Technical Assignment #1

CS 6573 – Penetration Testing and Vulnerability Analysis

# Rules:

Submit by the deadline listed on the assignment page. After this time, your homework will not be accepted. Keep your answers as short as possible. This assignment is based on Modules 1-4. Do cite your sources if you use other than the lecture slides, including templates. You may not work with anyone else. Both of these are considered cheating. Cheating will result with at a minimum: a zero for this assignment and could result in expulsion from NYU.

[100 pts] All questions in this section are regarding Indeed.

You are looking for public vulnerabilities for a website which participates in a bug bounty program.

Target: Indeed (https://www.indeed.com/)

Permission: https://bugcrowd.com/indeed

**=========== STOP AND READ PERMISSION DETAILS BEFORE CONTINUING ==============**

Scope: All authorized URLS that end in .com. No need to enumerate TLDs other than .com (e.g. \*.indeed.co.in). See permission page on Bugcrowd for specifics.

Perform recon to figure out the following details. If you use a tool, explain commands, arguments, and modules. Do not use any intrusive, aggressive, overly active, load-testing, or DoS-risking techniques. Do not use any automated vulnerability port-scanning tools or exploitation frameworks (e.g. Metasploit, Nessus, Retina, Qualys, Nexpose, OpenVAS). Recon-ng, Amass are okay.

1. [10 pts] What subdomain is out of scope regarding open redirects?

indeed.co.uk

1. [15 pts] Using passive methods only, what subdomains can you find for indeed.com?

*25 Subdomains listed here, see more in the attached spreadsheet. Please note, as many of the discovered subdomains came from the internet archive, not all may be valid.*

|  |  |  |
| --- | --- | --- |
| **Subdomains** | | |
| br.indeed.com | go.indeed.com | ph.indeed.com |
| ca.indeed.com | in.indeed.com | pt.indeed.com |
| employers.indeed.com | malaysia.indeed.com | support.indeed.com |
| es.indeed.com | mx.indeed.com | uk.indeed.com |
| fr.indeed.com | myjobs.indeed.com | www.indeed.com |
| cts.indeed.com | apply.indeed.com | paynow.indeed.com |
| engage.indeed.com | partnerships.indeed.com | resumebuilder.indeed.com |
| hire.indeed.com | partners.indeed.com | wiki.indeed.com |
| thevirus.aggtest.indeed.com | | |

1. [15 pts] From those subdomains, what unique IPs can you find?

*Must have at least 25 IP addresses*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Subdomains** | **IP Addresses** | | | |
| br.indeed.com | 162.159.129.67 | 162.159.130.67 | |  |
| ca.indeed.com | 162.159.129.67 | 162.159.130.67 | |  |
| employers.indeed.com | 18.116.0.135 | 3.141.60.172 | 3.131.4.75 | |
| es.indeed.com | 162.159.129.67 | 162.159.130.67 | |  |
| fr.indeed.com | 162.159.129.67 | 162.159.130.67 | |  |
| go.indeed.com | 3.130.123.174 | 3.131.108.23 | 18.189.52.126 | 3.19.195.73 |
| in.indeed.com | 162.159.129.67 | 162.159.130.67 | |  |
| malaysia.indeed.com | 162.159.129.67 | 162.159.130.67 | |  |
| mx.indeed.com | 162.159.129.67 | 162.159.130.67 | |  |
| myjobs.indeed.com | 3.128.222.240 | 18.218.184.174 | 18.119.36.85 | |
| ph.indeed.com | 162.159.129.67 | 162.159.130.67 | |  |
| pt.indeed.com | 162.159.129.67 | 162.159.130.67 | |  |
| support.indeed.com | 104.16.51.111 | 104.16.53.111 | |  |
| uk.indeed.com | 162.159.129.67 | 162.159.130.67 | |  |
| www.indeed.com | 162.159.129.67 | 162.159.130.67 | |  |
| cts.indeed.com | 3.130.25.196 | |  |  |
| engage.indeed.com | 18.155.181.43 | |  |  |
| hire.indeed.com | 108.156.211.94 | |  |  |
| apply.indeed.com | 162.159.130.67 | |  |  |
| partnerships.indeed.com | 18.119.36.85 | |  |  |
| partners.indeed.com | 3.19.195.73 | |  |  |
| paynow.indeed.com | 3.19.195.73 | |  |  |
| resumebuilder.indeed.com | 3.19.195.73 | |  |  |
| wiki.indeed.com | 76.77.155.130 | |  |  |
| thevirus.aggtest.indeed.com | 198.58.75.28 | |  |  |

1. [15 pts] What netblocks are granted to or associated with the company, and who owns them?

