

## **CYBERSECURITY PROJECT**

**PROJECT TITLE:** INCIDENT RESPONSE AND AUTOMATION

**VERSION:** Commercial/User Edition

**DATE CREATED:** 02-01-2025

### **Table of content:**

<b>S.NO</b>	<b>Name Of The Topic</b>	<b>Page Number</b>
1.	Project Abstract	02-03
2.	Project Scope <ul style="list-style-type: none"><li>• Objectives</li><li>• Requirements</li><li>• Environmental Setup</li></ul>	03-07
3.	Project Problem	07-08
4.	Project Solution	09
5.	Implementation <ul style="list-style-type: none"><li>• Installation of Virtualbox</li><li>• Monitoring Logs</li><li>• Installation of Splunk</li><li>• Installation of Owasp Juice-Shop</li><li>• Installation of zaproxy</li></ul>	10-28
6.	Conclusion	29-30

## Project Abstract

This project serves as an educational platform for individuals to gain practical experience in web application security and incident response. By leveraging the OWASP Juice Shop—a deliberately vulnerable web application—participants will engage in a series of exercises to simulate real-world scenarios in cybersecurity.

The project's main objectives include setting up the environment, deploying the application, monitoring security events, simulating attacks, detecting incidents, and developing a structured incident response process.

### Objectives

- Setting Up a Secure Environment
  - Create a secure and isolated environment using VirtualBox. This ensures that all activities, including vulnerability exploitation and security testing, are contained within a sandboxed setup, preventing any unintended impact on other systems.
  - Utilize Kali Linux 2024.3, a security-focused operating system, to serve as the testing platform. Kali Linux comes preloaded with an array of penetration testing tools that are invaluable for this project.
- Deploying the Application
  - Install and configure the OWASP Juice Shop web application within the environment. This application is designed to demonstrate common security vulnerabilities, providing a safe space for hands-on experimentation.
  - Configure the application to log security events and interactions for analysis.
- Monitoring for Security Events
  - Set up the ELK Stack (Elasticsearch, Logstash, Kibana) to collect, process, and visualize logs generated by the Juice Shop application.
    - Elasticsearch stores the logs for quick retrieval and analysis.
    - Logstash processes incoming logs, standardizes formats, and enriches data for actionable insights.
    - Kibana offers a graphical interface to visualize trends, anomalies, and potential security events.

- Simulating Attacks and Detecting Incidents
  - Use tools such as Nmap and OWASP ZAP to simulate network scanning, vulnerability discovery, and exploitation of the Juice Shop application.
    - Nmap identifies open ports, running services, and potential entry points in the application.
    - OWASP ZAP performs web application vulnerability scans, detecting issues such as SQL injection, XSS (Cross-Site Scripting), and insecure configurations.
  - Enable ModSecurity as a Web Application Firewall (WAF) to detect and log malicious activity. ModSecurity helps mitigate attacks by blocking known patterns and providing detailed logs.
- Developing an Incident Response Process
  - Analyze logs and alerts generated during simulated attacks using Kibana dashboards.
  - Document steps to identify, contain, and mitigate the detected incidents.
  - Develop a comprehensive incident response process, including:
    - Initial detection and validation.
    - Containment of the attack.
    - Eradication of the threat.
    - Recovery of the application.
    - Post-incident review to enhance future defenses.

## Project Scope

### 1. Objectives

The primary goal of this project is to provide hands-on experience in key areas of cybersecurity, particularly in **incident response** and **security automation**. The specific objectives are:

- **Practical Incident Response:** Learn to identify, analyze, and respond to security incidents in a controlled environment.
- **Understanding Vulnerabilities and Web Security:** Gain insights into common vulnerabilities, attack vectors, and best practices for securing web applications and systems.
- **Log Analysis Skills:** Develop the ability to collect, process, and analyze

- logs to detect suspicious activities and security events.
- **Automation for Incident Response:** Implement and test automated security monitoring and response mechanisms to enhance the speed and accuracy of incident handling.

## 2. Requirements

The tools and systems required for this project are categorized into **software** and **operating systems**:

### Software

#### VirtualBox:

A virtualization tool to set up and run virtual machines (VMs) for testing and experimentation. VirtualBox allows creating isolated environments where various operating systems and servers can interact securely.

#### ELK Stack:

- **Elasticsearch:** Stores logs and facilitates quick search and retrieval of log data.
- **Logstash:** Processes and enriches logs for structured analysis.
- **Kibana:** Provides a user-friendly interface to visualize and analyze logs.

#### Incident Response and Vulnerability Scanning Tools:

- **Wireshark:** For network traffic analysis.
- **Nmap:** For network mapping and vulnerability scanning.
- **OWASP ZAP:** For identifying vulnerabilities in web applications.
- **ModSecurity:** To monitor and protect web applications.

### Operating Systems

#### Kali Linux 2024.3:

- A penetration-testing Linux distribution with pre-installed tools for forensics, vulnerability assessments, and incident response.

#### Windows Server:

- Acts as a domain controller and file server to simulate enterprise-grade IT infrastructure.

#### Linux Server:

- Hosts the web server and database server for running applications and storing data.

#### Windows 11:

- Acts as client machines to simulate end-user interaction with the environment.

### 3. Environmental Setup

The setup involves deploying multiple virtual machines (VMs) within VirtualBox to replicate an enterprise IT environment. Each VM will play a specific role, creating a realistic simulation for testing security practices and incident response workflows.

#### Virtual Machines Configuration

##### Forensics OS: Kali Linux 2024.3

- **Purpose:**
  - Perform vulnerability scanning, penetration testing, and forensics.
  - Act as the primary machine for incident response activities.
- **Roles:**
  - Investigate security incidents by analyzing logs and network traffic.
  - Exploit vulnerabilities in the target systems to understand risks.

##### Windows Server

- **Purpose:**
  - Acts as a **Domain Controller** to manage user authentication and access.
  - Serves as a **File Server** for shared storage, simulating an enterprise setup.
- **Roles:**
  - Simulate attack scenarios such as privilege escalation or unauthorized access.
  - Provide event logs for analysis.

## Linux Server

- **Purpose:**
  - Hosts critical applications and data to simulate production servers.
- **Components:**
  - **Web Server:** Hosts a vulnerable web application, such as OWASP Juice Shop.
  - **Database Server:** Stores sensitive data to simulate real-world risks.
- **Roles:**
  - Serve as targets for vulnerability scanning and simulated attacks.
  - Generate logs for incident detection and response activities.

## Client Machines: Windows 11

- **Purpose:**
  - Simulate end-user activities such as accessing web applications, downloading files, and authenticating with the domain.
- **Roles:**
  - Act as sources of normal and malicious activity to test detection mechanisms.
  - Provide realistic traffic and behavior for monitoring.

## Additional Configuration Details

### Networking

- Set up an isolated virtual network in VirtualBox to interconnect the virtual machines securely.
- Use NAT or Host-Only networking to allow controlled internet access if required.

### Logging and Monitoring

- Configure all servers and applications to forward logs to the **ELK Stack**.
- Enable logging on the Windows Server (e.g., Windows Event Logs) and Linux Server (e.g., syslog).

### Attack Simulation

- Deploy tools like Nmap, Wireshark, and OWASP ZAP from the Kali Linux VM to simulate attacks.
- Test WAF rules and incident detection capabilities using ModSecurity.

### Automation

- Use scripts or tools to automate repetitive tasks such as log collection, alert generation, and basic incident response actions.

### Outcome

By setting up and using this environment, participants will:

- Gain a thorough understanding of IT infrastructure components and their vulnerabilities.
- Learn to detect, analyze, and respond to security incidents effectively.
- Build practical skills in setting up and managing a security monitoring environment.
- Experience real-world scenarios for vulnerability scanning, log analysis, and incident response.

## Project Problem

In today's digital landscape, the rise of sophisticated cyber threats has made web application security a critical priority for organizations. Many web applications are vulnerable to a range of attacks that can have severe consequences, including:

- **SQL Injection:** Exploiting database queries to manipulate or access sensitive information.
- **Cross-Site Scripting (XSS):** Injecting malicious scripts to compromise user sessions or steal data.
- **Cross-Site Request Forgery (CSRF):** Forcing users to execute unwanted actions on authenticated sessions.

These vulnerabilities can lead to:

- **Data Breaches:** Exposure of sensitive information.
- **Unauthorized Access:** Compromise of systems and user accounts.
- **Financial Losses:** Direct and indirect costs associated with remediation, fines, and reputational damage.

Despite the availability of advanced security tools and frameworks, a significant challenge persists:

- Many cybersecurity professionals, especially beginners, lack practical knowledge and hands-on experience.
- This gap hinders the effective identification and mitigation of vulnerabilities in real-world scenarios.

As cyber threats continue to evolve, bridging the gap between theoretical knowledge and practical application is crucial. Empowering professionals with the skills and tools to safeguard web applications is a vital step toward strengthening organizational security.

