Executive Summary of Penetration Test Conducted on Team A’s Website.

By

Team B Members (Kikelomo Obayemi, Suresh Melvin Sigera)

Date: 24 October 2021

1. Background

Team B has recently conducted a penetration test on Team A’s AWS website in order to determine its exposures to security threats and vulnerabilities. As stated in the proposal earlier submitted, this assessment was conducted using the Penetration Testing Execution Standard (PTES) methodology. The entire report has therefore been structured to follow the PTES steps.

2. Pre-engagement Interaction

Team A runs an E-commerce site used to provide payment services and advice for commercial website operators.

Application	Landing Page	URL
E-commerce Site	**https://nismphp2-env.eba-q2b3ptmn.us-east-1.elasticbeanstalk.com/**	**52.1.33.42**

Based on the available information, the following tools have been carefully selected for this pen test;

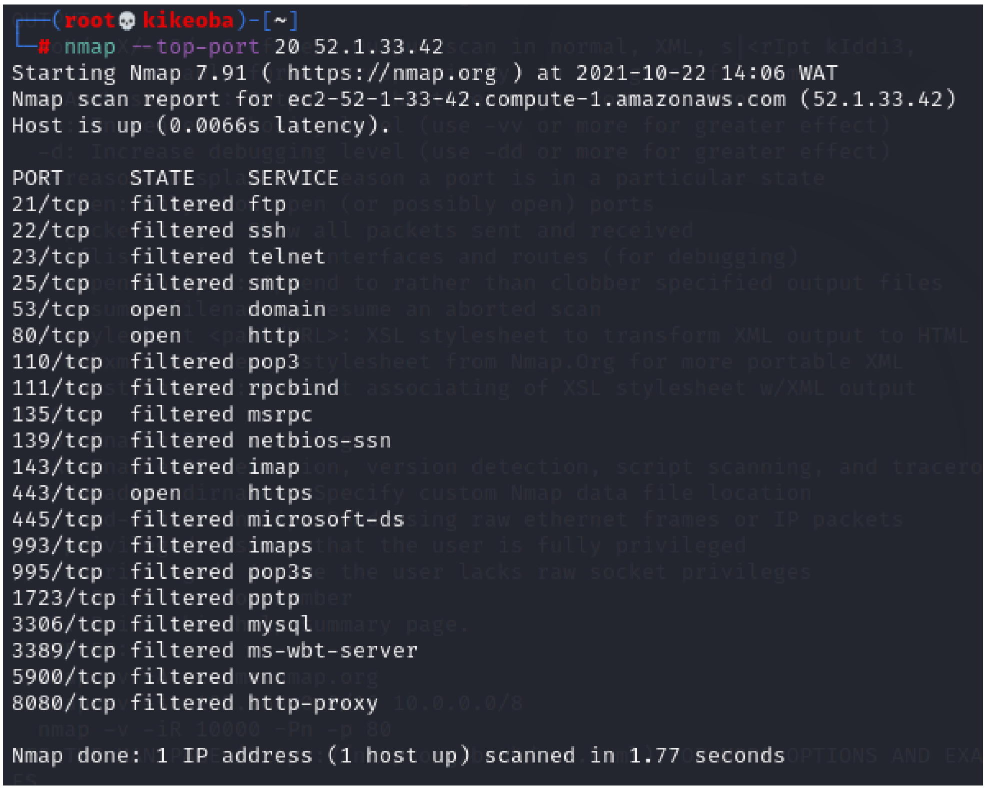
- Nmap: for port scanning
- OWASP ZAP: for web application pen testing
- Metasploit Framework (MSF): for exploitation of vulnerabilities
- Kali Linux: Advanced pen testing tool which contains a suite of over 600 tools (including Nmap, OWASP ZAP and MSF).

3. Intelligence Gathering

The first task conducted was to find basic information about Team A’s system. Using the **NMAP** tool and running the commands below, the following information was extracted:

