

[illegible]

Name	ArcherySec: Vulnerability Management Framework
URL	https://attackdefense.com/challengedetails?cid=2256
Type	DevSecOps Basics: Vulnerability Management

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Challenge Description

[Archerysec](#) is an open-source assessment and management framework for performing dynamic analysis on web applications.

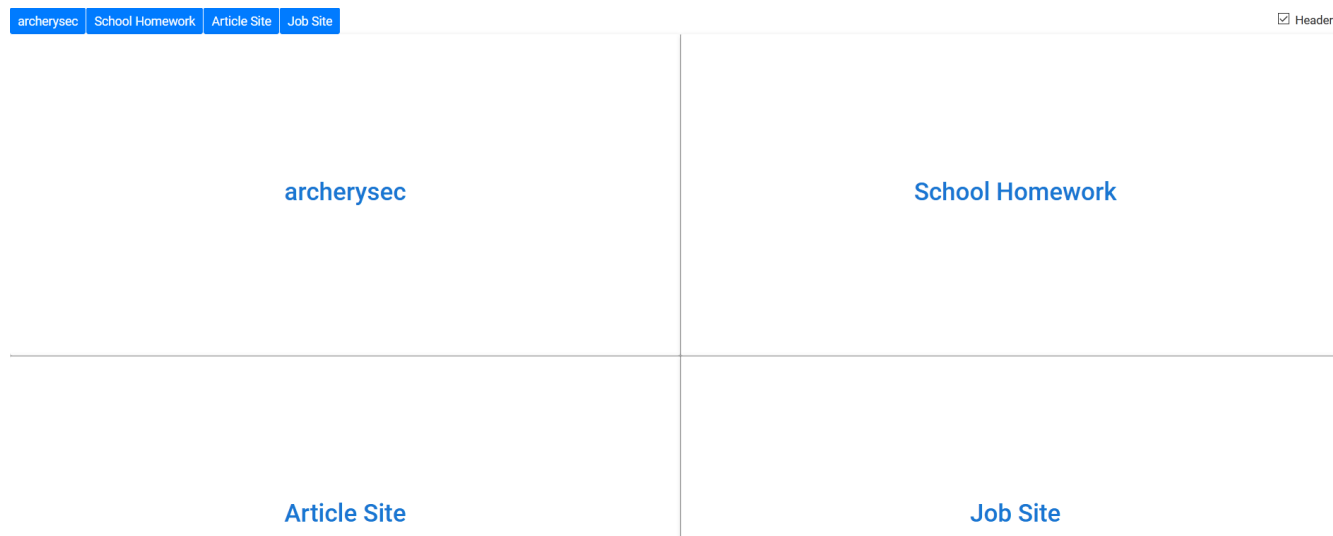
An Archerysec instance is provided to the user to perform tests on the websites. Three examples of vulnerable web portals are also provided. The details of these portals:

Web Portal	Web Portal URL
School Homework Web Portal	school-homework
Article Web Portal	article-site
Job Advertisement Web Portal	job-site

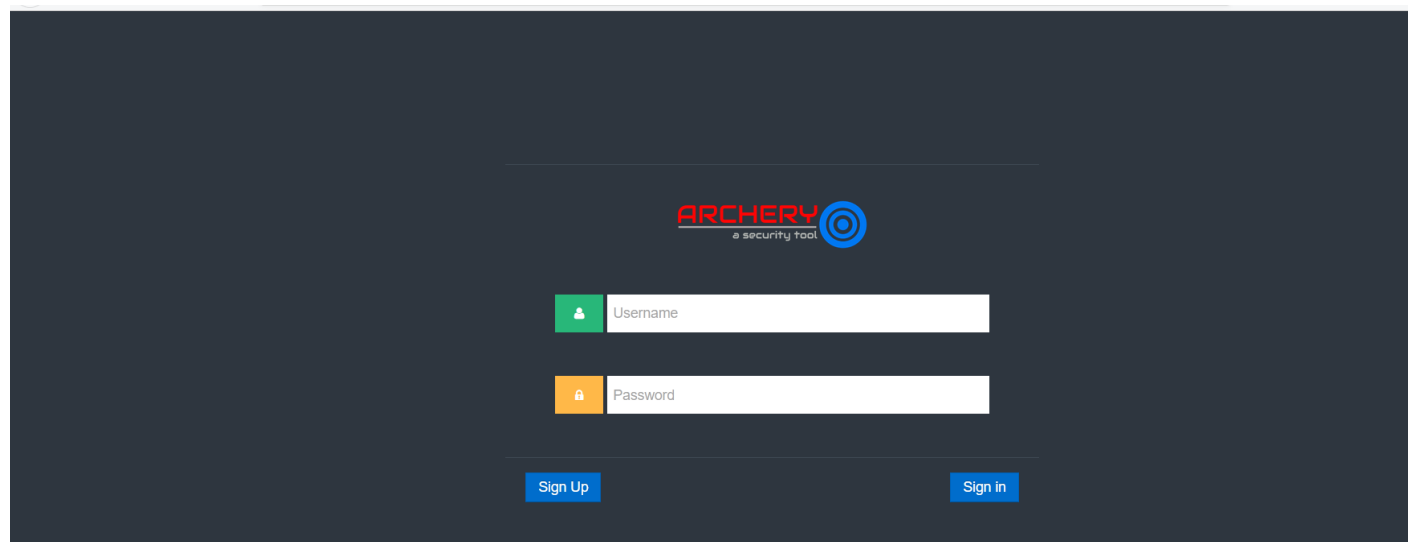
Objective: Analyze the web applications with Archerysec and identify the vulnerabilities!

Lab Setup

On starting the lab, the following interface will be accessible to the user.



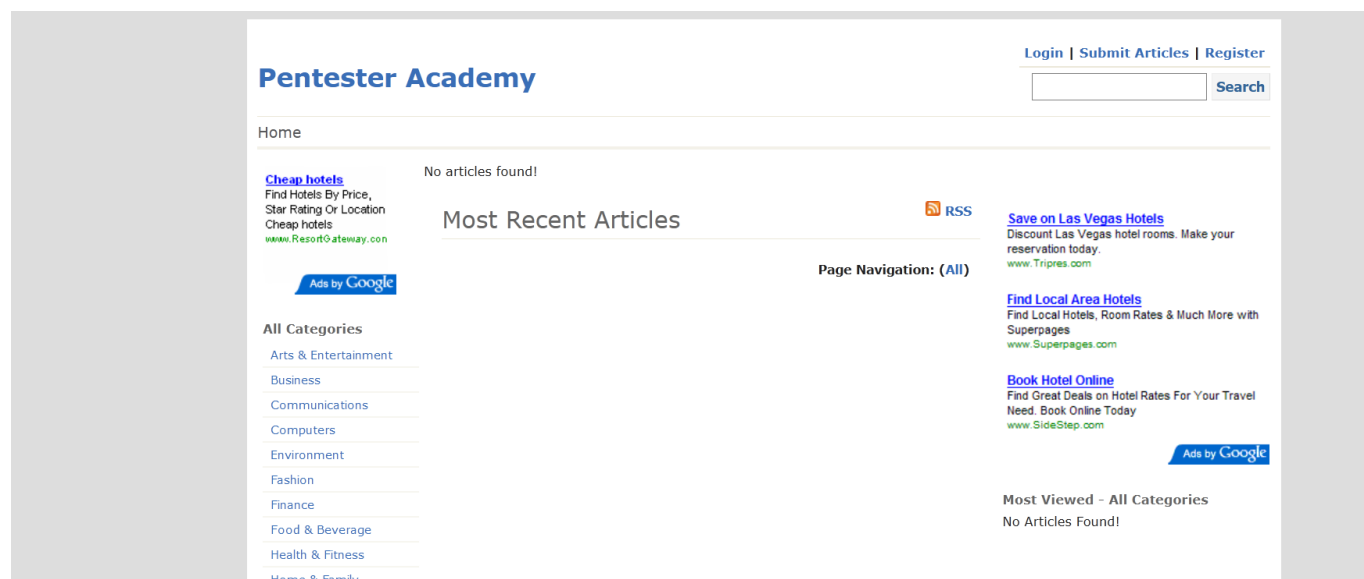
On choosing (clicking the text in the center) top left panel, **Archerysec** will open in a new tab



