USING VEGA VULNERABILITY SCANNER TO FIND VULNERABILITIES IN MOP AND CHAMELEON WEBSITE

ROSHAN JOSE - 221499484

Contents

VEGA VULNERABILITY SCANNER

Vega is an online vulnerability scanner and testing platform that is open source. Its purpose is to assist security experts in locating and evaluating vulnerabilities that are present in web applications. Both a graphical user interface (GUI) and a command-line interface (CLI) are made available to users for the purpose of analysing websites and online applications for potential vulnerabilities. Vega is distinguished by the following essential characteristics:

- Crawling: Vega has the capability to crawl websites in order to find and map out the structure of online applications. This includes recognising pages, forms, and input fields among other things.
- Scanning: In order to identify potential security flaws, it runs a number of different security tests, including SQL injection, cross-site scripting (XSS), directory traversal, and to detect vulnerabilities.
- Testing That Is Completely Automated Vega is able to automate the process of vulnerability scanning, which makes it much simpler for security experts to recognise any dangers.
- Reporting: Following a scan, Vega will provide thorough reports that highlight any vulnerabilities that were discovered and will also provide advice for how to fix them.
- Personalisation: Users have the ability to personalise the scanning process to their own requirements by customising the scan policies and regulations.
- Intercepting and Proxying: Vega is capable of acting as a proxy, enabling users to intercept and alter HTTP requests and replies in order to evaluate the behaviour of applications.

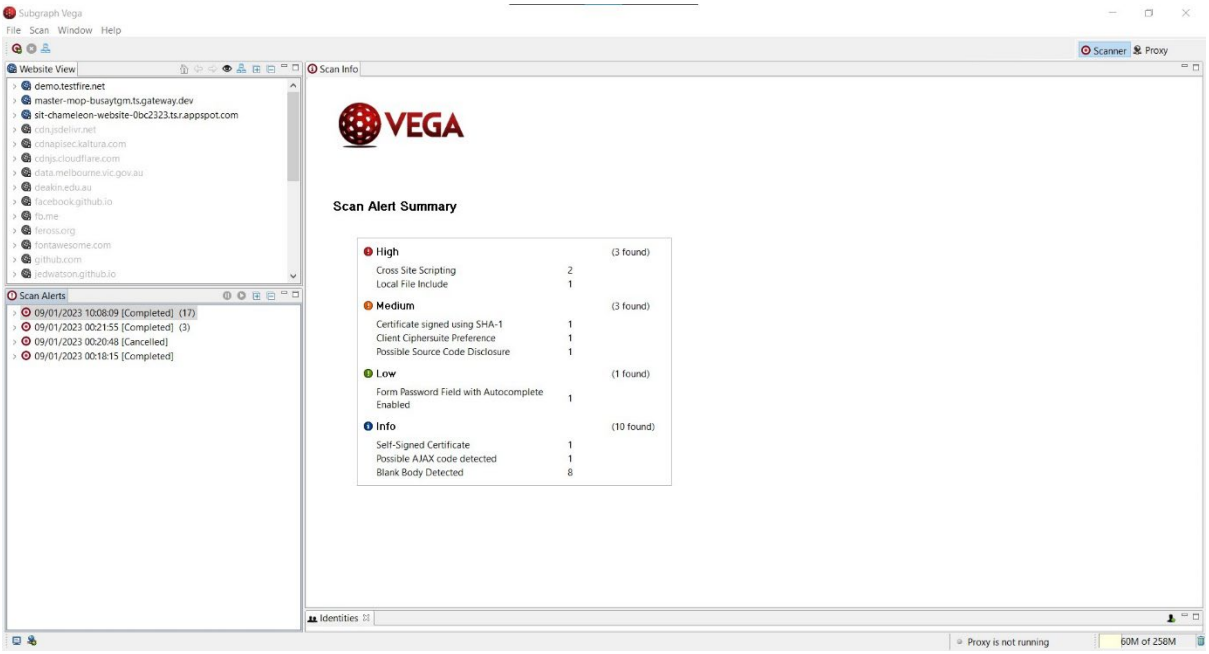
TUTORIAL VEGA

1. Download and Install Vega

First, download the Vega vulnerability scanner from its official website https://subgraph.com/vega/download.html and follow the installation instructions provided for your operating system. Also have to ensure you have Java installed on your system since Vega is a Java-based application and also required jdk version 8.

