ОПЕРАЦІЙНА СИСТЕМА LINUX

ЛІТНЯ ШКОЛА З КІБЕРБЕЗПЕКИ

ПРАКТИЧНЕ ЗАНЯТТЯ №1

Завдання №1. Імпорт ВМ Kali Linux в середовище VirtualBox.

Завдання №2. Хто ви в системі?

Ваша мета: Отримати базову інформацію про поточного користувача та операційну систему.

Команди:

whoami – ім'я поточного користувача

id – ідентифікатор користувача, групи та членство в групах

uname -a – виводить інформацію про операційну систему та ядро Linux

hostname – виводить ім'я поточного комп'ютера в мережі

pwd – виводить повний шлях до поточного каталогу.

Завдання:

1. Визначити ім'я поточного користувача.
2. Визначити ім'я комп'ютера.
3. Визначити поточний робочий каталог.
4. Визначити версію ядра Linux.

Завдання №3. Довідкова система Linux.

Ваша мета: Навчитися отримувати довідкову інформацію про команди в Linux.

Команди для виконання завдання:

man <команда>

<команда> --help

Приклад:

man ls

ls --help

Завдання:

1. Відкрити довідку для команд:

- ls

- cp

- rm

2. Знайти в довідці:

- Що означає ключ -r

- Що означає ключ -a

Питання для обговорення:

1. Чим відрізняються man та --help?
2. У яких випадках зручніше використовувати --help?
3. Як вийти з режиму перегляду man?

Завдання №4. Дослідження файлової системи Linux.

Ваша мета: Ознайомитися зі структурою файлової системи Linux.

Команди:

cd / - переміщення каталогами

ls – вивід вмісту поточного каталогу

ls -la – вивід детальної інформації про файли, включаючи приховані файли

Відвідати каталоги:

/home

/etc

/var

/usr

/tmp

Додатково:

1. Перейти у домашній каталог поточного користувача.
2. Відобразити приховані файли у домашньому каталозі.

Питання для обговорення:

1. Який каталог містить домашні каталоги користувачів?
2. Для чого використовується каталог /etc?
3. Який каталог призначений для тимчасових файлів?
4. Чому файл .bashrc не відображається при звичайному ls?

Завдання №5. Створення власного робочого простору.

Ваша мета: Створити у домашньому каталозі робочу структуру для Літньої школи з кібербезпеки.

Команди для виконання завдання:

mkdir – створення каталогу

Приклад:

mkdir OS_Linux

touch – створення порожнього файлу

Приклад:

touch notes.txt

Для перегляду структури каталогів у зручному вигляді використовується команда:

tree

Встановлення інструмента tree:

sudo apt install tree

Структура:

Створити директорію:

SummerCyberSchool

Та всередині неї наступні каталоги:

- OS_Linux

- Networking

- Crypto

- Stego

- CTF

У кожному каталозі створити відповідні файли:

OS_Linux:

- notes.txt

- commands.txt

Networking:

- notes.txt

- topology.txt

Crypto:

- tasks.txt

- answers.txt

Stego:

