📌Lab 28 — Gathering DNS information with dnsenum

✅ **Objective**
→ Learn how to gather DNS information on a target using the `dnsenum` tool.

✅ **Why? (Purpose)**
→ DNS enumeration helps attackers and defenders collect valuable data about a target:

- Usernames
- Computer names
- IP addresses
- Subdomains
- Mail servers
- Name servers
- Possible domain takeovers
  → In other words, it maps out the attack surface.

✅ **Lab Environment**
→ Kali Linux (can be run on a Virtual Machine)

What is dnsenum?

- **dnsenum** is a program (a script) you run in Kali Linux.
- Its job is to find out information about a website or domain, like:
  - Its servers (computers that run the site)
  - The IP addresses of these servers
  - Email servers

- Subdomains (for example, `login.google.com`, `mail.google.com`)
- Basically, anything connected to the domain's DNS records

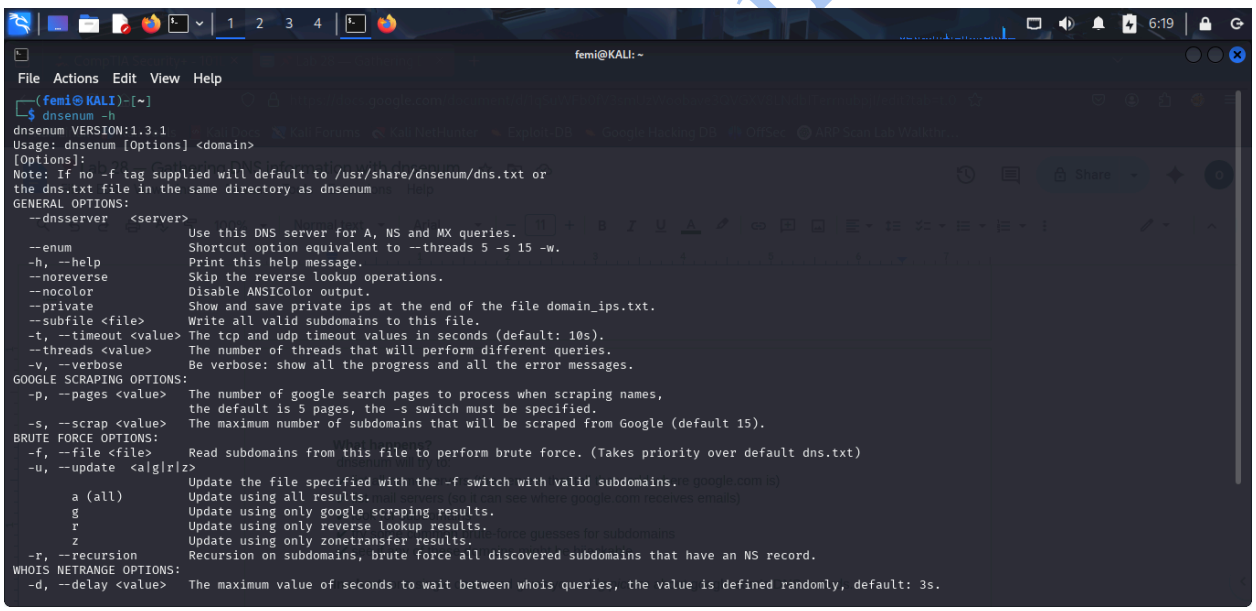
Think of it like a detective who tries to find out everything about a website's infrastructure.

