[[1]](#footnote-1)

The Benefits of a Vulnerable ASP.NET Web Application (December 2016)

Nicholas G. Gilbert, Student, Covenant College

*Abstract* — Web based cyber-attacks are a major threat to any person or business with an online presence. Applications such as the Damn Vulnerable Web App exist to demonstrate the most common web attacks including SQL Injection, XSS attacks, and brute force attacks. While the DVWA is the go-to application for demonstrating attacks on PHP applications, there is no comparable application for ASP.NET. The goal of this project is to create a DVWA-like application built on ASP.NET.

*Index Terms* — ASP.NET, MVC, Data Security, Information Security, Relational Databases, Web Design, SQL Server

# INTRODUCTION

A

uthor and web developer Peter Welch maintains a blog about his experiences building web applications. Not one to be politically correct, he refers to the internet as “its own special hellscape” from a developer’s perspective [1]. There are many more problems and challenges a programmer faces writing web applications as opposed to traditional programs which are downloaded and installed for offline use. Such challenges include optimizing code so web pages load quickly, maintaining compatibility for multiple, potentially outdated but still used browsers, and scaling applications to handle millions of users. However, the challenge this project will focus on is writing secure web applications.

Most web developers do not start from scratch. Modern websites are built on preexisting frameworks provided by others. With choices ranging from development frameworks like ASP.NET and Ruby on Rails to Content Management Systems like Wix and SquareSpace, web developers rarely have to build from scratch in 2016.

Different Frameworks still have to handle common threats to the security of applications built on them. Regardless of the chosen framework, threats like SQL injection and XSS attacks must still be considered. Frameworks can also have vulnerabilities specific to them like the ASP.NET POET vulnerability which jeopardizes the confidentiality of information stored in an ASP.NET application [2].

To demonstrate cybersecurity threats that web developers face, Dewhurst Security has developed the Damn Vulnerable Web App (DVWA). The DVWA is an intentionally vulnerable PHP/MySQL web app that allows penetration testers to hone their skills and developers see the kind of attacks that their applications can face. Such attacks include SQL injection, XSS attacks, command injection, file upload vulnerabilities, brute forcing, and more.



**Figure 1: The Damn Vulnerable Web App. This application, developed by Dewhurst Security, is used to demonstrate vulnerabilities in web applications that developers need to be aware of when building their applications. It also allows penetration testers to hone their skills without illegally hacking applications.**

# Project Objective

The DVWA is a great tool for demonstrating specific attacks that can occur against any web application. However, it does not take into account the role that frameworks play in the security of web applications. Only one version of the application exists which is written in PHP using a MySQL database. While that is a common combination of tools to use, it is far from the only combination.

The goal of this project is to provide a proof-of-concept for an application like the DVWA built using a non-PHP framework. This will not be a complete application as developing a complete application will take more time than can be given in a single semester. However, a proof-of-concept will be given in the form of an application that demonstrates a few of the most common attacks and how to secure against them. Data and test cases will also be provided to demonstrate the improved security of an application when it is coded to the standards given by the framework provider.

# Approach

## Choosing a Framework and Database

## Which Attacks to Demonstrate

# Collecting Data

# Faith Integration

# Conclusion

References

1. P. Welch, "Programming Sucks," in Still Drinking, 2014. [Online]. Available: https://www.stilldrinking.org/programming-sucks. Accessed: Nov. 28, 2016.
2. Microsoft, "Microsoft security bulletin MS10-070 - important," 2010. [Online]. Available: https://technet.microsoft.com/library/security/ms10-070. Accessed: Nov. 28, 2016.
3. J. Williams and D. Wichers. OWASP top 10 2013: The Ten Most Critical Web Application Security Risks. *OWASP Foundation*, April 2013

1. This paper was submitted for review on December 5, 2016 to Dr. Jeffrey Humphries, Associate Professor of Computer Science at Covenant College in Lookout Mountain, GA. This paper serves as the Bachelor’s Thesis of Nicholas G. Gilbert, fulfilling the requirement of COS492: SIP in Computer Science

   Nicholas G. Gilbert is an undergraduate student studying Computer Science at Covenant College, 14049 Scenic Highway Box 222, Lookout Mountain, GA 30750 USA (email: nickgilbert1994@gmail.com). [↑](#footnote-ref-1)