EC521 - Cybersecurity Final Project: Open Source INTelligence (OSINT) Lab Generation

Maha Ashour, Jack Belmont, Pierre-François Wolfe

Boston University, College of Engineering

BOSTON UNIVERSITY

Goals

Lab generation project

- Open Source INTelligence (OSINT) for identifying vulnerabilities
- Introduce useful tools
- Provide an interesting and ethical learning scenario
- Build lab infrastructure: vulnerable webserver, synthetic datasets, challenges, etc.

OSINT Cycle

- Collect (gathering)
- Process (data validation)
- Exploit (determine value of information)
- Produce (consumable/usable data)

Maltego

- Data Mining Tool
- Direct Graphs for Link Analysis
- Relationship Analysis
- Transforms for streamlined searching

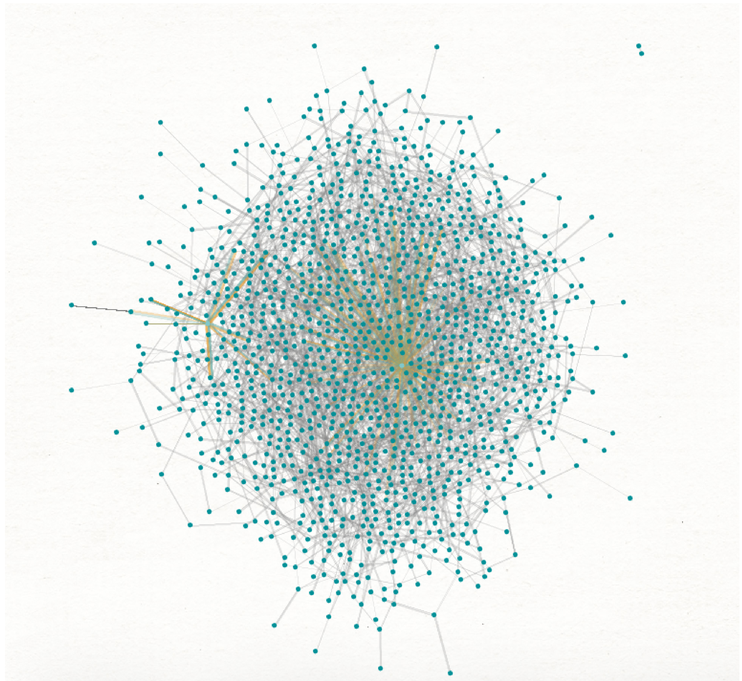
Other Tools

- **theHarvester**
  - Input: Web domain and data source (ex: bing, linkedin, shodan)
  - Output: Report with emails, subdomains, hosts, employee names, open ports, etc.
  - Method: Gathers public information about domain using custom python scripts for each data source
- **h8mail**
  - Input: [List of] email(s)
  - Output: Passwords - hashed or plain text
  - Method: Looks through "breach compilation" to find match
- **Metagoofil**
  - Input: Web domain
  - Output: Report with usernames, software/server versions, other info
  - Method: Uses Google search to gather document metadata from a given domain

Maltego: Custom Transforms

```python
    @classmethod
    def create_entities(cls, request, response):
        domain = request.Value
        harvester = 'theharvester'

        try:
            subprocess.call(['theharvester'],stdout=subprocess.DEVNULL, stderr=subprocess.DEVNULL)
        except OSError as e:
            harvester = 'theHarvester'

        try:
            sources = ['google', 'bing']
            f = open('./log.txt', 'w')
            for source in sources:
                getResults = subprocess.Popen([harvester,'-d',domain,
                    '-b',source],stdout=f,stderr=f)
                rc = getResults.wait()
                if rc == 0:
                    return_code = 0
            f.close()

            hits = 0
            if return_code == 0:
                f = open('./log.txt', 'r')
                for line in f:
                    matches  = re.findall(r'[\w\.-]+@[\w\.-]+', line)
                    for match in matches:
                        if match != "cmartorella@edge-security.com":
                            hits += 1
                            response.addEntity(Email, match)

            if hits == 0:
                response.addUIMessage('No emails found :(')

        except IOError:
            response.addUIMessage("An error occured during IO", messageType=UIM_PARTIAL)
```

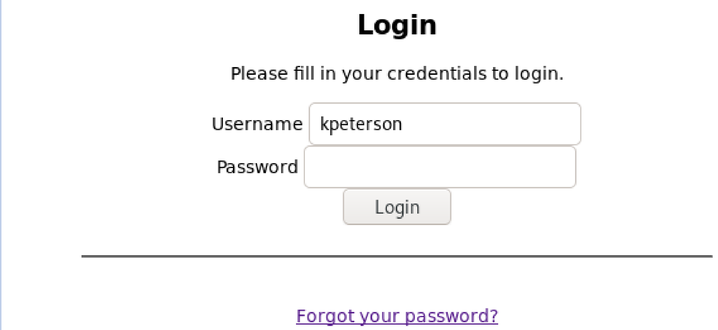
Maltego: Network Analysis



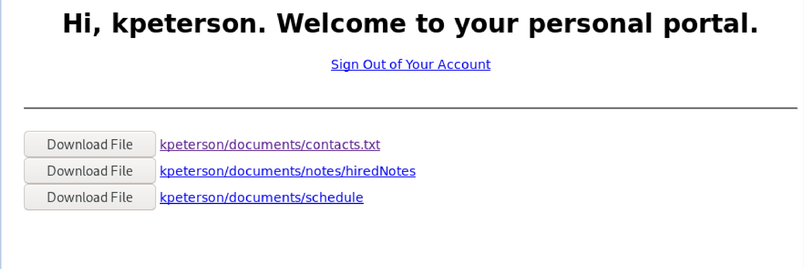
Vulnerable Webserver

Features

- Login page with "Forgot my password" function



- File download



Capture The Flag (CTF)