Comprehensive Vulnerability Analysis and Secure Network Design Report for OWASP Juice Shop

---

Executive Summary

This report provides a detailed analysis of vulnerabilities discovered in the OWASP Juice Shop application and proposes a comprehensive network design to mitigate these risks. The objective is to secure the application's environment, minimize attack surfaces, and safeguard sensitive data by implementing a layered, defense-in-depth architecture.

---

1. Vulnerabilities Discovered

1.1. Exposed Backup and Configuration Files

- Sensitive files (e.g., `.cer`, `.pem`, `.tgz`) are potentially exposed and may include server configurations, credentials, or cryptographic keys.
- **Impact**: If accessed, attackers can exploit credentials, modify server configurations, or compromise encryption protocols.

1.2. Missing Security Headers

- HTTP responses lack headers such as `X-Content-Type-Options`, `Strict-Transport-Security`, and `Content-Security-Policy`.
- **Impact**: This omission increases the risk of Cross-Site Scripting (XSS), clickjacking, and other injection-based attacks.

1.3. Uncommon and Leaky Headers

- Headers such as `x-recruiting` and `reporting-endpoints` unnecessarily expose internal operational details.
- **Impact**: Provides attackers with additional reconnaissance data for targeted exploitation.

1.4. Potentially Open Directories

- `/robots.txt` file references `/ftp`, which may contain sensitive files.
- **Impact**: If improperly secured, attackers could access confidential information through directory traversal.

1.5. Server Information Disclosure

- The application exposes server details such as "Cowboy" (HTTP server) in response headers.
- **Impact**: Increases susceptibility to exploits targeting known vulnerabilities in specific server software.

1.6. DNS and Service Exposure

- DNS records reveal associated IPs (e.g., AWS-hosted IPs) and load-balancing configurations.
- **Impact**: Attackers could use this information for Distributed Denial of Service (DDoS) or further enumeration.

1.7. Tool Configuration Failures

- Sublist3r and Gobuster scans were incomplete, leaving potentially hidden vulnerabilities and attack vectors undiscovered.
- **Impact**: Incomplete enumeration increases the likelihood of undetected attack surfaces.

---

2. Potential Impacts of Vulnerabilities

- **Sensitive Information Disclosure**: Exposed files and misconfigured directories can lead to unauthorized access to credentials, private keys, or sensitive files.
- **Cross-Site Scripting (XSS)**: Missing security headers make users susceptible to malicious scripts injected into the web application.
- **Service Exploitation**: Disclosed server information and exposed services increase the likelihood of attackers exploiting known vulnerabilities.
- **Increased Attack Surface**: Unmitigated issues expand the range of possible attack vectors, putting the application's integrity at risk.

---

3. Recommendations for Vulnerability Mitigation

3.1. Secure Files and Directories

- Restrict access to sensitive files such as `.pem` and `.tgz`.
- Remove unnecessary files from public directories.
- Implement `.htaccess` or server rules to block unauthorized file access.

3.2. Implement Security Headers

- Add critical security headers, such as:
  - `X-Content-Type-Options: nosniff`
  - `Strict-Transport-Security: max-age=31536000; includeSubDomains`
  - `Content-Security-Policy: default-src 'self'`
- Use tools like securityheaders.com to validate header configurations.

3.3. Protect Sensitive Paths

- Restrict `/ftp` and other referenced directories from `/robots.txt`.
- Disable directory listing and configure proper permissions.

3.4. Obfuscate Server Information

- Disable server version banners in HTTP response headers.
- Use reverse proxies like Nginx to mask backend server details.

3.5. Enhance DNS Security

- Implement DNSSEC to prevent DNS spoofing.
- Monitor DNS records and associated IPs for unauthorized changes.

3.6. Improve Vulnerability Scanning Tools

- Fix Sublist3r and Gobuster configuration issues to ensure thorough enumeration.
- Use updated and comprehensive wordlists for directory and subdomain discovery.

3.7. Conduct Regular Security Audits

- Schedule vulnerability scans with tools like Nikto, Nessus, and Burp Suite.
- Regularly review logs for unauthorized access attempts.

---

4. Secure Network Design Proposal

Objective

To design a secure network architecture for OWASP Juice Shop, minimizing the attack surface while ensuring the application's availability, confidentiality, and integrity.

