Chapter3:

Introduction:

Zed Attack Proxy (ZAP) is a free, open-source penetration testing tool being maintained under the umbrella of the Open Web Application Security Project (OWASP). ZAP is designed specifically for testing web applications and is both flexible and extensible.

At its core, ZAP is what is known as a “man-in-the-middle proxy.” It stands between the tester’s browser and the web application so that it can intercept and inspect messages sent between browser and web application, modify the contents if needed, and then forward those packets on to the destination. It can be used as a stand-alone application, and as a daemon process.



If there is another network proxy already in use, as in many corporate environments, ZAP can be configured to connect to that proxy.



ZAP provides functionality for a range of skill levels – from developers, to testers new to security testing, to security testing specialists. ZAP has versions for each major OS and Docker, so you are not tied to a single OS. Additional functionality is freely available from a variety of add-ons in the ZAP Marketplace, accessible from within the ZAP client.

Because ZAP is open-source, the source code can be examined to see exactly how the functionality is implemented. Anyone can volunteer to work on ZAP, fix bugs, add features, create pull requests to pull fixes into the project, and author add-ons to support specialized situations.

As with most open source projects, donations are welcome to help with costs for the projects. You can find a donate button on the owasp.org page for ZAP at <https://owasp.org/www-project-zap/>.

[http://172.30.2.67:8000](http://172.30.2.67:8000/)

[http://172.30.2.67:8001](http://172.30.2.67:8001/)

<http://172.30.2.67:8003>

2: Zap Installation:

The first thing to do is install ZAP on the system you intend to perform pentesting on. Download the appropriate installer from the [Download](https://www.zaproxy.org/download/) page.

Note that ZAP requires Java 11+ in order to run. The macOS installer includes an appropriate version of Java but you must install Java 11+ separately for Windows, Linux, and Cross-Platform versions. The Docker versions do not require you to install Java.

Once the installation is complete, launch ZAP and read the license terms. Click **Agree** if you accept the terms, and ZAP will finish installing, then ZAP will automatically start.

###### macOS

OWASP ZAP is currently not a verified developer with Apple. On macOS, you will see a message like:

“OWASP ZAP.app” cannot be opened because the developer cannot be verified.

To circumvent this warning, you would need to go to **System Preferences** > **Security & Privacy** at the bottom of the dialog. You will see a message saying that “OWASP ZAP” was blocked. Next to it, if you trust the downloaded installer, you can click **Open anyway**.

[http://172.30.2.67:8000](http://172.30.2.67:8000/)

[http://172.30.2.67:8001](http://172.30.2.67:8001/)

<http://172.30.2.67:8003>

3: Deep Dive into the ZAP:

The ZAP Desktop UI is composed of the following elements:

1. **Menu Bar** – Provides access to many of the automated and manual tools.
2. **Toolbar** – Includes buttons which provide easy access to most commonly used features.
3. **Tree Window** – Displays the Sites tree and the Scripts tree.
4. **Workspace Window** – Displays requests, responses, and scripts and allows you to edit them.
5. **Information Window** – Displays details of the automated and manual tools.
6. **Footer** – Displays a summary of the alerts found and the status of the main automated tools.

https://www.zaproxy.org/getting-started/images/zap-full-screen.png

While using ZAP, you can click **Help** on the Menu Bar or press F1 to access context-sensitive help from the ZAP Desktop User Guide. It is also available [online](https://www.zaproxy.org/docs/desktop/).

For more information about the UI, see [ZAP UI Overview](https://www.zaproxy.org/docs/desktop/ui/) in the ZAP online documentation.

ZAP also supports a powerful API and command line functionality, both of which are beyond the scope of this guide.

**IMPORTANT**: You should only use ZAP to attack an application you have permission to test with an active attack. Because this is a simulation that acts like a real attack, actual damage can be done to a site’s functionality, data, etc. If you are worried about using ZAP, you can prevent it from causing harm (though ZAP’s functionality will be significantly reduced) by switching to safe mode.

To switch ZAP to safe mode, click the arrow on the mode dropdown on the main toolbar to expand the dropdown list and select **Safe Mode**.

##### Running an Automated Scan

The easiest way to start using ZAP is via the Quick Start tab. Quick Start is a ZAP add-on that is included automatically when you installed ZAP.

To run a Quick Start Automated Scan :

1. Start ZAP and click the **Quick Start** tab of the Workspace Window.
2. Click the large Automated Scan button.
3. In the **URL to attack** text box, enter the full URL of the web application you want to attack.
4. Click the **Attack**

https://www.zaproxy.org/getting-started/images/zap-qstart-autoscan.png

ZAP will proceed to crawl the web application with its spider and passively scan each page it finds. Then ZAP will use the active scanner to attack all of the discovered pages, functionality, and parameters.

ZAP provides 2 spiders for crawling web applications, you can use either or both of them from this screen.

The traditional ZAP spider which discovers links by examining the HTML in responses from the web application. This spider is fast, but it is not always effective when exploring an AJAX web application that generates links using JavaScript.

