такого спарування можна привести і групу точок еліптичної кривої.

У цій схемі дано секретний ключ s та відкритий $Q = sP$, де P – породжуючий елемент групи. Нехай ми

маємо наступні хеш-функції $H_1 : \{0,1\} \to G_1$, тобто функцію, що відображає бінарні дані у елемент групи *G1*; $F_1 : \{0,1\}^{k_2} \to \{0,1\}^{k_1}$, тобто функцію, що відображає послідовність з k_2 елементів у послідовність з k_1 елементів; $F2 : \{0,1\}^{k_1} \to \{0,1\}^{k_2}$ - обернену попередній. Нехай повідомленням є $m \in \{0,1\}^{k_2}$. Тоді підпис будується аналогічним [6] чином, але замість симетричної трансформації здійснюються операції хешування: $f = F_1(m) \| (F_2(F_1(m)) \oplus m)$, при цьому довжина f дорівнює $k_1 + k_2 = q$ біт. При цьому повідомлення приховується повністю у підписі і воно повинно мати довжину, не більшу k_2.

Однак цю схему можна модифікувати і таким чином, щоб можна було приховувати і частину повідомлення. Що і зроблено у статті [8]:

$$f = F_1(m_2) \| (F_2(F_1(m_2)) \oplus m_2),$$

де m_2 – частина повідомлення, що потребує приховування, а хеш-функції визначені, як і в попередньому прикладі.

Відкрита частина m_1 ураховується далі при формуванні підпису, тобто цілісність тієї частини теж підтверджується. Тобто у цій схемі можна приховати дані довжиною k_2, хоча саме повідомлення, що підписується може бути більшого розміру.

Порівняння розглянутих алгоритмів формування ЕЦП

На основі матеріалів попереднього підрозділу складено таблицю для порівняння алгоритмів за одними і тими ж ознаками. Алгоритми побудовані таким чином, що одні і ті ж параметри майже усюди еквівалентні за значенням, що забезпечує наглядність порівняння. В табл. 3 використовуються такі позначення:

- IF – проблема факторизації великих чисел;
- DL – проблема дискретного логарифмування в скінченному полі;
- ECDL – проблема дискретного логарифмування в групі точок еліптичної кривої;
- *K* – довжина ключа;
- *q* – порядок циклічної підгрупи базової точки для алгоритмів, заснованих на дискретному логарифмуванні; має довжину, що дорівнює розміру скінченного поля F_P;
- H(M) – хеш-функція від вхідного повідомлення *M*; якщо в алгоритмі декілька хеш-функцій, то до позначення дописується порядковий номер ($H_1(M)$);
- LH(M) – довжина вихідного значення хеш-функції $H(M)$;
- *G* – точка на еліптичній кривій (довжина дорівнює $2q$).

Зауваження: розмір ключів розглядався у сенсі забезпечуваного рівня безпеки, що дорівнює 112 біт, який з 2012 р. рекомендовано в криптографічних алгоритмах. Довжина підпису, наведена в таблиці, від рівня безпеки не залежить, а показана залежно від параметрів самого алгоритму.

Таблиця 3 – Порівняння алгоритмів ЕЦП

Назва методу	**Проблема, на якій базується**	**Довжина підпису, біт**	**Розмір ключів, біт**	**Додаткові параметри алгоритму**	**Відновлення повідомлення/ обмеження довжини**
RSA	IF	2*K*	2048	-	+ / *L*(*M*)
DSA	DL	2q	2048	*H*(*M*)	-
ElGamal	DL	2*K*	2048	*H*(*M*)	-
Shnorr	DL	*LH*(*M*)+*q*	2048	*H*(*M*)	-
Nyberg-Rueppel (p-NEW)	DL	*K*+*q*	2048	-	+ / ≤ *q*
ECDSA	ECDL	2*q*	224	*H*(*M*)	-
ГОСТ Р- 34.10-2001	ECDL	2*q*	224	*H*(*M*)	-
Zhang	ECDL	*q*+*G*/3*q*	224	$H_3(M)=H_1(M)+H_2(M)$	+ / ≤(*q*-$LH_1(M)$)

IV Висновки

Проведений аналіз асиметричних схем формування ЕЦП, заснованих на проблемі дискретного

логарифмування над скінченими полем та еліптичними кривими, в результаті складена порівняльна таблиця цих алгоритмів. Описані базові стандарти, такі як DSA, ElGamal, ECDSA, ГОСТ Р 34.10-2001. Окрім цього розглядалися алгоритми, які аналогічно алгоритму RSA надають можливості відновлення повідомлення безпосередньо при верифікації цифрового підпису. Вони вирішують важливу задачу підпису – забезпечення, окрім цілісності даних, їх конфіденціальний захист. Проте можливості щодо відновлення всіх цих алгоритмів виявилися обмеженими: вони або не підходять для відновлення довгих повідомлень, коли потрібно вираховувати послідовно кілька підписів або один, що має великий розмір (схема p-NEW), або просто не були здатні на відновлення повідомлень довільної довжини (або їх частини довільної довжини) – схема Zhang.

