Student:	Email:
Bradley Adams	badams10@my.athens.edu

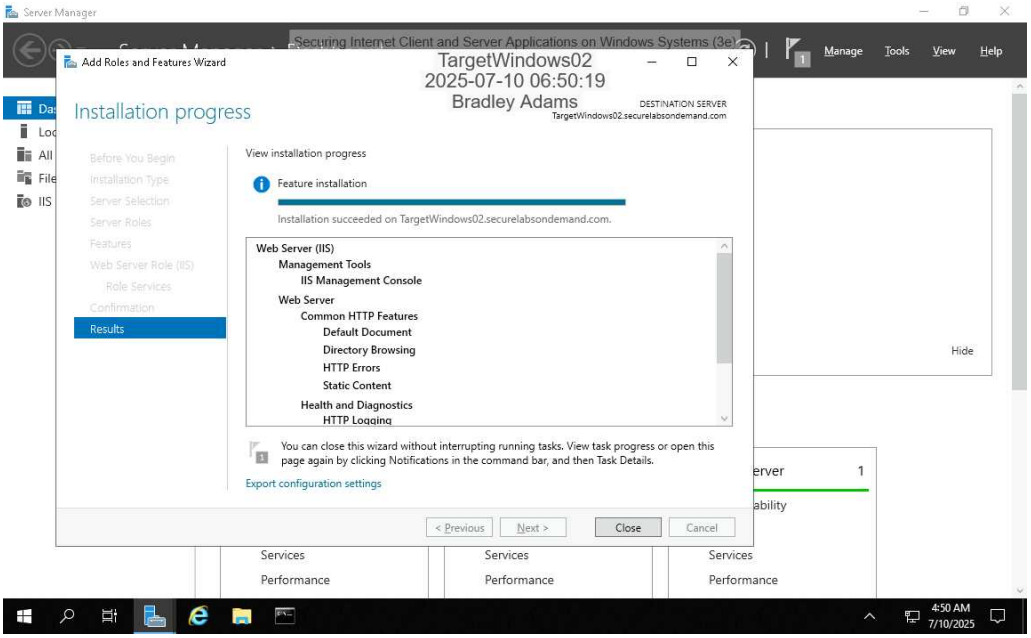
Time on Task:	Progress:
8 hours, 6 minutes	100%

Report Generated: Thursday, July 10, 2025 at 10:12 AM

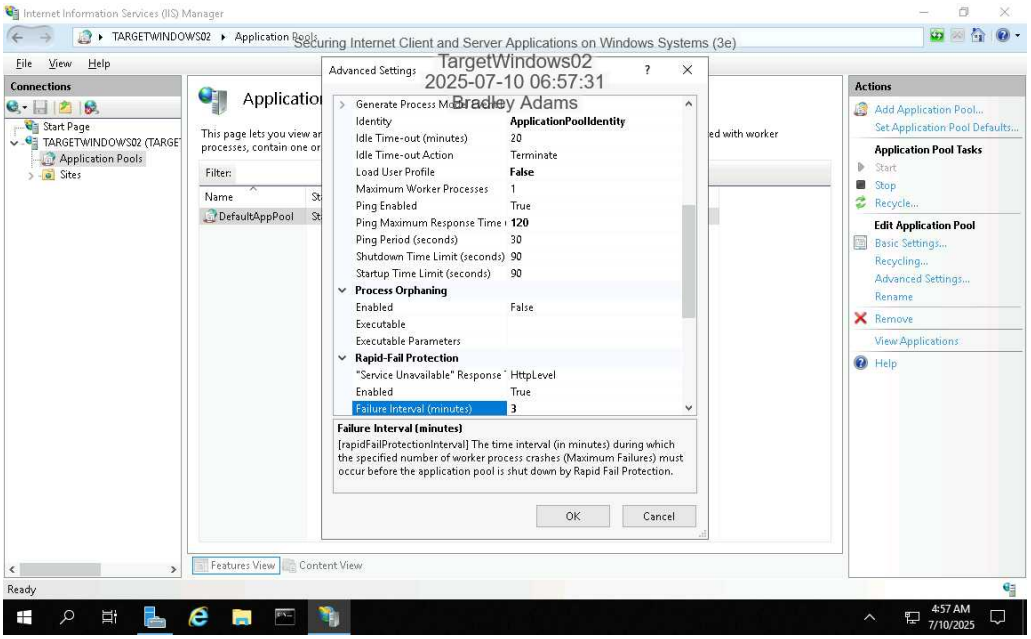
Section 1: Hands-On Demonstration

Part 1: Harden an IIS Web Server

13. **Make a screen capture** showing the **successful installation of the Web Server (IIS) role**.

