УДК 681.3.06

І.Д. ГОРБЕНКО, П.О. КРАВЧЕНКО

Харківський національний університет радіоелектроніки, Україна

БЕЗПЕКА КОМБІНОВАНОЇ СХЕМИ ІНФРАСТРУКТУРИ ВІДКРИТИХ КЛЮЧІВ ДЛЯ МОДЕЛІ ВИПАДКОВОГО ОРАКУЛА

Наводиться комбінована схема шифрування на ідентифікаторах. Доводиться, що в моделі випадкового оракула схема є стійкою проти атаки з підібраними відкритими текстами, за умови якщо проблема вирішення варіанту обчислювальної проблеми Діффі-Гелмана є складною. Запропонована комбінована схема шифрування поєднує кращі властивості шифрування на ідентифікаторах та традиційного шифрування з відкритими ключами та є безпечною проти атаки з підібраними відкритими текстами в моделі випадкового оракула за умови, якщо проблема вирішення обчислювальної проблеми Діффі-Гелмана є складною.

***Ключові слова**: семантична стійкість, модель випадкового оракула, ідентифікатор, таємний ключ, спарювання.*

Вступ

Схема шифрування з відритим ключем, в якості якого могла бути довільна строка, була запропонована Шаміром у 1984 р. Схема складається з чотирьох алгоритмів: (1) setup – генерує мастер-ключ та глобальні системні параметри, (2) extract – використовує мастер-ключ для генерації таємного ключа відповідного довільній строчці $ID \in \{0,1\}^n$, (3) encrypt – шифрує повідомлення за допомогою відкритого ключа ID, та (4) decrypt – розшифровує повідомлення за допомогою таємного ключа.

Метою пропозиції Шаміра було спрощення процедури управління сертифікатами та поштових систем. Коли А надсилає Б електронний лист у company.ua вона просто шифрує своє повідомлення на відкритому ключі bob@company.ua. Для такої схеми немає необхідності в отриманні сертифікату відкритого ключа Б. Коли Б отримує лист, він робить запит до третьої сторони, яку ми називаємо Уповноважений на генерацію ключів (УГК). Потім Б проходить автентифікацію таким самим чином, як і у центрі сертифікації та отримує свій таємний ключ.

Відносно такої схеми необхідно відмітити два недоліки. Перший, що ключ відомий не тільки користувачу а й УГК, тому він має можливість розшифровувати повідомлення будь-якого користувача. Другий – УГК повинен надсилати користувачу таємний ключ і для цього необхідно використовувати захищений канал зв'язку, що робить розподілення ключів більш складним.

Така схема була представлена в [2] і названа IDComby. В ній поєднані переваги шифрування на ідентифікаторах та традиційного шифрування з відкритими ключами. У цій статті наводиться доказ безпеки вказаної схеми для моделі випадкового оракула.

1. Визначення

Доказ безпеки схеми IDComby грунтується на використанні поняття семантична стійкість [3].

Задамо зловмисника А як IND-ID-CPA зловмисника. Перевага IND-ID-CPA зловмисника проти певної схеми В є функцією від параметра безпеки k: $\mathrm{Adv}_{\varepsilon,A}(k) = |\Pr[b = b'] - \frac{1}{2}|$.

Визначення 1. Система шифрування на ідентифікаторах є семантично стійкою якщо для будь-якого поліноміально обмеженого IND-ID-CPA зловмисника функція $\mathrm{Adv}_{\varepsilon,A}(k)$ є незначною. Для умов визначення 1 в подальшому будемо говорити, що В є IND-ID-CPA стійка.

Для аналізу безпеки конкретних криптографічних схем, Белларе та Роговей ввели поняття ідеалізованої моделі безпеки, яка має назву модель випадкового оракула [4].

Задамо IDComby схему (рис.1) шляхом визначення шести алгоритмів: Setup, SetKeyPair, Certify, Extract, Encrypt, Decrypt. Нехай k буде параметром безпеки алгоритму setup, а Ω генератором BDH параметрів.

