6. Intrusion Detection System Mr Mohit Tiwari, Raj Kumar, Akash Bharti, Jai Kishan, Volume 5, Issue 2 (March — April 2017), PP. 38–44
7. M. Rash, A. Orebaugh, G. Clark, B. Pinkard, J. Babbin: Intrusion Prevention and Active Response: Deploying Network and Host IPS (Syngress, Rockland, Massachusetts 2005).
8. NIST SP 800-51, Use of the Common Vulnerabilities and Exposures (CVE) Vulnerability Naming Scheme.
9. Open Source Host-based intrusion detection system, 2007. http://www.ossec.net/
10. R. Bace: Intrusion Detection (New Riders, Indianapolis 2000)
11. Rebecca Bace and Peter Mell, "NIST Special Publication on Intrusion Detection Systems," 16 August 2001.
12. Survey of Current Network Intrusion Detection Techniques, Sailesh Kumar.

Статтю подано до редакції 22.09.2019 р.

УДК 004.62 DOI: 10.33111/mise.98.15

Мамонова Г.В., к. фіз.-мат. н.,
доцент кафедри комп'ютерної математики та інформаційної безпеки,
Меднікова М.В.,
студентка 3-го курсу спеціальності «Кібербезпека», Київський національний економічний університет імені Вадима Гетьмана

Mamonova G.V., PhD in Physics and Mathematics,
Associate Professor of the
Computer Mathematics and Information Security Department,
Mednikova M.V.,
3rd year Student of the "Cybersecurity" speciality,
Kyiv National Economic University named after Vadym Hetman

КРИПТОГРАФІЧНИЙ АНАЛІЗ АЛГОРИТМУ DES

CRYPTOGRAPHIC ANALYSIS OF THE ALGORITHM DES

Анотація. *Сьогодні це дуже важлива безпека під час передачі даних. Оскільки все сьогодні передається через Інтернет, дуже ймовірно, що наші дані будуть взяті та використані. Ми провели міні-програмне забезпечення на мові c #, яке робить шифрування файлу у форматі .txt, і те, що ми будемо проводити в цій роботі, — це вимірювання часу шифрування різних розмірів файлів за допомогою алгоритму DES та AES алгоритм різних процесорів за допомогою шифрування файлів, за допомогою яких ми будемо проводити порівняння між двома алгоритмами, а також проводити порівняння між процесорами.*
Швидко розвиваються комп'ютерні інформаційні технології та вносять помітні зміни в наше життя. Усе частіше поняття «інформація» вико-

ристовується як позначення спеціального товару, який можна придбати, продати, обміняти на щось інше і т.п. При цьому вартість інформації перевершує вартість комп'ютерної системи, у якій вона знаходиться. Тому цілком природно виникає потреба в захисті інформації від несанкціонованого доступу, умисної зміни, крадіжки, знищення та інших злочинних дій. Саме тому важливим є використання різних криптоалгоритмів, таких як DES, для захисту інформації користувачів. Що і зумовлює актуальність вибраної теми.
Метою криптографії є надання можливості двом людям обмінюватися повідомленням таким чином, щоб інші люди не могли зрозуміти повідомлення. Кількість способів цього не закінчується, але тут ми торкнемось способів зміни тексту таким чином, щоб одержувач міг скасувати зміни та виявити оригінальний текст (Sumitra, 2013). У цій статті подано порівняння криптографічного алгоритму DES та криптографічного алгоритму AES в різних процесорах. Стаття складається з трьох розділів. Перша і друга частина починаються з алгоритмів DES і AES, продовжуються з останньою частиною, яка стосується результатів та експериментів.
***Ключові слова**: DES, кібербезпека, криптостійкість, захищеність, криптоалгоритм.*