- challenge1.txt

- challenge2.txt

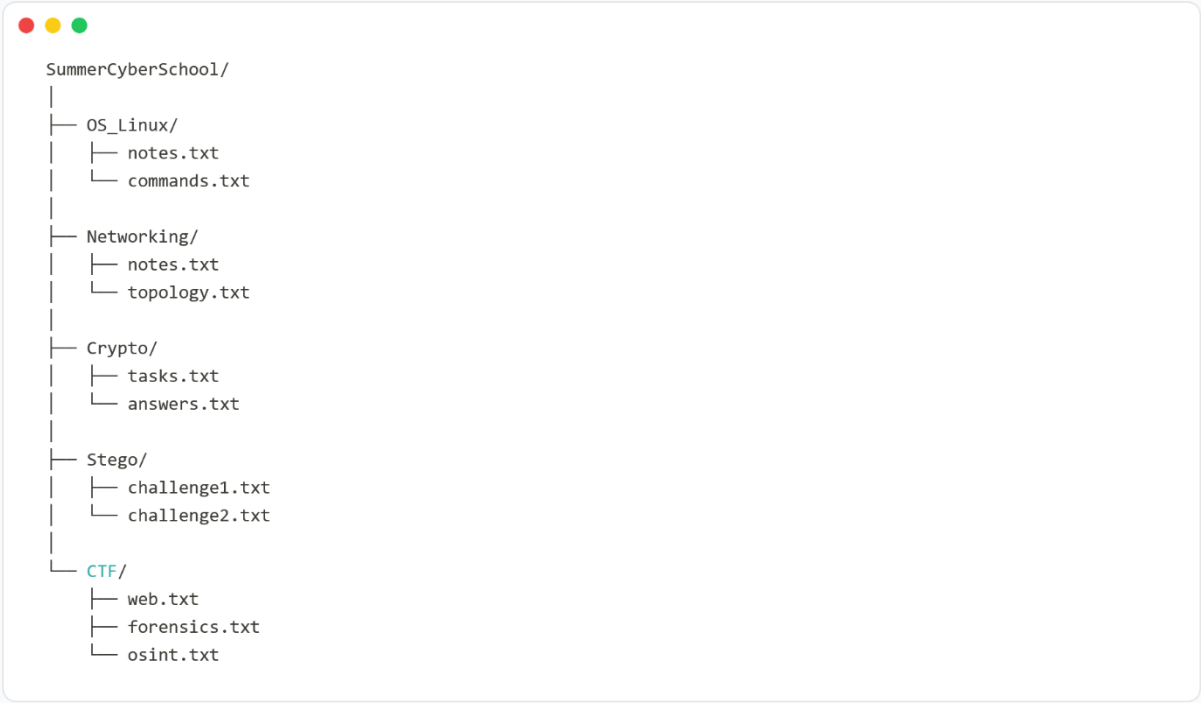
CTF:

- web.txt

- forensics.txt

- osint.txt

Структура робочого простору (візуально):



Завдання №6. Копіювання файлів і каталогів.

Ваша мета: Ознайомитися з операцією копіювання файлів і каталогів у Linux та виконати базові операції копіювання в структурі SummerCyberSchool.

Команди для виконання завдання:

cp – копіювання файлів

Приклад:

cp notes.txt notes_nackup.txt – створить копію файлу notes.txt з назвою notes_backup.txt

cp -r – копіювання каталогів рекурсивно

Приклад:

cp -r Crypto CryptoCopy – створить повну копію каталогу Crypto разом з усіма вкладеними файлами та підкаталогами

Завдання:

1. Створити резервну копію каталогу Crypto всередині каталогу CTF.

2. Скопіювати файл notes.txt з каталогу OS_Linux у каталог Networking.

3. Створити копію файлу CTF/web.txt у тому ж каталозі з новою назвою web_backup.txt.

Питання для обговорення:

1. Чим відрізняється копіювання файлу від копіювання каталогу?
2. Для чого використовується ключ -r у команді cp?
3. Чи змінюється оригінальний файл після копіювання?

Завдання №7. Переміщення/перейменування файлів і каталогів.

Ваша мета: Навчитися переміщати файли та каталоги між різними частинами структури.

Команди для виконання завдання:

mv – переміщення та перейменування

Приклад переміщення файлу:

mv notes.txt OS_Linux/ - перемістить файл notest.txt до каталогу OS_Linux

Приклад перейменування файлу:

mv notes.txt notes_backup.txt – перейменує файл notes.txt у notes_backup.txt

Приклад перейменування каталогу:

mv Test_folder Test_folder_Backup – перейменування каталогу

Завдання:

1. Перемістити файл challenge1.txt з каталогу Stego у каталог CTF.

2. Перемістити файл answers.txt з каталогу Crypto у каталог OS_Linux.

3. Перейменувати резервну копію каталогу Crypto (Завдання №6) у Crypto_Backup.

4. Перемістити перейменовану резервну копію каталогу Crypto_Backup у каталог Crypto.

Питання для обговорення:

1. Чим відрізняється операція переміщення від копіювання?

2. Що відбувається з оригінальним файлом після переміщення?

Завдання №8. Видалення файлів і каталогів.

Ваша мета: Навчитися безпечно видаляти файли та каталоги у Linux.

Команди для виконання завдання:

rm – видалення файлів

Приклад:

rm notes.txt – видалить файл notes.txt

rm -r – видалення каталогів

Приклад:

rm -r Test_folder – видалить каталог та всі файли й підкаталоги всередині нього

Завдання:

1. Видалити файл challenge2.txt з каталогу Stego.
2. Видалити файл osint.txt з каталогу CTF.
3. Видалити каталог Crypto_Backup з каталогу Crypto.