2. Launch Vega

After installation, launch Vega. You can use either the GUI (Graphical User Interface) or CLI (Command-Line Interface) version

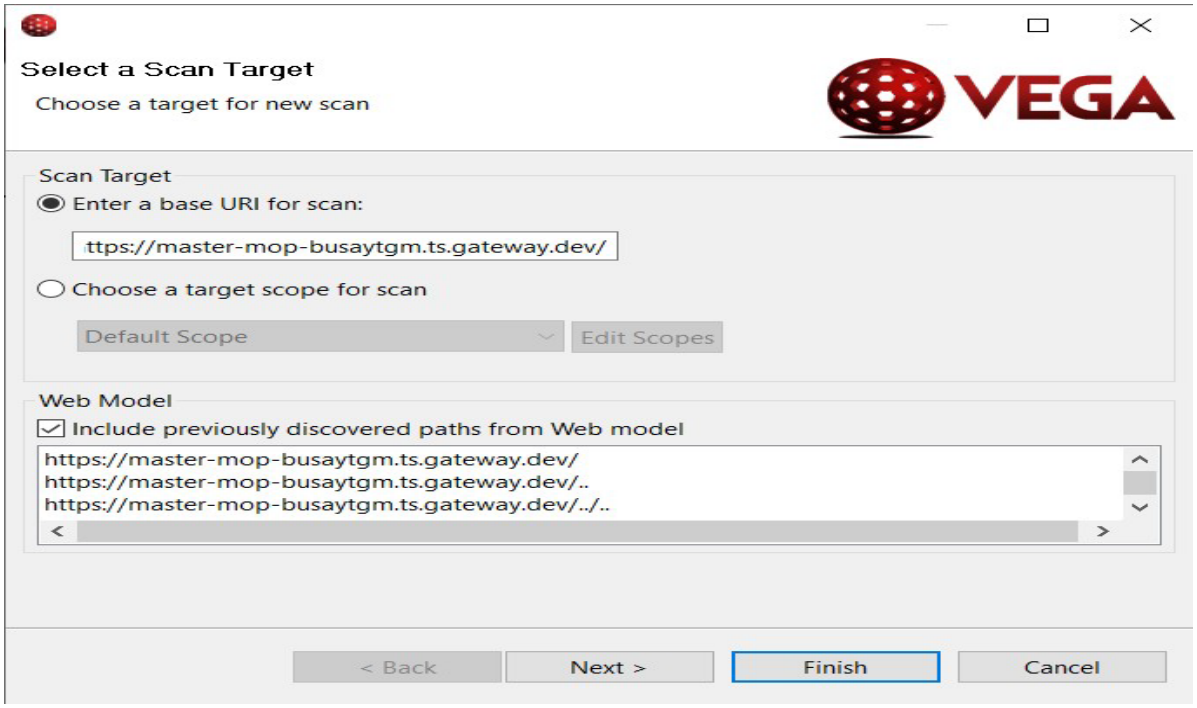


3. Set the Vega configuration

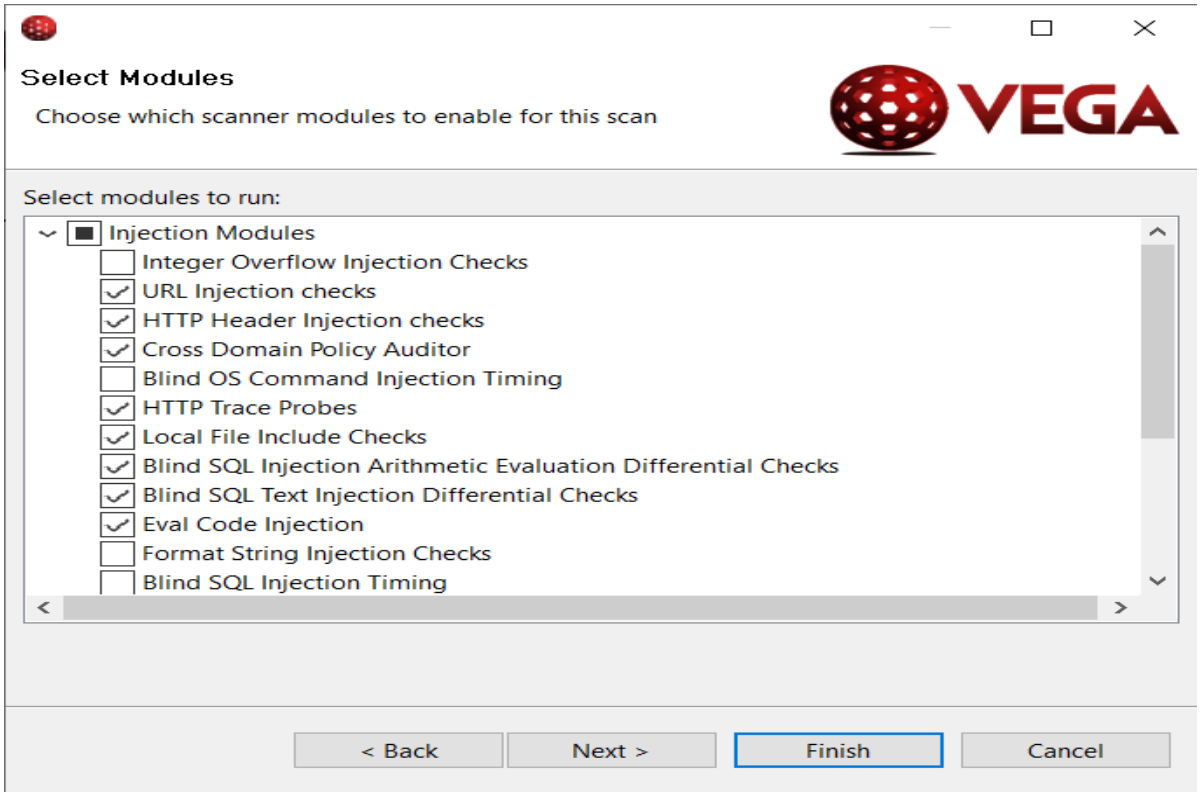
Before starting the scan, we can configure Vega according to requirements:

