Student:	Email:
Udo Udo Williams	raggg12@gmail.com

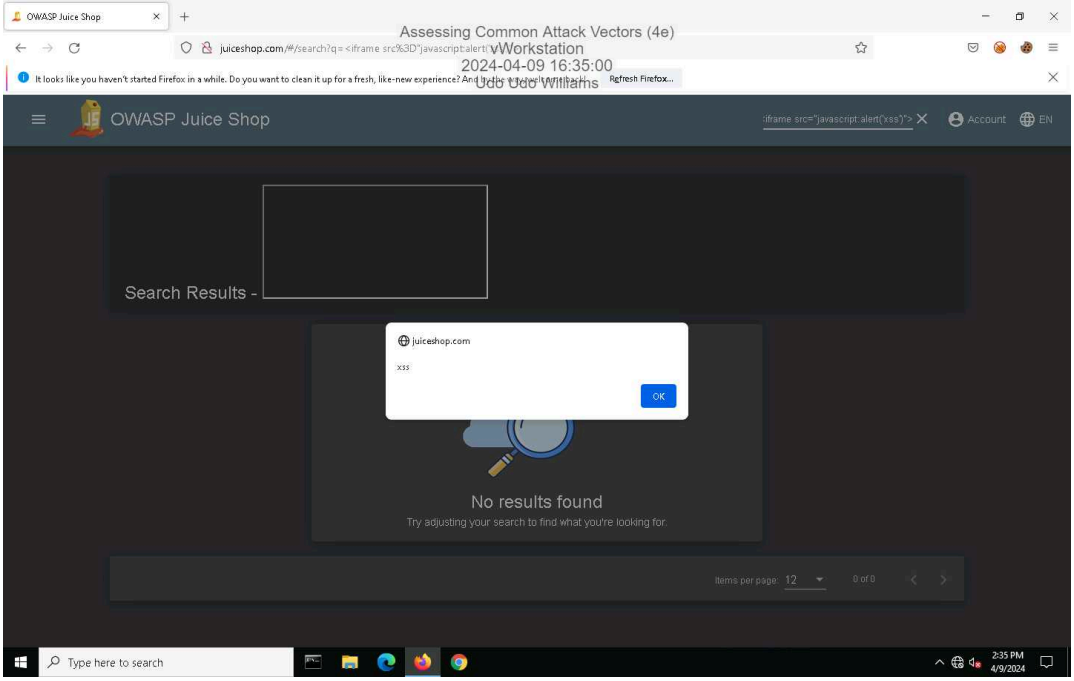
Time on Task:	Progress:
16 hours, 18 minutes	100%