***Abstract.** Nowadays it is highly important the security while data transmission. Since everything nowadays is transmitted through the Internet, it is very likely for our data to be taken and misused. What we have conducted is mini (minor) software in the c# language, which makes encryption of the file in .txt format, and what we will conduct in this paper is the measurement of time of encryption of different size of files with DES algorithm and AES algorithm of different CPUs by encrypting files, with which we will make comparisons between two algorithms and also make comparisons between the CPUs.*
Computer information technologies are evolving rapidly and are making a significant difference to our lives. Increasingly, the term "information" is used to refer to a special product that can be bought, sold, exchanged for something else, etc. The cost of information exceeds the cost of the computer system in which it is located. Therefore, there is a natural need to protect information from unauthorized access, intentional alteration, theft, destruction and other criminal activities. That is why it is important to use different crypto algorithms, such as DES, to protect user information. This is what made the topic chosen relevant.
The goal of cryptography is to make it possible for two people to exchange a message in such a way that other people cannot understand the message. There is no end to the number of ways this can be done, but here we will be concerned with methods of altering the text in such a way that the recipient can undo the alteration and discover the original text (Sumitra, 2013). In this paper there is provided a comparison of DES cryptographic algorithm and AES cryptographic algorithm in different CPU. The paper is composed in three sections. The first and the second part starts with the DES and AES algorithms, continuing with the last part which deals with the results and the experiments.
***Keywords**: DES, cybersecurity, crypto-stability, security, crypto-algorithm.*

Вступ: На сьогоднішній день завдяки повсюдному застосуванню відкритих мереж передачі даних, таких як Internet, і побудованих на їх основі мереж intranet і extranet криптографічні протоколи знаходять усе ширше застосування для вирішення різноманітного кола завдань і забезпечення послуг куристувачам мереж, кількість яких постійно збільшується.

Постановка проблеми: Криптографія сьогодні — це найважливіша частина всіх інформаційних систем. Криптографія забезпечує підзвітність, прозорість, точність і конфіденційність. І по мірі збільшення обчислювальних потужностей техніки та одночасного збільшення взаємозв'язку повсякденного життя з комп'юетрними мережами, збільшується важливість використання криптографії і безпосередньо криптографічних алгоритмів, таких як DES.

Виклад основного матеріалу: Data Encryption Standard (DES) — алгоритм з симетричним ключем для шифрування електронних даних. Довжина його ключа — 56 біт (доволі короткий), через що алгоритм і піддався критиці з самого початку, адже це робить його занадто небезпечним для більшості сучасних додатків. Але незважаючи на критику, DES мав великий вплив на розвиток сучасної криптографії.

DES був світовим стандартом протягом 25 років. У 1972 р. колишнє американське Національне бюро стандартів (NBS), яке тепер називається Національним інститутом стандартів та технологій (NIST), ініціювало проект з метою захисту комп'ютерів та даних цифрового зв'язку. У рамках цієї програми вони хотіли розробити єдиний, стандартний криптографічний алгоритм. Мотивація була такою:

- один алгоритм легше перевірити та сертифікувати, ніж тисячу;
- окрім того, було б легше дозволити взаємодію різних криптографічних пристроїв, що використовують його.

У 1974 році з'явився шифр Lucifer, розроблений Хорстом Фейстелем у лабораторіях IBM. Після секретної перевірки від АНБ Data Encryption Standard був прийнятий як федеральний стандарт у 1976 році та затверджений для використання у всіх несекретних урядових комунікаціях одним роком пізніше. Стандарт був переатестований у 1983, 1987 та 1993 роках без особливих проблеми. У 1997 році, оскільки алгоритм демонстрував деякі ознаки старості, його більше не можна було вважати безпечним алгоритмом, NIST вирішив запустити процес, пошуку наступника на наступних 20 років. Однак треба зазначити, що варіанти DES, такі як Triple-DES, досі вважається дуже безпечним.

Хоча віднедавна DES не має сертифіката, він все одно часто використовується і вартий того, щоб його вивчали. Основним плюсом DES є те, що його використання дає можливість досягти високої швидкості шифрування/дешифрування.

Початковий варіант DES постійно змінювався, доповнювався; зараз з'являються нові алгоритми на основі DES — NewDES, Triple DES та деякі інші. Необхідність розробки нових алгорит-

мів була зумовлена великою кількістю атак, яким піддавався алгоритм за роки свого існування. Окрім того велику роль зіграв бурхливий розвиток засобів обчислювальною та мікропроцесорної техніки. Він призвів до того, що 56-бітного ключа, який використовується в оригінальному варіанті DES, стало просто недостатньо для протистояння атакам, які реалізовувались методом brute force. Однак у комерційній сфері та системах електронних розрахунків DES і зараз лишається одним з найпопулярніших алгоритмів блочного шифрування.

