Sri Lanka Institute of Information Technology

Bug Bounty Report 07

WS Assignment

IE2062 – Web Security

IT23159730

W.H.M.S.R.Bandara

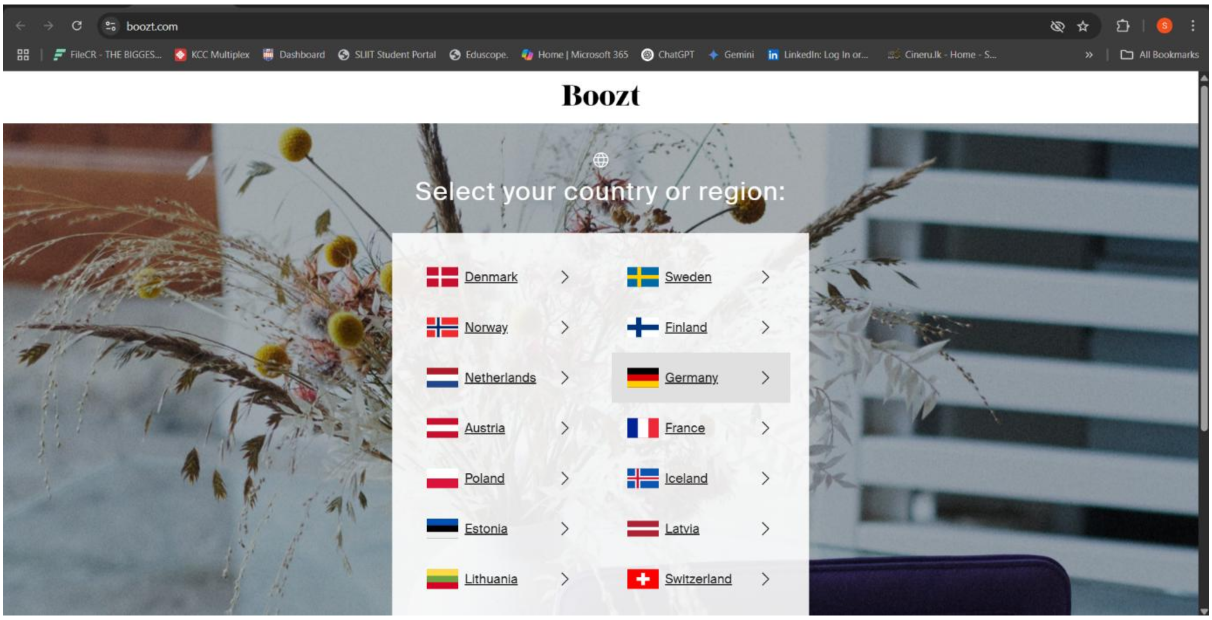
Vulnerability Title

Missing Anti-Clickjacking Header on www.boozt.com

Vulnerability Description

Clickjacking, also known as a UI redress attack, is a weakness in which an attacker tricks a user into clicking a link or button on a web page by overlaying it with a hidden or transparent iframe. This can lead to an unexpected action, like shopping, modification of account settings, or revelation of sensitive information, without the knowledge of the user. Anti-clickjacking headers, either X-Frame-Options or a Content Security Policy (CSP) with the frame-ancestors directive, prevent a webpage from being framed in an iframe on an untrusted web page, which mitigates this vulnerability.

