A green chameleon logo

Description automatically generated

**OWASP SCAN OF CHAMELEON**

**FINDING VULNERABILITY**

​​

**TABLE OF CONTENTS**

1-SUMMARY……………………………………………………………………………………..

2-INTRODUCTION……………………………………………………………………………..

3- TOOLS USED………………………………………………………………………………..

4-SCOPE OF TESTING………………………………………………………………………….

5-RESULTS…………………………………………………………………………………………

6-CONCLUSIONS……………………………………………………………………………….

7-REFERENCES……………………………………………………………………………………

**INTRODUCTION**

It is an assortment of instruments and methods for identifying, assessing, fixing, and documenting possible flaws in a system. The procedure is used in conjunction with other security measures to strengthen the system's defences and guarantee that any vulnerabilities are fixed before hackers can take advantage of them.

Additionally, vulnerability assessments serve as the cornerstone of a successful patch management plan, enabling developers to swiftly close security and performance gaps. By resolving vulnerabilities before they are released into production, comprehensive vulnerability management guarantees that web developers create a strong security posture.

**TOOLS USED**

There are numerous tools available for identifying weaknesses in networks, systems, and software programmes. Security experts, penetration testers, and ethical hackers frequently use these tools to find and fix security flaws. These are a few well-liked resources for identifying vulnerabilities:

Network Mapper, or Nmap:

An open-source tool that is both flexible and strong for network discovery and security auditing. Nmap can be used to detect versions, scan ports, and find possible security holes in systems that are networked.

URL: Nmap

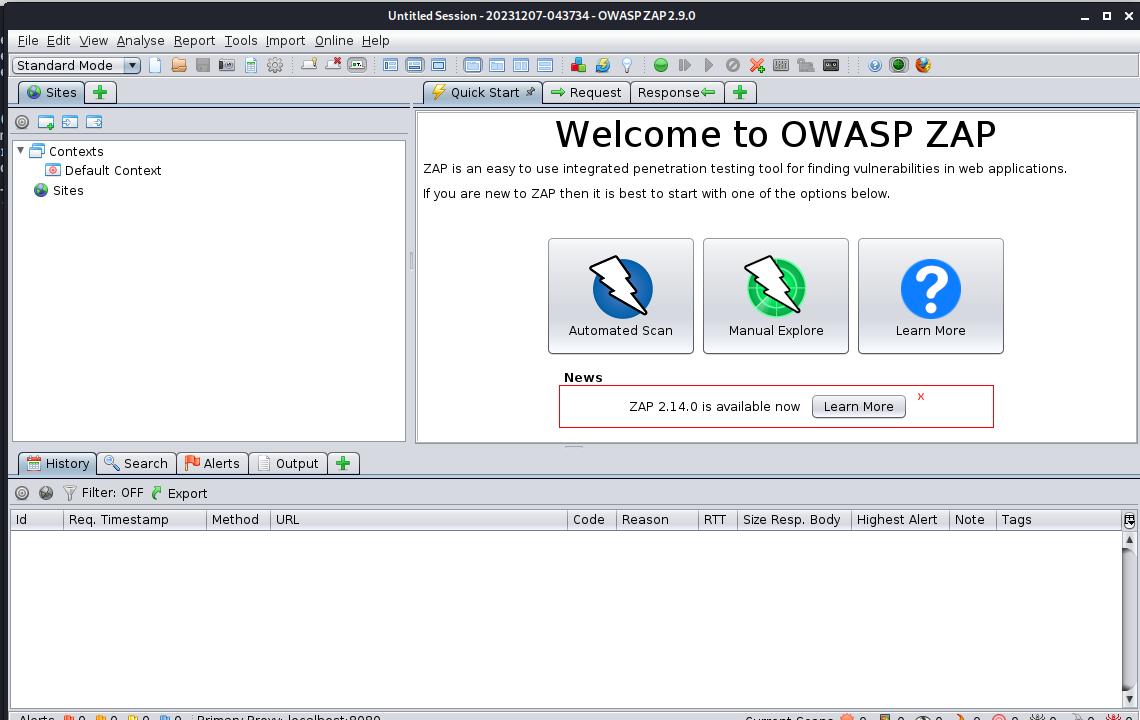
The OpenVAS (Open Vulnerability Assessment System) is an open-source vulnerability scanner that is intended to identify and evaluate security flaws in web apps and networks. It has a database of vulnerabilities that are known to exist and is capable of doing thorough scans.

