**Seth Ayers**

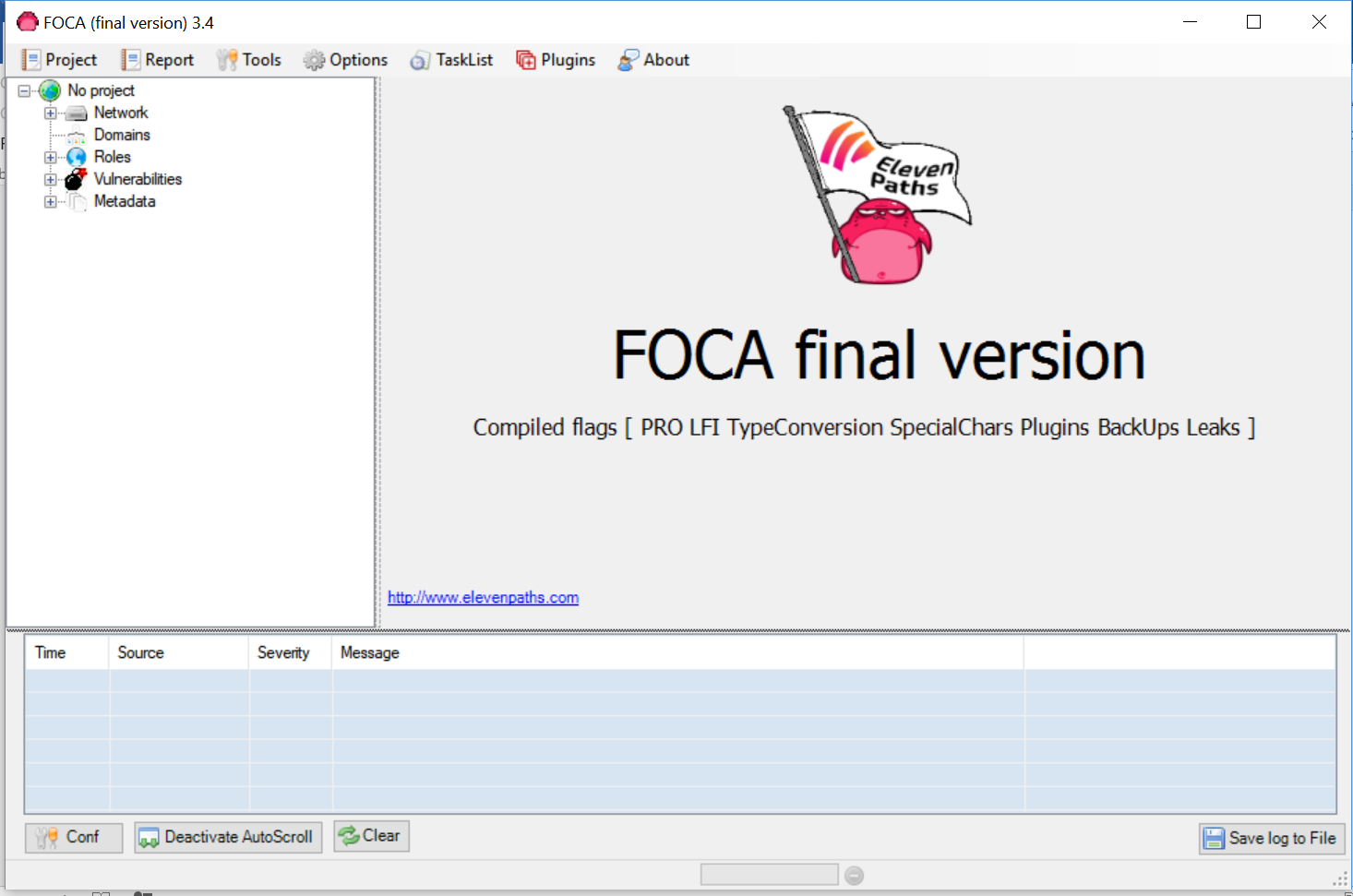
**02/02/2018**

**IT 430 – 02**

**Lab 2**

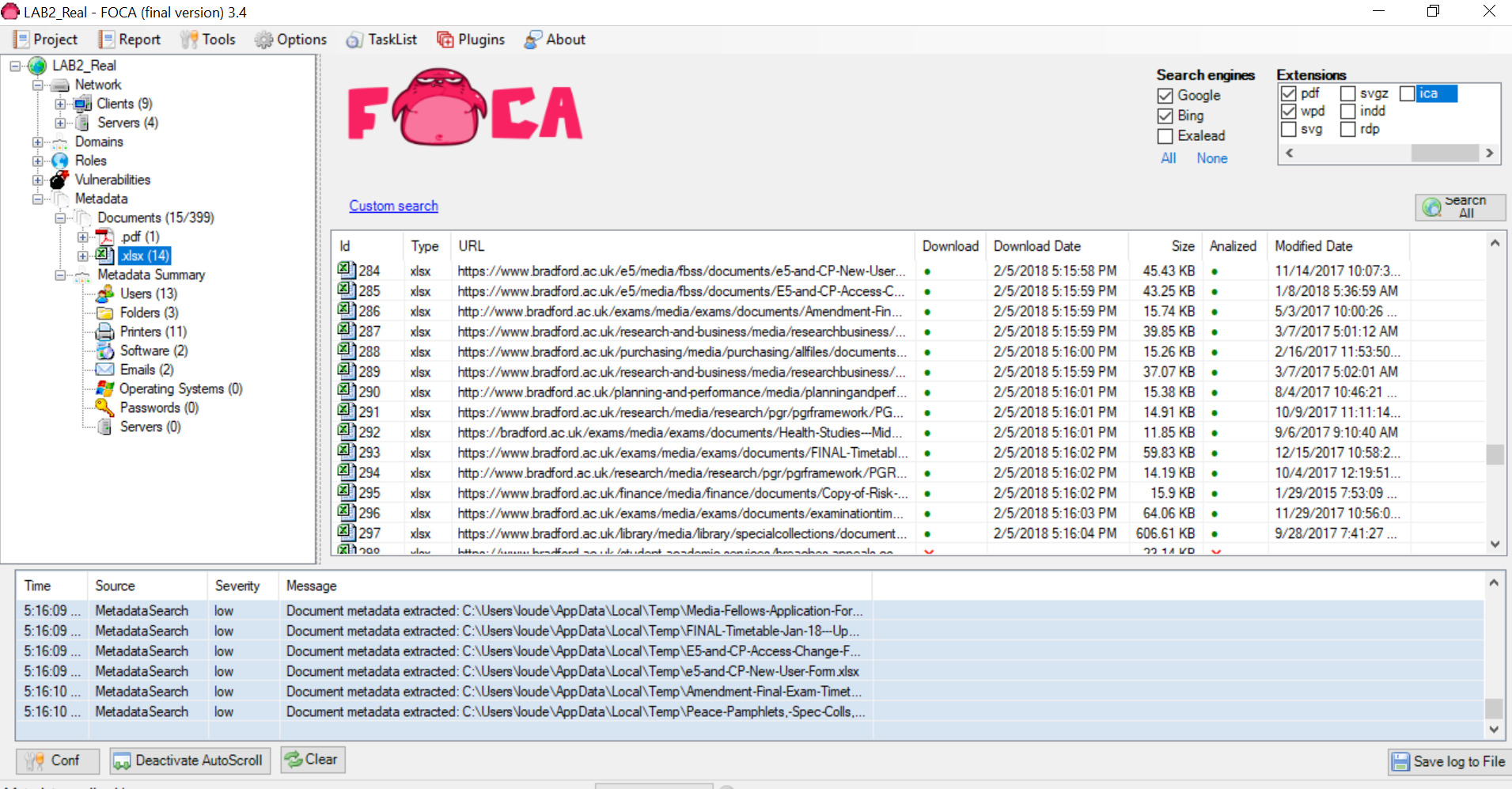
**SAyers\_Lab2.docx**

**Part 1 (FOCA):**



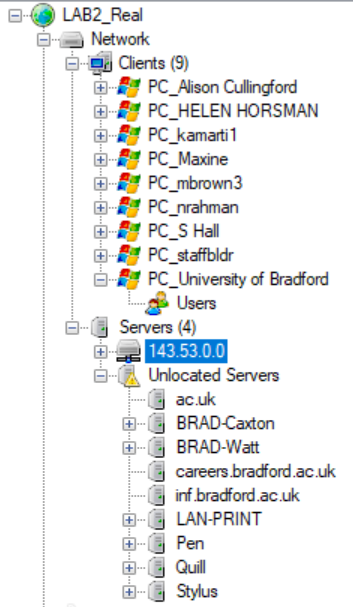
**Observation:**

This is the installed FOCA program on my workstation.



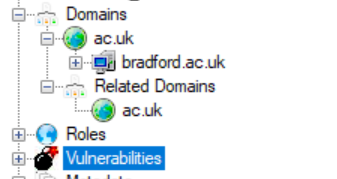
**Observation:**

These are the scan results of the website bradford.ac.uk. I downloaded a mix of pdf, doc, and xlsx file types. In this, the data was downloaded, extracted, and analyzed.

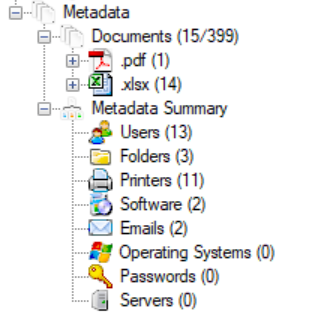


**Observation:**

In this section, data that was scanned was client and server information retrived by the meta data extraction. This can be used to provide a rough idea of a networks topography, with server, client, user, and even printer information shown. If looking for access to a specific domain, you can check the user who accessed the server or printer last, get their information. You then will be able to access the sever/printer/client with the right credentials.

  
**Observations:**

This section of FOCA displays roles unique to users or services unique to servers, vulnerabilities, and Domain information. The roles subtree includes DNS, IPS, Proxy, Remote Desktop, Kerberos, ect. The Domain subtree finds the main domains along with various A record information. This information also provides server software suites, roles description, and IP (reverse lookup) information. A hacker could use this information to search for known velernabilities that match the server OS versions, finding SQL injection possibilities, using user credentials to manipulate roles (or gaining access to roles.), etc.



**Observation:**

In my opinion, the Metadata section is the most useful feature in FOCA. It takes extracted data form the scanned site and organizes it into easy to read categories. The information gather is, but not limited to: Users, specific file’s (format at the user’s preference), folders and their locations on the server, email addresses, printers and their locations, passwords, etc. The Metadata feature omits the need of sifting through gigabytes of scanned and retrieved information, making the hacking of the system easier before admins can make the proper changes to defend the domain/network.

