УДК 004.056.2

О. Я. Матов[1], В. С. Василенко[2], М. Ю. Василенко[2]
[1]Інститут проблем реєстрації інформації НАН України
вул. М. Шпака, 2, 03113 Київ, Україна
[2]Національний авіаційний університет
вул. Космонавта Комарова, 1, 03058 Київ, Україна

Матричні завадостійкі криптографічні перетворення

Запропоновано використання блокового криптографічного перетворення для задач забезпечення конфіденційності інформаційних об'єктів автоматизованих систем.

Ключові слова: *інформація, конфіденційність, криптографічні перетворення, завади, спотворення, відновлення.*

Вступ

У сучасних умовах забезпечення високої надійності, ефективності й технологічності автоматизованих систем (АС) є можливим тільки за умови забезпечення високого рівня захищеності інформації, що циркулює в цих АС. Для цього відповідно до законів України про інформацію і її захист, а також до нормативних документів Системи технічного захисту інформації (ТЗІ) України в АС необхідне застосування спеціальних засобів захисту, що призначаються для досягнення оптимального для даної АС об'єднання чотирьох **властивостей захищеності інформації автоматизованих систем** [1–3]: конфіденційності, цілісності, доступності та спостереженості. Залежно від умов застосування, складності та класу АС, а також характеристик можливих загроз, вага цих функціональних властивостей може змінюватись, але проблеми забезпечення конфіденційності та цілісності інформації є одними з основних при розробці й впровадженні будь-яких захищених АС. При цьому досить часто виникає задача одночасного забезпечення конфіденційності та цілісності одних і тих же інформаційних об'єктів. Причини цього можуть мати як суб'єктивний, так і об'єктивний характер.

Система ТЗІ забезпечує конфіденційність інформації, якщо вона зберігається, чи передається так, що сторонні (неавторизовані) користувачі не мають змоги отримати доступ до неї (за умови зберігання її у відкритому вигляді) [2] чи розкрити її смисловий зміст (за умови зберігання її у перетвореному вигляді) [4]. Звернемо увагу на те, що відсутність доступу до інформації не гарантує неможливість її отримання, наприклад, завдяки витокам інформації технічними каналами.

Окрім того, при зберіганні інформації у відкритому вигляді в багатокористувацьких АС можливе навмисне чи ненавмисне ознайомлення з конфіденційною інформацією тих авторизованих користувачів, для яких ця інформація не є призначеною. Отже в багатьох випадках криптографічне перетворення інформації є чи не єдиним шляхом забезпечення її конфіденційності (з певною стійкістю до спроб розкриття її змісту — криптографічною стійкістю). На цей час широко відомими є декілька алгоритмів криптографічного перетворення [4], з яких в Україні рекомендовано застосування алгоритму за стандартом ГОСТ 28147-89. При цьому деякі з алгоритмів криптографічного перетворення для зворотного перетворення потребують наявності лише неспотвореної інформації, тобто інформації з гарантованою цілісністю.

У свою чергу, система ТЗІ забезпечує цілісність інформації [2], якщо вона зберігається, передається чи обробляється достовірною, повною і захищеною від ненавмисних і навмисних спотворень. Одним із основних способів забезпечення цілісності інформації в автоматизованих системах є застосування засобів контролю цілісності інформаційних об'єктів з її подальшим відновленням. Не зупиняючись на причинах порушення цілісності [5], слід підкреслити, що частина загроз цілісності, зокрема внаслідок її порушень з боку авторизованих чи неавторизованих користувачів, може бути виявленою, а отже й усунутою лише за рахунок застосування ефективних механізмів контролю і відновлення цілісності, в яких використовуються процедури захищених від підробок перетворень інформації.