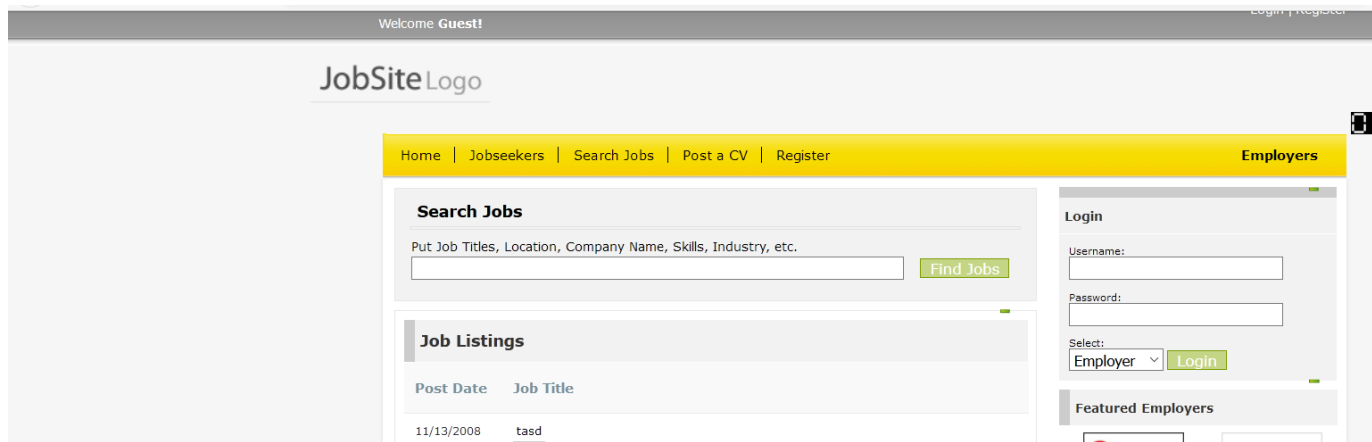
Similarly on selecting the top right panel, a web UI of **School Homework UI** will open in a new tab.



On selecting the bottom left panel, a web UI of **Article Site UI** will open in a new tab.



And on selecting the bottom right panel, a web UI of **Job Site UI** will open in a new tab.

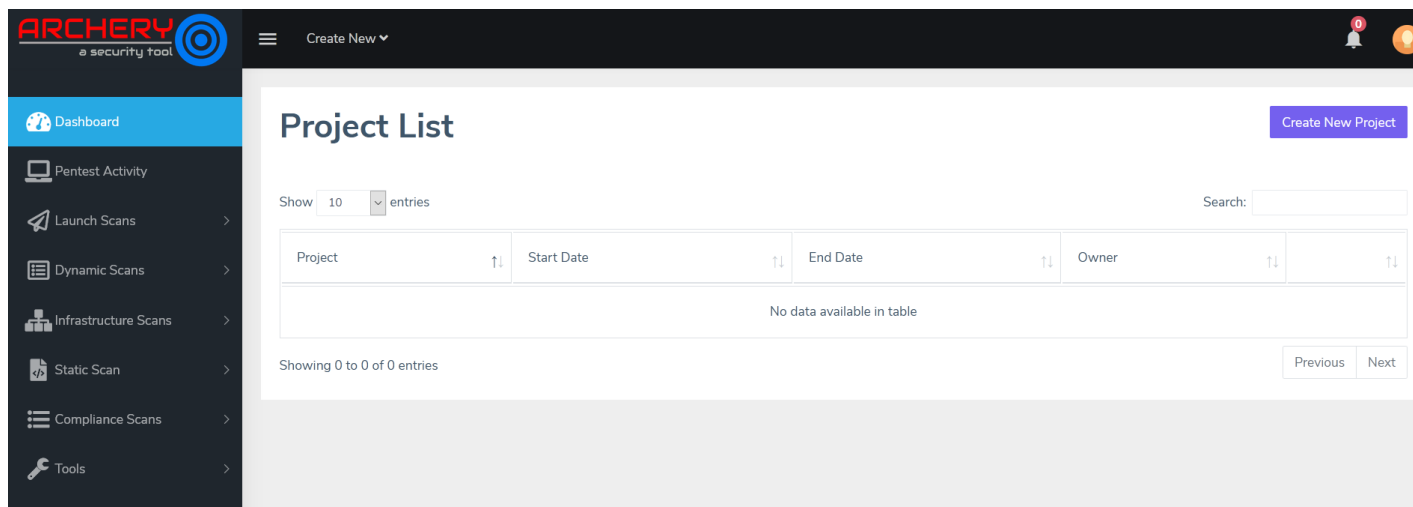


Solution

Step 1: Open the Archerysec page and log in using the credentials provided in the challenge description.

Credentials:

- **Username:** admin
- **Password:** admin



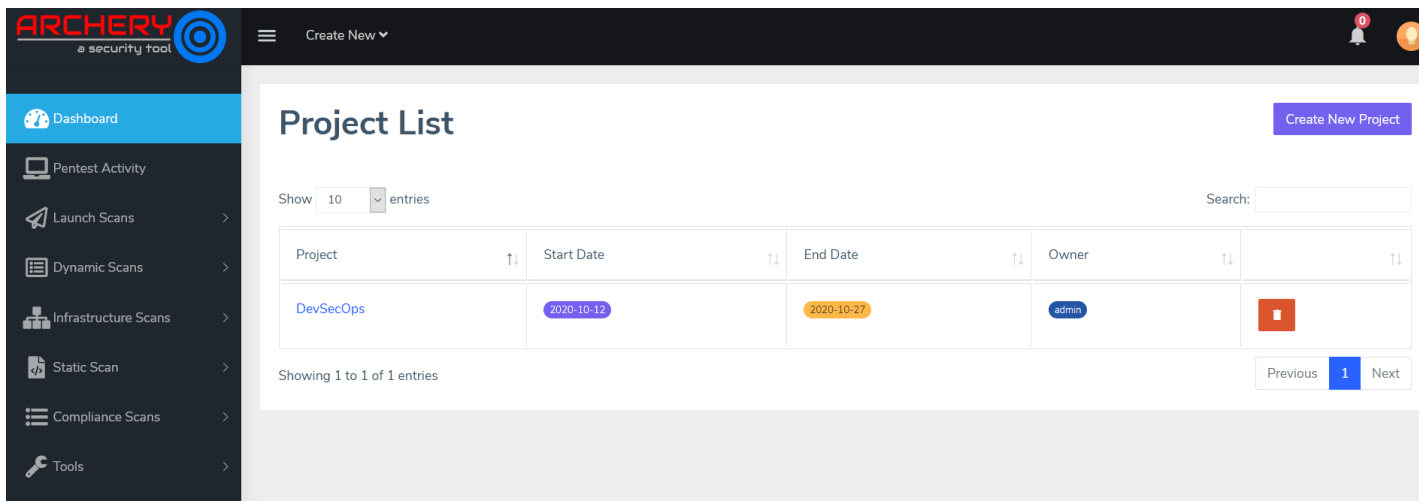
Click on the Create New Project.

The screenshot shows the Archery web application interface. On the left is a dark sidebar with a menu containing: Dashboard, Pentest Activity, Launch Scans, Dynamic Scans, Infrastructure Scans, Static Scan, Compliance Scans, and Tools. The top header is dark with the Archery logo, a 'Create New' dropdown, and a notification bell. The main content area is titled 'Create Project' and contains a form with the following fields: Project Name (text), Project Start Date (calendar icon, placeholder 'dd / mm / yyyy'), Project End date (calendar icon, placeholder 'dd / mm / yyyy'), Project Owner (text), Pentester (text), and Project Summary (text area). A green 'Save' button is at the bottom left of the form.