- Proxy Settings: Set up proxy settings if needed to route traffic through Vega.
- Scope: Define the scope of your scan by specifying the target website or web application's URL. You can also specify whether to include subdomains or specific paths.
- Authentication: If the website requires authentication, configure Vega to log in using the necessary credentials.
- Scan Policies: Configure scan policies and rules to specify which vulnerabilities you want to test for.
- Crawling Options: Adjust settings related to the website crawling process, such as the maximum depth and the delay between requests.

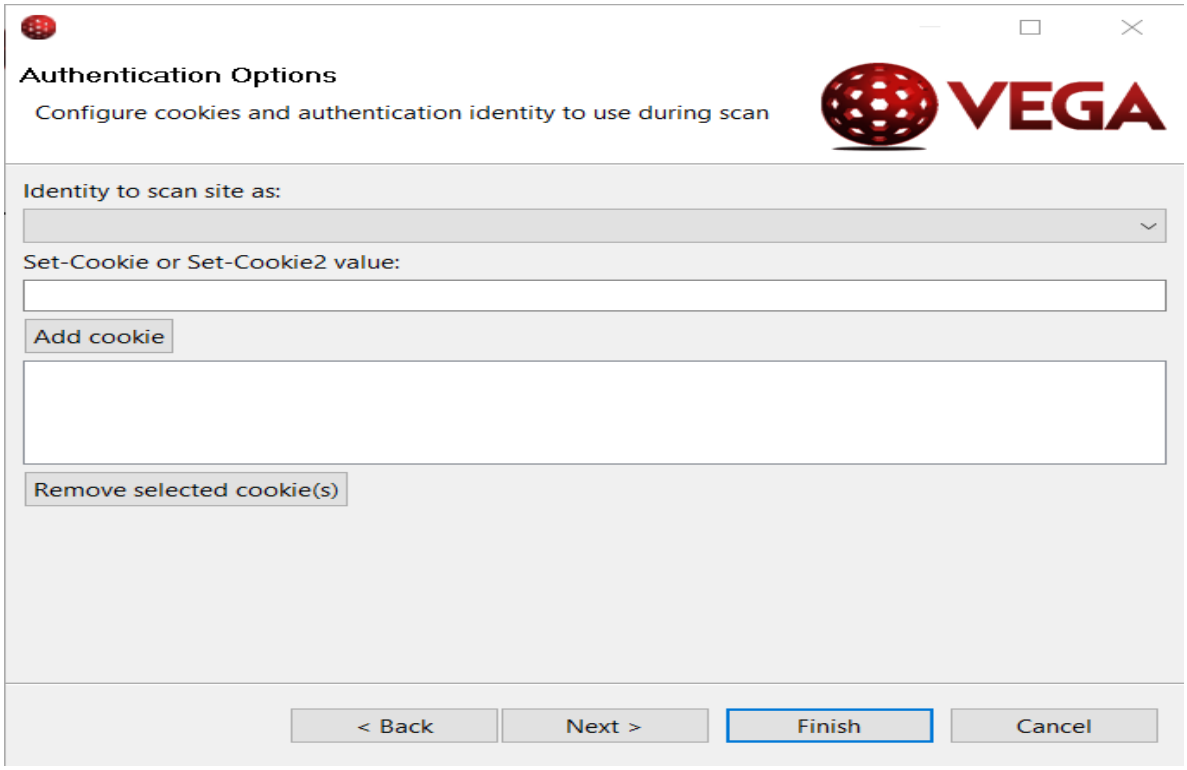
Once you've configured Vega, we can copy and paste the website url into the input field provided for scanning the target and initiate the scan in the GUI, by clicking the "Finish" button. See figure below



