Г.А. Могильний, М.А. Семенов, С. О. Переяславська, О .О. Смагіна

ЗАХИСТ ІНФОРМАЦІЇ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ
ДЕРЖАВНИЙ ЗАКЛАД
„ЛУГАНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА”

Г.А. Могильний, М.А. Семенов, С. О. Переяславська, О .О. Смагіна

ЗАХИСТ ІНФОРМАЦІЇ

Методичні рекомендації до виконання лабораторних робіт
для здобувачів освіти спеціальності
122 – „ Комп’ютерні науки”

Полтава
ДЗ „ЛНУ імені Тараса Шевченка”
2024

УДК 004.056(072)
М 74

Р е ц е н з е н т и :

Козуб Ю.Г. - доктор технічних наук, професор, в.о. завідувача кафедри математики та інформатики ДЗ «Луганський національний університет імені Тараса Шевченка».

Лахно В.А – доктор технічних наук, професор кафедри комп'ютерних систем, мереж та кібербезпеки, Національний університет біоресурсів і природокористування України, м. Київ

М74 Могильний Г.А., Семенов М.А., Переяславська С. О., Смагіна О. О. Захист інформації: методичні рекомендації до виконання лабораторних робіт для здобувачів освіти спеціальності 122 – Комп’ютерні науки. Полтава: ДЗ „ЛНУ імені Тараса Шевченка”, 2024. 53 с..

Методичні рекомендації структуровано відповідно до розділів навчальної робочої програми курсу "Захист інформації" для спеціальності 122 – „Комп’ютерні науки та інформаційні технології” кафедри інформаційних технологій та систем ДЗ ЛНУ імені Тараса Шевченка. Методичні рекомендації охоплюють основні групи питань з безпеки та захисту інформації: ідентифікація та виявлення загроз, методи захисту інформації в різних операційних системах, моніторинг комп'ютерних систем, застосування алгоритмів шифрування для захисту інформації. Особлива увага приділяється формуванню навичок та вмінь застосування методів та засобів забезпечення безпеки інформації.

Методичні рекомендації призначені для студентів фізико-математичного та технічного профілю, вчителів ліцеїв, коледжів, гімназій, слухачів курсів підвищення кваліфікації, а також для самоосвіти.

.

УДК 004.056(072)

Рекомендовано до друку Вченою радою
Луганського національного університету імені Тараса Шевченка
(протокол № 6 від 20 грудня 2024 р.)

ЗМІСТ

ВСТУП

У сучасну цифрову епоху захист персональних даних стає все більш важливим. З постійним потоком інформації та зростаючим занепокоєнням щодо конфіденційності важливо розуміти принципи та норми, що регулюють захист інформації. Вивчення курсу „Захист інформації” є невід’ємною частиною у загальному процесі підготовки студентів спеціальності 122 ”Комп’ютерні науки та інформаційні технології„. Однією із основних навчальних форм є лабораторні роботи, які відіграють провідну роль у формуванні практичних навичок та застосуванні набутих знань. Методичні рекомендації до виконання лабораторних робіт мають на меті формування у студентів вмінь та навичок, необхідних для ефективного захисту даних і особистої інформації. Лабораторні заняття логічно продовжують вивчення тем, розпочатих на лекціях..

Методичні рекомендації до виконання лабораторних робіт структуровані відповідно до навчальної робочої програми курсу „Захист інформації” й складаються з десяти лабораторних робіт, що охоплюють широкий спектр питань з захисту інформації. Тематика робіт спрямована на вивчення як базових основ кібербезпеки (ідентифікація загроз, парольний захист,) так і більш складних тем, пов’язаних з захистом інформації в різних операційних системах, комп’ютерних мережах, застосування криптографічних алгоритмів для шифрування інформації (DES, RSA тощо).

Зміст лабораторних занять передбачає роботу в комп’ютерних класах та самостійну роботу студентів. Методичні рекомендації можуть бути корисними для студентів інших спеціальностей у якості посібника для самостійного опанування технологій Java-програмування.

Примітка. ННІМІТ ДЗ ЛНУ імені Тараса Шевченка є членом Cisco Networking Academy, тому в цих методичних реклмендаціях застосовувалися матеріали курсів Cisco Networking Academy, посилання на які є в лабораторних роботах та джерельній базі.

ВИМОГИ ДО ВИКОНАННЯ Й ЗАХИСТУ ЛАБОРАТОРНИХ РОБІТ

Лабораторні роботи виконуються кожним студентом самостійно. Перед початком виконання лабораторної роботи студент повинен ознайомитися з теоретичним матеріалом, використовуючи джерела, наведені у списку рекомендованої літератури, чи будь-які інші; усвідомити завдання та порядок проведення роботи. Під час виконання лабораторної роботи студент повинен поетапно виконати завдання відповідно до варіанту. Результатом проведеної роботи можуть бути розроблені програмні додатки (мова програмування обертається студентом), застосування спеціальних програмних інструментів для реалізації задач захисту інформації тощо. Після виконання лабораторної роботи студент повинен оформити звіт, який має містити наступне: ПІБ студента, № групи, курсу, назву дисципліни, номер та назву теми лабораторної роботи, завдання до роботи (відповідно до номеру варіанту), розв'язання поставленого завдання. До звіту додаються файли проекту у вигляді архіву (якщо вони є). Лабораторна робота подається до захисту безпосередньо після її виконання. Під час захисту роботи студент демонструє результати лабораторної роботи. Кожна лабораторна робота оцінюється за бальною системою, яка встановлена робочою програмою курсу. Оцінюється якість підготовки, повнота виконання роботи, вміст збережених файлів, зміст відповідей та оформлення звітів. Оцінці "відмінно" відповідає виконання більше 90% пунктів роботи, "добре" – більше 75%, "задовільно" – більше 60%.

ЛАБОРАТОРНА РОБОТА №1. ІДЕНТИФІКАЦІЯ ЗАГРОЗ

Мета: Вивчення можливостей забезпечення функцій безпеки, які використовуються організаціями для збереження даних.

Частина 1: Дослідження загроз, які витікають від кібератак

Частина 2: Тріада CIA (конфіденційність, цілісність і доступність)

Необхідні ресурси

ПК або мобільний пристрій з доступом до Інтернету

В цій лабораторній роботі застосовувалися матеріали Cisco Networking Academy (http://surl.li/htmfim)

Короткі теоретичні відомості

Загрози сучасного кіберсвіту є реальною небезпекою. Ці загрози можуть призвести до хаосу у сучасному комп'ютер-центричному світі. Про ці загрози повинен знати кожен, однак повністю нейтралізувати їх зможуть тільки професіонали, які зможуть розпізнавати загрози та нейтралізувати кібер-злочинців. Силами таких організацій, як CompTIA, Cisco Systems та ISC2 та інших створені та реалізуються програми для навчання та сертифікації спеціалістів у галузі кібербезпеки.

Порядок виконання роботи

Частина 1: Дослідження загрози від кібератак

Поділіться на групи по 3-4 особи. Після перегляду відео, дайте відповідь на наступні питання.

a. Перегляньте відео “Navigating today's cyber security challenges”. Згідно думки авторів відео, які сучасні виклики с точки зору безпеки стоять перед бізнесом? Ви згідні з такою думкою? URL: https://www.youtube.com/watch?v=omQ_oDIPxgY.

b. Назвіть головні 5 способів порушення закону з використанням комп'ютерів. Як ви вважаєте, чи можуть ці

злочини вплинути на вас особисто? Ви чи ваші рідні були жертвами таких злочинів?

c. Назвіть самі розповсюджені типи кібератак. Як протистояти їм? URL**:**

https://www.mass.gov/info-details/know-the-types-of-cyber-threats

https://www.coursera.org/articles/types-of-cyber-attacks

http://surl.li/wamuwd

d. Проаналізуйте дослідження Deloitte The Global Future of Cyber Survey. Завантажте звіт. URL**:** http://surl.li/swlknp.

- До яких головних негативних наслідків для респондентів призводять інциденти та порушення кібербезпеки?
- Як респонденти реагують на нові кіберризики, пов'язані з появою GenAI?
- Які переваги очікують респонденти від ініціатив у галузі кібербезпеки?
- Де і як респонденти бачать ШІ як інструмент у своїх програмах кібербезпеки?

e. Вплив і масштаби недавніх кібератак викликає занепокоєння у багатьох ділових колах та державних структурах. Розкажіть про найбільш руйнівні кібератаки, які відбулися в останні роки.

Частина 2: Тріада CIA

Тріада CIA відноситься до конфіденційності, цілісності та доступності, описуючи модель, призначену для керівництва політиками інформаційної безпеки (infosec) в організації.

1: Дослідіть тріаду CIA.

a. Перегляньте відео. URL**:** https://www.youtube.com/watch?v=bhLbnOa4wno

- Що таке конфіденційність даних? Чому конфіденційність даних, що належать приватним осібам та організаціям є настільки важливою?
- Що таке цілісність даних? Вкажіть три способи порушення цілісності та достовірності даних.

• Що таке доступність системи? Що може статися, якщо критично важлива комп'ютерна система стане недоступною?

2: Дослідіть майбутні перспективи.

Підготуйте відповіді на наступні питання:

• Що нас очікую у майбутньому?

• Чи готові ми до збільшення кібератак?

• Які проблеми мають великі держави з кіберзлочинністю?

За результатами лабораторної роботи оформіть звіт.

Контрольні питання

1. Що таке кібератака?
2. Що таке тріада CIA?
3. Назвіть самі розповсюджені типи кібератак.