Питання для обговорення:

1. Чим небезпечна команда rm -r?
2. Чому важливо перевіряти шлях перед видалення файлів?

Завдання №9. Наповнення файлів у робочому просторі.

Ваша мета: Ознайомитися з базовими способами запису, перезапису та доповнення файлів у Linux та заповнити робочий простір SummerCyberSchool інформацією.

Команди для виконання завдання:

echo – запис тексту у файл.

> - перезапис файлу

>> - додавання тексту у файл

nano – текстовий редактор

Базові операції запису у файл

1. Запис (створення або перезапис):

> - створює файл або повністю перезаписує його вміст.

Приклад: *echo “some_text” > notes.txt*

2. Додавання даних:

>> - додає новий рядок у кінець файлу без видалення попереднього вмісту.

Приклад: *echo “some_more_text” >> notes.txt*

3. Альтернативний спосіб – nano (редагування файлу в інтерактивному режимі).

Корисні комбінації:

- CTRL + O – зберегти файл
- CTRL + X – вийти
- CTRL + K – вирізати рядок
- CTRL + U – вставити рядок
- CTRL + W – пошук

Завдання:

1. У файл OS_Linux/notes.txt командою “echo” записати коротке визначення Linux.

2. Командою “echo” додати у файл OS_Linux/notes.txt власне прізвище та ім’я, не перезаписуючи поточний вміст файлу.

3. У файл CTF/web.txt за допомогою текстового редактора nano додати будь-який текст та зберегти зміни.

4. За допомогою текстового редактора nano змінити вміст файлу OS_Linux/notes.txt, додавши номер групи поруч із прізвищем та ім’ям та зберегти зміни.

5. У будь-який зручний спосіб додайте до файлу CTF/forensics.txt наступний текст:

CTF FORENSICS OVERVIEW

Introduction

CTF forensics challenges simulate digital investigations where participants analyze collected evidence rather than exploit live systems. The primary objective is to reconstruct events that occurred on a target system, identify malicious activity, recover hidden information, and locate challenge flags. These tasks are designed to develop

analytical thinking, evidence handling skills, and familiarity with common forensic artifacts.

Investigators often receive disk images, memory dumps, packet captures, log files, application data, browser histories, or partially corrupted archives. Each artifact may contain only a small portion of the information required to solve the challenge. Success depends on correlating multiple evidence sources and building a coherent understanding of the incident.

Disk Analysis

Disk analysis focuses on examining storage media and file systems. Analysts recover deleted files, inspect metadata, identify hidden content, and reconstruct user activity timelines.

Common forensic artifacts include documents, browser downloads, registry data, system logs, and temporary files. File carving techniques are often used to recover data that no longer appears in the file system structure.

Memory Analysis

Memory forensics involves analyzing RAM captures obtained from running systems. Volatile memory frequently contains valuable evidence that never reaches disk.

Analysts search for running processes, network connections, command history, loaded modules, encryption keys, and malware artifacts that may disappear after a reboot.

Network Analysis

Network forensics examines packet captures and communication logs. Investigators reconstruct sessions, identify suspicious traffic, extract transferred files, and detect command-and-control activity.

Packet analysis often reveals attacker behavior, compromised hosts, credential transmission, and indicators of compromise.

Steganography

Steganography challenges focus on hiding information within files such as images, audio recordings, videos, or documents.

Analysts inspect metadata, embedded content, unusual file structures, and encoded messages to recover hidden information.

Incident Response

Incident response combines evidence from multiple sources to understand how an attack occurred. Analysts build timelines, identify affected systems, determine attacker actions, and assess impact.

The investigation process requires careful documentation and validation of findings to ensure conclusions are supported by available evidence.

Conclusion

CTF forensics develops practical investigation skills that closely resemble real-world digital forensic and incident response activities. Success depends on patience, attention to detail, and the ability to correlate evidence from multiple sources.

Питання для обговорення:

1. У чому різниця між операторами > та >>?

2. Що відбувається з вмістом файлу при використанні >?

Завдання №10. Перегляд та аналізу вмісту файлів у робочому просторі.

Ваша мета: Ознайомитися з базовими командами перегляду файлів у Linux та навчитися отримувати, аналізувати та швидко знаходити інформацію у файлах робочого простору SummerCyberSchool.