**Part II (Netcraft):**

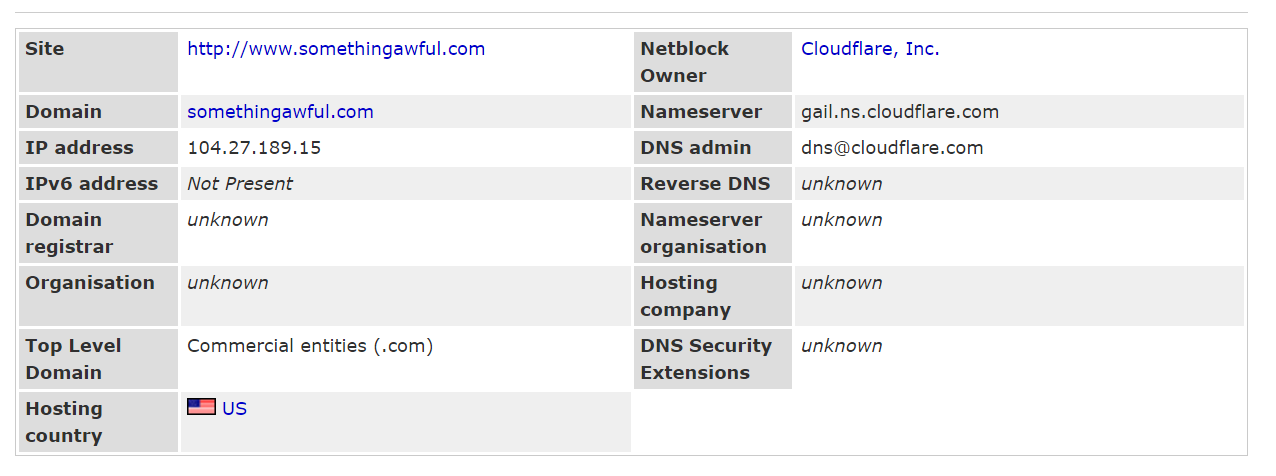
**Exercise 4.3:**

Netcraft is a passive scanning tool for user selected domains that can obtain running OS and host information, additional IP information, server data, and DNS information. For my example, I used somethingawful.com, an old comedy website with forum access. One important positive in using Netcraft is that it provides a scan of a certain site that cannot be tied back to the user requesting the scan.