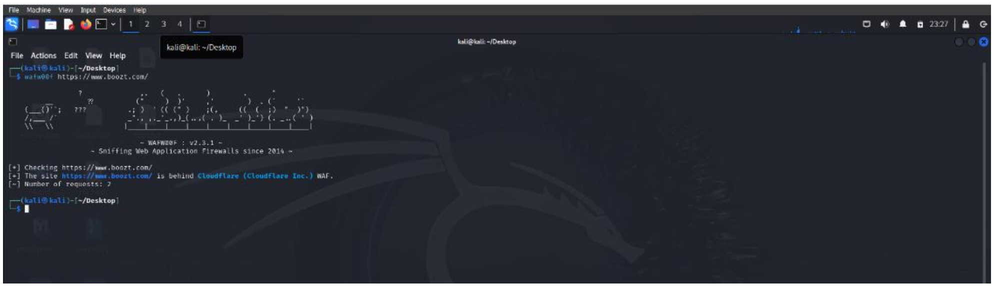
In a security test of www.boozt.com , an automated scan using OWASP ZAP (2.16.1) indicated that the X-Frame-Options header is not set, and the Content Security Policy (CSP) does not include a frame-ancestors directive (CWE-1021). This vulnerability is discovered on the URL https://www.boozt.com/nl/nl/dameskleding/sport/bekijk-alles . The absence of anti-clickjacking protection exposes the website to increased risks of clickjacking attacks, which can lead to the execution of unauthorized actions on behalf of the user, especially on an e-commerce site where users perform sensitive actions like ordering a product or modifying account details. The severity of this vulnerability is medium because it can enable clickjacking attacks. The ZAP scan also indicated other issues, such as the absence of a CSP header and cookies without the HttpOnly flag, which are part of the overall risk, but this report is focused on the absence of an anti-clickjacking header.



Other scans

Firewall Detection.

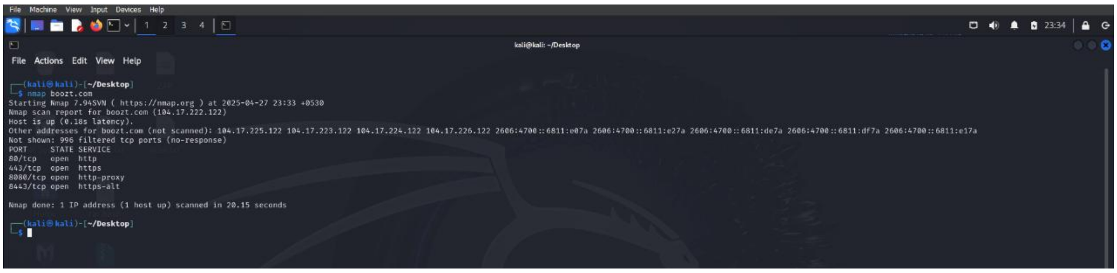
Before the automated scan with OWASP ZAP (2.16.0), Identify the Firewall by using Wafw00f Tool.



```
┌──(kali㉿kali)-[~/Desktop]
└─$ wafw00f https://www.boozt.com/

                ~ WAFW00F : v2.3.1 ~
    ~ Sniffing Web Application Firewalls since 2014 ~

[*] Checking https://www.boozt.com/
[+] The site https://www.boozt.com/ is behind Cloudflare (Cloudflare Inc.) WAF.
[~] Number of requests: 2

┌──(kali㉿kali)-[~/Desktop]
└─$
```

Open Ports Detection

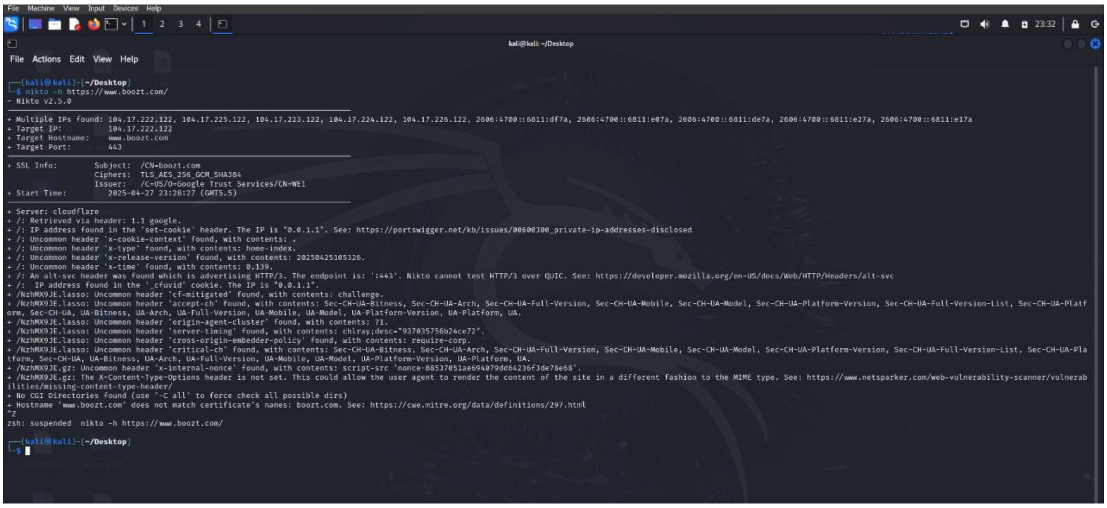
Done the nmap scan for detect open ports.



```
┌──(kali㉿kali)-[~/Desktop]
└─$ nmap boozt.com
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-27 23:33 +0530
Nmap scan report for boozt.com (104.17.222.122)
Host is up (0.18s latency).
Other addresses for boozt.com (not scanned): 104.17.225.122 104.17.223.122 104.17.224.122 104.17.226.122 2606:4700::6811:e07a 2606:4700::6811:e27a 2606:4700::6811:de7a 2606:4700::6811:df7a 2606:4700::6811:e17a
Not shown: 996 filtered tcp ports (no-response)
PORT     STATE SERVICE
80/tcp   open  http
443/tcp  open  https
8080/tcp open  http-proxy
8443/tcp open  https-alt

Nmap done: 1 IP address (1 host up) scanned in 20.15 seconds

┌──(kali㉿kali)-[~/Desktop]
└─$
```