Команди для виконання завдання:

cat <назва_файлу> – повний вивід вмісту файлу

less <назва_файлу> – посторінковий перегляд файлу

head <назва_файлу> – перегляд початку файлу

tail <назва_файлу> – перегляд кінця файлу

head -n 5 filename.txt – вивід перших N рядків

tail -n 5 filename.txt – вивід останніх N рядків

Контекст (CTF-підготовка):

У реальних CTF-завданнях та цифрових розслідуваннях аналітики працюють із великими текстовими файлами (логи, звіти, дампи даних). Завдання полягає не просто у відкритті файлу, а у швидкому знаходженні потрібних фрагментів інформації.

Завдання:

1. Вивести повний вміст файлу: OS_Linux/notes.txt.

2. Переглянути файл у посторінковому режимі CTF/web.txt.

3. Переглянути файл CTF/forensics.txt та знайти перший абзац, який описує: disk analysis.

4. Вивести перші 5 рядків файлу: CTF/forensics.txt

5. Вивести останні 5 рядків файлу: CTF/forensics.txt

6. Порівняти поведінку команд: cat vs less на файлі CTF/forensics.txt

Питання для обговорення:

1. У чому різниця між cat, less, head і tail?

2. Коли використання повного виводу файлу (cat) є неефективним?

Завдання №11. Пошук інформації у файлах (grep)

Ваша мета: Ознайомитися з командою grep та навчитися знаходити потрібну інформацію всередині текстових файлів.

Команда для виконання завдання:

grep – пошук рядків, що відповідають заданому шаблону

Контекст (CTF-підготовка):

У CTF-змаганнях та цифрових розслідуваннях аналітики часто працюють із великими логами, конфігураційними файлами та дампами даних. Команда grep дозволяє швидко знаходити потрібні слова, фрази або індикатори компрометації без необхідності переглядати весь файл вручну.

Приклад використання команди grep:

1. Пошук слова у файлі:

grep Linux notes.txt

2. Пошук без врахування регістру:

grep -i linux notes.txt

3. Вивід номерів рядків:

grep -n Linux notes.txt

4. Пошук у всіх текстових файлах каталогу:

*grep Linux *.txt*

5. Пошук декількох входжень у каталозі та підкаталогах:

grep -r Linux .

Завдання:

1. Знайти слово forensics у файлі CTF/forensics.txt.

2. Вивести номери рядків, у яких зустрічається слово CTF у файлі CTF/forensics.txt.

3. Виконати пошук власного прізвища у файлі OS_Linux/notes.txt.

Завдання №12. Пошук файлів у файловій системі (find та locate).

Ваша мета: Ознайомитися з командами пошуку файлів у Linux та навчитися знаходити потрібні файли у структурі SummerCyberSchool за назвою та шляхом.

Команди для виконання завдання:

find – пошук файлів у файловій системі в реальному часі.

locate – швидкий пошук файлів по базі даних системи.

Контекст (CTF-підготовка):

У CTF та цифровій криміналістиці аналітики часто працюють з великою кількістю файлів і директорій. Важливо вміти швидко знаходити потрібні артефакти: конфігурації, логи, скрипти або приховані файли.

Приклади використання команди find:

1. Пошук за назвою файлу:

```
find . -name filename.txt
```

2. Пошук без врахування регістру:

```
find . -iname filename.txt
```

3. Пошук тільки файлів:

```
find . -type f -name filename.txt
```

4. Пошук тільки директорій:

```
find . -type d -name foldername
```

5. Пошук за розширенням (наприклад, .txt):

```
find . -name "*.txt"
```

6. Пошук файлів більших за певний розмір:

```
find . -size +1M
```

7. Пошук файлів, змінених за останні 24 години:

```
find . -mtime -1
```

Приклади використання команди locate:

1. Базовий пошук файлу за назвою:

```
locate filename.txt
```

2. Пошук частини назви файлу:

```
locate notes
```

3. Обмеження кількості результатів (через head):

```
locate filename.txt | head -n 10
```

Завдання:

1. Знайти файл notes.txt у структурі SummerCyberSchool за допомогою find.

2. Знайти всі файли з розширенням .txt у каталозі SummerCyberSchool.

3. Оновити базу locate командою:

```
sudo updatedb
```

4. Використовуючи locate, виконати пошук будь-якого файлу зі структури.