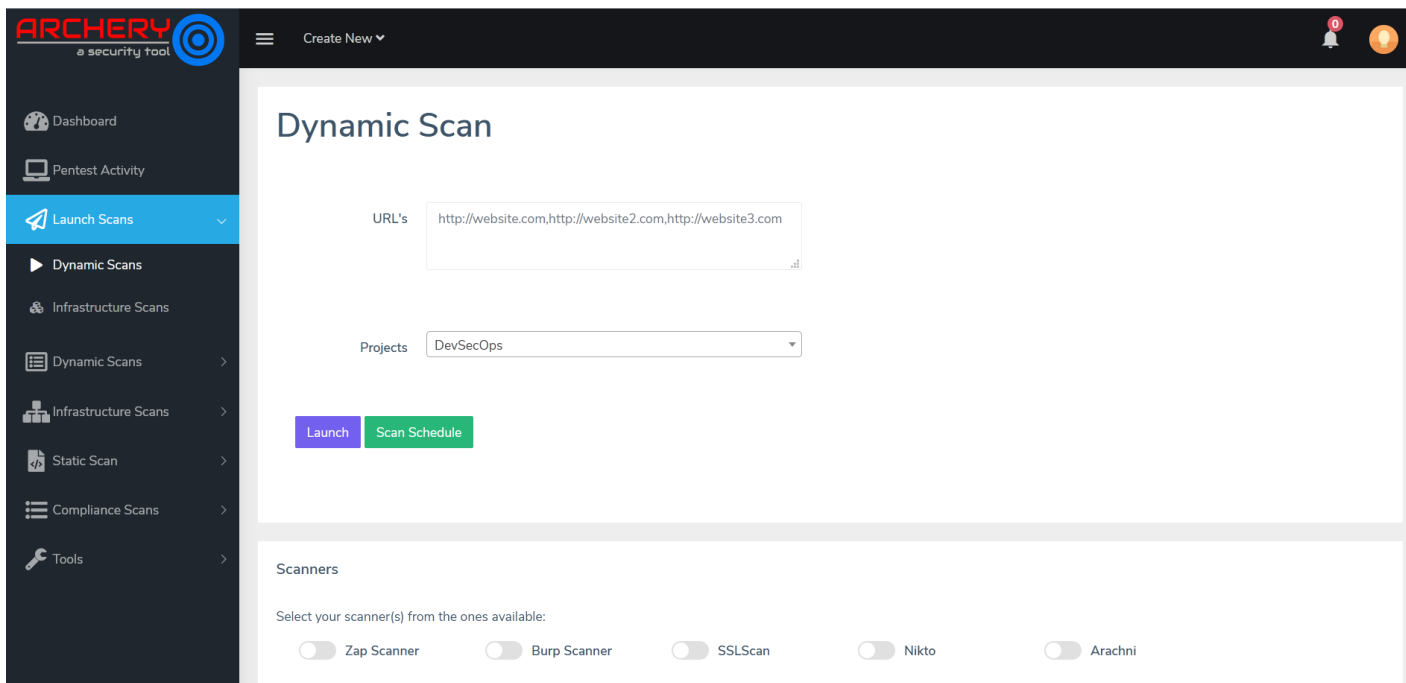
Step 2: Enter any data in the required fields.

This screenshot shows the same 'Create Project' form as the previous one, but with data entered into the fields. The 'Project Name' is 'DevSecOps'. The 'Project Start Date' is '12 / 10 / 2020' and the 'Project End date' is '27 / 10 / 2020', both with calendar icons to their right. The 'Project Owner', 'Pentester', and 'Project Summary' fields all contain the text 'admin'. The green 'Save' button remains at the bottom left.

Click on the Save button.



Step 3: Navigate to the “Dynamic Scans” sub-section located under “Launch Scans” section.

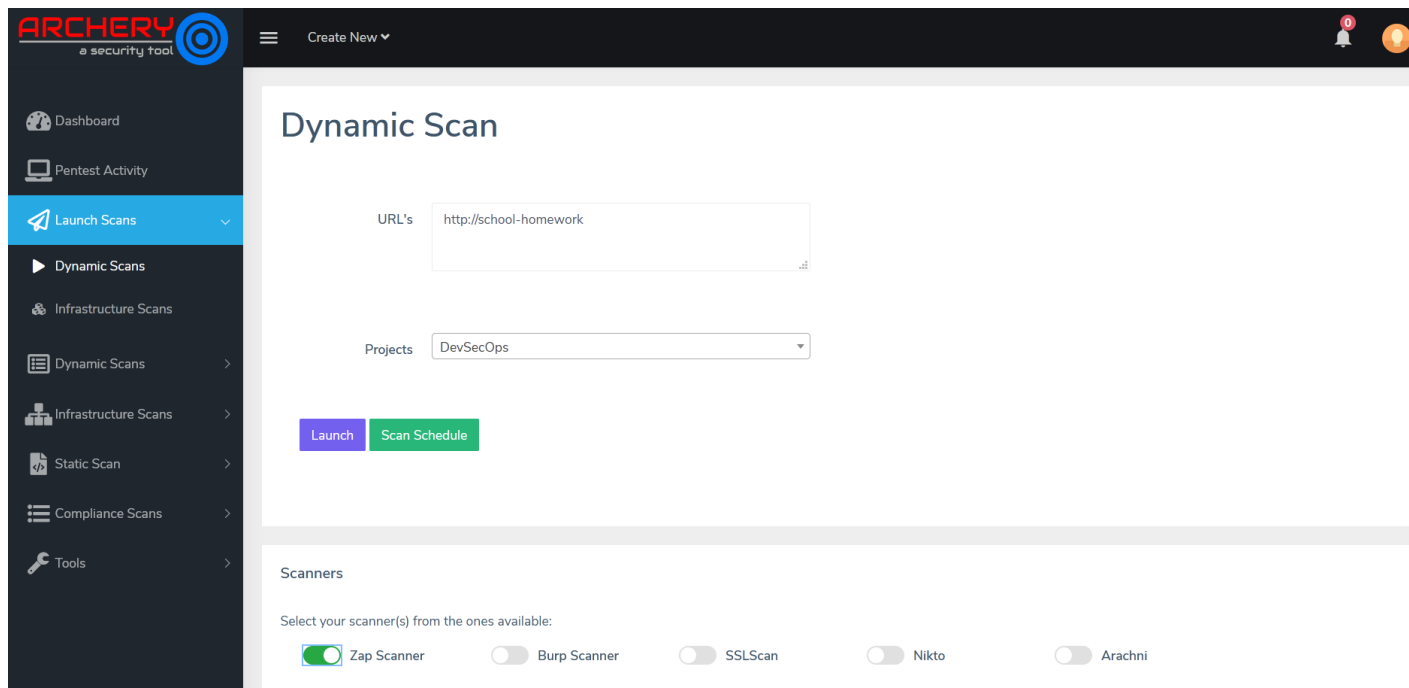


We will take one example at a time and run the tool on that.

Example 1: School Homework

Step 1: Enter the target URL in the “URL’s” field and select “Zap Scanner”

URL: http://school-homework



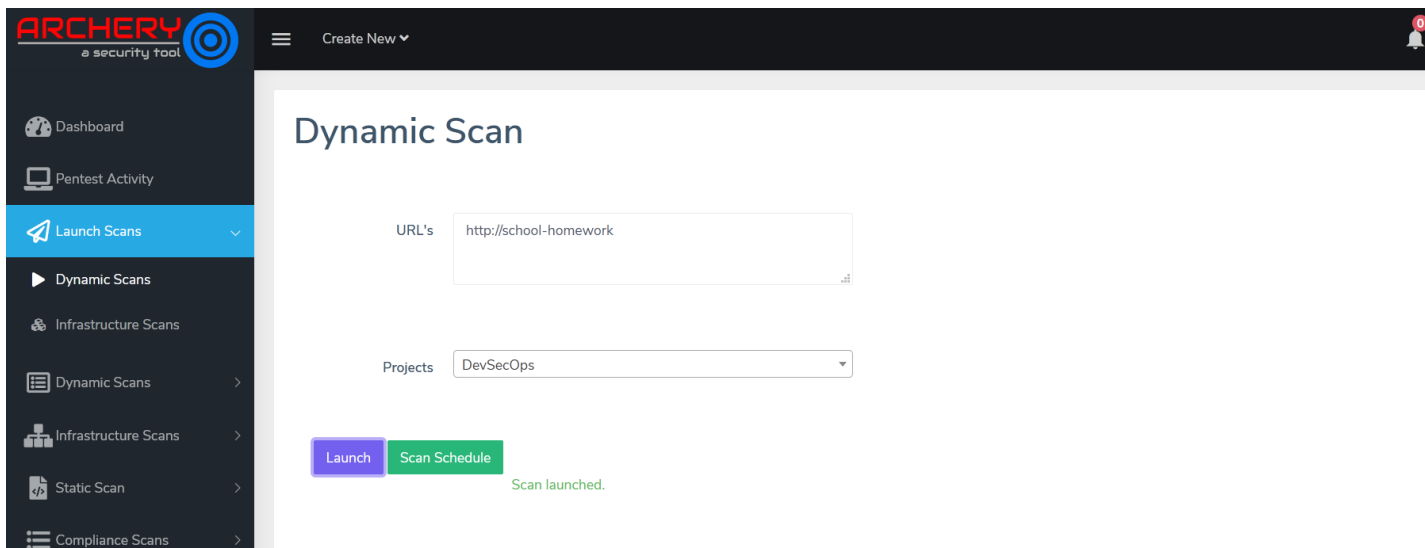
The screenshot shows the Archery security tool interface. The top navigation bar includes the Archery logo, a 'Create New' dropdown, and notification icons. The left sidebar contains a menu with options: Dashboard, Pentest Activity, Launch Scans (highlighted), Dynamic Scans, Infrastructure Scans, Static Scan, Compliance Scans, and Tools. The main content area is titled 'Dynamic Scan' and contains a form with the following fields:

- URL's: A text input field containing 'http://school-homework'.
- Projects: A dropdown menu showing 'DevSecOps'.
- Buttons: Two buttons, 'Launch' (purple) and 'Scan Schedule' (green).