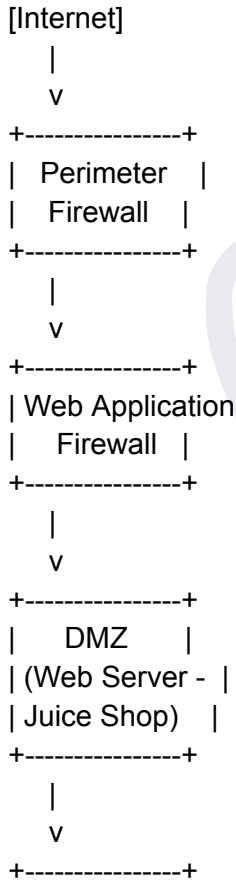
4.1. Network Architecture Overview

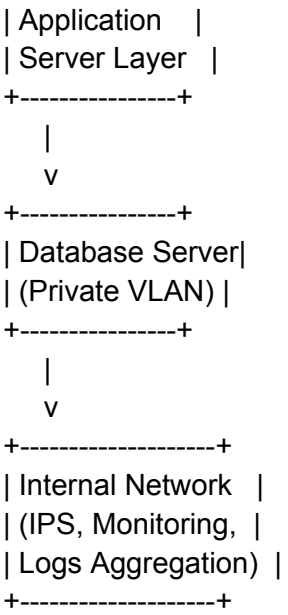
The architecture follows a **layered defense-in-depth model** and incorporates segmentation to isolate critical components.

Key Components

1. **Perimeter Firewall**
2. **Web Application Firewall (WAF)**
3. **DMZ (Demilitarized Zone)**
4. **Application Server Layer**
5. **Database Server**
6. **Internal Network (Monitoring and IPS)**
7. **Intrusion Prevention System (IPS)**
8. **Centralized Logging System**

4.2. Network Design Diagram





4.3. Security Control Descriptions

1. **Perimeter Firewall**:
    - Blocks unauthorized traffic at the network edge.
    - Filters based on IPs, protocols, and port numbers.
2. **Web Application Firewall (WAF)**:
    - Detects and blocks malicious requests targeting the Juice Shop application.
    - Protects against OWASP Top 10 threats, including SQL injection and XSS.
3. **DMZ (Demilitarized Zone)**:
    - Hosts the public-facing web server, isolating it from the internal network.
    - Reduces lateral movement risks.
4. **Application Server Layer**:
    - Processes requests from the web server.
    - Accessible only from the DMZ.
5. **Database Server**:
    - Stores sensitive application data.
    - Restricted access via private VLAN and encrypted communication channels.
6. **Intrusion Prevention System (IPS)**:
    - Monitors traffic for anomalies and blocks intrusions in real time.

7. **Monitoring and Logging System**:
    - Aggregates logs from all network components.
    - Enhances threat detection and supports incident response.

4.4. Risk Mitigation Strategies

- **Segmentation**: Strictly isolate the web server, application server, and database layers.
- **Access Control**: Enforce the principle of least privilege using role-based permissions.
- **Encryption**: Secure all communications using TLS (HTTPS) and encrypt sensitive data at rest.
- **Audits and Updates**: Regularly patch systems and conduct vulnerability assessments.