ЛАБОРАТОРНА РОБОТА №2. ГЕНЕРАЦІЯ ПАРОЛІВ. ЗЛАМ ПАРОЛІВ. ОЦІНКА СТУПЕНЯ СТІЙКОСТІ ПАРОЛЬНОГО ЗАХИСТУ

Мета: Генерація стійких паролів . Оцінка ступеня стійкості. Вияснення паролю користувача з використанням утіліти для зламу пароля.

Необхідні ресурси

ПК з Ubuntu LTS, що встановлена на віртуальній машині VirtualBox або VMware.

LastPass https://www.lastpass.com/

ohn the Ripper URL: http://www.openwall.com/john/

Короткі теоретичні відомості

Генератор паролів – програма для створення надійних паролів.

Оцінка ступень стійкості (надійність) паролю

Відомо, що основними методами злому паролів є:

• метод атаки по словнику (перебираються паролі, які є осмисленими комбінаціями символів або найбільш поширені парольні комбінації);

• гібридні атаки (перебираються паролі зі словника, але деяким чином доповнені). Так, наприклад, гібридна атака може перевіряти не тільки паролі зі словника, але і їх об'єднання, слова зі словника, записані в транслітеративній формі, паролі, отримані паролів словника зміною символів;

• метод повного перебору (генеруються всі можливі комбінації символів деякого алфавіту і підставляються як паролі). Основними об'єктами дослідження є паролі - комбінації певних символів. Оскільки найбільший інтерес представляє метод повного перебору (він дає гарантований успіх), то в основному завдання визначення стійкості паролів зводяться до комбінаторних завдань. У зв'язку з цим доцільно привести основні формули комбінаторики.

Вимоги до надійний паролів

1. Довжина пароля має бути мінімум вісім символів
2. Пароль повинен містити символи у верхньому і нижньому регістрах.
3. Пароль повинен містити число
4. Пароль повинен містити неалфавітний символ

Нехай A - потужність алфавіту паролів (кількість символів, які можуть бути використані при складанні пароля. Наприклад, якщо пароль складається тільки з малих англійських літер, то A = 26).

L - довжина пароля.

$S = A^L$ - число всіляких паролів довжини L, які можна скласти

З символів алфавіту A.

V - швидкість перебору паролів зловмисником.

T - максимальний термін дії пароля.

Тоді, ймовірність P підбору пароля зловмисником протягом строку його дії V визначається за такою формулою.

$$P = \frac{V * T}{S} = \frac{V * T}{A^L}$$

Цю формулу можна використовувати в зворотну сторону для вирішення наступного завдання.

ЗАВДАННЯ. Визначити мінімальні потужність алфавіту паролів A і довжину паролів L, що забезпечують можливість вибору пароля зловмисником не більше заданої P, при швидкості підбору паролів V, максимальному терміні дії пароля T.

Це завдання має неоднозначне рішення. При вихідних даних V, T, P однозначно можна визначити лише нижню межу S * числа всіляких паролів. Цілочисельне значення нижньої межі обчислюється за формулою:

$$S^* = \left\lceil \frac{V * T}{P} \right\rceil$$

$\lceil \; \rceil$ - - - частина числа, взята з округленням вгору.

Після знаходження нижньої межі S * необхідно вибрати такі A і L для формування S = AL, щоб виконувалася нерівність:

$$S^* \le S = A^L$$

При виборі S, що задовольняє нерівності, можливість вибору пароля зловмисника (при заданих V і T) буде менше, ніж задана P.

Необхідно відзначити, що при здійсненні обчислень за формулами, що наведені вище, величини повинні бути приведені до одних розмірностей.

Приклад.

Вихідні дані - $P = 10^{-6}$, T = 7 днів = 1 тиждень, V = 10 паролів / хвилину = 10 * 60 * 24 * 7 = 100800 паролів в тиждень.

Тоді,

$$S* = \left\lceil \frac{10800*1}{10^{-6}} \right\rceil = 108*10^{8}$$

Умові $S* \leq A^{L}$ задовольняють, наприклад, такі комбінації A і L, як A = 26, L = 8 (пароль складається з 8 малих символів англійського алфавіту), A = 36, L = 6 (пароль складається з 6 символів, серед яких можуть бути малі латинські букви і довільні цифри).

Порядок виконання роботи

1: Генерація паролю.

Встановить програму менеджера (генерації) паролів (наприклад, LastPass). Зайдіть на сторінку LastPass. Згенеруйте пароль Проаналізуйте, як залежить надійність паролю від довжини, та типу символів.

URL: https://www.lastpass.com/features/password- generator

2: Безпечне зберігання паролів

Популярним менеджером паролів є LastPass. Створіть пробний обліковий запис LastPass:

a. Відкрийте веб-браузер і перейдіть на https://lastpass.com/

b. Натисніть Start Trial для створення пробного облікового запису.

c. Заповніть поля, як зазначено в інструкції.

d. Встановіть майстер-пароль. Цей пароль дає вам доступ до вашого облікового запису LastPass.

e. Завантажте та встановіть клієнта LastPass відповідно до своєї операційної системи.

f. Відкрийте клієнт і увійдіть до системи за допомогою майстер-пароля LastPass.

g. Дослідіть менеджер паролів LastPass.

Коли ви додасте паролі до LastPass, де вони зберігаються?

Окрім вас, щонайменше, один інший суб'єкт має доступ до паролів. Хто цей суб'єкт?

3: За допомогою математичних методів оцінити ступень стійкості (надійність) паролю.

a. У таблиці 1 знайти для вашого варіанту значення характеристик P, V, T.

b. Обчислити за формулою нижню межу S * для заданих P, V, T.

c. Вибрати деякий алфавіт з потужністю A і отримати мінімальну довжину пароля L, при $S^* \leq A_j^L$ виконується умова

Варіанти завдань Таблиця 1

Варіант	***P***	***V*, *паролів/хвил***	***T*, *дні***
1	10^{-4}	15	14
2	10^{-5}	3	10
3	10^{-6}	10	5
4	10^{-7}	11	6
5	10^{-4}	100	12
6	10^{-5}	10	30
7	10^{-6}	20	21
8	10^{-7}	15	20
9	10^{-4}	3	15
10	10^{-5}	10	7

4. Злам паролів

В системі Ubuntu є чотири облікові записи користувачів: Аліса, Боб, Єва і Ерік. Вам потрібно вияснити паролі цих облікових записів, використовуючи утиліту для зламу паролів з відкритим вихідним кодом John the Ripper.

Відкрийте вікно терміналу в Ubuntu.

a. Увійдіть в Ubuntu, використовуючи ваші облікові дані. Як приклад: Користувач: cisco, Пароль: password.

b. Натисніть на значок терміналу, щоб відкрити термінал.

5: Запустіть John the Ripper.

a. У командному рядку введіть таку команду, щоб перейти в каталог, в якому знаходиться John the Ripper

cisco@ubuntu:~$ cd ~/Downloads/john-1.8.0/run

b. У командному рядку введіть таку команду:

```
cisco@ubuntu:~/Downloads/john-1.8.0/run$ sudo ./unshadow /etc/passwd /etc/shadow > mypasswd
```

cisco@ubuntu:~/Downloads/john-1.8.0/run$ sudo ./unshadow /etc/passwd /etc/shadow > mypasswd

Ця команда об'єднає файл /etc/passwd, в якому зберігаються облікові записи користувачів та файл /etc/shadow, де зберігаються паролі користувачів, в новий файл з ім'ям mypasswd.

УВАГА! Якщо вам не вдалося встановити John the Ripper можете скористатися наступним посиланням:

https://linuxhint.com/john_ripper_ubuntu/

Відкриваємо нашу діректорію (можна через консоль), і можна працювати

6: Відновіть паролі

a. Введіть в термінал наступну команду:

cisco@ubuntu:~/Downloads/john-1.8.0/run$ ./john --show mypasswd

```
cisco@ubuntu:~/Downloads/john-1.8.0/run$ ./john --show mypasswd
0 password hashes cracked, 5 left
```

Як показано вище, на даний момент немає зламаних паролів.

b. У командному рядку введіть таку команду:

cisco@ubuntu:~/Downloads/john-1.8.0/run$ ./john --wordlist=password.lst - - rules mypasswd --format=crypt

```
cisco@ubuntu:~/Downloads/john-1.8.0/run$ ./john --wordlist=password.lst --rules mypasswd --format=crypt
```

Програма John the Ripper використовує наперед визначений словник під назвою password.lst зі стандартним набором «правил» для обробки словника і витягує всі хеші паролів типу md5crypt, та crypt .