Основними перевагами є:

- властивості DES перемішування та розсіювання: кожен біт шифротексту базується на кількох бітах ключа, і зміна одного біта початкового тексту змінює в середньому половину біт шифротексту;
- DES був розроблений для роботи на апаратних засобах 1978 року, зараз він є доволі швидким для роботи в апаратному забезпеченні;
- простота використання DES завдяки структурі Фейстеля та нескладній логіці.

Основними недоліками є:

- 56-бітний розмір ключа, що є найбільшим дефектом DES;
- нетійкість DES до атак методу brute force. Однак використання TripleDES пом'якшує цю проблему через збільшення часу виконання зламу;
- повільна робота DES у програмному забезпеченні, адже він не був розроблений для програмного забезпечення;
- вразливість DES до атак з використанням диференціального та лінійного криптоаналізу. Хоча ці методи й потребують більше часу та пам'яті;
- використання статичних підстановок в S-боксах, що, незважаючи на велику кількість раундів, дозволяє криптоаналітикам проводити атаки на цей алгоритм.

Загалом алгоритм є не складним для розуміння і не важким у реалізації.

DES є реалізацією шифру Фейстеля і базується на двох основних ознаках криптографії: заміщенні і транспозиції.

Шифр Фейстеля — це ітераційний блок-шифр, тобто він включає послідовне повторення внутрішньої функції, що називається раундовою функцією.

Алгоритм складається із 16 ступенів, кожен з яких називається раундом.

Усього для отримання блоку зашифрованого повідомлення проходить 16 раундів. Кожен раунд виконує етапи заміщення та

переміщення. Результатом цього процесу є 64-бітний блок шифру. Хоча розмір блоку і складає 64 біти, проте ефективна довжина ключа DES складає лише 56 біт. Вісім з 64 біт не використовуються алгоритмом шифрування (вони функціонують тільки як контрольні біти). Загальну структуру DES зображено на рис. 1.

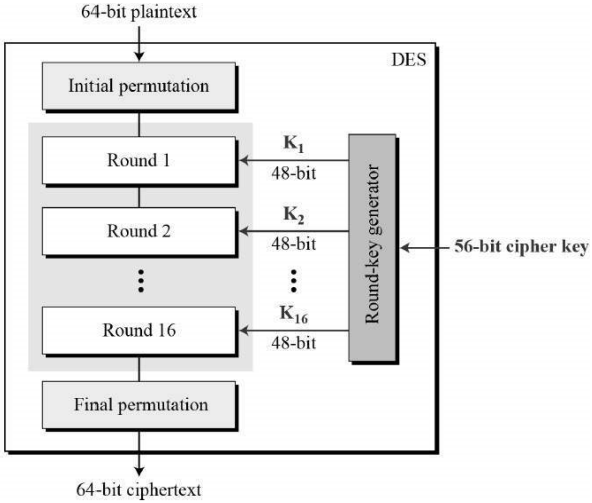

Рис. 1. Структура DES

Алгоритм складається з кількох кроків.

Початкова перестановка

Початкова перестановка є прямим блоком перестановок (P-блоками). Приміняється для того, щоб здійснити початкове розсіювання статистичної структури повідомлення. Початкова перестановка показана на рис. 2. Наприклад, вказується, що IP замінює перший біт початкового тексту на 40-й біт оригінального простого тексту, другий біт на — на 8-й біт початкового текстового блоку і так далі.

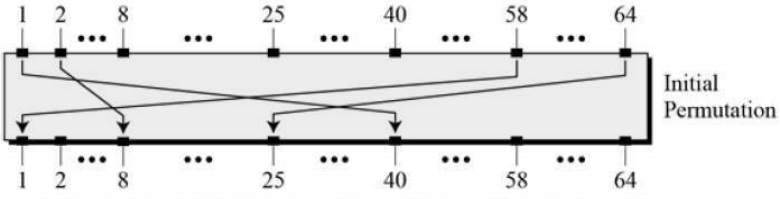

Рис. 2 Початкова перестановка

Як уже було сказано раніше, вхідний 64-бітний блок розбивається на 2 рівні частини (по 32 біти), LPT — ліва частина та RPT — права частина. Кожен із 16 раундів, у свою чергу, складається з кроків широкого рівня.