Below the form, there is a 'Scanners' section with the text 'Select your scanner(s) from the ones available:'. It lists five scanners with toggle switches:

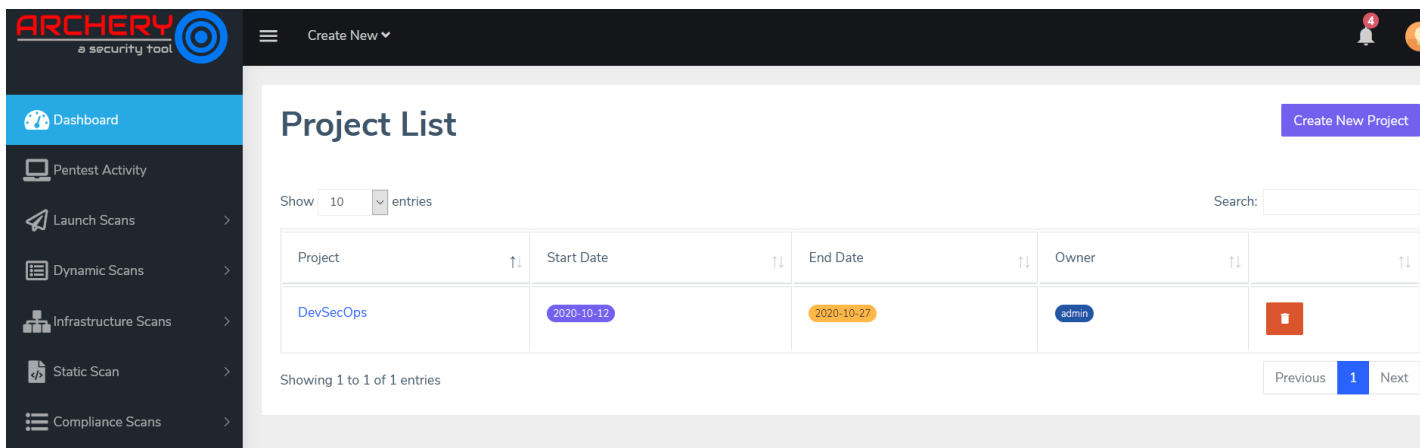
- Zap Scanner: ☒
- Burp Scanner: ☐
- SSLScan: ☐
- Nikto: ☐
- Arachni: ☐

Click on the Launch button.

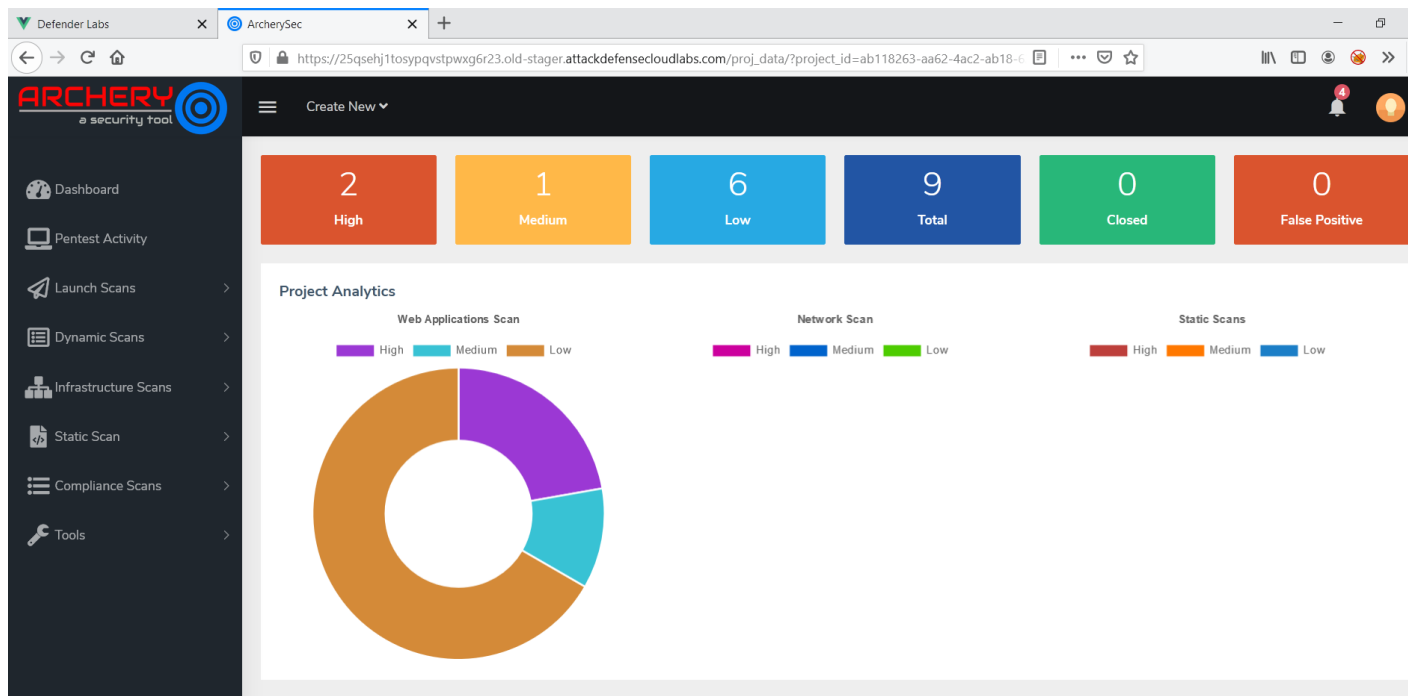


The tool will start the automated attack on the target website.

Step 2: Navigate to the Dashboard of the Archerysec.



Click on the Project name “DevSecOps”.



The summary is available for the scan performed on school-homework website.

Note: It could take some time to scan and generate the report. Refresh the page in between time intervals to check.

Step 3: Scroll down to the “Dynamic Scan List”

Dynamic Scan List

Show 10 entries

Search:

URL	Status	Date Time	Total	High	Medium	Low
http://school-homework	100% Completed	Oct. 19, 2020, 5:10 a.m.	9	2	1	6