---

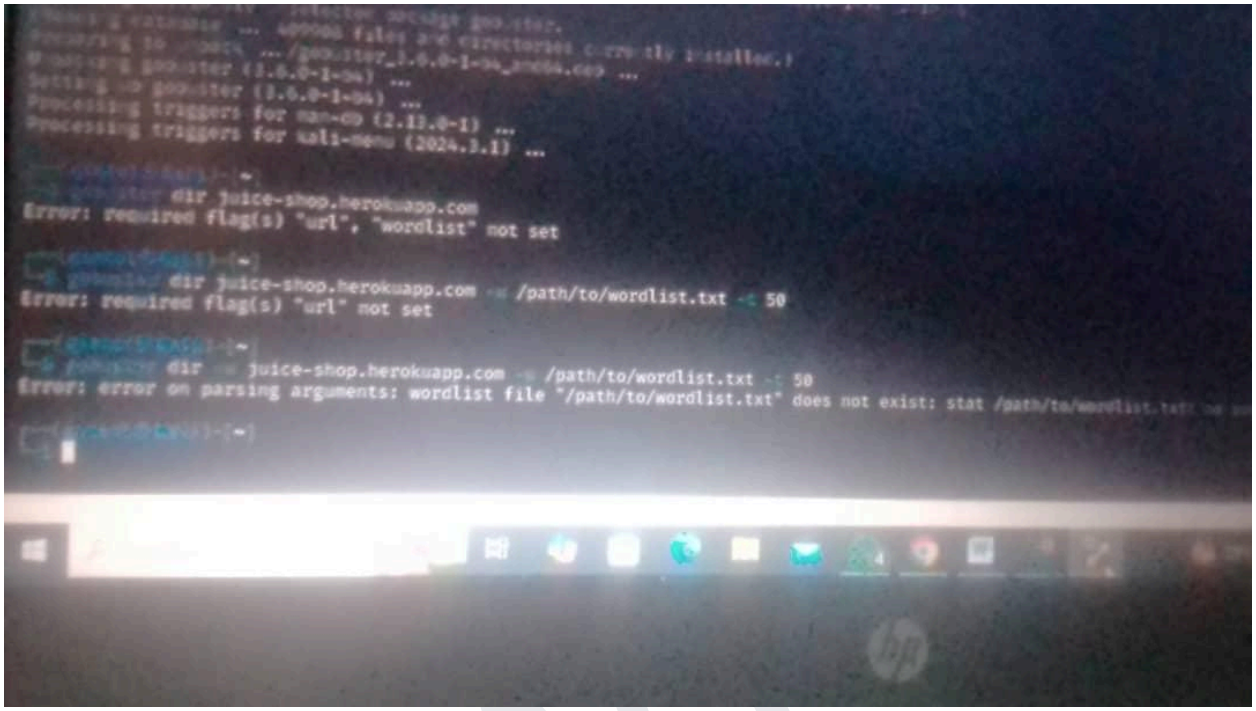
5. Conclusion

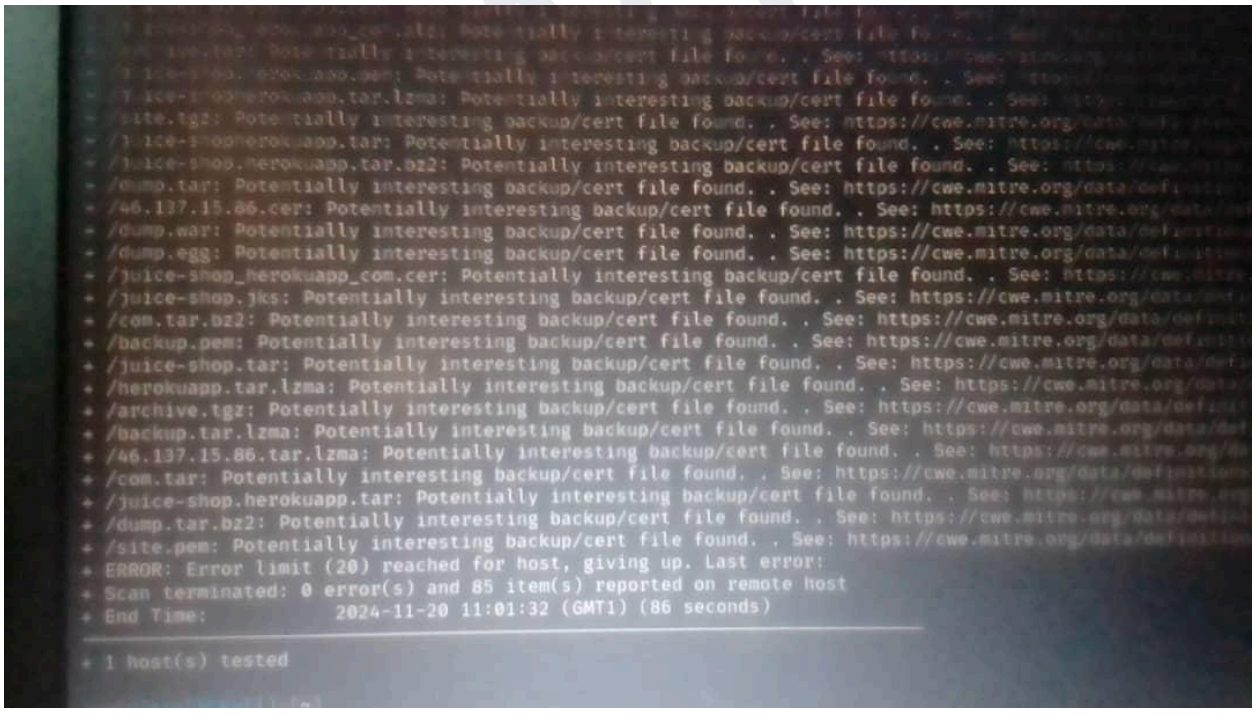
This report identifies significant vulnerabilities in OWASP Juice Shop and outlines a secure network architecture to address them. The proposed layered design ensures robust security through segmentation, firewalls, intrusion prevention, and comprehensive monitoring.