## **Project Solution**

This project aims to address the following key problems:

### **1. Lack of Practical Experience:**

- Many cybersecurity beginners lack hands-on experience with real-world web application security tools and techniques.
- There is a need for an educational platform that provides a safe and controlled environment to practice these skills.

### **2. Inadequate Incident Response:**

- Organizations often struggle with developing effective incident response plans and playbooks.
- There is a need to simulate and practice incident detection and response to improve readiness for actual security incidents.

### **3. Difficulty in Monitoring and Analysis:**

- Monitoring web applications for security events and analyzing logs can be challenging without the right tools and configurations.
- Implementing comprehensive logging and monitoring systems is crucial for effective incident detection and response.



## 4. Understanding of Web Vulnerabilities:

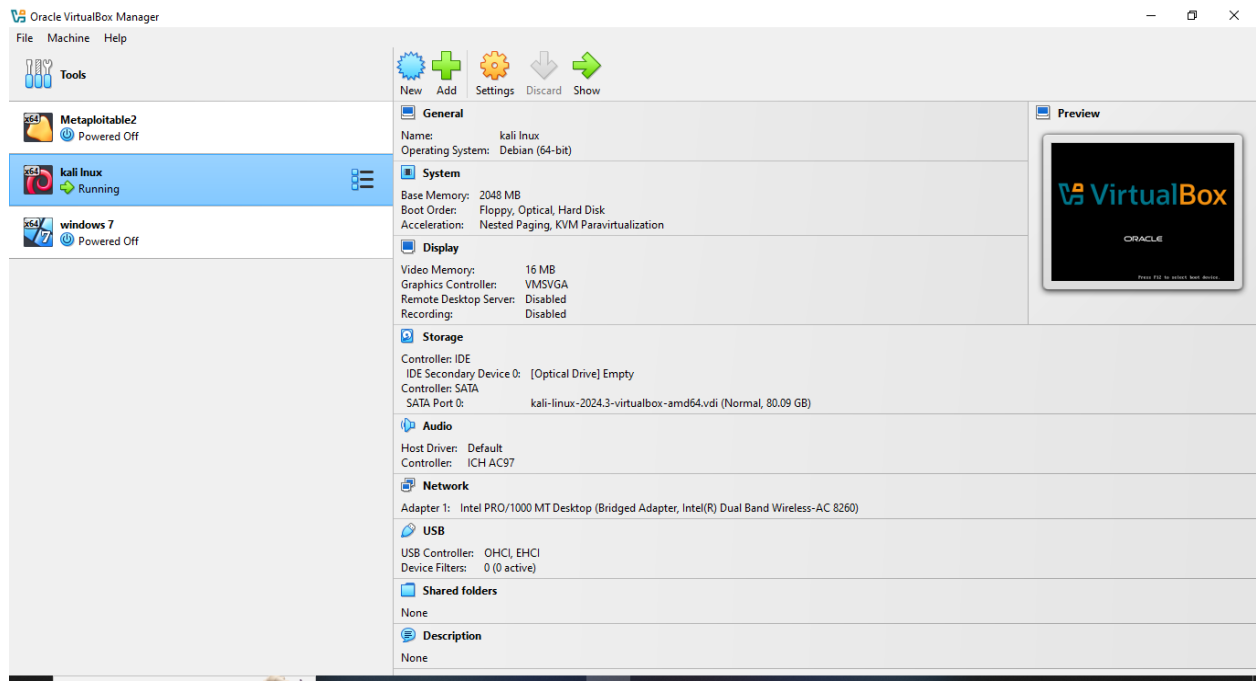
- Beginners need to understand common web vulnerabilities and how to exploit them to better defend against real-world attacks.
- There is a need for structured learning that includes both vulnerability assessment and practical attack simulation.

## Installation of Kali Linux in Virtual Box:

### Step 1: Installation of Kali Linux on Virtualbox

<https://kali.download/base-images/kali-2024.3/kali-linux-2024.3-virtualbox-amd64.7z>

### Step 2: Configuration of Kali Linux on Virtualbox



## Monitoring Logs:

### Step 1:

- Need to install splunk
- Opening the Splunk in the Local host
- Local Host port number:8000
- Home Page of the Splunk has been Loaded

### Steps to install splunk:

➤ `sudo /opt/splunk/bin/splunk start`

**Purpose:** Starts the Splunk service.

**What it does:**

- Runs Splunk from its installation directory, located at `/opt/splunk/bin/`.
- The `sudo` command ensures administrative privileges are used, as starting Splunk often requires elevated permissions.
- During the first run, you may be prompted to accept the Splunk license agreement and set an admin username and password.

**Result:** The Splunk service starts, allowing you to access it via the web interface (`http://localhost:8000`).

➤ `ls -l`

**Purpose:** Lists files and directories in the current working directory with detailed information.

**What it does:**

- Displays contents in a long listing format.
- Provides details like file permissions, ownership, size, and last modification date.

**Useful for:** Checking the attributes of files, such as ensuring that `splunk-9.4.0-6b4ebe426ca6-linux-amd64.deb` is present and has the appropriate permissions.

➤ `sudo dpkg -i splunk-9.4.0-6b4ebe426ca6-linux-amd64.deb`

**Purpose:** Installs the Splunk `.deb` package on your Linux system.

**What it does:**

- The `dpkg -i` command installs the specified `.deb` package.
- `splunk-9.4.0-6b4ebe426ca6-linux-amd64.deb` is the package file for Splunk Enterprise version 9.4.0.
- The `sudo` command ensures you have the necessary permissions to install system software.

**Result:** The Splunk application is installed, typically in the `/opt/splunk` directory.

➤ `ls /opt`

**Purpose:** Lists the contents of the `/opt` directory.

**What it does:**

- Displays all files and directories inside `/opt`, which is often used for optional or third-party software installations.
- Verifies that the `splunk` directory exists, confirming that Splunk has been successfully installed.

➤ `sudo /opt/splunk/bin/splunk start`

**Purpose:** Starts the Splunk service again after installation.

**What it does:**

- Ensures that the Splunk service is running and ready to use.
- Since the command was used earlier, it might now skip initial setup prompts and directly start the service.