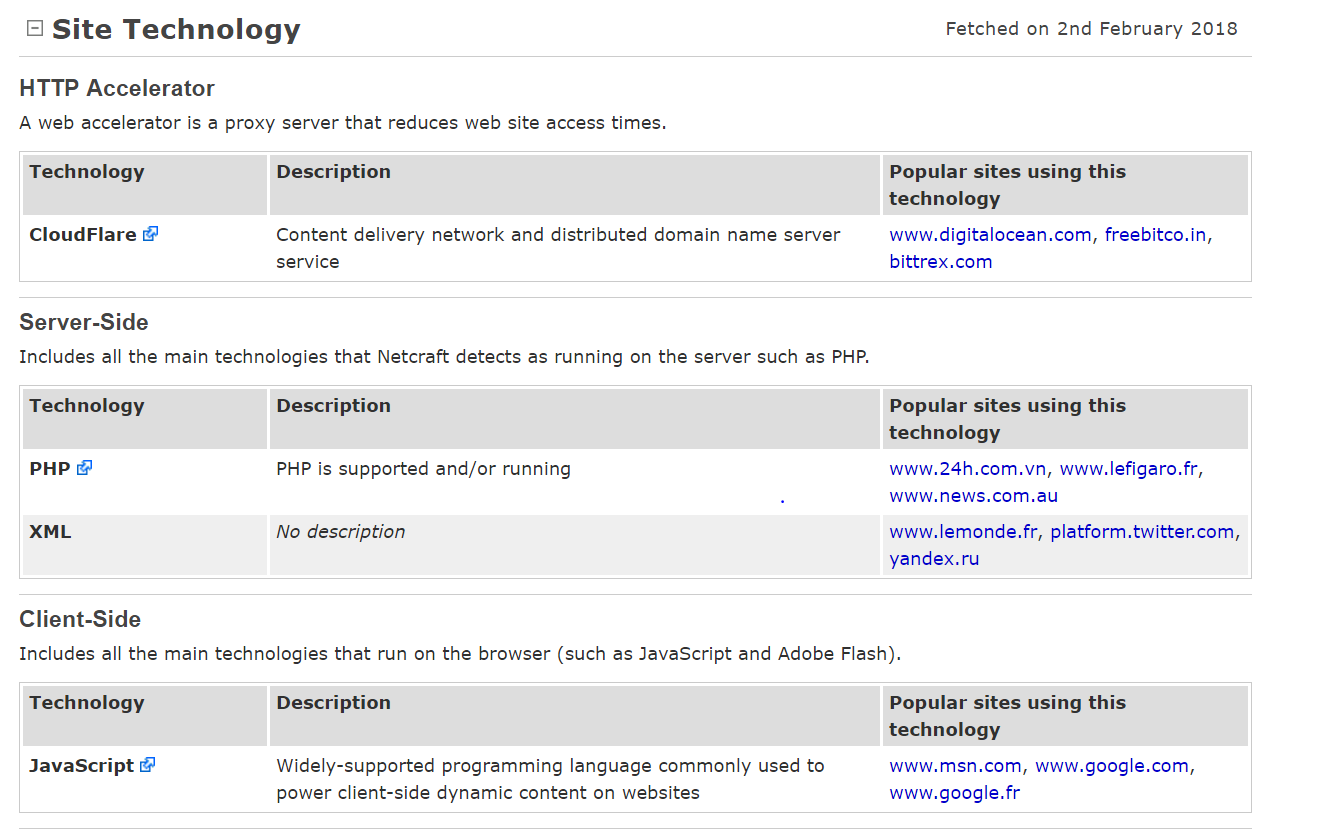
**Observation:**

Background information on the site, including its creation, title, and short description.