A. Conducting a port scan to detect open ports and services used.

command: nmap --top-ports 20 **52.1.33.42**



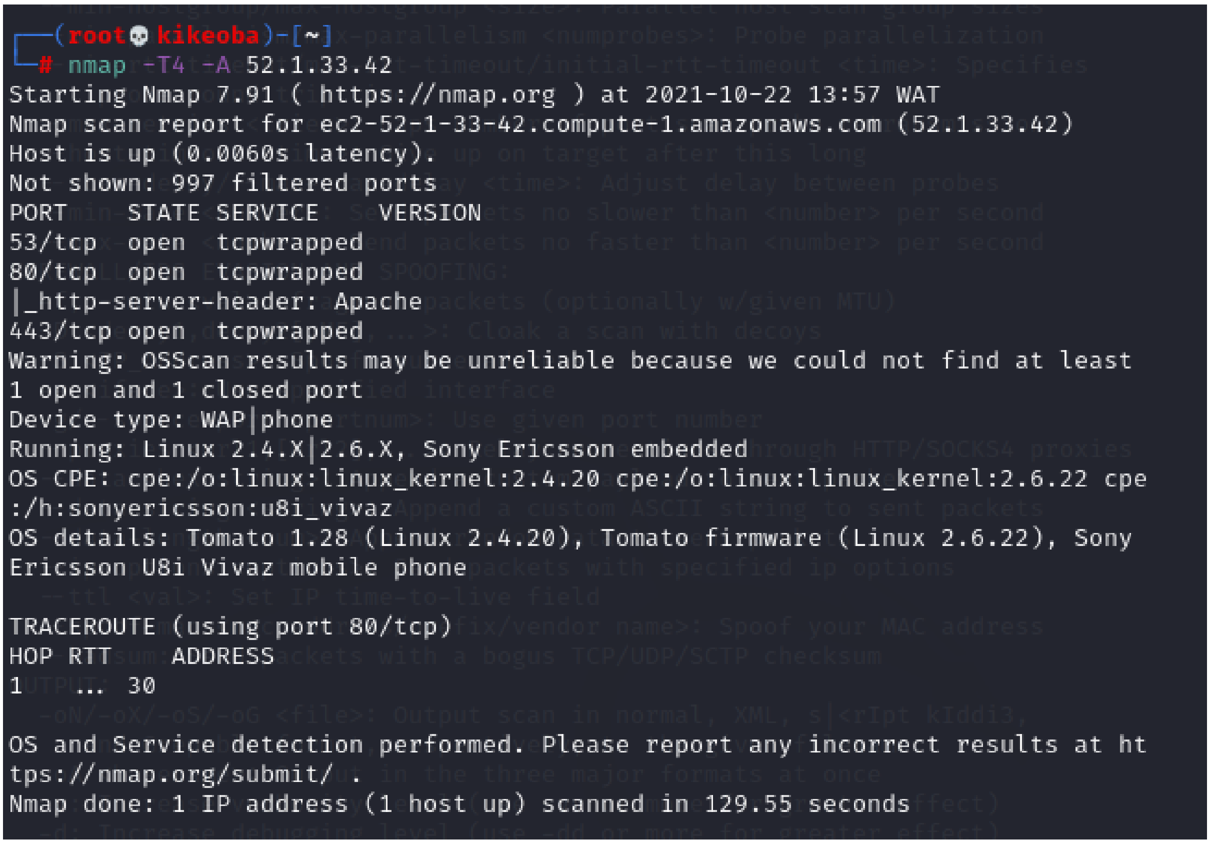
```
┌──(root💀kikeoba)-[~]
└─# nmap --top-port 20 52.1.33.42
Starting Nmap 7.91 ( https://nmap.org ) at 2021-10-22 14:06 WAT
Nmap scan report for ec2-52-1-33-42.compute-1.amazonaws.com (52.1.33.42)
Host is up (0.0066s latency).

PORT     STATE    SERVICE
21/tcp   filtered ftp
22/tcp   filtered ssh
23/tcp   filtered telnet
25/tcp   filtered smtp
53/tcp   open     domain
80/tcp   open     http
110/tcp  filtered pop3
111/tcp  filtered rpcbind
135/tcp  filtered msrpc
139/tcp  filtered netbios-ssn
143/tcp  filtered imap
443/tcp  open     https
445/tcp  filtered microsoft-ds
993/tcp  filtered imaps
995/tcp  filtered pop3s
1723/tcp filtered pptp
3306/tcp filtered mysql
3389/tcp filtered ms-wbt-server
5900/tcp filtered vnc
8080/tcp filtered http-proxy

Nmap done: 1 IP address (1 host up) scanned in 1.77 seconds
```

Figure 3a: Port scan of top 20 ports

As seen in figure 3a, of all 20 ports, port 53, 80 and 443 are open. Port 53 is open to access the domain name server, ports 80 and 443 are open for web connection via the http or https protocols respectively. All other ports' status indicates “filtered”. Nmap is not able to determine if the other 17 ports are open as the packets are filtered off (possibly by a firewall) before reaching the ports (Chen, 2018).

B. Checking the operating system running on the server

command: nmap -T4 -A **52.1.33.42** runs an aggressive scan on the IP address which return the operating system, performs a traceroute and also checks for open ports



```
┌──(root💀kikeoba)-[~]
└─# nmap -T4 -A 52.1.33.42
Starting Nmap 7.91 ( https://nmap.org ) at 2021-10-22 13:57 WAT
Nmap scan report for ec2-52-1-33-42.compute-1.amazonaws.com (52.1.33.42)
Host is up (0.0060s latency).
Not shown: 997 filtered ports
PORT    STATE SERVICE    VERSION
53/tcp  open  tcpwrapped
80/tcp  open  tcpwrapped
|_http-server-header: Apache
443/tcp open  tcpwrapped
Warning: OSScan results may be unreliable because we could not find at least
1 open and 1 closed port
Device type: WAP|phone
Running: Linux 2.4.X|2.6.X, Sony Ericsson embedded
OS CPE: cpe:/o:linux:linux_kernel:2.4.20 cpe:/o:linux:linux_kernel:2.6.22 cpe
:/h:sonyericsson:u8i_vivaz
OS details: Tomato 1.28 (Linux 2.4.20), Tomato firmware (Linux 2.6.22), Sony
Ericsson U8i Vivaz mobile phone

TRACEROUTE (using port 80/tcp)
HOP RTT    ADDRESS
1   ... 30

OS and Service detection performed. Please report any incorrect results at ht
tps://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 129.55 seconds
```