У наведених нижче результатах відображаються паролі для кожного облікового запису.

```
Loaded 8 password hashes with 8 different salts (crypt, generic crypt(3) [?/64])
Press 'q' or Ctrl-C to abort, almost any other key for status
password1        (Eric)
12345            (Bob)
123456           (Alice)
password         (cisco)
password         (Eve)
5g 0:00:20:50 100% 0.003998g/s 125.4p/s 376.6c/s 376.6C/s Tnting..Sssing
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

У командному рядку введіть таку команду:

cisco@ubuntu:~/Downloads/john-1.8.0/run$ ./john --show mypasswd

```
cisco@ubuntu:~/Downloads/john-1.8.0/run$ ./john --show mypasswd
cisco:password:1000:1000:Cisco,,,:/home/cisco:/bin/bash
Alice:123456:1001:1001::/home/Alice:
Bob:12345:1002:1002::/home/Bob:
Eve:password:1003:1003::/home/Eve:
Eric:password1:1004:1004::/home/Eric:

5 password hashes cracked, 3 left
cisco@ubuntu:~/Downloads/john-1.8.0/run$
```

Дайте відповідь на питання:
Скільки паролів було зламано?

За результатами лабораторної роботи оформіть звіт.

Контрольні питання

1. Назвіть вимоги до надійних паролів.
2. Назвіть основні методи зламу паролів.
3. Як оцінити надійність паролю?

ЛАБОРАТОРНА РОБОТА №3. ШИФРУВАННЯ ДАНИХ ПРИ ЗБЕРІГАННІ ЗА ДОПОМОГОЮ EFS

Мета: Знайомство з шифрувальної файлової системою EFS
Необхідні ресурси
ПК або мобільний пристрій з доступом до Інтернету
ОС Windows 10, 11

Короткі теоретичні відомості

Шифрована файлова система (EFS) - це компонент Windows, що дозволяє зберігати відомості на жорсткому диску в зашифрованому форматі. Дана система здійснює «прозоре шифрування» даних, що зберігаються у файловій системі NTFS.

Шифрування Encrypting File System здійснюється за допомогою відкритого і закритого ключів, автоматично генеруються системою при першому використанні вбудованих засобів EFS. У процесі шифрування каталогу або файлу система EFS створює унікальний номер (FEK), який шифрується майстер-ключем. У свою чергу майстер-ключ шифрується призначеним для користувача ключем. Що стосується закритого ключа користувача, то він також захищається, але на цей раз хешем призначеного для користувача системного пароля.

Причому, відкрити зашифровані EFS-системою файли можна тільки за допомогою тієї облікового запису, в якій вони були зашифровані. Навіть якщо жорсткий диск з захищеними даними буде знято і підключений до іншого комп'ютера, прочитати їх все одно не вдасться. З іншого боку, втрата користувачем пароля від свого облікового запису, пошкодження або перевстановлення операційної системи призведе до недоступності раніше зашифрованих файлів. На щастя, розробники Windows передбачили й такий сценарій, і запропонували просте рішення, а саме збереження сертифікатів шифрування на знімний носій.

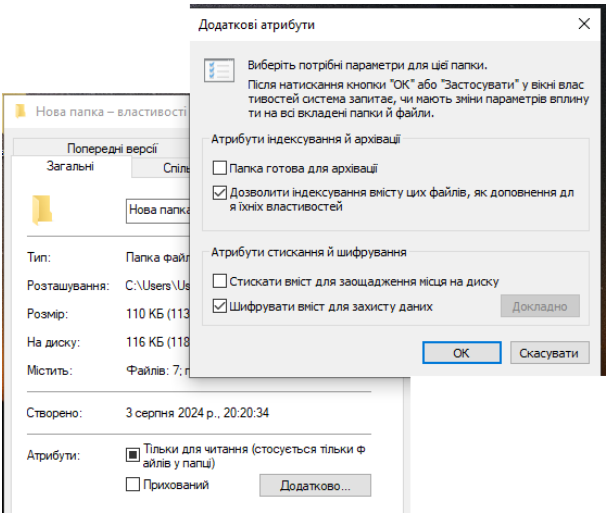
«Варто зауважити, що EFS не шифрується файли, що передаються по мережі, тому для захисту переданих даних необхідно використовувати інші протоколи захисту даних (IPSec або WebDAV)»[1].

Порядок виконання роботи

Ніяких попередніх налаштувань шифрування за допомогою EFS в Windows не вимагає. Припустимо, нам потрібно захистити папку з зображеннями. У властивостях папки вибираємо *Додатково*. Після чого в додаткових атрибутах встановлюємо галочку *Шифрувати вміст для захисту даних.*

Тиснемо *Застосувати* і підтверджуємо запит на зміну атрибутів. Застосувати шифрування можна тільки до одного каталогу або ж до каталогу і всього його змісту.

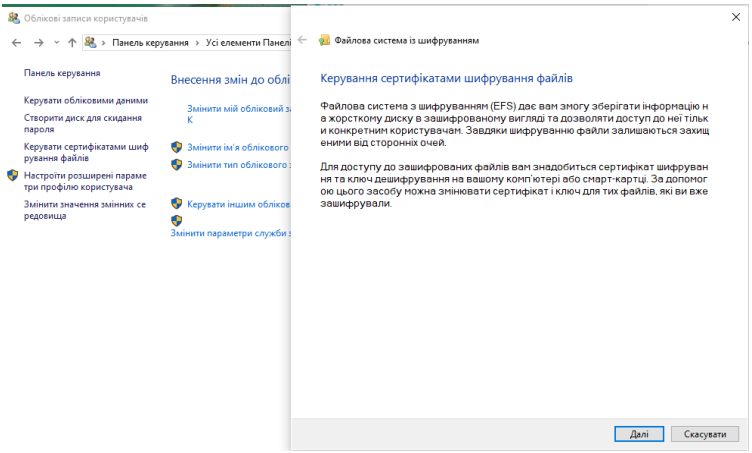
[1] https://uk.wikipedia.org/wiki/Encrypting_File_System



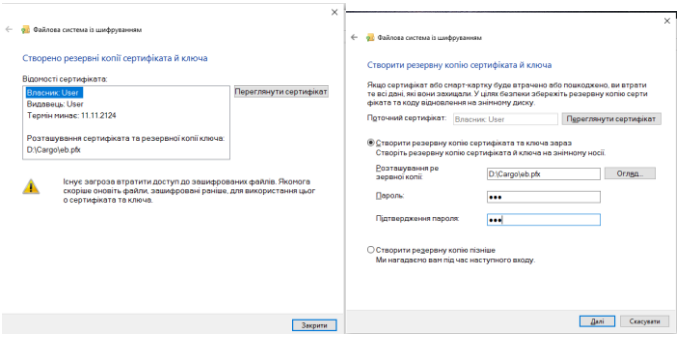
Якщо ви шифруєте дані вперше, система запропонує створити резервну копію ключа і сертифіката шифрування. Не варто нехтувати цією порадою, адже від випадкового пошкодження операційної системи або втрати пароля облікового запису Windows ніхто не застрахований.

Отримати доступ до сертифікатів можна в системному треї (панель задач, праворуч). Якщо повідомлення не з’явилося, скористуйтеся обліковим записом користувачів в панелі керування.

Зашифровані файли можна переглядати, редагувати, копіювати, видаляти та інше, при цьому шифрування і дешифрування буде проводитися на льоту, непомітно для користувача. Однак всі ці дії будуть доступні тільки для конкретної облікового запису. В принципі, таким чином зашифрувати можна будь-який файл або папку за винятком системних. Більш того, шифрувати останні категорично не рекомендується, так як це може привести до неможливості завантаження Windows.

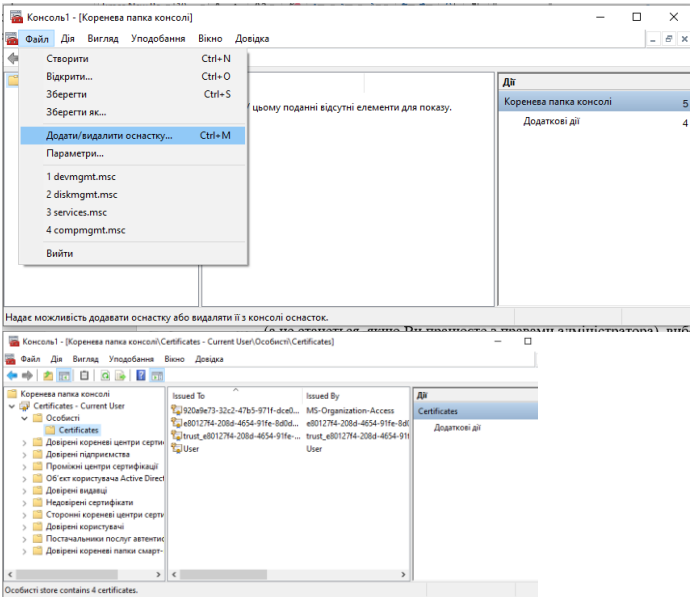


Отже, у вікні майстра резервної копії чітко слідуємо вказівкам. Далі подаємо ім'я файлу сертифікату і зберігаємо його на заздалегідь підключений знімний носій.



Для підтвердження автентичності відкритих ключів використовуються *сертифікати.* Тепер розглянемо, як зберігаються сертифікати. Операційна система Windows забезпечує захищене сховище ключів і сертифікатів. Працювати зі сховищем можна використовуючи налаштування консоль управління mmc "*Сертифікати*". З меню *Пуск -> Виконати* запустите консоль командою *mmc*. У меню *Файл* виберіть

Додати / видалити оснастку, а в списку оснасток виберіть *Сертифікати*. Якщо буде запропоновано вибір (а це станеться, якщо Ви працюєте з правами адміністратора), виберіть пункт Це оснащення завжди буде керувати сертифікатами для: "Мого облікового запису".

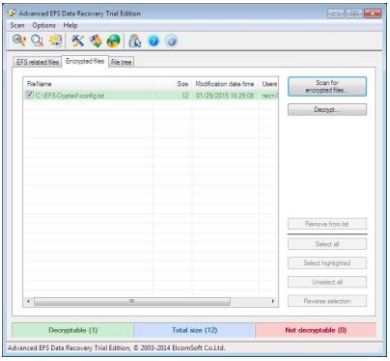


Таким чином, ми можемо переглядати сертифікати поточного користувача.

Використання програми Advanced EFS Data Recovery для розшифрування зашифрованих EFS файлів

Утиліта Advanced EFS Data Recovery - працює в двох режимах - майстер і експерт (Expert mode).

Перейдіть на вкладку EFS-дані (перша вкладка). Додайте пароль користувача для прискорення пошуку ключів (введіть ім'я користувача (ваш обліковий запис) і пароль як текст. Це прискорить час пошуку ключів.



Проскануйте ключі. Перейдіть на вкладку з зашифрованими файлами (2 вкладка). Знайдіть зашифровані файли. Результат пошуку повинен бути підсвічений зеленим кольором (як на малюнку вище). Виділіть потрібний файл і натисніть кнопку *Дешифрувати*. Перейдіть в папку, де розміщений розшифрований файл. Відкрийте його.

Завдання до лабораторної роботи.

1. Працюючи під своїм обліковим записом, створіть папку, помістивши туди текстовий файл з розширенням txt. У файлі запишіть своє прізвище. (Обмеження на розмір файлу пов'язаний з тим, що в подальшому будуть використовуватися пробні версії програм, які мають обмеження на розмір файлу). Зробіть шифрування папки з файлом txt. Спробуйте зробити додавання файлів, перейменування папки, копіювання її на інший диск. Опишіть ваші дії і результати, демонструючи їх скриншотами.
2. Спробуйте прочитати зашифрований файл під іншим користувачем. Що буде відбуватися? (для цього необхідно попередньо створити додатковий обліковий запис). Опишіть ваші дії і результати, демонструючи їх скриншотами.
3. Зайдіть під своїм обліковим записом. Видаліть сертифікат користувача (консоль mmc, Certificates). Завершіть сесію (краще перезавантажити комп'ютер) і увійдіть знову. Спробуйте відкрити зашифрований файл. Опишіть ваші дії і результати, демонструючи їх скриншотами.

4. Розшифруйте папку з використанням програми Advanced EFS Data Recovery. Для цього зайдіть в систему під обліковим записом адміністратора. Для цього додається до лабораторної роботи дистрибутив демо-версії програми (при запиті ключа ліцензії це питання ігнорувати і виконувати установку далі). Дії по роботі з програмою і розшифровкою файлу опишіть в звіті і продемонструйте результати скриншотами.

За результатами лабораторної роботи оформіть звіт.

Контрольні питання

1. За допомогою чого підтверджується автентичність відкритих ключів?
2. Що таке відкритий ключ?
3. Дайте характеристику системі Encrypting File System – EFS.

ЛАБОРАТОРНА РОБОТА №4. РОЗМЕЖУВАННЯ ПРАВ В LINUX

Мета: Отримання практичних навичок роботи в консолі з атрибутами файлів, закріплення теоретичних основ дискреційного розмежування доступу в сучасних системах з відкритим кодом на базі ОС Linux.

Необхідні ресурси

ПК з Ubuntu LTS, встановленою у VirtualBox або на віртуальній машині VMware.

Підготовка до виконання лабораторної роботи

1. В встановленої при виконанні попередньої лабораторної роботи операційної системи ласка зареєструйтесь користувача guest: useradd guest

2. Задайте пароль для користувача guest: passwd guest

3. Аналогічно створіть другого користувача guest2.

4. Додайте користувача guest2 в групу guest: gpasswd -a guest2 guest

Завдання 1. Основні атрибути

Постарайтеся послідовно виконати всі пункти, заносячи ваші відповіді на поставлені питання і зауваження до звіту.

1. Увійдіть в систему від імені користувача guest.

2. Визначте директорію, в якій ви перебуваєте, командою pwd. Порівняйте її із запрошенням командного рядка. Визначте, чи є вона вашої домашньої директорією? Якщо немає, зайдіть в домашню директорію.

3. Уточніть ім'я вашого користувача командою whoami.

4. Уточніть ім'я вашого користувача, його групу, а також групи, куди входить користувач, командою id. виведені значенняuid, gid та ін. запам'ятаєте. Порівняйте результат роботи id з команди groups.