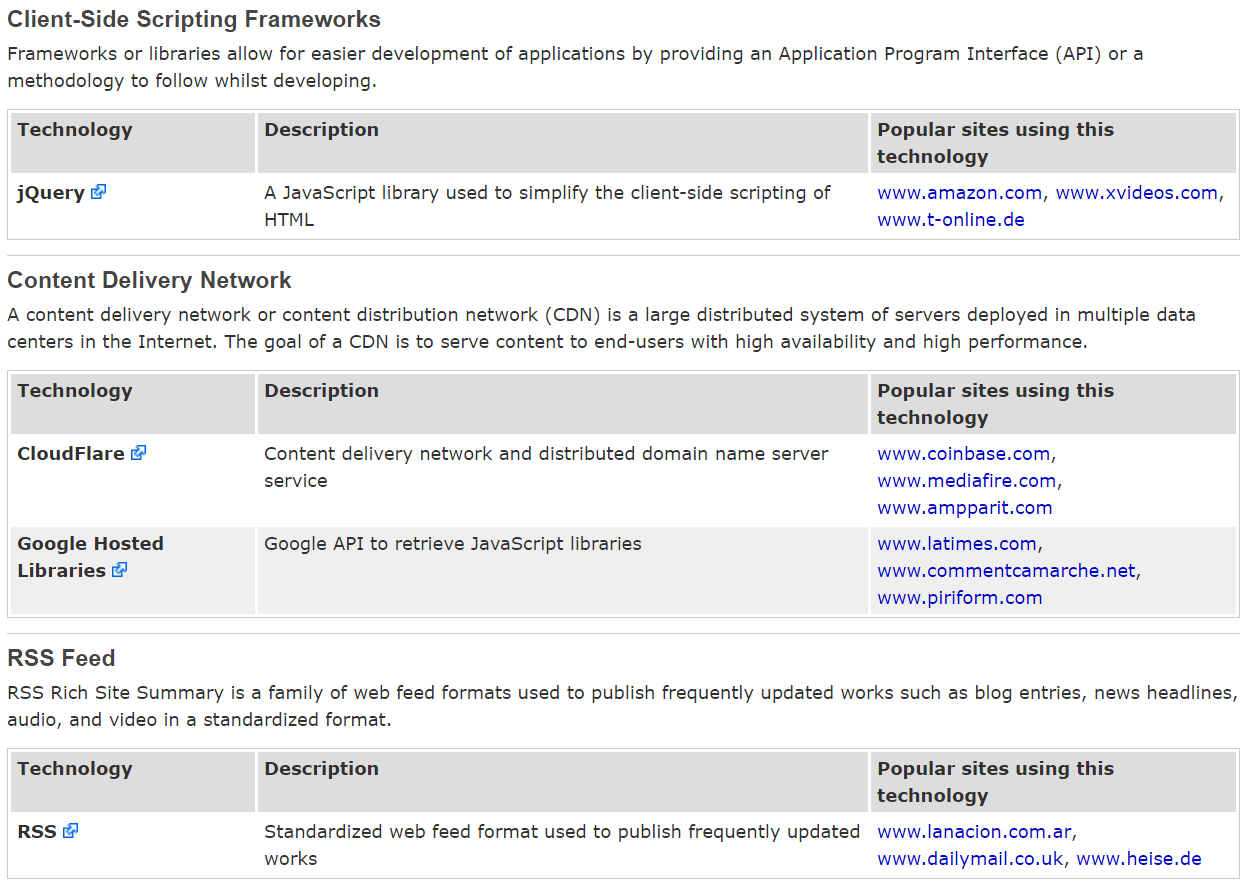
**Observation:**

Shows the netblock server owner, it’s nameserver, and its DNS server. It also shows the IP address (resolved from the DNS), as well as the native location of the host. Overall general information.



**Observation:**

The Site Technology section can provide proxy server information, sever side technologies that are running, and client technology that run on a browser. This information