Report Generated: Tuesday, April 9, 2024 at 10:48 PM

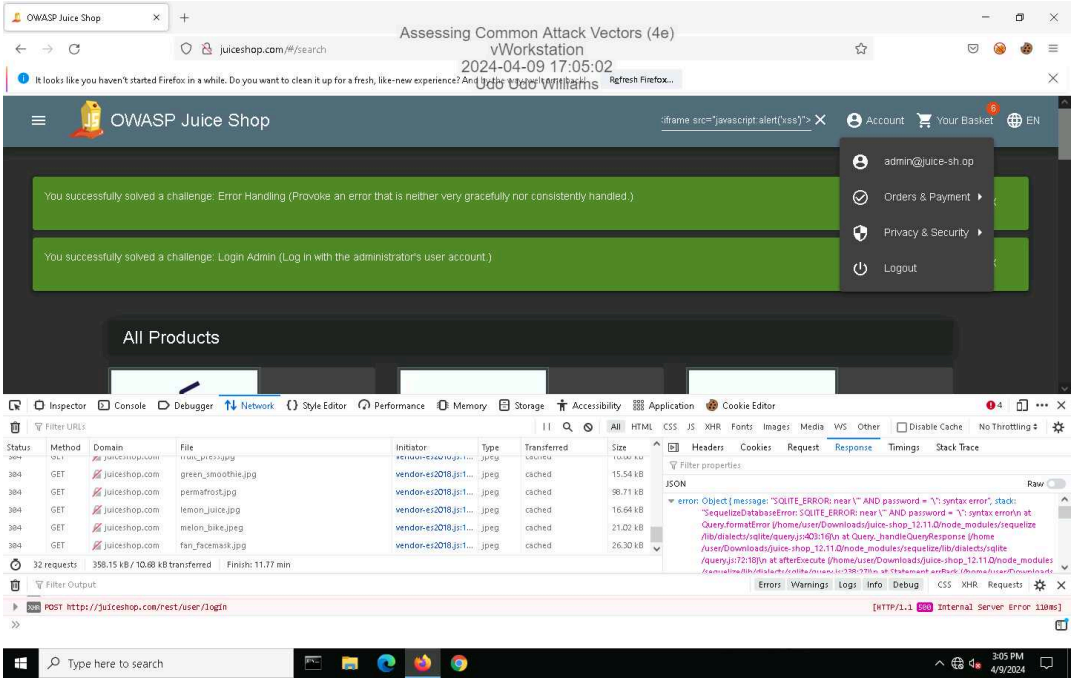
Section 1: Hands-On Demonstration

Part 1: Perform an Injection Attack

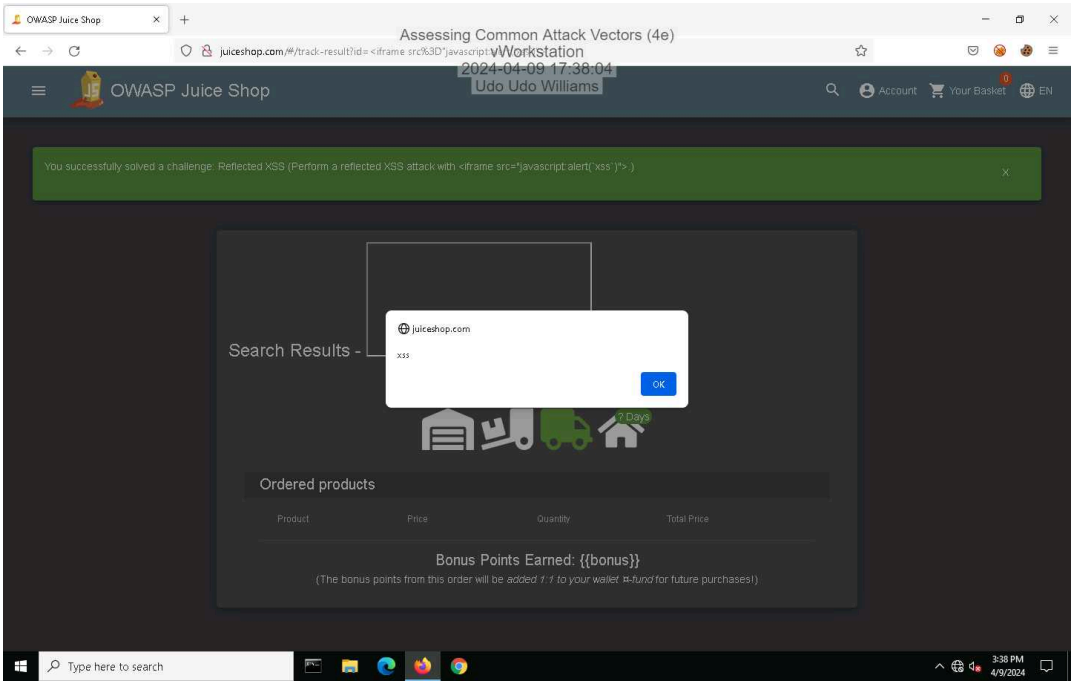
11. **Make a screen capture** showing the **DOM XSS dialog box**.



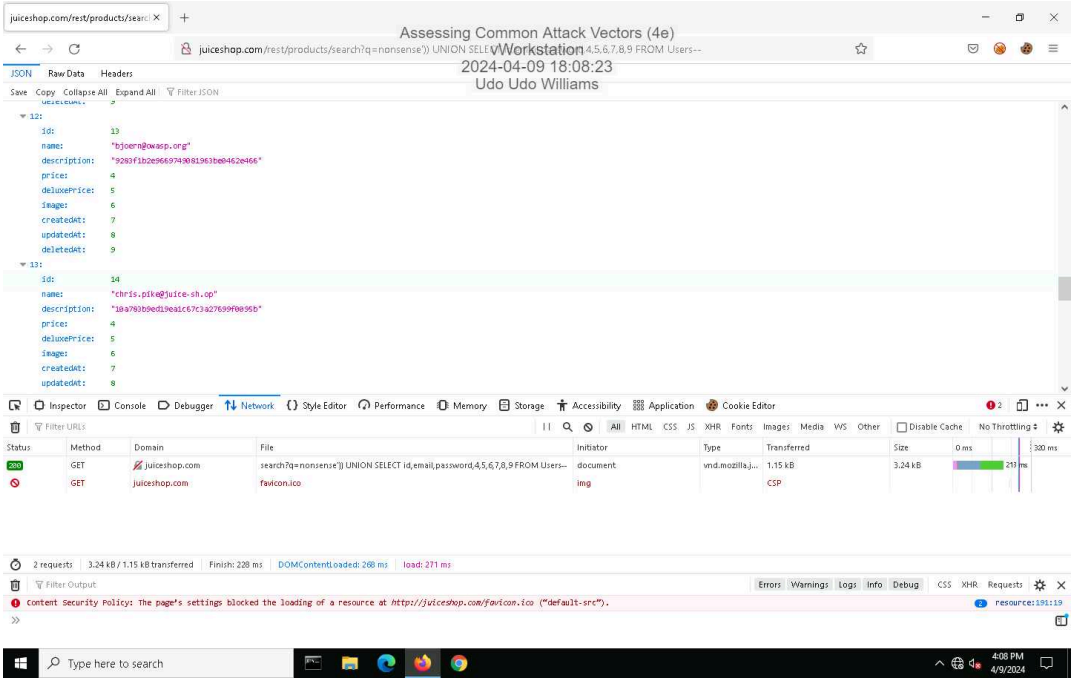
21. **Make a screen capture** showing the **successful admin login**.