Showing 1 to 1 of 1 entries

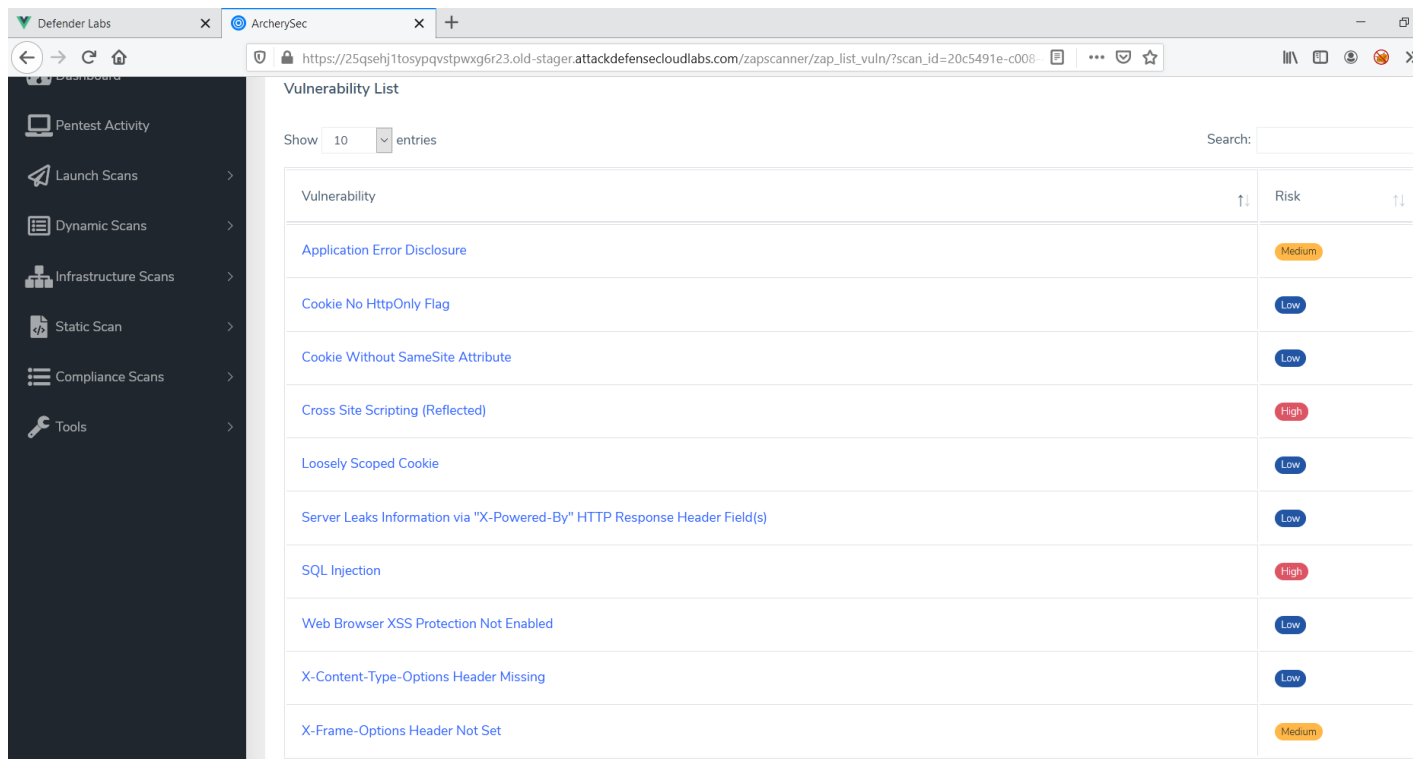
Previous 1 Next

Static Scan List

Show 10 entries

Search:

Click on the scanned website name “http://school-homework”



Vulnerability	Risk
Application Error Disclosure	Medium
Cookie No HttpOnly Flag	Low
Cookie Without SameSite Attribute	Low
Cross Site Scripting (Reflected)	High
Loosely Scoped Cookie	Low
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	Low
SQL Injection	High
Web Browser XSS Protection Not Enabled	Low
X-Content-Type-Options Header Missing	Low
X-Frame-Options Header Not Set	Medium

Note: The results from the scan can vary. Sometimes some more results are shown but sometimes not, as OWASP ZAP is performing Quick scan in the background.

Issues Detected

- Cross-Site Scripting
- SQL Injection
- Application Error Disclosure

Note: Only the ‘Medium’ and ‘High’ risk issues are written above.

Step 4: The details on these issues can be checked by clicking on the issue name.

Check the endpoints affected with XSS vulnerability.

Defender Labs x ArcherySec x +

https://25qsehj1tosypqvstpwg6r23.old-stager.attackdefensecloudlabs.com/zapscanner/zap_vuln_details/?scan_id=20c5491e-cl

ARCHERY
a security tool

Create New

Vulnerability List

Show 10 entries Search:

	Vulnerability	Risk	JIRA Ticket	False Positive	Status		
	Cross Site Scripting (Reflected)	High	None	Yes	Open		

Showing 1 to 1 of 1 entries Previous 1 Next

False Positive Vulnerability List

Show 10 entries Search:

Click on the vulnerability name “Cross Site Scripting (Reflected)”.

Defender Labs x ArcherySec x +

https://25qsehj1tosypqvstpwg6r23.old-stager.attackdefensecloudlabs.com/zapscanner/zap_vuln_check?vuln_id=dbf1fcb7-9561

ARCHERY
a security tool

Create New

Cross Site Scripting (Reflected)

Create Jira Ticket

▼ Description

Cross-site Scripting (XSS) is an attack technique that involves echoing attacker-supplied code into a user's browser instance. A browser instance can be a standard web browser client, or a browser object embedded in a software product such as the browser within WinAmp, an RSS reader, or an email client. The code itself is usually written in HTML/JavaScript, but may also extend to VBScript, ActiveX, Java, Flash, or any other browser-supported technology.

When an attacker gets a user's browser to execute his/her code, the code will run within the security context (or zone) of the hosting web site. With this level of privilege, the code has the ability to read, modify and transmit any sensitive data accessible by the browser. A Cross-site Scripted user could have his/her account hijacked (cookie theft), their browser redirected to another location, or possibly shown fraudulent content delivered by the web site they are visiting. Cross-site Scripting attacks essentially compromise the trust relationship between a user and the web site. Applications utilizing browser object instances which load content from the file system may execute code under the local machine zone allowing for system compromise.

There are three types of Cross-site Scripting attacks: non-persistent, persistent and DOM-based.

Non-persistent attacks and DOM-based attacks require a user to either visit a specially crafted link laced with malicious code, or visit a malicious web page containing a web form, which when posted to the vulnerable site, will mount the attack. Using a malicious form will oftentimes take place when the vulnerable resource only accepts HTTP POST requests. In such a case, the form can be submitted automatically, without the victim's knowledge (e.g. by using JavaScript). Upon clicking on the malicious link or submitting the malicious form, the XSS payload will get echoed back and will get interpreted by the user's browser and execute. Another technique to send almost arbitrary requests (GET and POST) is by using an embedded client, such as Adobe Flash.

Persistent attacks occur when the malicious code is submitted to a web site where it's stored for a period of time. Examples of an attacker's favorite targets often include message board posts, web mail messages, and web chat software. The unsuspecting user is not required to interact with any additional site/link (e.g. an attacker site or a malicious link sent via email), just simply view the web page containing the code.

▼ Instance

▼ Solutions

Click on the “Instance” drop-down to check the vulnerable endpoint.

The screenshot shows the ArcherySec web application security tool interface. The browser address bar displays the URL: `https://25qsehj1tosypqvstpwg6r23.old-stager.attackdefensecloudlabs.com/zapscanner/zap_vuln_check?vuln_id=dbf1fcb7-9561`. The left sidebar contains a navigation menu with the following items: Dashboard, Pentest Activity, Launch Scans, Dynamic Scans, Infrastructure Scans, Static Scan, Compliance Scans, and Tools. The main content area is titled "Cross Site Scripting (Reflected)" and includes a "Create Jira Ticket" button. The report details are as follows:

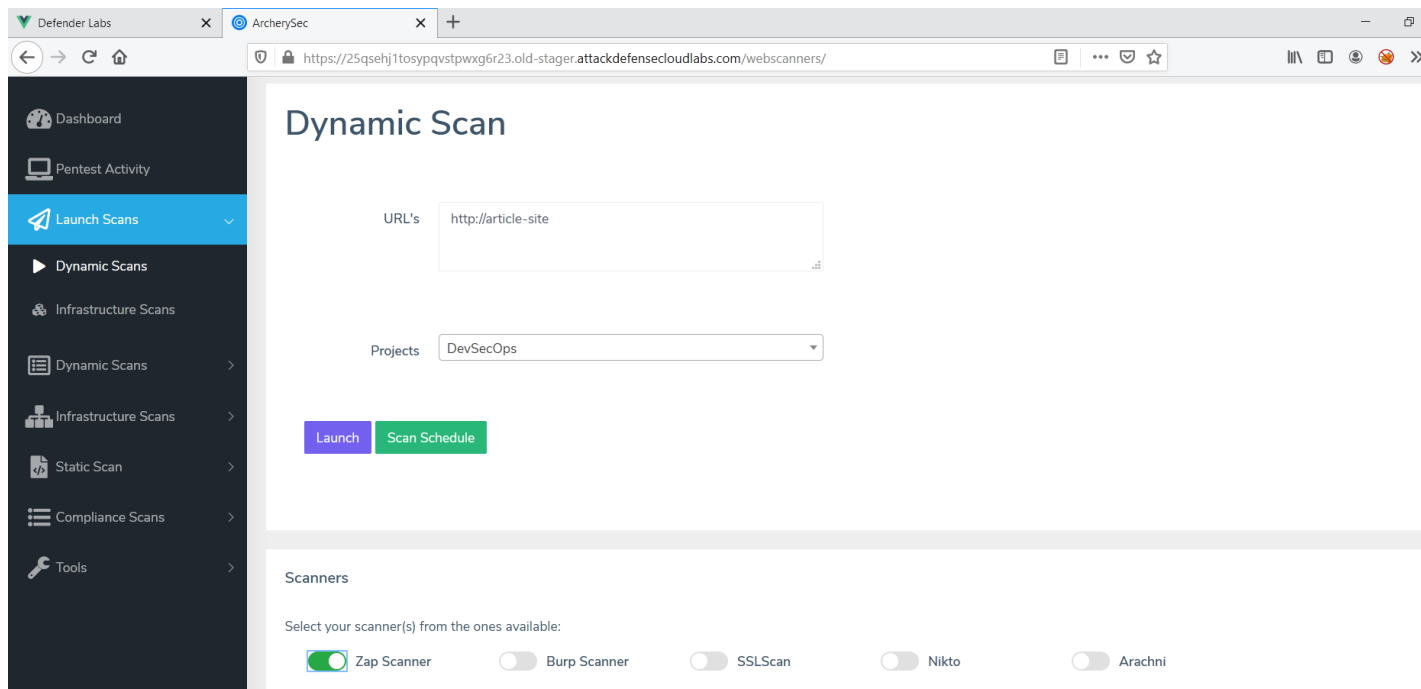
Description
Instance
URI: <code>http://school-homework/view.php?class=javascript%3Aalert%281%29%3B&print=1</code>
Method: GET
Parameter: class
Attack: <code>javascript:alert(1);</code>
Evidence: <code>javascript:alert(1);</code>
URI: <code>http://school-homework/view.php?class=javascript%3Aalert%281%29%3B</code>
Method: GET
Parameter: class

The XSS payload and the vulnerable endpoint is shown in the report.