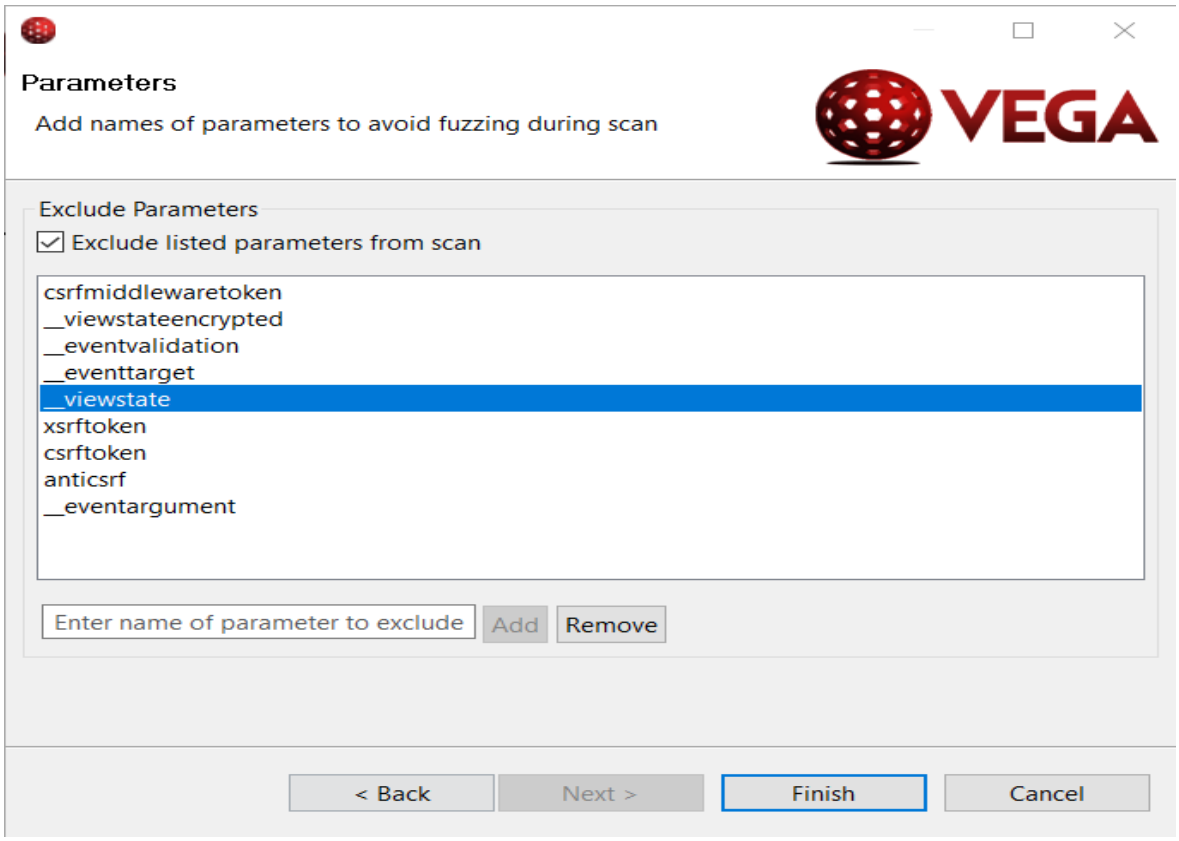
And if you want to specify the modules to run during the scanning you can press next button and select the required modules that that you want. See below figure



You can also configure the authentication options such as cookie configuration scanning identity.