## Step1:

```
kali linux [Running] - Oracle VirtualBox
File Machine View Input Devices Help

kali@kali:~/Downloads
$ ls
splunk-9.4.0-6b4eb426ca6-linux-amd64.deb

kali@kali:~/Downloads
$ sudo dpkg -i splunk-9.4.0-6b4eb426ca6-linux-amd64.deb

[sudo] password for kali:
Selecting previously unselected package splunk.
(Reading database ... 48165 files and directories currently installed.)
Preparing to unpack splunk-9.4.0-6b4eb426ca6-linux-amd64.deb ...
no need to run the pre-install check
Unpacking splunk (9.4.0) ...
```

```
kali linux [Running] - Oracle VirtualBox
File Machine View Input Devices Help

kali@kali:~/Downloads
Preparing to unpack splunk-9.4.0-6b4eb426ca6-linux-amd64.deb ...
no need to run the pre-install check
Unpacking splunk (9.4.0) ...
Setting up splunk (9.4.0) ...
complete

kali@kali:~/Downloads
$ dpkg --get-selections
Package: splunk
Status: install ok installed
Maintainer: Splunk Inc. <info@splunk.com>
Architecture: amd64
Version: 9.4.0
Description: Splunk The platform for machine data.

kali@kali:~/Downloads
$ cd /opt/

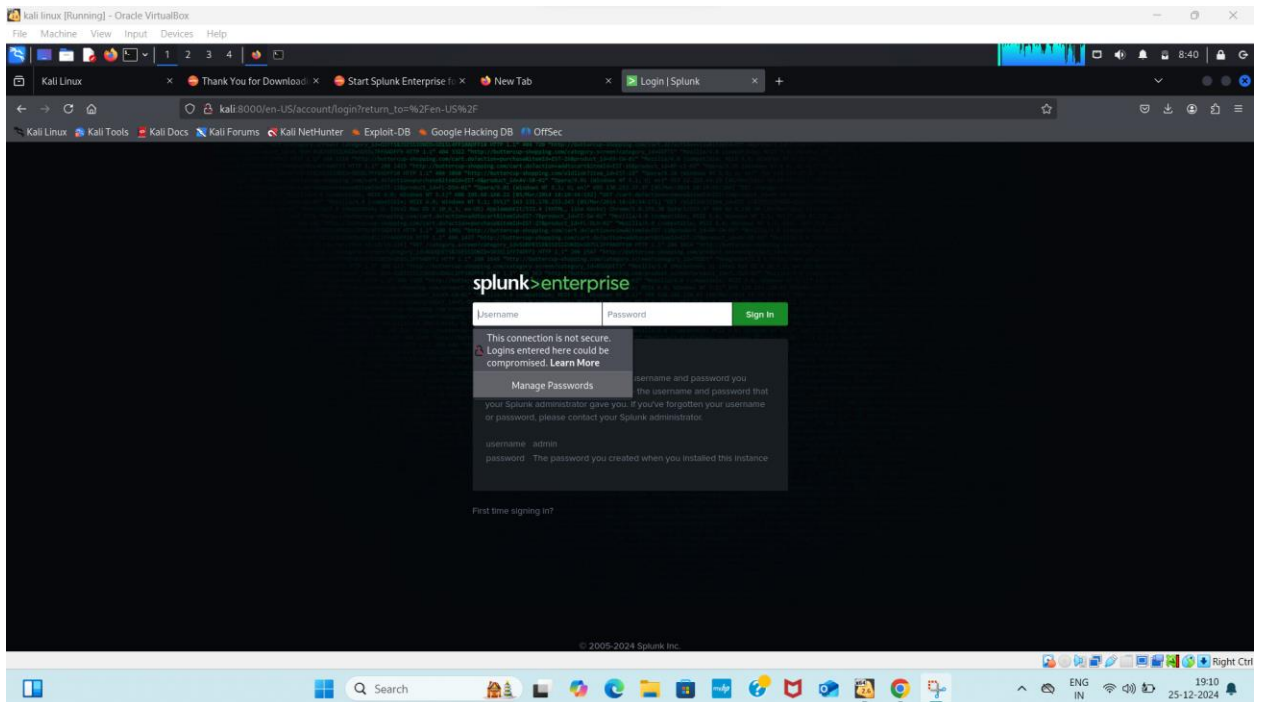
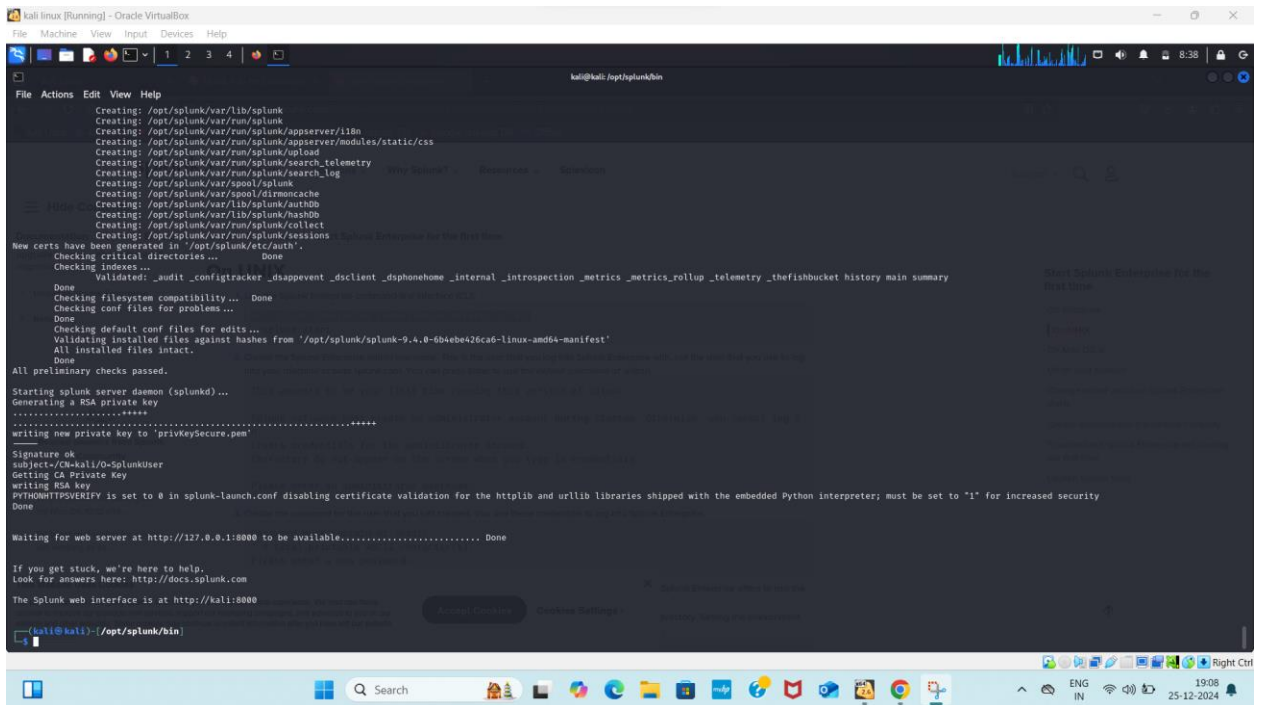
kali@kali:~/opt
$ ls
microsoft splunk

kali@kali:~/opt
$ cd splunk

kali@kali:~/opt/splunk
$ ls
bin  copyright.txt  etc  ftr  include  lib  license-eula.txt  LICENSE.txt  openssl  opt  quarantined_files  README-splunk.txt  share  splunk-9.4.0-6b4eb426ca6-linux-amd64-manifest  suidtag

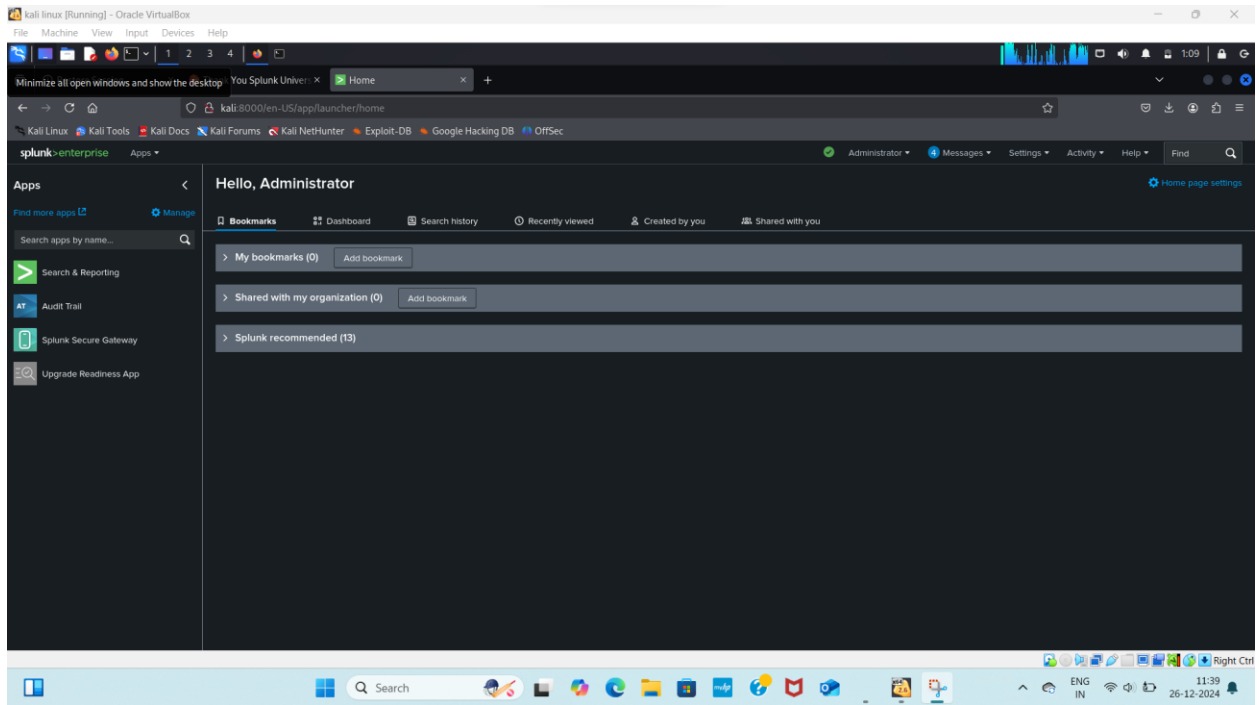
kali@kali:~/opt/splunk
$ cd bin/

kali@kali:~/opt/splunk/bin
$ ls
2to3-3.7  createdb  genWebCert.py  mongod-5.0  patroni_wale_restore  pg_recvwal  postgres  python3.7  scrubber.py  supervisor-simulator
2to3-3.9  createuser  genWebCert.sh  mongod-6.0  pcre2-config  pg_recvlogical  prichunkpng  python3.7m  searchtest  syncobj_admin
blowfish  decompilator.py  id3e3  mongod-7.0  pcre2test  pg_resetwal  priforgepng  python3.9  setSplunkEnv  tarit.py
bottle.py  dropdb  id3e3.7  mongodump  pg_archivecleanup  pg_resetxlog  prirtrpng  pyvenv-3.7  shc_upgrade_template.py  tcpx.py
btutil  dropuser  id3e3.9  mongod_upgrade  pg_basebackup  pg_restore  pripalpng  pyvenv-3.7  signtool  tcsa.py
brcube  egg  inidb  mongorestore  pg_bench  pg_test_fsync  priplnglch  rapidimg  sbin  tsidprobe
bz2p2  etcd  inidb  moah_self_storage_archiver.py  pgbench  pg_test_timing  priplnglch  recover-metadata  spl2-orchestrator  tsidprobe_plo
cdiff  etcdctl  installit.py  node  pg_bouncer  pg_upgrade  priplngtopam  reindexdb  spl2-orchestrator  tsid_scan.py
classify  etcdctl  jave  openssl  pg_checktimes  pg_verifybackp  priplngtopam  reindexdb  splunk  tsid_scan.py
clusterdb  exporttool  jsmn  parse_xml_buckets.py  pg_config  pg_waldump  priplngtopam  reindexdb  splunk  tsid_scan.py
ColdStorageArchiver_GCP.py  fill_summary_index.py  locktest  patroni  pg_controldata  priplngtopam  reindexdb  splunk  tsid_scan.py
ColdStorageArchiver.py  genWebCert.py  locktool  patroni_aus  pg_ctl  priplngtopam  reindexdb  splunk  tsid_scan.py
coldToForexample.py  genWebCertA.sh  mongod  patroni_harman  pg_dump  priplngtopam  reindexdb  splunk  tsid_scan.py
compup  genSignedServerCert.py  mongod-4.2  patronicil  pg_dumpall  priplngtopam  reindexdb  splunk  tsid_scan.py
copyright.txt  genSignedServerCert.sh  mongod-4.4  patroni_raft_controller  pg_isready  priplngtopam  reindexdb  splunk  tsid_scan.py
```