Це пов'язане з тим, що основною задачею забезпечення цілісності інформаційних ресурсів є підтримка такого стану системи, коли неможливе приховування факту будь-якої несанкціонованої модифікації захищеної інформації (вставки, вилучення, підміна і т.п.). З цією метою до складу інформації, що захищається, включають надлишкову інформацію — образ, відображення цієї інформації (ознака цілісності, цифровий підпис), процедура формування якого відома лише власнику інформації й авторизованим користувачам. Тобто образи, що формуються, повинні мати певну стійкість до підробок — імітостійкість. При цьому відомі механізми контролю цілісності з використанням цифрових підписів інформаційних об'єктів базуються на застосуванні процедур виявлення порушень цілісності — перевірки цифрового підпису і на наступному відновленні спотвореної інформації за рахунок повторних передач неспотвореної інформації чи повторних записів неспотвореної інформації з резервної копії. Обидві ці операції вимагають значних часових витрат. Для підвищення оперативності процесів забезпечення цілісності необхідною є розробка і застосування погоджених між собою швидкодіючих процедур як виявлення порушення цілісності інформації, так і її відновлення. Такими процедурами є процедури, що ґрунтуються на застосуванні коригувальних завадостійких кодів. Однак відомі завадостійкі коди не в змозі забезпечити головну з необхідних при цьому властивостей — імітостійкість, унаслідок чого їхнє використання в механізмах контролю цілісності є неможливим. Це пов'язано з тим, що механізми формування контрольних ознак, які можна було б використовувати як відповідні образи (сигнатур, геш-функцій і т.п.) не забезпечують скритності їхнього формування, тому що як константи (наприклад, елементи кодувальних таблиць, див. нижче), так і механізми обрахування цих кодів є, як правило, загальновідомими. В окремих випадках, коли таку скритність можна було б забезпечити

(приклад — коди Ріда–Соломона), кількість елементів перетворення (підматриць кодувальної матриці) є обмеженою настільки, що важко говорити про необхідну імітостійкість відповідних контрольних ознак.

Слід звернути увагу на те, що одночасне забезпечення і конфіденційності, і цілісності інформаційних об'єктів при використанні відомих алгоритмів досягається послідовним застосуванням процедур криптографічного перетворення і процедур обчислення цифрового підпису. При зворотному перетворенні спочатку перевіряється цілісність інформації, а потім здійснюється її дешифрування. Тобто ці процеси є двофазними і при прямому, і при зворотному перетворенні, за рахунок чого продуктивність засобів оброблення інформації дещо знижується.

У статті запропоновано використання механізмів, які дозволяють забезпечити як суміщення (однофазність) означених процедур, так і їхнє окреме застосування. Ці механізми використовують одну й ту ж математичну базу, що дозволяє розробити сімейство алгоритмів, які, на думку авторів, не поступаються, а в деяких випадках є кращими від відомих.

Кодові перетворення на базі матричної алгебри, код умовних лишків як основа таких перетворень

Під кодовими перетвореннями будемо розуміти результат множення вихідного коду A довжиною в n символів (слово визначеного алфавіту, число в деякій системі числення і т.п.), з можливим розширенням його до k символів, що розглядається як матриця розмірності $(1\times n)$, де n — число символів цього вихідного коду на кодувальну матрицю G розмірності $(k\times k)$, де $k \geq n$, елементами якої є деякі числа. Далі вважається, що операції множення і додавання при обчисленні елементів закодованого слова (при множенні матриць) можуть бути або лінійними, або нелінійними, наприклад, модульними — виконуються (усі чи окремі з них) за модулем (залежно від типу коду — малої чи великої величини), або логічними, в тому числі у вигляді порозрядних логічних додавань і множень.

У результаті такого множення одержують перетворений код — матрицю $B = A\times G$ розмірністю $(1\times k)$. Ясно, що для зворотного перетворення, тобто для одержання вихідного коду A із B досить виконати множення B на матрицю G^{-1}, зворотну G: $A = B\times G^{-1} = A\times G\times G^{-1}$. Матриця G у теорії завадостійкого кодування зветься породжуючою, а матриця G^{-1} — перевірочною.

Примітка: Звернемо увагу на те, що в разі використання нелінійних операцій (операцій за модулем, логічних операцій та ін.) під час визначення елементів породжуючої матриці чи під час векторного множення, отримання зворотної (перевірочної) матриці відомими механізмами лінійної алгебри може бути неможливим. У цих випадках перевірочну матрицю G^{-1} отримують, виходячи з властивостей коду.