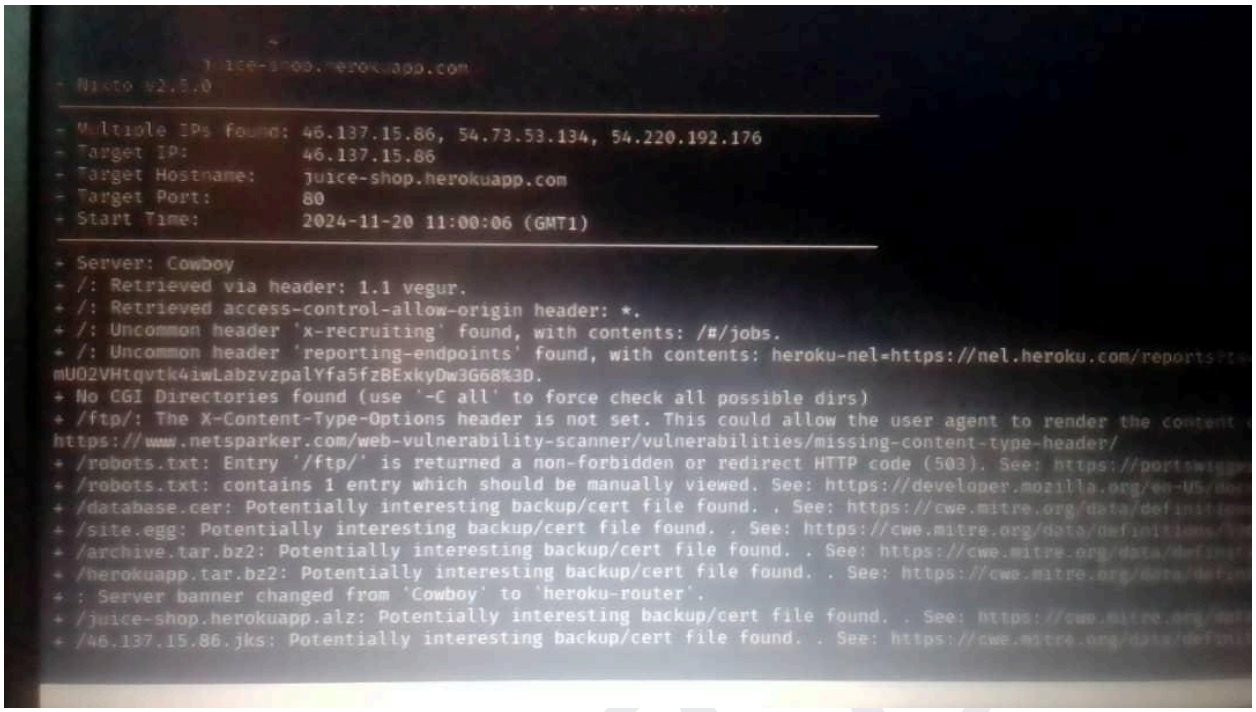
Immediate actions include securing exposed files, implementing missing security headers, and fixing scanning tool configurations. Long-term measures, such as regular audits and security updates, will sustain the application's defenses against emerging threats.

