Conclusion

A series of various tests and scans were done. These include penetration test and scanning using a number of tools. Overall, the findings of the web application under review indicated both good and bad security configurations. While the application allows fingerprinting of the web server it is hosted on, which was discovered to be an Apache web server, it does not disclose the version of Apache hosting it. There was no privacy policy or any link to the policy discovered on the webserver, and this may violate regulations such as (GPDR). Although privacy policy is not a security configuration, we recommend that a link to the policy or policy document should be provided on the web application. We recommend using SSL/TLS in contrast with HTTP since this includes data protection from unauthorised access and privacy. This configuration is a component that qualifies compliance with standards such as PCI DSS.

The following list is a summary of items requiring remediation :

|  |  |  |
| --- | --- | --- |
| Description | Risk Rating | Page Reference |
| Issue 1. Sensitive Data Exposure |  |  |
| Issue 2. Cross-Site Scripting. |  |  |
| Issue 3. Security Misconfiguration. |  |  |
| Issue 4. Using Components with Known vulnerabilities |  |  |
| Issue 5. Absence of Privacy Policy |  |  |
| Issue 6. Compliance with regulations and standards. |  |  |

The BeanStalk report shows that the website lacks a security header X-XSS-Protection. According to MDN web docs, (N.D) This is a feature of Internet Explorer, Chrome and Safari that stops pages from loading when they detect reflected cross-site scripting ([XSS](https://developer.mozilla.org/en-US/docs/Glossary/Cross-site_scripting)) attacks.

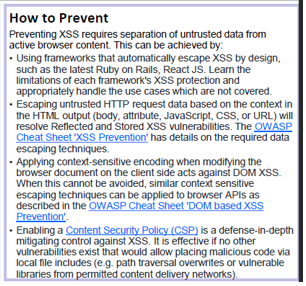


Figure 2: OWASP A7 Cross-Site Scripting (XSS) (Adapted from (Dehalwar et al., 2018)

Security Misconfiguration

This is improper web application configuration which can lead to countless errors. It is commonly caused by unnecessary features not being disabled, default accounts still being used, unpatched or unmaintained server code, references to old versions of services. Attackers can exploit any security misconfiguration to gain access, elevate privileges, or violate the confidentiality or integrity of the data.

Figure 3 shows how to prevent security misconfiguration.

****

Fig3. Security misconfiguration prevention. (Adaptable from (Diem Shin, 2019)

Using Components with Known Vulnerabilities

(ISC) 2, (2006) states that components, such as libraries, frameworks and other software modules, almost run with full privileges. If a vulnerable component is exploited, such an attack can facilitate severe data loss or server takeover. Applications vising components with known vulnerabilities may undermine application defences and enable a range of possible attacks and impacts.

Figure 4 illustrates how a vulnerable component can lead to attacks

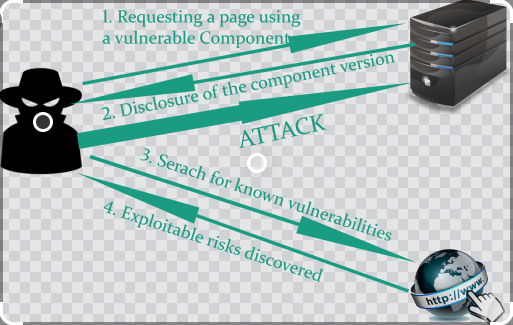


Figure 4: Using components with known vulnerabilities (Anonymous, N.D)

Vulnerable library packages need to be constantly updated. If no updates are available, try creating a PR patching the vulnerability or get a new update with no vulnerabilities.

Lastly, the test and scans will have to be carried out again after implementing and correcting all flaws that were picked up by the reports.