

- Filename: eccouncil-ceh31250-v10-13-1-1-hacking\_web\_servers.md
  - Show Name: CEHv10 (312-50)
  - Topic Name: Attacks and Exploits
  - Episode Name: Hacking Web Servers
  - Description: In this episode, Daniel and Zach take a look at common attack tactics and techniques used when hacking web servers. Here they go over the possible motivations for targeting web servers. They also look at both general and specific vulnerabilities associated with web servers that an attacker may exploit and the common methodology employed.
- 

## Hacking Web Servers

- **Why would a bad actor target a web server?**
  - Hacktivism
  - Flag Planting
  - Rival organization
  - Data theft
    - CC Numbers
    - PII
  - Launch further attacks on other users/organizations
  - Pivot to other internal systems
- **What kind of web servers are we likely to encounter?**
  - Microsoft IIS (Internet Information Service)
    - Full enterprise level Web Server
  - Apache
    - Most popular
    - Open Source
  - Others
    - NGINX
- **What kind of vulnerabilities lead to a compromise of web servers?**
  - OS Vulns
  - Application Vulns
    - Command/Code injection
      - SQLi
      - XSS
      - Authentication Bypass
      - Directory Traversal
  - Misconfigurations
    - Default settings
    - Viewable config files
    - Directory listing enabled
  - Poor/No security controls
    - Weak/default/no accounts/passwords
      - Brute-force/Dictionary attacks
    - Weak/No encryption
  - Denial of Service

- DNS Amplification Attack
- Advanced attacks
  - HTTP Response Splitting
  - Web Cache Poisoning
- How does one begin an attempt to compromise a Web Server?
  - Gather Info
    - Learn about the organization that runs the server
    - Gather possible usernames and passwords
      - TheHarvester
      - kewl
      - Whois search
    - Investigate the site running on the server
      - Check for "powered by" language
      - Look at source code
      - Attempt to discover coding languages and frameworks used
      - Look for hidden content
        - robots.txt files can be helpful
    - Use nmap
      - Enumerate services
        - Versioning
        - OS
    - Create an offline copy for testing
      - Httrack
    - Look for site directories and interesting pages
      - dirb, gobuster, dirbuster, ZAP, Burp
    - Scan for known vulnerabilities
      - Nikto
      - Skipfish
      - Acunetix
      - Exploit-db
    - Use Intercept Proxies
      - Burp
      - ZAP
  - Now attack the server
    - Brute-force authentication
      - Hydra, Ncrack, Patator, Medusa
    - Use intercept proxies
      - Modify GET/POST data
      - Injection attacks
    - Use exploitation frameworks
      - Metasploit
        - nmap Scan bWAPP
        - See distcc port open
        - Search for distcc
        - Exploit

- **What can be done to harden security for a web server?**

- Tried and true hardening techniques
  - Patches/Updates
    - Patch management systems help with this
  - Secure coding practices
  - Disable unnecessary services
  - Use encryption
  - Change default accounts/creds
  - Disable directory listing
  - Monitor logs
  - Employ WAF
  - URL Filtering
  - Regularly perform vulnerability analysis and Pentesting