Setup. Алгоритм отримує на вхід параметр безпеки $k \in Z^+$ та формує системні параметри:

$$< q, G_1, G_2, \hat{e}, n, P, P_{PUB}, H_1, H_2 >, \qquad (1)$$

де q – велике просте число,

G_1, G_2 – групи порядку q,

$\hat{e}$ – білінійне відображення $G_1 \times G_1 \to G_2$,

P – генератор групи G_1,

P_{PUB} – відкритий ключ,

H_1, H_2 – геш функції.

SetKeyPair. Вибирається випадкове $s_1 \in Z_q^*$ та обчислюється s_1P – відкритий ключ.

Certify. Вхідні дані $< s_1P, t, ID_{s1} >$ підписуються на таємному ключі s_{CA} та формується сертифікат відкритого ключа.

Extract: Для строки $ID \in \{0,1\}^*$ виконується алгоритм: (1) обчислюється $Q_{ID} = H_1(ID) \in G_1^*$ та (2) встановлюється таємний ключ шифрування $d_{ID} = s_1P, ID_{s1}$ де s_1 таємний ключ.

Encrypt. Для за шифрування повідомлення M на відкритому ключі ID_{s1}, (1) обчислюється $Q_{ID} = H_1(ID) \in G_1^*$, (2) вибирається випадкове $r \in Z_q^*$ та (3) обчислюється криптограма згідно формули

$$<rP, M \oplus H_2(\hat{e}(s_1P, H_1(ID_{s1}))^r)>,$$

де s_1P, ID_{s1} – відкритий ключ та відкритий ідентифікатор одержувача відповідно.

Decrypt. Для вхідного повідомлення $C =< U, V >$, яке було зашифроване на відкритому ідентифікаторі ID_{s1}, відкрите повідомлення отримується згідно формули: $M = V \oplus H_2(\hat{e}(U, s_1H(ID_{s1})))$, де $U = rP$.

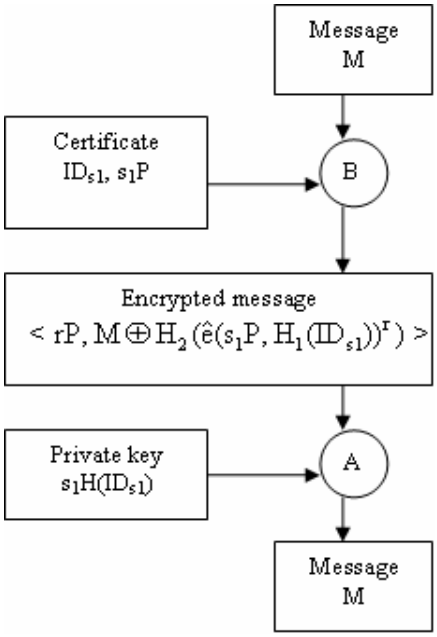

Рис. 1. IDComby схема

2. Доказ стійкості

Далі, дослідимо стійкість цієї схеми. Наступна теорема показує, що схема IDComby шифрування є семантично стійкою схемою на ідентифікаторах (IND-ID-CPA) за умови якщо BDH є складною в групах, що генеруються G.

Теорема 1. Припустимо, що геш-функції H_1, H_2 є випадковими оракулами. Тоді IDComby є семантично стійкою схемою на ідентифікаторах (IND-ID-CPA), якщо BDH є складною в групах, що генеруються G. Тобто, припустимо, що є IND-ID-CPA атакуючий, що має перевагу $\varepsilon(k)$ проти схеми IDComby. Припустимо, що A робить якнайбільше $q_E > 0$ запитів на отримання таємного ключа, та $q_{H_2} > 0$ запитів до H_2. Тоді існує алгоритм B, що вирішує BDH у групі, що генерується G з перевагою якнайменше:

$$Adv_{G,B}(k) \geq \frac{2\varepsilon(k)}{e(1+q_E)\cdot q_{H_2}}. \qquad (2)$$

Приймемо що час роботи B дорівнює $O(time(A))$.

Щоб довести цю теорему визначимо схему шифрування з відкритими ключами, що називається BasicPub. BasicPub описується трьома алгоритмами: keygen, encrypt, decrypt.

Keygen: Алгоритм отримує на вхід параметр безпеки $k \in Z^+$.