Figure 3b: Operating system information

Figure 3b above shows Linux operating system is being used, version 2.4.x, 2.6.x

C. Loading the landing page to verify the web connection protocol

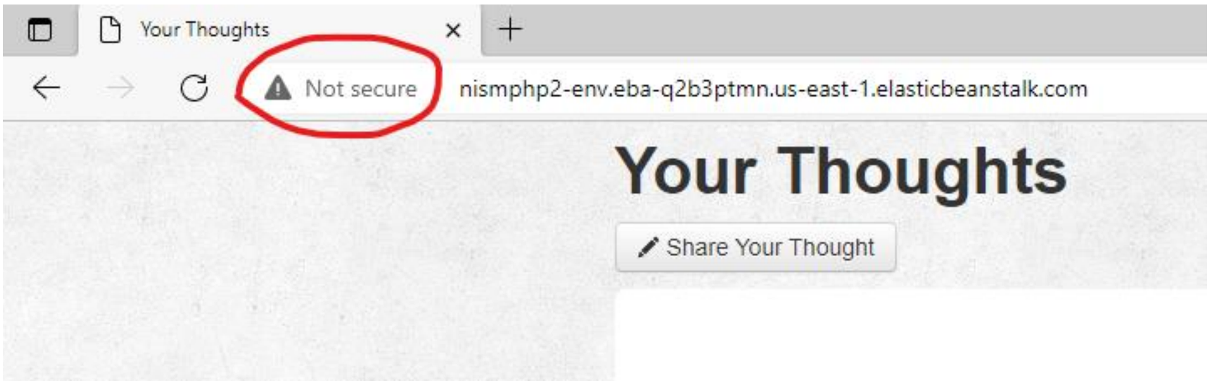
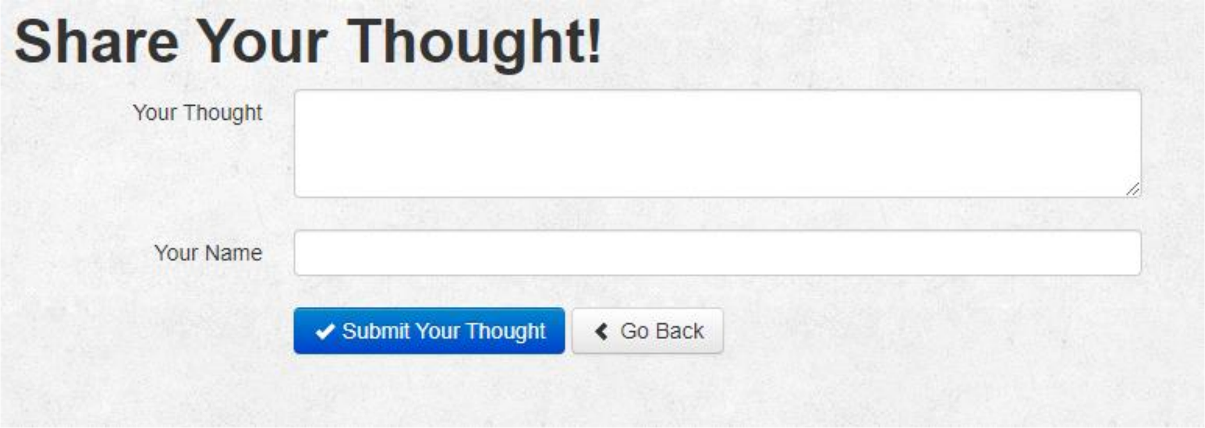


Figure 3c: Unsecure HTTP connection used on Team A’s website

Figure 3c above shows that the web connection is done via HyperText Transfer Protocol (HTTP). HTTP connections transfer data as plain text without any form of encryption.

