



IE4040
INFORMATION ASSURANCE &
AUDITING
4rd Year, Semester I

Assignment
IAA-Assignment

Submitted to
Sri Lanka Institute of Information Technology

In partial fulfillment of the requirements for the
Bachelor of Science Special Honors Degree in Information Technology

08/05/2020

TABLE OF CONTENT

1. Introduction	4
1.1. Audit.....	4
1.2. IT Audit.....	4
1.2.1. Importance of IT Audit.....	5
2. Web Application Audit.....	6
2.1. Audit Scope.....	6
2.2 Audit CheckList.....	6
2.1. Web application auditing tools.....	7
3. Vega Vulnerability Scanner	8
3.1. Vega Vulnerability Scanner Features.....	8
3.2. Installing Vega in a windows environment.....	9
3.3. Start the Vega Scanner.....	9
3.4. Configure the Vega Scanner.....	10
3.5. Scan a Website with Vega.....	12
3.6. Vega Alert Report.....	19
3.7. Summery.....	21
4. Reference.....	22

LIST OF FIGURES

Figure 3.2.1 Vega Application.....	9
Figure 3.3.2 Vega Scanner and Vega Proxy.....	9
Figure 3.4.1 preferences menu.....	10
Figure 3.4.2 Preferences.....	10
Figure 3.4.3. Scanner Preferences.....	11
Figure 3.4.4. General Preferences.....	12
Figure 3.5.1 New Scan wizard.....	12
Figure 3.5.2 Selecting the target URI for scan.....	13
Figure 3.5.3 Editing the target scope.....	14
Figure 3.5.4 selecting modules.....	15
Figure 3.5.5 Authentication Options.....	16
Figure 3.5.6 Parameters.....	17
Figure 3.5.7 Start the scanning.....	18
Figure 3.5.8 Vega Console.....	18
Figure 3.6.1 Vega Alert.....	19
Figure 3.6.2 website view.....	19
Figure 3.6.3. Scan Alerts.....	20

1. INTRODUCTION

1.1. AUDIT

Audit is specified in the sense that a process or quality system complies with requirements on-site verification activity, for example inspection or review. An audit may extend or may be unique to a task, method or development step of the entire organization. There are specific administrative reasons for such audits, such as audit records, risk, performance or the follow-up to corrections. In general, an audit is an investigation of an existing system and generate a report about the current status. [1]

1.2. IT AUDIT

The IT audit is an investigation of an organization information technology infrastructure, policies and operations. It also adds an evaluation, to suggest improvements. IT audits should assess whether IT policies secure company assets, guarantee data integrity and are consistent with overall business objectives. [2]

Some of IT audit categories,

1. **“Systems& Applications:** This focuses on the systems and applications within an organization. It makes sure they are appropriate, efficient, valid, reliable, timely and secure on all levels of activity.
2. **Information Processing Facilities:** Verifies that process is working correctly, timely and accurately, whether in normal or disruptive conditions.
3. **Systems Development:** To see if those systems which are under development are being created in compliance with the organization’s standards.
4. **Management of IT and Enterprise Architecture:** Making sure that IT management is structured and processes in a controlled and efficient manner.
5. **Client/Server, Telecommunications, Intranets and Extranets:** This spotlights telecommunication controls, such as a server and network, which is the bridge between clients and servers.” [3]

IT audit goals concentrate on current internal controls and function to reduce business risk as planned. Such audit priorities include ensuring compliance with the standards of legislation and regulations, and confidentiality, integrity, data availability and information systems.

1.2.1. Importance of an IT audit[4]

I. Reduces Risks Related to IT

Some of the main benefits of IT auditing is that it can help mitigate risk related to information technology process and network functionality, integrity and confidentiality. The reliability, performance and performance of IT systems can also be enhanced by covering a wide range of risks by continuous risk assessment and risk analysis within an organization. Therefore, once the risk is discovered, the audit helps the IT security team to mitigate and handle this risk.

II. Improves Security of Data

Once the risk is identified using an audit, an organization can improve its security structure to ensure that the data of the organization is not exposed to outsiders. In turn, it enables businesses to rethink or reinforce poorly designed or inefficient controls, thereby improving data security.

III. Fraud Detection and Prevention

IT auditing helps to avoid fraud from happening in businesses. A continuous review of the activities of a business and the implementation of strict internal control systems may avoid different forms of fraud and discrepancies in other accounts. Audit experts help to develop and change internal control processes which aim at preventing fraud.

IV. Enhances IT Governance

In order to ensure compliance with all company rules, legislation, and IT department enforcement, the IT audit plays an important role. When IT management understands well the risks, controls and importance of the technology environments of a organization, this strengthens IT governance.

2. WEB APPLICATION AUDIT

Web application audit is the best way to ensure that your application is secure and prevents hacking your web application. It's easy to find out how to get things up to speed by knowing components, taking security issues and assessing results.

The goal of the web app review is to review the codebase of an application to decide whether the code does anything that it should not do. Audits will also determine how code can be treated so as to do something wrong and whether programs can clearly communicate confidential data. A superior web application audit will define the correct safety measures implemented by developers.[5]

Risks of not securing your web applications,[6]

- Sales loss or even business shut down.
- Judgments, legal costs and regulatory penalties.
- Damages for reputation loss.
- Problems accepting payment cards.
- Economic losses by fraud.

Goals of periodical website audit,

- Identify vulnerabilities and potential security breaches.
- Analyze your website security status as seen by potential attackers.
- Determine the real business risks for all the players of your company.

2.1. Audit Scope

I am going to audit <https://fritzing.org/> website. Fritzing is an open-source hardware initiative that makes electronics accessible as a creative material for anyone. they offer a software tool, a community website, and services in the spirit of Processing and Arduino, fostering a creative ecosystem that allows users to document their prototypes, share them with others, teach electronics in a classroom, and layout and manufacture professional PCBs. I am going to audit this website to find any security risks of this website. Also, I did not have any credentials to access critical information and resources of this site. To audit this website I use Vega vulnerability scanner.

2.2. Audit Checklist

Bash environment variable blind OS injection checks

XSS injection checks

Remote file include Checks

Local file include Checks

Cross domain policy auditor checks
Shell injection checks
Eval code injection checks
HTTP trace probes checks
Blind SQL text injection differential checks
HTTP header injection checks
URL injection checks
XML injection checks
HTTP Authentication over Uncrypted HTTP checks
HTTP Header checks
Insure Script checks
Internal IP Address checks
Sourecode disclosure Module checks
Cookie Securty Module checks

2.3. Web application auditing tools

Web application security audits can be performed automatically, using commercially available tools, as well as manually or open source applications, going over each separate application module. Scanners don't have access to source code, only do feature checking and try to find security vulnerabilities.

Ex :- VEGA

Grabber

Zed Attack Proxy

Wapiti

W3af

3. VEGA VULNERABILITY SCANNER

Vega is a web security scanner free and open source and web security testing tool for web applications security testing. Vega is written in Java and offers a GUI based environment. It is available for OS X, Linux and Windows. Also Vega was developed by subgraph.

Most important thing is Vega can find SQL injection, header injection, directory listing, shell injection, cross site scripting, file inclusion and other web application vulnerabilities. This tool can also be extended using a powerful API written in JavaScript. Vega also probes for TLS / SSL security settings and identifies opportunities for improving the security of your TLS servers.[7]

Vega includes an automatic scanner for quick testing and an interim proxy for tactical testing. Vega provide Vega Scanner, Vega Proxy, Proxy Scanner and also Scanner with credentials.