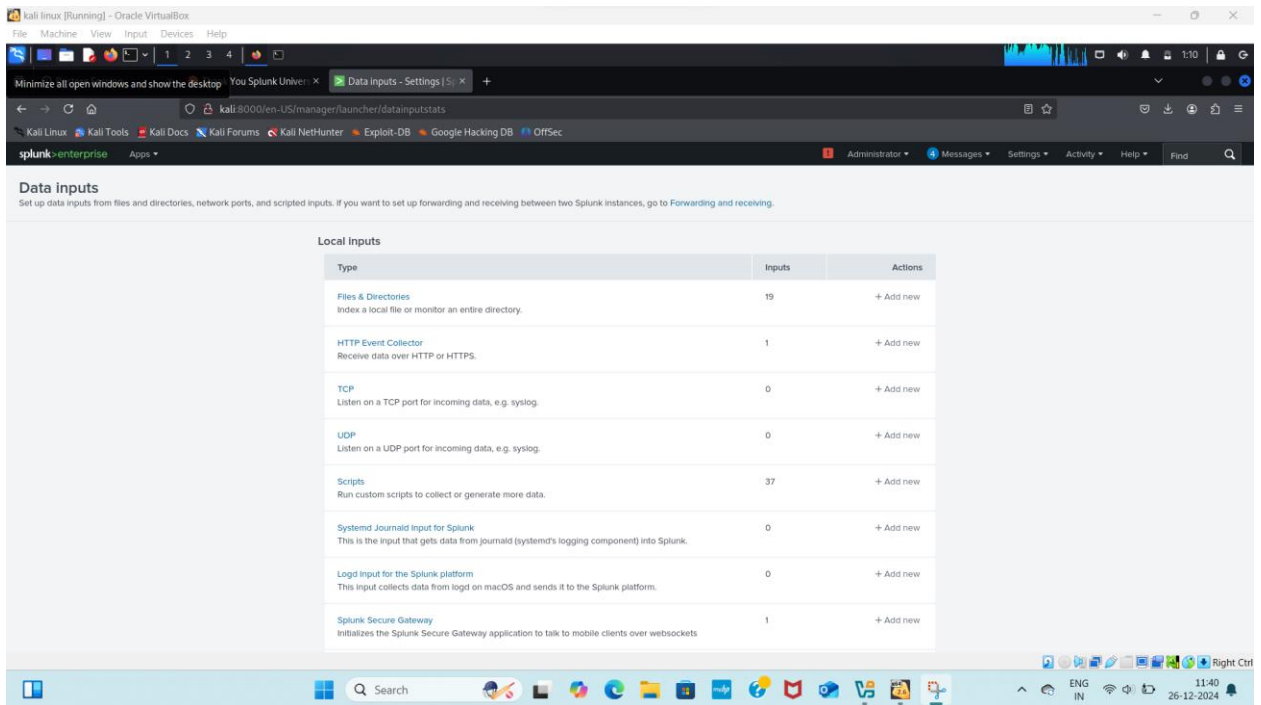
## Step2:

- Open the settings in the top Right corner
- Select the Data Input



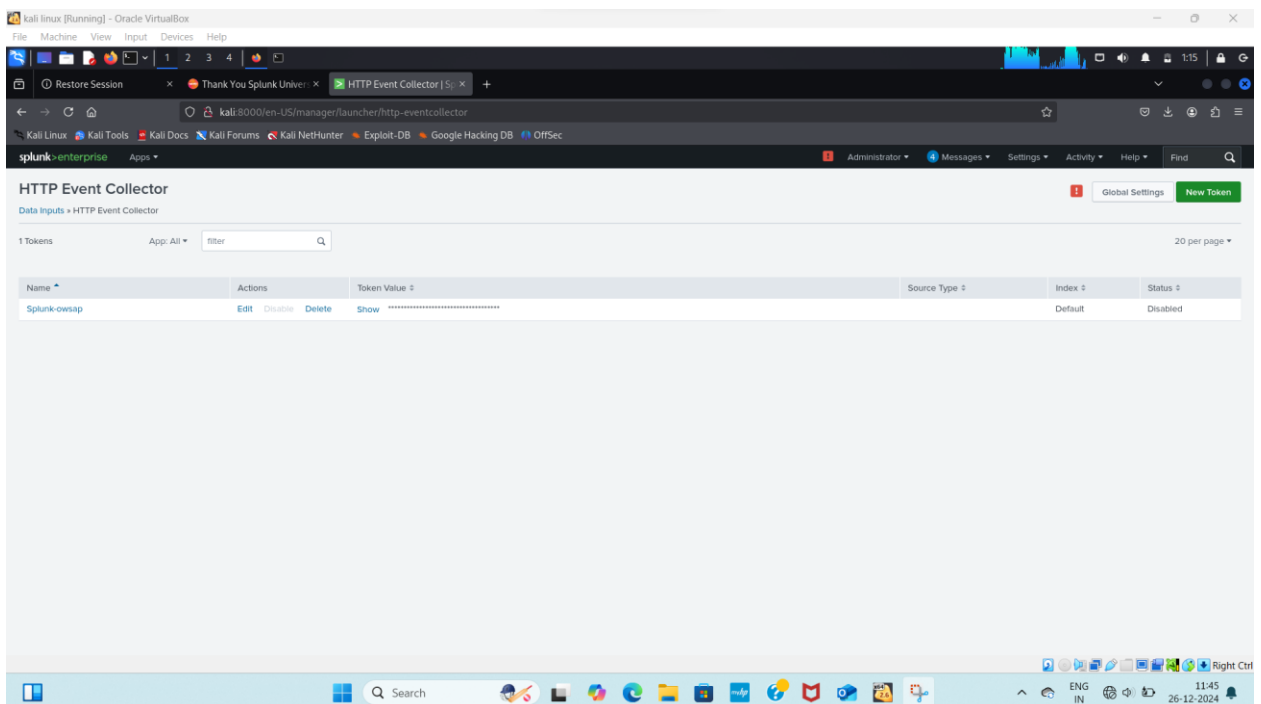
## Step3:

- In Data Input there will some Local Input
- In which select the http event collector
- Http event is used to collect the logs from the Owasp Juice Shop



## Step4:

- After Opening the Http event collector there will option to add the “New token”
- New token is used for indicating the token id which the logs from website will allocated with the “New token”



## Process for creating an token and Add Data

### Step5:

The screenshot shows the 'Add Data - Select Source' configuration page in Splunk Enterprise. The page is titled 'Add Data' and has a progress bar with four steps: Select Source, Input Settings, Review, and Done. The 'Select Source' step is currently active. On the left, there is a list of source types: Files & Directories, HTTP Event Collector, TCP / UDP, Scripts, Systemd Journal Input for Splunk, Log Input for the Splunk platform, Splunk Secure Gateway, Splunk Secure Gateway Mobile Alerts TTL, and Deep Link Dashboard Modular Input. The 'HTTP Event Collector' option is selected. On the right, there is a form to configure a new token. The form fields are: Name (Splunk-owsap), Source name override (optional), Description (optional), Output Group (optional) (None), and Enable indexer acknowledgement (checkbox). Below the form is an FAQ section with questions about the HTTP Event Collector.

**Add Data** Select Source Input Settings Review Done < Back Next >

**Files & Directories**  
Upload a file, index a local file, or monitor an entire directory.

**HTTP Event Collector**  
Configure tokens that clients can use to send data over HTTP or HTTPS.

**TCP / UDP**  
Configure the Splunk platform to listen on a network port.

**Scripts**  
Get data from any API, service, or database with a script.

**Systemd Journal Input for Splunk**  
This is the input that gets data from journald (systemd's logging component) into Splunk.