Розмірність породжуючої матриці G (рис. 1), правила вибору чи формування її елементів (підматриць) визначаються видом перетворення, а також можливостями побудови зворотних матриць G^{-1}. Звичайно породжуюча матриця з причин, викладених нижче, має розмірність $(k\times k)$. Оскільки розмірність k перевищує довжину вихідного коду n, то можливі варіанти використання підматриць матриці G

чи доведення довжини вихідного коду до k. Для визначеності будемо вважати також, що умови існування зворотної матриці G^{-1} виконуються.

Перший варіант — криптоперетворення. При використанні вихідного коду довжиною в n символів і підматриці g матриці G (рис. 1) з n рядків і n стовпців (чи, що теж саме, окремої матриці $(n \times n)$) і визначених правилах вибору чи формування її елементів можна одержати матриці для криптографічних перетворень (шифрування) вихідного тексту.

$$G = \begin{pmatrix} g_{11} & g_{12} & \cdot & g_{1n} & \cdot & g_{1k} \\ g_{21} & g_{22} & \cdot & g_{2n} & \cdot & g_{2k} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ g_{n1} & g_{n2} & \cdot & g_{nn} & \cdot & g_{nk} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ g_{k1} & g_{k2} & \cdot & \cdot & \cdot & g_{kk} \end{pmatrix}$$

Рис. 1. Загальний вид кодувальної матриці

Код, що отриманий у результаті множення вихідного коду на кодувальну матрицю, є деяким криптографічним перетворенням вихідного коду. Якщо механізм формування елементів кодувальної матриці є секретним, чи механізм формування елементів кодувальної матриці є загальновідомим, але при їхньому формуванні використовуються деякий секретний параметр — ключ, то зашифрований код має визначену криптографічну стійкість, тобто стійкість до спроб криптоаналітиків одержати із зашифрованого коду (часто з використанням певної частки відкритого вихідного тексту) ключ, чи власне вихідний код (текст).

Така криптографічна стійкість є основною властивістю таких перетворень і досить часто визначається числом варіантів ключів.

Другий варіант — завадостійке кодування. Описані у варіанті 1 перетворення забезпечують надзвичайно важливу властивість захищеності інформації — конфіденційність, однак не дозволяють вирішувати проблему контролю, а тим більше відновлення цілісності інформації. (Єдиним, мабуть, виключенням є випадок, коли факт неможливості дешифрування зашифрованого слова можна тлумачити як факт наявності в ньому спотворення). Це пов'язано з тим, що операція обчислення нової матриці $B = A \times g$ не приводить до збільшення в закодованому слові кількості інформації (появі в ньому нової інформації), що необхідна для наступного виявлення факту і місця спотворення та його величини.

Отже, для перетворень, що дозволяють здійснювати контроль цілісності (можливо з наступним її відновленням) необхідно ввести потрібну для цього додаткову інформацію, тобто використовувати матриці розмірності ($k > n$) і, як наслідок цього, вихідні слова для кодування довжиною k символів. Тоді вихідне слово з n символів перетвориться на закодоване слово, як варіант — на завадостійкий код, довжиною в k символів.

Цей варіант передбачає розширення вихідного коду довжиною в n символів до вихідного слова для кодування довжиною в k символів і використання кодува-

льної матриці спеціального виду — породжуючої матриці (у термінах завадостійкого кодування). Найбільш простою процедурою перетворення вихідного коду довжиною в n символів на вихідне слово для кодування довжиною в k символів є додавання (вставка) $r = (k - n)$ додаткових символів, наприклад у кінець вихідного коду (у деяких кодах, наприклад у кодах Хеммінга, така вставка може здійснюватися і між символами вихідного коду). Матриця, що породжує, у цьому випадку (рис. 2) як підматрицю g містить одиничну матрицю, r додаткових рядків і стовпців, елементи яких у n рядках визначаються необхідними властивостями (типом) завадостійкого коду.