5. Порівняйте отриману інформацію про ім'я користувача з даними, виведеними в запрошенні командного рядка.

6. перегляньте файл / Etc / passwd командою cat / etc / passwd

Знайдіть в ньому свій обліковий запис. Визначте uid користувача. Визначте gid користувача. Порівняйте знайдені значення з отриманими в попередніх пунктах.

Зауваження: в разі, коли результат роботи команди не вміщується на одному екрані монітора, натискайте стрілку вгору-вниз (утримуючи клавішу shift, натискайте page up і page down) Або програму grep в якості фільтра для виведення тільки рядків, що містять певні літерні сполучення:

cat / etc / passwd | grep guest

7. Визначте існуючі в системі директорії командою ls -l / home /

Чи вдалося вам отримати список піддиректорій директорії / home? Які права встановлені на директоріях?

8. Перевірте, які розширені атрибути встановлені на поддіректоріях, що знаходяться в директорії / home, Командою:

lsattr / home

Чи вдалося вам побачити розширені атрибути директорії?

Чи вдалося вам побачити розширені атрибути директорій інших користувачів?

9. Створіть в домашній директорії піддиректорію dir1 командою mkdir dir1

Визначте командами ls -l і lsattr, які права доступу і розширені атрибути були виставлені на директорію dir1.

10. Зніміть з директорії dir1 всі атрибути командою chmod 000 dir1

і перевірте з її допомогою правильність виконання команди ls -l

11. Спробуйте створити в директорії dir1 файл file1 командою echo "test"> / home / guest / dir1 / file1

Поясніть, чому ви отримали відмову у виконанні операції по створенню файлу?

Оцініть, як повідомлення про помилку відбилося на створенні файлу? Перевірте командою чи дійсно файл file1 не всередині директорії dir1.

ls -l / home / guest / dir1

12. Заповніть таблицю «Встановлені права і дозволені дії» (див. Табл. 1), виконуючи дії від імені власника директорії (файлів), визначивши шляхом за допомогою досвіту, які операції дозволені, а які ні. Якщо операція дозволена, занесіть в таблицю знак «+», якщо не дозволена, знак «-».

Мінімальні права для здійснення операцій Таблиця 1

Права директорії	Права файлу	Створення файлу	Видалення файлу	Запис в файл	Зміна директорії	Перегляд файлів в директорії	Перейме-нування файлу	Зміна атрибутів файлу
d(000)	(000)	-	-	-	-	-	-	-
d--x------ (100)	(000)	-	-	-	+	-	-	+
drwx------ (700)	-rwx------ (700)	+	+	+	+	+	+	+

Зауваження 1: при заповненні табл. 1 розглядаються лише перші три атрибути файлів і директорій, а: r, w, x, для «власника». Решта атрибутів також важливі (особливо при використанні доступу від імені різних користувачів, що входять в ті чи інші групи). Перевірка всіх атрибутів при всіх умовах значно збільшила б таблицю. В даному прикладі пропонується розглянути 3 + 3 атрибута, тобто $2^6 = 64$ варіанти.

Зауваження 2: в ряді дій при виконанні команди видалення файлу ви можете зіткнутися з питанням: «видалити захищений від запису пустий звичайний файл dir1 / file1?» Зверніть увагу, що наявність цього питання не дає змоги зробити правильний висновок про те, що файл можна видалити. У ряді випадків, при відповіді «у» (так) на зазначене питання, можливо отримати ще одне повідомлення: «неможливо видалити dirl / file1: Немає доступу".

13. На підставі заповненої таблиці визначте ті чи інші мінімально необхідні права для виконання операцій усередині директорії dir1, заповніть табл. 2.

Завдання 2. Два користувача

1. Здійсніть вхід в систему від двох користувачів на двох різних консолях: guest на першій консолі і guest2 на другий консолі.

2. Для обох користувачів командою pwd визначте директорію, в якій ви перебуваєте. Порівняйте її з запрошеннями командного рядка.

3. Уточніть ім'я вашого користувача, його групу, хто входить в неї до яких груп належить він сам. Визначте командами groups guesti groups guest2, в які групи входять користувачі guest і guest2. Порівняйте результати роботи команди groups з командами id -Gni id -G.

Мінімальні права для здійснення операцій Таблиця 2

операція	мінімальні права на директорію	мінімальні права на файл
створення файлу		
видалення файлу		
читання файлу		
запис в файл		
перейменування файла		
створення піддиректорії		
видалення піддиректорії		

3. Порівняйте отриману інформацію з вмістом файла / Etc / group. Перегляньте файл командою

cat / etc / groupВід імені користувача guest2 зареєструйте користувача guest2 в групі guest командою newgrp guest

4. Від імені користувача guest змініть права директорії / Home / guest, Дозволивши всі дії для користувачів групи:

chmod g + rwx / home / guest

5. Від імені користувача guest зніміть з директорії / Home / guest / dir1 всі атрибути командою

chmod 000 dirl

и перевірте правильність зняття атрибутів.

7.Змінюючи атрибути у директорії dir1 і файлу file1 від імені користувача guest і роблячи перевірку від користувача

guest2, заповніть табл 3. визначивши шляхом досвіду, які операції дозволені, а які ні. Якщо операція дозволена, занесіть в таблицю знак «+», якщо не дозволена, знак «-».

Мінімальні права для здійснення операцій (для груп)Таблиця 2

Права директорії	Права файлу	Створення файлу	Видалення файлу	Запис в файл	Зміна директорії	Перегляд файлів в директорії	Перейме-нування файлу	Зміна атрибутів файлу
d(000)	(000)	-	-	-	-	-	-	-
d--x------(010)	(000)	-	-	-	+	-	-	+
d--- rwx--- (070)	-rwx--- (070)	+	+	+	+	+	+	+

Порівняйте табл. 1 і табл. 3.

На підставі заповненої таблиці визначте ті чи інші мінімально необхідні права для виконання користувачем guest2 операцій всередині директорії dir1 і заповніть табл. 1, змінивши її назву на «Мінімальні права для здійснення операцій від імені користувачів, що входять в групу».

За результатами лабораторної роботи оформіть звіт.

Контрольні питання

1. Що таке розмежування прав доступу?
2. Якою командою можна змінити всі атрибути для директорії?

ЛАБОРАТОРНА РОБОТА №5
ДОСЛІДЖЕННЯ БЕЗПЕКИ МЕРЕЖІ ЗА ДОПОМОГОЮ NMAP

Мета роботи: формування навичків дослідження безпеки мережі за допомогою nmap.

Необхідні ресурси

ПК або мобільний пристрій з доступом до Інтернету
ОС Windows 10, 11

NMAP. URL: https://nmap.org/

Короткі теоретичні відомості

1.1 Загальні характеристики

Nmap «призначений для сканування мереж з будь-якою кількістю об'єктів, визначення стану об'єктів сканируемой мережі, а також портів і відповідних їм служб. Для цього nmap використовує багато різних методів сканування, таких»[2], як:

UDP connect (),
TCP connect (),
TCP SYN (напіввідкрите),
FTP proxy (прорив через ftp),
Reverse-ident,
ICMP (ping),
FIN-сканування,
ACK-сканування,
Xmas tree-сканування,
SYN-сканування,
NULL-сканування.

Результатом роботи Nmap є список відсканованих портів віддаленої машини із зазначенням номера та стану порту, типу використовуваного протоколу, а також назви служби, закріпленої за цим портом. Порт характеризується трьома можливими станами «відкритий», «фільтрується» і «нефільтрованого»:

- відкритий (open) - дистанційна машина прослуховує даний порт;
- фільтрується (filtered) - міжмережевий екран, пакетний фільтр або інший пристрій блокує доступ до цього порту і nmap не зміг визначити його стан;
- «Нефільтрованого» (closed) - за результатами сканування nmap сприйняв даний порт як закритий, при цьому засоби захисту не завадили nmap визначити його стан. Цей стан nmap визначає в будь-якому

[2] https://thepresentation.ru/informatika/dosldzhennya-strukturi-merezh-nformatsyno-telekomunkatsyno-sistemi

випадку (навіть якщо більшість сканованих портів хоста фільтруються).

Залежно від зазначених опцій, nmap також може визначити наступні характеристики об'єкту сканування хоста:

- операційна система хоста,
- метод генерації TCP ISN,
- ім'я користувача власника процесу, зарезервованих сканований порт,
- символьні імена, відповідні документи, що скануються IP-адресами і т.д.

1.2 Методи сканування

1.2.1 TCP connect () (-sT)

Найбільш загальний метод сканування TCP портів. Функція connect(), яка присутня в будь-який ОС, дозволяє створити з'єднання з будь-яким портом видаленої машини. Якщо зазначений як аргумент порт відкритий і прослуховується скануемой машиною, то результат виконання connect () буде успішним (тобто з'єднання буде встановлено), в іншому випадку вказаний порт є закритим, або доступ до нього заблоковано засобами захисту.

Для того, щоб використовувати цей метод, користувач може не мати ніяких привілеїв на скануючому хості. Цей метод сканування легко виявляється цільовим (тобто Скануемое) хостом, оскільки його log-файл буде містити запротокольовані численні спроби з'єднання і помилки виконання даної операції. Служби, які обробляють підключення, негайно заблокують доступ адресою, який викликав ці помилки.

1.2.2.TCP SYN (-sS)

Сканування SYN є стандартним і найпопулярнішим варіантом сканування. Його можна виконати швидко, скануючи тисячі портів за секунду у швидкій мережі. Сканування SYN є відносно ненав'язливим і прихованим. Це також дозволяє чітко та надійно розрізняти стани open, closed, та filtered .

Сканування SYN можна запитати, передавши -sS опцію в Nmap. Для цього потрібні привілеї raw-пакетів, і є стандартним скануванням TCP, якщо вони доступні. Таким чином, коли Nmap

запускається як root або адміністратор, -sS зазвичай не вказується. Приклад: сканування SYN показує три стани портів