🟦 Step-by-Step, Explained from Scratch

✅ Task 1 — First check if the tool works

In Kali, open your terminal and type:

`dnsenum -h`

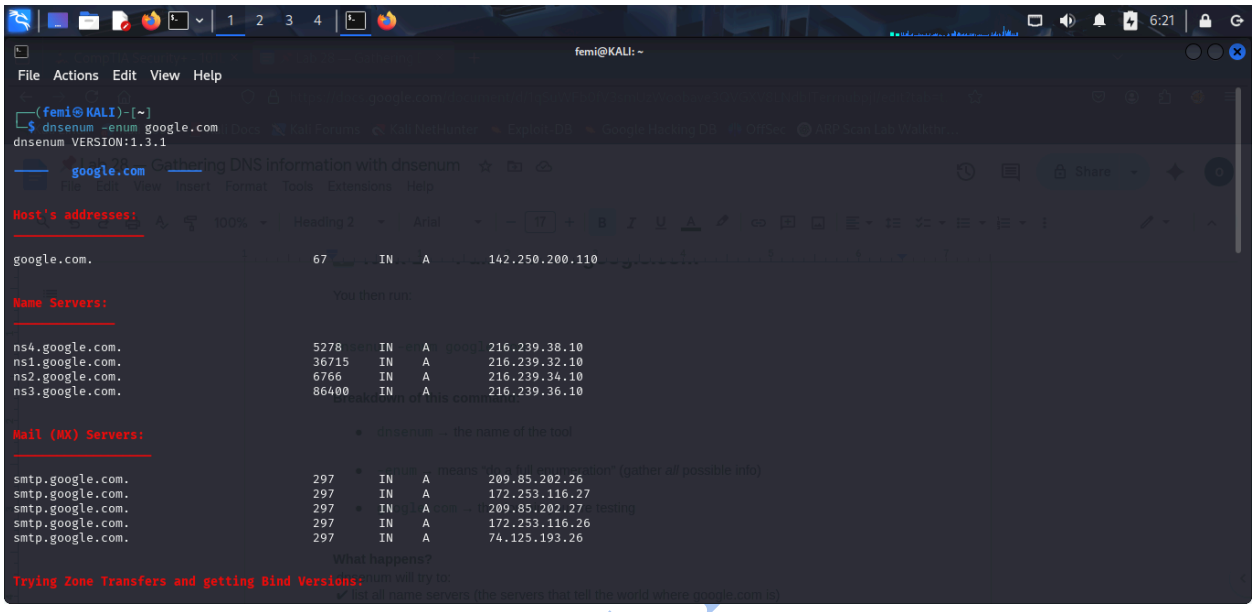


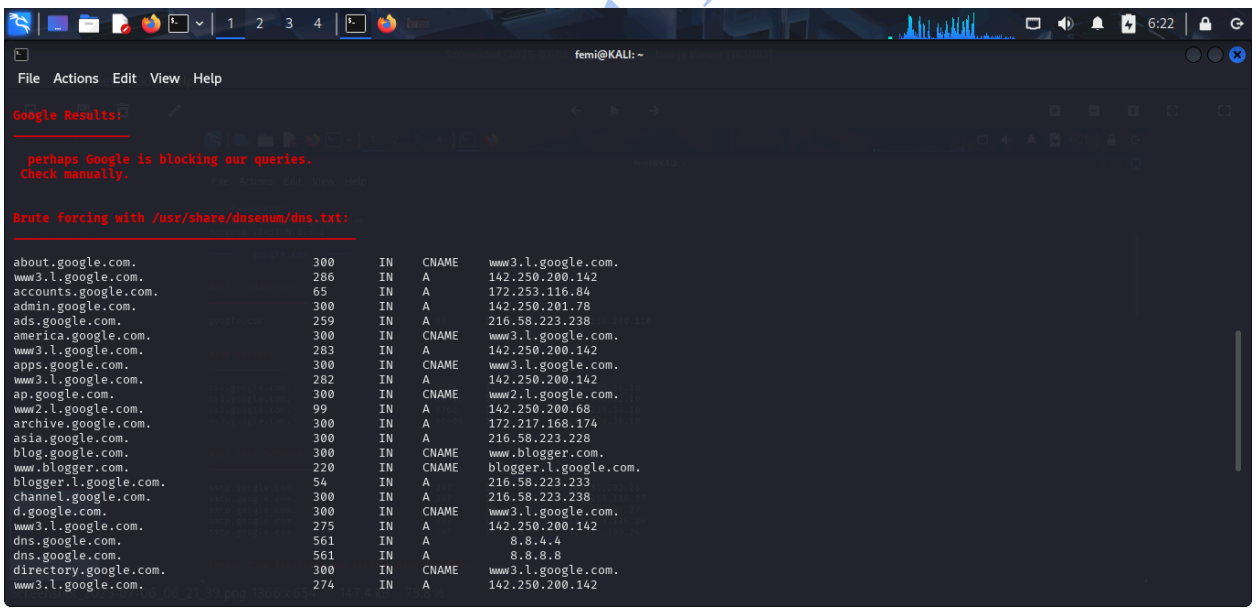
This just shows you the **help** screen of dnsenum, explaining the options you can use. Nothing scary — it's like asking the tool: *"What can you do?"*