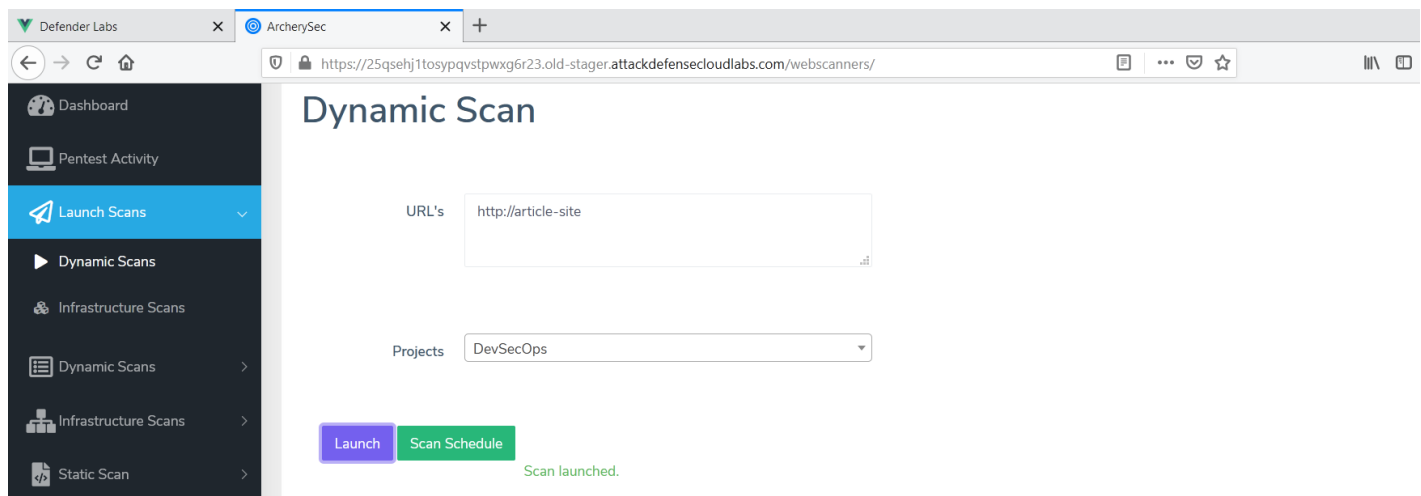
Example 2: Article Site

Step 1: Navigate to the “Dynamic Scans” sub-section under “Launch Scans” section and enter the URL in the URL’s field and select “Zap scanner”.

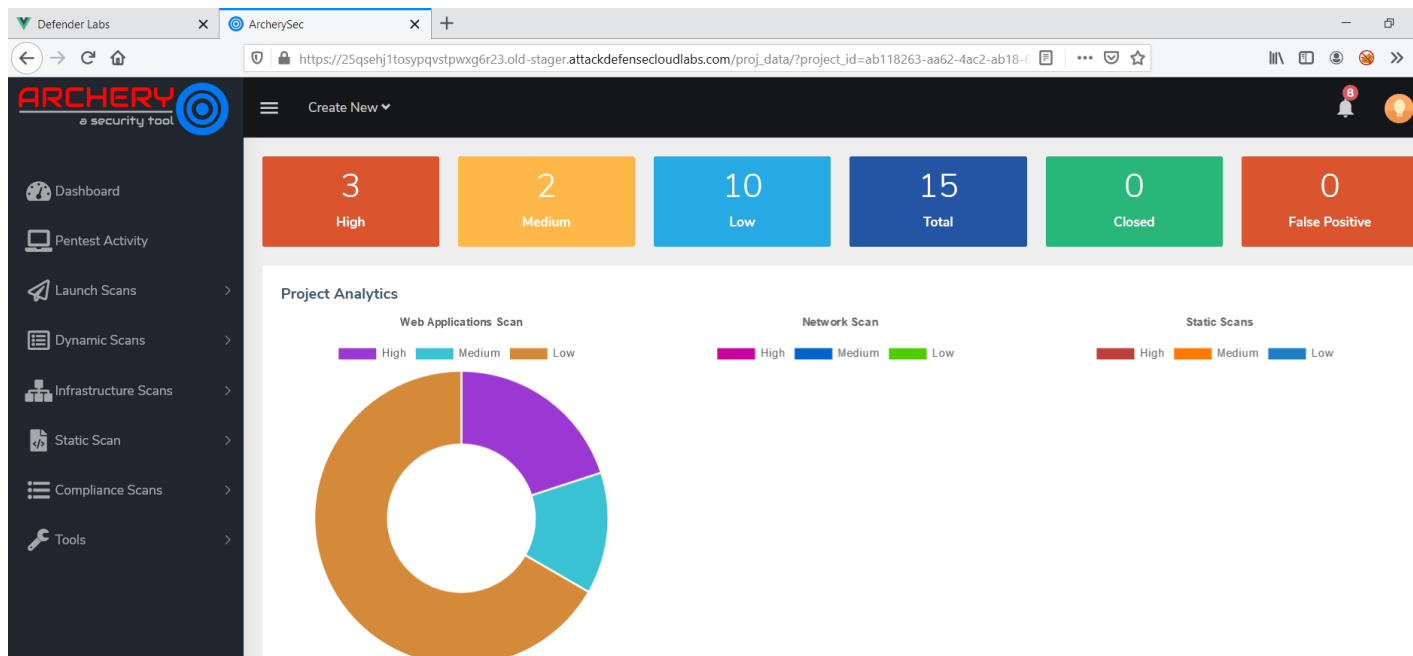
URL: `http://article-site`



Click on the Launch button to start the attack on the website.



Step 2: Navigate to the “DevSecOps” project tab located on the Dashboard.



The sheet has been updated with the new scan report. It might take some time to reflect results on the page

Step 3: Scroll down to the “Dynamic Scan List”

The screenshot shows the 'Dynamic Scan List' table. The table has columns for URL, Status, Date Time, Total, High, Medium, and Low. There are two entries for the URL 'http://school-homework', both with a status of '100% Completed'. The first entry is dated Oct. 19, 2020, 5:10 a.m. and the second is dated Oct. 19, 2020, 5:23 a.m. The table includes a search bar and pagination controls.

URL	Status	Date Time	Total	High	Medium	Low
http://school-homework	100% Completed	Oct. 19, 2020, 5:10 a.m.	9	2	1	6
http://school-homework	100% Completed	Oct. 19, 2020, 5:23 a.m.	6	1	1	4

Note: The list shows the same URL name. This is a glitch and to check the results of the article-site, click on the URL name with the latest timestamp.

Vulnerability List

Show 10 entries

Search:

Vulnerability	Risk
Absence of Anti-CSRF Tokens	Low
Cross-Domain JavaScript Source File Inclusion	Low
Directory Browsing	Medium
Information Disclosure - Suspicious Comments	Low
SQL Injection - MySQL	High
Timestamp Disclosure - Unix	Low

Showing 1 to 6 of 6 entries

Previous 1 Next

Issues Detected

- Directory Browsing
- SQL Injection

Step 4: The details on these issues can be checked by clicking on the issue name.

Check the endpoints affected with SQL Injection vulnerability.