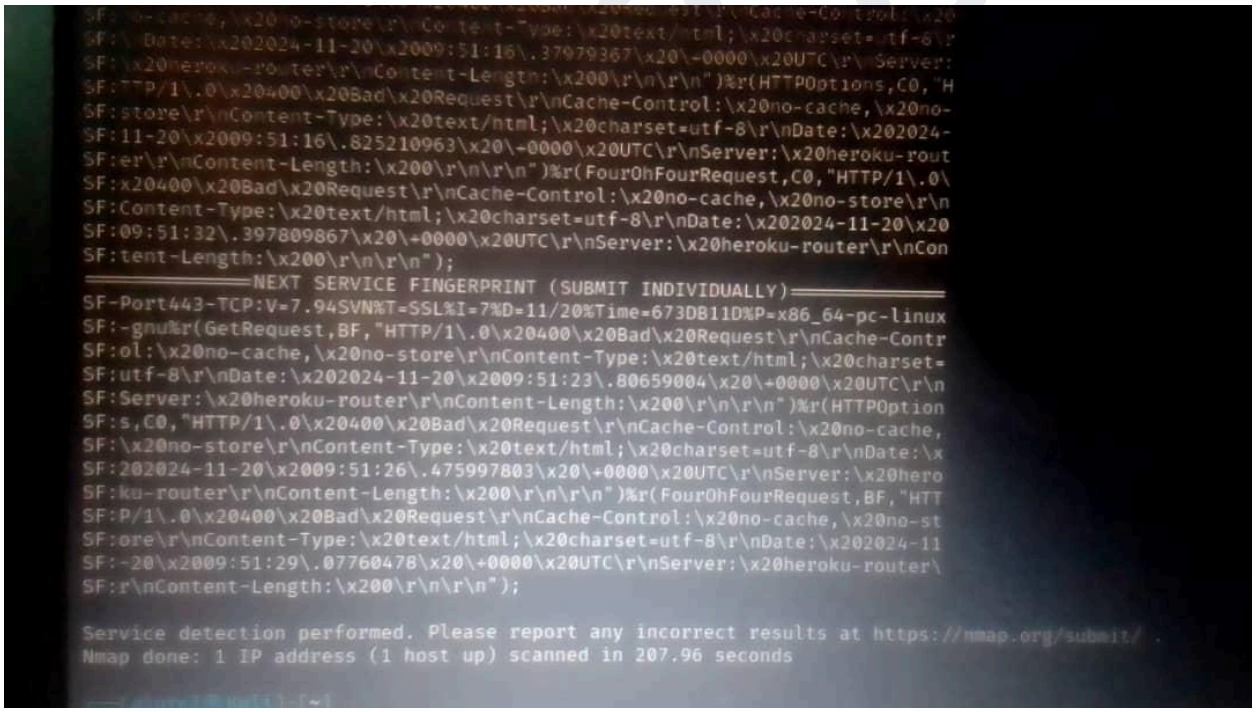
This approach provides a scalable, resilient foundation for maintaining Juice Shop's security while allowing for future growth and adaptation to evolving risks.