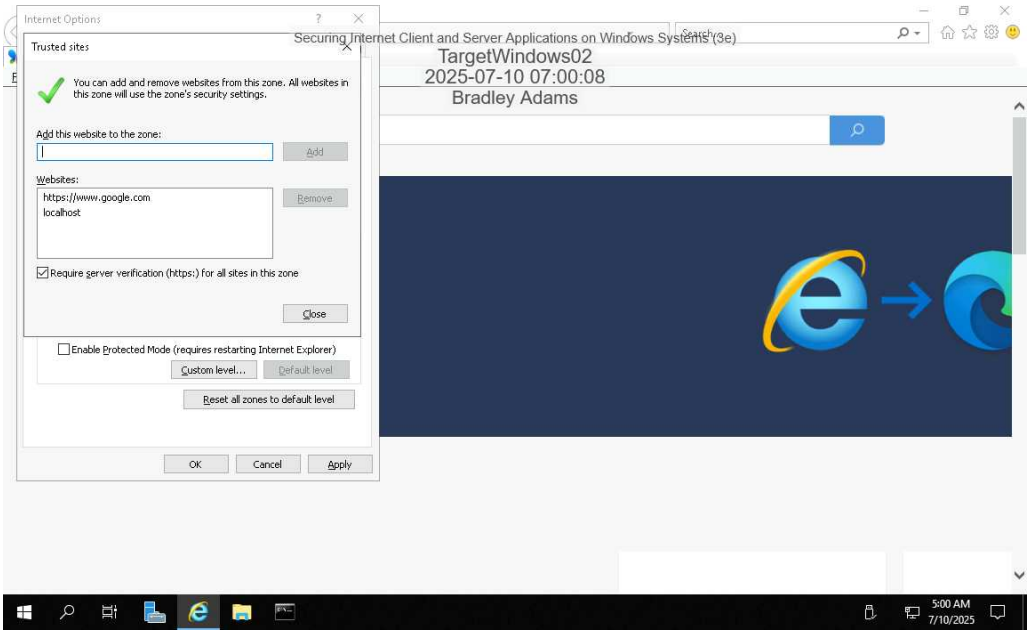


30. **Make a screen capture** showing **both changes to the Advanced Settings**.

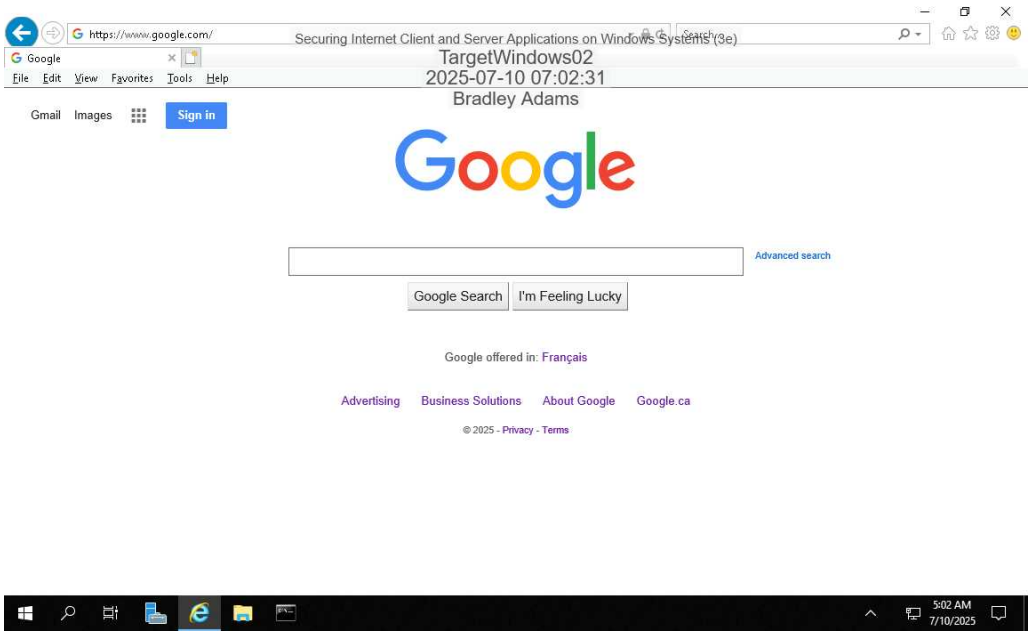


Part 2: Harden an Internet Explorer browser

7. **Make a screen capture** showing the **change to the Trusted Sites dialog box**.