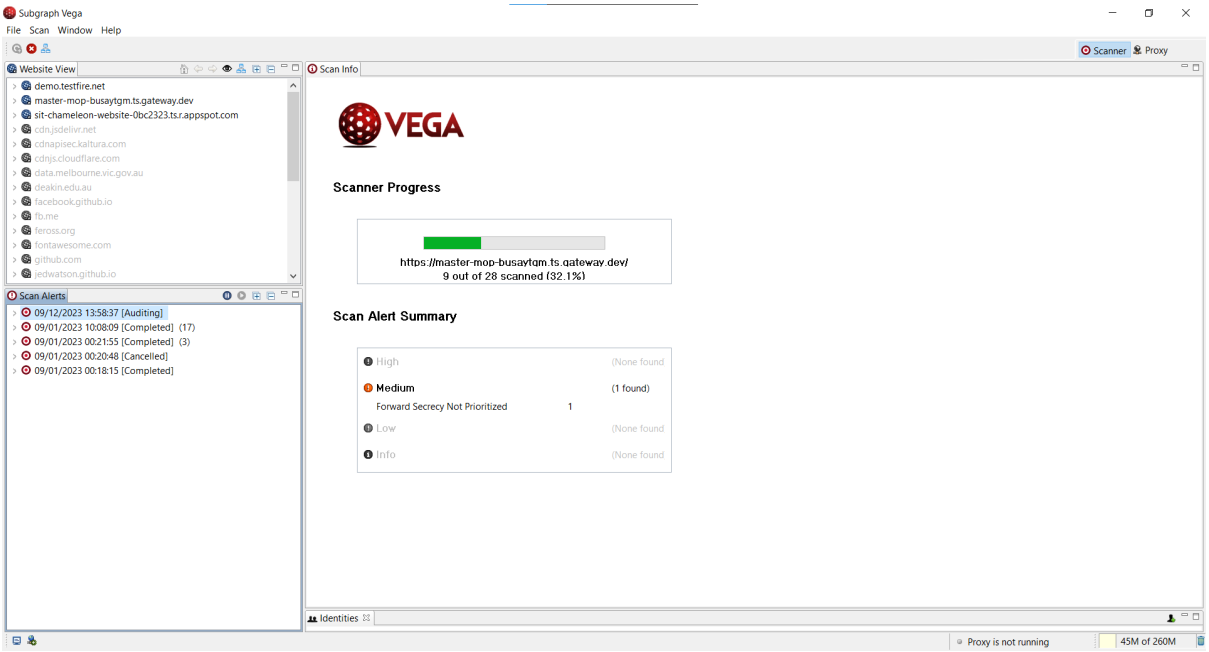


If you would like to exclude or add some parameters during the scanning inorder to avoid fuzzing you can do it as well. see below figure



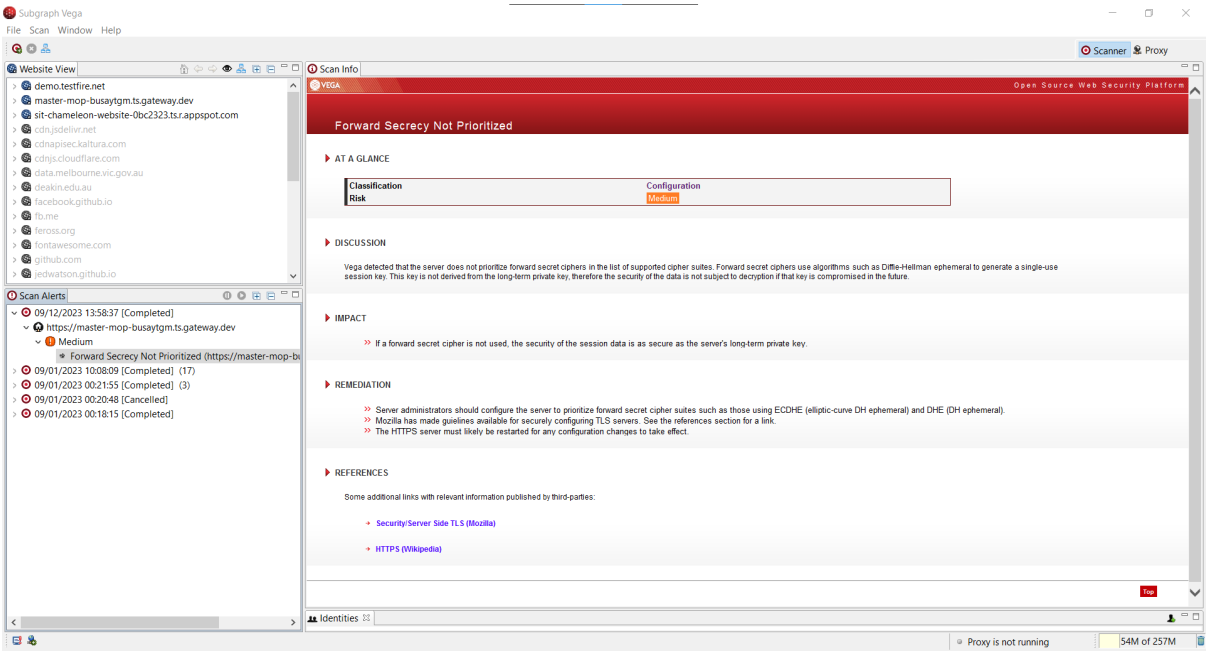
4. Monitor the Scan

Then you can press the finish button and it's going to start the scanning during the scan, Vega will start crawling the target website and perform security tests based on your configuration. You can monitor the progress of the scan and view the vulnerabilities detected in real-time.