Scanning for common Issues

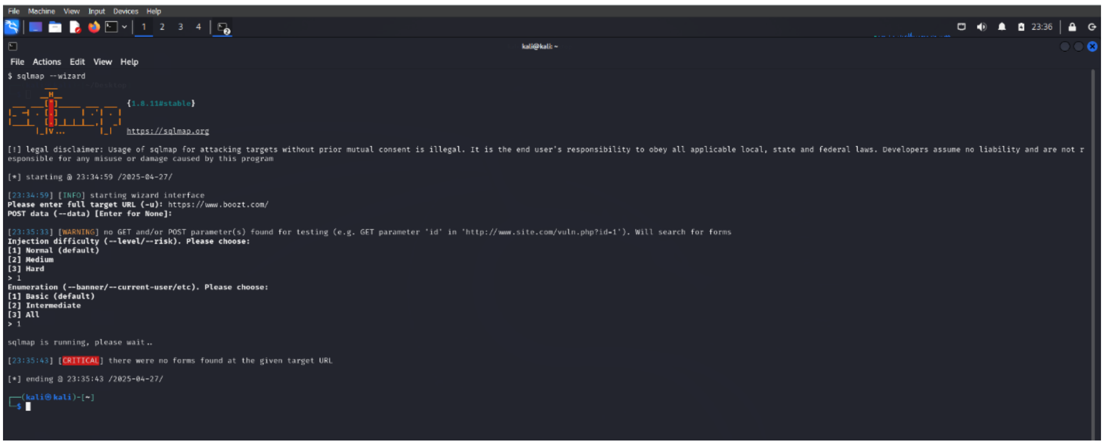
These scans done by with Nikto.



```
┌──(kali㉿kali)-[~/Desktop]
└─$ nikto -h https://www.boozt.com/
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Multiple IPs found: 104.17.222.122, 104.17.225.122, 104.17.223.122, 104.17.224.122, 104.17.226.122, 2606:4700::6811:df7a, 2606:4700::6811:e07a, 2606:4700::6811:de7a, 2606:4700::6811:e27a, 2606:4700::6811:e17a
+ Target IP:          104.17.222.122
+ Target Hostname:    www.boozt.com
+ Target Port:        443
---------------------------------------------------------------------------
+ SSL Info:        Subject:  /CN=boozt.com
                   Ciphers:  TLS_AES_256_GCM_SHA384
                   Issuer:   /C=US/O=Google Trust Services/CN=WE1
+ Start Time:         2025-04-27 23:28:27 (GMT5.5)
---------------------------------------------------------------------------
+ Server: cloudflare
+ /: Retrieved via header: 1.1 google.
+ /: IP address found in the 'set-cookie' header. The IP is "0.0.1.1". See: https://portswigger.net/kb/issues/00600300_private-ip-addresses-disclosed
+ /: Uncommon header 'x-cookie-context' found, with contents: .
+ /: Uncommon header 'x-type' found, with contents: home-index.
+ /: Uncommon header 'x-release-version' found, with contents: 20250425105326.
+ /: Uncommon header 'x-time' found, with contents: 0.139.
+ /: An alt-svc header was found which is advertising HTTP/3. The endpoint is: ':443'. Nikto cannot test HTTP/3 over QUIC. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/alt-svc
+ /:  IP address found in the '_cfuvid' cookie. The IP is "0.0.1.1".
+ /NzhMX9JE.lasso: Uncommon header 'cf-mitigated' found, with contents: challenge.
+ /NzhMX9JE.lasso: Uncommon header 'accept-ch' found, with contents: Sec-CH-UA-Bitness, Sec-CH-UA-Arch, Sec-CH-UA-Full-Version, Sec-CH-UA-Mobile, Sec-CH-UA-Model, Sec-CH-UA-Platform-Version, Sec-CH-UA-Full-Version-List, Sec-CH-UA-Platf
orm, Sec-CH-UA, UA-Bitness, UA-Arch, UA-Full-Version, UA-Mobile, UA-Model, UA-Platform-Version, UA-Platform, UA.
+ /NzhMX9JE.lasso: Uncommon header 'origin-agent-cluster' found, with contents: ?1.
+ /NzhMX9JE.lasso: Uncommon header 'server-timing' found, with contents: chlray;desc="937035756b24ce72".
+ /NzhMX9JE.lasso: Uncommon header 'cross-origin-embedder-policy' found, with contents: require-corp.
+ /NzhMX9JE.lasso: Uncommon header 'critical-ch' found, with contents: Sec-CH-UA-Bitness, Sec-CH-UA-Arch, Sec-CH-UA-Full-Version, Sec-CH-UA-Mobile, Sec-CH-UA-Model, Sec-CH-UA-Platform-Version, Sec-CH-UA-Full-Version-List, Sec-CH-UA-Pla
tform, Sec-CH-UA, UA-Bitness, UA-Arch, UA-Full-Version, UA-Mobile, UA-Model, UA-Platform-Version, UA-Platform, UA.
+ /NzhMX9JE.gz: Uncommon header 'x-internal-nonce' found, with contents: script-src 'nonce-88537051ae694079dd64236f3de78e68'.
+ /NzhMX9JE.gz: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerab
ilities/missing-content-type-header/
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Hostname 'www.boozt.com' does not match certificate's names: boozt.com. See: https://cwe.mitre.org/data/definitions/297.html
^Z
zsh: suspended  nikto -h https://www.boozt.com/

┌──(kali㉿kali)-[~/Desktop]
└─$
```