3.1. Vega Vulnerability Scanner Features

- **Automated Scanner**

Vega includes a website crawler powering its automated scanner. Vega can automatically log into websites when supplied with user credentials.

- **Intercepting Proxy**

Vega can be used to observe and interact with communication between clients and servers, and will perform SSL interception for HTTP websites.

- **Proxy Scanner**

The Vega proxy can also be configured to run attack modules while the user browses the target link. This allows for semi-automated, user-driven security testing to ensure maximum code coverage.

- **GUI Based**

Vega has a well-designed graphical user-interface.

- **Multi-platform**

Vega is written in Java and runs on Linux, OS X, and Windows.

- **Extensible**

Vega detection modules are written in JavaScript. It is easy to create new attack modules using the rich API exposed by Vega.

3.2. Installing Vega in a windows environment

Vega packages 32 and 64bits for Windows, OS X or Linux can be downloaded at <https://subgraph.com/vega/download/>. After the downloading Vega setup, you can run the Vega install setup wizard and install the Vega in your computer. After completing the installation Vega application was launch as shown in figure 3.2.1.

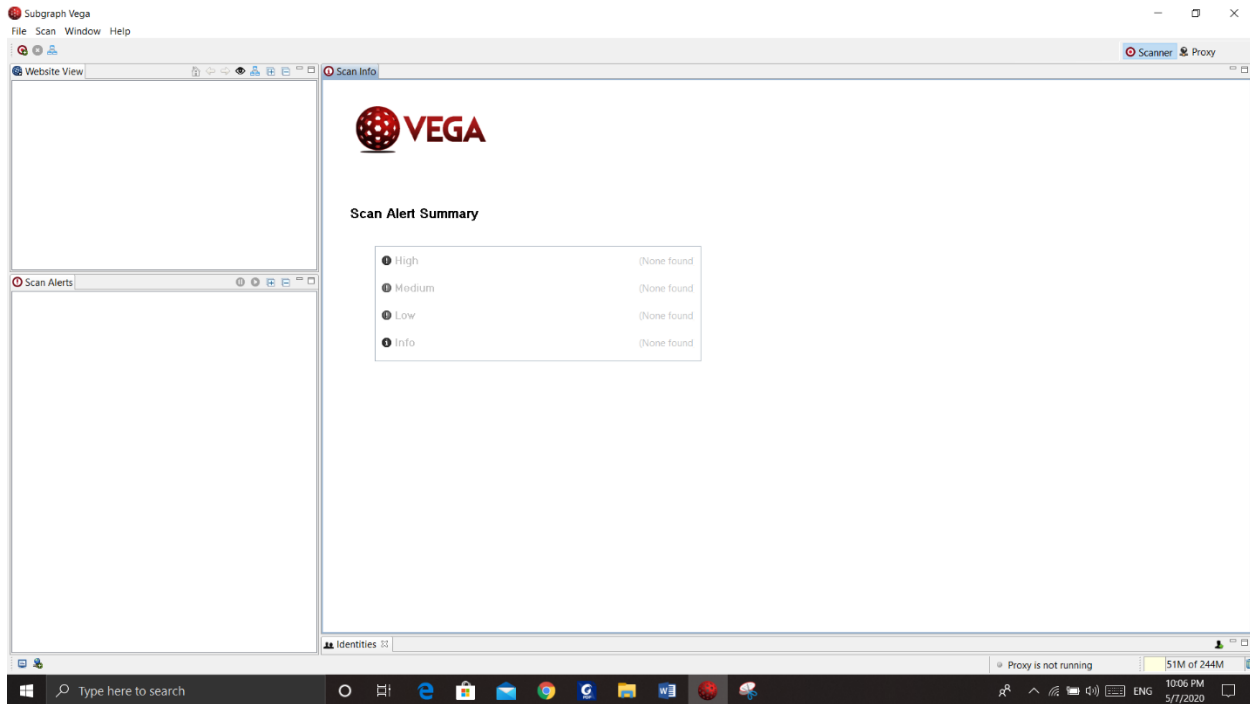


Figure 3.2.1 Vega Application

3.3. Start the Vega Scanner

After successfully installing the app, you can double click the Vega icon and open the app as shown in Figure 3.3.1. You see the Vega workspace from the scanner perspective, when you launch Vega for the first time. Normally Vega has two perspectives called Scanner and proxy as shown in figure 3.3.2.

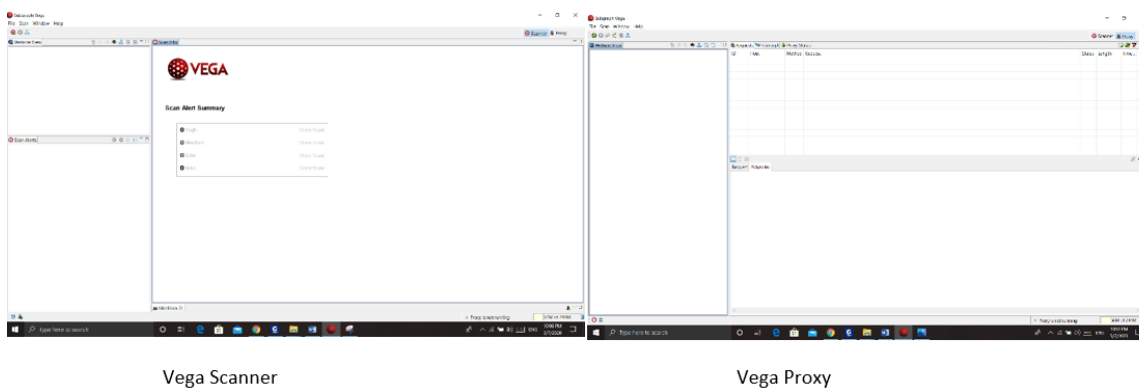


Figure 3.3.2 Vega Scanner and Vega Proxy

The original model for Vega in its work area is shown in Figure 1 above. In the panels Web View, Scan Alerts, Scan Info and Identities you can see items arranged in this style. These objects can be shifted, altered and the workspace can be modified completely. The user can at any time restore the original layout by selecting "Restore Perspective" from the window menu .

3.4. Configure the Vega Scanner

After start application in the first time, you can configure the available preferences in the window menu.

Windows menu → Preferences

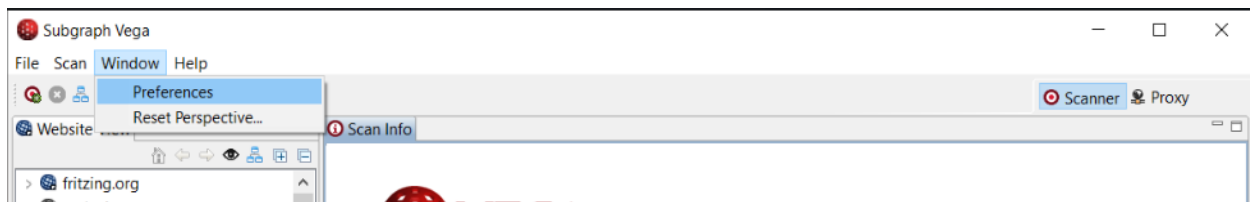


Figure 3.4.1 preferences menu

Under the preferences, there are three types of preferences as shown in figure 3.4.2. those are General, Proxy, and Scanner.