can be used to compile a list of known vulnerabilities, depending on software versions. This would require a more in-depth, aggressive scan, but is a good starting point.

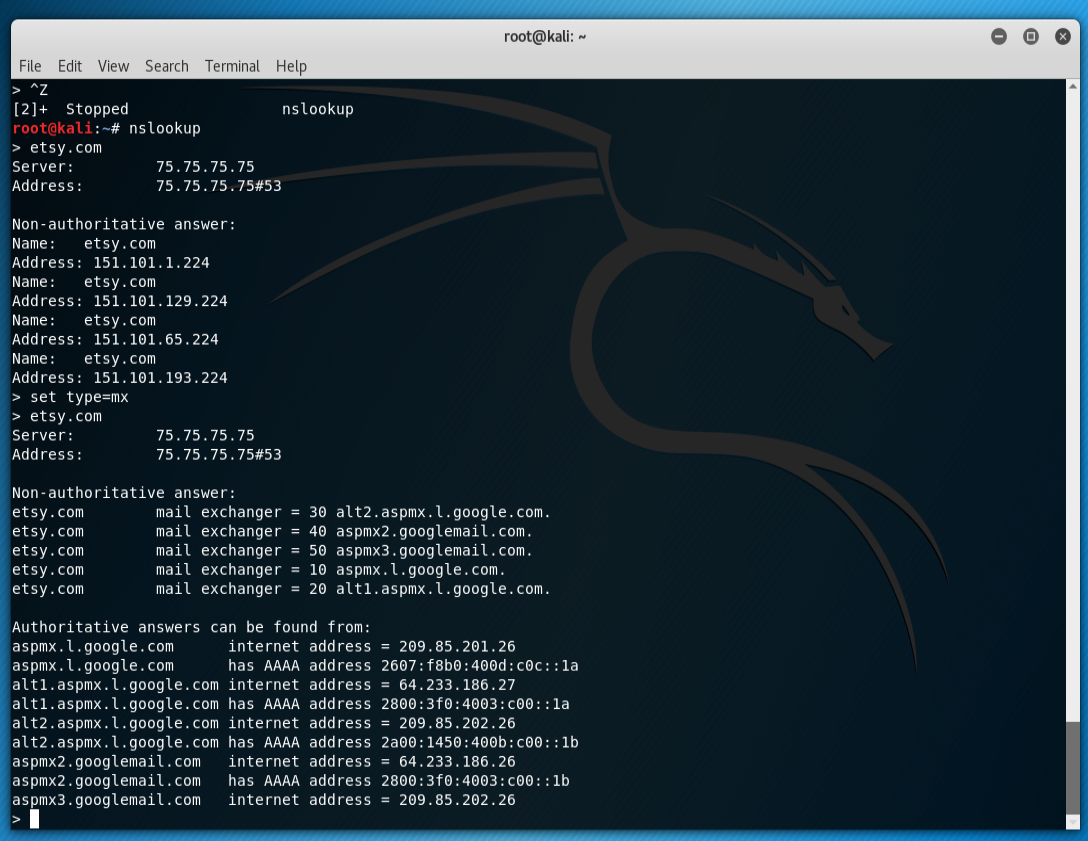


**Observation:**

In addition to the proxy and server/client technologies, the Site Technology section shows scanned information on scripting frameworks, API’s, content delivery networks, and standardized update via RSS.