For AJAX applications, ZAP’s AJAX spider is likely to be more effective. This spider explores the web application by invoking browsers which then follow the links that have been generated. The AJAX spider is slower than the traditional spider and requires additional configuration for use in a “headless” environment.

ZAP will passively scan all of the requests and responses proxied through it. So far ZAP has only carried out passive scans of your web application. Passive scanning does not change responses in any way and is considered safe. Scanning is also performed in a background thread to not slow down exploration. Passive scanning is good at finding some vulnerabilities and as a way to get a feel for the basic security state of a web application and locate where more investigation may be warranted.

Active scanning, however, attempts to find other vulnerabilities by using known attacks against the selected targets. Active scanning is a real attack on those targets and can put the targets at risk, so do not use active scanning against targets you do not have permission to test.

##### Interpret Your Test Results

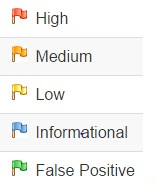
As ZAP spiders your web application, it constructs a map of your web applications’ pages and the resources used to render those pages. Then it records the requests and responses sent to each page and creates alerts if there is something potentially wrong with a request or response.

##### See Explored Pages

To examine a tree view of the explored pages, click the **Sites** tab in the Tree Window. You can expand the nodes to see the individual URLs accessed.

##### View Alerts and Alert Details

The left-hand side of the Footer contains a count of the Alerts found during your test, broken out into risk categories. These risk categories are:



To view the alerts created during your test:

1. Click the **Alerts** tab in the Information Window.
2. Click each alert displayed in that window to display the URL and the vulnerability detected in the right side of the Information Window.
3. In the Workspace Windows, click the **Response** tab to see the contents of the header and body of the response. The part of the response that generated the alert will be highlighted.

### Exploring an Application Manually

The passive scanning and automated attack functionality is a great way to begin a vulnerability assessment of your web application but it has some limitations. Among these are:

* Any pages protected by a login page are not discoverable during a passive scan because, unless you’ve configured ZAP’s authentication functionality, ZAP will not handle the required authentication.
* You don’t have a lot of control over the sequence of exploration in a passive scan or the types of attacks carried out in an automated attack. ZAP does provide many additional options for exploration and attacks outside of passive scanning.

Spiders are a great way to explore your basic site, but they should be combined with manual exploration to be more effective. Spiders, for example, will only enter basic default data into forms in your web application but a user can enter more relevant information which can, in turn, expose more of the web application to ZAP. This is especially true with things like registration forms where a valid email address is required. The spider may enter a random string, which will cause an error. A user will be able to react to that error and supply a correctly formatted string, which may cause more of the application to be exposed when the form is submitted and accepted.

You should explore all of your web application with a browser proxying through ZAP. As you do this, ZAP passively scans all the requests and responses made during your exploration for vulnerabilities, continues to build the site tree, and records alerts for potential vulnerabilities found during the exploration.

It is important to have ZAP explore each page of your web application, whether linked to another page or not, for vulnerabilities. Obscurity is not security, and hidden pages sometimes go live without warning or notice. So be as thorough as you can when exploring your site.

You can quickly and easily launch browsers that are pre-configured to proxy through ZAP via the Quick Start tab. Browsers launched in this way will also ignore any certificate validation warnings that would otherwise be reported.

https://www.zaproxy.org/getting-started/images/zap-qstart-manualexplore.png

To Manually Explore your application:

1. Start ZAP and click the **Quick Start** tab of the Workspace Window.
2. Click the large Manual Explore button.
3. In the **URL to explore** text box, enter the full URL of the web application you want to explore.
4. Select the browser you would like to use
5. Click the **Launch Browser**

This option will launch any of the most common browsers that you have installed with new profiles.

If you would like to use any of your browsers with an existing profile, for example with other browser add-ons installed, then you will need to manually configure your browser to proxy via ZAP and import and trust the ZAP Root CA Certificate. See the ZAP Desktop User Guide for more details.

By default the ZAP Heads Up Display (HUD) will be enabled. Unchecking the relevant option on this screen before launching a browser will disable the HUD.

[http://172.30.2.67:8000](http://172.30.2.67:8000/)

[http://172.30.2.67:8001](http://172.30.2.67:8001/)

<http://172.30.2.67:8003>

4: DeeP Dive Into ZAP Part2:

The ZAP Desktop UI is composed of the following elements:

Menu Bar – Provides access to many of the automated and manual tools.

Toolbar – Includes buttons which provide easy access to most commonly used features.

Tree Window – Displays the Sites tree and the Scripts tree.

Workspace Window – Displays requests, responses, and scripts and allows you to edit them.

Information Window – Displays details of the automated and manual tools.

Footer – Displays a summary of the alerts found and the status of the main automated tools.

While using ZAP, you can click Help on the Menu Bar or press F1 to access context-sensitive help from the ZAP Desktop User Guide. It is also available online.

For more information about the UI, see ZAP UI Overview in the ZAP online documentation.

ZAP also supports a powerful API and command line functionality, both of which are beyond the scope of this guide.