Завдання №13. Пошук виконуваних програм (which).

Ваша мета: Ознайомитися з командою which та навчитися визначати місце розташування виконуваних файлів програм.

Команда для виконання завдання:

which <команда>

Приклад:

which bash

Завдання:

1. Знайти розташування команд:
   - bash
   - nano
   - grep
   - find
2. Знайти розташування команди tree.
3. Порівняти результати для:
   - which bash
   - which ls

Завдання №14. Пайплайни.

Ваша мета: Ознайомитися з механізмом передавання результатів між командами та логічним виконанням команд у Bash.

Команди для виконання завдання:

| - пайплайн

grep – пошук рядків за шаблоном

Контекст (CTF-підготовка):

Під час аналізу логів, мережевого трафіку та великих наборів даних аналітики часто об'єднують декілька команд у єдиний ланцюжок для швидкого пошуку інформації.

Приклади використання:

cat notes.txt | grep Linux – вивести вміст файлу та знайти рядки, що містять слово Linux.

ls | grep txt – вивести лише файли, назви яких містять txt

find . -name “.txt” | grep notes – знайти всі текстові файли та відобразити лише ті, що містять у назві notes.*

Завдання:

1. Вивести вміст файлу CTF/forensics.txt та знайти у ньому слово:

forensics

2. Вивести список усіх текстових файлів у структурі SummerCyberSchool та відобразити лише файли, у назві яких зустрічається:

notes

3. Знайти всі файли з розширенням .txt та відобразити лише ті, що знаходяться в каталозі:

CTF

ПРАКТИЧНЕ ЗАНЯТТЯ №2

Завдання №1. Аналіз процесів у Linux.

Ваша мета: Ознайомитися з процесами, які працюють у системі та навчитися отримувати інформацію про них.

Контекст (CTF-підготовка):

Під час CTF-змагань та аналізу компрометованих систем аналітик повинен швидко визначити, які процеси запущені в системі, які з них споживають ресурси та чи немає серед них підозрілих програм.

Команди для виконання завдання:

ps – перегляд процесів поточного користувача

ps aux – перегляд усіх процесів у системі

top – моніторинг процесів у реальному часу

htop – розширений моніторинг процесів

Завдання:

1. Переглянути список власних процесів.
2. Переглянути список усіх процесів системи.
3. Знайти процес із найбільшим використанням процесора.
4. Відкрити top або htop та дослідити інформацію про процеси.

Питання для обговорення:

1. Що таке PID процесу?
2. Яка різниця між ps та top?
3. Чому один користувач може бачити процеси інших користувачів?

Завдання №2. Завершення процесів.

Ваша мета: Навчитися завершувати процеси різними способами.

Контекст (CTF-підготовка):

Під час реагування на інцидент аналітик може виявити шкідливий процес, який необхідно негайно зупинити до початку детального аналізу.

Команди для виконання завдання:

kill – завершення процесу за PID

killall – завершення процесів за назвою

Приклади використання:

kill 1234 – завершення процесу за PID 1234

killall firefox – завершення процесів за назвою firefox

Завдання:

1. Запустити тестовий процес:

sleep 300

2. Знайти його PID.
3. Завершити процес через kill.
4. Повторно запустити кілька процесів sleep.
5. Завершити їх командою killall.

Питання для обговорення:

1. Чим відрізняється kill та killall?
2. Навіщо потрібен PID?
3. Чому не всі процеси можна завершити звичайним користувачем?

Завдання №3. Фонові процеси.

Ваша мета: Навчитися запускати та контролювати процеси у фоновому режимі.

Контекст (CTF-підготовка):

Під час аналізу великих логів або запуску скриптів аналітик часто виконує довготривалі задачі у фоні, продовжуючи працювати в терміналі.

Команди для виконання завдання:

& - запуск процесу у фоні

jobs – перегляд фонових задач

bg – продовження процесу у фоні

fg – повернення процесу на передній план

Приклад використання:

sleep 300 &

jobs

fg %1

Завдання:

1. Запустити процес sleep у фоновому режимі.
2. Переглянути список фонових задач.

3. Повернути процес на передній план.
4. Завершити (вибити) процес за допомогою Ctrl+C.