Vulnerability List

Show 10 entries

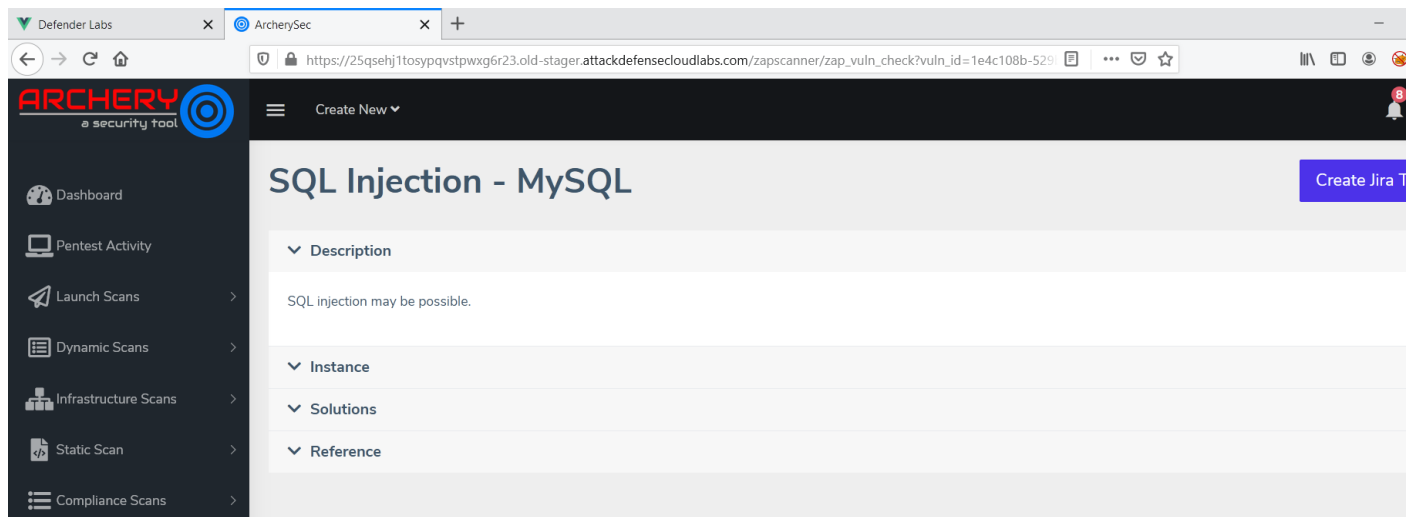
Search:

Vulnerability	Risk	JIRA Ticket	False Positive	Status
SQL Injection - MySQL	High	None	Yes	Open

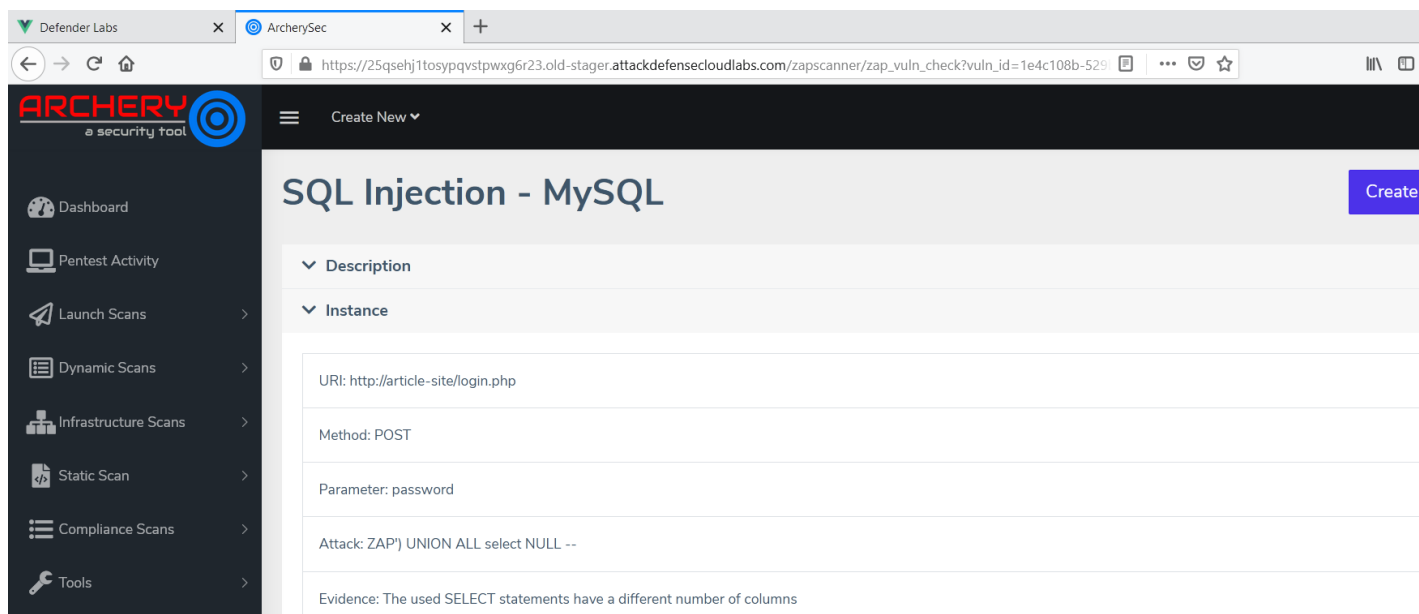
Showing 1 to 1 of 1 entries

Previous

Click on the vulnerability name “SQL Injection - MySQL”



Click on the “Instance” drop-down to check the vulnerable endpoint.

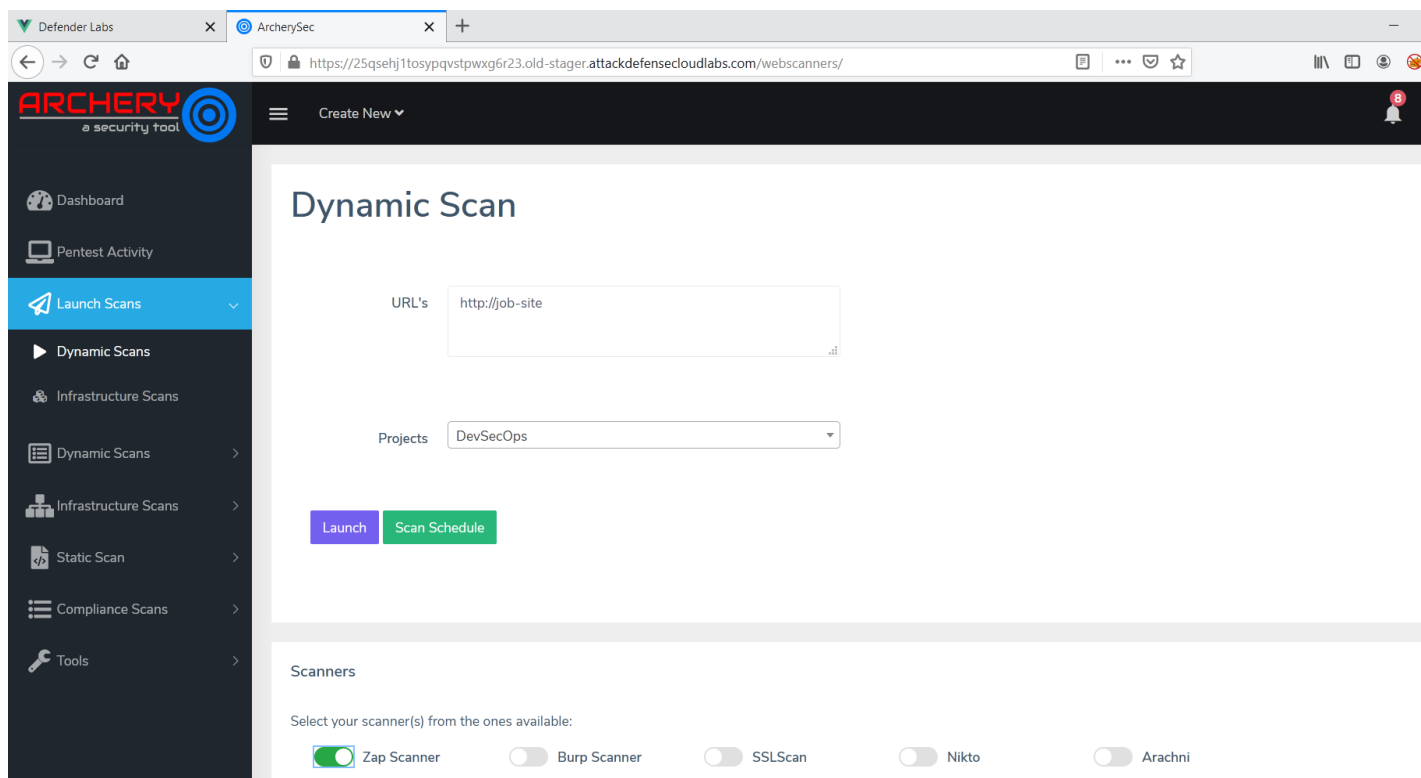


The POST request contains the payload in the password field which is vulnerable to SQL injection attack.