26. **Make a screen capture** showing the **successful Reflected XSS injection**.

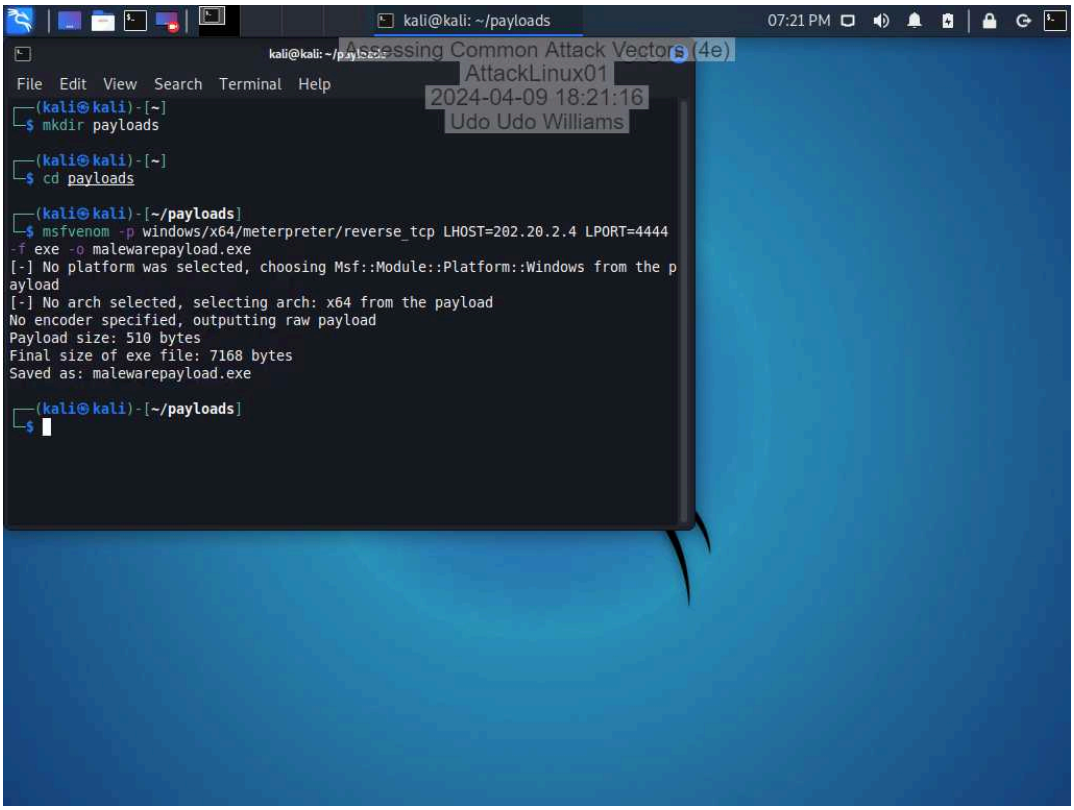


42. **Make a screen capture** showing the **user with the @owasp.org email**.

