Reconnaissance, Info Gathering & OSINT

Overview: **Information gathering (reconnaissance)** is the initial phase of penetration testing, focused on collecting public information to identify vulnerabilities and create a testing strategy. It's divided into **passive** and **active** methods.

Passive Information Gathering

Passive information gathering involves collecting data without direct interaction, minimizing detection risk. **Open-Source Intelligence (OSINT)** gathers information from publicly available sources. Techniques include: **Web Scraping, Google Dorking, and social media profiling.** Tools include: **host, nslookup, dig, whois, knockpy, netdiscover, traceroute, whatweb, theHarvester, sherlock, wfw00f, Google Dorking, OSINT framework.**

Active Information Gathering

Active information gathering (scanning) involves direct interaction with the target to collect data like open ports and vulnerabilities. It's more detectable. Should **ONLY** be performed with written permission. **nmap** is a common tool. Addressed in later handouts.

Host

host is a utility for DNS lookups, converting names to IP addresses and vice versa.

- `$ host arifbutt.me`
- `$ host bbc.com` (Shows A, AAAA, and MX records)

Nslookup

Nslookup (Name Server Lookup) maps names to IP addresses and vice versa and can retrieve specific DNS records (A, AAAA, MX, NS, TXT).

- `$ nslookup arifbutt.me`
- Shows DNS server used, answer type (authoritative/non-authoritative), and IP address.
- **Reverse DNS lookup**: Resolves an IP address to a domain name. `$ nslookup 68.65.120.238`

Dig

dig (Domain Information Groper) provides detailed DNS lookup output.

- `$ dig google.com`
- Shows query details, response header, question section, answer section, and query statistics (query time, server used, response size).
- **Time to Live (TTL)** specifies how long a server can cache DNS information.

Whois

Whois retrieves domain registration information from whois databases. This includes registry, registrar, registration and expiration dates, name servers, and contact details of the domain owner (registrant). Useful for determining domain name availability.

- `$ sudo apt-get install whois`
- `$ whois google.com`

Output shows:

- **Registry Domain ID:** Unique identifier within the registry.
- **Registrar WHOIS Server:** Address of the server managing the domain's information.
- **Registrar URL:** Website of the domain registrar.
- **Dates Section:** Updated, Creation, and Expiration dates.
- **Domain Status Section:** Current status of the domain.
- **Name Server Section:** Authoritative DNS servers.

Whois with IP address: Provides information about the organization owning/managing the IP address, including IP range (netblock) and organization details.

- `$ whois 8.8.8.8`

Online web services for domain information: * `https://whois.domaintools.com/` * `https://centralops.net/co/` * `https://ipinfo.io/`

Knockpy

Knockpy is an open-source tool for **subdomain enumeration**. Identifies subdomains by sending requests and collecting responses.

- `$ git clone https://github.com/guelfoweb/knock.git`
- `$ cd knock`
- `$ pip3 install -r requirements.txt`
- `$ which knockpy`
- `$ knockpy –-version`
- `$ knockpy -h`
- `$ knockpy -d pu.edu.pk –-recon --bruteforce –-threads 50`

Netdiscover

Netdiscover is an active/passive network discovery tool using ARP to identify hosts in a LAN.

- `$ sudo apt-get install netdiscover`
- `$ man netdiscover`

Active Scanning: Sends ARP requests to every IP in a range.

- `$ sudo netdiscover –r 10.0.2.0/8`

Passive Scanning: Listens to network traffic to detect devices without sending requests.

- `$ sudo netdiscover -p -r 10.0.2.0/8`

TraceRoute

Traceroute traces the path packets take from your device to a remote server. Shows routers/ gateways (hops) and round-trip-time (RTT) for each hop.

- `$ traceroute -I arifbutt.me`

Output shows:

- IP of the target.
- Hop number, IP of the next router, and three RTT values in milliseconds.
- Asterisks indicate no response within the timeout limit.

Whatweb

Whatweb identifies web technologies used on a target website.