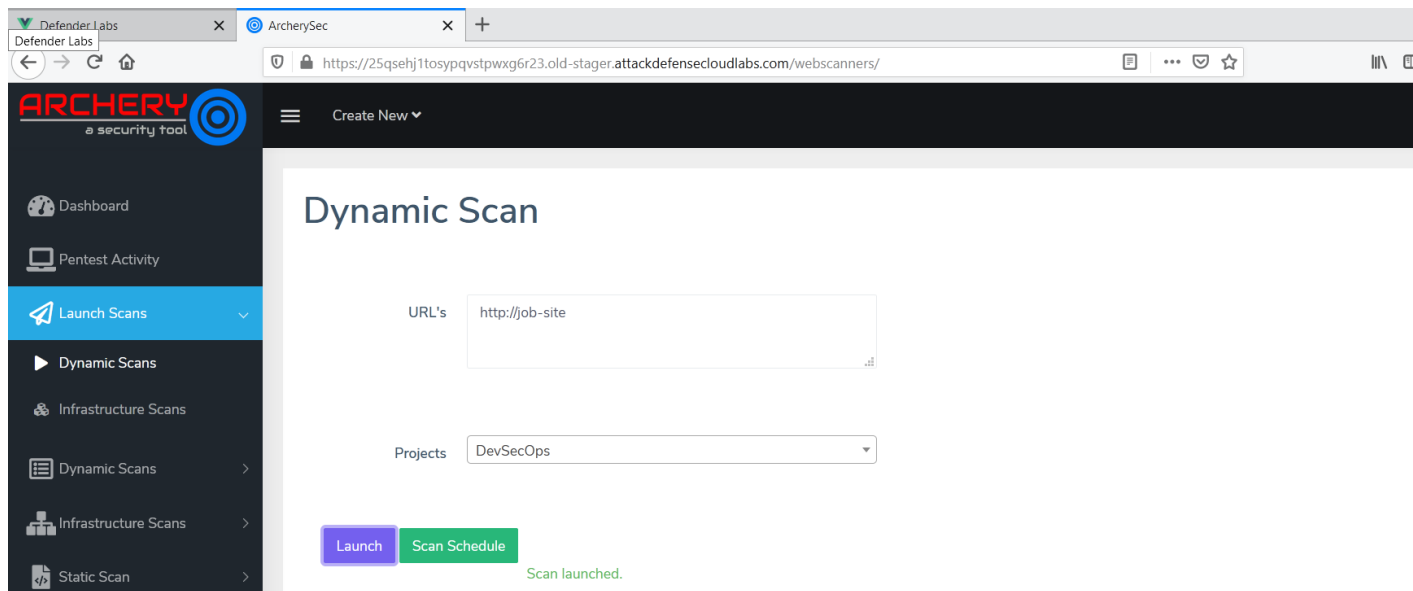
Example 3: Job Site

Step 1: Navigate to the “Dynamic Scans” under “Launch Scans” section and enter the URL in the URL’s field and select “Zap scanner”.

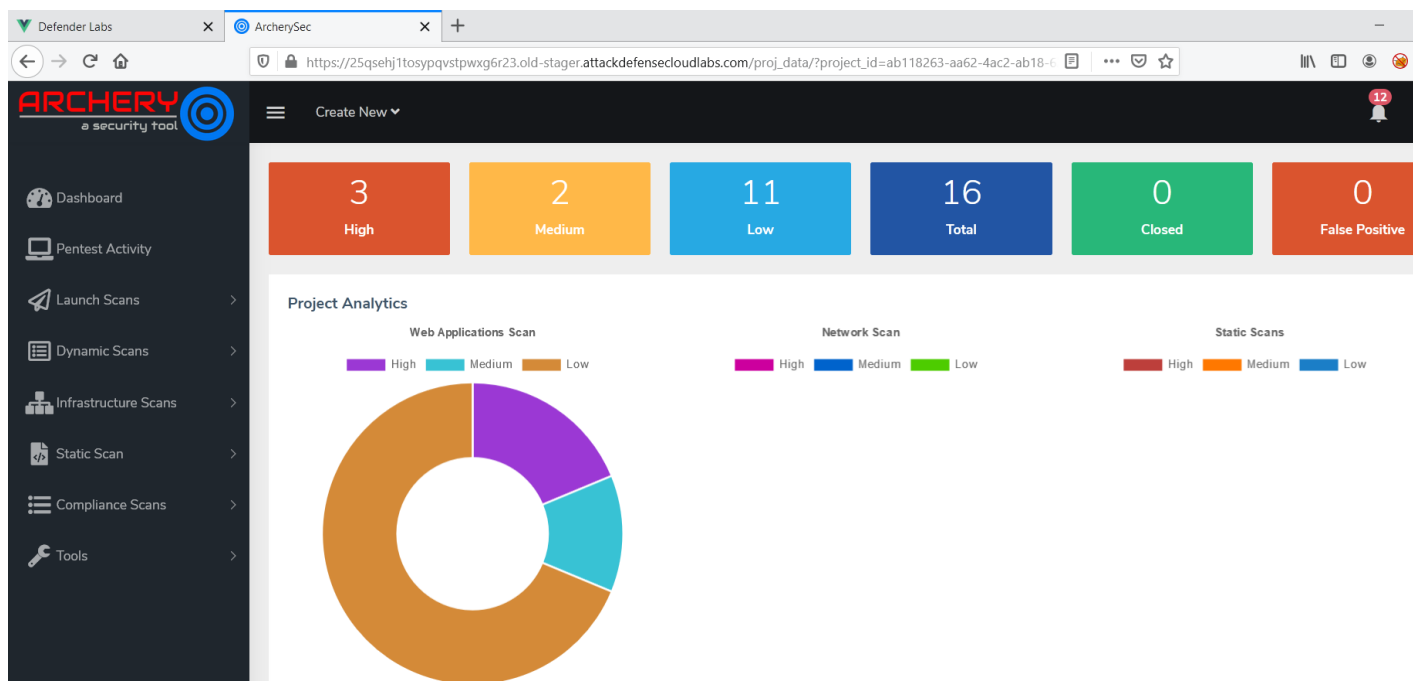
URL: http://job-site



Click on the Launch button.



Step 2: Navigate to the “DevSecOps” project tab located on the Dashboard.



Step 3: Scroll down to the “Dynamic Scan List”

The screenshot shows the ArcherySec web application interface. The browser tab is labeled 'ArcherySec' and the URL is https://mpl7ommiga5yvx5j8d33djfwh.old-stager.attackdefensecloudlabs.com/proj_data/?project_id=1657e448-ba1a-4317-9738. The main content area is titled 'Dynamic Scan List'. It features a 'Show 10 entries' dropdown and a search bar. Below this is a table with the following columns: URL, Status, Date Time, Total, High, and Medium. The table contains three entries, all for the URL 'http://school-homework' and with a status of '100% Completed'. The first entry has a date time of 'Oct. 19, 2020, 6:26 a.m.' and counts of 9 Total, 2 High, and 2 Medium. The second entry has a date time of 'Oct. 19, 2020, 6:27 a.m.' and counts of 7 Total, 1 High, and 1 Medium. The third entry has a date time of 'Oct. 19, 2020, 6:29 a.m.' and counts of 0 Total, 0 High, and 0 Medium. At the bottom of the table, it says 'Showing 1 to 3 of 3 entries' and there is a 'Previous' button.

URL	Status	Date Time	Total	High	Medium
http://school-homework	100% Completed	Oct. 19, 2020, 6:26 a.m.	9	2	2
http://school-homework	100% Completed	Oct. 19, 2020, 6:27 a.m.	7	1	1
http://school-homework	100% Completed	Oct. 19, 2020, 6:29 a.m.	0	0	0

Click on the URL name with the latest timestamp

The screenshot shows the ArcherySec web application interface. The browser tab is labeled 'ArcherySec' and the URL is https://mpl7ommiga5yvx5j8d33djfwh.old-stager.attackdefensecloudlabs.com/zapscanner/zap_list_vuln/?scan_id=94734885-5346-4f44. The main content area is titled 'Vulnerability List'. It features a 'Show 10 entries' dropdown and a search bar. Below this is a table with the following columns: Vulnerability and Risk. The table is empty, and a message 'No data available in table' is displayed. At the bottom of the table, it says 'Showing 0 to 0 of 0 entries' and there is a 'Previous' button.

Vulnerability	Risk
---------------	------

There were no issues found. The results might vary from instance to instance.

Learnings

Perform dynamic code analysis on web applications using ArcherySec.