Крок 1. Алгоритм генерує дві групи G_1, G_2 порядку q та білінійне відображення $\hat{e}: G_1 \times G_1 \to G_2$. Вибирається випадковий генератор групи $P \in G_1$.

Крок 2. Вибирається випадкове $s \in Z_q^*$ як таємний ключ та обчислюється $P_{pub} = sP$ – відкритий ключ. Обирається випадкове $Q_{ID} \in G_1^*$

Крок 3. Вибирається криптографічна геш-функція $H_2: G_2 \to \{0,1\}^n$ для деякого n.

Крок 4. Відкритий ключ є $< q, G_1, G_2, \hat{e}, n, P, P_{pub}, Q_{ID}, H_2 >$. Таємний ключ $d_{ID} = sQ_{ID}$. Для шифрування повідомлення $M \in \langle M \rangle$ вибирається випадкове $r \in Z_q^*$ та обчислюється криптограма $C = <rP, M \oplus H_2(g^r)>$, де $g = \hat{e}(P_{pub}, Q_{ID})$.

Нехай $C =< U, V >$ було зашифроване на відкритому ключі, а повідомлення отримується за наступною формулою:

$$M = V \oplus H_2(\hat{e}(U, d_{ID})). \qquad (3)$$

Це завершує описання BasicPub. Далі доведемо теорему у два Кроки. Спочатку покажемо, що IND-ID-CPA атака на IDComby може бути перетворена на IND-CPA атаку на BasicPub. Цей Крок покаже, що запити таємних ключів не допоможуть атакуючому. Далі покажемо, що BasicPub є IND-CPA безпечною, якщо BDH є складною.

Лема 1. Нехай H_1 буде випадковим оракулом з $\{0,1\}^*$ у G_1^*. Припустимо, що A є атакуючим, що має перевагу $\varepsilon(k)$ проти схеми IDComby. Припустимо, що A робить якнайбільше $q_E > 0$ запитів на отримання таємного ключа. Тоді існує IND-CPA атакуючий B, що має перевагу якнайменше $\frac{\varepsilon(k)}{e(1+q_E)}$ проти BasicPub. Час роботи B дорівнює $O(time(A))$.

Доведення. Покажемо, як побудувати IND-CPA атакуючого B, що використовує A для отримання переваги $\frac{\varepsilon(k)}{e(1+q_E)}$ проти BasicPub. Гра між відповідаючим та атакуючим B починається, коли відповідаючий генерує випадковий відкритий ключ, використовуючи алгоритм keygen з BasicPub. Результатом є $< q, G_1, G_2, \hat{e}, n, P, P_{pub}, Q_{ID}, H_2 >$ та таємний ключ $d_{ID} = sQ_{ID}$. Відповідаючий віддає K_{pub} алгоритму B. Алгоритм B повертає два повідомлення M_0 та M_1 та очікує отримати BasicPub зашифроване повідомлення M_b на K_{pub} де $b \in \{0,1\}$. Тоді алгоритм B повертає гіпотезу $b' \in \{0,1\}$ для b.

Алгоритм B працює, взаємодіючи з A у IND-ID-CPA грі наступним чином (B симулює відповідаючого для A).

Установка: Алгоритм B дає A системні параметри схеми IDComby –

$$params =< q, G_1, G_2, \hat{e}, n, P, P_{pub}, H_1, H_2 > .$$

Тут $q, G_1, G_2, \hat{e}, n, P, P_{pub}, H_2$ беруться з K_{pub}, а H_1 – випадковий оракул, що контролюється B як показано нижче.

H_1-запити: В деякий час алгоритм A може здійснити запит до випадкового оракула H_1. Для відповіді на ці запити алгоритм B підтримує список кортежів $< ID_j, Q_j, b_j, c_j >$ як описано нижче. Цей список ми позначимо як H_1^{list}. Спочатку список пустий.

Коли A робить запит ID_i до оракула H_1, алгоритм B відповідає:

1. Якщо запит ID_i вже присутній у H_1^{list} у кортежі $< ID_i, Q_i, b_i, c_i >$, алгоритм B відповідає значенням $H_1(ID_i) = Q_i \in G_1^*$.

