E.g My target is a school called ug.edu.gh (University Of Ghana)

So we are to gather information of our target;

When conducting reconnaissance on a target like ug.edu.gh, the goal is to gather as much information as possible in both a passive and active manner to assess potential weaknesses, vulnerabilities, or attack vectors. Gathering this information can help you understand the network, its infrastructure, and its defenses. Here’s how you can gather this information and why each piece of it is important.

**1. Why Gather Information?**

Understanding your target’s environment gives insight into:

* **Potential Attack Surfaces:** Identifying weak points such as unpatched systems, vulnerable software, or misconfigurations.
* **Security Posture:** Assessing how well the target is defended. For instance, knowing if certain ports are open or closed may reveal firewall rules or security measures.
* **Operational Intelligence:** Learning about employees, domains, email structures, and infrastructure to help in social engineering or phishing attacks.
* **Planning Exploits:** Once vulnerabilities are identified, they can be exploited using relevant techniques or tools.

**There Two Phases Of Information Gathering**

**Phases of Reconnaissance**

Reconnaissance is typically broken down into two phases: **Passive Recon** and **Active Recon**. Let’s explore both phases and the specific techniques used to gather valuable information.

**Phase 1: Passive Reconnaissance**

In passive reconnaissance, you are not interacting directly with the target. This helps avoid detection, as the target’s systems won’t flag your activities. Instead, you’re using publicly available information

**Tools:**

* dig
* nslookup
* dnsenum
* whois
* Search Engine Queries (Google Dorking)

**Why:**

* To find subdomains, servers, and the IP infrastructure behind the domain.
* To find mail servers or other critical systems that could be vulnerable.

**Phase 2: Active Reconnaissance**

In active reconnaissance, you directly interact with the target system. This phase is riskier because it can potentially alert the target to your activities.

**2.6 Network Scanning (Port Scanning)**

Port scanning helps identify open services on the target, which could provide insight into vulnerabilities in the system.

**Tools:**

* nmap
* masscan
* Shodan

**Why:**

* To find open ports, running services, and their versions (which can reveal vulnerabilities).
* To detect firewalls, intrusion detection systems (IDS), or proxy systems.

**PASSIVE** **PENTESTING NETWORK**

**Discovering hosts from the outside;**

find IPs responding from the Internet. In this situation you have some scope of IPs (maybe even several ranges) and you just to find which IPs are responding.

ICMP is the simplest way to discovery hosts is up and running; This sends some **ICMP** packets and **expect** responses from the target system

ping -c 1 ug.edu.gh # 1 echo request to a host

* fping -g 199.66.11.0/24 # Send echo requests to ranges
* host ug.edu.gh # Tells you the sysyems is up and running
* whois <Target IP> #Gives more info about the server running
* nslookup
* whatweb -a 1 <URL Of Targert> #Stealthy
* whatweb -a 3 <URL> #Aggresive
* webtech -u <URL>
* webanalyze -host https://google.com -crawl 2
* ./fierce --domain <Target IP>
* theHarvester **-d ug.edu.gh -l 500 -b google**
* theHarvester --no-error <Target IP>

You can visit www.die.net.com to learn about all the various command [Linux Documentation (die.net)](https://linux.die.net/)

**Google Dorking**

site:ug.edu.gh

intitle:"index of" site:ug.edu.gh

filetype:xls OR filetype:doc site:ug.edu.gh

**Discovering Hosts from the Inside;**

If you are inside the network one of the first things you will want to do is to discover other hosts. Depending on how much noise you can/want to do, different actions could be performed:

Passive

You can use these tools to passively discover hosts inside a connected network:

Sudo netdiscover -p

p0f -i eth0 -p -o /tmp/p0f.log

# Bettercap

net.recon on/off #Read local ARP cache periodically

net.show

set net.show.meta true #more info

**ACTIVE PENTESTING NETWORK**

**Discovering hosts from the outside;**

use **nmap** to send other types of ICMP packets (this will avoid filters to common ICMP echo request-response)

* nmap <Target IP>
* nmap -PE -PM -PP -sn -n 199.66.11.0/24 #Send echo, timestamp requests and subnet mask
* nmap -A <Targert IP> # OS Detection

-sV <//> # Detect service version

-p- <//> # Scan all 65535 ports

-O <//> # Enable OS detection

-sT <//> # tcp port scanning

-sS <//> # Syn scan

-sU <//> # UDP scan

-sX <//> #Xmas scan

-sN <//> # Null scan

nmap -sU -sV --version-intensity 0 -F -n <Target IP>/24

# The -sV will make nmap test each possible known UDP service packet

# The "--version-intensity 0" will make nmap only test the most probable

**Discovering hosts from the outside;**

#ARP discovery

nmap -sn <Network> #ARP Requests (Discover IPs)

netdiscover -r <Network> #ARP requests (Discover IPs)

#NBT discovery

nbtscan -r 192.168.0.1/24 #Search in Domain

# Bettercap

net.probe on/off #Discover hosts on current subnet by probing with ARP, mDNS, NBNS, UPNP, and/or WSD

set net.probe.mdns true/false #Enable mDNS discovery probes (default=true)

set net.probe.nbns true/false #Enable NetBIOS name service discovery probes (default=true)

set net.probe.upnp true/false #Enable UPNP discovery probes (default=true)

set net.probe.wsd true/false #Enable WSD discovery probes (default=true)

set net.probe.throttle 10 #10ms between probes sent (default=10)

#IPv6

alive6 <IFACE> # Send a pingv6 to multicast.