COMMAND LINE INTERACTION ON KALI LINUX

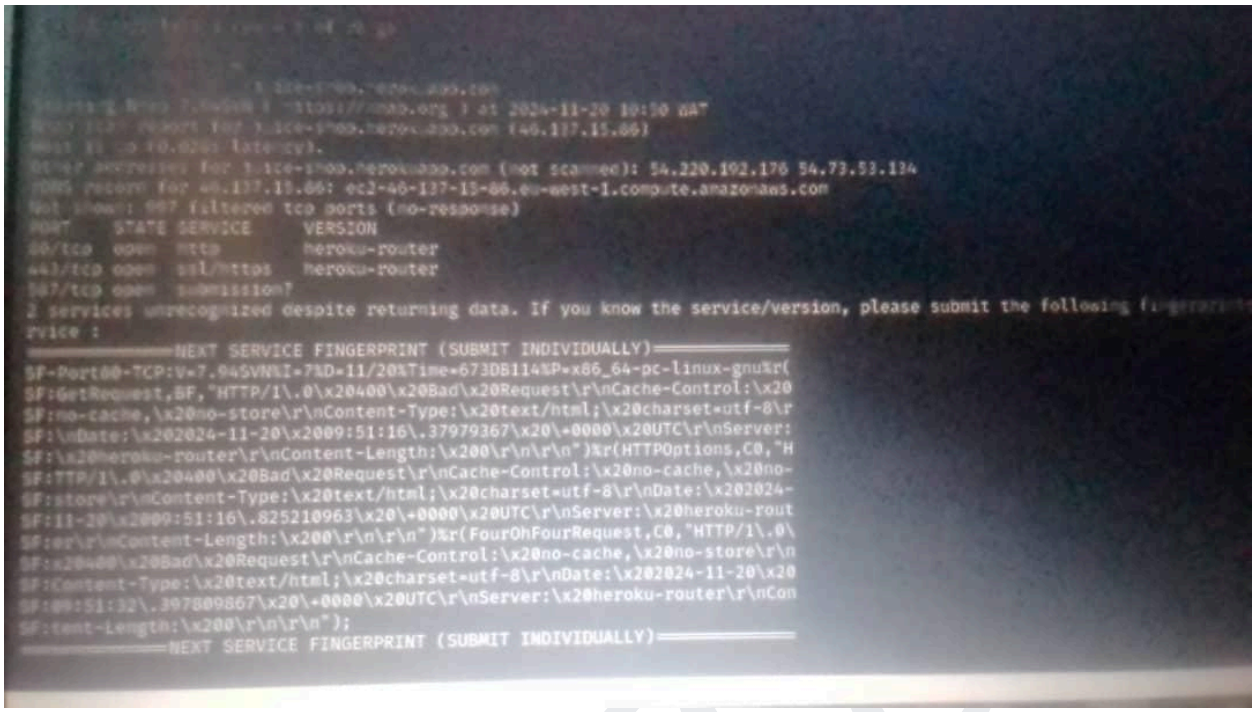


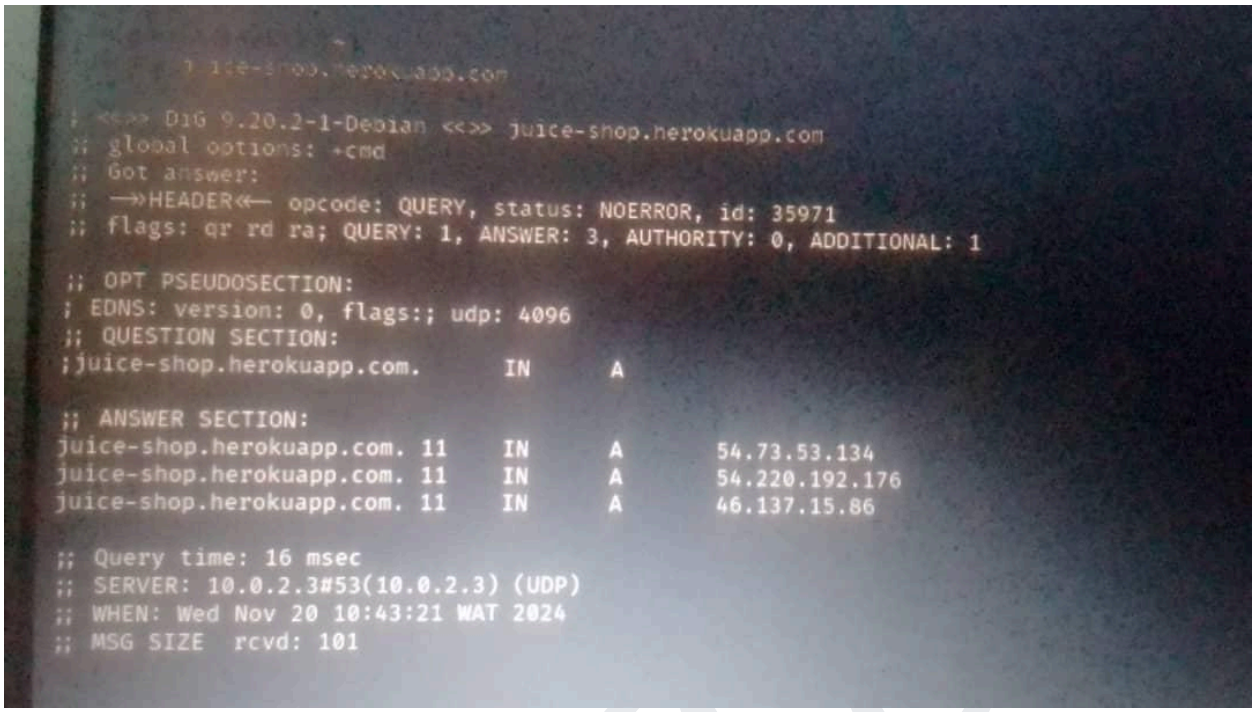



+ Nikto v2.5.0
+ Multiple IPs found: 46.137.15.86, 54.73.53.134, 54.220.192.176
+ Target IP: 46.137.15.86
+ Target Hostname: juice-shop.herokuapp.com
+ Target Port: 80
+ Start Time: 2024-11-20 11:00:06 (GMT1)
+ Server: Cowboy
+ /: Retrieved via header: 1.1 vegur.
+ /: Retrieved access-control-allow-origin header: *.
+ /: Uncommon header 'x-recruiting' found, with contents: /#/jobs.
+ /: Uncommon header 'reporting-endpoints' found, with contents: heroku-nel=https://nel.heroku.com/reports?
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /ftp/: The X-Content-Type-Options header is not set. This could allow the user agent to render the content
https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ /robots.txt: Entry '/ftp/' is returned a non-forbidden or redirect HTTP code (503). See: https://portswigger
+ /robots.txt: contains 1 entry which should be manually viewed. See: https://developer.mozilla.org/en-US/doc
+ /database.cer: Potentially interesting backup/cert file found. . See: https://cwe.mitre.org/data/definitions
+ /site.egg: Potentially interesting backup/cert file found. . See: https://cwe.mitre.org/data/definitions/
+ /archive.tar.bz2: Potentially interesting backup/cert file found. . See: https://cwe.mitre.org/data/defini
+ /herokuapp.tar.bz2: Potentially interesting backup/cert file found. . See: https://cwe.mitre.org/data/defin
+ : Server banner changed from 'Cowboy' to 'heroku-router'.
+ /juice-shop.herokuapp.alz: Potentially interesting backup/cert file found. . See: https://cwe.mitre.org/data
+ /46.137.15.86.jks: Potentially interesting backup/cert file found. . See: https://cwe.mitre.org/data/defini


