**Vaibhav Shukla**

**MIS 5211**

**Introduction to Ethical Hacking Analysis Report 2 – Scanning Exercise**

**Date-10/25/2016**

## Executive Summary

For this assignment, I conducted Nessus vulnerability Scan on my two personal systems

### 1) VM Ware WorkStation with Fedora Unix Operating System Installed

Scan took almost 5 min to give a result

- 1 Critical vulnerability
- 2 Medium vulnerability
- 1 Low vulnerability
- 27 Info

Synopsis on Critical and Medium Vulnerability

**Critical Vulnerability**-Fedora Operating system was not supported as it was the Fedora 21 version installed and the version has already had its End of life so it doesn't give any security updates and patches pertaining to this operating system. The step to counter is upgrading the version of the operating system installed to Fedora 23.

**Medium Vulnerability**-FTP was enabled in the system so its allows anonymous users to access the FTP server so if FTP server was not I use it should be disabled or using a secured FTP can be a solution like SFTP (part of the SSH suite) or FTPS (FTP over SSL/TLS).

HTTP trace was allowed in the system. This can cause the XST attack as user can lure the customer to click on the particular link which will trigger an asynchronous HTTP trace call and this will start collecting cookie information via JavaScript and then sends it over to another hacker server that collects the cookie information so the attacker can create a session hijack attack. This is easily mitigated by removing support for HTTP TRACE on all webservers.

### 2) Personal Laptop

Scan took almost 10 min to give result

- 0 Critical vulnerability
- 4 Medium vulnerability
- 0 Low vulnerability
- 30 Info

**Vaibhav Shukla**

**MIS 5211**

**Introduction to Ethical Hacking Analysis Report 2 – Scanning Exercise**

**Date-10/25/2016**

**Medium Vulnerability**

The server's X.509 certificate does not have a signature from a known public certificate authority. The browser maintains an internal list of popular CAs and their public keys and uses the appropriate public key to decrypt the signature back into the digest. It then finds its own digest from the plain text in the certificate and compares the two. If both digests match, the integrity of the certificate is verified and certificate is valid. If the CA is not listed in browser, it throws an exception to add a certificate to the list of trusted CA and this vulnerability pertains to any one of certificate added during such exception.