У результаті множення вихідного слова для кодування на кодувальну матрицю одержують k-символьний код, у якому перші n елементів збігаються з відповідними елементами вихідного коду, а інформація, що формується в додаткових, надлишкових r символах закодованого слова в теорії завадостійкого кодування, зветься контрольною ознакою.

$$\begin{pmatrix} 1 & 0 & . & 0 & . & g_{1k} \\ 0 & 1 & . & 0 & . & g_{2k} \\ . & . & . & . & . & . \\ 0 & 0 & . & 1 & . & g_{nk} \\ . & . & . & . & . & . \\ 0 & 0 & . & . & . & 1 \end{pmatrix}$$

Рис. 2. Загальний вид кодувальної матриці для завадостійких кодів

Якщо, наприклад, використовувати кодувальну матрицю, у якій $r - n = 1$, а n елементів k-го стовпця дорівнюють одиниці, то одержимо завадостійкий код (рис. 3), у якому контрольну ознаку отримують шляхом додавання (наприклад, порозрядного логічного чи по модулю 2^b, де b — двійкова довжина символів вихідного коду, тобто його довжина в бітах, і т.д.) усіх n елементів вихідного коду (еквівалент контрольного додавання).

$$G = \begin{pmatrix} 1 & 0 & . & 0 & 1 \\ 0 & 1 & . & 0 & 1 \\ . & . & . & . & . \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Рис. 3. Загальний вид кодувальної матриці для випадку коду,
що виявляє спотворення (контрольне додавання)

При контролі цілісності здійснюють множення закодованого слова на перевірочну матрицю G^{-1}, внаслідок чого одержують вектор — рядок із n інформацій-

них символів (можливо з порушеною цілісністю) та ($k - n$) так званих синдромів помилок, елементи яких при виборі надмірності, що достатня для рішення задач виправлення помилок, несуть інформацію про наявність, місце і величину спотворень у коді, що перевіряється. При недостатній надмірності ці елементи несуть інформацію про місце чи просто про наявність спотворень (коди із виявленням спотворень).

Третій варіант — завадостійка криптографія. Відзначимо [6], що використання кодувальної матриці виду 1 дозволяє використовувати однофазні процедури перетворення, що, на думку авторів, дає змогу підвищити загальну швидкодію засобів перетворення.

З цією метою необхідно розширити вихідний код на r символів (у найбільш простому випадку на один), арифметичні значення яких з умов технологічності слід обирати такими, що дорівнюють нулю ($a_j = 0, j = 1, 2, \ldots, r$). При цьому отримується вихідне слово для кодування довжиною в k символів $A = (a_1, a_2, \ldots, a_k)$. Крім того, слід сформувати кодувальну матрицю G за правилами відповідного криптографічного перетворення. В цій матриці ($k - r$) стовпчиків забезпечують розрахунок надлишкових символів, які є необхідними для забезпечення контролю цілісності (або залежно від їхньої кількості чи величини — контролю і поновленню цілісності). Після матричного множення $A_{\text{зкр}} = A \times G$ отримують зашифроване слово $A_{\text{зкр}}$, в якому n символів є суто криптографічним перетворенням вихідного слова A, а ($k - r$) символів забезпечують наступний контроль цілісності (чи контроль і поновлення цілісності).

Для зворотного перетворення необхідно здійснити векторне множення вектора $A_{\text{зкр}}$ на зворотну матрицю G^{-1} таку, що $A = A_{\text{зкр}} \times G \times G^{-1}$. Не зупиняючись на техніці отримання зворотної матриці, відмітимо, що, вочевидь, під час останнього перетворення операції векторного множення повинні забезпечити не лише зворотне криптографічне перетворення, але й надати змогу виявити та скорегувати можливі спотворення (порушення цілісності) в $A_{\text{зкр}}$.

Для ілюстрації можливості реалізації механізму завадостійкого криптографічного перетворення нижче розглянуто один із його варіантів.

Варіант блокової завадостійкої криптографії

Як варіант блокового завадостійкого криптографічного перетворення пропонується перетворення вихідного m-символьного цифрового коду (блоку відкритого тексту з m символів), який вважається деяким числом A у позиційній системі числення, на число $A_{\text{слк}}$ у системі числення в лишкових класах [8]. З урахуванням викладеного вище відмітимо, що з цією метою необхідно зробити наступні дії.

1. Символи вихідного блоку розглядати як символи a_i ($i = 1, 2, \ldots, m$) обраного позиційного представлення (цифри в позиційній системі числення числа A) з відповідними ваговими коефіцієнтами $c_i = 256^i$, за умови представлення символів вихідного коду як байтів. Неважко зрозуміти, що діапазон представлення таких чисел у цьому випадку дорівнює $0 \le A < 256^m$.