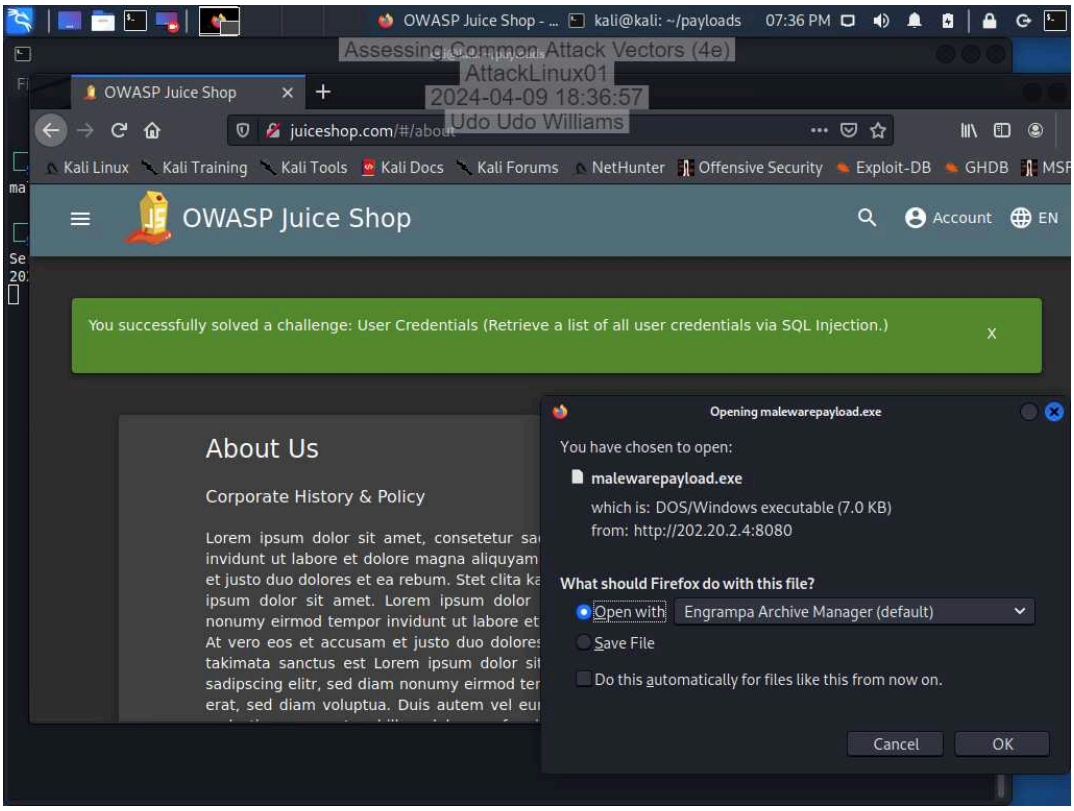


Part 2: Perform a Malware Attack

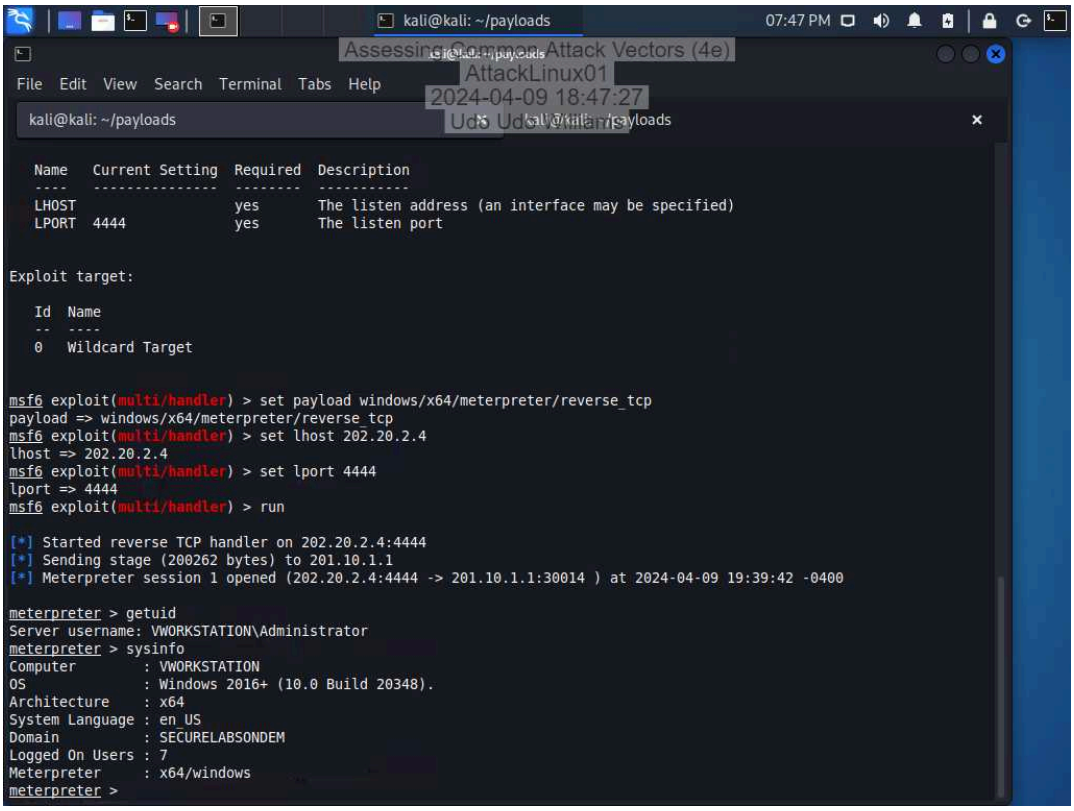
6. **Make a screen capture** showing the **msfvenom output**.



23. **Make a screen capture** showing the **Opening malwarePayload.exe dialog box**.