2. Якщо його немає, то B генерує використовуючи випадкову монету $coin \in \{0,1\}$ таку, що $Pr[coin = 0] = \delta$ для деякої δ, що буде визначено пізніше.

3. Алгоритм B обирає випадкове $b \in Z_q^*$. Якщо $coin = 0$, то обчислюється $Q_i = bP \in G_1^*$. Якщо $coin = 1$, то обчислюється $Q_i = bQ_{ID} \in G_1^*$.

4. Алгоритм B додає кортеж $< ID_i, Q_i, b, coin >$ у H_1^{list} та відповідає A значенням $H_1(ID_i) = Q_i$. Відмітимо, що Q_i мають рівномірне розподілення у G_1^* та незалежні з точки зору A.

Фаза 1: Нехай ID_i буде запитом на отримання таємного ключа, що надходить від алгоритму A.

Алгоритм B відповідає на цей запит наступним чином.

Для відповіді на H_1-запит для отримання $Q_i \in G_1^*$ виконується попередній алгоритм, причому $H_1(ID_i) = Q_i$. Нехай $< ID_i, Q_i, b_i, coin_i >$ буде відповідним кортежем у H_1^{list}. Якщо $coin_i = 1$, то B повідомляє про помилку та процес завершується. Атака на BasicPub закінчується невдачею.

Також відомо, що $coin_i = 0$ і отже $Q_i = b_i P$. Позначимо $d_i = bP_{pub} \in G_1^*$. Відмітимо, що $d_i = sQ_i$, тоді d_i – відповідний таємний ключ для ID_i, який повертається алгоритму A.

Перевірка: Алгоритм A вирішує, що фаза 1 пройдена та виробляє відкритий ключ ID_{ch} а також два повідомлення M_0, M_1 на котрих він хоче бути перевіреним на легальність. Алгоритм B відповідає таким чином:

1. Алгоритм B дає challenger два повідомлення M_0, M_1. Challenger відповідає BasicPub шифртекстом $C =< U, V >$ таким, що C – зашифроване повідомлення M_c для випадкового $c \in \{0,1\}$.

2. Далі, B запускає алгоритм для відповіді на H_1-запити для отримання $Q \in G_1^*$ такого, що $H_1(ID_{ch}) = Q$. Нехай $< ID_{ch}, Q, b, coin >$ буде відповідним кортежем у H_1^{list}. Якщо Якщо $coin_i = 0$, B повідомляє про помилку та завершується. Атака на BasicPub закінчується невдачею.

3. Відомо, що $coin_i = 1$ і отже $Q = bQ_{ID}$. Відмітимо, що коли $C =< U, V >$ то $U \in G_1^*$. Встановимо $C^{/} =< b^{-1}U, V >$, де b^{-1} – інверсія $b \pmod q$. Алгоритм B відповідає A challenge шифртекстом $C^{/}$.

Фаза 2: Алгоритм B відповідає на запити отримання таємного ключа як і в Фазі 1.

Догадка: Наприкінці, алгоритм A видає гіпотезу $c^{/}$ для c. Алгоритм B видає $c^{/}$ як гіпотезу для c.

Твердження: Якщо алгоритм B не преривався на протязі симуляції, то з погляду A, це була реальна атака. Більш того, якщо B не переривався, то:

$$| \Pr[c = c^{/}] - \frac{1}{2} | \geq \varepsilon . \qquad (4)$$

Доведення. Відповіді на H_1-запити такі самі як і в реальній атаці, тому що кожна відповідь рівномірно та незалежно розполідена над G_1^*. Усі відповіді на запити отримання таємного ключа коректні. Наприкінці, challenge шифртекст $C^{/}$ даний A є ID-Comby шифртектом повідомлення M_c для деякого випадкового $c \in \{0,1\}$. Тому, з визначення алгоритму A отримаємо

$$| \Pr[c = c^{/}] - \frac{1}{2} | \geq \varepsilon . \qquad (5)$$

Для завершення доказу леми 1 залишається обчислити ймовірність, з якою алгоритм B преривається під час симуляції. Припустимо, що A робить q_E запитів на отримання таємного ключа. Тоді ймовірність того, що B не перерветься у фазі 1 чи фазі 2 є δ^{q_E}. Ймовірність, що він не перерветься на кроці догадки є $1 - \delta$. Тоді, ймовірність того, що B не перерветься на протязі симуляції дорівнює $\delta^{q_E}(1-\delta)$. Це значення максимальне у $\delta_{opt} = 1 - \frac{1}{q_E + 1}$. Використовуючи δ_{opt}, ймовірність того, що B не перерветься, якнайменше $\frac{1}{e(q_E + 1)}$. Це показує, що перевага B якнайменше $\frac{\varepsilon}{e(q_E + 1)}$, що і необхідно було довести.