OpenVAS website

**SCOPE OF TESTING**

Put another way, there is a long list of possible flaws in the website that you should look for. Such audits depend on either specific workflows or the OWASP methodology. Searching a website for vulnerabilities requires a great deal of creativity. Using a precise framework or particular tools—especially if they are open-source—is not restricted. However, security auditing is no laughing matter. Not surprisingly, some organisations attempt to impose policies on how these checks-ups are carried out in order to prevent a penetration tester from missing anything due to wishful thinking. Using the OWASP Web Security Testing Guide is one of the best ways to accomplish this. This is a thorough explanation of the guidelines for identifying vulnerabilities in web applications. Its creators assembled and explained the. Making use of the WhatWeb tool mentioned above is a good place to start if you need to determine whether it is possible to compromise a hand-coded website. But keep in mind that in this case, you are searching for all embedded services and their versions, not examining a CMS. Numerous versions of the framework are vulnerable to hacking. For instance, publicly available exploits can be used to compromise out-of-date versions of Ruby on Rails or Apache Tomcat. You can also get important hints by figuring out the versions of the programming languages. For example, fresh PHP vulnerabilities periodically come to light and may remain unpatched for weeks following their discovery.

**RESULTS**



For the testing of the chameleon website I used the OWASP zap tool to test the vulnerabilities present in the website. For that I clicked on the automated scan and then as shown in the below screenshot-

**A screenshot of a computer

Description automatically generated**

hot- I pasted the chameleon website link in the url to attack option and then started the attack as shown in below screenshot and it showed a plenty of the vulnerabilities present currently in the chameleon website.

