**NISM Personal Reflection**

# **Introduction**

# The new era of information technology promotes "Internet of Things (IoT), Big Data, Industry 4.0, BYOD (Bring Your Own Device) and CYOD (Choose Your Own systems, the Internet of Things (IoT), Big Data, Industry 4.0, BYOD (Bring Your Own Device) and CYOD (Choose Your Own Device) trends." (Pereira et al., 2017) resulting in security vulnerabilities which can lead to high information risks.

# The NISM module provides an opportunity to identify vulnerabilities in an existing system using multiple tools available in the market. The module encourages students to go beyond the traditional learning method and assist in the evolution of thought processes to understand the mindset of malicious actors.

# **Security Tools**

Following tools have been used for scanning the webserver by the author

* 1. NMAP (Ferranti, 2018)

The NMAP has been able to identify the following:

* The actual name of the hosted webserver.
* Open ports and their types
* Possible Operating System
* The host-key (Which represents the system can only be accessed thru key)
* Webservice Type
* CSRF vulnerabilities
  1. NIKTO (Obbayi, 2019)

The NIKTO has been able to identify the following:

* Server Type
* CSRF and click-jacking vulnerabilities
* File Structure vulnerabilities
  1. LBD

The LBD has been able to identify the following:

* Server type
* No Load balancer was identified
* Webserver name
  1. Nessus (Obbayi, 2019)

The Nessus has been able to identify the following:

* The name of the webserver
* File Structure vulnerability.
* Cypher block chain vulnerabilities related to SSH
* Apache version
* SYN vuklnerabilities
* Identified public key authentication
* Server configuration issues
* Identified exposed bootstrap loaders
  1. ZAP (Obbayi, 2019)

The ZAP identified the following:

* CSRF vulnerabilities
* Cross-Domain vulnerabilities

# **Manual Identification**

# The following codes with API are exposed, where the codes used are visible.

* http://www.123easyinvite.com/assets/css/bootstrap-responsive.min.css
* <https://ajax.googleapis.com/ajax/libs/jquery/1.8.3/jquery.min.js?s=2>

# **Design and Implementation Corelation:**

The design document (link) has been justified since all the tools included and additional tools were utilized to identify the vulnerabilities, please check the link for the final submission (link).

STRIDE has been used to identify the issues using the data from the different tools utilized to identify the Findings and their remediation plans.

**STRIDE**

STRIDE identifies system vulnerabilities following specific threat categories, which are: Spoofing, Data Tampering, Repudiation, Disclosure of Information, DOS, and Privilege escalation – as shown in Figure 1 (Shevchenko, 2018).

Diagram

Description automatically generated

**Figure 1: STRIDE Methodology (EC-Council, 2020).**

Figure 2 displays the individual attack vectors at each stage of STRIDE:

Table

Description automatically generated

**Figure 2: STRIDE Attack Vectors (EC-Council, 2020).**

# **Data Analytics**

# The details obtained from the tools were utilized to identify the major vulnerabilities and their remediations:

**FND 1:** The site is currently missing a Load Balancer or Intrusion Prevention System (IPS), resulting in a successful D-DoS attack. This weakness in the system can result in service disruption.

**Remediation Plan (Recommendations):** Use a Load balancer or IPS

**FND 2:** This plugin is a SYN 'half-open' port scanner. It shall be reasonably quick even against a firewalled target. Note that SYN scans are less intrusive than TCP (full connect) scans against broken services, but they might cause problems for less robust firewalls and also leave unclosed connections on the remote target if the network is loaded.

**Remediation Plan:**

* Protect your target with an IP filter.
* Implement a hardware/software Firewall.

**FND 3:** The website is susceptible to Man In The Middle attack, and the data transferred is not encrypted since it has HTTP port 80 open.

**Remediation Plan:** Block port 80

**FND 4:** The SSH port number, SSH version, the Apache webserver version, the Fully Qualified Domain Name of the system, and the Operating System type is visible can result in targeted D-Dos and Brute force attack. 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.

**Remediation Plan:**

* Change SSH configuration and update Apache to the latest version.
* Contact the vendor or consult product documentation to disable CBC mode cypher encryption, and enable CTR or GCM cypher mode encryption.

**FND 5:** The system is susceptible to Cross-Site Request Forgery and can lead to unauthorized privileged access to the attacker

**Remediation Plan:**

* Block port 80
* Enable X-Framework Option
* Implementing Content Security Policy's "frame-ancestors" directive
* Change FRAMESET to "DENY"
* X-XSS-Protection header to be defined
* Anti-MIME-Sniffing header X-Content-Type-Options to be set as 'nosniff'
* Block access to the following pages to ensure JavaScript source files are loaded from only trusted sources, and the sources can't be controlled by end-users of the application.
  + http://www.123easyinvite.com/assets/css/bootstrap.min.css
  + http://www.123easyinvite.com/assets/css/bootstrap-responsive.min.css
  + https://ajax.googleapis.com/ajax/libs/jquery/1.8.3/jquery.min.js?s=2

# **Roles and Responsibility**

Tasks had been assigned depending upon the experience and understanding of the subject matter. Author was tasked with the following:

* Identify the tools to be used along with other team members
* Identify the vulnerabilities using NIKTO, LBD, NMAP, ZAP and Nessus
* Identifying the key security findings along with the remediation plans and to assign criticality.

# **Learnings**

The module has been helpful for the author to:

* Identify the penetration testing tools available in the industry and their functionality.
* Implementation of the pen-testing tools.
* Identify vulnerabilities and their remediation.
* Identify vulnerabilities manually and interpret the security challenges.

**Reference List**

EC-Council (2020) What Is Stride Methodology In Threat Modeling? Available from: <https://blog.eccouncil.org/what-is-stride-methodology-in-threat-modeling/> [Accessed 12 June 2021].

Ferranti, M. (2018) What is Nmap? Why you need this network mapper. Network World. Available from: [https://www.networkworld.com/article/3296740/what-is-nmap-why-you-need-this-network-mapper.html](%20https://www.networkworld.com/article/3296740/what-is-nmap-why-you-need-this-network-mapper.html) [Accessed 07/06/2021].

Obbayi, L. (2019) A brief introduction to the Nessus vulnerability scanner. Infosec Resources. Available from: [https://resources.infosecinstitute.com/topic/a-brief-introduction-to-the-nessus-vulnerability-scanner/](%20https://resources.infosecinstitute.com/topic/a-brief-introduction-to-the-nessus-vulnerability-scanner/) [Accessed 07/06/2021].

Pereira, T., Barreto, L., & Amaral, A. (2017) Network and information security challenges within Industry 4.0 paradigm. Available From: https://www.sciencedirect.com/science/article/pii/S2351978917306820 [Accessed 24 July 2021]

Shevchenko, N. (2018) Threat Modeling: 12 Available Methods. Available from: <https://insights.sei.cmu.edu/sei_blog/2018/12/threat-modeling-12-available-methods.html> [Accessed 12 June 2021].