Далі, покажемо, що BasicPub семантично стійка система з відкритим ключем, якщо BDH припущення виконується.

Лема 2. Нехай H_2 буде випадковим оракулом з G_2 у $\{0,1\}^n$. Припустимо, що A є c IND-CPA атакуючим, що має перевагу $\varepsilon(k)$ проти схеми BasicPub. Припустимо, що A робить якнайбільше $q_{H_2} > 0$ запитів до H_2. Тоді існує алгоритм B, що вирішує проблему BDH для G з перевагою якнайменше $\frac{2\varepsilon(k)}{q_{H_2}}$ за час $O(time(A))$.

Доведення. Алгоритм B отримує на вхід BDH параметри $< q, G_1, G_2, \hat{e} >$, що вироблені G та випадковий кортеж $< P, aP, bP, cP >=< P, P_1, P_2, P_3 >$ BDH проблеми для цих параметрів, тобто P є випадковим у G_1^* та a, b, c є випадковими у Z_q^*, де $*$ – це порядок Z. Нехай $D = \hat{e}(P,P)^{abc} \in G_2$ буде рішенням BDH проблеми. Алгоритм B знаходить D шляхом взаємодії з A наступним чином:

Установка: Алгоритм B обчислює BasicPub відкритий ключ – $K_{pub} =< q, G_1, G_2, \hat{e}, n, P, P_{pub}, Q_{ID}, H_2 >$, встановлюючи $P_{pub} = P_1$ та $Q_{ID} = P_2$. Тут H_2 – випадковий оракул, що контролюється B так, як описано нижче. Алгоритм B дає A K_{pub} – відкритий ключ схеми BasicPub. Відмітимо, що (невідомий) таємний ключ для K_{pub} це $d_{ID} = aQ_{ID} = abP$.

H_2-запити: В деякий час алгоритм A може здійснити запит до випадкового оракула H_2. Для відповіді на ці запити алгоритм B підтримує список кортежів $< X_j, H_j >$. Цей список ми позначимо як H_2^{list}. Спочатку список пустий. Коли A робить запит X_i до оракула H_2, алгоритм B відповідає.

1. Якщо запит X_i вже присутній у H_2^{list} у кортежі $< X_i, H_i >$, алгоритм B відповідає значенням $H_2(X_i) = H_i$.

2. Якщо ні, B вибирає випадкову строку $H_i \in \{0,1\}^n$ та додає кортеж $< X_i, H_i >$ до H_2^{list}. Він відповідає A значенням $H_2(X_i) = H_i$.

Перевірка: Алгоритм A виробляє два повідомлення M_0, M_1 на котрих він хоче бути перевіреним на легальність. Алгоритм B обирає випадкову строку $R \in \{0,1\}^n$ та обчислює шифртекст C таким як $C =< P_3, R >$. Алгоритм B віддає C як challenge до A. Відмітимо, що за визначенням, шифртекст C дорівнює $R \oplus H_2(\hat{e}(P_3, d_{ID})) = R \oplus H_2(D)$.

Догадка: Алгоритм A повертає його гіпотезу $c^{/} \in \{0,1\}$. У цій точці B обирає випадковий кортеж $< X_j, H_j >$ з H_2^{list} та повертає X_j як вирішення конкретної BDH задачі.

Алгоритм B симулює реальне середовище для A (він симулює challenger та оракула H_2). Ми покажемо, що алгоритм B повертає коректну відповідь D з ймовірністю якнайменше $\frac{2\varepsilon}{q_{H_2}}$. Доказ базується на порівнянні поведінки A у симуляції та у реальній IND-CPA грі (проти реального challenger та реального випадкового оракула H_2).