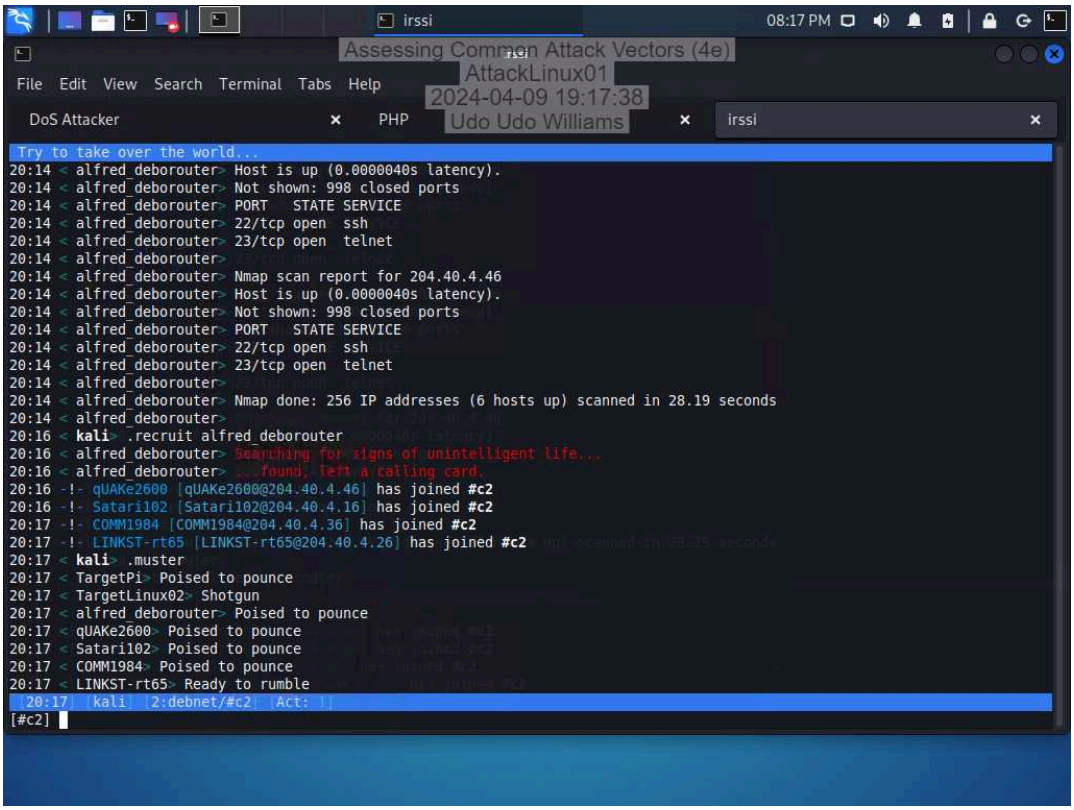
36. **Make a screen capture** showing the **output of the sysinfo command**.