Література: ***1.*** *Брюс Шнайер. Прикладная криптография. Протоколы, алгоритмы, исходные тексты на Си. - М.: Изд. ТРИУМФ, 2008. – 816 с.* ***2****. Nyberg K., Rueppel R. A New Signature Scheme Based on the DSA Giving Message Recovery / K. Nyberg, R. Rueppel // 1st ACM Conference on Computer and Communications Security. – Springer-Verlag, 1998. – P. 182 – 193.* ***3.*** *Ateniese, G., Medeiros, B.D. A Provably Secure Nyberg-Rueppel Signature Variant with Applications./ G. Ateniese, B.D Medeiros // IACR Cryptology ePrint Archive 2004.– 2004. – P. 93-110.* ***4****. Sujata M., Banshidhar M. A Digital Signature Scheme with Message Recovery and without One-Way Hash Function / M. Sujata, M. Banshidhar // International Conference on Advances in Computer Engineering. – Bangalore, 2010. – P. 265-267.* ***5.*** *Miyaji A. A message recovery signature scheme equivalent to DSA over elliptic curves // In Proceedings of ASIACRYPT'96. – Springer-Verlag, 1996. – P. 1-14.* ***6.*** *Pintsov L., Vanstone S. Postal Revenue Collection in the Digital Age / L. Pintsov, S. Vanstone // Financial Cryptography, Lecture Notes in Computer Science 1962. – Springer, 2000. – P. 105-120.* ***7.*** *Zhang. F. Identity-based partial message recovery signatures (or How to shorten ID-based signatures) / F. Zhang, W. Susilo, Y. Mu // Lecture Notes in Computer Science, 3570. – 2005. – P. 45–56.* ***8.*** *An Efficient ID-based Digital Signature with Message Recovery Based on Pairing / Raylin Tso, Chunxiang Gu, Takeshi Okamoto, and others // IACR Cryptology ePrint Archive, Vol. 2006. – 2006/195.* ***9.*** *Мао В. Современная криптография: теория и практика. – Пер. с англ. – М.: Издательский дом Вильямс, 2005. – 768 с.*

УДК 004.056.5

АНАЛИЗ И ОЦЕНКА НОРМАТИВНЫХ ДОКУМЕНТОВ, ПРИМЕНЯЕМЫХ ДЛЯ ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ SMART GRID СИСТЕМ

Алексей Юдин, Глеб Пирогов
ГосНИИ Спецсвязи

Аннотация: **Статья описывает требования и нормы, устанавливаемые нормативными документами, применяющимися в мире для обеспечения информационной безопасности Smart Grid.**
Abstract: **This paper describes the requirements and standards set by regulations which apply in the world of information security Smart Grid.**
Ключевые слова: **Информационная безопасность, Smart Grid, международный стандарт.**

Введение

Актуальность проведения анализа зарубежных нормативных документов, касающихся информационной безопасности, вызвана тем, что на сегодня в мире Smart Grid системы только начинают создаваться и развиваться, а в Украине они вообще нормативно не закреплены и не описаны. Под термином Smart Grid будем понимать электрическую сеть которая использует информационные и коммуникационные технологии, а также информацию о поведении поставщиков и потребителей, с целью автоматизации процесса улучшения продуктивности, надежности, экономичности и стойкости производства и распространения электрической энергии. Также, необходимо подчеркнуть, что внедрение Smart Grid систем направлено на обеспечение энергетической безопасности Украины, то есть способности государства обеспечить максимально надежное, технически безопасное, экологическое и обоснованно достаточное энергообеспечение экономики и населения [1].

В статье будем рассматривать Smart Grid систему как совокупность подсистемы передачи электрической энергии и информационно-телекоммуникационной подсистемы [2 – 4] и в соответствии с этим осуществлять анализ существующих нормативных документов.