Нехай H буде подією, згідно якої алгоритм A робить запит для $H_2(D)$ під час симуляції (це означає, що у кінці симуляції D з'являється у деякому кортежі у H_2^{list}). Ми покажемо, що $Pr[H] \geq 2\varepsilon$. Це означає , що алгоритм B повертає D з ймовірністю якнайменше $\frac{2\varepsilon}{q_{H_2}}$. Також вивчимо подію H у реальній грі, тобто подію, що A робить запит для $H_2(D)$ під час взаємодії з реальним challenger та реальним випадковим оракулом H_2.

Твердження 1: $Pr[H]$ під час симуляції дорівнює $Pr[H]$ у реальній атаці.

Доведення. Нехай H_l буде подією, що A робить запит $H_2(D)$ в одному з перших l запитів до H_2 оракула. Доведемо методом індукції по l, що $Pr[H_l]$ у реальній атаці дорівнює $Pr[H_l]$ під час симуляції для усіх $l \geq 0$. Очевидно, що $Pr[H_0] = 0$ як для симуляції так і для реальної атаки. Тепер припустимо, що для деякого $l > 0$ для реальної атаки маємо. Покажемо, що умова виконується і для H_l. Відомо , що:

$$\begin{aligned} &Pr[H_l] = Pr[H_l \mid H_{l-1}]Pr[H_{l-1}] \\ &+ Pr[H_l \mid \neg H_{l-1}]Pr[\neg H_{l-1}] = \\ &= Pr[H_{l-1}] + Pr[H_l \mid \neg H_{l-1}]Pr[\neg H_{l-1}] \end{aligned} \quad (6)$$

Доведемо, що $Pr[H_l \mid \neg H_{l-1}]$ під час симуляції реальної атаки дорівнює $Pr[H_l \mid \neg H_{l-1}]$. Щоб перевірити це, відмітимо, що доки A не зробить запит $H_2(D)$, з його точки зору симуляція не відрізняється від реальної атаки. Дійсно, відкритий ключ та challenge розподілені так як і у реальній атаці. Усі відповіді на H_2 – запити рівномірно розподілені та незалежні у $\{0,1\}^n$. Тому $Pr[H_l \mid \neg H_{l-1}]$ під час симуляції дорівнює $Pr[H_l \mid \neg H_{l-1}]$ під час реальної атаки. Індукція по l показує, що $Pr[H_l]$ під час симуляції та реальної атаки співпадають.

Твердження 2: У реальній атаці маємо, що $Pr[H] \geq 2\varepsilon$.

Доведення. У реальній атаці, якщо A ніколи не робить запит $H_2(D)$, то розшифрування C незалежно з точки зору A (так як $H_2(D)$ незалежно з точки зору A). Тобто, у реальній атаці $Pr[c = c' \mid \neg H] = 1/2$. Згідно з визначенням A, відомо, що у реальній атаці $|Pr[c = c' \mid -1/2] \geq \varepsilon$. Покажемо, що ці два факта підтверджують, що $Pr[H] \geq 2\varepsilon$. Щоб зробити це, спочатку отримаємо верхню та нижню границі для $Pr[c = c']$:

$$\begin{aligned} &Pr[c = c'] = Pr[c = c' \mid \neg H]Pr[\neg H] + \\ &+ Pr[c = c' \mid H]Pr[H] \leq Pr[c = c' \mid \neg H]Pr[\neg H] + \\ &+ Pr[H] = \frac{1}{2} + \frac{1}{2}Pr[H]. \end{aligned} \quad (7)$$

$$\begin{aligned} &Pr[c = c'] \geq \\ &\geq Pr[c = c' \mid \neg H]Pr[\neg H] = \frac{1}{2} - \frac{1}{2}Pr[H]. \end{aligned} \quad (8)$$

З цього виходить, що

$$\varepsilon \leq |Pr[c = c'] - \frac{1}{2}| \leq \frac{1}{2}Pr[H].$$

Тому у реальній атаці $Pr[H] \geq 2\varepsilon$.