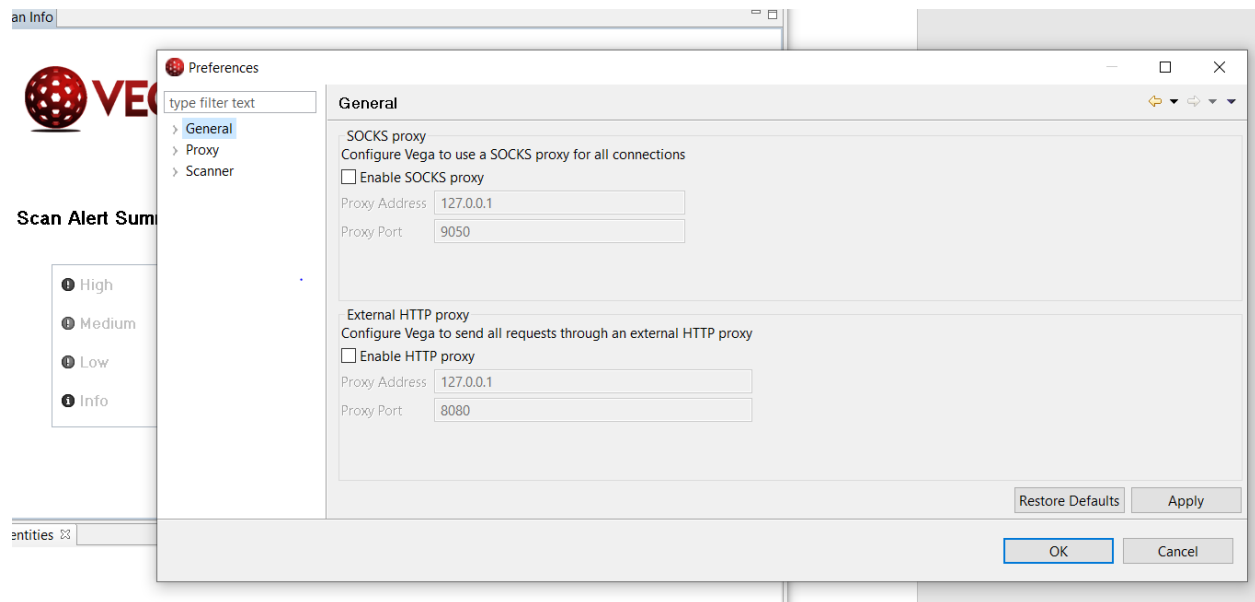


Figure 3.4.2 Preferences

Vega repetitively searches websites, generating an internal site representation in a tree-like data structure consisting of entities called "path State nodes." Path state nodes may include POST or GET parameter directories, files or directories. Complex websites can result in long scans and extensive data status systems, so that Vega provides configurable parameters that restrict the search range of preferences for the scanner as shown in figure 3.4.3.[8]

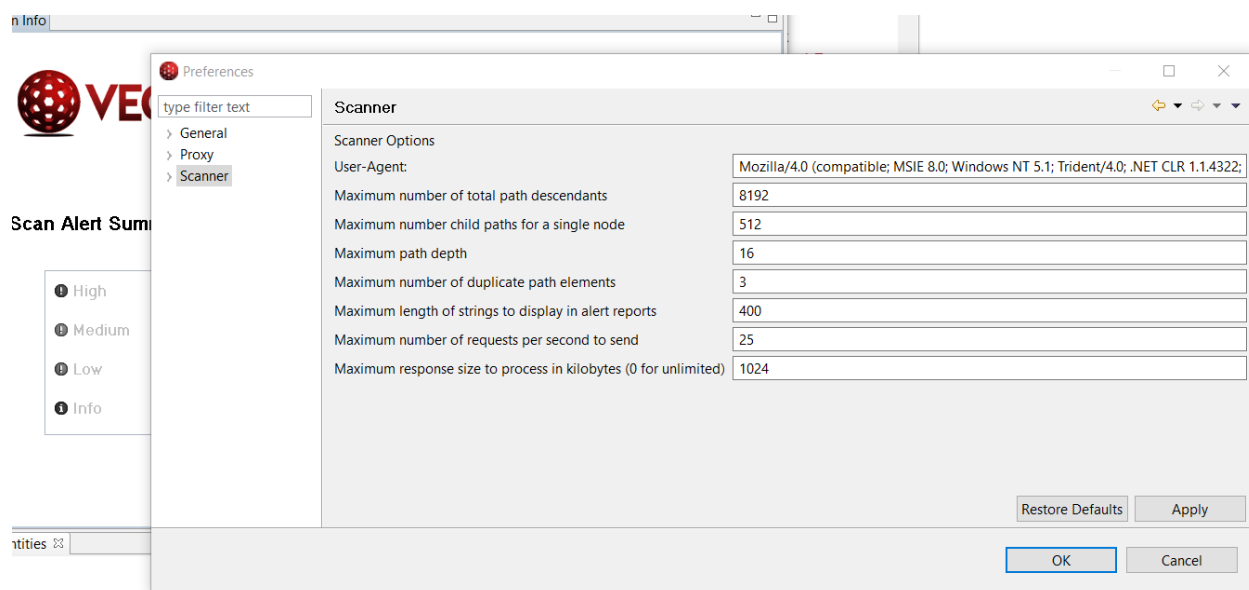


Figure 3.4.3. Scanner Preferences

- “Maximum number of Total number of path descendants

This is the total children of a node + all its children. Children of a path state node could be its subdirectories, or its parameters, with one node for each in a set of parameters.

- Maximum number of child paths for a single node

Limits on the number of children per node (subdirectories + files + parameters).

- Maximum path depth

The limit on the hierarchy of path state nodes (e.g. /level1/level2/level3/level4..)

- Maximum number of duplicate path elements

The maximum number of permitted duplicate, adjacent path nodes. For example: /images/images/images.

- Maximum length of strings to display in alert reports

The alerts can include text from the module, such as the response body. The level of permitted module verbosity can be configured here by the user.

- Maximum number of requests to send per second

This setting regulates the speed at which Vega scans.”[8]

In this report, we only used Scanner preferences only therefore we did not configure proxy preferences. Also in general, you prefer to anonymize your Vega scans and proxy all connections, check the "Enable SOCKS proxy" option under General, and enter a proxy address and port. This helps to hide the origin of your scans.

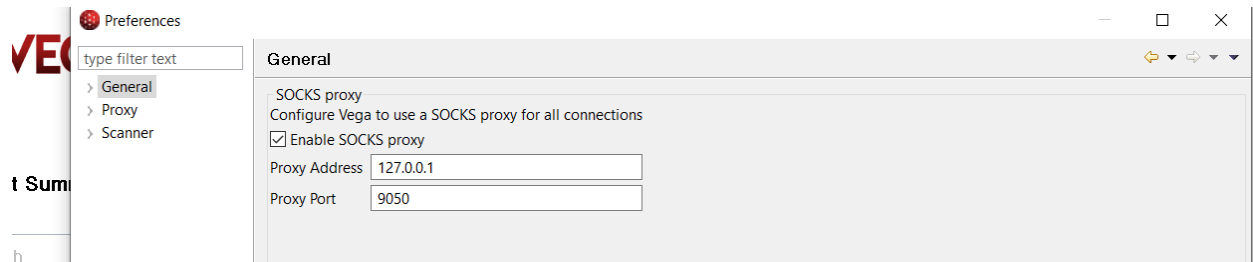


Figure 3.4.4. General Preferences

That's about it for Vega's preferences. The Listener settings in "Proxy" are preferences unrelated to the scanner. The "Debug" preferences in the Scanner section are there for Vega developers.