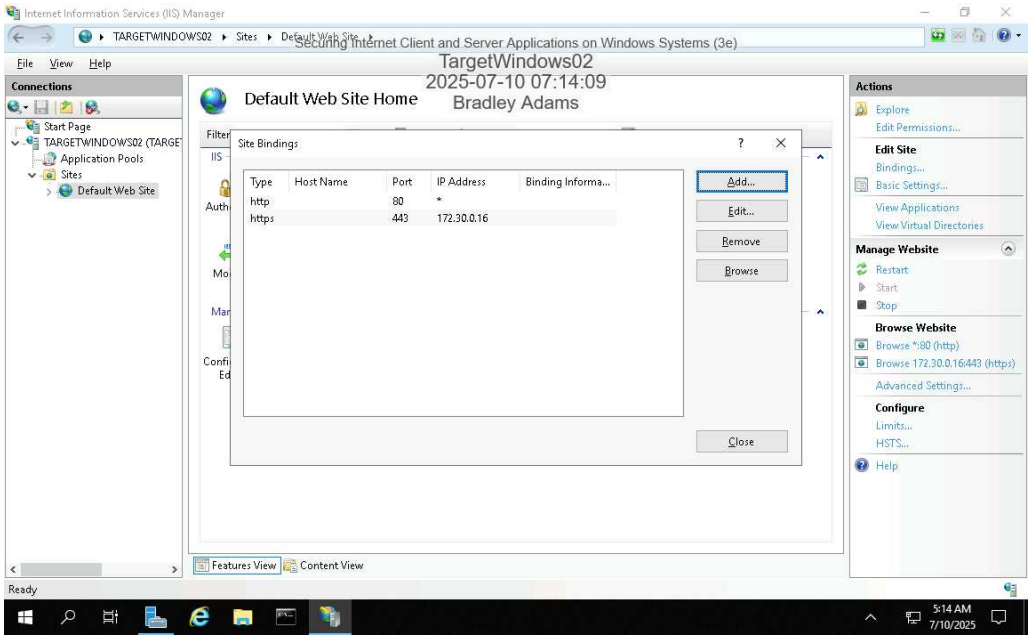
20. **Make a screen capture** showing the **google.com website in the browser window**.



Section 2: Applied Learning

Part 1: Harden an IIS Web Server

8. **Make a screen capture** showing the **new HTTPS binding**.



13. **Document** the purpose of the Managed pipeline mode settings.

The Managed Pipeline Mode setting in Application Pools determines how the server processes requests for managed code applications. This setting is critical because it controls the integration between IIS and the ASP.NET runtime, which affects how HTTP requests are handled within the server pipeline. Selecting the appropriate pipeline mode is crucial for ensuring compatibility and optimal performance when deploying web applications on Microsoft Windows Web Server.

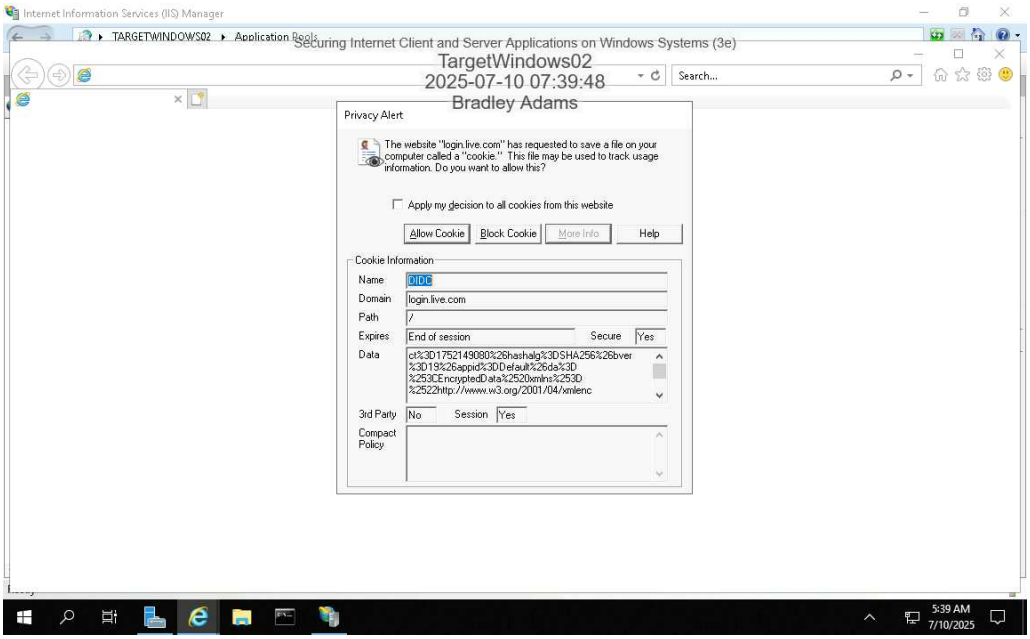
The Integrated mode enables IIS and the ASP.NET runtime to collaborate through a unified request-processing pipeline. This means that managed modules, such as those for authentication or error handling, can be used alongside native IIS modules seamlessly. Integrated mode offers greater flexibility, improved performance, and enhanced features, including enhanced request filtering and security.