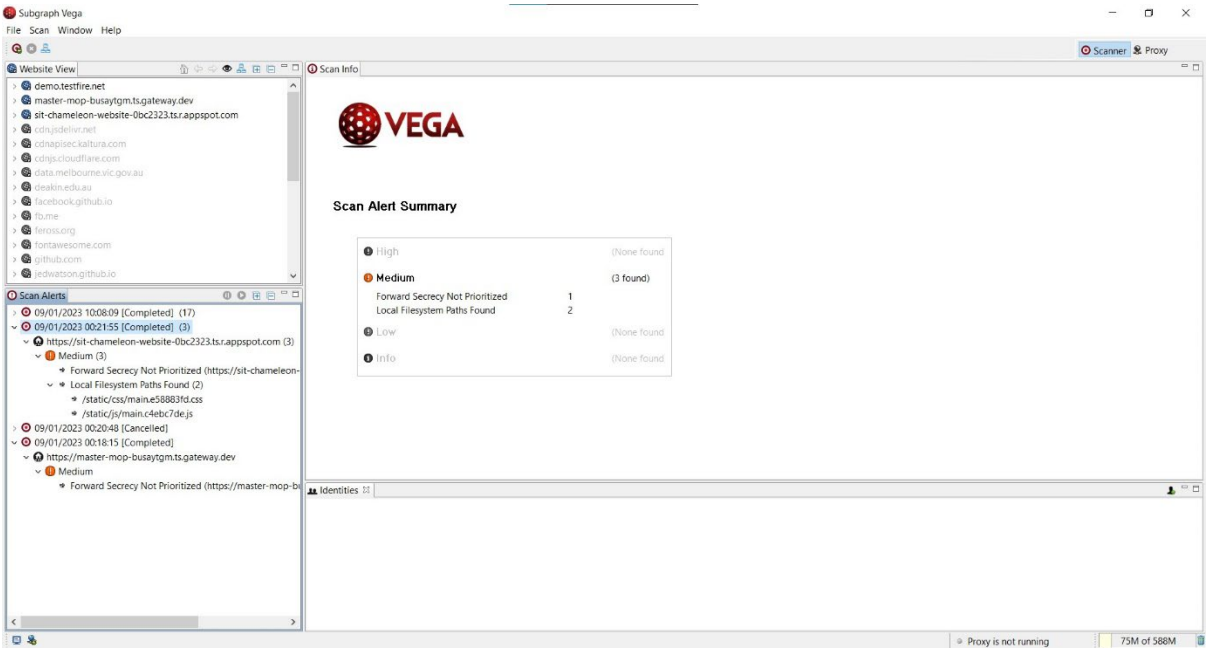


5. Review the Results

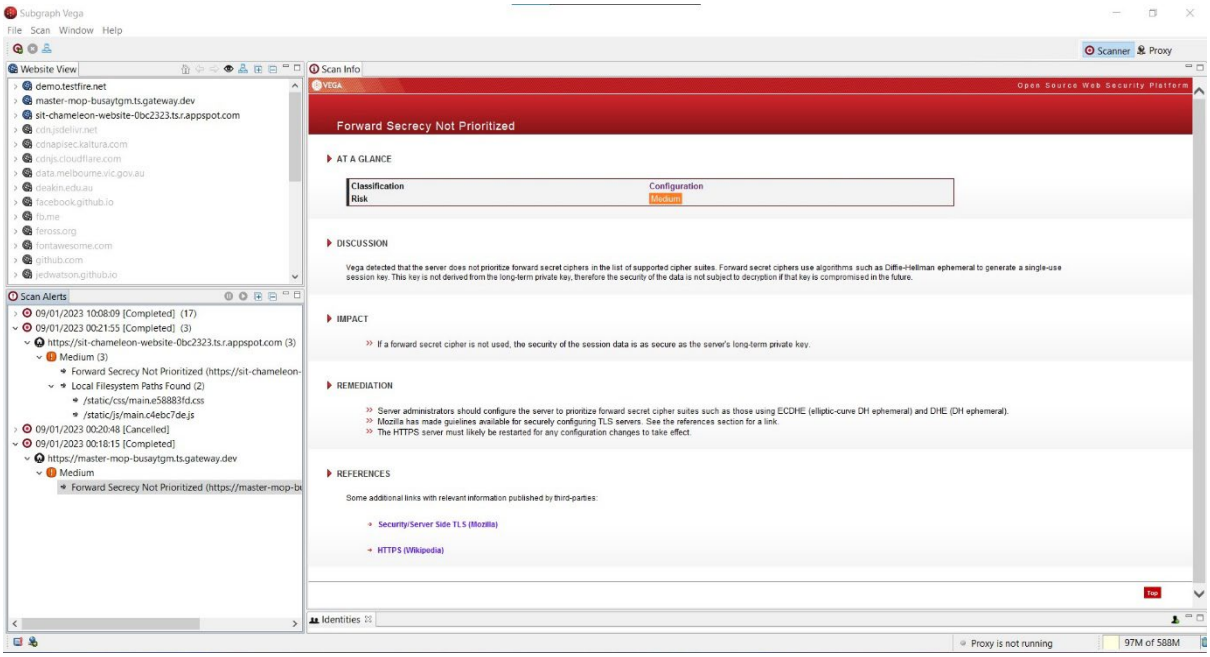
After the scan is complete, Vega will generate a report highlighting any vulnerabilities found. It will provide details about the vulnerabilities, including their severity and recommendations for remediation.