IMPORTANT: You should only use ZAP to attack an application you have permission to test with an active attack. Because this is a simulation that acts like a real attack, actual damage can be done to a site’s functionality, data, etc. If you are worried about using ZAP, you can prevent it from causing harm (though ZAP’s functionality will be significantly reduced) by switching to safe mode.

To switch ZAP to safe mode, click the arrow on the mode dropdown on the main toolbar to expand the dropdown list and select Safe Mode.

Running an Automated Scan

The easiest way to start using ZAP is via the Quick Start tab. Quick Start is a ZAP add-on that is included automatically when you installed ZAP.

To run a Quick Start Automated Scan :

Start ZAP and click the Quick Start tab of the Workspace Window.

Click the large Automated Scan button.

In the URL to attack text box, enter the full URL of the web application you want to attack.

Click the Attack

ZAP will proceed to crawl the web application with its spider and passively scan each page it finds. Then ZAP will use the active scanner to attack all of the discovered pages, functionality, and parameters.

ZAP provides 2 spiders for crawling web applications, you can use either or both of them from this screen.

The traditional ZAP spider which discovers links by examining the HTML in responses from the web application. This spider is fast, but it is not always effective when exploring an AJAX web application that generates links using JavaScript.

For AJAX applications, ZAP’s AJAX spider is likely to be more effective. This spider explores the web application by invoking browsers which then follow the links that have been generated. The AJAX spider is slower than the traditional spider and requires additional configuration for use in a “headless” environment.

ZAP will passively scan all of the requests and responses proxied through it. So far ZAP has only carried out passive scans of your web application. Passive scanning does not change responses in any way and is considered safe. Scanning is also performed in a background thread to not slow down exploration. Passive scanning is good at finding some vulnerabilities and as a way to get a feel for the basic security state of a web application and locate where more investigation may be warranted.

Active scanning, however, attempts to find other vulnerabilities by using known attacks against the selected targets. Active scanning is a real attack on those targets and can put the targets at risk, so do not use active scanning against targets you do not have permission to test.

Interpret Your Test Results

As ZAP spiders your web application, it constructs a map of your web applications’ pages and the resources used to render those pages. Then it records the requests and responses sent to each page and creates alerts if there is something potentially wrong with a request or response.

See Explored Pages

To examine a tree view of the explored pages, click the Sites tab in the Tree Window. You can expand the nodes to see the individual URLs accessed.

View Alerts and Alert Details

The left-hand side of the Footer contains a count of the Alerts found during your test, broken out into risk categories. These risk categories are:

To view the alerts created during your test:

Click the Alerts tab in the Information Window.

Click each alert displayed in that window to display the URL and the vulnerability detected in the right side of the Information Window.

In the Workspace Windows, click the Response tab to see the contents of the header and body of the response. The part of the response that generated the alert will be highlighted.

Exploring an Application Manually

The passive scanning and automated attack functionality is a great way to begin a vulnerability assessment of your web application but it has some limitations. Among these are:

Any pages protected by a login page are not discoverable during a passive scan because, unless you’ve configured ZAP’s authentication functionality, ZAP will not handle the required authentication.

You don’t have a lot of control over the sequence of exploration in a passive scan or the types of attacks carried out in an automated attack. ZAP does provide many additional options for exploration and attacks outside of passive scanning.

Spiders are a great way to explore your basic site, but they should be combined with manual exploration to be more effective. Spiders, for example, will only enter basic default data into forms in your web application but a user can enter more relevant information which can, in turn, expose more of the web application to ZAP. This is especially true with things like registration forms where a valid email address is required. The spider may enter a random string, which will cause an error. A user will be able to react to that error and supply a correctly formatted string, which may cause more of the application to be exposed when the form is submitted and accepted.

You should explore all of your web application with a browser proxying through ZAP. As you do this, ZAP passively scans all the requests and responses made during your exploration for vulnerabilities, continues to build the site tree, and records alerts for potential vulnerabilities found during the exploration.

It is important to have ZAP explore each page of your web application, whether linked to another page or not, for vulnerabilities. Obscurity is not security, and hidden pages sometimes go live without warning or notice. So be as thorough as you can when exploring your site.

You can quickly and easily launch browsers that are pre-configured to proxy through ZAP via the Quick Start tab. Browsers launched in this way will also ignore any certificate validation warnings that would otherwise be reported.

To Manually Explore your application:

Start ZAP and click the Quick Start tab of the Workspace Window.

Click the large Manual Explore button.

In the URL to explore text box, enter the full URL of the web application you want to explore.

Select the browser you would like to use

Click the Launch Browser

This option will launch any of the most common browsers that you have installed with new profiles.

If you would like to use any of your browsers with an existing profile, for example with other browser add-ons installed, then you will need to manually configure your browser to proxy via ZAP and import and trust the ZAP Root CA Certificate. See the ZAP Desktop User Guide for more details.

By default the ZAP Heads Up Display (HUD) will be enabled. Unchecking the relevant option on this screen before launching a browser will disable the HUD.