3.5. Scan a Website with Vega

After successfully configuring Vega, you can start auditing a website. Open the "Scan" menu in the top left corner to start the scan and click "Start New Scan." Vega prompts us through the Select Target window.

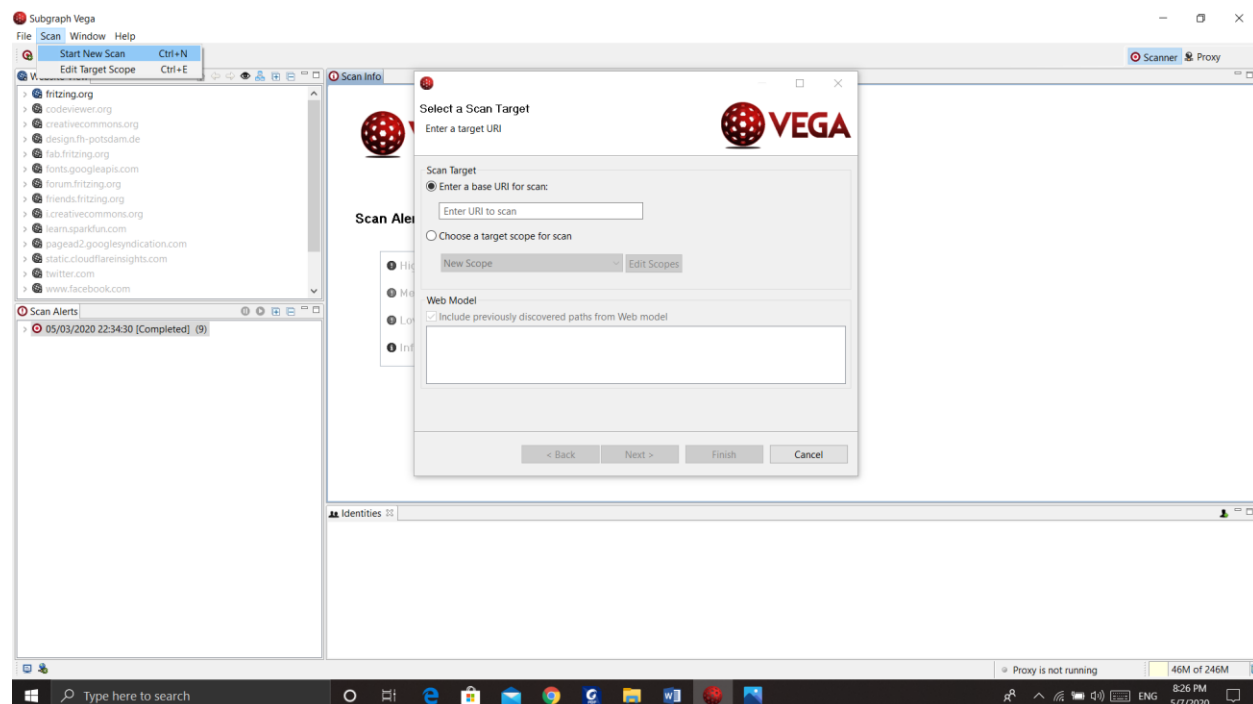


Figure 3.5.1 New Scan wizard

In here you can enter website URL as a target and click next. Otherwise, you can edit target scope by tick “choose a target scope for scans”.

Select a Scan Target
Choose a target for new scan

Scan Target

☒ Enter a base URI for scan:

☐ Choose a target scope for scan

New Scope Edit Scopes

Web Model

☒ Include previously discovered paths from Web model

< Back Next > Finish Cancel

Figure 3.5.2 Selecting the target URI for scan

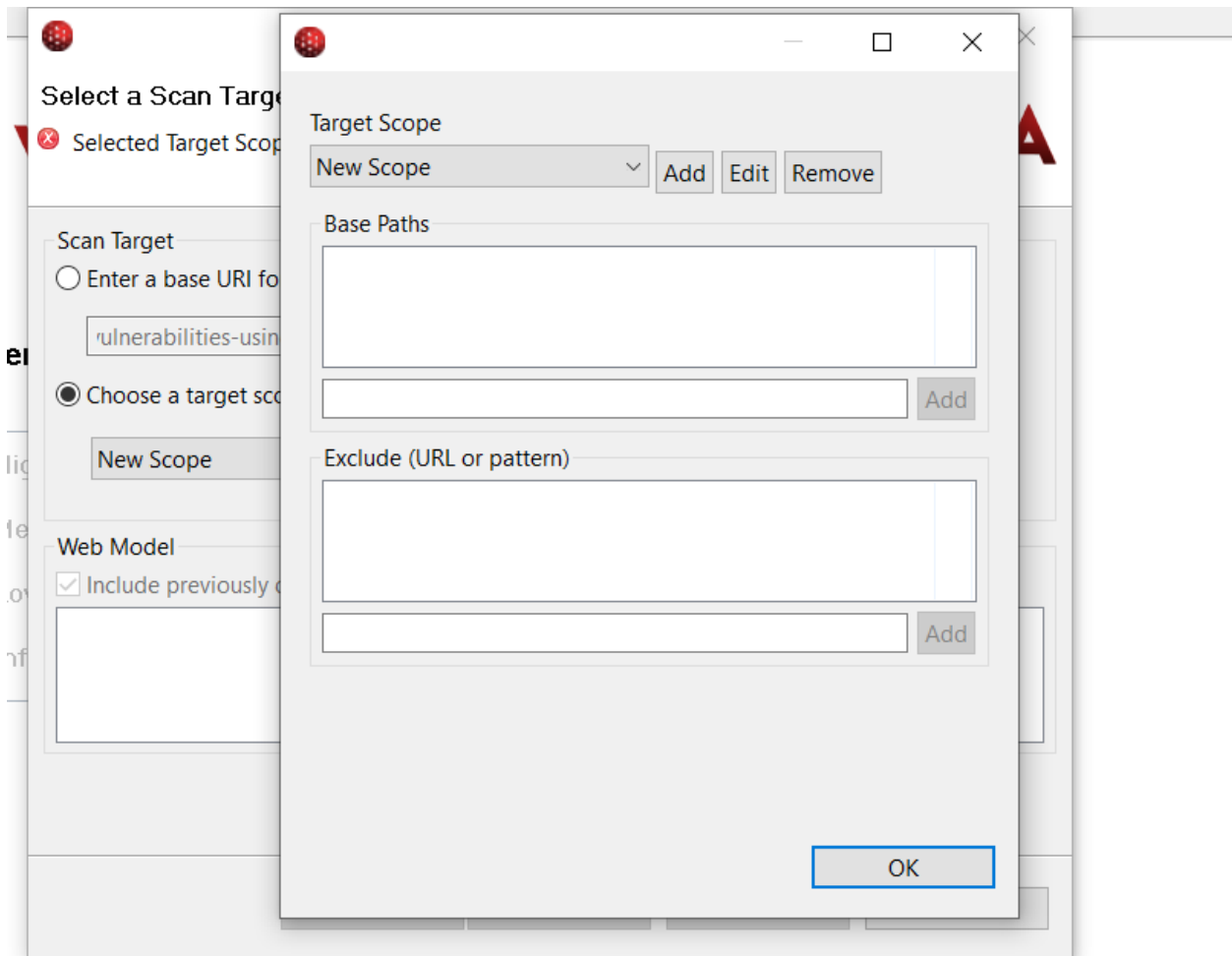


Figure 3.5.3 Editing the target scope

Target scopes allow multiple base URIs and exclusions that will not be scanned by Vega. Here you can add URL that needs to exclude and edit the target scope as you want.

