### **Website Scanned:**

hack-yourself-first.com  
**Date of Scan:** 7 September 2025  
**Tools Used:** Nmap, Nikto, WhatWeb, Whois, Netsparker report reference

### **Findings**

Below is a summary of what I discovered about the target website during my scans.

#### **1. Operating System**

* The exact operating system couldn’t be confirmed from the basic scan.
* Based on the server type (Microsoft-IIS/10.0), it is very likely running on **Windows Server**.
* A more detailed Nmap scan with OS detection (nmap -O) could help confirm this.

#### **2. Web Server Software**

* The server is running **Microsoft-IIS/10.0**.
* This was identified through the Server header in HTTP responses and verified using the WhatWeb tool.

#### **3. Technology Stack**

* The application is built using **ASP.NET MVC 5.1** and **ASP.NET 4.0.30319**.
* The X-Powered-By: ASP.NET and X-AspNetMvc-Version headers confirmed this.

#### **4. CMS (Content Management System)**

* No standard CMS (like WordPress or Drupal) detected.
* The site uses a **custom-built web application** developed with ASP.NET MVC.

#### **5. Protection Mechanisms**

* **HSTS is not enabled** – this means users could potentially connect over insecure HTTP.
* No evidence of a **Web Application Firewall (WAF)** or **CDN** such as Cloudflare.
* Security headers like X-Frame-Options, Content-Security-Policy, and X-XSS-Protection were missing.

This lack of security headers leaves the site vulnerable to certain types of attacks.

#### **6. Hosting Information**

* The hosting provider was not immediately identifiable in the initial scan.
* Running a whois or dig command would help identify the IP address and hosting company.

Example:

whois hack-yourself-first.com

dig hack-yourself-first.com

#### **7. Open Ports**

* Expected open ports:
  + **80 (HTTP)**
  + **443 (HTTPS)**
* Additional scans should be run to check for other open ports, such as:
* nmap -sS -p- hack-yourself-first.com
* If port **22 (SSH)** or **3389 (RDP)** is open, that would be unusual and potentially a security concern.

#### **8. Known Vulnerabilities**

Based on my scans and the official **Netsparker report** for this site:

* **SQL Injection**
  + Confirmed in the Cylinders parameter of the CarsByCylinders page.
  + This allows attackers to run malicious database queries.
* **Login Forms over HTTP**
  + Login forms are being served over insecure HTTP instead of HTTPS, exposing user credentials to interception.
* **Missing or Weak Security Headers**
  + Strict-Transport-Security (HSTS) missing.
  + X-Frame-Options missing (vulnerable to clickjacking).
  + No Content-Security-Policy header.
* **Error Message Disclosure**
  + Pages like /trace.axd and /elmah/detail reveal sensitive debugging information that could help attackers.

#### **9. Software Versions**

| **Software** | **Version Found** | **Status** |
| --- | --- | --- |
| Microsoft-IIS | 10.0 | Up-to-date (with Windows updates) |
| ASP.NET MVC | 5.1 | Outdated, newer versions available |
| ASP.NET Framework | 4.0.30319 | Outdated, update recommended |

**Recommendation:** Update ASP.NET MVC and ensure Windows updates are current to reduce exposure to known vulnerabilities.

### **Summary Table**

| **Category** | **Finding** |
| --- | --- |
| **Operating System** | Likely Windows Server |
| **Web Server** | Microsoft-IIS/10.0 |
| **Tech Stack** | ASP.NET MVC 5.1, ASP.NET 4.0.30319 |
| **CMS** | None detected – custom-built |
| **Protection** | No HSTS, missing headers, no WAF/CDN |
| **Hosting** | Needs WHOIS lookup |
| **Open Ports** | 80 (HTTP), 443 (HTTPS), others TBD |
| **Vulnerabilities** | SQLi, HTTP login form, missing headers, error disclosures |

## **FAQ Section – Example Questions**

**Q1. How do I check if HSTS is enabled?**  
Use:

curl -I https://hack-yourself-first.com

Look for the Strict-Transport-Security header. If it’s missing, HSTS is not enabled.

**Q2. How can I detect what technologies the site uses?**  
Run:

whatweb hack-yourself-first.com

Or use browser extensions like **Wappalyzer**.

**Q3. How do I confirm a SQL Injection vulnerability?**  
Use **sqlmap**:

sqlmap -u "https://hack-yourself-first.com/CarsByCylinders?Cylinders=4" --dbs

Only test on authorized systems!

## **Reflection – E-Portfolio Entry**

**Challenges I Faced:**  
At first, identifying the exact operating system was tricky. Nmap’s OS detection gave general results, so I had to infer from the web server type and version. Another challenge was that some scans were blocked or limited, possibly due to network protections.

**How I Overcame Them:**  
I combined different tools instead of relying on one. For example:

* Used **WhatWeb** to confirm technologies.
* Cross-checked results with Netsparker’s report for accuracy.
* Supplemented scans with manual header reviews using curl.

**Impact on Final Report:**  
These experiences reminded me that penetration testing isn’t about one perfect tool — it’s about **layering different techniques** and correlating results. In my final report, I will emphasize the importance of cross-verifying findings and clearly documenting each step.

### **Citations & References**

* Troy Hunt – Hack Yourself First: [https://hack-yourself-first.com](https://hack-yourself-first.com/)
* Netsparker Sample Report: https://files.troyhunt.com/hackyourself-netsparker
* OWASP Cheat Sheets: [https://cheatsheetseries.owasp.org](https://cheatsheetseries.owasp.org/)
* NVD (National Vulnerability Database): https://nvd.nist.gov/