# Integration and Testing

## Device Integration

The program’s code will be contained within a Raspberry Pi. Raspberry Pi is a low-cost computer of small size developed by the Raspberry Pi Foundation in the United Kingdom with the purpose of helping promote the study of computer science and bringing programming to developing countries, but it also sees extended use in other fields such as robotics[[1]](#footnote-1).

Part of the success of the Raspberry Pi is due to the wide variety of models that exist, as well as their accessible price. By default the Raspberry Pi doesn’t include peripherals such as mice, keyboards or even a screen, but most models include several ports that allow such devices to be connected externally. Some models include an Ethernet port that allows the Raspberry Pi to access the Internet.

Raspberry Pi can work with many different operating systems, but this project will use Raspbian, a Debian-based Linux distribution and the primary operating system officially provided by the Raspberry Pi Foundation.

Note that this program can be executed from any laptop, and does not strictly require a Raspberry Pi to run. A Raspberry Pi is chosen for the purposes of these examples because of its small size and portability, as well as its low cost.

## Dependences

In order for the program’s code to run, a series of requirements or dependences must be installed beforehand in the device that will be used to run the scan, whether it is a Raspberry Pi, a laptop or any other compatible device.

First, the appropriate repositories must be cloned to the device. This can be easily achieved through the command line in any operating system that accepts orders via bash shell such as GNU/Linux or OS X. If prompted for a username and password while cloning from GitBook, those fields might be left blank.



Installing GitBook is also required, assuming it’s not installed already:

Finally, it’s necessary to install OWASP ZAP and its Python API. It might be downloaded and installed from the official page, or directly via shell like this:

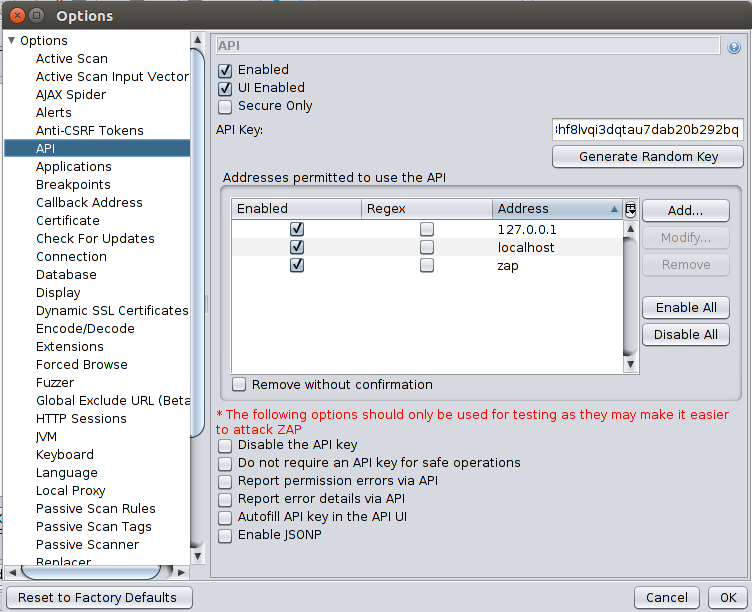


The target containing directory might be changed at will, remembering to change the appropriate “owasp\_location” variable inside the code to reflect that change. It is recommended to always install the newest version available.

In order to solve possible conflicts between different versions of urllib3, running the following command is recommended:



As a last step before the code can be executed, OWASP ZAP must be opened once (zap.sh file) in order to configure the API key. The API key is an optional security measure that isn’t strictly required to run OWASP ZAP, but is nonetheless recommended since it helps prevent malicious sites from accessing the ZAP API. Its configuration can be found under Tools → Options → API:



1 API key configuration in OWASP ZAP

The “Generate Random Key” button will, as its name implies, generate a new API key that can be copied to the api\_key variable used in the code. Alternatively, the “Disable the API key” option may be ticked and the api\_key variable left blank, but this is again not recommended.

## Graphical Interface

1. http://elinux.org/RPi\_Hub [↑](#footnote-ref-1)