If the "Next" button has been pressed, the user will be able to pick the Injection and Response Processing modules to use to scan the application. The modules are extended functionality units written in JavaScript as the Vega engine is in Java, which also contains the interpreter from Rhino JS. Vega therefore supports two types of modules: basic modules and response modules. Basic modules have options in the Inject module that operate on path state nodes, including URIs which are considered to be files or directories and URIs with parameters, each of which is a distinct path state node. Response processing modules are those which run on all server responses. Within the shared knowledge base, both modules will store information and generate alerts within XML format. For each of the two modules, we will explore the choices and pick our choices.[8]

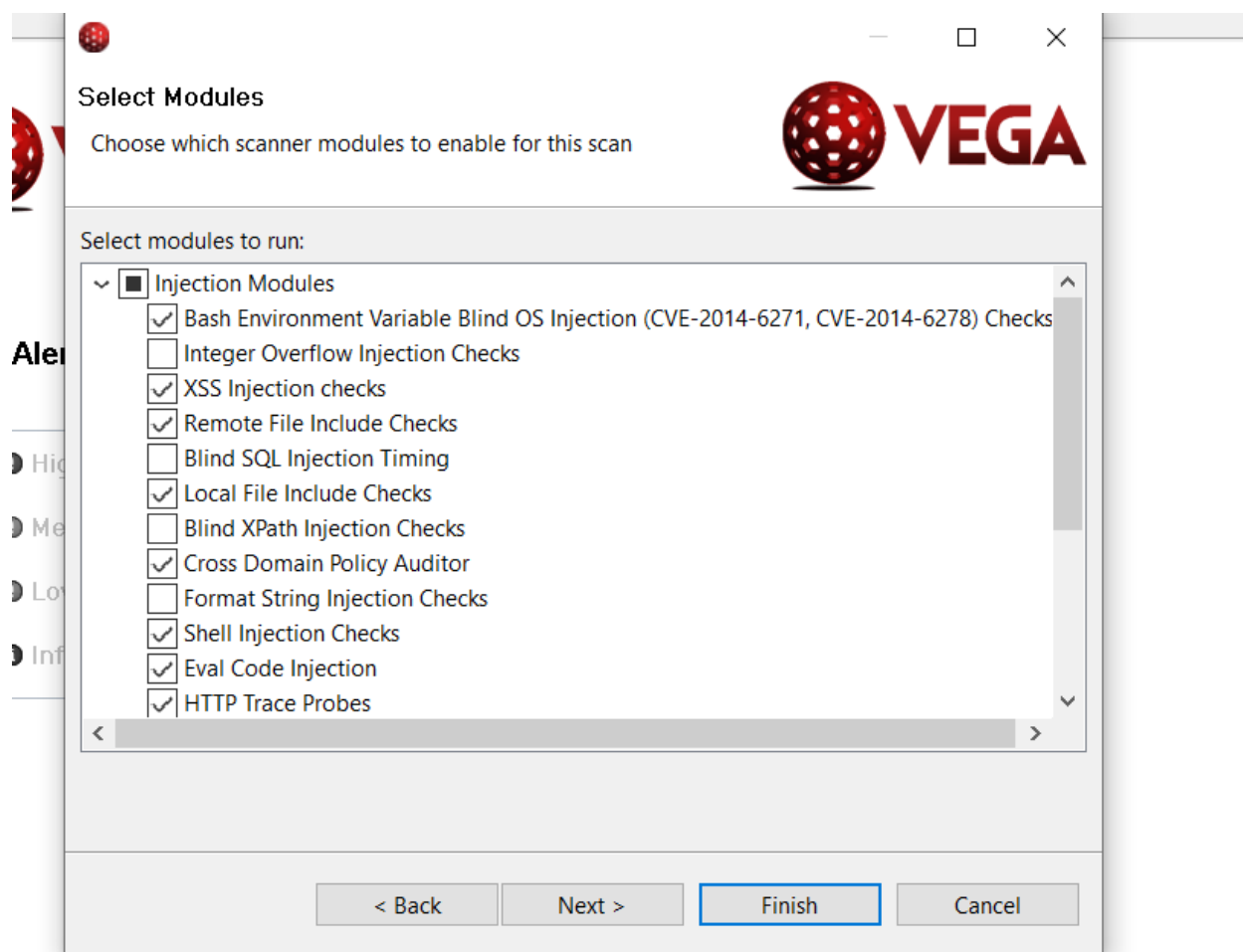


Figure 3.5.4 selecting modules

“Vega supports the configuration of credentials for performing automated scans while authenticated to the application or server. These credentials include:

Basic HTTP

Digest HTTP

NTLM

Macro (form based authentication)

Vega also permits the configuration of cookies that will be sent with all scanner requests.”[8]

Suppose a company employs you for a web review and they do not want to provide you with credentials to test whether you are able to find any vulnerabilities on their web resources without giving you any authorization. This is an important thing to remember, and so let us take this into account in our practice and do not complete any credentials information as shown in figure 3.5.5