D. Accessing the second page by clicking “Share Your Thought”

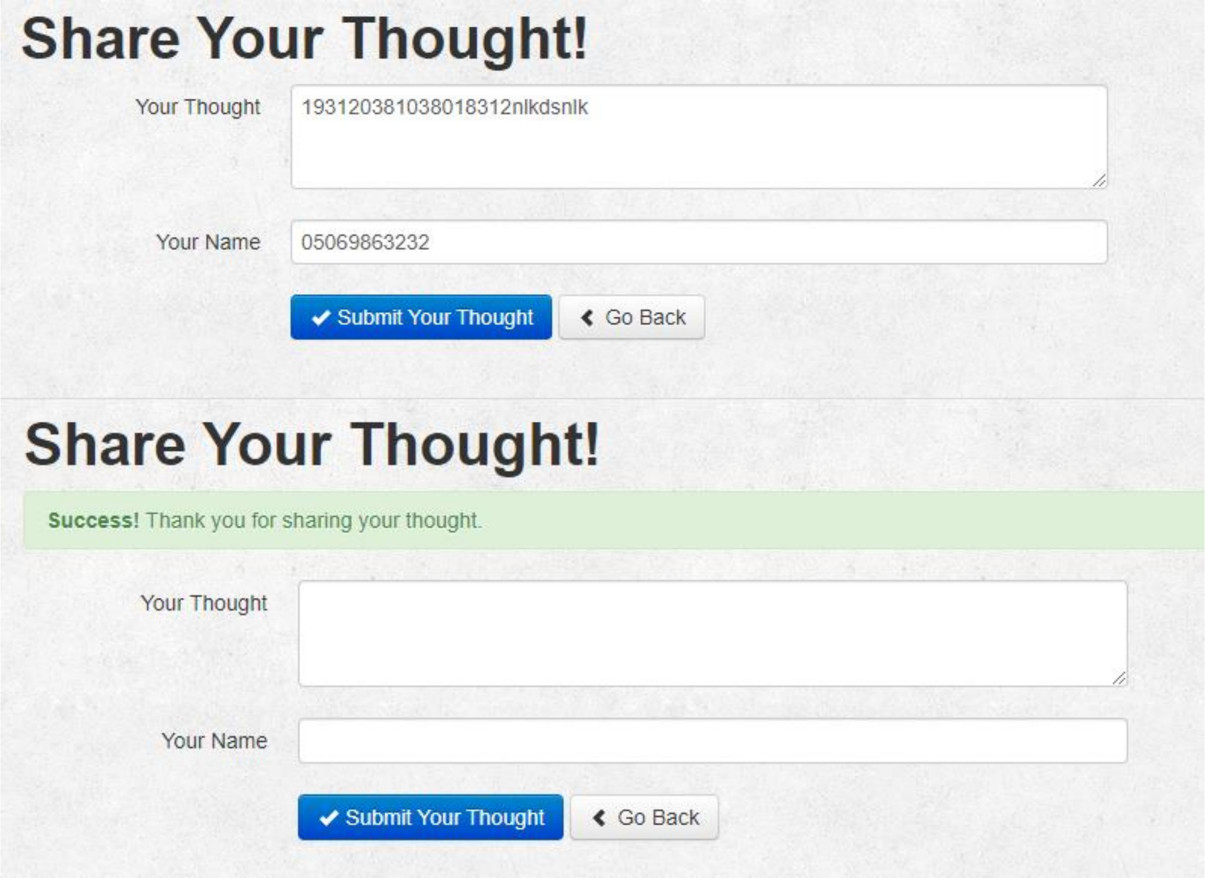


Share Your Thought!

Your Thought:

Your Name:

Figure 3d: Second page (http://nismphp2-env.eba-q2b3ptmn.us-east-1.elasticbeanstalk.com/add)



Share Your Thought!

Your Thought: 193120381038018312nlkdsnlk

Your Name: 05069863232

Share Your Thought!

Success! Thank you for sharing your thought.

Your Thought:

Your Name:

Figure 3e: non-validated user input accepted

Figure 3e shows a form on the second page accepting numbers in a space where only text should be allowed. User input is not validated

4. High Level Threat Modelling

As required by the PTES Methodology, a high-level threat modelling was conducted to analyse four key elements of the business namely business assets, business processes, threat agents and threat capability (PTES, 2014). The goal is to draw Team A’s attention to every aspect that poses a risk to its organisation.

4.1 Business Assets

Team A’s business assets outlined below have been classified as high, medium or low risk based on their value to attackers.

High	very valuable
Medium	somewhat valuable
Low	of little value

Table 4a: Classification Index

- Employee and Customer Information: This includes Personal Identifiable Information, Financial Information and Health Information (for employees) - **High**
- Company Data: This includes company policies, business plans, trade secrets, strategic information and user accounts - **High**
- Technical Information: IT architecture design information. - **High**

Rationale: Oerting (n.d) as cited in (Pickup, 2017) reported that not only do hackers go after financial information as most business owners think, they also go after company’s intellectual property, business plans, top secrets, strategic information and personal data.

4.2 Business Processes

The business processes that could pose a risk to Team A have also been classified using the index in Table 4a.

- IT and HR business processes are out of scope for this pen test as required information is not available.
- Third party integration processes **- High.**
  Rationale: E-commerce sites use a lot of third-party applications to provide excellent services and ensure their customers stay satisfied and loyal. Some commonly used applications are payment gateway, shipping gateway, inventory management system, customer relationship management system, live chats, notifications system amongst others (agile, n.d). Microsoft researchers remark that it is difficult to securely integrate third party applications as they usually contain logical flaws that provide a pathway to malicious individuals to launch an attack (Xing et al., 2013).

4.3 Threat Agents and Capability

Both internal and external threat agents have been identified for Team A and they have been categorised as severe, high, medium and low risk based on the following factors as outlined by PTES (2014): the tools available to them for use in their daily jobs, capability in exploiting the system and their accessibility to develop such exploits through third parties.

4.4.1 Internal Threat Agents Capability Analysis

Table 4b below shows the classification of threats identified

Threat Agent	Ranking	Rationale
Employees	**Low to Severe**	**Low**: In general, employees are not considered to be a threat as they earn a living from the organisation and most likely want it to succeed (PTES, 2014) **Medium to High**: poorly trained employees could be caught in phishing attacks. With the advent of BYOD, employees are increasingly accessing company network from personal devices that are not managed by the organisation (Morrow, 2012) **Severe**: disgruntled employee who could intentionally target the system.
Management Executives	**Low, High**	**Low**: Management Executives also want the company to succeed so they are less likely to pose a threat **High**: They approve policies. Bad policies are a threat to an

		organisation. They also have access to privileged information (PTES, 2014).
Administrators	**Low, High**	**Low**: They are usually trained individuals who have a high level of security awareness. **High**: they have privileged rights and information and are skilled enough to launch an inside attack.
Contractors	**Low to Medium**	Contractors occasionally gain physical access to certain areas in the organisation and as such could tamper with devices intentionally or otherwise

Table 4b: Internal threat agents capability analysis

4.4.2 External Threat Agents Capability Analysis

Threat Agent	Ranking	Rationale
Customers	**Low**	Although customers access applications from devices that are not been managed by organisations, their access is fairly restricted
Competitors	**Low to Medium**	Competitors use customer journey hijacking techniques in the form of pop-up ads to distract customers from e-Commerce sites (Namogoo, 2019). Unfortunately, these ads also contain malwares that could infect customers' systems when clicked and as such