**Part III (NSlookup):**

**Exercise 4.5:**



**Observation:**

The nslookup command provides information about non-authoritative, authoritative, and Mail exchanger DNS servers w/ records. Also displayed are the fully qualified domain names associated with the inputted address. IPv4 and AAAA (IPv6) are displayed for the mail servers as well.

**Part IV NMAP:**

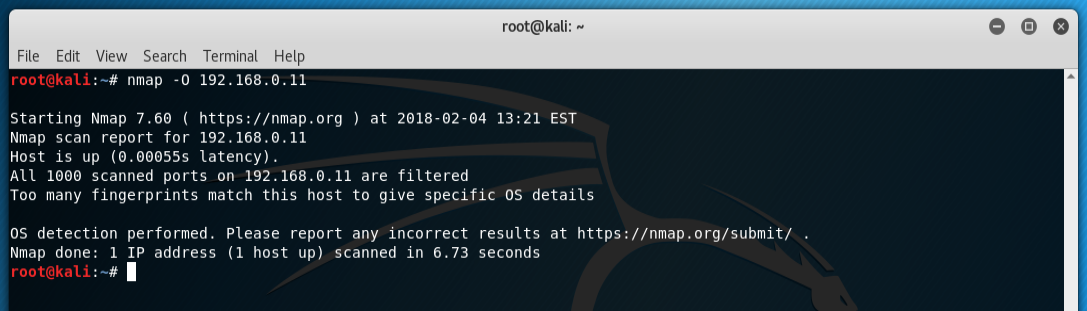
For this section, I will be scanning my own network, specifically my home desktop, using various nmap commands. I will preform these scans in a white box fashion, tailoring my selections to commands that I personally know will possibly bypass my firewalls.

**Command 1:**

Nmap -O XXX.XXX.XXX.XXX

Used to find a hosts operating system.

**Result:**



**Observation:**

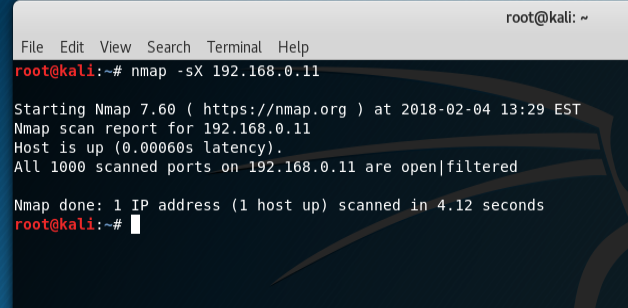
Attempted OS scan of host desktop. Results were unsuccessful as all ports were filtered which resulted in a universal fingerprint match. Therefor the nmap function was unable to give details of the OS. Good news for my system. However, the nmap did give the indication that the host was “up” or running, information which can be used at another time.

**Command 2:**

Nmap -sX XXX.XXX.XXX.XXX

Scans a firewall to find a security venerability.

**Result:**



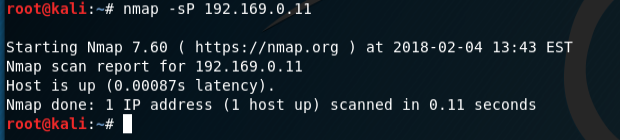
**Observation:**

Performed a Xmas tree scan to the host. The firewall filtered and dropped all requests. Again, the host is shown as up.

