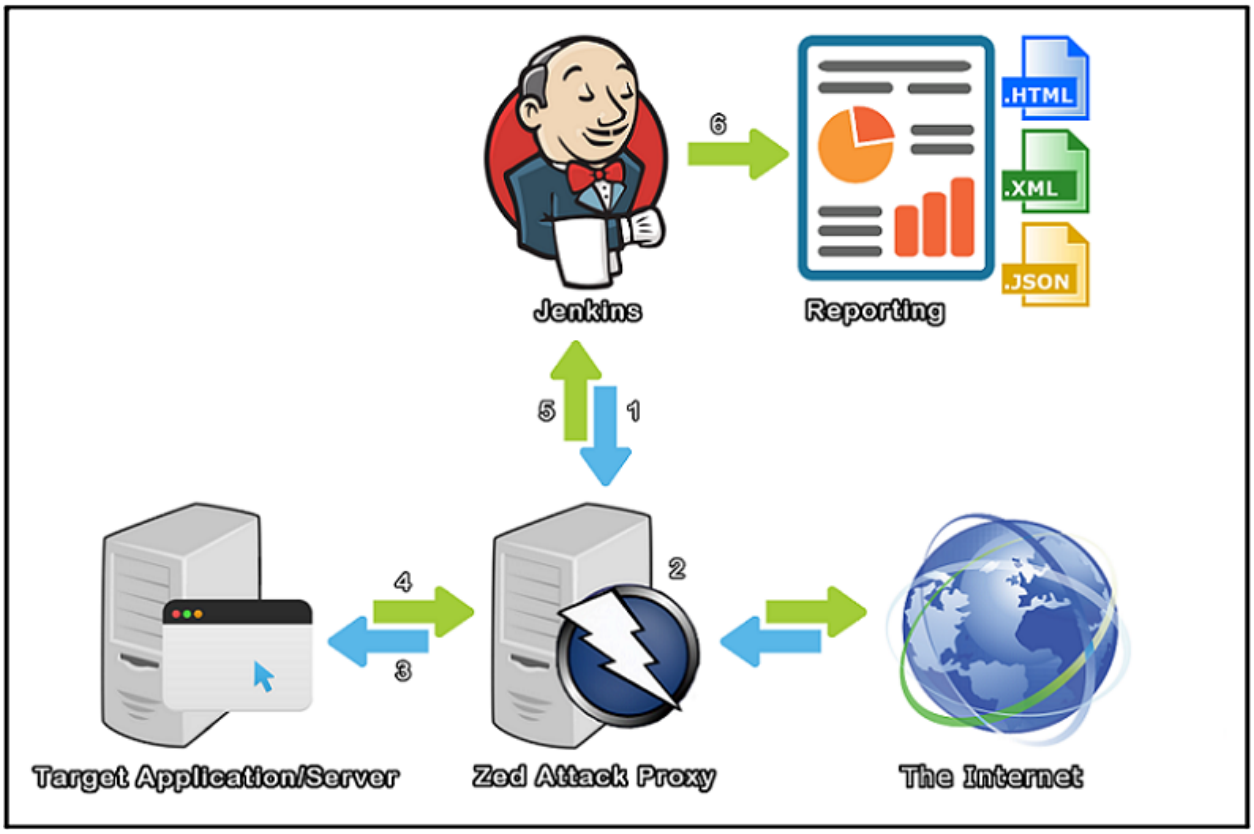
Automatic security tests in Jenkins with OWASP ZAP

Grégoire Willmann Follow
Sep 26, 2018 · 4 min read



OWASP ZAP is a very popular tool used to find vulnerabilities in your codebase and in your instance/server setup.

OWASP ZAP logo

What it basically does is crawl through your website and then scan for vulnerabilities on all the URLs it found during the crawl.

A **session** is an instance of a test. Inside a session you can have multiple contexts.

Contexts help ZAP only scan the URLs you want.

For example if you include directly bootstrap in your pages with:

```
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com
/bootstrap/3.3.7/css/bootstrap.min.css">
```

ZAP will inevitably find this URL. And since you most certainly don't want ZAP to scan https://maxcdn.bootstrapcdn.com for vulnerabilities, you exclude it of the context.
So you include or exclude URLs from the context based on what you want it to scan.

Before following this guide, you should probably play the OWASP ZAP client on your computer to **understand the basic concepts**.

Brace yourself it's going to be a long journey to setup the OWASP ZAP Jenkins plugin!

Download and install OWASP ZAP on your Jenkins instance

Go to https://github.com/zaproxy/zaproxy/wiki/Downloads and download the version of the client for your platform.

Unzip it and move the folder to `/usr/local/bin` for example.

Then set the environment variable `ZAPROXY_HOME` to the path of your ZAP proxy installation folder:

```
vim /etc/environment
```

and paste the following content:

```
ZAPROXY_HOME=/usr/local/bin/ZAP_2.x.x/
```

Install the OWASP ZAP plugin

To install the official OWASP ZAP plugin on your Jenkins instance go

to `Manage Jenkins -> Manage Plugins -> Available (it is a tab) -> look for OWASP ZAP.`

plugin to install

Install it.

Configure the plugin by going to `Manage Jenkins -> Configure System` and filling out the following fields.

Port 8089 is an example, you can choose the port you want here

Create a new Jenkins job

Create a new freestyle project and fill in the following fields:

- Discard old builds

To make sure our project doesn't use too much space

- Build Trigger (optional)

To run the job every sunday at 2AM

- Add the `Execute ZAP` build step

Inside the Execute ZAP build step:

It should reflect the fields values filled in the step where you installed the plugin

Specifies where the OWASP ZAP bin is installed on our Jenkins instance

Should be the path to the directory of the Jenkins job you are creating

Remember when we talked about context? Here you specify which URLs should be included and excluded. Here http://10.0.40.3 is where I host the website I want to test. The * means that I want ZAP to include in the context all the URLs starting by http://10.0.40.3

Tell ZAP to first crawl for URLs and then scan the URLs it found

Tell ZAP which reports to generate and where to place them

Finally go back to the `Session Management` section which requires more explanation than the other ones:

If you tick the checkbox `Persist Session` ZAP will create a new session for you. It is the easiest option to setup but also the least thorough.

You see if your web application has a login page, ZAP won't know the credentials to use in order to gain access to the private zone of your web app. So ZAP will only attack the public part of your website and miss a good portion of it.

To help ZAP know the credentials, what you would have to do is use the GUI client on your computer to generate a ZAP session in which you assign a valid session cookie for example. You would then export and upload the session you created to your new Jenkins Job folder and then tick the `Load Session` checkbox and select your session in the select list.

For our basic example we will tick the Persist Session checkbox

- Add a `Publish HTML reports` post-build step

And that's it! Either manually build the job or wait for your cron schedule to execute it and you should see the HTML report of ZAP tests in your Jenkins job dashboard.

Click on Vulnerability Report to see the results of the security tests

Let me know if I missed anything!