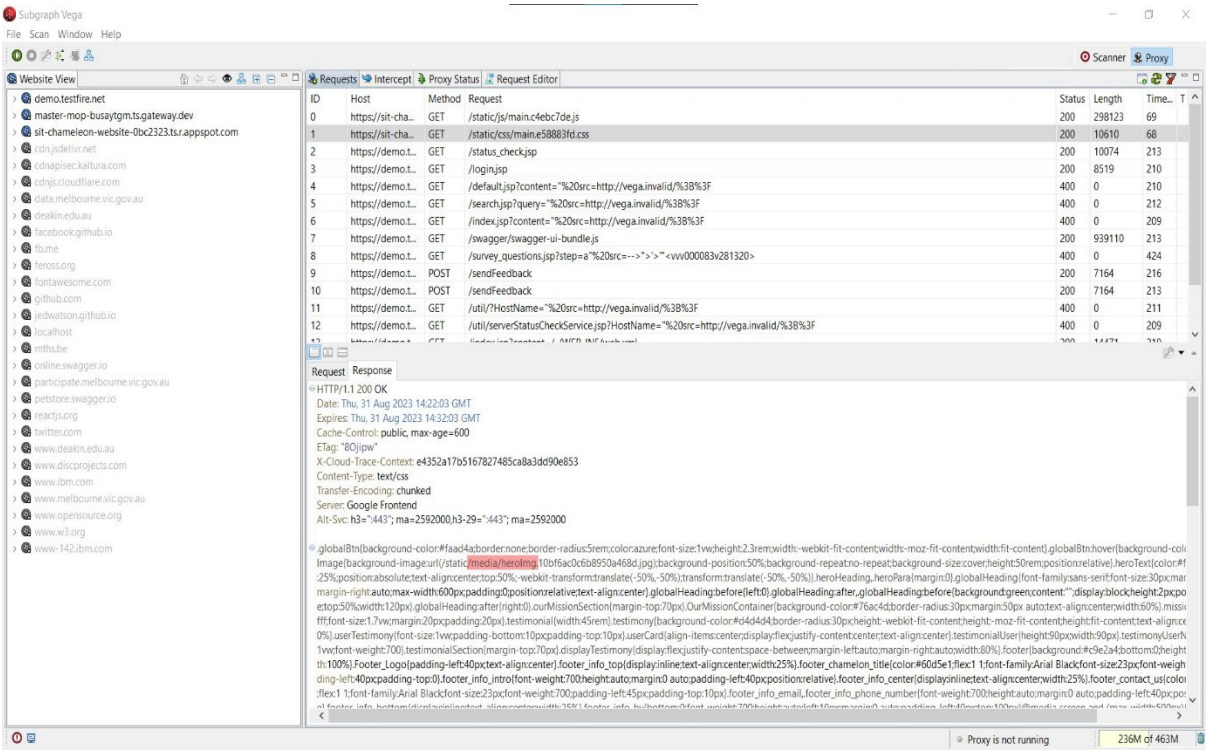
Here are some of the results of the MOP and chameleon website vulnerability scanning details. In the below screenshot we can see that the severity towards having vulnerability is medium.