SF:Date:\x202024-11-20\x2009:51:16\.37979367\x20\+0000\x20UTC\r\nServer:
SF:\x20heroku-router\r\nContent-Length:\x200\r\n\r\n")%r(HTTPOptions,C0,"H
SF:TTP/1\.0\x20400\x20Bad\x20Request\r\nCache-Control:\x20no-cache,\x20no-
SF:store\r\nContent-Type:\x20text/html;\x20charset=utf-8\r\nDate:\x202024-
SF:11-20\x2009:51:16\.825210963\x20\+0000\x20UTC\r\nServer:\x20heroku-rout
SF:er\r\nContent-Length:\x200\r\n\r\n")%r(FourOhFourRequest,C0,"HTTP/1\.0\
SF:x20400\x20Bad\x20Request\r\nCache-Control:\x20no-cache,\x20no-store\r\n
SF:Content-Type:\x20text/html;\x20charset=utf-8\r\nDate:\x202024-11-20\x20
SF:09:51:32\.397809867\x20\+0000\x20UTC\r\nServer:\x20heroku-router\r\nCon
SF:tent-Length:\x200\r\n\r\n");
==============NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)==============
SF-Port443-TCP:V=7.94SVN%T=SSL%I=7%D=11/20%Time=673DB11D%P=x86_64-pc-linux
SF:-gnu%r(GetRequest,BF,"HTTP/1\.0\x20400\x20Bad\x20Request\r\nCache-Contr
SF:ol:\x20no-cache,\x20no-store\r\nContent-Type:\x20text/html;\x20charset=
SF:utf-8\r\nDate:\x202024-11-20\x2009:51:23\.80659004\x20\+0000\x20UTC\r\n
SF:Server:\x20heroku-router\r\nContent-Length:\x200\r\n\r\n")%r(HTTPOption
SF:s,C0,"HTTP/1\.0\x20400\x20Bad\x20Request\r\nCache-Control:\x20no-cache,
SF:\x20no-store\r\nContent-Type:\x20text/html;\x20charset=utf-8\r\nDate:\x
SF:202024-11-20\x2009:51:26\.475997803\x20\+0000\x20UTC\r\nServer:\x20hero
SF:ku-router\r\nContent-Length:\x200\r\n\r\n")%r(FourOhFourRequest,BF,"HTT
SF:P/1\.0\x20400\x20Bad\x20Request\r\nCache-Control:\x20no-cache,\x20no-st
SF:ore\r\nContent-Type:\x20text/html;\x20charset=utf-8\r\nDate:\x202024-11
SF:-20\x2009:51:29\.07760478\x20\+0000\x20UTC\r\nServer:\x20heroku-router\
SF:r\nContent-Length:\x200\r\n\r\n");

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 207.96 seconds


PORT STATE SERVICE VERSION
80/tcp open http heroku-router
443/tcp open ssl/https heroku-router
587/tcp open submission?
2 services unrecognized despite returning data. If you know the service/version, please submit the following
NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)
SF-Port80-TCP:V=7.94SVN%I=7%D=11/20%Time=673DB114%P=x86_64-pc-linux-gnu%r(
SF:GetRequest,BF,"HTTP/1\.0\x20400\x20Bad\x20Request\r\nCache-Control:\x20
SF:no-cache,\x20no-store\r\nContent-Type:\x20text/html;\x20charset=utf-8\r
SF:\nDate:\x202024-11-20\x2009:51:16\.37979367\x20\+0000\x20UTC\r\nServer:
SF:\x20heroku-router\r\nContent-Length:\x200\r\n\r\n")%r(HTTPOptions,C0,"H
SF:TTP/1\.0\x20400\x20Bad\x20Request\r\nCache-Control:\x20no-cache,\x20no-
SF:store\r\nContent-Type:\x20text/html;\x20charset=utf-8\r\nDate:\x202024-
SF:11-20\x2009:51:16\.825210963\x20\+0000\x20UTC\r\nServer:\x20heroku-rout
SF:er\r\nContent-Length:\x200\r\n\r\n")%r(FourOhFourRequest,C0,"HTTP/1\.0\
SF:x20400\x20Bad\x20Request\r\nCache-Control:\x20no-cache,\x20no-store\r\n
SF:Content-Type:\x20text/html;\x20charset=utf-8\r\nDate:\x202024-11-20\x20
SF:09:51:32\.397809867\x20\+0000\x20UTC\r\nServer:\x20heroku-router\r\nCon
SF:tent-Length:\x200\r\n\r\n");
NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)

