Mapping

Mapping the Application

- Gathering and examining the key attack surfaces of an application to figure out the surfaces susceptible to exploitation

- Typical Application, common browsing, following every link and experimenting every feature like a normal user is enough for enumerating content. If site contains a "site map" this is a good starting point.

- Comprehensive Approach
  → Web Spidering :
    ⇒ Automated Spidering
    ⇒ User Directed Spidering

- Brute Force Techniques for Content Discovery
  → Brute forcing directories and subdirectories
  → Best approach is to automate some functionality and combine it with manual review.

- Look for inferences from already discovered content
  → Patterns and Schemes
  → Numerical Sequences
  → Common and mandatory directories

- Use of Public Information
  → Use search engines' caching abilties
  → Use waybackmachine for browsing snapshots

- Analyzing the Application

  → Identifying User Input Entry Points
    ⇒ URL
    ⇒ parameter in URL
    ⇒ parameter in POST
    ⇒ Every Cookie
    ⇒ Every HTTP Header
      • User-Agent
      • Referer
      • Accept
      • Accept-Language
    ⇒ Out of band Channels

  → Identifying Server Technologies
    ⇒ Banner Grabbing - Headers such as Server, Templates used for HTML pages, Custom Headers, URL query string params
    ⇒ HTTP Fingerprinting
    ⇒ File Extensions - Querying for common server file extension (asp, aspx, php, py , cfm)
    ⇒ Directory Names - Querying for common server directories (servlet, pls, SilverStream)
    ⇒ Session Tokens - Presence of propriety session tokens

  → Identifying Server Functionality
    ⇒ Dissecting Requests
    ⇒ Extrapolating Application Behaviour
    ⇒ Isolating Unique Application Behaviour

  → Mapping the Attack Surface