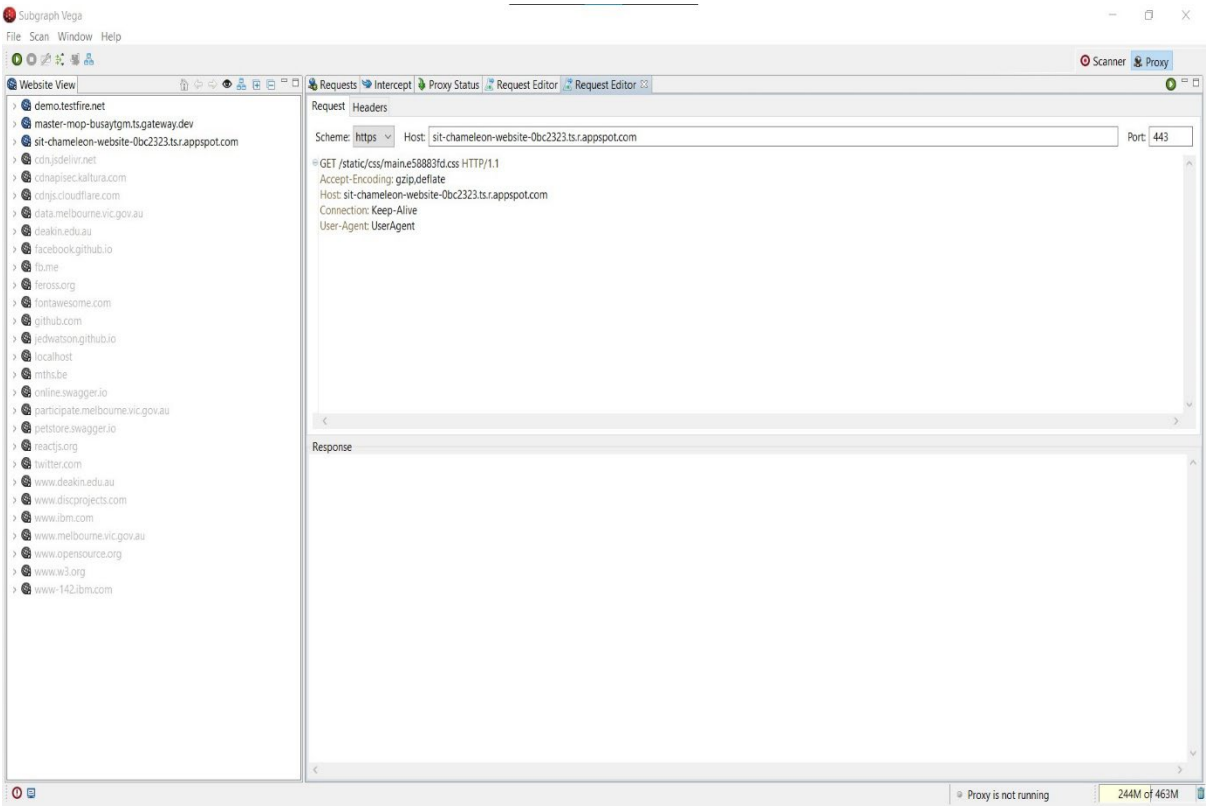


After scanning of the chameleon website, we can see that the forward secret cipher is not used, so it impacts and remediation is shown in the below screenshot.



In the proxy section we can handle the GET and POST request that is being send through the website during events like logging in and get the information passed.


Subgraph Vega
File Scan Window Help
Scanner
Proxy
Website View
Requests
Intercept
Proxy Status
Request Editor
ID Host Method Request Status Length Time...
0 https://sit-cha... GET /static/js/main.c4ebc7de.js 200 298123 69
1 https://sit-cha... GET /static/css/main.e58883fd.css 200 10610 68
2 https://demo.t... GET /status_check.jsp 200 10074 213
3 https://demo.t... GET /login.jsp 200 8519 210
4 https://demo.t... GET /default.jsp?content="%20src=http://vega.invalid/%3B%3F 400 0 210
5 https://demo.t... GET /search.jsp?query="%20src=http://vega.invalid/%3B%3F 400 0 212
6 https://demo.t... GET /index.jsp?content="%20src=http://vega.invalid/%3B%3F 400 0 209
7 https://demo.t... GET /swagger/swagger-ui-bundle.js 200 939110 213
8 https://demo.t... GET /survey_questions.jsp?step=a"%20src=-->">'>"<vvv000083v281320> 400 0 424
9 https://demo.t... POST /sendFeedback 200 7164 216
10 https://demo.t... POST /sendFeedback 200 7164 213
11 https://demo.t... GET /util/?HostName="%20src=http://vega.invalid/%3B%3F 400 0 211
12 https://demo.t... GET /util/serverStatusCheckService.jsp?HostName="%20src=http://vega.invalid/%3B%3F 400 0 209
Request
Response
HTTP/1.1 200 OK
Date: Thu, 31 Aug 2023 14:22:03 GMT
Expires: Thu, 31 Aug 2023 14:32:03 GMT
Cache-Control: public, max-age=600
ETag: "80jipw"
X-Cloud-Trace-Context: e4352a17b5167827485ca8a3dd90e853
Content-Type: text/css
Transfer-Encoding: chunked
Server: Google Frontend
Alt-Svc: h3=":443"; ma=2592000,h3-29=":443"; ma=2592000
Proxy is not running
236M of 463M


Subgraph Vega
File Scan Window Help
Scanner
Proxy
Website View
Requests
Intercept
Proxy Status
Request Editor
Request Editor
Request
Headers
Scheme: https
Host: sit-chameleon-website-0bc2323.ts.r.appspot.com
Port: 443
GET /static/css/main.e58883fd.css HTTP/1.1
Accept-Encoding: gzip,deflate
Host: sit-chameleon-website-0bc2323.ts.r.appspot.com
Connection: Keep-Alive
User-Agent: UserAgent
Response
Proxy is not running
244M of 463M

Here are some of the other vulnerabilities found in the chameleon and MOP website.