The Classic mode emulates the older IIS 6.0 processing model, where ASP.NET handles requests only after they pass through the IIS pipeline. In this mode, the request processing is split, and managed code executes later in the pipeline. This mode may be necessary for legacy applications that rely on the older model and cannot function correctly under the Integrated pipeline.

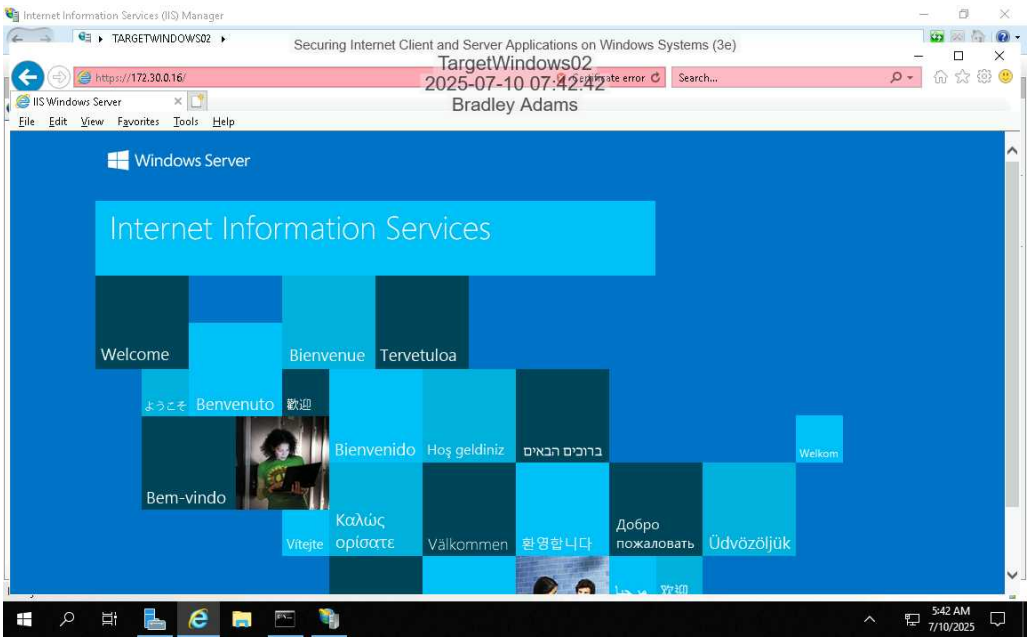
Sources:
https://www.infoworld.com/article/2250731/how-to-manage-and-configure-application-pools-in-iis.html
https://learn.microsoft.com/en-us/dotnet/api/microsoft.web.administration.applicationpool.managedpipelinemode?view=iis-dotnet

Part 2: Harden an Internet Explorer browser

8. **Make a screen capture** showing the **cookie information**.



11. **Make a screen capture** showing the **TargetWindows02 website in the browser window**.



Section 3: Challenge and Analysis

Part 1: Analysis and Discussion

Why are there so few steps to secure default IIS in Windows Server 2019?

The default installation requires relatively few hardening steps because Microsoft has designed it with a strong security posture from the beginning. Microsoft employs a secure-by-design philosophy. By installing only the core web server components and requiring administrators to add additional modules, the default configuration minimizes the attack surface. It uses low-privilege application pool identities, supports modern secure protocols, disables outdated ones, and restricts script or executable execution unless explicitly configured. Logging is enabled by default to support auditing and incident response. Administrators must still configure HTTPS, apply updates, and enforce firewall and access controls tailored to their specific environment and application needs.