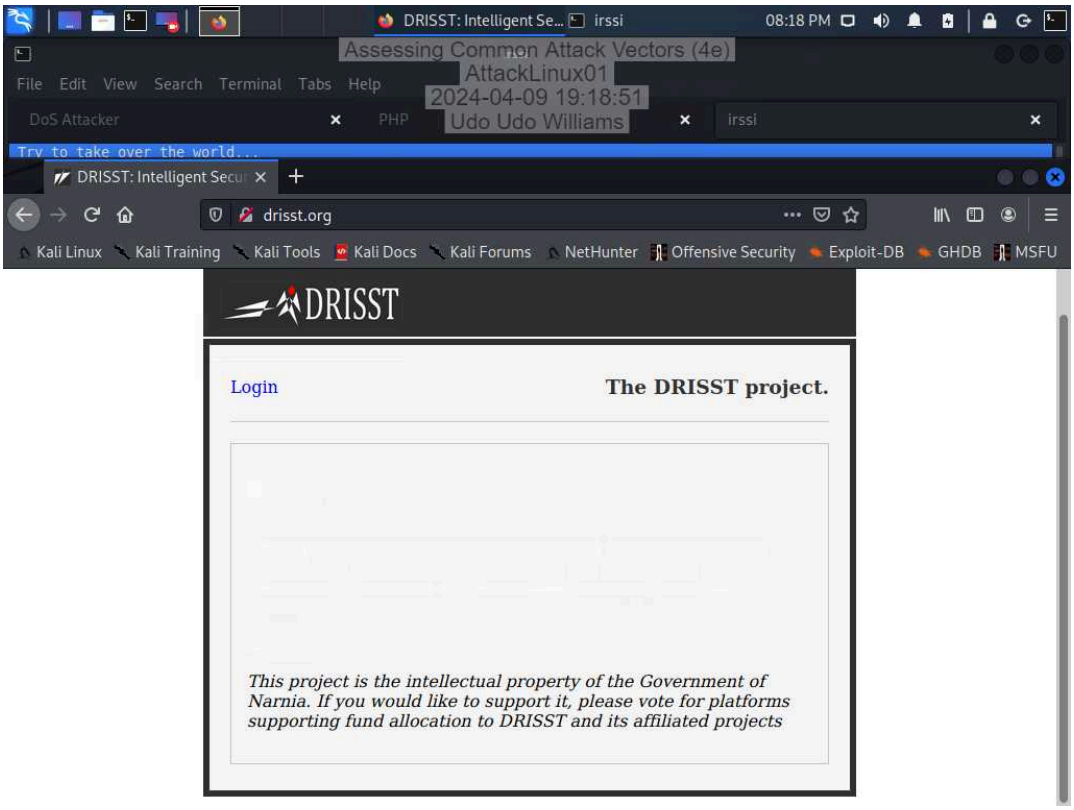
Section 2: Applied Learning

Part 1: Perform a Distributed Denial-of-Service Attack

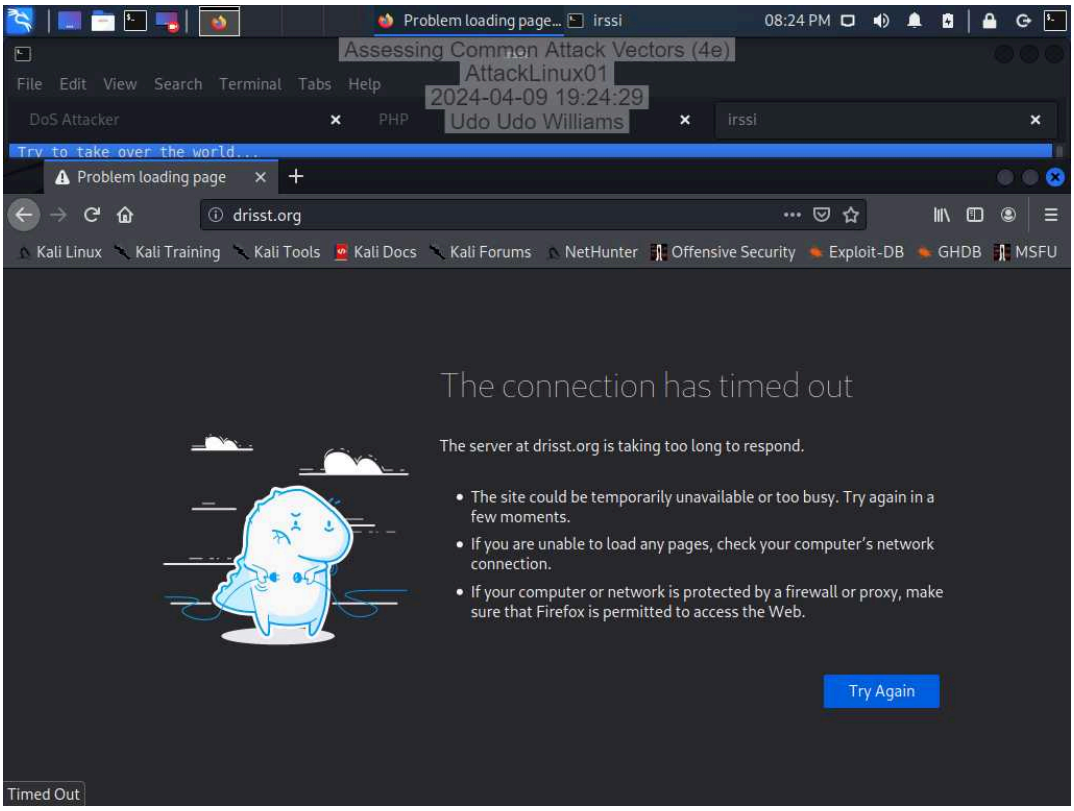
25. **Make a screen capture** showing the **newly recruited hosts**.



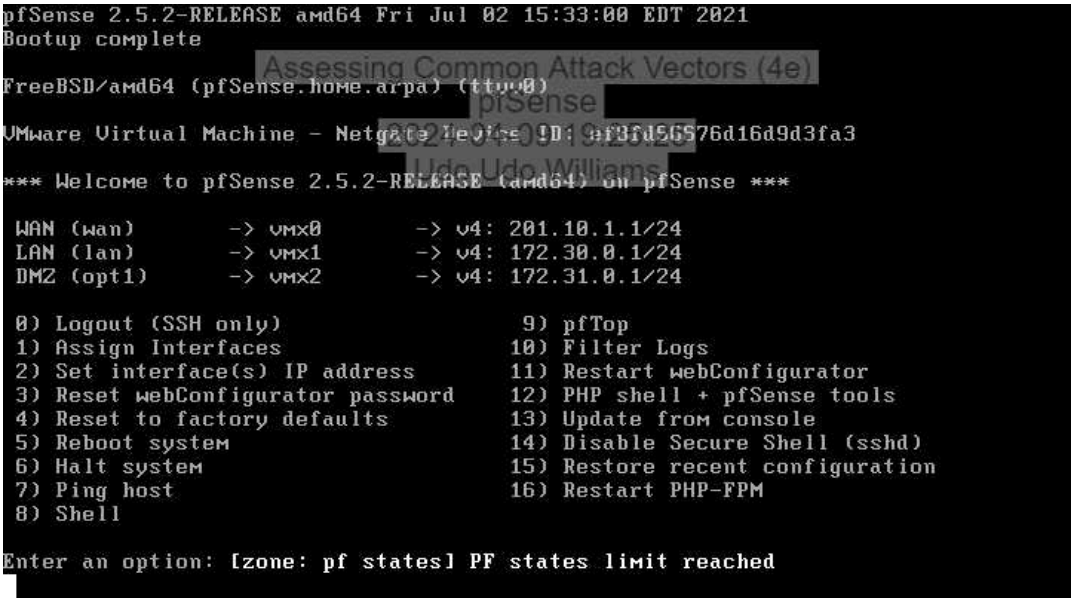
28. **Make a screen capture** showing the **drisst.org webpage**.



