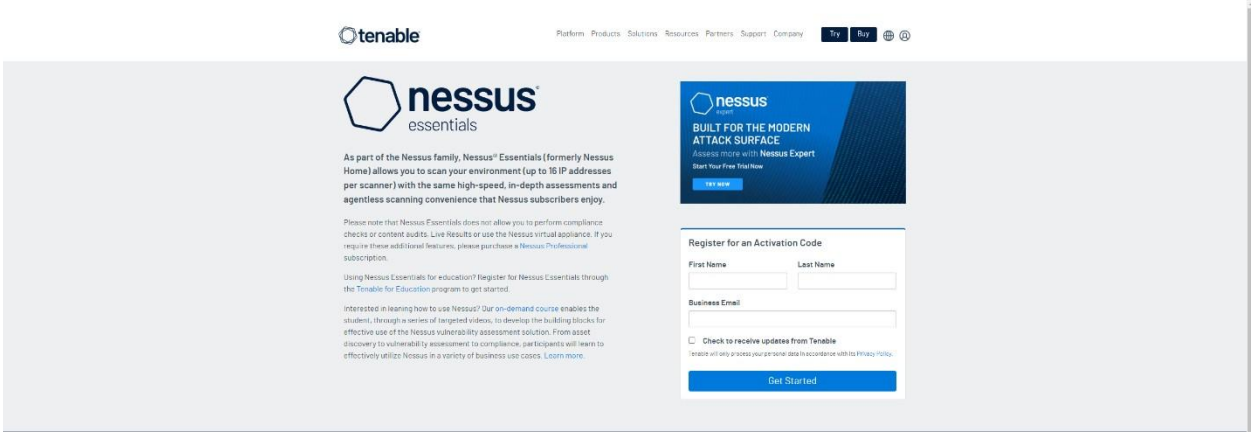
OVERVIEW:

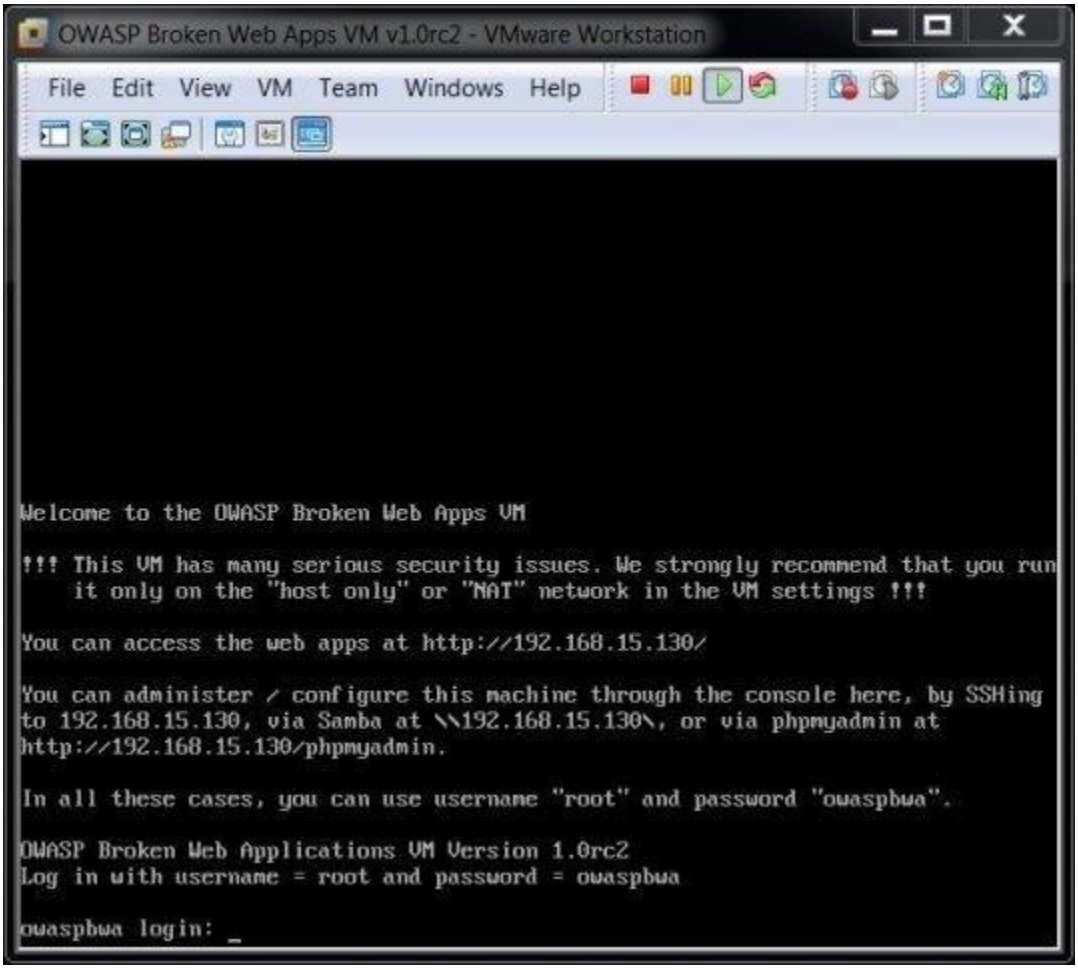
The goal of this lab was to conduct a vulnerability assessment on the OWASP BWA web server and gain a deeper understanding of how vulnerability scanners like Nessus operate. A Nessus scan was performed on the OWASP BWA VM to identify security weaknesses. The detected vulnerabilities were then researched to determine effective mitigation strategies for reducing risk.