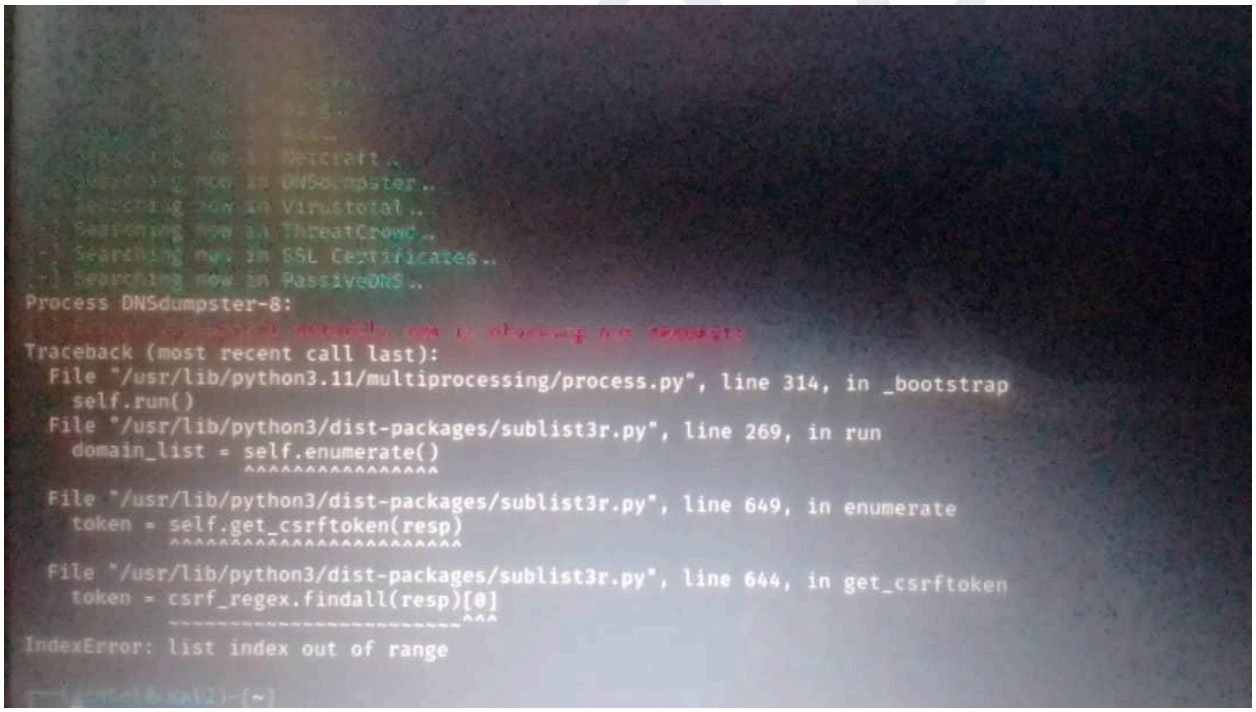


```
; <<>> DiG 9.20.2-1-Debian <<>> juice-shop.herokuapp.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 35971
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;juice-shop.herokuapp.com.      IN      A

;; ANSWER SECTION:
juice-shop.herokuapp.com. 11    IN      A       54.73.53.134
juice-shop.herokuapp.com. 11    IN      A       54.220.192.176
juice-shop.herokuapp.com. 11    IN      A       46.137.15.86

;; Query time: 16 msec
;; SERVER: 10.0.2.3#53(10.0.2.3) (UDP)
;; WHEN: Wed Nov 20 10:43:21 WAT 2024
;; MSG SIZE  rcvd: 101
```



```
Process DNSdumpster-8:
Traceback (most recent call last):
  File "/usr/lib/python3.11/multiprocessing/process.py", line 314, in _bootstrap
    self.run()
  File "/usr/lib/python3/dist-packages/sublist3r.py", line 269, in run
    domain_list = self.enumerate()
                  ^^^^^^^^^^^^^^^^
  File "/usr/lib/python3/dist-packages/sublist3r.py", line 649, in enumerate
    token = self.get_csrftoken(resp)
            ^^^^^^^^^^^^^^^^^^^^^^^^
  File "/usr/lib/python3/dist-packages/sublist3r.py", line 644, in get_csrftoken
    token = csrf_regex.findall(resp)[0]
            ~~~~~~~~~~~~~~~~~~~~~~~~^^^
IndexError: list index out of range
```