**Log Input for the Splunk platform**  
This input collects data from logd on macOS and sends it to the Splunk platform.

**Splunk Secure Gateway**  
Initializes the Splunk Secure Gateway application to talk to mobile clients over websockets.

**Splunk Secure Gateway Mobile Alerts TTL**  
Cleans up storage of old mobile alerts.

**Deep Link Dashboard Modular Input**  
Introduces the Deep Link Dashboard Modular input to complete registrations.

Configure a new token for receiving data over HTTP. [Learn More](#)

Name: Splunk-owsap

Source name override: optional

Description: optional

Output Group (optional): None

Enable indexer acknowledgement: ☐

**FAQ**

- > What is the HTTP Event Collector?
- > How do I set up the HTTP Event Collector?
- > How do I view and configure the tokens that I can use to send data to the HTTP Event Collector?
- > What clients can send data to the HTTP Event Collector?
- > What port and protocol does the HTTP Event Collector receive data on and how can I change that?
- > What is an output group?

### Step6: Review

The screenshot shows the 'Add Data - Review' configuration page in Splunk Enterprise. The page is titled 'Add Data' and has a progress bar with four steps: Select Source, Input Settings, Review, and Done. The 'Review' step is currently active. The page displays a summary of the configuration details for the HTTP Event Collector token.

**Add Data** Select Source Input Settings Review Done < Back Submit >

**Review**

Input Type: Token

Name: Splunk-owsap

Source name override: N/A

Description: N/A

Enable indexer acknowledgement: No

Output Group: N/A

Allowed indexes: N/A

Default index: default

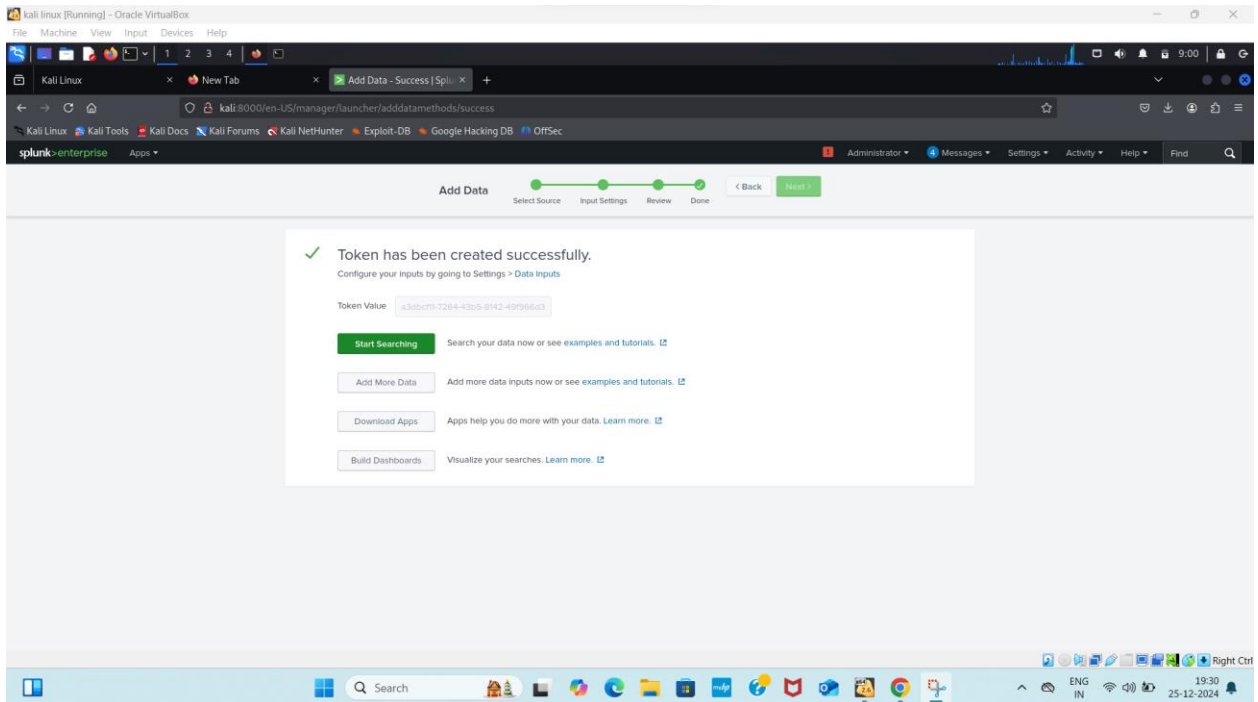
Source Type: Automatic

App Context: launcher



## Step7:

- Token has been created Successfully



## Steps to download OWASP juice-shop Application:

OWASP Juice Shop is an intentionally vulnerable web application for security training.

### Steps:

**Install Node.js and npm:** Juice Shop requires Node.js. Install it with the following commands:

- `sudo apt update`

**Purpose:** Updates the package list of your system.

**What it does:**

This command retrieves the latest package lists from the repositories configured on your system. It ensures that you get the most recent versions of packages and dependencies when installing or upgrading software.

➤ `tar -xvzf juice-shop_17.1.1`

**purpose:** Extracts the contents of the `juice-shop_17.1.1` tarball file.

**Options explained:**

**-x:** Extract the files from the tarball.

**-v:** Verbose mode, shows the progress and lists the files being extracted.

**-z:** Specifies that the tarball is compressed with gzip.

**-f:** Indicates the name of the file to process.

**What it does:**

This command extracts the `juice-shop_17.1.1` tarball into the current directory. The extracted folder will contain the source files for OWASP Juice Shop.

➤ `sudo apt-get install nodejs`

**Purpose:** Installs Node.js, a JavaScript runtime environment.

**What it does:**

This command downloads and installs Node.js, which is required to run JavaScript code on the server-side, such as the Juice Shop application.

➤ `sudo apt-get install npm`

**Purpose:** Installs npm (Node Package Manager), a tool for managing Node.js packages.

**What it does:**

npm is essential for downloading and managing dependencies for Node.js projects. Juice Shop relies on several Node.js libraries and frameworks that are managed through npm.

- `cd juice-shop_17.1.1`

**Purpose:** Changes the current directory to the extracted Juice Shop folder.

**What it does:**

This moves your terminal session into the `juice-shop_17.1.1` directory, where the application's source code and dependencies are located. All subsequent commands will be executed relative to this directory.

- `pwd`

**Purpose:** Prints the current working directory.

**What it does:**

This displays the full path of the directory you are currently in. In this case, it shows the path to the `juice-shop_17.1.1` directory, helping you confirm the location of the application files.

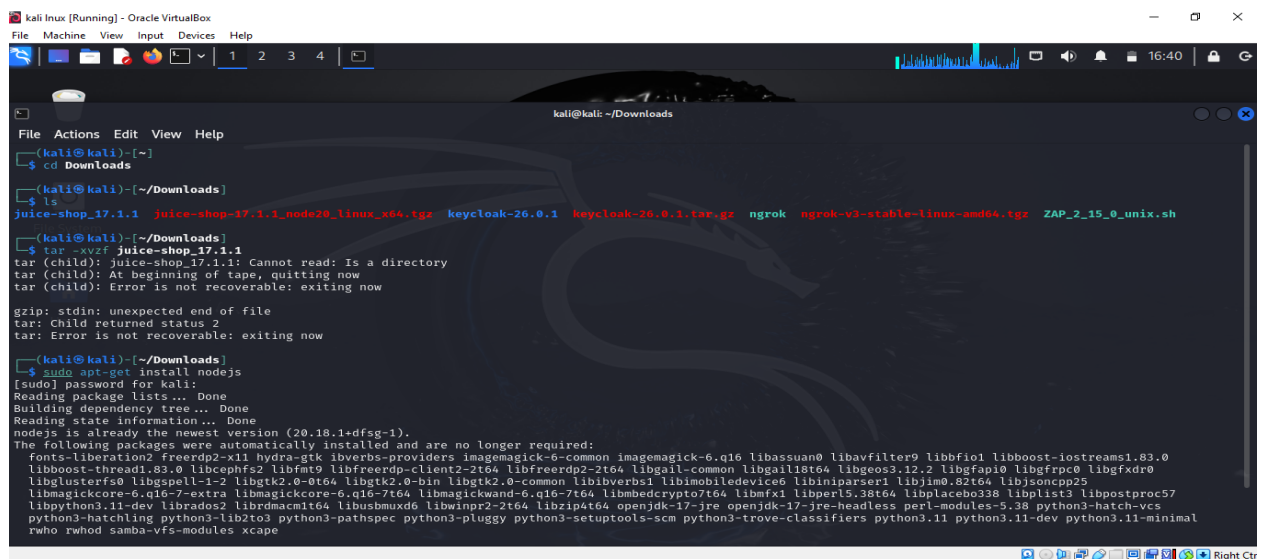
- `npm start`

**Purpose:** Starts the Juice Shop application.

**What it does:**

This command looks for a script named `start` defined in the `package.json` file of the Juice Shop project.