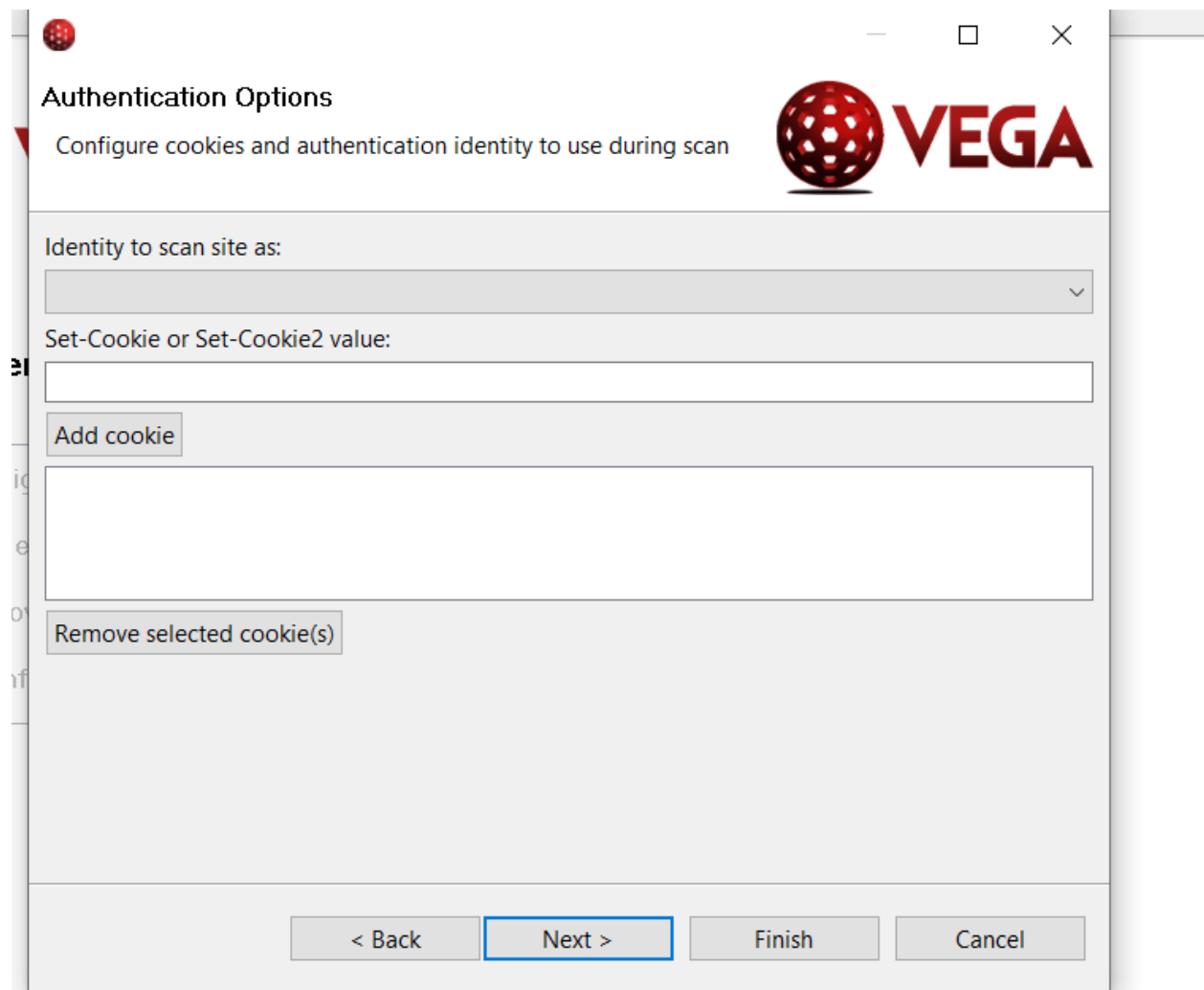
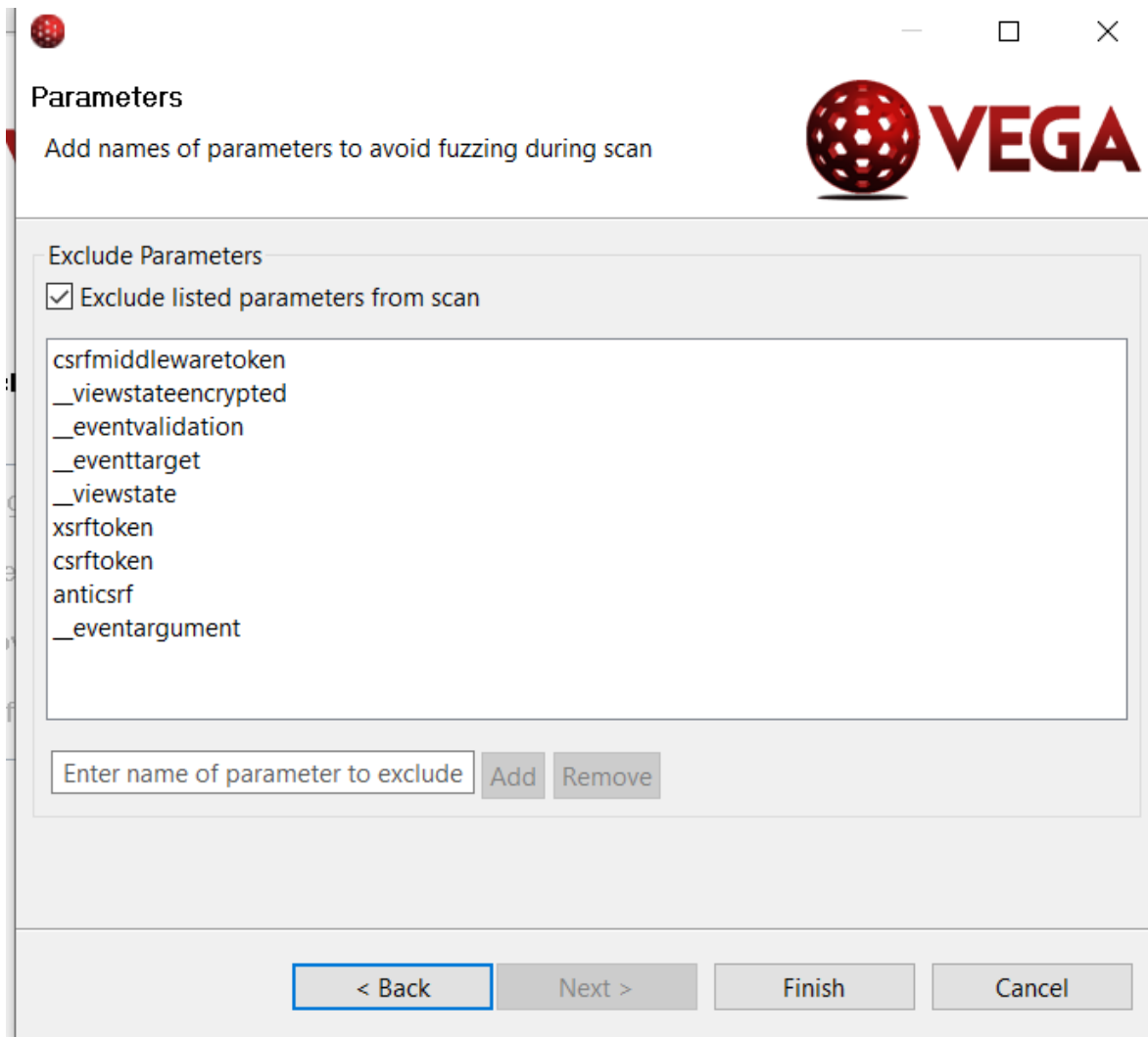
The image shows a screenshot of a software window titled "Authentication Options". The window has a standard Windows-style title bar with a red icon on the left and minimize, maximize, and close buttons on the right. Below the title bar, the text "Configure cookies and authentication identity to use during scan" is displayed. To the right of this text is the Vega logo, which consists of a red sphere with white dots and the word "VEGA" in bold red letters. The main area of the window contains several input fields and buttons. At the top, there is a label "Identity to scan site as:" followed by a dropdown menu. Below this is a label "Set-Cookie or Set-Cookie2 value:" followed by a text input field. Underneath the text input field is a button labeled "Add cookie". Below the "Add cookie" button is a larger text input field. At the bottom of this section is a button labeled "Remove selected cookie(s)". At the very bottom of the window, there are four buttons: "< Back", "Next >", "Finish", and "Cancel". The "Next >" button is highlighted with a blue border.

Figure 3.5.5 Authentication Options.

Depending on the value in these tools, parameters may be added or excluded. We leave these choices with the default selections for our tactical review as shown in figure 3.5.6



The screenshot shows a Windows-style dialog box titled "Parameters" with a red sphere icon in the top-left corner. The main title bar includes standard minimize, maximize, and close buttons. Below the title bar, the word "Parameters" is displayed in bold, followed by the instruction "Add names of parameters to avoid fuzzing during scan". In the top-right corner, there is a red sphere icon and the word "VEGA" in a bold, red, sans-serif font. The main content area is a light gray panel with a section titled "Exclude Parameters". Inside this section, there is a checked checkbox labeled "Exclude listed parameters from scan". Below the checkbox is a list box containing the following parameters: csrfmiddlewaretoken, __viewstateencrypted, __eventvalidation, __eventtarget, __viewstate, xsrftoken, csrftoken, antiscrf, and __eventargument. At the bottom of the list box area, there is a text input field with the placeholder "Enter name of parameter to exclude", followed by "Add" and "Remove" buttons. The bottom of the dialog box features four buttons: "< Back" (highlighted with a blue border), "Next >", "Finish", and "Cancel".

Figure 3.5.6 Parameters

After this click finish button, then the scan will start as shown in figure 3.5.7.