ANALYSIS:

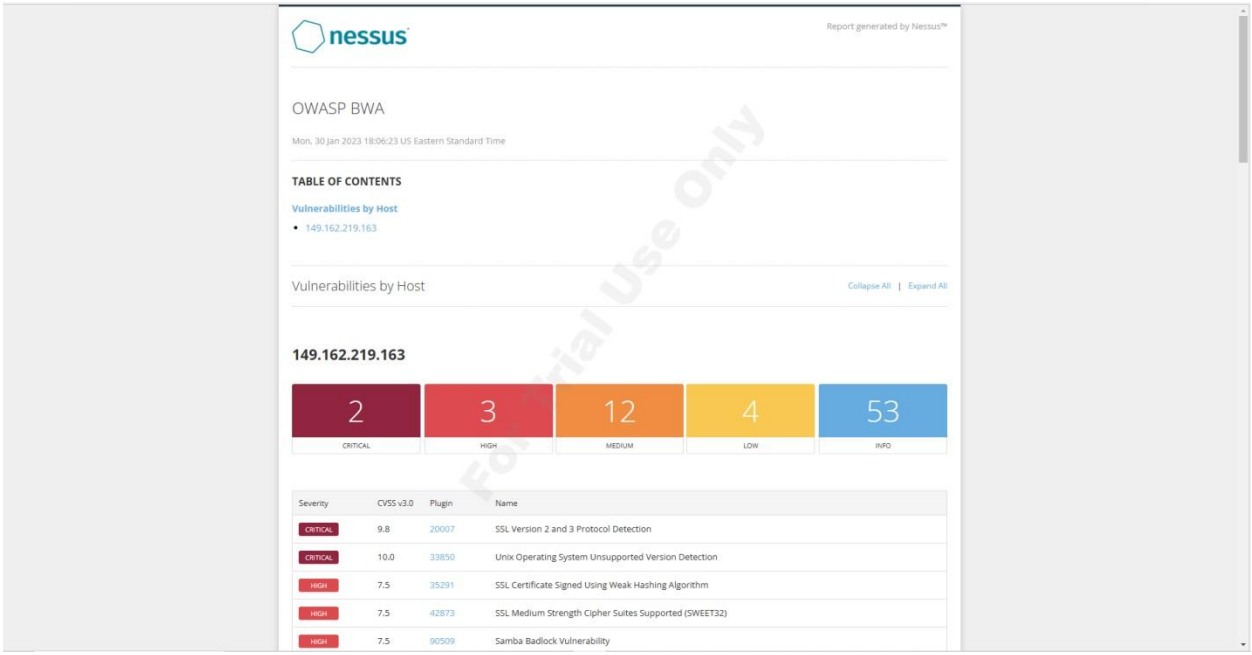
The first step was to download Nessus Essentials and create an account. Through this we would be able to perform our vulnerability scan



OWASP BWA was then downloaded and loaded on to a VM. In my case this was VMWare.



In Nessus you are able to scan OWASP BWA by inserting its IP address. This was done and after a couple of minutes a report was returned.



nessus

Report generated by Nessus™

OWASP BWA

Mon, 30 Jan 2023 18:06:23 US Eastern Standard Time

TABLE OF CONTENTS

Vulnerabilities by Host

Collapse All | Expand All

149.162.219.163

CRITICAL	HIGH	MEDIUM	LOW	INFO
2	3	12	4	53

Severity	CVSS v3.0	Plugin	Name
CRITICAL	9.8	20007	SSL Version 2 and 3 Protocol Detection
CRITICAL	10.0	33850	Unix Operating System Unsupported Version Detection
HIGH	7.5	35291	SSL Certificate Signed Using Weak Hashing Algorithm
HIGH	7.5	42873	SSL Medium Strength Cipher Suites Supported (SWEET32)
HIGH	7.5	90509	Samba Badlock Vulnerability

Critical:

The first of the critical vulnerabilities was SSL
Version 2 and 3 Protocol Detection.

These versions of SSL are affected by several cryptographic flaws, including: An insecure padding scheme with CBC ciphers & Insecure session renegotiation and resumption schemes. A solution to this problem would be to consult the application's documentation to disable SSL 2.0 and 3.0. Then use TLS 1.2 (with approved cipher suites) or higher instead.

The second vulnerability was Unix Operating System Unsupported Version Detection. This means that we were running a version of Unic that was no longer supported. The simple solution is to upgrade to a version of Unix that is supported.

High:

The first high vulerabilty that is shown is that the certificate was signed using a weak hashing algorithm. The solution would be to have SSL certificate reissued.

Another high vulnerability is that the version of Samba we are using is affected by a flaw called Badlock, which leaves us vulnerable to man-in-the-middle attacks. The solution would be to upgrade to Samba version 4.2.11 / 4.3.8 / 4.4.2 or later.

Medium:

One interesting medium vulnerability is that we are using a older version of JQuery, which leaves us vulnerable to cross site scripting. The simple solution is to upgrade to JQuery version 3.5.0 or later.

Another medium vulnerability is that the remote web server supports the TRACE and/or TRACK methods. TRACE and TRACK are HTTP methods that are used to debug web server connections. The solution is to disable this HTTP methods.

Low:

A low vulnerability is that the SSH server is configured to support Cipher Block Chaining (CBC) encryption. This may allow an attacker to recover the plaintext message from the ciphertext. The solution would be to disable CBC mode cipher encryption and enable CTR or GCM cipher mode encryption.

Another low vulnerability is that the remote host has a key that is shorter than 2048 bits. Some browser SSL implementations may reject keys less than 2048 bits after January 1, 2014. Additionally, some SSL certificate vendors may revoke certificates less than 2048 bits before January 1, 2014. The solution would be to replace the certificate in the chain with the RSA key less than 2048 bits in length with a longer key, and reissue any certificates signed by the old certificate.