Крок 1: Перетворення ключів

Початковий 64-бітний ключ перетворюється в 56-бітний, відкидаючи кожен 8-й біт початкового ключа. Таким чином для кожного 56-бітний ключ доступний. Безпосередньо з цього 56-бітного ключа в кожному раунді генерується окремий 48-бітний субключ з використанням процесу, що називається перетворенням ключа. Для цього 56-бітний ключ ділиться навпіл, кожна 28 біт. Ці половини зсуваються по колу вліво на одну або дві позиції, залежно від раунду.

Наприклад, якщо в раунді з номерами 1, 2, 9, 16 зсув виконується тільки на одну позицію, то для інших раундів, коловий зсув виконується на дві позиції. Кількість бітів ключа, зсунутих за раунд показано на рис. 4.

Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
#key bits shifted	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Рис. 4. Кількість зсунутих бітів за певний раунд

Після такого зсуву обираються 48 з 56 бітів. Таблицю для вибору бітів наведено на рис. 5. Наприклад, після зсуву біт 14 переміщається у першу позицію, біт 17 переміщається в другу позицію і так далі. Якщо уважно подивитись на таблицю, можна зрозуміти, що вона складається тільки з 48-бітних позицій. Біт під номером 18 відкидається (ми не знайдемо його в таблиці), як і 7 інших, щоби зменшити 56-бітний ключ до 48-бітного.

Оскільки процес перетворення ключа включає в себе перестановку, а також вибір 48-бітної підмножини початкового 56-бітного ключа, він називається перестановкою стиснення.

14	17	11	24	1	5	3	28	15	6	21	10
23	19	12	4	26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40	51	45	33	48
44	49	39	56	34	53	46	42	50	36	29	32

Рис. 5 Таблиця для вибору бітів

Через цю техніку перестановки стиснення в кожному раунді використовується різна підмножина ключових бітів. Завдяки цьому DES було не легко зламати.

Крок 2: Перестановка розширення

Після IP у нас було два блоки по 32 біти, звані як LPT та RPT. Оскільки правий блок 32-бітний, а раундовий ключ 48-бітний, спочатку треба розширити правий блок до 48 біт. Це реалізується за допомогою перестановки розширення. Біти переставляються і тому це називається перестановкою розширення. Це відбувається, коли 32-бітний RPT ділиться на 8 блоків, кожен з яких по 4 біти. Потім кожен-бітний блок розширюється на відповідний 6-бітний як вказано на рис. 6. Цей процес призводить до розширення, а також перестановки вхідного біта під час створення виводу.

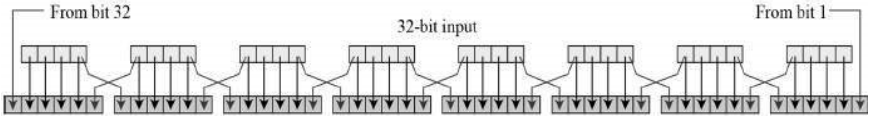

Рис. 6. Схема розширення правого блоку

Графічно зображена логіка перестановок зазвичай описується як таблиця в специфікації DES, що показана на рис. 7.

32	01	02	03	04	05
04	05	06	07	08	09
08	09	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	31	31	32	01

Рис. 7. специфікація DES

Крок 3: S-блокова перестановка

Блоки заміщення — S-блоки виконують реальне мікшування (заплутування). DES використовує 8 S-блоків, кожен з 6-бітним входом і 4-бітним виходом. Підстановка в S-блоках виконується за таким правилом: номер рядка задається першим і останнім

входом S-блока, а номер стовпчика — середніми чотирма бітами входу. Бітове представлення числа в комірці задано вхідною послідовністю і буде виходом S-блоку.

Правило S-блоків зображено на рис. 8.

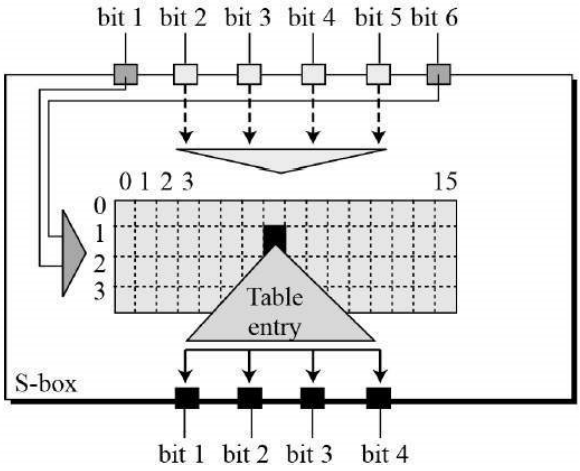

Рис. 8. Правило S-блоків

Всього вісім таблиць S-блоків. Вихід усіх восьми s-блоків потім об'єднується в 32-бітний блок. Потім виконується пряма перестановка — 32-бітний вихід S-блоків потім піддається прямій перестановці.