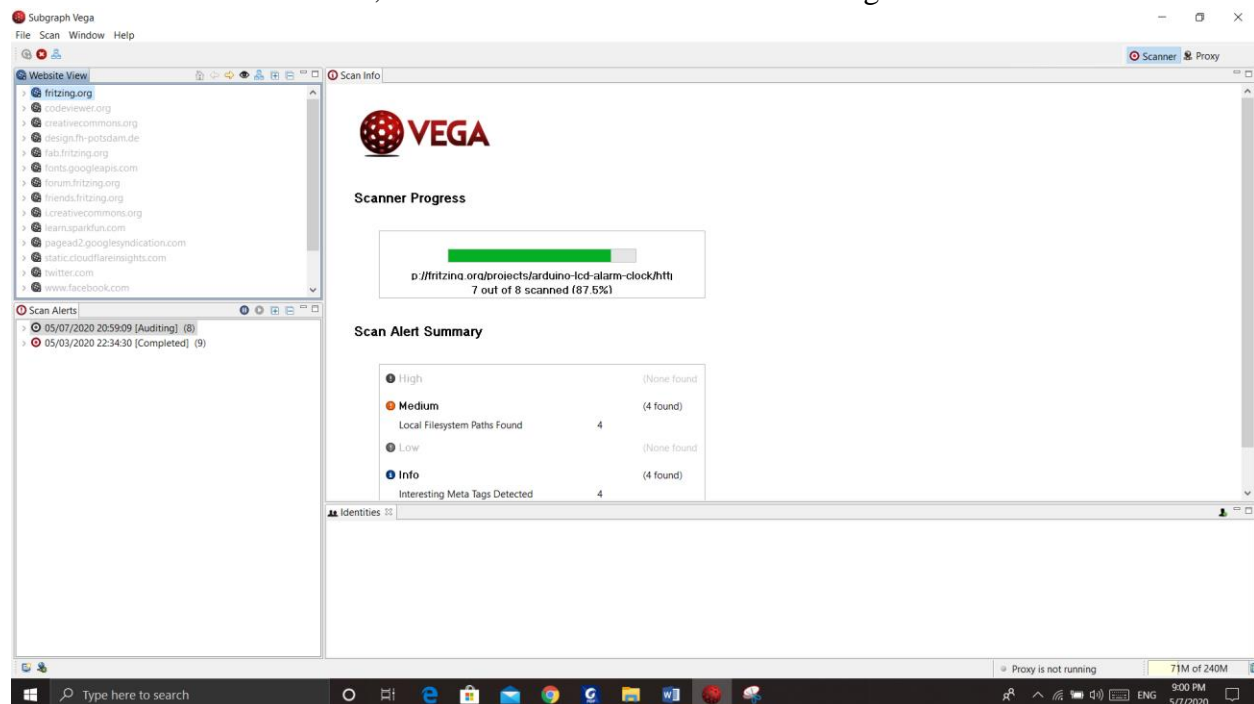


Figure 3.5.7 Start the scanning

As the scan begins, Vega displays progress in a progress bar inside the panel item "Scan Info." Track the check that Vega conducts.

While Scanning the website you can look at the Vega console. It will display entire content that Vega has published into Vega log can be seen.

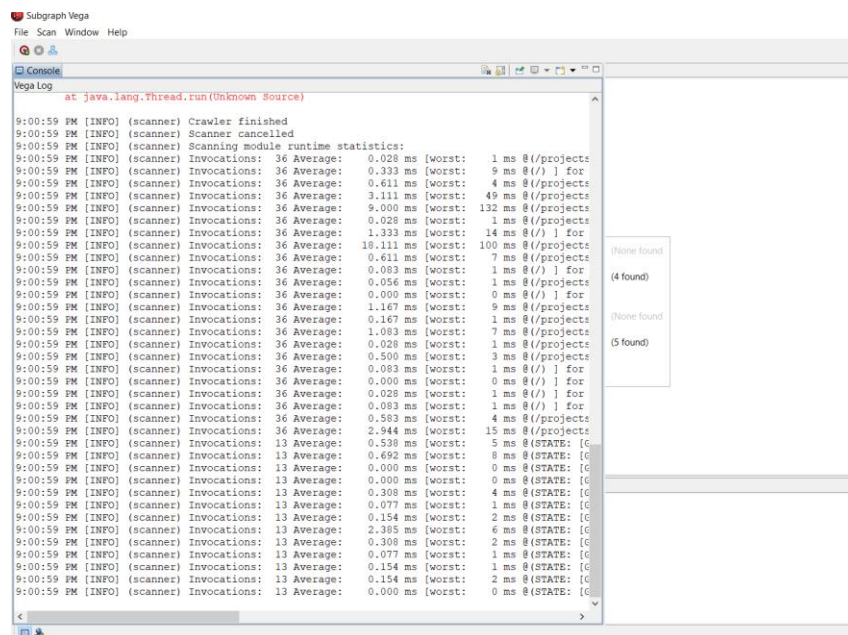


Figure 3.5.8 Vega Console

3.6. Vega Alert Report

After completing the scanning Vega is generating alert report as shown in figure 3.6.1. In this report you can see risks are categorized in four types.

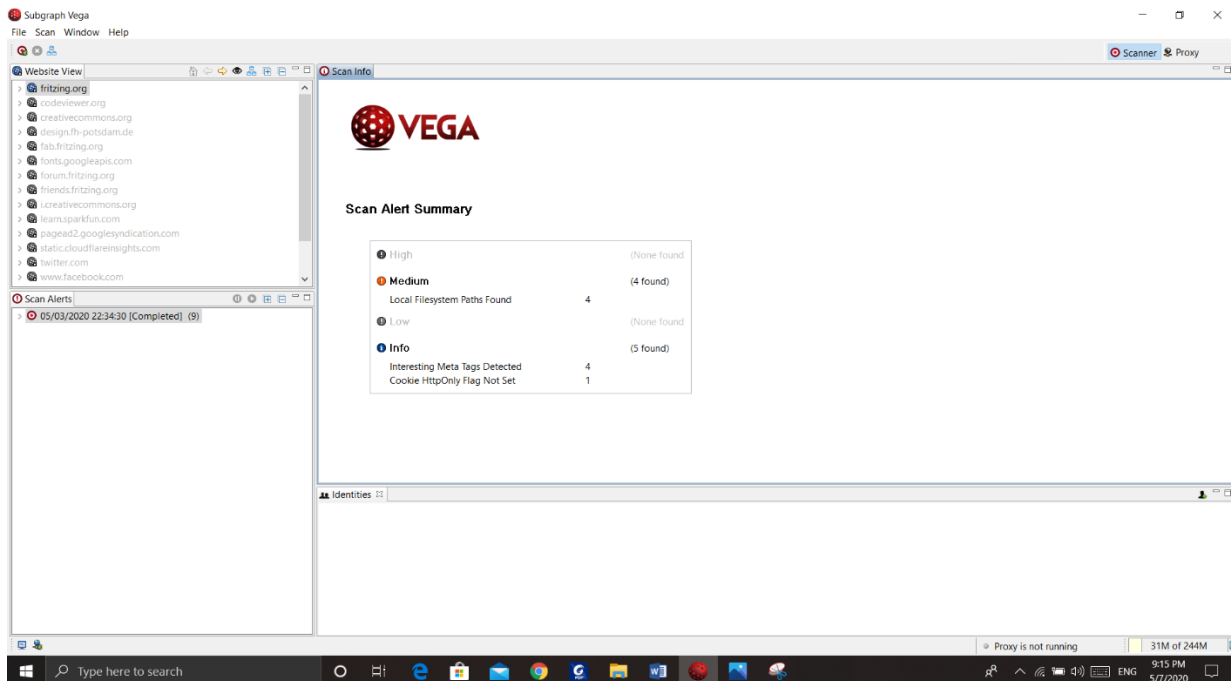


Figure 3.6.1 Vega Alert

“Vega will build a list in the top right corner of the paths crawled and seen. The greyed-out paths are those that have not been accessed. Vega will not crawl links on other websites”. [8]