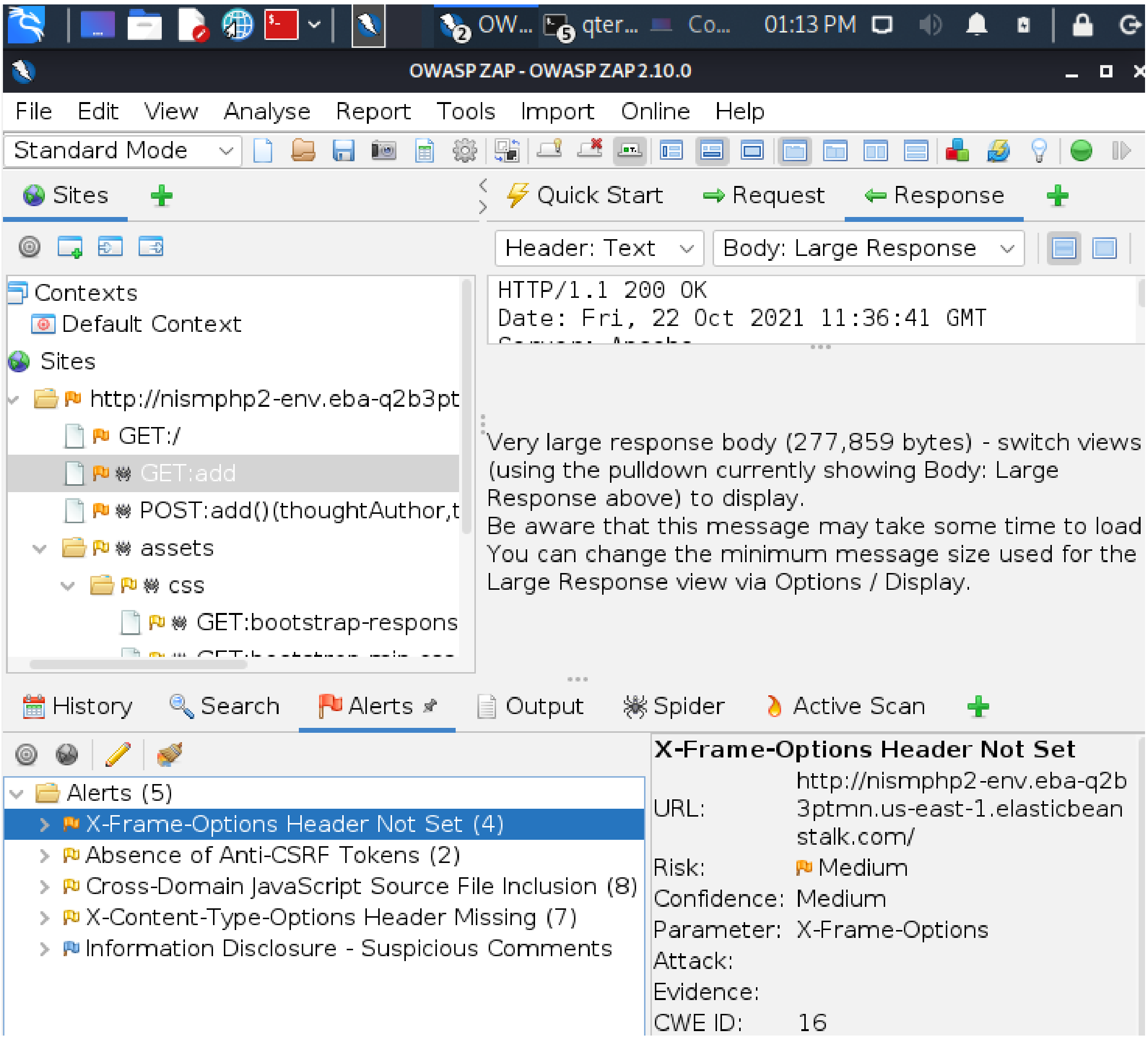
		creates a bad experience for the customer.
Business Partners	**Low**	Same as in customers above.
Hackers	**Severe**	They have a clear motive which is to attack the system in various possible ways.

Table 4c: External threat agents capability analysis

5. Vulnerability Assessment

5.1 Vulnerability Scanning

The OWASP ZAP tool was used to probe for vulnerabilities on Team A’s website. Figures 5a and 5b show the code-related risks identified on Team A’s website after scanning with OWASP ZAP.



```
HTTP/1.1 200 OK
Date: Fri, 22 Oct 2021 11:36:41 GMT
```

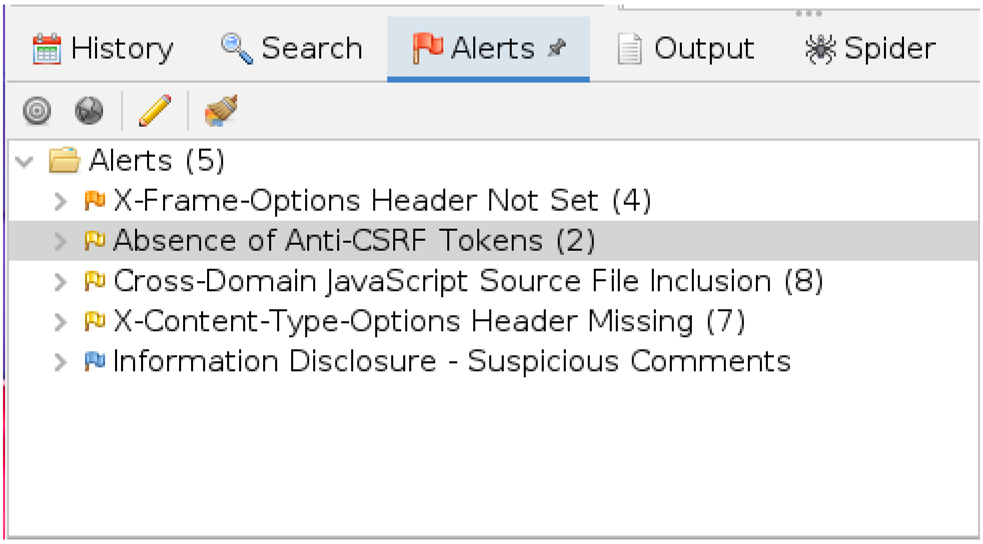
Very large response body (277,859 bytes) - switch views (using the pulldown currently showing Body: Large Response above) to display.
Be aware that this message may take some time to load
You can change the minimum message size used for the Large Response view via Options / Display.

