|  |
| --- |
| Security Assessment Report |
|  |
| Prepared By: Scott Taylor, Jane Huang, Safraz Iqbal for CS1050-Assignment#5 |
| Date: Feb. 11, 2018 |
|  |

Security Sensitive Information

Table of Contents

[1. Exective Summary 2](#_Toc506400506)

[2. Summary Of Attempted Exploitions 2](#_Toc506400507)

[Server 1: Marks Mail: status OWNED 2](#_Toc506400508)

[Server 2: Portfolio: status OWNED 2](#_Toc506400509)

[Server 3: Kevin’s Website: status FAILED 2](#_Toc506400510)

[Server 4: Best Coffee: status FAILED 3](#_Toc506400511)

[Server 5: Exploitation: status FAILED 3](#_Toc506400512)

[5. Recommandations 3](#_Toc506400513)

# Exective Summary

We have been given a corporate environment with 5 servers for a penetration test. Our goal is to find the vulnerabilities and exploit each server. With following the penetration test methodology and process as below combined using the several of tools, we are able to access and execute our actions to the severs and applications. The summary of attempted exploitation presented in section 2 of this report. The recommendations to mitigate the vulnerabilities that we successful exploited to those servers and applications have been given.

1). Reconnaissance / discovery

2). Enumeration

3). Vulnerability Analysis

4). Execution

5). Document findings

# Summary Of Attempted Exploitions

## Server 1: Marks Mail: status OWNED

* Hostname mail.ctf
* IP Address 10.10.10.6
* NMAP scan reveals open ports 143 IMAP, 443 HTTPS, 22 SSH
* medusa -h mail.ctf -u mark -P /usr/share/wordlists/rockyou.txt -M imap -r 1 -t 100
* password revealed, ssh into server with password

## Server 2: Portfolio: status OWNED

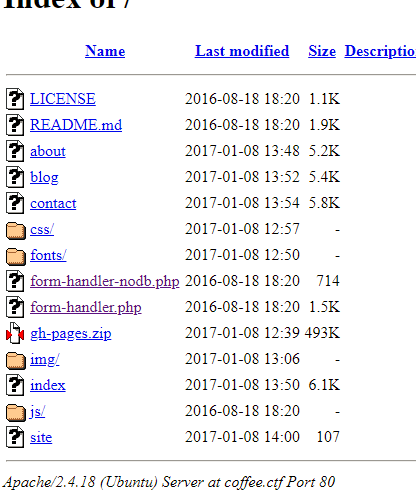
* Hostname portfolio.ctf
* IP Address 10.10.10.7
* NMAP scan reveals open ports 80 HTTP, 22 SSH, Apache Version 2.4.18
* DIRB scan reveals directories /img /mail /vendor
* Investigated directories, found contact\_me.php
* Attempted to inject using BURP successfully
* Sqlmap –u <http://10.10.10.7/contact_me.php>?from=someone --dump reveals access to database and the contents of the table flag

## Server 3: Kevin’s Website: status FAILED

* Hostname www1.ctf
* IP address 10.10.10.9
* NMAP scan port 80 HTTP, 22 SSH, Apache Version 2.4.7
* DIRB scan reveals directory CGI-Bin, unable to access
* Metasploit attempt for Apache Version failed

## Server 4: Best Coffee: status FAILED

* Hostname coffee.ctf
* IP Address 10.10.10.10
* NMAP scan open port 80 HTTP, 22 SSH, Apache version 2.4.18
* Investigating traversable directory structure



* Possible RFI/LFI vulnerability

## Server 5: Exploitation: status FAILED

* Hostname search.ctf
* IP Address 10.10.10.8
* NMAP scan open port 80 HTTP, 22 SSH
* Dirb scan reveals <http://10.10.10.8/search/search.php> to search
* Attempting injection clue echo system($\_GET[‘cmd’]); no luck

# 5. Recommandations

**Server 1:** We were able to complete a dictionary attack in relatively little time. Mark needs to secure his password by making it much longer and more complicated

**Server2:** We were able to dump the SQL tables present on the server. Inputs need to be sanitized in the code to not allow us read or write access. Least-privilege connection to the database must be enforced so that never connect to the database from a web page or web service as high level administrator account.

The user’s input from the webpage needed to be validated and restrict the Meta-character in input fields such as “\*” for SQL statement. Use parameterized/or prepared SQL statement construction to avoid the SQL injections.

Server3: Even though we did have the time to successfully execute our actions but the observation of the vulnerabilities in apache server version out of dated (latest version 2.4.29) and CGI-bin directory exposure could cause potential Denial of Service (DOS) attacks. So this server needs to be hardened and correctly configured to mitigate the potential attacks.

Server4: Even though we did have the time to successfully execute our actions but the observation of the vulnerabilities in apache server version out of dated (latest version 2.4.29) and web site map exposure indicated the web server misconfigured and index.php page maybe missing or crashed. The Apache server need to be patched and configured correctly.

Server5: Even though we did have the time to successfully execute our actions but the observation of Get command i.e. System($\_GET[‘cmd’]) in script indicated the vulnerability that the user’s inputs can be directly executed with any validations on server side. The user input field need to be validated.