Питання для обговорення:

1. Для чого використовуються фонові процеси?
2. Що відбувається після закриття терміналу?
3. У яких випадках аналітику корисно використовувати фоновий режим?

Завдання №4. Робота зі службами Linux.

Ваша мета: Ознайомитися з системними службами Linux.

Контекст (CTF-підготовка):

Під час аналізу інцидентів часто необхідно визначити, які служби працюють на сервері та які з них можуть бути використані зловмисником.

Команди для виконання завдання:

```
systemctl status <назва служби>
systemctl start <назва служби>
systemctl stop <назва служби>
systemctl restart <назва служби>
```

Приклад використання:

```
systemctl status ssh
sudo systemctl restart ssh
```

Завдання:

1. Перевірити стан служби SSH.
2. Зупинити службу SSH та перевірити зміни.
3. Дослідити статус планувальника задач (cron).
4. Перезапустити службу SSH.

Питання для обговорення:

1. Що таке служба (service)?
2. Чому важливо контролювати запущені служби?

Завдання №5. Встановлення програмного забезпечення.

Ваша мета: Навчитися встановлювати програмне забезпечення з репозиторіїв Linux.

Контекст (CTF-підготовка):

Під час участі у CTF-змаганнях або роботи фахівця з кібербезпеки часто виникає необхідність швидко встановлювати нові інструменти для аналізу мережі, файлів або системи.

Команди для виконання завдання:

apt update – оновлення списку пакетів, доступних для встановлення

apt install <назва_пакета> – встановлення пакета

apt remove <назва_пакета> - видалення пакета

Приклад використання:

sudo apt update

sudo apt install fastfetch

fastfetch

sudo apt remove fastfetch

Примітка:

Для коректної роботи apt install на ВМ Kali Ethical Hacker необхідно виконати наступні команди:

sudo nano /etc/resolv.conf

додати рядок в кінець файлу:

nameserver 8.8.8.8

Зберегти файл

В терміналі Kali:

sudo wget -O /usr/share/keyrings/kali-archive-keyring.gpg https://archive.kali.org/archive-keyring.gpg

Завдання:

1. Оновити список пакетів системи.
2. Встановити утиліту fastfetch.
3. Запустити утиліту та переглянути інформацію про систему.
4. Визначити:

   - Версію операційної системи.
   - Версію ядра Linux.
   - Обсяг оперативної пам’яті.
   - Модель процесора.

5. Видалити утиліту з системи.

Питання для обговорення:

1. Звідки Linux отримує інформацію про доступні пакети?

2. Чим відрізняється apt update та apt install?

3. Чому встановлення програм із офіційних репозиторіїв вважається безпечнішим?

Завдання №6. Робота з архівами.

Ваша мета: Навчитися створювати та розпаковувати архіви.

Контекст (CTF-підготовка):

У CTF-завданнях та цифровій криміналістиці докази часто передаються у вигляді архівів. Аналітик повинен уміти швидко їх розпаковувати та створювати власні архіви для передачі результатів роботи.

Команди для виконання завдання:

zip

unzip

tar

Приклади використання:

zip -r school.zip SummerCyberSchool – створення .zip архіву, де:

-r (recursive) – рекурсивно додати всі вкладені каталоги та файли.

unzip school.zip – розпакування архіву

tar -cvf school.tar SummerCyberSchool – створення .tar apіxy, де:

-c (create) – створити архів.

-v (verbose) – показувати файли під час архівації.

-f (file) – вказати файл архіву.

school.tar – назва архіву.

SummerCyberSchool – каталог для архівації.

tar -xvf school.tar - розпакування архіву, де:

-x (extract) – витягнути файли з архіву.

-v (verbose) – вказати архів для розпакування.

school.tar – архів, який потрібно розпакувати.

У Linux часто використовується формат ***.tar.gz***, де архів додатково стискається алгоритмом Gzip.

tar -czvf school.tar.gz SummerCyberSchool

tar -xzvf school.tar.gz - розпакування архіву, де:

-z – використання стиснення Gzip.

Завдання:

1. Створити ZIP-архів каталогу SummerCyberSchool.
2. Видалити оригінальний каталог.
3. Відновити каталог із архіву.
4. Створити TAR-архів цього ж каталогу.
5. Повторно видалити каталог SummerCyberSchool.
6. Розпакувати TAR-архів.

ПРАКТИЧНЕ ЗАНЯТТЯ №3

Завдання №1. Аналіз прав доступу до файлів.