33. **Make a screen capture** showing the **failed connection to drisst.org**.

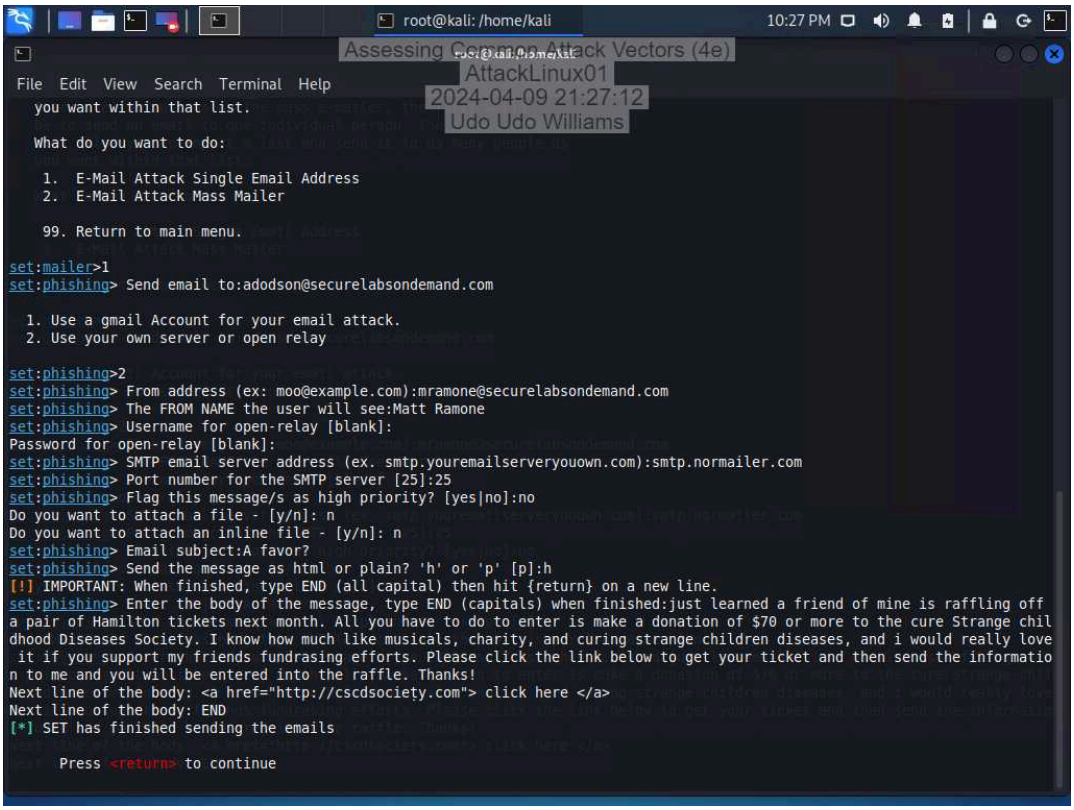


35. **Make a screen capture** showing the **"PF states limit reached" error message**.