Alerts (5)
- X-Frame-Options Header Not Set (4)
- Absence of Anti-CSRF Tokens (2)
- Cross-Domain JavaScript Source File Inclusion (8)
- X-Content-Type-Options Header Missing (7)
- Information Disclosure - Suspicious Comments

X-Frame-Options Header Not Set

URL: http://nismphp2-env.eba-q2b3ptmn.us-east-1.elasticbeanstalk.com/
Risk: Medium
Confidence: Medium
Parameter: X-Frame-Options
Attack:
Evidence:
CWE ID: 16

Figure 5a: OWASP ZAP Scan



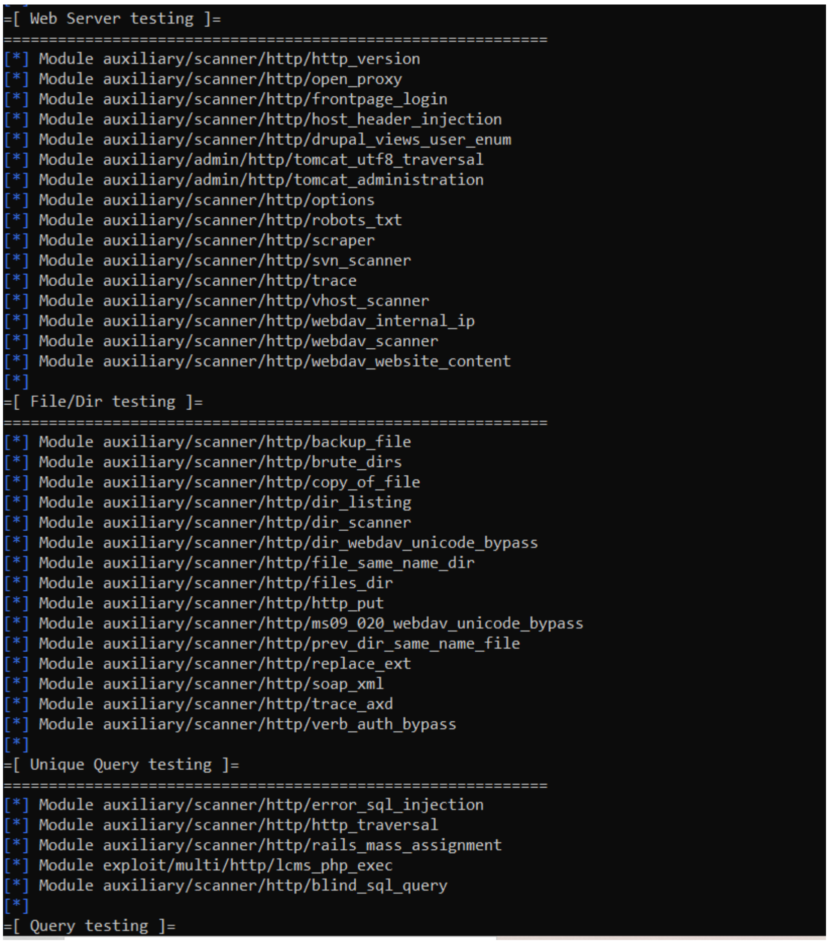
Alerts (5)

- X-Frame-Options Header Not Set (4)
- Absence of Anti-CSRF Tokens (2)
- Cross-Domain JavaScript Source File Inclusion (8)
- X-Content-Type-Options Header Missing (7)
- Information Disclosure - Suspicious Comments

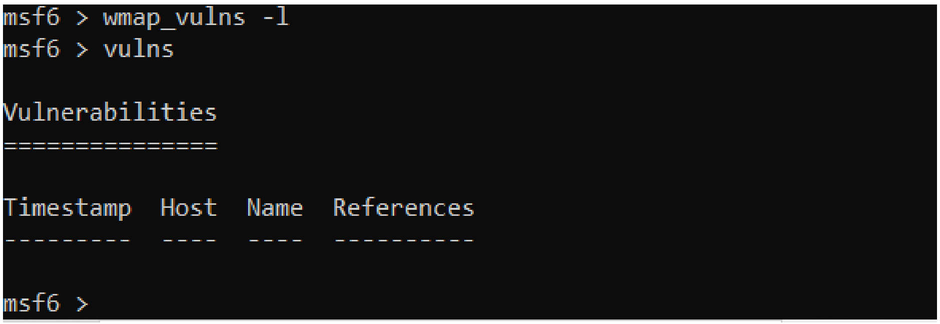
Figure 5b: Code-related risks identified by OWASP ZAP scan

The OWASP ZAP discovered five issues within the html codes used in developing Team A website. Of all five issues (shown in Figure 5b above), one is of medium risk, three are of low risk and one is informational. Section 6 provides further explanation of risks identified.