```
krad# nmap -p22,113,139 scanme.nmap.org

Starting Nmap ( https://nmap.org )
Nmap scan report for scanme.nmap.org (64.13.134.52)
PORT    STATE    SERVICE
22/tcp  open     ssh
113/tcp closed   auth
139/tcp filtered netbios-ssn

Nmap done: 1 IP address (1 host up) scanned in 1.35 seconds
```

1.2.3 «Невидиме» FIN, Xmas Tree і NULL-сканування (-sF, -sN, -sX)

Ці методи використовуються в разі, якщо SYN-сканування з яких-небудь причин виявилося непрацездатним. Ідея полягає в наступному. У FIN-скануванні в якості запиту використовується FIN-пакет. У Xmas Tree використовується пакет з набором прапорів FIN | URG | PSH, а NULL-сканування використовує пакет без прапорів. ОС хоста, що сканується, повинна відповісти на такий пакет, який прибув на закритий порт, пакетом RST, в той час як відкритий порт повинен ігнорувати ці пакети. Розробники Microsoft Windows, як зазвичай, вирішили повністю ігнорувати всі загальноприйняті стандарти і піти своїм шляхом. Тому будь-яка ОС сімейства Windows не посилає у відповідь RST-пакет, і дані методи не будуть працювати з цими ОС. Однак у всьому є свої плюси, і в nmap ця ознака є основним для розрізнення операційних систем, що володіють такою властивістю. Якщо в результаті FIN-сканування ви отримали список відкритих портів, то це не Windows. Якщо ж всі ці методи видали результат, що всі порти закриті, а SYN-сканування виявило відкриті порти, то ви скоріше за все маєте справу з ОС Windows.

1.2.4. Методи виявлення хоста

Nmap пропонує широкий спектр методів виявлення хостів, окрім стандартного exo-запиту ICMP.

TCP SYN Ping ( )-PS<port list> Опція –PS надсилає порожній TCP-пакет із встановленим прапором SYN. Типовим портом призначення є 80 (можна налаштувати під час компіляції шляхом заміни DEFAULT_TCP_PROBE_PORT_SPEC в nmap.h), але альтернативний порт можна вказати як параметр. Можна вказати список портів (наприклад, -PS22-25,80,113,1050,35000), і в цьому випадку зонди будуть спробовані для кожного порту паралельно.

TCP ACK Ping ( )-PA<port list> Пінг TCP ACK дуже схожий на пінг SYN. Різниця, як ви могли здогадатися, полягає в тому, що прапор TCP ACK встановлено замість прапора SYN. Такий пакет ACK претендує на підтвердження даних через встановлене з'єднання TCP, але такого з'єднання не існує. Тому віддалені хости повинні завжди відповідати пакетом RST, розкриваючи своє існування в процесі.

UDP Ping ( )-PU<port list> Іншим варіантом виявлення хоста є UDP ping, який надсилає UDP-пакет на вказані порти. Список портів приймає той самий формат, що й для попередньо обговорених -PSопцій -PA . Якщо порти не вказано, за замовчуванням буде 40 125. Це типове значення можна налаштувати під час компіляції шляхом зміни DEFAULT_UDP_PROBE_PORT_SPEC в nmap.h. За замовчуванням використовується вкрай незвичайний порт, оскільки надсилання на відкриті порти часто є небажаним для цього конкретного типу сканування.

Типи ping ICMP (-PE, -PPi -PM) На додаток до незвичайних типів виявлення хостів TCP і UDP, які обговорювалися раніше, Nmap може надсилати стандартні пакети, надіслані повсюдною програмою ping . Nmap надсилає пакет ICMP типу 8 (exo-запит) на цільові IP-адреси, очікуючи у відповідь тип 0 (exo-відповідь) від доступних хостів.

Ping протоколу IP ( )-PO<protocol list> Найновішим варіантом виявлення хоста є ping протоколу IP, який надсилає IP-пакети з указаним номером протоколу в їх IP-заголовку. Список протоколів приймає той самий формат, що й списки портів у розглянутих раніше варіантах виявлення хостів TCP і UDP. Якщо

протоколи не вказано, за замовчуванням надсилається кілька IP-пакетів для ICMP, IGMP і IP-in-IP.

1.2.5 Визначення версій

Після того, як визначені відкриті TCP і / або UDP порти за допомогою будь-якого методу сканування, nmap взаємодіє з цими портами намагаючись визначити що саме ховається за ними. Nmap намагається визначити протокол обміну служби (наприклад, ftp, ssh, telnet, http), назва програми (наприклад, ISC Bind, Apache http, Solaris telnetd), номер версії та іноді різні деталі, наприклад, чи є можливість підключиться до X-сервера або номер версії SSH-протоколу. Якщо nmap зібраний з підтримкою OpenSSL, він створює з'єднання з SSL-серверами намагаючись визначити що приховано за шифрованих каналом. Якщо виявлені RPC-служби, nmap визначає програми, які обслуговують RPC-порти, і їх версії. Деякі UDP порти залишаються в стані «open | filtered» після UDP-сканування, якщо сканування не змогло визначити, чи є порт відкритим або фільтрується,.

1.2.6. Сканування TCP ACK (-sA)

Це сканування відрізняється від інших, розглянутих досі, тим, що воно ніколи не визначає open(або навіть open|filtered) порти. Він використовується для відображення наборів правил брандмауера, визначення того, чи мають вони статус чи ні, і які порти фільтруються. ACK-сканування вмикається, якщо вказати -sAопцію.

Інтерпретація відповіді на запит сканування ACK

Таблиця 1.

Відповідь зонда	Присвоєний стан
Відповідь TCP RST	unfiltered
Відповіді не отримано (навіть після повторної передачі)	filtered
Помилка недоступності ICMP (тип 3, код 1, 2, 3, 9, 10 або 13)	filtered

1.2.7. «Прорив через FTP»

Цікавою «можливістю» протоколу є підтримка «довірених» (proxy) ftp-з'єднань. Тобто, з довіреного хоста source.com можна з'єднатися з FTP-сервером target.com і відправити файл, що знаходиться на ньому, на будь-яку адресу Internet. Nmap використовує цю можливість для сканування портів з «довірою» FTP-сервера. Отже, ви можете підключитися до FTP-сервера «над» фаєрволом і потім просканувати заблоковані їм порти (наприклад, 139-й). Якщо ftp-сервер дозволяє читати і записувати дані в будь-якої каталог (наприклад, / incoming), ви також можете відправити будь-які дані на ці порти.

1.3 Визначення ОС віддаленого хоста

Визначення типу і версії операційної системи віддаленого хоста є досить актуальним на початковому етапі реалізації атаки на хост. Залежно від того, яка ОС встановлена на віддалений хост, атакуючий буде планувати свої подальші дії, впливаючи на відому «діру» (якщо така є) в безпеки встановленої на хості ОС. При цьому, чим точніше атакуючий визначить тип і версію ОС віддаленого хоста, тим ефективніше буде виконаний його «злом». У підтвердженні цього, розглянемо кілька можливих ситуацій.

Припустимо, здійснюється спроба проникнення на віддалений хост. В результаті сканування портів було виявлено, що 53-й порт хоста відкритий. На підставі даної ознаки можна припустити, що на хості встановлена одна з версій ОС UNIX, і виконується одна з уразливих версій демона bind. Якщо це вельми умовне припущення є вірним, атакуючий має тільки одну спробу використовувати виявлену "діру", оскільки невдала спроба атаки «підвісить» демона і порт виявиться закритим, після чого атакуючому доведеться шукати нові «дірки» в безпеці віддаленого хоста.

Якщо атакуючий точно визначить тип і версію ОС віддаленого хоста, він може відповідним чином скоординувати свої дії, проаналізувавши інформацію, що стосується відомих проблем в безпеці певної ОС.

Використовуючи програмні засоби, що забезпечують визначення ОС віддаленого хоста, атакуючий здатний просканувати безліч хостів і визначити тип і версію ОС, встановлену на кожному з них. Потім, коли хто-небудь опублікує в мережі Інтернет інформацію про виявлену «діру» в безпеки конкретної ОС, атакуючий автоматично отримує список уразливих хостів, на яких встановлена дана версія ОС.

