# Introduction

## Summary

The aim of this project is to develop a vulnerability scanner for web applications that can, in an automated fashion and with minimal user input, scan a web application and generate a report detailing any vulnerabilities found along with general recommendations to mitigate them.

The program will be as minimalistic and easy to use as possible, so that a person without any technical know-how might be able to run it.

## Motivation

Penetration testing, or pen-testing, is a controlled attack on a system with the aim of finding vulnerabilities the way an attacker would, in order to neutralize them. While it is an effective method to find breaches, many companies forgo its use due to its high cost, which might range from anything from $4,000 up to $20,000 (roughly from 3,500€ to 18,000€)[1].

Studies state that nearly half of breaches exceeded $500,000 (450,000€) in losses in 2015. A survey sent to security professionals reveals that 24% of the respondents had experienced a data or system breach in the last year, and yet 8% of them never perform penetration testing[2].

Thus, while penetration testing is certainly expensive, the losses incurred because of insecure web applications are far steeper. The relatively low budget spent in web application testing compared to other types of security practices might denote an insufficient awareness of the severity of breaches related to web applications.

The purpose of this project is not so much to build an extensive high-end vulnerability scanner, but to raise awareness about the importance of proper security testing. If such a simple and inexpensive tool is able to detect vulnerabilities in a target web application, a skilled attacker with sophisticated tools might put it in jeopardy with much more ease.

## Objectives

This project will automate a simple security audit and generate a report, with the assistance of the OWASP ZAP vulnerability scanner. The device will try different attacks or payloads in an automated fashion to search for vulnerabilities in a target web application. Web applications are susceptible to diverse kinds of attacks, such as XSS (cross-site scripting) or SQL injection, and this makes them common targets for attackers. The findings will be saved into a portable document format (pdf) for later analysis.

While the results might not be comparable to those obtained by a manual pen-testing from a professional, it should be able to find at least common vulnerabilities, and for a very low price. If such a simple, inexpensive tool is capable of compromising a system, then it should become immediately evident that any malicious attacker would also be able exploit those vulnerabilities, and such a system should be deemed insecure and unreliable.

Along with the report, the program will offer a set of general recommendations to secure a typical SME infrastructure against the sort of attacks performed by the program, based on the kinds of vulnerabilities found in the target.

This sort of device is not intended to actually solve the issues found, but merely inform about them. As such, it is designed in such a way that any salesperson should be able to use it without requiring previous technical knowledge.

This sort of simple vulnerability scan could be used to encourage companies to enlist the services of a professional pen-tester that could remediate the found issues, as well as any other problems that an automated scan is unable to detect, in order to avoid financial damage or theft of sensitive data.