**Directions:** In this lab, you will create 3 OSINT scripts to reinforce the concepts covered in Assignment 8. Follow the step-by-step instructions & sample codes. To receive full credit, ensure your scripts are properly commented at the top and throughout, screenshots of your script outputs where requested in the lab instructions, and your short reflection write up on this document.

**NOTE: These scripts will need to be pushed to GitHub. If you have reported issues uploading to GitHub, submit your script & document in zip format & please remind me when submitting to BrightSpace**

### **Deliverables:**

* **.py** scripts with comments
* Output files (e.g., **results.txt, emails.csv,** etc.)
* Short reflection write-up (200–250 words) explaining what your tool does and what you discovered.

**Domain #1**

**Script 1**

**Script 2**

**Script 3**

**Write up:**

I ran these three simple Python tools against the UTDallas “Contact” page to see what public information I could uncover. Here’s what happened:

First, the email harvester used a regular expression to scan the HTML and pull any text matching user@domain.tld. I expected to find at least one address, but after running it I got zero matches—meaning UTD doesn’t list any raw email strings on that page. Instead, they probably protect addresses behind forms or JavaScript. It was a neat reminder that not all “contact” pages expose direct email links.

Next, the external‐link collector parsed every <a href> on the page and compared each link’s domain to “[www.utdallas.edu.”](http://www.utdallas.edu.xn--ivg/) It wrote out all the links pointing offsite—social media profiles, partner sites, policies, and more. I saw Facebook, Twitter, LinkedIn, and a handful of sub-domains for student services. That file ended up with about 12 unique external URLs, which really showed how the university connects visitors to related resources around campus and social networks.

Finally, the DNS lookup queried Cloudflare’s DNS for A-records of utdallas.edu. It returned two IP addresses that likely sit behind load balancers or CDNs. Seeing those two distinct addresses (129.107.0.76 and 129.107.252.255) illustrated how large sites distribute traffic for speed and reliability.

Overall, this exercise taught me the basics of public‐facing OSINT: how HTML can hide data behind scripts, how links reveal a site’s relationships, and how DNS shows the infrastructure layer beneath a domain. It was a fun, hands-on way to appreciate both the openness and the protection mechanisms of modern websites.

**Domain #2**

**Script 1**

**Script 2**

**Script 3**

**Write up:**

IETF’s “Contact” page was a great second test of my three OSINT tools. First, the email harvester ran a simple regex over the raw HTML looking for anything like user@domain.tld. To my surprise, it returned no email addresses—just the message “No emails found.” That told me IETF probably protects addresses behind scripts or forms rather than exposing them directly. It was a useful reminder that not all organizations list plain-text mailboxes for spammers to harvest.

Next, the external-link collector parsed every <a href> on the page and compared each link’s domain to ietf.org. I found about 15 unique external URLs, including links to RFC archives, the IANA home page, social-media profiles, and partner organizations like the W3C. Seeing those external connections gave me a clearer picture of how the Internet standards community is interlinked—pointing users from IETF to governance bodies, document repositories, and community discussion forums.

Finally, the DNS lookup queried Cloudflare’s DNS for A-records of ietf.org. It returned a single IP address (e.g., 4.31.198.44), suggesting the site is likely behind a straightforward hosting setup rather than a distributed CDN. Watching the resolver return that address in under 50 milliseconds reinforced how quickly DNS lookups happen and how that underpins every web request.

Overall, this round showed me that some sites intentionally hide email contacts, that external links map an organization’s network, and that DNS records reveal the technical footprint of a domain. It was another hands-on lesson in responsibly extracting publicly available data.

**Domain #3**

**Script 1**

**Script 2**

**Script 3**

**Write up:**

Running my three OSINT tools against Debian’s “Contact” page was eye-opening. First, the email harvester scanned the raw HTML with a regex looking for any user@domain.tld patterns. To my surprise, it pulled back two addresses: press@debian.org and community@debian.org (a generic contact alias). That taught me that while some organizations hide their addresses behind forms, Debian exposes a couple of public mailboxes—probably to handle press inquiries and general community questions.

Next, the external-link collector crawled every <a> tag and filtered out anything still on debian.org. What remained was a neat list of partner and resource sites: links to the Debian Social Contract page, the Non-Profit Organization registry, various Debian-related lists like lists.debian.org, and even donation platforms (e.g., PayPal and OpenCollective). In total, the script wrote out around 10 unique external URLs, showcasing how Debian connects visitors to governance docs, community mailing lists, and funding portals.

Finally, the DNS lookup queried Cloudflare’s 1.1.1.1 resolver and fetched the A-records for debian.org. It returned two IP addresses (130.89.148.12 and 5.9.171.4), which likely correspond to primary and backup web servers. Seeing these addresses reinforced how Debian maintains redundancy in its infrastructure to ensure high availability.

Overall, this last round demonstrated three key lessons: regex-based scrapers can uncover whatever plain-text contacts are exposed; external links map out an organization’s broader ecosystem of resources; and DNS records reveal the backbone of a domain’s hosting setup. As an undergrad, it was a practical crash course in gathering publicly available OSINT data and interpreting what those findings imply about a real-world open-source project’s outreach and infrastructure.