2. Визначити розміри кодувальної матриці та вихідного слова для кодування. З цією метою необхідно:

— вибрати сукупність основ системи числення в лишкових класах з $n \geq m$ взаємно простих чисел p_j (j = 1, 2, …, n), де p_j — j-та основа (елемент криптографічного ключа, за допомогою якого забезпечується потрібна імітостійкість, див. далі). Кількість (n) основ p_j (основ, які утворюють діапазон представлення чисел у лишкових класах — «робочих» основ) слід обирати такою, щоб забезпечити умову $256^m \leq P = \prod_{j=1}^{j=n} p_j$, де P — діапазон представлення («робочий» діапазон) системи числення;

— вибрати так звану «контрольну» основу (основу, за допомогою якої вводиться потрібна надлишковість) — p_k з умови

$$p_k > 2p_n \cdot p_{n-1},$$

де величини p_n і $p_{n\text{-}1}$ є найбільшими з основ p_j. У разі необхідності (наприклад, виходячи з умов технологічності обчислювальних процесів) слід застосувати складені контрольні основи у вигляді $p_k = \prod_{s=1}^{s=r} p_{ks}$, де r — кількість складених основ для обрахування контрольної;

— визначити загальну кількість основ системи числення в лишкових класах як $k = n + r$. Ця кількість визначає розмірність кодувальної та перевірочної матриць ($k \times k$), а величина r, окрім того, кількість додаткових (надлишкових) символів у вихідному коді для перетворення;

— розширити вихідний код на r (у найбільш простому випадку, при r = 1 — на один) символів $a_i = 0$, ($i = n + 1, \ldots, n + r$).

3. Створити кодувальну матрицю G, для якої як елементи g_{ij} використовувати величини

$$g_{ij} = \{c_i\}_{pj},$$

де знак $\{c_i\}_{pj}$ означає обчислення лишку (відрахування) від розподілу c_i на p_j.

4. Визначити елементи зворотної матриці G^{-1}. Із зауважень, що викладені в примітці, витікає, що в даному випадку елементи зворотної матриці слід визначати, виходячи із властивостей коду.

Відомо [6, 7], що як зворотну матрицю G^{-1} можна використати спрощену матрицю виду, як показано на рис. 4.

$$G^{-1} = \begin{pmatrix} g_{11} & B_{21} \\ g_{12} & B_{22} \\ . & . \\ g_{1n} & B_{2n} \\ . & . \\ g_{1k} & B_{2k} \end{pmatrix}.$$

Рис. 4. Вид спрощеної зворотної матриці

Як елементи першого стовпчика цієї матриці використовуються величини $g_{1i} = m_i / p_i$, а елементами другого стовпчика є так звані ортогональні базиси системи числення $B_{2j} = m_i \cdot P_j$, де $P_j = ((\prod_{j=1}^{j=k} p_j)/p_j)$; m_i — вагові коефіцієнти ортогональних базисів, такі, що $m_i = \{1/P_j\}_{pj}$; *позначка* $\{X\}_y$ *означає операцію по модулю y (обчислення лишку від розподілу X на y)*.

Для ілюстрації можливості реалізації розглянутих механізмів завадостійкого криптографічного перетворення нижче пропонуються варіанти відповідних алгоритмів.

Таким чином, із викладеного робимо висновок, що елементи матриці для зворотного перетворення із системи лишкових класів (СЛК) у позиційну систему числення (ПСЧ) можна визначати шляхом класичних математичних перетворень лише в окремих випадках. Більш універсальним є визначення таких матриць, виходячи із властивостей коду, тобто як значення ортогональних базисів системи. Останнє було визначено в [4] при розгляді питання про завадостійкі перетворення, коли як зворотну матрицю G^{-1} запропоновано використати спрощену зворотну матрицю виду

$$G^{-1} = \begin{pmatrix} g_{11} & B_{21} \\ g_{12} & B_{22} \\ . & . \\ g_{1k} & B_{2k} \end{pmatrix}.$$

Такий підхід розв'язує проблему щодо визначення матриць для декодування (для зворотного перетворення із СЛК у ПСЧ) у разі, коли детермінант кодувальної матриці дорівнює нулю. Тим самим знімаються й обмеження з вибору основ СЛК і, таким чином, обмеження щодо криптографічної стійкості коду (див. примітку).

