Spidering Cheat Sheet

by binca via cheatography.com/44948/cs/13321/

Definition

Also known as "crawling"

Involves following web links to download a copy of an entire site.

Analyze offline to discover: potential security weaknesses in code, list of keywords for password-guessing, confidential data, and names, emails, addresses, and phone numbers.

May spider a site many times w/various tools. Scanning primes many tools.

Manual spidering is done by browsing a site and saving each page.

Automated scans are common but if the site is too complex it may fail.

Robot Control

Automated tools are often referred to as robots o bots.

Developers control bots by using a robots.txt file either placed in the root directory of the web app or using meta tags on individual pages.

Robots Exclusion Protocol is an unofficial commonly implemented protocol that uses robots.txt to specify which user-agent types should be disallowed access to certain directories and individual pages.

Tags that prevent page caching:
<META HTTP-EQUIV="PRAGMA" CONTENT="NO-CACHE">
<META HTTP-EQUIV="CACHE CONTROL" CONTENT="NO-CACHE">

Tags that control:
<META NAME="ROBOTS" CONTENT="INDEX, NOFOLLOW">
<META NAME="GOOGLEBOT" CONTENT="NOARCHIVE">

Automated Spidering with ZAP

ZAP interception proxy includes spidering capabilities

Primed by using interception proxy to navigate to the site

Does well but dynamically generated links on client can cause it to miss pages

Has separate AJAX spider for dynamic sites

Will show out of scope targets

Wappalyzer

Provides detailed understanding of technologies running on a web app including OS, web servers, packaged web apps, languages, frameworks, and APIs

ZAP + Wappalyzer

Leverages Wappalyzer functionality through the use of Technology Detection extension in ZAP marketplace

The extension is passive

Only implements some functionality

Not release quality

ZAP Forced Browse

Based on inactive DirBuster Project, a dictionary attack

Seeks to find unlinked content using a number of default wordlists or one provided by user

Forced Browse = entire site

Forced Browse Directory = focuses on 1 directory

Forced Browse Directory + Children = recursive forced browse against a directory and any discovered sub-directories

Automated Spidering with Burp

Similar to ZAP and Paros

Automated Spidering with wget

Console-based web browser that runs on most platforms and has basic spidering capabilities saves retrieved items in a directory

Obeys robot.txt unless invoked with the **-e robots=off option**

-r invokes recursion of discovered links

-l [N] specifies max link recursion where N = #, default is set to 5

It can retrieve via HTTP, HTTPS, and FTP

Popularity due to script inclusion

Syntax: **wget -r [domain] -l 3 -P /tmp**

CeWL

Custom word list generator.

Spiders a website generating a word list from the EXIF data of any images and the site contents

Syntax: **./cewl.rb [domain]**

By **binca**
cheatography.com/binca/

Not published yet.
Last updated 9th November, 2017.

Analyzing Results: What to Look For

(HTML) Comments that reveal sensitive or useful information

Commented code and links

Disabled functionality

Linked servers such as content and app servers

HTML Comments

Contents in the HTML that are included in the server response to the client such as dev notes, explanations of functionality and variables, usernames and passwords. Should move comments to server-side.

Disabled Functionality

Reveals previous or future sections of site.

In the case of "disabled" functionality may be able to still invoke or it may indicate security weaknesses.

Examples:
Links that have been commented out
Client-side code that has been commented out, which may be server-side code now