1.3.1 Реалізація методу комплексного опитування стека TCP / IP в NMAP

Розглянемо реалізацію методу дослідження ОС віддаленого хоста шляхом комплексного опитування його TCP / IP стека, званим інакше методом зняття «відбитків» (fingerprint) стека TCP / IP і використовуваним в сканері NMAP.

Для визначення ОС віддаленого хоста, версія якої невідома, необхідно мати певну інформацію про те, як ОС відомих версій реагують на певні види запитів, описаних вище, інакше кажучи - скласти «відбиток» стека TCP / IP операційної системи.

Для цього необхідно віддалений або локальний хост, тип і версія ОС якого заздалегідь відомі, протестувати усіма описаними вище способами, проаналізувати результати тестів і на основі отриманих даних скласти загальну характеристику (або т.зв. «відбиток») стека TCP / IP віддаленого хоста, прив'язавши його до конкретного типу і версії ОС.

Зібравши достатньо велика кількість таких відбитків (хоча можна починати і з одним), можливо тими ж методами досліджувати хост, тип і версія ОС якого заздалегідь невідома. Склавши з отриманих результатів відбиток і зіставивши його з вже наявними, можна визначити, який ОС відповідає отриманий відбиток і на підставі цього зробити висновок про ОС досліджуваного хоста.

Алгоритм отримання відбитка стека TCP / IP наступний. Спочатку проводиться сканування портів віддаленого хоста з метою визначення відкритих портів і служб, що функціонують на досліджуваному хості. Потім проводиться кілька тестів,

поетапно виконують опитування стека TCP / IP віддаленого хоста з метою виявлення розглянутих вище ознак.

На основі отриманих від хоста відповідей складається відбиток, який потім порівнюється з уже наявною базою відбитків, і приймається рішення про тип і версії ОС досліджуваного хоста.

2. Використання

nmap [Метод (и) сканування] [Опції] <Хост або мережу # 1, [# N]>

2.1 Опції (частково)

2.1.1 Опції вибору методу сканування

-sTTCP Connect ()
-sSTCP SYN
-sF FIN-сканування
-sX Xmas Tree сканування
-sN NULL-сканування
-sP Ping-сканування
-sV визначення версій
-sU UDP-сканування
-sO сканування протоколів IP
-sI <Zombie_хост [: порт]> - Сканування «вхолосту»
-sA ACK-сканування
-sW TCP Window
-sR RPC-сканування

-b<Ftp relay host> - «Прорив через FTP». Як аргумент передаєтсяURL ftp-сервера, що використовується в якості «довіреної» (ім'я користувача: пароль @ сервер: порт)

2.1.2 Деякі опції настройки і вибору додаткових можливостей

-P0 Не проводити ping-опитування хостів перед їх безпосереднім скануванням. Ця опція дозволяє просканувати мережі, що блокують обробку ICMP-луни за допомогою міжмережевих екранів. Прикладом такої мережі є microsoft.com.

-O Ця опція дозволяє визначити операційну систему сканування хоста за допомогою методу TCP / IP fingerprint. Іншими словами, nmap активізує потужний алгоритм, що функціонує на основі аналізу властивостей мережевого програмного забезпечення встановленої на ньому ОС. В результаті сканування виходить формалізований «відбиток», що складається зі стандартних тестових запитів і «відповідей» хоста на них. Потім отриманий відбиток порівнюється з наявною базою стандартних відповідей відомих ОС, і на підставі цього приймається рішення про тип і версії ОС скануєтся хоста. Цей метод вимагає наявності хоча б одного закритого і одного відкритого порту на цільовому хості.

-p <Діапазон (и) _портів>

Ця опція вказує nmap, які порти необхідно просканувати. Наприклад, '-p 23' означає сканування 23 порту на цільовій машині. Якщо вказано вираз типу '-p 20-30,139,60000-' Nmap буде сканувати порти з номерами з 20 по 30 включно, 139 і від 60000 і вище (до 65535). За замовчуванням nmap сканує всі порти в діапазоні 1-1024

-v (Збільшити рівень вербальності) Збільшити рівень деталізації результатів сканування.

Збільшення рівня вербальності тягне за собою висновок більшої кількості інформації під час сканування. Коли Nmap передбачає, що сканування займе більше кількох хвилин, буде

виводитися приблизний час завершення роботи і відкриті порти в міру їх виявлення. Задайте цю опцію двічі або більше для збільшення кількості виведеної інформації.

2.2 Способи завдання цільового хоста

Все, що не є опцією або її аргументом, nmap сприймає як адреса або ім'я цільового хоста (тобто хоста, що піддається скануванню). Найпростіший спосіб задати сканований хост - вказати його ім'я або адреса в командному рядку після вказівки опцій і аргументів. Якщо ви хочете просканувати підмережу IP-адрес, вам необхідно вказати параметр '/ mask' («маска») після імені або IP-адреси хоста, що сканується.

Nmap дозволяє також гнучко вказати цільові IP-адреси, використовуючи списки і діапазони для кожного їх елемента. Наприклад, необхідно просканувати підмережу класу B з адресою 128.210. *. *. Задати цю мережу можна будь-яким з наступних способів: 128.210. *. *, 128.210.0-255.0-255, 128.210.1-50,51-255.1,2,3,4,5-255,128.210.0.0 / 16

Наведемо ще один приклад. Якщо ви вказали в якості цільового IP-адреси рядок '*. *. 5.6-7', nmap сканує всі IP-адреси, що закінчуються на 5.6 або 5.7.

-iR- для сканування в межах всього Інтернету або будь-яких досліджень, вам, можливо, потрібно буде вибрати цілі довільно.

Завдання до лабораторної роботи

Всі пункти завдань виконуються на ОС в Ubuntu за допомогою утиліти nmap (https://nmap.org/). Для виконання деяких команд вам потрібно повноваження супер (root).

1. Отримайте список відкритих портів (TCP і UDP) машини (localhost). Виконайте декілька методів сканування. Порівняйте результати.
2. Визначте операційну систему цієї ж машини.

3. Визначте, чи працює хост localhost (пінг-сканування)

4. За допомогою конструкції команди Nmap вибрати випадковим чином 500 хостів і просканувати їх на наявність запущених на них веб-серверів (в протоколі HTTP). Скористайтеся опцією, яка активує вербальний режим. Перебір хостів відключити опцією -PN, тому що посилка пари попередніх запитів з метою визначення доступності хоста є недоцільною, коли вас цікавить лише один порт на кожному хості.

5. Проскануйте всі TCP порти хоста scanme.nmap.org. Скористайтеся опцією, яка збільшує рівень деталізації результатів сканування. Порівняйте результати відпрацювання команди з це опцією і без неї.

6. Визначити, які IP протоколи (TCP, ICMP, IGMP і т.д.) підтримуються хостом scanme.nmap.org

7. Визначте захищені фаєрволом порти хоста scanme.nmap.org.

8. Проведіть потайне SYN сканування підмережі машин мережі «класу С», в якій розташована машина Scanme. Також зробіть спробу визначення операційної системи на кожному працюючому хості. Вам буде потрібно повноваження супер (root).

9. Для чотирьох хостів, (вибрати самостійно), виконайте наступні дії Виберіть оптимальний метод сканування портів (обґрунтуйте).

10. Визначте, чи захищений хост фаєрволом.

11. Визначте підтримувані протоколи.

12. Визначте відкриті і Фільтровані порти і по можливості версії служб.

За результатами лабораторної роботи оформіть звіт.

Контрольні питання

1. Якими можливостями володіє утиліта nmap?
2. Дайте порівняльну характеристику основних методів сканування.
3. Які стани портів розпізнає nmap (і їх характеристика)?
4. Який стан порту дозволяє нам зробити висновок, що порт може бути захищений брандмауером?

ЛАБОРАТОРНА РОБОТА №6
ДОСЛІДЖЕННЯ ПРОЦЕСУ ШИФРУВАННЯ ПОВІДОМЛЕННЯ ЗА ДОПОМОГОЮ ТАБЛИЦІ ВІЖЕНЕРА

Мета і зміст:

