eJPT Certification

Lab: Burp Suite Basics

09/1/2020

Learning Objectives:

* Use Burp Suite intruder.
* Use Burp Suite repeater.
* Learn how to find hidden resources on web application.

Tools:

* Kali Linux.
* Burp Suite Community Edition.
* Web Browser.

Notes:

1. Setup OpenVPN tunnel
   1. Start vm lab
   2. Download opvn configuration file
   3. run VPN
2. Setup connection to web application
   1. Target application: <http://172.16.160.102>
   2. A screenshot of a social media post

      Description automatically generatedScreen shot of target site. Message in DOM.
3. Create burp suite project and set up proxy intercept between burp and web application.
4. Conduct web analysis in burp.
   1. Refresh web application to trigger proxy notication in burp. Forward GET request to repeater so that we can analyze the server response.

* A screenshot of a social media post

  Description automatically generatedif we look closely in the <html> body we can see there is a note to remove the robots.txt file when the developers are done.
  + This file is an internet exposed file that that contains instructions for automated web browsing tools called “web crawlers”.
    - Web crawlers scan the whole internet to find up-to date search results.

1. Added robots.txt to URL path to see if that leads us to anymore valuable information.
   1. Provides us with a small text file full of paths.
      * User-agent: \*
      * Disallow: /cgi-bin/
      * Disallow: /includes/
      * Disallow: /images/
      * Disallow: /scripts/
      * Disallow: /\*?debug=\*
      * Disallow: /connections/
      * Disallow: /backup/
      * Disallow: /settings/
2. We can now test these paths with an automation tool from burp called “intruder”
   1. By specifying an attack target, we can configure the attack details. In this particular case I want to test all the paths that we have discovered from robots.txt.
   2. We can now create a payload file to upload to intruder, which contains our paths, and then configure a GET path so that a get path will run for each path listed in the file.
   3. In return we receiver a list of responses from the server letting us know if any of our requests were good.
3. What we will find ultimately after a little bit of debugging is username and password for database:127.0.0.1
4. Once we analyze our responses, we can analyze which paths came back positive. In this case only one path is good. We can now add that path back into our URL path and see where it takes us.