Executes that script to start the application. By default, this will start a server on port 3000, making Juice Shop available at <http://localhost:3000>.



```
kali Linux [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali@kali: ~/Downloads
File Actions Edit View Help
(kali@kali)-[~]
└─$ cd Downloads
(kali@kali)-[~/Downloads]
└─$ ls
juice-shop_17.1.1  juice-shop-17.1.1_node20_linux_x64.tgz  keycloak-26.0.1  keycloak-26.0.1.tar.gz  ngrok  ngrok-v3-stable-linux-amd64.tgz  ZAP_2.15.0_unix.sh
(kali@kali)-[~/Downloads]
└─$ tar -xvzf juice-shop_17.1.1
tar (child): juice-shop_17.1.1: Cannot read: Is a directory
tar (child): At beginning of tape, quitting now
tar (child): Error is not recoverable: exiting now
gzip: stdin: unexpected end of file
tar: Child returned status 2
tar: Error is not recoverable: exiting now
(kali@kali)-[~/Downloads]
└─$ sudo apt-get install nodejs
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nodejs is already the newest version (20.18.1-dfsg-1).
The following packages were automatically installed and are no longer required:
  fonts-liberation2 freerdp2-x11 hydra-gtk ibverbs-providers imagemagick-6-common imagemagick-6.q16 libassuan0 libavfilter9 libbfi0 libboost-iostreams1.83.0
  libboost-thread1.83.0 libcephfs2 libfmt9 libfreerdp-client2-2t64 libfreerdp2-2t64 libgail-common libgail18t64 libgeos3.12.2 libgfan10 libgfrpc0 libgfxdr0
  libgusterfs0 libgspell-1-2 libgtk2-0-6t64 libgtk2-0-bin libgtk2-0-common libibverbs1 libimobiledevice6 libinputparser1 libjim0.82t64 libjsoncpp25
  libmagiccore-6.q16-7-extra libmagiccore-6.q16-7t64 libmagicwand-6.q16-7t64 libmbcrypted07t64 libmfx1 libperl5.38t64 libplacebo338 libplist3 libpostproc57
  libpython3.11-dev librados2 librdmacm1t64 libusbmuxd6 libwinpr2-2t64 libzip6t64 openjdk-17-jre openjdk-17-jre-headless perl-modules-5.38 python3-hatch-vcs
  python3-hatchling python3-lib2to3 python3-pathspect python3-pluggy python3-setuptools-scm python3-trove-classifiers python3.11 python3.11-dev python3.11-minimal
  rwho rwho-d samba-vfs-modules xcape
```

```
kali Linux [Running] - Oracle VirtualBox
File Machine View Input Devices Help

kali@kali: ~/Downloads

File Actions Edit View Help
Reading state information... Done
nodejs is already the newest version (20.18.1+dfsg-1).
The following packages were automatically installed and are no longer required:
 fonts-liberation2 freerdp2-x11 hydra-gtk ibverbs-providers imagemagick-6-common imagemagick-6.q16 libassuan0 libavfilter9 libbfi0 libboost-iostreams1.83.0
 libboost-thread1.83.0 libcephfs2 libfnt9 libfreerdp-client2-2t64 libfreerdp2-2t64 libgail-common libgail18t64 libgeos3.12.2 libgfpapi0 libgfrpc0 libgfxdr0
 libglusterfs0 libgspell-1-2 libgtk2.0-0t64 libgtk2.0-bin libgtk2.0-common libibverbs1 libimobiledevice6 libiniparser1 libjim0.82t64 libjsoncpp25
 libmagickcore-6.q16-7-extra libmagickcore-6.q16-7t64 libmagickwand-6.q16-7t64 libmbcrypto7t64 libmfx1 libperl5.38t64 libplacebo338 libplist3 libpostproc57
 libpython3.11-dev librados2 librdmacm1t64 libusbmuxd6 libwinpr2-2t64 libzip4t64 openjdk-17-jre openjdk-17-jre-headless perl-modules-5.38 python3-hatch-vcs
 python3-hatchling python3-lib2to3 python3-pathspect python3-pluggy python3-setuptools-scm python3-trove-classifiers python3.11 python3.11-dev python3.11-minimal
 rwho rwho-d samba-vfs-modules xcave
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 407 not upgraded.

(kali@kali)~[~/Downloads]
$ sudo apt-get install npm
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
npm is already the newest version (9.2.0-ds1-3).
The following packages were automatically installed and are no longer required:
 fonts-liberation2 freerdp2-x11 hydra-gtk ibverbs-providers imagemagick-6-common imagemagick-6.q16 libassuan0 libavfilter9 libbfi0 libboost-iostreams1.83.0
 libboost-thread1.83.0 libcephfs2 libfnt9 libfreerdp-client2-2t64 libfreerdp2-2t64 libgail-common libgail18t64 libgeos3.12.2 libgfpapi0 libgfrpc0 libgfxdr0
 libglusterfs0 libgspell-1-2 libgtk2.0-0t64 libgtk2.0-bin libgtk2.0-common libibverbs1 libimobiledevice6 libiniparser1 libjim0.82t64 libjsoncpp25
 libmagickcore-6.q16-7-extra libmagickcore-6.q16-7t64 libmagickwand-6.q16-7t64 libmbcrypto7t64 libmfx1 libperl5.38t64 libplacebo338 libplist3 libpostproc57
 libpython3.11-dev librados2 librdmacm1t64 libusbmuxd6 libwinpr2-2t64 libzip4t64 openjdk-17-jre openjdk-17-jre-headless perl-modules-5.38 python3-hatch-vcs
 python3-hatchling python3-lib2to3 python3-pathspect python3-pluggy python3-setuptools-scm python3-trove-classifiers python3.11 python3.11-dev python3.11-minimal
 rwho rwho-d samba-vfs-modules xcave
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 407 not upgraded.

(kali@kali)~[~/Downloads]
$
```

```
kali Linux [Running] - Oracle VirtualBox
File Machine View Input Devices Help

kali@kali: ~/Downloads/juice-shop_17.1.1

File Actions Edit View Help
ls
bom.json CODE_OF_CONDUCT.md CONTRIBUTING.md encryptionkeys HALL_OF_FAME.md LICENSE node_modules REFERENCES.md server.ts uploads
bom.xml config ctf.key frontend i18n logs package.json routes SOLUTIONS.md views
build config.schema.yml data ftp lib models README.md SECURITY.md swagger.yml

(kali@kali)~[~/Downloads/juice-shop_17.1.1]
$ npm start

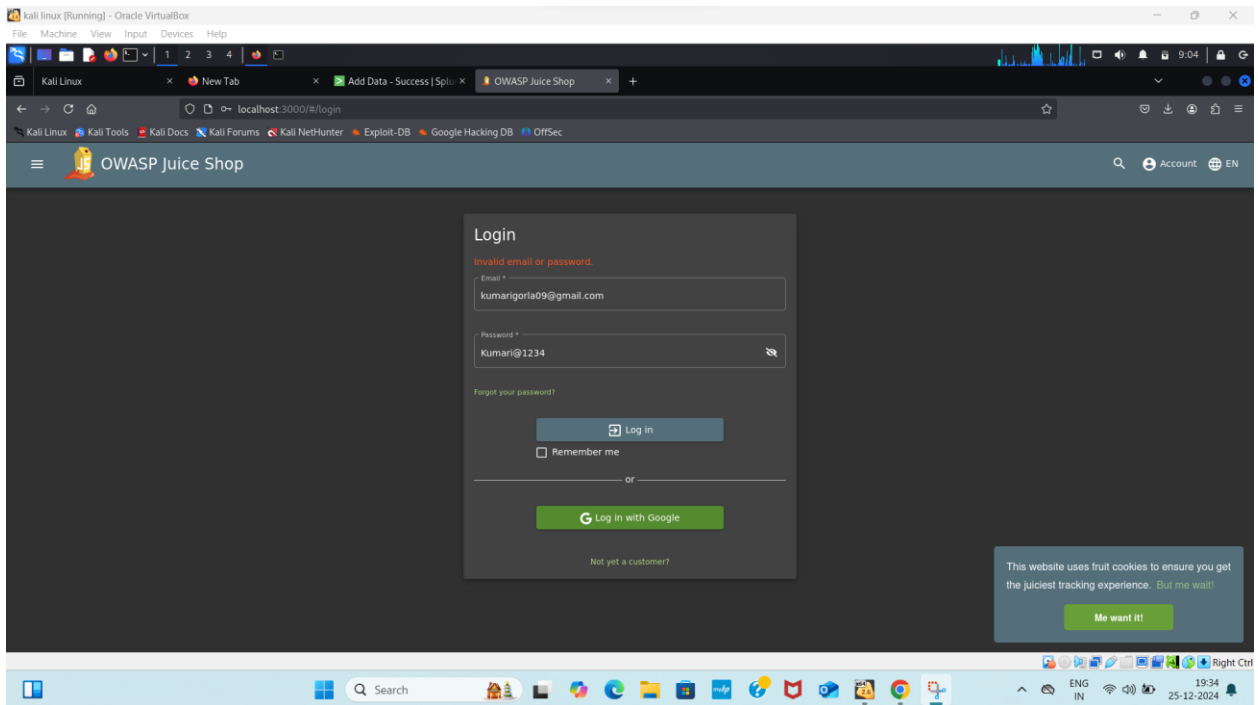
> juice-shop@17.1.1 start
> node build/app

info: Detected Node.js version v20.18.1 (OK)
info: Detected OS linux (OK)
info: Detected CPU x64 (OK)
info: Configuration default validated (OK)
info: Entity models 19 of 19 are initialized (OK)
info: Required file server.js is present (OK)
info: Required file main.js is present (OK)
info: Required file runtime.js is present (OK)
info: Required file index.html is present (OK)
info: Required file polyfills.js is present (OK)
info: Required file vendor.js is present (OK)
info: Required file styles.css is present (OK)
info: Port 3000 is available (OK)
info: Chatbot training data botDefaultTrainingData.json validated (OK)
info: Server listening on port 3000
warn: Domain https://www.alchemy.com/ is not reachable (NOT OK in a future major release)
warn: "Mint the HoneyPot" challenge will not work as intended without access to https://www.alchemy.com/
warn: "Wallet Depletion" challenge will not work as intended without access to https://www.alchemy.com/

Right Ctrl
```