Detect and exploit SQL injection flaws

Detect and exploit SQL injection flaws


File Machine View Input Devices Help
1 2 3 4
23:36
kali@kali: ~
File Actions Edit View Help

```
$ sqlmap --wizard
        ___
       __H__
 ___ ___[']_____ ___ ___  {1.8.11#stable}
|_ -| . [.]     | .'| . |
|___|_  [.]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not r
esponsible for any misuse or damage caused by this program

[*] starting @ 23:34:59 /2025-04-27/

[23:34:59] [INFO] starting wizard interface
Please enter full target URL (-u): https://www.boozt.com/
POST data (--data) [Enter for None]:

[23:35:33] [WARNING] no GET and/or POST parameter(s) found for testing (e.g. GET parameter 'id' in 'http://www.site.com/vuln.php?id=1'). Will search for forms
Injection difficulty (--level/--risk). Please choose:
[1] Normal (default)
[2] Medium
[3] Hard
> 1
Enumeration (--banner/--current-user/etc). Please choose:
[1] Basic (default)
[2] Intermediate
[3] All
> 1

sqlmap is running, please wait..

[23:35:43] [CRITICAL] there were no forms found at the given target URL

[*] ending @ 23:35:43 /2025-04-27/

┌──(kali㉿kali)-[~]
└─$ 
```

