|  |
| --- |
|  |
| PENETRATION TESTING REPORT |
| REVIEWED BY: COMMONWELATH BANK |
| APPROVED BY: HACKTHISITE |
|  |

Contents

[1. Purpose 3](#_Toc199924715)

[2. Summary 3](#_Toc199924716)

[3. Project Details / Scope 3](#_Toc199924717)

[4. Findings 3](#_Toc199924718)

# Purpose

# completing the “Basic” web challenge from HackThisSite.org, which is an online platform that provides a safe and legal environment for students like you to improve their cyber security skills through a variety of challenges. The challenge is divided into 11 levels and each level ranges from easy to difficult. The purpose of this challenge is to test your skills and knowledge in identifying vulnerabilities and exploiting them. By completing this challenge, you will gain a better understanding of how to identify and exploit vulnerabilities in web applications

# Summary:

# As a cyber security generalist at CommBank, it's important to have a basic understanding of penetration testing. Penetration testing is a way to check the security of computer systems and networks by simulating an attack. This helps identify weaknesses in the system and evaluate the effectiveness of security measures. By regularly doing this, organisations can find and fix potential security problems before they can be exploited by bad people.

# Project Details / Scope

The project details describe the project approach, and the process used according to the defined scope. The scope defines the penetration testing scope, for example, Wi-Fi Network Security Penetration Testing.

# Findings

* Level 1

Password was hidden in the HTML source code

Solution: Inspecting page source revealed the password in the code.

* Level 2

Network Security Sam set up a password protection script. He made it load the real password from an unencrypted text file and compare it to the password the user enters. However, he neglected to upload the password file...

Solution: Due to the absence of a matching file, simply pressing submit would grant access

* Level 3

This time Network Security Sam remembered to upload the password file, but there were deeper problems than that.

Solution: inspect >> <form action " method="post">

<input type="hidden" name="file" value="password.php">

<input type="password" name="password"><br><br>

<input type="submit" value="submit"></form>

Post method posts the password.php to "/missions/basic/3/index.php , hence replacing index.php with password.php reveals the password

* Level 4  
  This time Sam hardcoded the password into the script. However, the password is long and complex, and Sam is often forgetful. So, he wrote a script that would email his password to him automatically in case he forgot. Here is the script

approach: Modify html to send password to us by viewing source and editing

* level 5

Sam has gotten wise to all the people who wrote their own forms to get the password. Rather than learn the password, he decided to make his email program a little more secure.

Similar approach to level 4

* Level 6

Network Security Sam has encrypted his password. The encryption system is publicly available and can be accessed with this form:

Use the form to figure out how the encryption works and use it to decrypt the password

* Level 7

This time Network Security Sam has saved the unencrypted level7 password in an obscurely named file saved in this very directory.  
  
In other unrelated news, Sam has set up a script that returns the output from the UNIX cal command. Here is the script:

Approach: using unix command ls which lists all items in a directory the appropriate index directory will be there

* Level 8

Sam remains confident that an obscured password file is still the best idea, but he screwed up with the calendar program. Sam has saved the unencrypted password file in /var/www/hackthissite.org/html/missions/basic/8/  
  
However, Sam's young daughter Stephanie has just learned to program in PHP. She's talented for her age, but she knows nothing about security. She recently learned about saving files, and she wrote a script to demonstrate her ability  
  
approach: <!-- # exec cmd=”ls ../” --> will take you to the directory containing the index folder with the password

* Level 9

Similar approach with a slight difference 🡪 <!-- # exec cmd=”ls /var/www/hackthissite.org/html/mission/basic/9/” -->

* Level 10

Change authorization using javascript:alert(document.cookie=”level10\_authorized=yes”;

* Level 11

e/l/t/o/n/.htaccess , then access ignore index “DaAnswer”, within a statement will be given containing the password e.g “The answer is easy! Just look a little harder”