Conclusion: Final Evaluation of GetIntoPC's Security Readiness

CONDUCTED BY -E-DOT

Penetration Testing Report for getintopc.com

Table of Contents

---

1. Introduction

The purpose of this report is to document the penetration testing conducted on **getintopc.com**. The testing aims to evaluate the site's security posture by identifying potential vulnerabilities, assessing web application security, and reviewing network configurations. This report outlines the findings from the tests performed using various penetration testing tools and provides recommendations to mitigate identified risks.

---

2. Objective

The main objectives of the penetration test are:

- To identify vulnerabilities within the web application that could be exploited by attackers.
- To assess the effectiveness of existing security measures and configurations.
- To provide actionable recommendations for improving the security posture of **getintopc.com**.

---

3. Methodology

Tools Used

The following tools were utilized for the penetration testing process:

1. **SQLMap**: A powerful tool for automating the process of detecting and exploiting SQL injection vulnerabilities. SQLMap can perform various tests on web applications to retrieve database management system information.
2. **Nikto**: A web server scanner that scans for various vulnerabilities, configuration issues, and security concerns. Nikto can identify missing security headers, outdated software, and potential file exposure vulnerabilities.
3. **Nmap**: A network scanning tool used to discover hosts and services on a computer network. Nmap provides information about open ports, running services, and potential operating system fingerprints.
4. **WhatWeb**: A web application fingerprinting tool that identifies the technologies used on a web application, including server software, frameworks, and CMS platforms.
5. **Metasploit**: A penetration testing framework that allows security professionals to find and exploit vulnerabilities in systems. Metasploit can be used for SQL injection, remote code execution, and other types of attacks.

---

4. Findings

4.1 SQL Injection Testing

The first test involved assessing the endpoint http://getintopc.com/page?id=1 for SQL injection vulnerabilities using SQLMap. The following findings were observed:

- **Connection Errors**: The web server returned 403 Forbidden responses, indicating that access to the page was restricted.
- **Injection Assessment**: SQLMap determined that the id parameter does not appear to be injectable based on initial tests.

Command Example:

bash

Copy code

```
sqlmap -u "http://getintopc.com/page?id=1" --dbs
```

Detailed Findings:

- The server's response to injection attempts showed consistent errors, suggesting that the server may be protected by a Web Application Firewall (WAF) or similar security mechanisms.

Recommendations:

- To further investigate potential injection points, increase the level and risk parameters to perform more comprehensive tests:

bash

Copy code

```
sqlmap -u "http://getintopc.com/page?id=1" --dbs --level=5 --risk=3
```

- Consider using tamper scripts to bypass WAF protections:

bash

Copy code

```
sqlmap -u "http://getintopc.com/page?id=1" --dbs --tamper=space2comment
```

4.2 Vulnerability Scanning

The Nikto scan revealed several vulnerabilities related to server configurations and security practices:

- **Missing Security Headers**: The absence of X-Frame-Options and X-Content-Type-Options headers increases the risk of clickjacking and MIME type confusion attacks.
- **Cloudflare Protection**: The site is protected by Cloudflare, which may mitigate some vulnerabilities but can also complicate testing.

Command Example:

bash

Copy code

```
nikto -h http://getintopc.com
```

Detailed Findings:

- The results highlighted that without proper security headers, the site could be vulnerable to certain attack vectors. Additionally, the presence of a proxy (Cloudflare) may obscure some vulnerabilities, making further testing necessary.

4.3 Network Scanning

The Nmap scan identified the following open ports on the target host 172.67.75.211:

- **80/tcp**: Open for HTTP
- **443/tcp**: Open for HTTPS
- **8080/tcp**: Open for HTTP
- **8443/tcp**: Open for HTTPS

Command Example:

bash

Copy code

```
nmap -sV getintopc.com
```

Detailed Findings:

- All traffic is proxied through Cloudflare, which can mask the actual server and services running behind it. The presence of multiple open ports increases the attack surface, necessitating further scrutiny of the services running on those ports.

Recommendations:

- Regularly review and update firewall rules to limit access to only necessary ports.
- Consider implementing rate limiting and IP whitelisting on open ports to reduce exposure.

4.4 Web Application Fingerprinting

Using WhatWeb, various technologies used by **getintopc.com** were identified:

- **Content Management System (CMS)**: WordPress version 6.4.3
- **Plugins**: WordPress Super Cache and other relevant plugins.

Command Example:

bash

Copy code

```
whatweb http://getintopc.com
```

Detailed Findings:

- Identifying the WordPress version and associated plugins can help pinpoint known vulnerabilities. It is essential to keep the CMS and plugins updated to mitigate risks.

---

5. Recommendations

General Recommendations

- **Regular Software Updates**: Ensure that the WordPress installation and all plugins are updated to their latest versions to reduce vulnerabilities associated with outdated software.
- **Security Headers Implementation**: Configure the web server to include necessary security headers:
  - X-Frame-Options: SAMEORIGIN
  - X-Content-Type-Options: nosniff
  - Strict-Transport-Security: max-age=31536000; includeSubDomains; preload
- **WAF Configuration**: If a Web Application Firewall (WAF) is in use, configure it to provide additional logging and alerting on suspicious activities.

Advanced Testing Recommendations

- **Obtain Permission for Deeper Testing**: Before conducting more invasive tests, such as SQL injection and remote code execution, ensure explicit permission from the website administrators.
- **Regular Security Audits**: Conduct regular security audits and penetration tests to stay ahead of potential vulnerabilities and emerging threats.

---

6. Conclusion

The penetration testing conducted on **getintopc.com** revealed several areas for improvement, particularly concerning security headers and the potential for SQL injection. Although the site is protected by Cloudflare, vulnerabilities related to the underlying web application remain a concern.

Implementing the recommendations provided in this report will significantly enhance the security posture of the site, reducing the risk of exploitation and ensuring better protection of user data.