Figure 5c: Metasploit Scan



```
=[ Web Server testing ]=
============================================================
[*] Module auxiliary/scanner/http/http_version
[*] Module auxiliary/scanner/http/open_proxy
[*] Module auxiliary/scanner/http/frontpage_login
[*] Module auxiliary/scanner/http/host_header_injection
[*] Module auxiliary/scanner/http/drupal_views_user_enum
[*] Module auxiliary/admin/http/tomcat_utf8_traversal
[*] Module auxiliary/admin/http/tomcat_administration
[*] Module auxiliary/scanner/http/options
[*] Module auxiliary/scanner/http/robots_txt
[*] Module auxiliary/scanner/http/scraper
[*] Module auxiliary/scanner/http/svn_scanner
[*] Module auxiliary/scanner/http/trace
[*] Module auxiliary/scanner/http/vhost_scanner
[*] Module auxiliary/scanner/http/webdav_internal_ip
[*] Module auxiliary/scanner/http/webdav_scanner
[*] Module auxiliary/scanner/http/webdav_website_content
[*]
=[ File/Dir testing ]=
============================================================
[*] Module auxiliary/scanner/http/backup_file
[*] Module auxiliary/scanner/http/brute_dirs
[*] Module auxiliary/scanner/http/copy_of_file
[*] Module auxiliary/scanner/http/dir_listing
[*] Module auxiliary/scanner/http/dir_scanner
[*] Module auxiliary/scanner/http/dir_webdav_unicode_bypass
[*] Module auxiliary/scanner/http/file_same_name_dir
[*] Module auxiliary/scanner/http/files_dir
[*] Module auxiliary/scanner/http/http_put
[*] Module auxiliary/scanner/http/ms09_020_webdav_unicode_bypass
[*] Module auxiliary/scanner/http/prev_dir_same_name_file
[*] Module auxiliary/scanner/http/replace_ext
[*] Module auxiliary/scanner/http/soap_xml
[*] Module auxiliary/scanner/http/trace_axd
[*] Module auxiliary/scanner/http/verb_auth_bypass
[*]
=[ Unique Query testing ]=
============================================================
[*] Module auxiliary/scanner/http/error_sql_injection
[*] Module auxiliary/scanner/http/http_traversal
[*] Module auxiliary/scanner/http/rails_mass_assignment
[*] Module exploit/multi/http/lcms_php_exec
[*] Module auxiliary/scanner/http/blind_sql_query
[*]
=[ Query testing ]=
```



```
msf6 > wmap_vulns -l
msf6 > vulns

Vulnerabilities
===============

Timestamp  Host  Name  References
---------  ----  ----  ----------

msf6 >
```

The screenshot above contains the results of a security audit performed by using Metasploit. WMAP, a web application vulnerability scanner integrated with Metasploit has been used to conduct web application scanning from within the Metasploit Framework.

Commands executed on Metasploit:

wmap_sites -a http://52.1.33.42 *adds site as a target*

wmap_run -e scans *the target*.

wmap vuln -l *checks the database to see if any vulnerability was found*

vulns *shows the vulnerability*

From the output of **Figure 5c**, we can see that WMAP has reported zero vulnerability as shown under the 'vulns' parameter.

6. Report

6.1 Conclusion of findings.

After conducting a penetration test on Team A's website, Team B has found some issues of concern. The DREAD model developed by Microsoft (2003) has been used to rate the threats according to the following factors: Damage Potential (D), Reproducibility (R), Exploitability (E), Affected Users (A), Discoverability(D). The scores assigned to each threat range from 1 to 3 with 1 being low, 2 for Medium and 3 for High.

Table 6a DREAD Ratings Table

Risks Identified	D	R	E	A	D	Total
Unsecure HTTP Protocol used	3	3	2	3	3	**14**
Input form not Secure	3	2	1	3	3	**12**

Code vulnerabilities	1	1	1	1	3	**7**

6.1.1 Rationale for DREAD Ratings

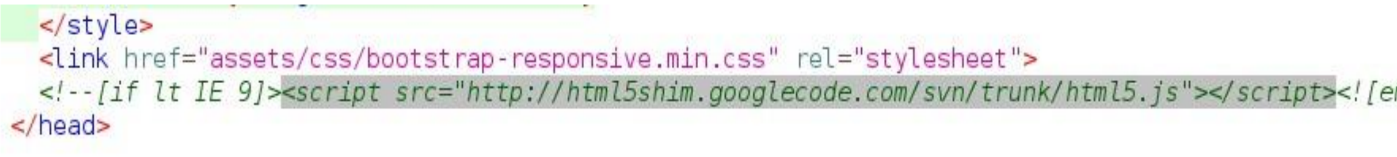
Use of HTTP Protocol: HTTP is used to communicate between the browser and the server and as earlier mentioned, it is not secure. It transfers data in plain text. Team A’s customers are at risk of their user credentials and financial information being stolen by an attacker eavesdropping on the network.

Input form not secure: input validation is not implemented on an input form on Team A’s website. Figure 3e shows numbers being accepted into a text field. An attacker could take advantage of this opportunity to inject all kinds of malicious code into the system leading to buffer overflow, sql injection and cross site scripting attacks (NTT, 2021)

Code Vulnerabilities

The OWASP ZAP tool identified five issues within the html code. Only four of them are of concern and are explained below

- X-Frame Option Header Not Set: X-frame header options is used to indicate how a browser renders a page and if not set, can lead to click-jacking attacks; a method of tricking a user into clicking a button that it not what they think it is (Mozilla,2021).
- Absence of anti-CSRF tokens: CSRF stands for Cross-Site Request Forgery. Anti-CSRF tokens were not found in the html submission form which means there is no way for the web server to verify whether a consistent request was intentionally provided by the user.
- Cross-Domain Javascript Source File inclusion: the code snippet (shown in Figure 6a) contains a third-party script file that cannot be controlled by end users of the application.



```
  </style>
  <link href="assets/css/bootstrap-responsive.min.css" rel="stylesheet">
  <!--[if lt IE 9]><script src="http://html5shim.googlecode.com/svn/trunk/html5.js"></script><![e
</head>
```

Figure 6a: Code snippet showing external JavaScript file inclusion

- X-Content-Type Options Header Missing: not setting the X-content-type options header allows web browsers to perform MIME sniffing, a situation where the browser determines how to process a URL rather than using a pre-defined method (Mozilla, 2021). This vulnerability makes it possible for an attacker to induce client-side code execution that can cause the website to work incorrectly (Coalfire, 2021).