Affected Components

- **Domain:** www.boozt.com
- **Endpoint:** /nl/nl/dameskleding/sport/bekijk-alles
- **Vulnerability:** Missing X-Frame-Options or frame-ancestors directive
- **Risk Level:** Medium
- **Confidence:** Medium
- **Detection Method:** Passive Scan

Impact Assessment

The absence of an anti-clickjacking header could lead to the following risks:

- **Clickjacking Attacks:** Attackers can frame the Boozt website in an evil iframe, tricking users into performing unintended actions such as adding products to their cart, making purchases, or modifying account settings without their knowledge.
- **Data Exposure:** Clickjacking can lead to exposure of sensitive user data, e.g., session cookies or personal details, when the attack is combined with other vulnerabilities (e.g., XSS, more likely due to the missing CSP header identified by ZAP).
- **Financial Loss:** Clickjacking on an online shopping website can lead to unauthorized purchases or account changes, leading to direct financial loss to users or the website.
- **Business Impact:** The security vulnerability may cause financial loss, legal consequences, and loss of customer trust, which is necessary for an e-commerce site like Boozt.
- **Reputation Risk:** Vulnerability may lead to negative publicity and loss of user trust, impacting the site's reputation in the competitive e-commerce world.
- **Higher Risk:** Absence of CSP header and other vulnerabilities found by ZAP (e.g., cookies without HttpOnly flag) increase the overall exposure, which makes clickjacking more severe when combined with XSS or session hijacking.

Steps to Reproduce

1. Configure OWASP ZAP

- Launch OWASP ZAP and set the target https://www.boozt.com/ .
- Add the site to the context and ensure it’s in scope.

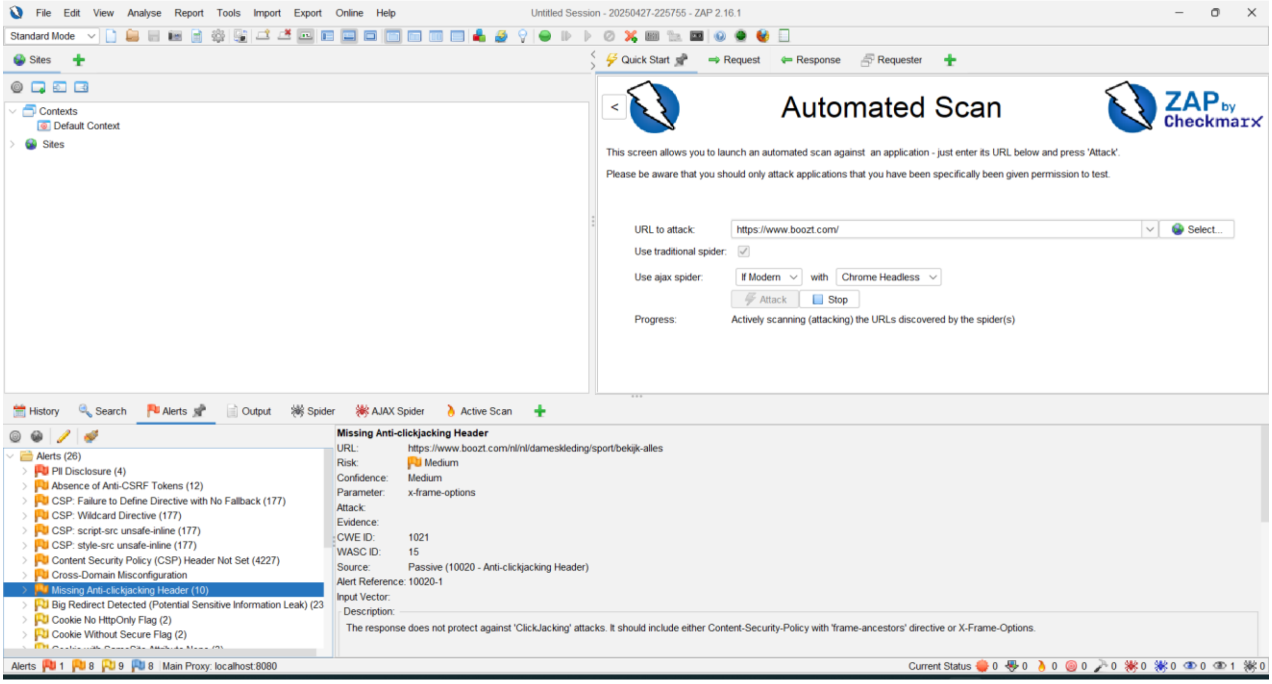
2. Perform an Automated Scan

    - Spider the site with the AJAX Spider and detect dynamic content.
    - Start an automated scan by clicking the "Attack" button in the Automated Scan window.
    - Wait until the scan finishes.