- `$ sudo apt-get install whatweb`
- `$ whatweb -v pucit.edu.pk`

Information gathered:

- **HTTP Headers:** Server type, content type, and metadata.
- **Web Server Information:** (e.g., Apache, Nginx, IIS).
- **CMS (Content Management System):** (e.g., WordPress, Joomla, Drupal).
- **Frameworks and Libraries:** (e.g., Django, Ruby on Rails, jQuery, React).
- **Plugins and Extensions:** Any detected plugins or extensions.

Whatweb - Aggressive Scan

Aggressive scan using the `-a` option with an aggression level (e.g., `3`) to scan an IP range (e.g., `10.0.2.1-10.0.2.254`). Suppress errors with `--no-errors`. * `$ whatweb -v -a 3 10.0.2.1-10.0.2.254 --no-errors`

TheHarvester

TheHarvester is an **OSINT** tool for gathering information about a target, including domain names, IP addresses, and email addresses from public sources (search engines, social media).

- `$ sudo apt-get install theharvester`
- `$ theharvester -help`

Key options: * `-d`: Specifies the domain. * `-l`: Specifies the limit of search results (default 500). * `-b`: Specifies the source (e.g., yahoo, google, bing, baidu, shodan).

Example: * `$ theHarvester -d pucit.edu.pk -l 100 -b yahoo` Another tool is mentioned `hunter.io`

Sherlock

Sherlock is a command-line tool to find usernames across social media platforms and websites.

Installation: `* $ sudo apt update * $ sudo apt install python3 python3-pip git * $ git clone https://github.com/sherlock-project/sherlock.git * $ cd sherlock * $ sudo pip3 install -r requirements.txt * $ sherlock -h * $ sherlock --version * $ sherlock <username>`

Wafw00f

Wafw00f identifies Web Application Firewalls (WAF). It helps to: * Detect WAF presence and type. * Analyze WAF vendors and types.

Installation: `* $ sudo apt update * $ sudo apt install python3 python3-pip git * $ git clone https://github.com/EnableSecurity/wafw00f.git * $ cd wafw00f * $ sudo pip3 install . * $ man wafw00f * $ wafw00f <target_url> * $ wafw00f -i <urls.txt>`

Google Hacking/Dorking

Google Dorking uses advanced search operators to find information not readily available.

- `filetype:`: Searches for specific file types (e.g., `filetype:pdf "Advanced Network Security"`).
- `inurl:`: Finds words within a URL (e.g., `inurl:admin.php`).
- `intitle:`: Searches for terms in a webpage title (e.g., `intitle:"index of"`).
- `link:`: Finds pages that link to a specific URL (e.g., `link:arifbutt.me`).
- `site:`: Searches within a specific site (e.g., `site:pucit.edu.pk inurl:admin`).
- `intext:`: Searches for text within the content of a webpage (e.g., `site:daraz.pk intext:admin`).

OSINT Framework

Open-Source Intelligence (OSINT) involves collecting and analyzing publicly available information for various purposes.

- The **OSINT Framework** (https://osintframework.com/) is a categorized collection of free OSINT tools and resources.

The framework organizes tools into categories such as:

- **Search Engines**: Specialized web search tools.
- **People Search**: Resources for finding information about individuals.
- **Usernames and Social Media**: Tools for tracking social media activity and enumerating usernames.
- **Email Addresses**: Tools for finding and verifying email addresses.
- **Domain and IP Information**: Resources for gathering website and domain information.
- **Public Records**: Access to government and organizational databases.

- **Geolocation**: Tools for extracting geolocation data.
- **Malware and Threat Intelligence**: Tools for analyzing malware and threat data.
- **Metadata and File Analysis**: Tools for extracting metadata from files.
- **Dark Web Tools**: Resources for navigating the dark web.

Disclaimer

The handouts are for **educational purposes only**. Misuse of the information can result in criminal charges, and the authors are not responsible for illegal actions taken by individuals.