6.3 Recommendations.

Based on our findings, we have categorised the list of concerns, in order of priority and their risk ratings.

Issue	High Risk	Medium Risk	Low Risk
Unsecure http protocol used	x		
Input form not secure	x		
HTML Code vulnerabilities		x	x

Table 6b: Findings Matrix

6.3.1 Security standards

Team B also found some other concerns relating to global security standards

Payment Card Industry Data Security Standard (PCIDSS)

The PCIDSS is an information security standard designed to create secure payment solutions for all stakeholders who operate a global infrastructure for processing payments (PCI Security, 2021). One of the requirements of the PCIDSS is for stakeholders to ensure that transmission of cardholder data is encrypted across a network (ControlCase, 2020). Using http protocol on Team A’s website is not PCI compliant as it doesn’t encrypt user data in any way, rather http transmits data in plain text (Brown. 2017)

General Data Protection Regulation

By using the unsecure http protocol for web connection, Team A’s website is in contravention of Article 24 of the GDPR which requires that a data controller must implement appropriate technical and organisation measures to ensure that data is securely processed in accordance with the regulation (GDPR, 2016)

6.3.2 Proposed Solutions

Issue	Our recommended solution
Unsecure http protocol used	Use HTTPS protocol. HTTPS provides encryption of data and protects customers from attackers eavesdropping on the network (Brown, 2017)
Input form not secure	OWASP recommends input validation implementation on web forms to ensure that only required data is accepted into a system component. Input validation techniques include the use of allowlisting, denylisting and regular expressions (OWASP, 2018).
HTML Code vulnerabilities	Address code vulnerabilities by: • Ensuring external files are from a trusted source. • Implementing use of anti-csrf packages such as OWASP CSRFguard to protect against cross site request forgery (Mitre, 2021). • Setting X-Frame header options to “SAMEORIGIN” or “DENY” to ensure that the web pages are properly rendered and protected against clickjacking attacks (Mozilla, 2021). • Set the X-Content-Type-Options header to 'nosniff' for all web pages to prevent MIME sniffing on web browsers (Microsoft, 2016)

References

Agilie (n.d) Available from: https://agilie.com/en/blog/most-vital-third-party-integrations-for-e-commerce-websites [Accessed 22 October 2021]

Brown T. (2017). What is an SSL port? A technical guide for HTTPS. Available from: https://www.godaddy.com/garage/whats-an-ssl-port-a-technical-guide-for-https/ [Accessed 18 October 2021]

Chen Y. (2018).Nmap.Available from: https://wiki.onap.org/display/DW/Nmap [Accessed 20 October 2021]

Coalfire (2021) MIME sniffing in browsers and the security implications. Available from: https://www.coalfire.com/the-coalfire-blog/april-2021/mime-sniffing-in-browsers-and-the-security [Accessed 23 October 2021]

Controlcase (2020) PCIDSS Checklist. Available from: https://www.controlcase.com/pci-dss-checklist-2020/ [Accessed 23 October 2021]

Microsoft (2003) Threat Modelling Available from: https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff648644(v=pandp.10)?redirectedfrom=MSDN [Accessed 21 October 2021]

Microsoft (2016). Reducing MIME type Security Risks. Available from: http://msdn.microsoft.com/en-us/library/ie/gg622941%28v=vs.85%29.aspx [Accessed 21 October 2021]

Mitre (2021) CWE-352 Cross Site Request Forgery. Available from: http://cwe.mitre.org/data/definitions/352.html [Accessed 23 October 2021]

Morrow, B. (2012). BYOD security challenges: control and protect your most sensitive data. *Network Security*, *2012*(12):5-8.

Mozilla (2021) X-frame Options. Available from: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options [Accessed 21 October 2021]

Mozilla (2021) MIME types (IANA Media types). Available from: https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics_of_HTTP/MIME_types#mime_sniffing [Accessed 21 October 2021]

Namogoo (2019) State of Customer Journey Hijacking. 2019 Benchmark Report

NTT (2021) Input Validation Available from: https://www.whitehatsec.com/glossary/content/input-validation [Accessed 20 October 2021]

OWASP (2018) Available from: https://owasp.org/www-project-proactive-controls/v3/en/c5-validate-inputs [Accessed 23 October 2021]

PCI Security (2020) PCI Security Available:https://www.pcisecuritystandards.org/ [Accessed 20 October 2021]

Pickup O. (2017). Which of your business assets are at risk. Available from:https://www.telegraph.co.uk/business/open-economy/which-of-your-business-assets-are-at-risk/ [Accessed 20 October 2021]

PTES (2014) Available from: http://www.pentest-standard.org/index.php/Main_Page [Accessed 21 October 2021]

Xing, L., Yangyi C., XiaoFeng W., and Shuo C. (2013) "InteGuard: Toward Automatic Protection of Third-Party Web Service Integrations." In *NDSS*.