3. Analyze Alerts

    - Alert: "Missing Anti-clickjacking Header" (CWE-1021).
    - URL: https://www.boozt.com/nl/nl/dameskleding/sport/bekijk-alles .
    - Risk: Medium, Confidence: Medium.
    - Source: Passive scan (10020 - Anti-clickjacking Header).

Proof of Concept Screenshot



This screen allows you to launch an automated scan against an application - just enter its URL below and press 'Attack'.

Please be aware that you should only attack applications that you have been specifically been given permission to test.

URL to attack: https://www.boozt.com/
Use traditional spider: ☑
Use ajax spider: If Modern with Chrome Headless
Progress: Actively scanning (attacking) the URLs discovered by the spider(s)

Alerts (26)
- PII Disclosure (4)
- Absence of Anti-CSRF Tokens (12)
- CSP: Failure to Define Directive with No Fallback (177)
- CSP: Wildcard Directive (177)
- CSP: script-src unsafe-inline (177)
- CSP: style-src unsafe-inline (177)
- Content Security Policy (CSP) Header Not Set (4227)
- Cross-Domain Misconfiguration
- Missing Anti-clickjacking Header (10)
- Big Redirect Detected (Potential Sensitive Information Leak) (23
- Cookie No HttpOnly Flag (2)
- Cookie Without Secure Flag (2)

Missing Anti-clickjacking Header
URL: https://www.boozt.com/nl/nl/dameskleding/sport/bekijk-alles
Risk: Medium
Confidence: Medium
Parameter: x-frame-options
Attack:
Evidence:
CWE ID: 1021
WASC ID: 15
Source: Passive (10020 - Anti-clickjacking Header)
Alert Reference: 10020-1
Input Vector:
Description:
The response does not protect against 'ClickJacking' attacks. It should include either Content-Security-Policy with 'frame-ancestors' directive or X-Frame-Options.

Alerts 1 8 9 8 Main Proxy: localhost:8080 Current Status 0 0 0 0 0 0 0 0 1 0

Proposed Mitigation or Fix

For to prevent clickjacking attack options due to absence of the anti-clickjacking header, do these steps below:

1. **Set X-Frame-Options Header:** Add the following header to all server responses:

   ```
   X-Frame-Options: SAMEORIGIN
   ```
   or
   ```
   X-Frame-Options: DENY
   ```

2. **Use Content-Security-Policy (CSP):** Add a `Content-Security-Policy` header that specifies permitted sources for frames:

   ```
   Content-Security-Policy: frame-ancestors 'self';
   ```

3. **Apply Defense in Depth:** Implement both `X-Frame-Options` and `frame-ancestors` directives for broader browser support.
4. **Test Implementation:** After applying headers, verify protection using online tools like [Clickjacking Testers] or by manually testing frame behavior.

Conclusion.

OWASP ZAP scan on https://www.boozt.com/ indicated the absence of an anti-clickjacking header, with more risk for clickjacking attacks. Such vulnerability is particularly concerning on an online store where customers are performing sensitive activities like submitting orders or editing account details, which are prone to being abused through clickjacking. Absence of a CSP header and other issues that ZAP reports (e.g., cookies missing HttpOnly flag) enhance the overall threat by increasing the attack surface, and these have to be remediated concurrently. The deployment of the X-Frame-Options header, supplementing with a CSP frame-ancestors directive, and the usage of complementary security controls will significantly reduce the vulnerability of clickjacking attacks and enhance the platform's security stance. This report includes step-by-step instructions to reproduce the issue and actionable mitigation steps. Ongoing monitoring, secure coding, and regular audits are the best ways to keep the platform secure against future attacks, especially in the competitive e-commerce environment.