### **Web scrape**

vim web\_scrape.py

import lxml.html

import requests

def main():

page = requests.get(‘<http://quotes.toscrape.com>’)

tree = lxml.html.fromstring(page.content)

authors = tree.xpath(‘//small[@class=”author”]/text()’)

Nested

<small class=”author” itemprop=”author”><strong>Albert Einstein</strong></small>

def main():

authors = tree.xpath(‘//small[@class=”author”]/strong()’)

### **Host Discovery**

(Determine if hosts exists on the network using **quick** port agnostic scans

**Ping Sweep** Sends one icmp echo request packet to each host on the 192.168.1.0/24

* Linux: for i in {1..254} ;do (ping -c 1 192.168.1.$i | grep "bytes from" &) ;done
* Windows: for /L %i in (1,1,255) do @ping -n 1 -w 200 192.168.1.%i > nul && echo 192.168.1.%i is up.

Host Discovery (with ping sweep)

for i in {1..31} ; do (ping -c 1 192.168.28.$i | grep “bytes from” &) ;done

### **Port Enumeration**

(figure out what ports are open)

proxychains nmap -PN -sT 192.168.65.10

**Big scan**

nmap -PN -sT 192.168.28.96 -p-

### **Port Interrogation**

(interact with discovered hosts and ports to determine the best way to leverage each available service)

Use nc to interrogate a web server:

* nc -Cv 127.0.0.1 80
* Type: GET / HTTP/1.0 to get a HTTP Response header from the server.

Use nmap to perform service detection on port 22 of your opstation:

* nmap -sV 127.0.0.1 -p 22

Using nikto to perform a vulnerability scan on your opstation:

* nikto -h 127.0.0.1 -p 80
  + Also shows other information like what HTTP methods are allowed and various CVE vulnerabilities.

### **Advanced Network Scanning**

/usr/share/nmap/scripts

ls | grep smb

nmap --script-help

nmap - -script-help http-enum.nse

nmap -Pn 192.168.65.10 -p 80 - -script http-enum

/templates/: Potentially interesting folder

nmap --script <filename>|<category>|<directory>

nmap --script-help "ftp-\* and discovery"

nmap --script-args <args>

nmap --script-args-file <filename>

nmap --script-help <filename>|<category>|<directory>

nmap --script-trace

### **Reconnaissance PE**

lin-ops:~$ ssh student@10.50.35.200 -L 60601:localhost:22

host discover

jump:~$ for i in {96..127}; do (ping -c 1 192.168.28.$i | grep ‘bytes from’ &); done

192.168.28.97

192.168.28.98

192.168.28.99

192.168.28.100

192.168.28.105

192.168.28.111

192.168.28.120

lin-ops:~$ ssh student@localhost -p 60601 -D 9050 -NT

lin-ops:~$ firefox > settings > network settings > manual proxy > socks host: 127.0.0.1 port 9050 > ok