1. Поглибити знання з основ багатоалфавітного шифрування.
2. Дослідити основні характеристики алгоритму шифрування Віженера.

Необхідні ресурси

ПК або мобільний пристрій з доступом до Інтернету

ОС Windows 10, 11, IDE для розробки програм (мава програмування за вибором студента)

Короткі теоретичні відомості

Найбільш простими є шифри заміни або підстановки, особливістю яких є заміна символів (слів) відкритого тексту відповідними символами, які належать алфавітом шифротекста. Розрізняють: одноалфавітну, багатоалфавітну заміну.

Прикладом багатоалфавітного шифру заміни є система Віженера. Шифрування здійснюється за таблицею, що представляє собою квадратну матрицю розмірності n × n, де n - число символів алфавіту (для російського алфавіту - 32, для

українського - 33). Перший рядок містить всі символи алфавіту. Кожний наступний рядок виходить з попередньої шляхом циклічного зсуву вправо на один символ (або вліво).

Вибирається ключ або ключова фраза. Після чого процес зашифровування здійснюється наступним чином:

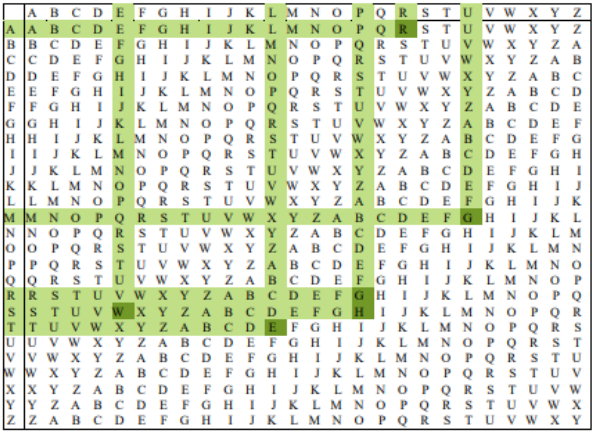
- Під кожною буквою вихідного повідомлення послідовно записуються літери ключа (якщо ключ коротше - його використовують кілька разів).
- Кожна буква шифротекста знаходиться на перетині стовпця таблиці, визначається буквою відкритого тексту, і рядки, яка визначається буквою ключа.

Приклад. Повідомлення PURPLE, зашифроване ключем SMART за допомогою таблиці Віженера, перетвориться у шифротекст HGRGEW.

Таблиця Віженера Таблиця 1.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Шифрування повідомлення за таблицею Віженера Таблиця 2.



Методика і порядок виконання роботи

- Вивчити теоретичний матеріал роботи.
- Провести дослідження системи багатоалфавітної заміни Віженера.
- Розробити програму, що шифрує і дешифрує текстове повідомлення за допомогою системи багатоалфавітної заміни Віженера. Додаток має містити графічний інтерфейс, можливість завдання ключового слова, текстового повідомлення як в інтерфейсі, так і за допомогою завантаження файлів. Максимальна кількість балів - 8.
- Завдання підвищеної складності: розробити клієнт-серверний додаток, що реалізує алгоритм системи багатоалфавітної заміни Віженера. Клієнт посилає серверу текстове повідомлення і ключове слово, алгоритм шифрування, дешифрування відбувається на стороні сервера. Максимальна кількість балів -10.

- Студент виконує одне із запропонованих завдань. Вибір інструмента реалізації програми на розсуд студента.

За результатами лабораторної роботи оформіть звіт. ,

Контрольні питання

1. Дайте визначення шифру багатоалфавітної заміни. Які шифри даного виду ви знаєте.
2. Дайте визначення шифру багатоалфавітної заміни.
3. Назвіть основні переваги та недоліки шифру багатоалфавітної заміни.

ЛАБОРАТОРНА РОБОТА №7
ПРОГРАМНА РЕАЛІЗАЦІЯ ШИФРУ DES

Мета роботи - створити криптографічний систему шифрування даних, яка базується на алгоритмі шифрування DES. Алгоритм DES є першим симетричним алгоритмом блокового шифрування даних.

Необхідні ресурси

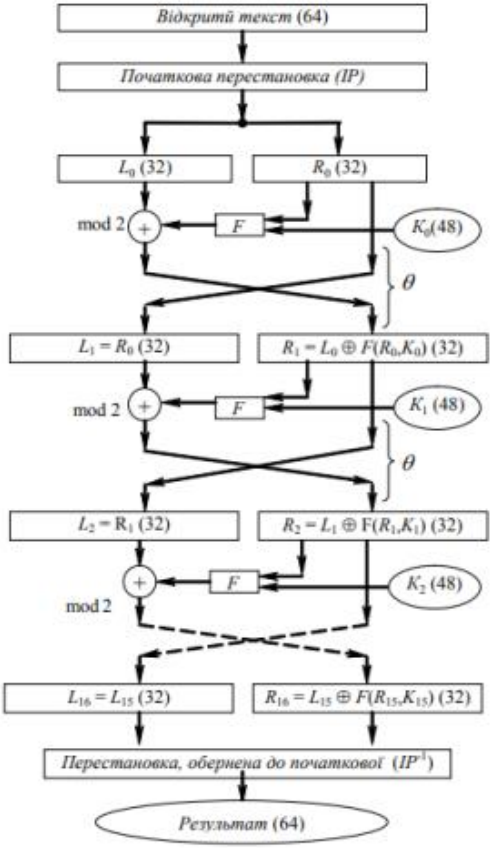
ПК або мобільний пристрій з доступом до Інтернету

ОС Windows 10, 11, IDE для розробки програм (мава програмування за вибором студента)

Короткі теоретичні відомості

DES Здійснює шифрування 64-бітових блоків даних с помощью 56-бітового ключа. Алгоритм вікорістовує только стандартних арифметику 64-бітовіх чисел и логічні операции. Розшифрування в DES є операцією зворотнього шифрування и віконується шляхом повторення операцій шифрування в зворотній послідовності

Структурна схема алгоритму DES



Завдання до роботи

Програмна реалізація криптографічного системи, заснованої на алгоритмі шифрування DES. У програмній реалізації повинен бути розроблений інтерфейс, зручний для експлуатації програми, в інтерфейсі слід передбачити:

- два режими формування ключа - ключ заданий через файл або поле введення;

- введення початкової інформації з сформованого заздалегідь файлу і з областей введення, які є в інтерфейсі програми;

- зашифрований текст може збережуться в файлі і відображатися у вікні.

- режими шифрування і дешифрування інформації.

Оцінка -8 балів

Завдання підвищеної складності

Програмна реалізація криптографічного системи, заснованої на алгоритмі шифрування DES. У програмній реалізації повинен бути розроблений інтерфейс, зручний для експлуатації програми, в інтерфейсі слід передбачити:

- режими формування ключа - ключ заданий (через файл або поле введення), ключ генерується за замовчуванням;

- введення початкової інформації з сформованого заздалегідь файлу і з областей введення, які є в інтерфейсі програми;

- режими шифрування і дешифрування інформації.

- зашифрований текст може збережуться в файлі і відображатися у вікні. передбачити

- можливість перегляду ключа в шістнадцятковому і символьному вигляді;

- програма повинна показувати час шифрування.

Оцінка -10 балів

2. Дослідити лавинний ефект (дослідження проводити на одному блоці тексту):

1) для біта, який буде змінюватися, додаток повинен дозволяти ставити його позицію (номер) у відкритому тексті або в ключі;

2) додаток повинен вміти після кожного раунду шифрування підраховувати число біт, що змінилися в зашифрованому тексті при зміні одного біта у відкритому тексті або в ключі;

3) додаток може будувати графіки залежності числа біт, що змінилися в зашифрованому тексті, від раунду шифрування, або графіки можна будувати в сторонньому ПО, але тоді додаток для шифрування має зберігати в файл необхідну для побудови графіків інформацію.

Оцінка: + 5 балів Разом - 15 балів

Зробити висновки про виконану роботу.

Підготувати звіт по роботі. У звіті описати алгоритм DES, описати структуру представлення даних в програмі, основні функції програми, призначення функцій, вхідні і вихідні параметри функцій. У звіт включити опис алгоритму генерації ключа, деталі програмної реалізації, які представляють інтерес з точки зору розробника.

За результатами лабораторної роботи оформіть звіт.

Контрольні питання

1. За яким принципом побудований блоковий шифр DES?
2. Який шифр передував шифру DES?
3. Вказати довжину початкового ключа в алгоритмі DES.
4. Перерахувати основні етапи формування ключів в алгоритмі DES.
5. Скільки раундів в алгоритмі DES?

ЛАБОРАТОРНА РОБОТА №8. ПРОГРАМНА РЕАЛІЗАЦІЯ АСИМЕТРИЧНОГО АЛГОРИТМУ RSA

Мета роботи- засвоїти основні принципи асиметричного шифрування з відкритим ключем на прикладі алгоритму RSA,

Необхідні ресурси

ПК або мобільний пристрій з доступом до Інтернету

ОС Windows 10, 11, IDE для розробки програм (мава програмування за вибором студента)

Короткі теоретичні відомості

RSA (абревіатура прізвищ Rivest, Shamir та Adleman) — «криптографічний алгоритм з відкритим ключем, що базується на обчислювальній складності задачі факторизації великих цілих чисел.

RSA став першим алгоритмом такого типу, придатним і для шифрування, і для цифрового підпису. Алгоритм застосовується до великої кількості криптографічних застосунків»[3].

Алгоритм використовує відкритий і секретний ключі. Відкритий ключ використовується для шифрування даних, закритий – для розшифрування.

Приклад

[3] https://uk.wikipedia.org/wiki/RSA

Етап	Опис операції	Результат операції
Генерація ключів	Обрати два простих різних числа	$p = 3557,$ $q = 2579$
	Обчислити добуток	$n = p \cdot q = 3557 \cdot 2579 = 9173503$