Варіанти алгоритмів блокової завадостійкої криптографії

Алгоритмами блокової завадостійкої криптографії є алгоритми прямого (шифрування) та зворотного (дешифрування) криптографічних перетворень.

Алгоритм завадостійкого блокового криптографічного перетворення вихідного слова A на слово $A_{\text{слк}}$ зводиться до операції векторного множення $A_{\text{слк}} = A \times G$. При цьому всі операції при обчисленні символів перетвореного коду α_i слід виконувати за відповідними модулями p_j. Унаслідок цього вихідний код $A = a_1, a_2, \ldots, a_n, 0$ (при $r = 1$) перетвориться на число в лишкових класах $A_{\text{слк}} = \alpha_1, \alpha_2, \ldots, \alpha_n, \ldots, \alpha_k$, відносно якого можуть бути застосовані відомі механізми контролю, або контролю та відновлення цілісності.

Якщо при цьому правило вибору основ p_j (їхніх величин) не відоме неавторизованому користувачу, то отриманий унаслідок описаного криптографічного перетворення код $A_{\text{слк}}$ має і певну криптографічну стійкість, аналіз якої виходить за межі даної статті, і яку неважко довести до потрібної. Тобто, механізми, які запропоновані нижче на базі перетворень з області числення в системі лишкових класів, дозволяють використовувати їх у задачах завадостійкої криптографії.

Алгоритм зворотного блокового криптографічного перетворення [8] включає контроль цілісності слова, яке дешифрується, його поновлення (корекцію можливих спотворень) та власне зворотне перетворення.

Контроль цілісності і корекція можливих спотворень здійснюються після векторного множення слова, яке дешифрується, на елементи першого стовпчика (рис. 4). Для цього пропонується використання механізмів відомого [7] завадостійкого коду з корекцією спотворень — коду умовних лишків (ЛУ-коду). Відповідно з правилами ЛУ-коду операції під час векторного множення слова, яке дешифрується, на елементи першого стовпчика слід виконувати так, щоб отримати дробову частину результату операції — величину Z:

$$Z = \sum_{i=1}^{i=n1} \frac{б_i \cdot m_i}{p_i} - \left[\sum_{i=1}^{i=n1} \frac{б_i \cdot m_i}{p_i} \right].$$

У цьому виразі позначка $[X]$ — обчислення цілої частини змінної X; змінна $n1$ приймає значення $(n + r)$; змінна α_i — числовий (двійковий) еквівалент i-го інформаційного символу контрольованої частини файлу (базового кодового слова).

Отримане при цьому значення величини Z порівнюється з константою коду

$$Z < 1/p_k,$$

де змінна p_k, як і раніше, контрольна основа коду.

Якщо ця нерівність задовольняється, то це є критерієм того, що цілісність даного кодового слова не порушена. Якщо ж ця нерівність не задовольняється, то це є критерієм того, що цілісність даного кодового слова порушена, і здійснюється його відновлення відповідно до нижчевикладеного Z-алгоритму відновлення цілісності. Після цього здійснюється контроль цілісності наступного базового кодового слова доти, поки не закінчиться контроль усього повідомлення чи носія.

Відновлення інформації при контролі цілісності з використанням властивостей цього коду не вимагає використання резервних копій, а є сугубо розрахунковим з повним використанням інформації, що зосереджена в надлишкових символах — у контрольних ознаках кожного з перетворених слів.

Відповідно до Z-алгоритму корекція спотвореної змінної $\widetilde{\alpha}_i$, тобто обчислення неспотвореного значення цієї ж змінної α_i відбувається згідно з виразом

$$\alpha_i = \{\widetilde{\alpha}_i - \{[Z \cdot p_i] \cdot R_i\}_{p_i}\}_{p_i},$$

у якому зміст усіх змінних збігається з раніш визначеними.

В останньому виразі не визначеним є лише значення i — номера перекрученого символу $\widetilde{\alpha}_i$. Це значення знаходиться із системи нерівностей:

$$Z = Z \cdot p_i - \left[Z \cdot p_i \right] < p_i / p_k, (i = 1, 2,..., n).$$

За шукане значення і приймається номер тієї нерівності (того p_i), для якої задовольняється ця умова.