Крок 5: P-блокова перестановка

На цьому етапі 32-бітний вихід підстановки S-блоків переставляється відповідно до P-блоків. Ця перестановка переносить кожен вхідний біт у вихідне положення; жоден біт не використовується двічі і жодні біти не ігноруються. Це називається прямою перестановкою або просто перестановкою. Нарешті результатом P-блокової перестановки є розширена ліва половиною початкового 64-бітового блоку. Потім ліву і праву половинки міняють місцями і починається ще один раунд. І таких 16 підряд.

Крок 6: Кінцева(фінальна) перестановка

Кінцева перестановка є зворотною від початкової перестановки і описана в таблиці. Зверніть увагу, що ліва і права половини не міняються місцями після останнього раунду DES; натомість зв'язаний блок R_{16} L_{16} використовується як вхід до кінцевої перестановки. Тут нічого не відбувається; обмін половинними блока-

ми і зміщення по колу перестановки дало б точно такий же результат. Отже, алгоритм можна використовувати як для шифрування, так і дешифрування.

Після всіх підстановок, перестановок, XOR і зміщення навколо, можна подумати, що алгоритм дешифрування зовсім інший і такий же заплутаний, як алгоритм шифрування. Але навпаки, різні операції були обрані спеціально для отримання дуже корисної властивості. Один і той же алгоритм працює як для шифрування, так і для дешифрування.

Через невелику кількість можливих ключів (усього 2^{56}), з'являється можливість їх повного перебору на швидкодіючій обчислювальній техніці за реальний час. У 1998 році Electronic Frontier Foundation, використовуючи спеціальний комп'ютер DES-Cracker, вдалося зламати DES за 3 дні.

DES уже зламували за допомогою таких атак:

- Диференціальний криптоаналіз. Ця атака вимагає шифрування 2^{47} відкритих текстів, обраних нападаючим, і для її виконання потрібні приблизно 2^{47} кроків. Теоретично будучи крапкою розриву, ця атака непрактична через надмірні вимоги до підбору даних і складності організації атаки за обраним відкритим текстом. DES є відносно захищеним для такої атаки.
- Лінійний криптоаналіз. Цей метод дозволяє відновити ключ DES за допомогою аналізу 24^3 відомих відкритих текстів, при цьому потрібно приблизно 2^{43} кроків для виконання. Перший експериментальний криптоаналіз DES таким методом був успішно виконаний протягом 50 днів на автоматизованих робочих місцях 12 HP 9735.

Хоча для лінійного і диференціального криптоаналізу потрібно досить великий обсяг пам'яті для збереження обраних (відомих) відкритих текстів до початку атаки. Але особливу загрозу DES несе метод простого перебору.

Для збільшення криптостійкості DES, з'явилося кілька його нових варіацій: double DES (2DES), triple DES (3DES), DESX, G-DES.

Методи 2DES і 3DES засновані на DES, але збільшують довжину ключів (2DES — 112 біт, 3DES — 168 біт) і тому збільшується криптостійкість.

Метод DESX. Це посилений варіант DES, підтримуваний інструментарієм RSA Security. DESX відрізняється від DES тим, що кожен біт вхідного відкритого тексту DESX логічно підсумовується по модулю 2 з 64 бітами додаткового ключа, а потім шифрується за алгоритмом DES. Кожен біт результату також логіч-

но підсумовується по модулю 2 з іншими 64 бітами ключа. Головною причиною використання DESX є простою в обчислювальному сенсі спосіб значно підвищити стійкість DES до атак повного перебору ключа.

Метод G-DES розроблений для підвищення продуктивності DES на основі збільшення розмірів шифрованого блоку. Заявлялося, що G-DES захищений так само, як і DES. Однак було показано, що G-DES з рекомендованими параметрами легко зламуються, а при будь-яких змінах параметрів шифр стає ще менш захищений, ніж DES.