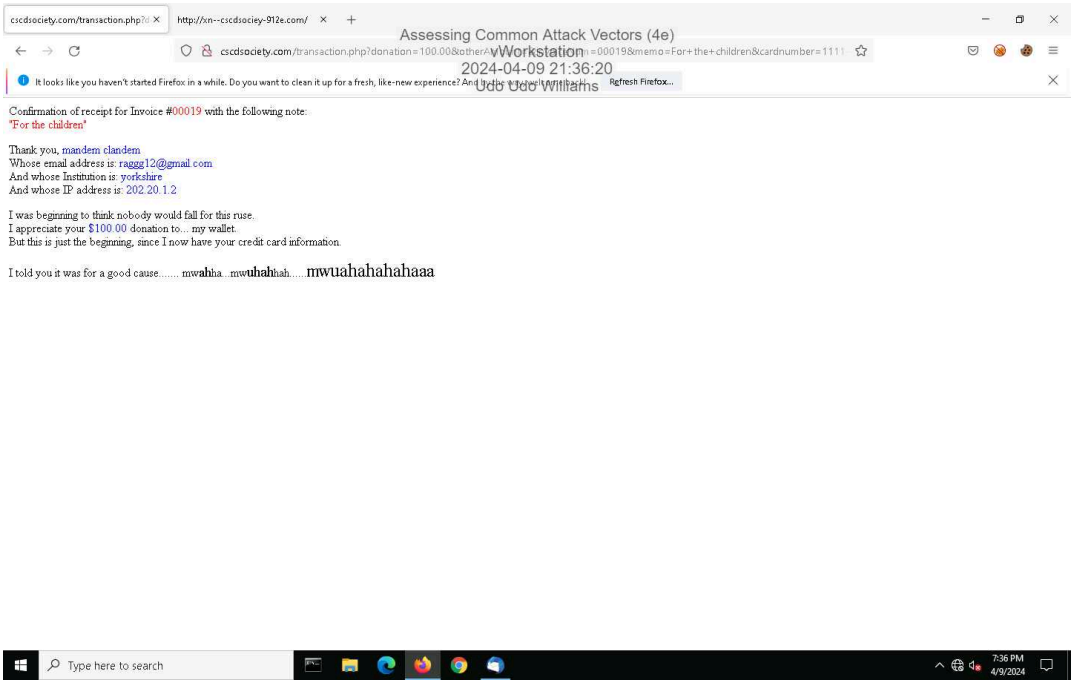


Part 2: Perform a Social Engineering Attack

24. **Make a screen capture** showing the **finished SET phishing email composition**.



36. **Make a screen capture** showing the **transaction.php page in the browser**.



Section 3: Challenge and Analysis

Part 1: Recommend Defensive Measures

Identify and **describe** at least two defensive measures that can be used against injection attacks. Be sure to cite your sources.

1. Filter Database Inputs Although input filtering alone cannot stop SQL injection attacks, filtering database input from websites and applications provides fundamental security to eliminate SQL injection vulnerabilities. Many attackers attempt to exploit extended URLs and special character handling to explore databases and execute commands to gain unauthorized access or exfiltrate and delete data.
sourceshttps://www.esecurityplanet.com/threats/how-to-prevent-sql-injection-attacks/#filter-database-inputs
https://www.w3schools.com/sql/sql_injection.asp
2. Restrict Database Code Input filtering is a good starting point, but attackers can find other ways to bypass inputs using zero-day vulnerabilities, credentials compromise, and more. Organizations can restrict the code available to a database to further control and limit the ability of attackers to exploit SQL injection vulnerabilities. Database managers should reduce functionality, use stored procedures, whitelist user inputs, and enforce prepared statements and parameterization. These tactics limit the database strictly to the capabilities needed for the task and prevent unexpected uses and exploits.

Identify and **describe** at least two defensive measures that can be used against malware attacks. Be sure to cite your sources.

Protect against malware Taking a layered approach with next-generation endpoint monitoring tools, including AMP for Endpoints, next-generation firewalls (NGFW), and an intrusion prevention system (IPS), will help you deploy security from the endpoint to email to the DNS layer.
Partition network Reduce the risk of outbreak exposure by isolating your network using network segmentation.
sources
https://www.cisco.com/c/en/us/products/security/malware-protection-best-practices-detection-prevention.html#~best-practices

Identify and **describe** at least two defensive measures that can be used against denial-of-service attacks. Be sure to cite your sources.

Attack surface reduction: Limiting attack surface exposure can help minimize the effect of a DDoS attack. Several methods for reducing this exposure include restricting traffic to specific locations, implementing a load balancer, and blocking communication from outdated or unused ports, protocols, and applications.Rate limiting: Rate limiting restricts the volume of network traffic over a specific time period, essentially preventing web servers from getting overwhelmed by requests from specific IP addresses. Rate limiting can be used to prevent DDoS attacks that use botnets to spam an endpoint with an abnormal amount of requests at once.
sourcehttps://www.cloudflare.com/learning/ddos/how-to-prevent-ddos-attacks/

Identify and **describe** at least two defensive measures that can be used against social engineering attacks. Be sure to cite your sources.

Install and maintain anti-virus software, firewalls, and email filters to reduce some of this traffic. (See Understanding Firewalls for Home and Small Office Use, Protecting Against Malicious Code, and Reducing Spam for more information.)
Be suspicious of unsolicited phone calls, visits, or email messages from individuals asking about employees or other internal information. If an unknown individual claims to be from a legitimate organization, try to verify his or her identity directly with the company.
https://www.cisa.gov/news-events/news/avoiding-social-engineering-and-phishing-attacks

Part 2: Research Additional Attack Vectors

Describe the additional attack vector you selected and **identify** at least two defensive measures that can be used against it. Be sure to cite your sources.

Defense Against Session HighjackingHTTPS: The use of HTTPS ensures that there is SSL/TLS encryption throughout the session traffic. Attackers will be unable to intercept the plaintext session ID, even if the victim's traffic was monitored. It is advised to use HSTS (HTTP Strict Transport Security) to guarantee complete encryption.
session Key: It is advised to regenerate session keys after their initial authentication. This renders the session ID extracted by attackers useless as the ID changes immediately after authentication.
source
https://www.globalsign.com/en/blog/session-hijacking-and-how-to-prevent-it