Дешифрування здійснюється шляхом векторного множення слова, яке дешифрується, на елементи другого стовпчика (рис. 4). Операції під час цього множення слід виконувати за модулем, який дорівнює повному діапазону представлення $R = P \cdot p_k$.

Унаслідок такої операції зашифроване слово (блок, число в лишкових класах) перетворюється на слово відкритого тексту (блок, число в позиційній системі числення).

При розробленні технології криптозахисту інформаційних об'єктів слід враховувати, принаймні, дві особливості. Перша з них пов'язана з обмеженими можливостями ЛУ-коду (як і будь-якого іншого завадостійкого корегувального коду) по виправленню спотворень у блоці для дешифрування (в термінах завадостійкого кодування — в базовому кодовому слові). Друга особливість пов'язана зі збільшенням розрядності інформаційних об'єктів (на r символів на кожне базове кодове слово) під час шифрування та з необхідністю зменшення цієї розрядності до початкової після дешифрування.

Врахування першої особливості здійснюється наступним чином. Розглянутий в попередньому розділі алгоритм забезпечує завадостійкі криптографічні перетворення (блокові шифрування-дешифрування) в разі виконання звичайної для завадостійких корегувальних кодів умови — всі спотворення в базовому кодовому слові зосереджені в межах одного символу (для даного варіанта — одного байту). Саме тоді є можливим виправлення спотворень. У випадку ж наявності в межах базового кодового слова більшої кількості спотворень їхнє виправлення при раніше визначеній надлишковості є неможливим. Зрозуміло, що як раз такі умови в каналах зв'язку, особливо з урахуванням здатності спотворень групуватися в пакети, є більш імовірними. Нагадаємо відоме співвідношення для визначення кількості спотворень $n_{\text{сп}}$ у повідомленні з k елементарних (двійкових) символів для каналу з відомими інтенсивністю завад v чи ймовірністю спотворення одного елементарного символу $P_{сп}$ на часовому проміжку, який дорівнює часу t_n передачі повідомлення [8]:

$$n_{сп} = vt_n \approx kP_{сп}.$$

Вихід з цієї ситуації є відомим — застосування механізмів перемежування. Неважко упевнитись, що перемежування слід здійснювати з такою глибиною λ, щоб при довжині символів (у бітах) b_c виконувалась умова

$$n_{\text{сп}} \le (\lambda - 1)\, b_c, \qquad (1)$$

звідки:

$$\lambda \ge [n_{сп} / b_c] + 1,$$

тобто мінімальне значення $\lambda \geq 2$. У разі врахування можливостей більш значного порушення цілісності інформаційних об'єктів (наприклад, внаслідок дій зловмисників), ніж це витікає з виразу (1), глибина перемежування може вибиратись і значно більшою (див. далі). При цьому існують можливості організації перетворень із сталою, або змінною величиною λ, тобто із сталою (блоково-груповий контроль) чи змінною довжиною узагальнених кодових слів, наприклад, контроль по файлам, якщо обмін здійснюється інформаційними об'єктами типу файл. Під узагальненим кодовим словом тут розуміється впорядкований певним чином (наприклад, за правилами перемежування) блок інформації довжиною в $N = m\lambda$ при шифруванні та $N = k\lambda$ символів при дешифруванні. Звернемо увагу, що при обробці інформації по файлах глибина перемежування може бути досить значною ($\lambda = [N_\text{ф}/m] + 1$, де $N_\text{ф}$ — загальна кількість символів в інформаційному об'єкті перед його шифруванням), а цей інформаційний об'єкт можна розглядати як своєрідне узагальнене кодове слово.

При такій організації узагальнених кодових слів врахувати другу особливість дуже просто. Дійсно, для цього достатньо під час шифрування здійснювати приформування інформації, яка зосереджена в надлишкових символах (у кількості λr символів на кожне узагальнене кодове слово), в чітко визначених алгоритмом місцях.