	Обчислити функцію Ейлера	$\varphi(n) = (p-1)(q-1) = 9167368$
	Обрати відкриту експоненту	$e = 3$
	Обчислити секретну експоненту	$d = e^{-1} \pmod{\varphi(n)}$ $d = 6111579$
	Опублікувати *відкритий ключ*	$\{e, n\} = \{3, 9173503\}$
	Зберегти *секретний ключ*	$\{d, n\} = \{6111579, 9173503\}$
Шифрування	Обрати текст для шифрування	$m = 111111$
	Обчислити шифротекст	$c = E(m)$ $= m^e \mod n$ $= 111111^3 \mod 9173503$ $= 4051753$
Розшифрування	Обчислити вихідне повідомлення	$m = D(c) =$ $= c^d \mod n$ $= 4051753^{6111579} \mod 9173503$ $= 111111$

Завдання до роботи

У програмній реалізації алгоритму RSA повинен бути розроблений інтерфейс, зручний для експлуатації програми, в інтерфейсі передбачити режим завдання параметрів системи за умовчанням і режим генерування параметрів системи. У програму включити найпростіший алгоритм формування простих чисел і перевірки чисел на простоту. Підготувати звіт по роботі. У звіт включити опис алгоритму RSA і алгоритм перевірки чисел на простоту.

Оцінка - 10 балів

Завдання підвищеної складності. Програмна реалізація комбінованих алгоритмів шифрування.

У програмній реалізації комбінованого алгоритму RSA + DES повинен бути розроблений інтерфейс, зручний для експлуатації програми, в інтерфейсі передбачити режим завдання параметрів системи за умовчанням і режим генерування параметрів системи. Шифрування відкритого тексту відбувається за допомогою симетричного алгоритму, асиметричний алгоритм використовується для шифрування самого симетричного ключа (див. Теоретичні відомості). У програму включити найпростіший алгоритм формування простих чисел і перевірки чисел на простоту. Підготувати звіт по роботі.

Оцінка -15 балів.

За результатами лабораторної роботи оформіть звіт.

Контрольні питання

1. Дати визначення односторонньої функції.
2. Дати визначення односторонньої функції з секретом.
3. На якій основі зазвичай створюються криптографічні системи з відкритими
4. ключами?
5. Перерахувати число параметрів в криптографічного системі RSA.
6. Перерахувати секретні параметри системи RSA.
7. Перерахувати відкриті параметри системи RSA.
8. На якому математичному результаті базується система RSA.

ЛАБОРАТОРНА РОБОТА №9. КРИПТОГРАФІЧНІ ХЕШ-ФУНКЦІЇ

Мета роботи

Вивчити основні підходи перетворення масиву даних за допомогою хешування.

Необхідні ресурси

ПК або мобільний пристрій з доступом до Інтернету

ОС Windows 10, 11, IDE для розробки програм (мава програмування за вибором студента)

Короткі теоретичні відомості

Хешування — це практика перетворення даного ключа або рядка символів в інше значення з метою безпеки. На відміну від стандартного шифрування, хешування завжди використовується для одностороннього шифрування, а хешовані значення дуже важко декодувати.

Контрольні суми – «це нескладні, вкрай швидкі і легко реалізовані апаратні алгоритми, використовувані для захисту від ненавмисних спотворень, в тому числі помилок апаратури. За швидкістю обчислення вони в десятки і сотні разів швидше, ніж криптографічні хеш-функції, і значно простіше в апаратної реалізації» [16]. Головний недолік- це доволі низька крипостійкість.

У даній лабораторній роботі розглядаються тільки криптографічні хеш-функції.

Сімейство алгоритмів SHA

Сімейство алгоритмів SHA (Secure hash standard) включає в себе 5 алгоритмів обчислення хеш-функції: SHA-1, SHA-224, SHA-256, SHA-384, SHA-512. Чотири останні хеш-функції об'єднуються в підродина SHA-2. Алгоритм SHA-1 розроблений Агентством національної безпеки США (NSA) в 1995 році. Алгоритми підродини SHA-2 також розроблені Агентством

національної безпеки США. Ці алгоритми використовуються в SSL, SSH, S / MIME, DNSSEC, X.509, PGP, IPSec, при передаі файлів по мережі (BitTorrent).

Між собою алгоритми відрізняються за крипостійкістю, яка забезпечується для даних, що хешуються, а також розмірами блоків і слів даних, що використовуються при хешування. Основні відмінності алгоритмів можна представити у вигляді таблиці 1:

Властивості алгоритмів SHA. Таблиця 1

алгоритм	довжина дайджесту повідомлення (біт)	довжина внутрішнього стану (біт)	довжина блока (біт)	довжина сообщення (біт)	довжина слова (біт)	кількість ітерацій в циклі
SHA-1	160	160	512		32	80
SHA-224	224	256	512		32	64
SHA-256	256	256	512		32	64
SHA-384	384	512	1 024		64	80
SHA-512	512	512	1 024		64	80

Завдання

Реалізувати програму з графічним інтерфейсом, що дозволяє виконувати порівняння двох числових масивів чисел за допомогою хеш-функції (алгоритм SHA-2.)

Оцінка -10

Завдання з підвищеною складністю

Реалізувати додаток -калькулятор хеш-функції (алгоритм SHA-2.) графічний інтерфейс повинен дозволяти вводити в поле або зчитувати з файлу текстове повідомлення будь довжини. Отримане значення хеш-функції має представлятися в шістнадцятковому вигляді і зберігатися в файл

Оцінка -15 балів.

За результатами лабораторної роботи оформіть звіт.

Контрольні питання

1. Що таке хеш-функція? Коли вона є криптографически стійкою? Що таке лавинний ефект?
2. Алгоритм MD5.
3. Сімейство алгоритмів SHA.

СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. Про захист інформації в інформаційно-комунікаційних системах : Закон України від 05.07.1994 р. №80/94-ВР URL: https://zakon.rada.gov.ua/laws/show/80/94-%D0%B2%D1%80#Text (дата звернення: 01.12.20204)
2. Бурячок В. Л., Толубко В.Б., Хорошко В. О., Толюпа С.В. Інформаційна та кібербезпека: соціотехнічний аспект. – К. : ДУТ, 2015. – 288 с.
3. Глинчук Л.Я. Криптологія: навч. –метод. посіб.- Луцьк: Вежа-друк, 2014. – 164 с.
4. Інформаційна безпека / Остроухов В., Присяжнюк М., Фармагей О., Чеховська М. та ін.. – К.; Ліра-К, 2021. – 412 с..
5. Кібербезпека : сучасні технології захисту. Навчальний посібник для студентів вищих навчальних закладів. / С. Е. Остапов, С. П. Євсеєв, О.Г. Король. – Львів: «Новий Світ-2000», 2020 . – 678 с.
6. Козіна Г.Л. Криптографія від історії до сучасних стандартів: навч.посібник. – Запоріжжя : НУ «Запорізька політехніка», 2020. – 192 с.
7. Лабораторна робота - Аутентифікація, авторизація та облік. *Cisco Networking Academy* URL: http://surl.li/rmomyl (дата звернення: 01.12.20204)
8. Лабораторна робота - Ідентифікація загроз. *Cisco Networking* Academy URL: http://surl.li/htmfim (дата звернення: 01.12.20204)
9. Методологія захисту інформації. Аспекти кібербезпеки: підручник/ Г.М. Гулак – К.: Видавництво НА СБ України, 2020. – 256 с.
10. Прикладна криптологія. Методичні рекомендації до виконання лабораторних робіт. – Житомир: Державний університет «Житомирська політехніка», 2021. – 104 с.

11. Операційні системи : навчальний / О. В. Задерейко, С. Л. Зіноватна, А. А. Толокнов. – Одеса : Фенікс, 2022. – 140 с.
12. Остапов С.Е. Валь Л.О. Основи криптографії: навчальний посібник. – Чернівці: Книга– XXI, 2008. – 188 с.
13. Тарнавський Ю.А. Технології захисту інформації. – К.: КПІ ім.. Ігоря Сікорського, 2018. – 162 с.
14. Яковенко Е., Журавель І., Горбати І., Бондарев А. Інформаційна безпека. – Львів : Львівська політехніка, 2019. – 580 с.
15. Piyush Verma Wireshark Network Security – Birmingham: PACKT publishing, 2015. – 138 p.
16. Wiki ТНТУ. *Тернопільський національний технічний університет ім. Івана Пулюя*. URL: https://wiki.tntu.edu.ua/Хешування (дата звернення: 01.12.20204)
17. William Stalling Effective cybersecurity Understanding and using standards and best practices – NY: Addison-Wesley, 2019. – 768.