Sources:
https://www.infoworld.com/article/2250731/how-to-manage-and-configure-application-pools-in-iis.html
https://learn.microsoft.com/en-us/iis/get-started/introduction-to-iis/introduction-to-iis-architecture

Part 2: Tools and Commands

Use the Internet to research hardening suggestions for your preferred browser. Describe the hardening measures you plan to make on your own Internet browser as a result of this lab.

My go to browser is Mozilla Firefox. Here are a few hardening suggestions:

Configuring Enhanced Tracking Protection in strict mode and enabling DNS-over-HTTPS not only prevents cross-site tracking but safeguards your DNS queries from interception. This sends your lookups through trusted encrypted channels and neutralizes operator-level network tampering.

Using a custom user.js file with privacy?focused settings systematically turns off telemetry, studies, and crash reporting across the board. This guarantees Firefox won’t phone home, drastically reducing passive data leakage.

Deploying uBlock Origin with robust filter lists (including EasyList, EasyPrivacy, AdGuard Tracking Protection, phishing, and malware lists) gives an extra layer of protection against intrusive ads and malicious domains.

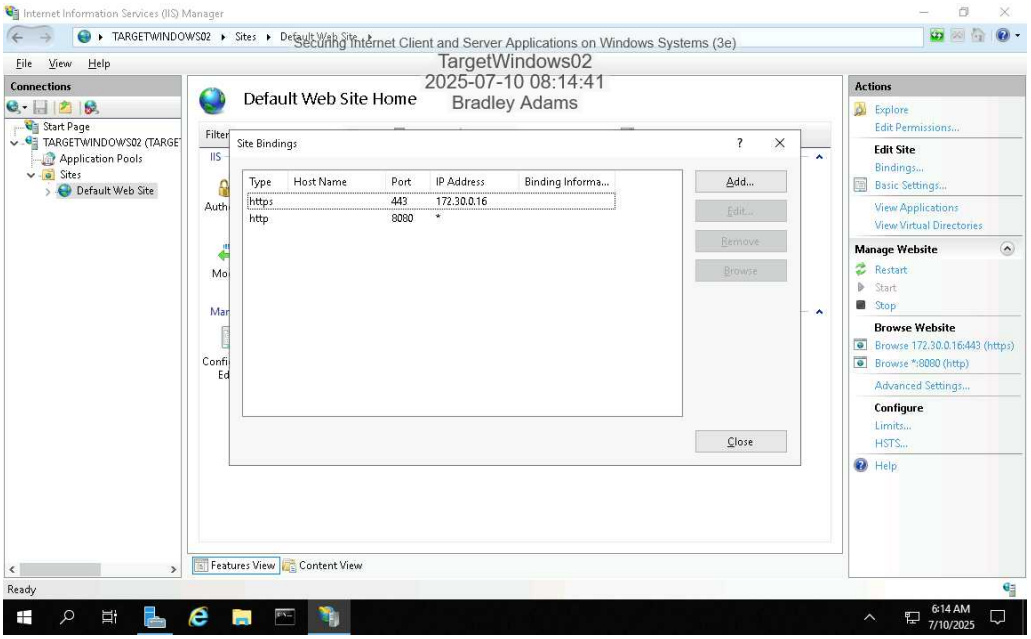
Activating privacy.resistFingerprinting (RFP) drastically hampers fingerprint-based tracking by standardizing behaviors, turning off system font/color leaks, and locking down APIs.

Enabling HTTPS?Only Mode everywhere forces every website connection to use secure TLS with robust ciphers, preserving data integrity and confidentiality even if you accidentally visit insecure URLs.

Source:
https://brainfucksec.github.io/firefox-hardening-guide-2025

Part 3: Challenge Exercise

Make a screen capture showing the **updated default port number**.



Use the internet to research what port 8080 is typically used for. Provide a summary of your findings.

Port 8080 is most commonly employed as an alternative HTTP port, running web or application servers when the default port 80 is unavailable, restricted, or already assigned to another service. This often occurs in development environments where administrative privileges are required to bind to ports below 1024. The 8080 port is popular for proxy services, load balancers, or secondary internal sites. It is frequently used in conjunction with other management interfaces to distinguish services without altering domain names.

Source:
https://cheapsslweb.com/blog/what-is-port-8080-http-port-80-vs-8080-vs-443-difference/