Ще інший варіант DES використовує незалежні суб-ключі. Навідміну від алгоритму DES, у цьому варіанті використовується 768-бітний ключ (розділений на 16 48-бітових підключів) замість 16 залежних 48-бітних ключів, створюваних за ключовим графіком алгоритму DES. Хоча очевидно, що використання незалежних суб-ключів значно ускладнить повний пошук ключа, але стійкість до атаки диференціальним або лінійним криптоаналізом ненабагато перевищить стійкість звичайного DES.

Висновки: Прийняття стандарту шифрування DES стало потужним поштовхом до широкого застосування шифрування в комерційних системах. Введення цього стандарту — відмінний приклад уніфікації та стандартизації засобів захисту.

Стандартизація останнім часом набуває міжнародного характеру, підтвердження тому — міжнародний стандарт ISO 8372:1987, розроблений на основі криптоалгоритму DES.

Алгоритм DES був затверджений 20 років тому, проте за цей час комп'ютери зробили немислимий стрибок у швидкості обчислень, і зараз не так уже й важко зламати цей алгоритм шляхом повного перебору всіх можливих варіантів ключів (а в DES використовується всього 8-байтний), що недавно здавалося абсолютно неможливим.

Криптографія сьогодні — це найважливіша частина всіх інформаційних систем: від електронної пошти до стільникового зв'язку, від доступу до мережі Internet до електронних грошей. Криптографія забезпечує підзвітність, прозорість, точність і конфіденційність. Вона запобігає спробам шахрайства в електронній комерції і забезпечує юридичну силу фінансових транзакцій. Криптографія допомагає встановити вашу особистість, але й забезпечує вам анонімність. Вона заважає хуліганам зіпсувати сервер і не дозволяє конкурентам залізти у ваші конфіденційні документи. А в майбутньому, у міру того як комерція і комунікації будуть все тісніше зв'язуватися з комп'ютерними мережами,

криптографія стане життєво важливою, разом з чим і такі алгоритми, як DES, стануть життєво необхідними.

Література

1. Панасенко С.П. Алгоритми шифрування: Спеціальний справочник / СПб.: БХВ-Пітербург, 2009 — 576 с.
2. Петров А.А. Комп'ютерна безпека. Криптографічні методи захисту: Навч. посібник / М.: ДМК, 2000 — 448 с.
3. Род Стівенс Алгоритми теорія та практичне застосування: пер. з англ.м. / Москва: вид. «Э», 2016 — 544 с.
4. Романець Ю.В., Тимофєєв П.А., Шаньгін В.Ф. Захист інформації в комп'ютерних системах та мережах / Москва: Радио и связь, 2001 — 376 с.
5. Шаньгін В. Ф. Захист інформації в комп'ютерних системах та мережах: Підручник / Москва: ДМК, 2012 — 560 с.
6. Ященко В.В. Введення в криптографію: Навч. посібник / СПб., 2001.
7. Electronic Frontier Foundation Cracking DES: Навчальний посібник / O`Reilly&Associates, 1998. — 277 с.
8. Matt Curtin Brute Force: Cracking the Data Encryption Standard: Підручник / Copernicus, 2005. — 292 с.

References

1. Panasenko S.P. Encryption Algorithms: Manual / St. Petersburg: 2009 — 576 p.
2. Petrov A.A. Computer security. Cryptographic methods of protection: Educ. manual / 2000 — 448 s.
3. Rod Stevens Algorithms theory and practical application / Moscow, 2016. — 544 p.
4. Romanets Y.V., Timofeev P.A., Shangin V.F. Information Security in Computer Systems and Networks / Moscow, 2001 — 376 p.
5. Shangin V.F. Information protection in computer systems and networks: Educ. manual / Moscow, 2012 — 560 p.
6. Yashchenko V.V. Introduction to Cryptography: Educ. manual / St. Petersburg, 2001.
7. Electronic Frontier Foundation Cracking DES: Educ. manual/ O`Reilly&Associates, 1998. — 277 c.
8. Matt Curtin Brute Force: Cracking the Data Encryption Standard: Manual / Copernicus, 2005 — 292 c.

Статтю подано до редакції 14.09.2019 р.