**A screenshot of a computer

Description automatically generated**

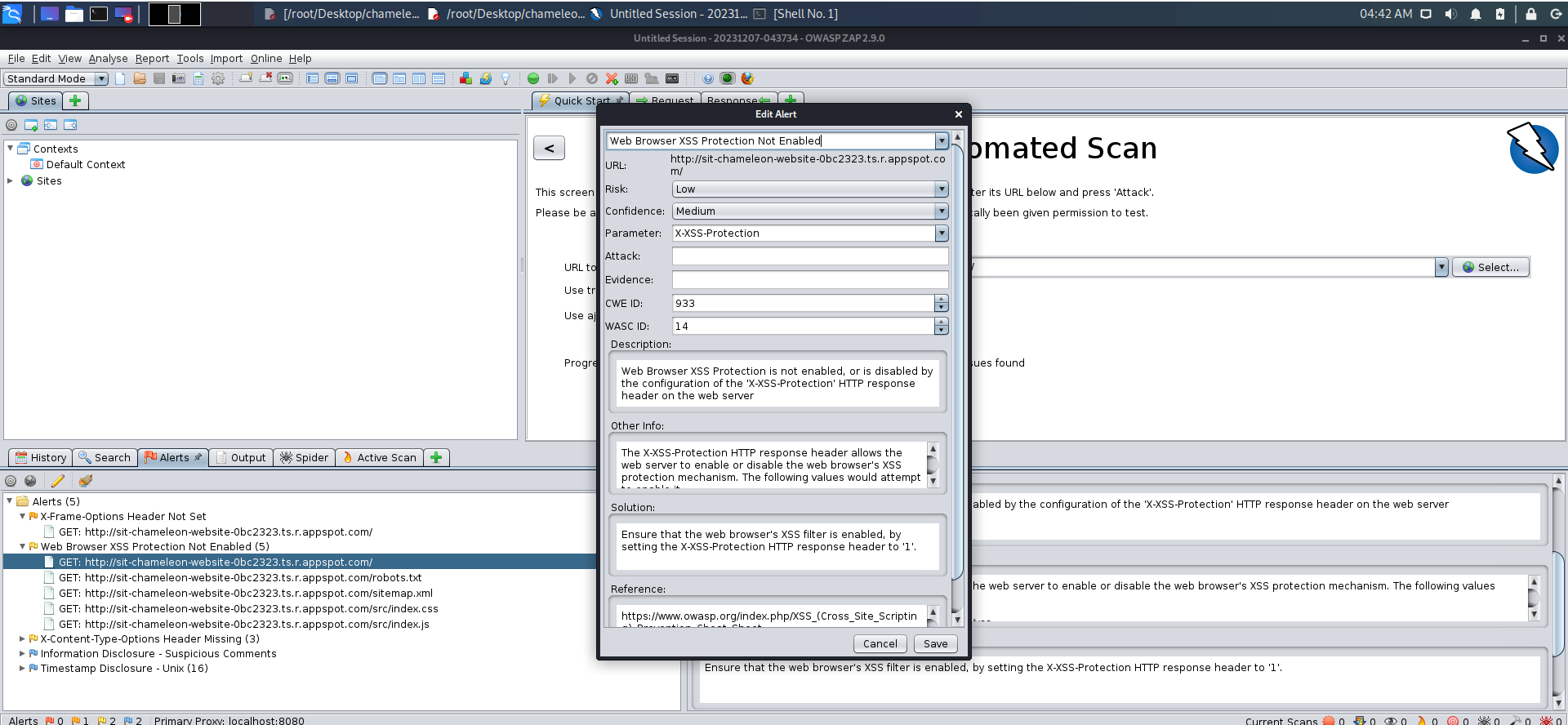
**A screenshot of a computer

Description automatically generated**

Here can be seen plenty of alerts which can seen such as the X-frame options, web browser protection not enabled and many more things which are lacking for this website.

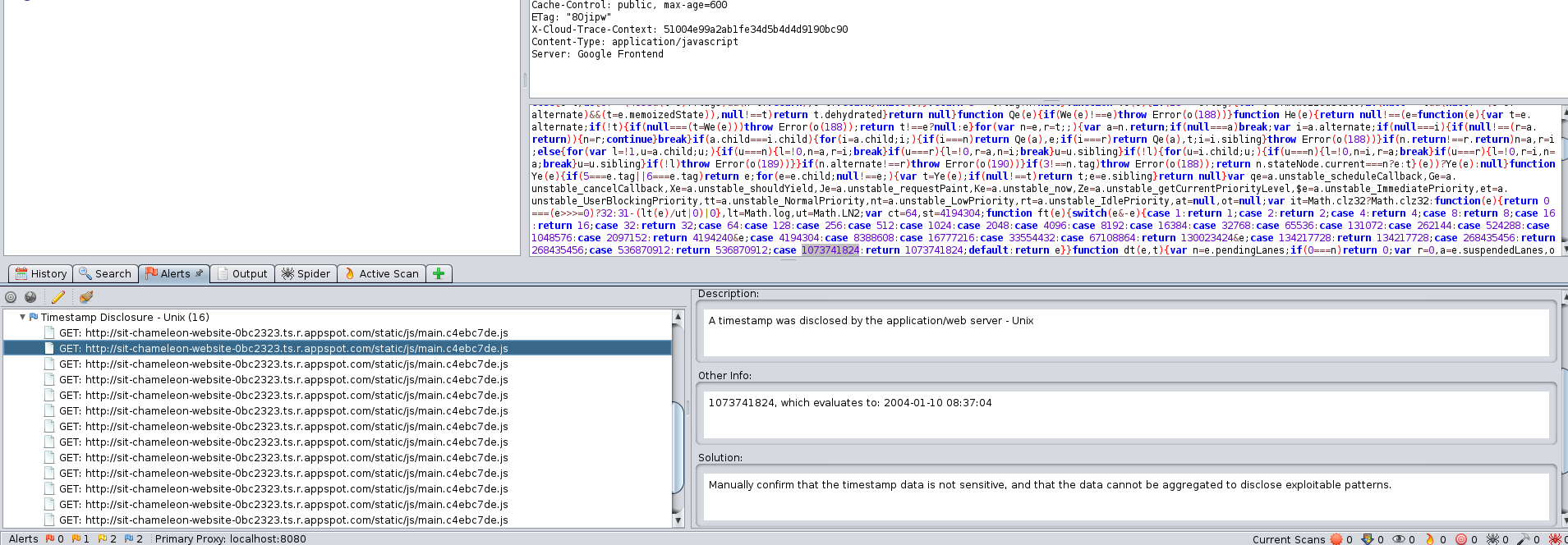
**A close-up of a computer screen

Description automatically generated**

****

**A screenshot of a computer

Description automatically generated**

****

In the above screenshot we can see thay there are many edit alerts for this website and the solution can also be provided with them as well.

**CONCLUSION**

Although safe coding practices are crucial, businesses can strengthen their security posture by proactively detecting and fixing security flaws with the aid of an effective vulnerability assessment. The OWASP top 10 is a valuable resource for teams seeking to strengthen their security measures, as it enumerates prevalent vulnerabilities in web applications and provides realistic remediation strategies.

References:

1. Anon, (n.d.). *【How To Find Vulnerability In A Website】Crashtest Security ®*. [online] Available at: <https://crashtest-security.com/how-to-find-vulnerabilities/>