Ваша мета: Навчитися переглядати права доступу до файлів та каталогів.

Контекст (CTF-підготовка): Під час аудиту безпеки Linux-систем аналітики часто перевіряють права доступу до файлів. Неправильно налаштовані права можуть призвести до витоку даних або підвищення привілеїв.

Команди для виконання завдання:

ls -l

stat

Приклади використання:

ls -l

ls -l SummerCyberSchool

stat notes.txt

Завдання:

1. Переглянути права доступу до файлу OS_Linux/notes.txt.
2. Переглянути права доступу до каталогу SummerCyberSchool.
3. Визначити власника файлу OS_Linux/notes.txt.
4. Визначити групу файлу OS_Linux/notes.txt.

Питання для обговорення:

1. Що означають символи rwx?
2. Хто такі user (u), group (g) та others (o)?
3. Чому каталоги також мають права доступу?

Завдання №2. Зміна прав доступу.

Ваша мета: Навчитися змінювати права доступу до файлів.

Команди для виконання завдання:

chmod <права> <файл>

або

chmod <опція> <файл>

де:

1. r (read) – читання

2. w (write) – запис

3. x (execute) – виконання

4. u (user) – власник файлу

5. g (group) – група

6. o (others) – інші користувача

Приклади використання:

chmod 644 notes.txt – Власник: читання та запис; група та інші користувачі: лише читання

chmod 600 notes.txt – Доступ лише для власника файлу

chmod 755 script.sh – Власник може читати, записувати та виконувати файл; інші користувачі можуть читати та виконувати.

chmod +x script.sh – додати право на виконання файлу

chmod -x script.sh – забрати право на виконання файлу

chmod go-r secret.txt – забрати право на читання для групи та інших користувачів.

Завдання:

1. Створити файл CTF/notes.txt

2. Переглянути поточні права доступу до файлу.

3. Змінити права до файлу так, щоб:

- Власник міг читати та записувати файл.

- Група могла лише читати файл.

- Інші користувачі не мали жодного доступу.

4. Створити файл script.sh та за допомогою текстового редактора nano додати в нього наступний вміст:

```
#!/bin/bash
echo "Hello World!"
```

5. Надати файлу script.sh право на виконання.

6. Запустити створений скрипт командою *./script.sh* та перевірити результат його виконання.

Завдання №3. Створення користувачів та груп у Linux.

Ваша мета: Ознайомитися з механізмом керування обліковими записами у Linux та створити нового користувача для подальших практичних робіт.

Команди для виконання завдання:

adduser – створення користувача

passwd – зміна пароля користувача

id – перегляд відомостей про користувача

su – зміна користувача

groupadd – створення групи

usermod – додавання користувача до групи

Приклади використання:

sudo adduser analyst – створення користувача analyst

id analyst – перегляд UID, GID та груп користувача

sudo groupadd ctf – створення групи ctf

sudo usermod -aG ctf analyst – додавання користувача analyst до групи ctf

su analyst – перехід до облікового запису користувача

exit – завершення сеансу користувача

Завдання:

1. Створити нового користувача:

analyst

2. Встановити пароль:

thunder515253ice

3. Перевірити інформацію про створеного користувача.

4. Створити групу:

ctf_team

5. Додати користувача *analyst* до групи *ctf_team*.

6. Переконайтеся, що користувача успішно створено, шляхом виводу вмісту файлу /etc/passwd.

7. Виконати вхід під користувачем analyst.

8. Визначити домашній каталог користувача analyst.

9. Створити у домашньому каталозі користувача файл:

investigation.txt

10. Переглянути поточні права на файл investigation.txt.

11. Вийти з облікового запису analyst та повернутися до користувача kali.

12. Перевірити, чи належить користувач analyst до групи ctf_team.

Завдання №4. Зміна власника та групи файлу.

Ваша мета: Навчитися змінювати власника та групу файлів у Linux.

Контекст (CTF-підготовка):

Під час аналізу скомпрометованої системи важливо визначати, кому належать файли. Зловмисники часто створюють власні файли або змінюють їх власника для приховування активності.