*Must have at least 4 netblocks*

|  |  |
| --- | --- |
| **Netblock** | **Owner** |
| 162.158.0.0/15 | Cloud Flare |
| 18.32.0.0 - 18.255.255.255 | Amazon |
| 3.128.0.0/9 | Amazon |
| 104.16.0.0/12 | Cloud Flare |
| 76.77.144.0/20 | Cyrus One |
| 198.58.72.0/21 | Cyrus One |
| 108.156.0.0/14 | Amazon |

1. [20 pts] What employee email addresses can you find?

*Must have at least 6 email addresses*

|  |  |  |
| --- | --- | --- |
| **First** | **Last** | **Email** |
| Joe | Esposito | joe@indeed.com |
| Kelsey | Byrne | kbyrne@indeed.com |
| Shannon | Rubes | rshannon@indeed.com |
| Shannon | Rubes | srubes@indeed.com |
| Vince | Gambo | vgambo@indeed.com |
| Will | Balistrieri | wbalistrieri@indeed.com |

1. [25 pts] Using a non-aggressive method we covered in class, such as Google dorking, polite recon-ng modules, or Eyewitness, find at least one endpoint, service, or exposure which could be used for future research or testing. For example, an API that doesn’t require a key/token, an interesting file, error page, service, etc. It does not have to be a proven vulnerability, just something that should be researched more as we enumerate the attack surface. Provide the method and the findings.

Note: If you find something that you think could lead to a bounty, remember that this is an open and active paying program. Anything you submit will be visible to the TA(s) and the professor. If you discover anything you would like to research more or that you think may lead to a bounty, I encourage you to continue your recon and enumeration, and find something else to submit for this answer that you are okay with sharing.

After grepping through my GAU output, I discovered an interesting subdomain at autocomplete.indeed.com.

A screen shot of a computer screen

Description automatically generated with low confidence

The majority of saved URLs seemed to be user searches of some sort. I used GoWitness to screenshot many of the URLs hoping that I could see some sensitive information. Unfortunately, the returned screenshots were all blank.

A screenshot of a computer screen

Description automatically generated with low confidence

The root directory of the web application responds to a simple HTTP GET request with “ok.”

A picture containing screenshot, white, rectangle

Description automatically generated

In an attempt to get more information, I requested the /log directory on the site. The application responded with a 405 Error request and a message that Get requests were not supported.

A picture containing text, font, screenshot, white

Description automatically generated

Because I found this site through archive.org and many of the stored URLs are not responding, it seems like this API stored user information, but is no longer in use. I believe that more in-depth fuzzing could reveal sensitive information stored in the API.

Please see the following documentation for a walkthrough of this investigation.

# Subdomain Enumeration

I began by using GAU (Get All URLs). GAU is an open-source tool written in Go that queries archive.org for archived URLs for a domain and it’s subdomains.

A picture containing text, font, screenshot, graphics

Description automatically generated

After getting a whopping 172,223,419 results, I combed through them and used cut to pull out the individual subdomains.

A picture containing text, screenshot, font

Description automatically generated

During this time, I also used TheHarvester to pull additional information about indeed.com subdomains and ip addresses.

A screenshot of a computer

Description automatically generated

# Netblock Enumeration

After identifying 25 valid-looking subdomains, I performed dns lookups to get the IP addresses associated with each subdomain. From there, I used whois to get information about the Ip address owners and the netblocks owned by each.

A screenshot of a computer

Description automatically generated

# Employee Email Enumeration

I initially intended to use TheHarvester to pull employee email addresses for indeed.com. However, recent updates to Google’s captcha program mean that TheHarvester is now blocked from retrieving information through those sources. Instead, I used the whois\_pcos module from recon-ng.

A screenshot of a computer

Description automatically generated