Звернемо увагу на те, що при обробленні інформації у межах *кожного із узагальнених кодових слів* можна виправити виявлені в ньому спотворення, довжина яких B_B може бути (при їхньому довільному розташуванні в межах узагальненого кодового слова) від одного до $B_B = [(\lambda - 1)\cdot b_c + 1]$ двійкових символів (біт). Тобто найбільш можлива довжина довільно розташованих у межах узагальненого кодового слова спотворень, що виправляються, коли ще можливе дешифрування, дорівнює $(\lambda - 1)$ символів. Загальна кількість спотворень такої довжини, що виправляються, дорівнює кількості узагальнених кодових слів у складі файлу.

Слід зазначити, що умови застосування цих процедур впливають на вибір методу організації оброблення. В умовах збереження інформації (на деяких носіях) слід враховувати те, що:

1) метод блоково-групового оброблення інформації дозволяє розкрити ***багато груп спотворень малої довжини***, що безумовно є дуже корисним для ***організації контролю цілісності*** тих носіїв, на які спотворення мають природний, а не штучний характер. Такими носіями можуть бути, наприклад, ***резервні копії програмних засобів чи баз даних*** тощо. Але для контролю спотворень великої довжини, а це є притаманним спотворенням штучного характеру, цей вид контролю застосовувати недоцільно;

2) оброблення інформації у файлах дозволяє викрити значно меншу кількість спотворень, чим при попередньому виді контролю, але найбільшої, максимальної довжини, і тому цей вид контролю доцільно застосовувати при контролі цілісності інформації файлів, у разі потреби такого контролю, наприклад, при контролі цілісності інформації, що дискретно оперативно змінюється.

При обміні інформацією слід врахувати наступне:

1) методи блоково-групового оброблення, з їхніми визначеними вище особливостями, можуть бути легко пристосованими для контролю процесів обміну без використання механізмів вирішального зворотного зв'язку, що приведе до підви-

щення швидкості обміну (довжина блоку повинна відповідати довжині повідомлення, прийнятої у відповідному протоколі. Це просто реалізується, наприклад, у протоколі X.25);

2) контроль цілісності інформації у блоках, довжина яких перевищує довжину повідомлення (контроль у файлах), дозволяє застосовувати принципи каскадних кодів і забезпечувати цілісність в умовах чи то впливу перешкод великої тривалості (кількість спотворень перевищує коригувальні властивості внутрішнього коду), чи то тривалих (у тому ж розумінні) завмирань сигналів.

Отже, в статті розглянуто питання використання матричних завадостійких криптографічних перетворень для задач забезпечення конфіденційності та цілісності інформаційних об'єктів. Проаналізовано деякі з можливих варіантів їхньої побудови і застосування та їхні можливості.

1. *Нормативний* документ Системи технічного захисту інформації «Загальні положення про захист інформації в комп'ютерних системах від несанкціонованого доступу» (НД ТЗІ 1.1-002-99).

2. *Нормативний* документ Системи технічного захисту інформації «Критерії оцінки захищеності інформації в комп'ютерних системах від НСД» (НД ТЗІ 2.5-004-99).

3. *Нормативний* документ Системи технічного захисту інформації «Класифікація автоматизованих систем і стандартні функціональні профілі захищеності оброблюваної інформації від несанкціонованого доступу» (НД ТЗІ 2.5-005-99).

4. *Введение* в криптографию / Под. ред. В.В. Ященко. — СПб.: Питер, 2001. — 288 с.: іл.

5. *Василенко В.С.* Цілісність інформації в автоматизованих системах / В.С. Василенко, М.П. Короленко // Корпоративні системи. — 1999. — № 3. — С. 52–57.

6. *Василенко В.С.* Використання методу завадостійкої криптографії в системах обробки кредитно-фінансової інформації / В.С. Василенко, С.Г. Курочкін // Машинна обробка інформації. Міжвідомчий науковий збірник. — Вип. 60. — 1997. — С. 169–174.

7. *Василенко В.С.* Механізми контролю цілісності та її поновлення / В.С. Василенко, М.М Будько, М.П. Короленко // Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні. — 2000. — С. 130–139.

8. *Василенко В.С.* Варіант завадостійкого криптографічного перетворення / В.С. Василенко, В.М. Горицький // Современные проблемы телекоммуникаций: зб. доповідей на 6-й міжнародній наук.-техн. конф. 19–22 серпня 2003 р. (Ч. 1) // Одеська національна академія зв'язку ім. А.С. Попова. — С. 71–73.

Надійшла до редакції 18.11.2011