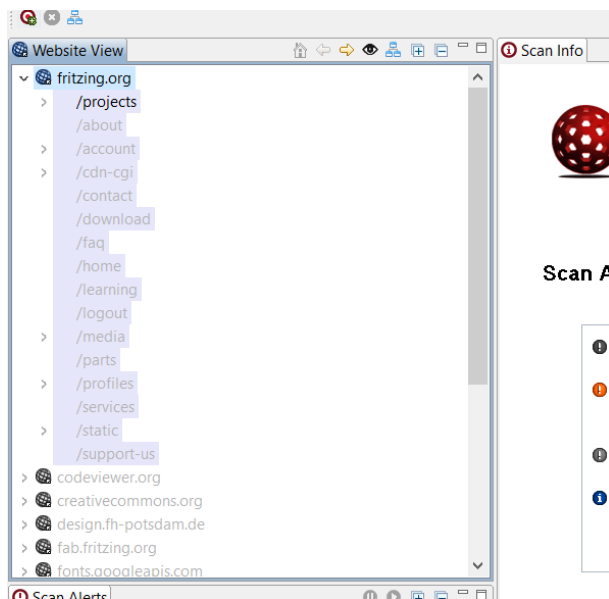


Figure 3.6.2 website view

In the Scan-alert box give us to path state node and the alert report of that node as shown in figure 3.6.3. That alert report is included at a glance, request, resource content, discussion, impact, remediation, and references.

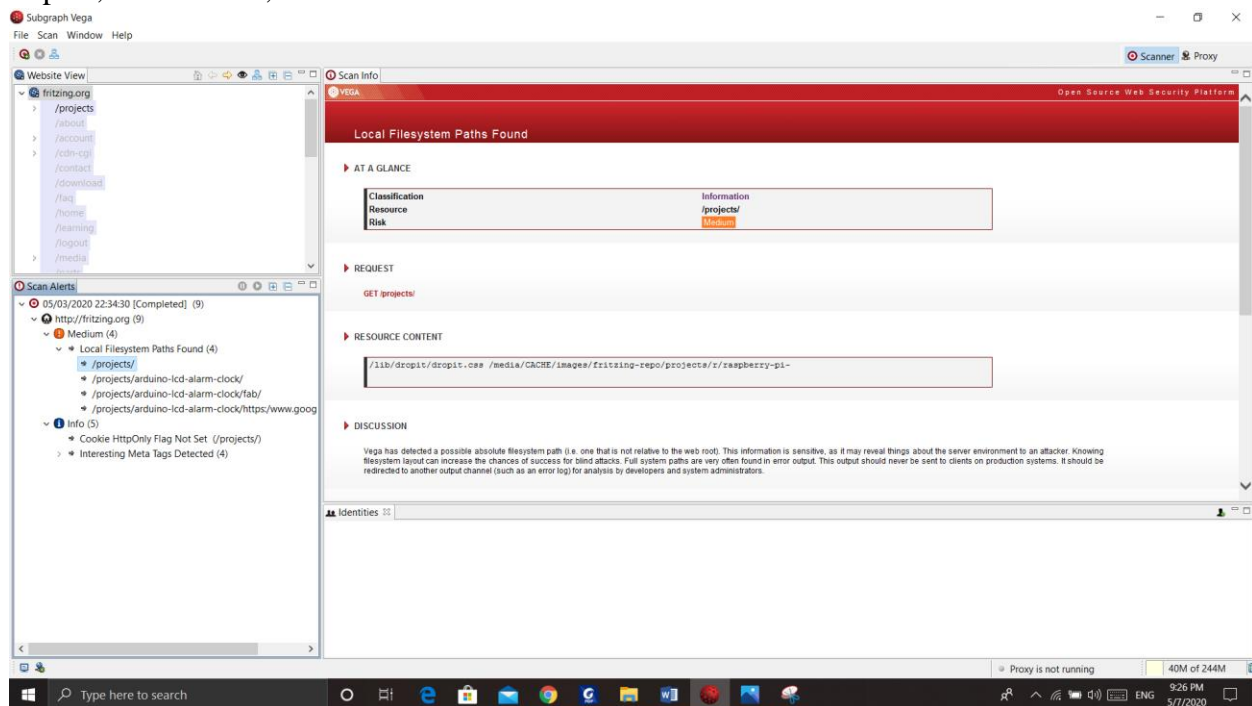


Figure 3.6.3. Scan Alerts

AT A GLANCE shown in figure 3.6.3. brings an overview of the vulnerability identified.

The REQUEST information brings the details of the request done by Vega and response of the web site.

The RESOURCE CONTENT shows the information about the resource taken by Vega where the security specialist can analyze the source code.

The DISCUSSION segment discusses signs and effects of exploitation of the weakness found by Vega.

The IMPACT section, as the name implies, describes the potential hazard found by the Vega engine and what an attacker can do when working around a reported exploitation.

The REMEDIATION section brings us useful information in regarding to what can be done inorder to minimize or completely eliminate the issue.

3.7. Summery

Vega provides security analysts with an excellent method to grasp penetration tests on Web applications. The wide range of modules entitles even inexperienced users to discover and measure their website severity in detail in possible security risks. Anyone who wants to improve their website security would like Vega and its usability.

4. REFERENCE

- [1]"What is an Audit? - Types of Audits & Auditing Certification | ASQ", *Asq.org*, 2020. [Online]. Available: <https://asq.org/quality-resources/auditing>. [Accessed: 08- May- 2020]
- [2]"What is IT audit (information technology audit)? - Definition from WhatIs.com", *SearchCompliance*, 2020. [Online]. Available: <https://searchcompliance.techtarget.com/definition/IT-audit-information-technology-audit>. [Accessed: 08- May- 2020]
- [3]W.Malsam and W. Malsam, "IT Audit: Definition & Quick Guide ProjectManager.com", *ProjectManager.com*, 2020. [Online]. Available: <https://www.projectmanager.com/blog/it-audit>. [Accessed: 08- May- 2020]
- [4]J. Strauss, "Why Your Business Needs to Have Routine IT Audits?", *Colocation America*, 2020. [Online]. Available: <https://www.colocationamerica.com/blog/importance-of-routing-it-audits>. [Accessed: 08- May- 2020]
- [5]"Web Application Audit", *Veracode*, 2020. [Online]. Available: <https://www.veracode.com/security/web-application-audit>. [Accessed: 08- May- 2020]
- [6]"Web application audit & assesment cyber security | Puffin Security", *Puffin Security*, 2020. [Online]. Available: <https://www.puffinsecurity.com/cyber-security-audit-assesment/web-application/>. [Accessed: 08- May- 2020]
- [7]"Vega Vulnerability Scanner", *Subgraph.com*, 2020. [Online]. Available: <https://subgraph.com/vega/index.en.html>. [Accessed: 08- May- 2020]
- [8]"Vega-Scanner", *Subgraph.com*, 2020. [Online]. Available: <https://subgraph.com/vega/documentation/Vega-Scanner/index.en.html>. [Accessed: 08- May- 2020]