Команди для виконання завдання:

chown – зміна власника файлу

chgrp – зміна групи файлу

Приклади використання:

sudo chown analyst investigation.txt – змінює власника файлу investigation.txt на користувача analyst.

sudo chown analyst:ctf_team investigation.txt – одночасно змінює власника файлу на користувача analyst та групу файлу на ctf_team.

sudo chgrp ctf_team investigation.txt – змінює лише групу файлу на ctf_team, не змінюючи його власника.

Завдання:

1. Створити файл *evidence.txt.*
2. Переглянути його власника та групу.
3. Змінити власника файлу на *analyst*.
4. Змінити групу файлу на *ctf_team*.
5. Переконайтеся, що зміни застосувалися.

Завдання №5. Аналіз логів Linux.

Ваша мета: Ознайомитися з журналами подій Linux.

Контекст (CTF-підготовка):

У цифровій криміналістиці журнали подій є одним з головних джерел інформації під час розслідування інцидентів.

Команди для виконання завдання:

journalctl

cat

less

Примітка:

У більшості Linux-систем журнали подій (логи) зберігаються в каталозі:

/var/log

Приклади використання:

journalctl – перегляд усіх доступних записів системного журналу.

journalctl -n 20 – вивід останніх 20 записів журналу.

journalctl -xe – перегляд останніх подій із детальною інформацією про помилки та служби.

ls /var/log – перегляд файлів журналів у каталозі логів.

less /var/log/syslog – посторінковий перегляд системного журналу (якщо файл присутній у дистрибутиві).

Завдання:

1. Переглянути останні 20 записів системного журналу.
2. Знайти події, пов’язані з входом користувачів у систему.
3. Визначити час останнього запуску системи.
4. Переглянути вміст каталогу /var/log.

Завдання №6. Пошук SUID-файлів.

Ваша мета: Ознайомитися зі спеціальними бітами доступу Linux.

Контекст (CTF-підготовка):

У багатьох завданнях формату CTF, в яких передбачено підвищення прав доступу (Privilege Escalation), саме неправильно налаштовані SUID-файли дозволяють отримати підвищені привілеї.

Команди для виконання завдання:

find

ls -l

Приклади використання:

find / -perm -4000 2>/dev/null, де:

/ - почати пошук з кореневого каталогу (по всій файловій системі).

-perm -4000 – знайти файли з установленим SUID-бітом.

2>/dev/null -приховати повідомлення про помилки:

ls -l /usr/bin/passwd, де:

-l (long format) – детальна інформація:

- права доступу.

- кількість посилань.

- власник.

- група.

- розмір файлу.

- дата зміни.

- ім'я файлу.

Завдання:

1. Знайти всі SUID-файли у системі.
2. Знайти серед результатів файл passwd.
3. Переглянути його права доступу.
4. Визначити, який символ вказує на наявність SUID-біта.

Завдання №7. Віддалений доступ до Linux через SSH.

Ваша мета: Ознайомитися з технологією SSH та виконати підключення до Linux-системи через віддалений термінал.

Контекст (CTF-підготовка):

Під час CTF-змагань, пентестів та адміністрування систем фахівці часто працюють із віддаленими Linux-серверами через SSH. Цей протокол дозволяє безпечно керувати системою через мережу без фізичного доступу до комп'ютера.

Команди для виконання завдання:

ip a – перегляд мережевих інтерфейсів та IP- адрес.

systemctl – керування службами Linux.

ssh username@ip-address – підключення до віддаленої системи через SSH із зазначенням імені користувача та IP-адреси цільового хоста.

Приклади використання:

ip a – перегляд IP-адрес комп'ютера

sudo systemctl status ssh – перевірка стану SSH-сервера

sudo systemctl start ssh – запуск SSH-сервера.

sudo systemctl enable ssh – автоматичний запуск SSH після перезавантаження системи.

ssh analyst@192.168.1.100 – підключення до віддаленої системи під користувачем analyst.

Завдання:

1. Визначити IP-адресу своєї Kali Linux.
2. Перевірити, чи запущено SSH-сервер.
3. Якщо SSH не запущений – запустити його.
4. Переконатися, що служба SSH працює.
5. Виконати SSH-підключення до власної машини під користувачем:

analyst

6. Після успішного входу визначити:
   - Поточного користувача.
   - Ім'я комп'ютера.
   - Поточний каталог.
7. Вийти із SSH-сесії.

Питання для обговорення:

1. Для чого використовується протокол SSH?
2. Які переваги SSH для адміністрування Linux-системи?