Для завершення доказу леми 2 відмітимо, що з твердження 1 та 2 для симуляції ми отримали, що $Pr[H] \geq 2\varepsilon$. Тому наприкінці симуляції D з'являється у H_2^{list} з ймовірністю якнайменше 2ε. Тому B виробляє коректну відповідь з ймовірністю якнайменше $\frac{2\varepsilon}{q_{H_2}}$.

Як підсумок доказ теореми 1. Теорема 1 безпосередньо витікає з лем 1 та 2. Об'єднання обох редукцій показує, що IND-ID-CPA атакуючий на IDComby з перевагою $\varepsilon(k)$ знаходить рішення BDH з перевагою якнайменше:

$$\frac{2\varepsilon(k)}{e(1+q_E)q_{H_2}}.$$

Висновки

Аналіз безпеки схеми на ідентифікаторах ID-Comby підтвердив її заявлені властивості. Таким чином IDComby схема поєднує кращі властивості шифрування на ідентифікаторах та традиційного шифрування з відкритими ключами. Показано, що IDComby є безпечна проти атаки з підібраними відкритими текстами в моделі випадкового оракула за умови, якщо проблема вирішення обчислювальної проблеми Діффі-Гелмана є складною.

Література

1. Shamir A. Identity-based cryptosystems and signature schemes / A. Shamir. – Advances in Cryptology – Crypto '84, Lecture Notes in Computer Science. – Vol. 196. – Springer-Verlag, 1984. – P. 47-53.

2. Бондаренко М. Комбінована інфраструктура відкритих ключів / М. Бондаренко, П. Кравченко // Прикладна радіоелектроніка. – 2009. – Т.5. – С. 327–329.

3. Bellare M. Relations among notions of security for public-key encryption schemes / M. Bellare, A. Desai, D. Pointcheval, P. Rogaway // Advances in Cryptology. – Crypto '98, Lecture Notes in Computer Science. – Vol. 1462. – Springer-Verlag, 1998. – P. 26-45.

4. Bellare M. Random oracles are practical: a paradigm for designing efficient protocols / M. Bellare, P. Rogaway // ACM conference on Computers and Communication Security. – 1993. – P. 62-73.

5. Boneh M. Identity based encryption from the Weil pairing / M. Boneh, M. Franklin // Advances in Cryptology – Crypto 2001. – Lecture Notes in Computer Science 2139. – Springer, 2001. – P. 213-229.

Поступила в редакцию 12.02.2010

Рецензент: д-р техн. наук О.В. Потій, Харківський національний університет радіоелектроніки, Україна.

БЕЗОПАСНОСТЬ КОМБИНИРОВАННОЙ СХЕМЫ ИНФРАСТРУКТУРЫ ОТКРЫТЫХ КЛЮЧЕЙ ДЛЯ МОДЕЛИ СЛУЧАЙНОГО ОРАКУЛА

И.Д. Горбенко, П.А. Кравченко

Приводится комбинированная схема шифрования на идентификаторах. Доказывается, что в модели случайного оракула схема стойкая против атаки с подобранными открытыми текстами при условии что решение варианта вычислительной проблемы Диффи-Хеллмана остается сложным. Предложенная комбинированная схема шифрования обьединяет наилучшие свойства шифрования на идентификаторах и традиционного шифрования с открытыми ключами.

Ключевые слова: семантическая стойкость, модель случайного оракула, идентификатор, секретный ключ, спаривание.

SECURITY OF COMBINED PUBLIC KEY INFRASTRUCTURE SCHEME IN RANDOM ORACLE MODEL

I.D. Gorbenko, P.O. Kravchenko

We propose a combined encryption scheme for identifiers. We prove that in the random oracle model our scheme resistant to chosen plaintext attack, assumed that the decision version of the computational Diffie-Hellman problem is difficult. The proposed combined encryption scheme combines the best features of identity-based encryption and the traditional public key encryption.

Keywords: semantic security, random oracle model, identity, private key, pairing.

Горбенко Іван Дмитрович – д-р техн. наук, проф., зав. кафедри безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, Україна, e-mail: gorbenko@kture.kharkov.ua.

Кравченко Павло Олександрович – аспірант кафедри безпеки інформаційних технологій, Харківський національний університет радіоелектроніки, Україна.