**Command 3:**

Nmap -sP XXX.XXX.XXX.XXX

Scans a network to check to see what servers/devices are up and running



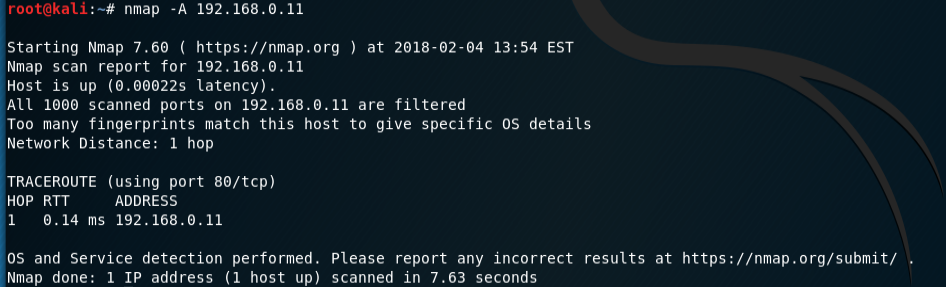
**Observation:**

Also known as a host discovery or ping scan, this was successful in identifying that my desktop was up and functional. As I understand it, this would be one of the first steps in identifying valid targets and commencing advanced scans.

**Command 4:**

Nmap -A XXX.XXX.XXX.XXX

Scans OS and service versions.



**Observation:**

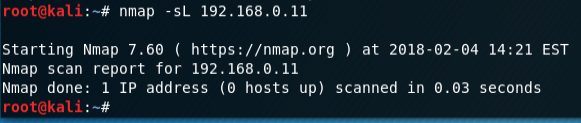
This can be considered the “beefed up” version of the OS scan above. If successful, it will not only tell you more detailed information of the OS, but the hops it took to get there as well.

**Command 5:**

Nmap -sL XXX.XXX.XXX.XXX

Commands host names to the scan.

**Result:**



**Observation:**

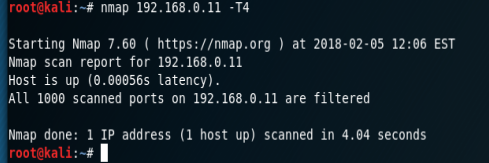
This is supposed to attach a host name to the standard scan. Interestingly, although my focus host is still on, the scan came up with the host being offline. I’m not sure if my desktop firewall started to black communication from my laptop. I’ll have to investigate further.

**Command 6:**

Nmap XXX.XXX.XXX.XXX -T4

Commits an extremely fast, aggressive nmap scan.

**Result:**



**Observation:**

This aggressively and quickly scanned the target. This can be used for quick scans of targets, however, it is highly detectable by firewalls and anti-intrusion systems. Used in the “quantity over quality” approach where detection isn’t a factor.

**Command 7:**

(I decided to do preform two commands to show the versatility of the -p switch)

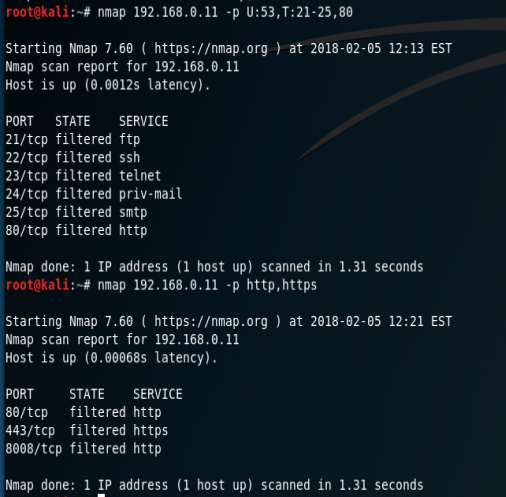
Nmap XXX.XXX.XXX.XXX -p U:XX,T:XX

Scans both UDP and TCP ports with specified parameters

Nmap XXX.XXX.XXX.XXX -p http,https

Scans ports for specified services

**Result:**



**Observations:**

These scans allow for more detailed, refined scans on specific ports or services. This is incredibly helpful in determining what ports are open to what traffic, and what services are using the user provided parameters.