## Step8:

- Login credentials



- Above image is the owasp juice shop login page where after entering the login credentials it will send the logs to splunk and we need to address the log in the splunk
- Need to check the log in the splunk and we need to check the alterusing the automatic scanner tool which is “Zaproxy”.
- The log will be monitored through splunk

## Step9:

### Accessing the log file

- Below image is the log source get collected in the splunk after entering the login credentials
- The logs are stored in the access.log file
- Cat command is used to see the access.log file





- The above is Http Event collector where we are able to see all the logs including the timestamp when the log has been loaded etc..
- Thus the successful configuration and monitoring of the logs in the splunk

## Installation of Zaproxy:

- OWASP Zed Attack Proxy (ZAP) is an open-source web application security scanner and penetration testing tool developed by the Open Web Application Security Project (OWASP). It is designed to find security vulnerabilities in web applications and services.

### Commands to install Zaproxy:

➤ cd Downloads

**Purpose:** Changes the current working directory to the Downloads folder.

**What it does:**

- The cd command (short for "change directory") moves your terminal session into the Downloads directory.
- This directory typically contains files downloaded from the internet, such as the ZAP\_2\_15\_0\_unix.sh file.

➤ ls

**Purpose:** Lists the contents of the current directory.

**What it does:**

The ls command displays all files and folders in the Downloads directory.

This helps confirm that the `ZAP_2_15_0_unix.sh` file is present in the directory.

➤ `chmod +x ZAP_2_15_0_unix.sh`

**Purpose:** Makes the `ZAP_2_15_0_unix.sh` file executable.

**What it does:**

- The `chmod` command modifies the permissions of a file.
- The `+x` option gives the file executable permission, allowing it to be run as a script or program.
- Without this step, attempting to execute the file might result in a "Permission denied" error.

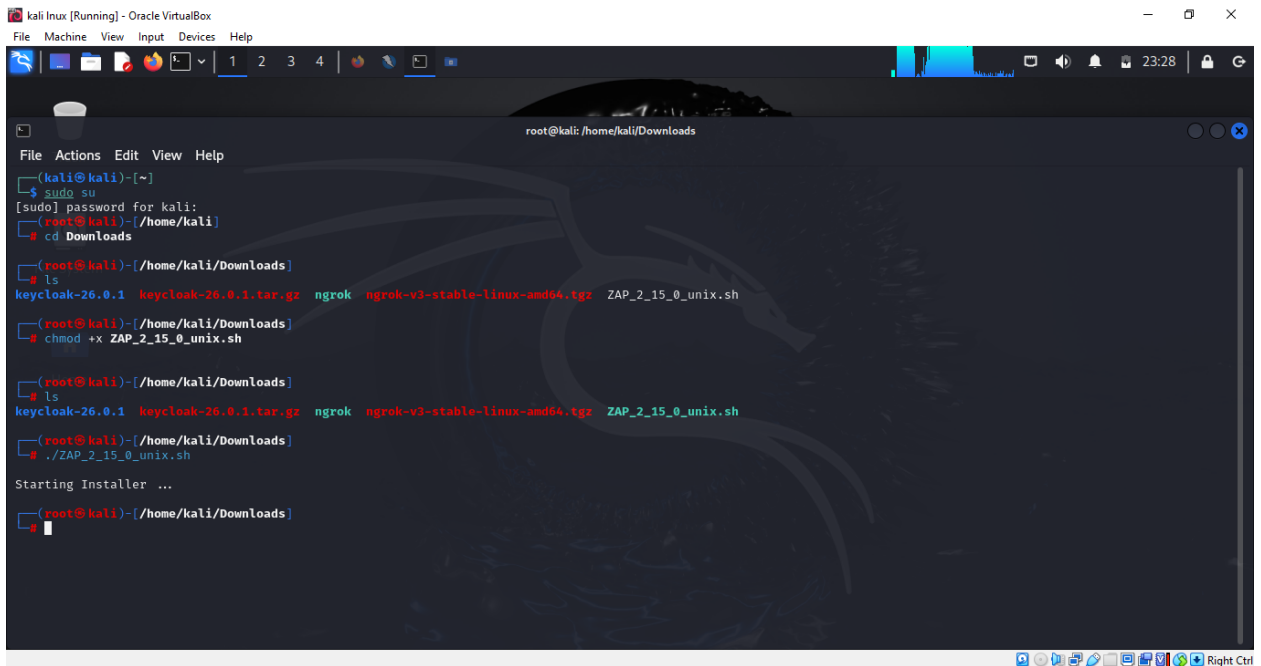
➤ `./ZAP_2_15_0_unix.sh`

**Purpose:** Executes the `ZAP_2_15_0_unix.sh` installation script.

**What it does:**

- The `./` tells the shell to execute the script from the current directory (not from the system's PATH).
- This script is the installer for OWASP ZAP (Zed Attack Proxy) version 2.15.0.
- Running the script initiates the installation or setup process for ZAP on your system.

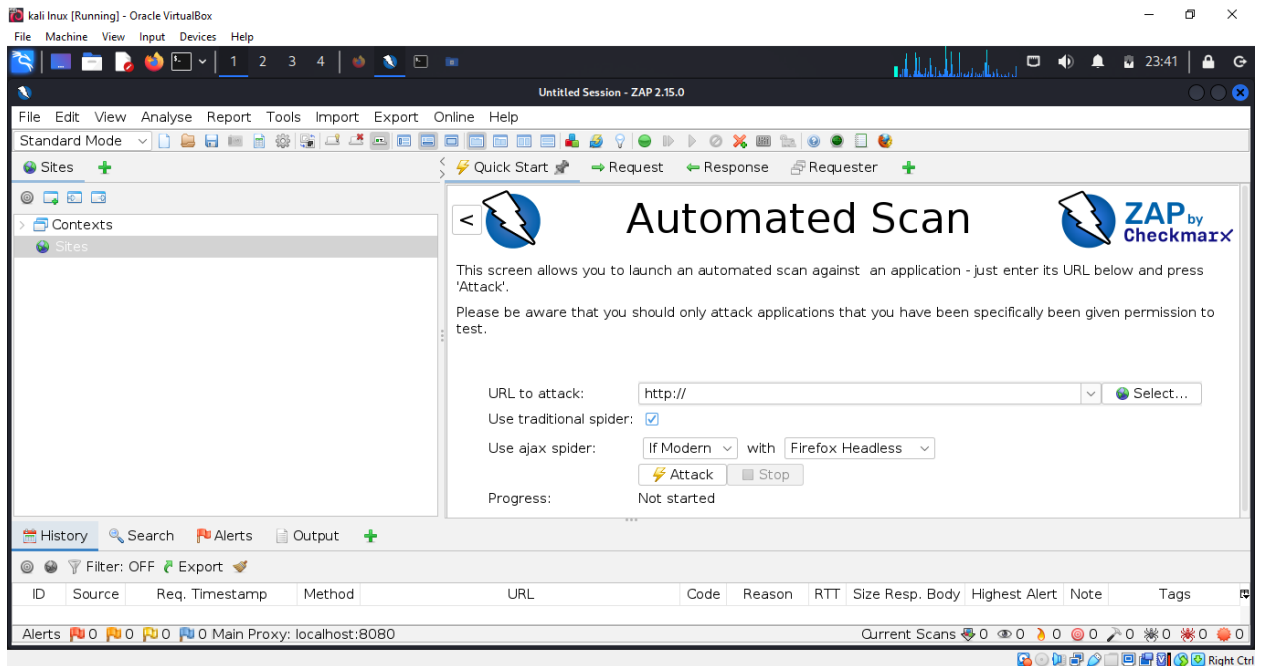




```
kali Linux [Running] - Oracle VirtualBox
File Machine View Input Devices Help
1 2 3 4
root@kali: /home/kali/Downloads

File Actions Edit View Help
(kali@kali)~]
$ sudo su
[sudo] password for kali:
(root@kali)~]
$ cd Downloads
(root@kali)~]
$ ls
keycloak-26.0.1 keycloak-26.0.1.tar.gz ngrok ngrok-v3-stable-linux-amd64.tgz ZAP_2_15_0_unix.sh
(root@kali)~]
$ chmod +x ZAP_2_15_0_unix.sh
(root@kali)~]
$ ls
keycloak-26.0.1 keycloak-26.0.1.tar.gz ngrok ngrok-v3-stable-linux-amd64.tgz ZAP_2_15_0_unix.sh
(root@kali)~]
$ ./ZAP_2_15_0_unix.sh
Starting Installer ...
(root@kali)~]
$
```

- **Interception Proxy:** ZAP can intercept and inspect HTTP and HTTPS traffic between the browser and the web application, allowing testers to see and modify requests and responses.
- **Automated Scanner:** ZAP includes automated scanners that can crawl a web application and test for common vulnerabilities, such as SQL injection, Cross-Site Scripting (XSS), and Cross-Site Request Forgery (CSRF).
- **Manual Testing Tools:** ZAP offers various tools for manual security testing, including a request editor, an active scanner, a spider for crawling, and a Fuzzer for exploring input fields.



## Step by Step process for Vulnerability Scanning:

### Install and Launch ZAP:

- Ensure ZAP is installed and then launch the application.

### Configure ZAP to Proxy Traffic:

- Set up your browser to use ZAP as a proxy (default: localhost:8080).
- Configure SSL settings in your browser to accept ZAP's root CA certificate if you need to intercept HTTPS traffic.

### Start ZAP:

- Open ZAP and make sure it's running.

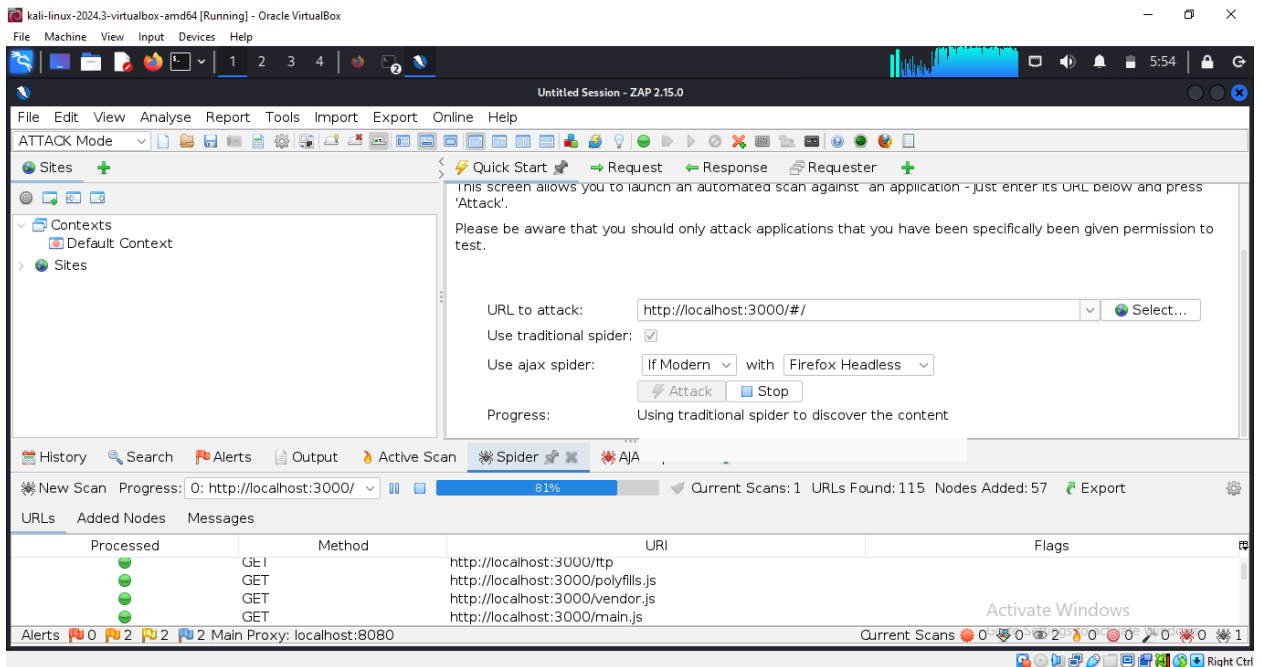
### Access Juice Shop through ZAP:

- In your browser, navigate to `http://localhost:3000` with ZAP running as the proxy. This will allow ZAP to intercept and log the traffic.

### Run Automated Scan:

- In ZAP, right-click on the site node

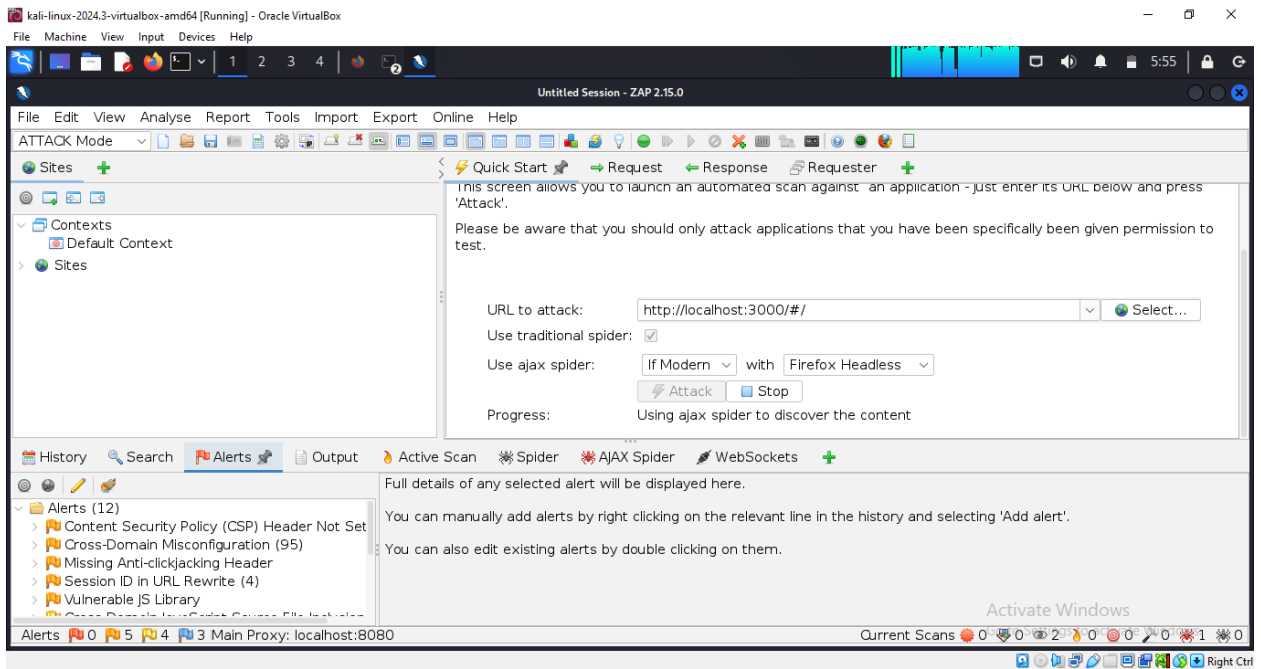
- (e.g.,http://localhost:3000) in the "Sites" tab.
- Select "Attack" > "Spider to discover content".
- After the spider completes, right-click on the site node again and select "Attack" > "Active Scan".



## Review Results:

- Monitor the progress and results of the scan in the "Active Scan" tab.
- Analyze the identified vulnerabilities and issues reported by ZAP.

Reviewing the result by checking the high alerts:



- The above image indicates the high vulnerabilities while Scanning using Zaproxy.

## Conclusion:

### Integration of Splunk and OWASP Juice Shop

This incident response and automation project demonstrates the effectiveness of integrating Splunk and OWASP Juice Shop for security testing and incident response.

- **Splunk for Monitoring and Analysis:** Splunk provides comprehensive monitoring and analysis capabilities, enabling valuable insights into security events and incidents. This enhances incident response capabilities by allowing for rapid identification and investigation of security issues.
- **OWASP Juice Shop for Security Testing:** OWASP Juice Shop, with its intentionally vulnerable design, serves as an ideal platform for testing and simulating various security scenarios. This allows for the evaluation and refinement of response strategies in a controlled environment.
- **Benefits:**
  - **Improved Security Posture:** By actively testing and simulating security incidents, organizations can identify and address vulnerabilities proactively, improving their overall security posture.
  - **Enhanced Incident Response Capabilities:** The insights gained from Splunk analysis can significantly improve the speed and effectiveness of incident response.
  - **Valuable Learning Experience:** This approach provides a valuable learning experience in managing and automating security operations.

### Key Takeaways:

- This project highlights the importance of advanced monitoring and analysis tools like Splunk in enhancing security operations.
- OWASP Juice Shop provides a valuable resource for security testing and training.
- The lessons learned and techniques developed in this project can serve as a foundation for future improvements in cybersecurity practices and automation.

**Note:** This is 100% of the project has been completed and reviewed by the trainer.

**Team Members:**

1. Katta Monika
2. Swamireddy Ramya
3. Gorla Lova Kumari