✅ Task 1 — Full scan on google.com

You then run:

`dnsenum -enum google.com`





Breakdown of this command:

- `dnsenum` → the name of the tool

- `-enum` → means “do a full enumeration” (gather *all* possible info)

- `google.com` → the domain you are testing

What happens?
dnsenum will try to:
✔ list all name servers (the servers that tell the world where google.com is)
✔ list mail servers (so it can see where google.com receives emails)
✔ look for subdomains
✔ try some common brute-force guesses for subdomains
✔ see if any of these domains might be hijackable

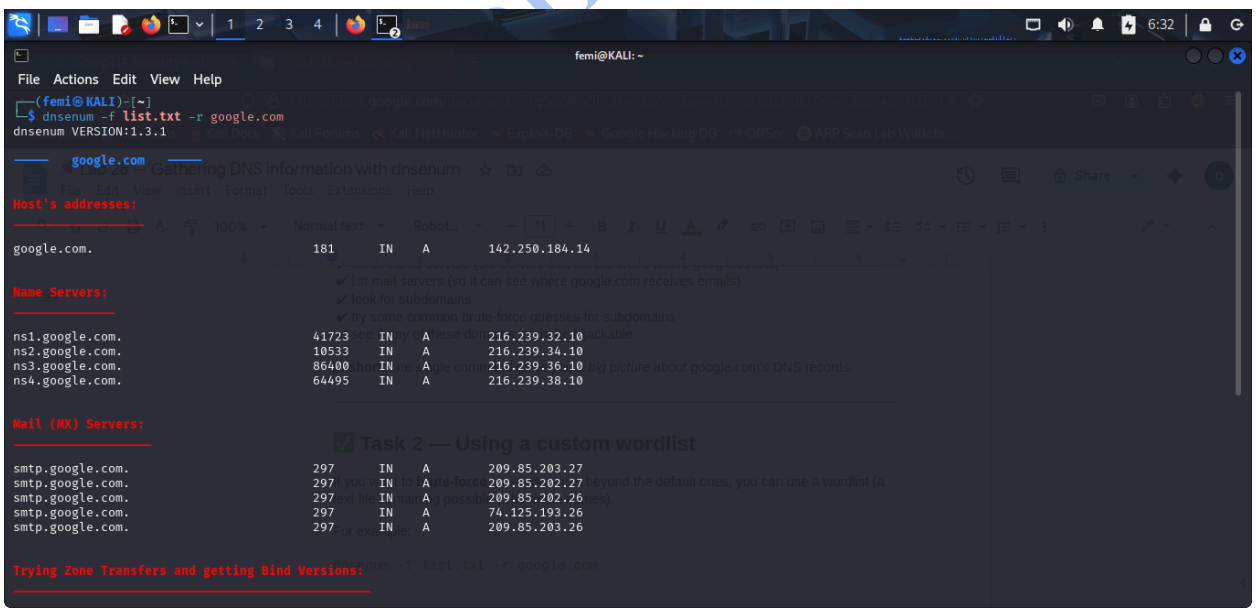
In short: one single command gives you a *big picture* about google.com’s DNS records.

✅ Task 2 — Using a custom wordlist

If you want to **brute-force** for subdomains beyond the default ones, you can use a wordlist (a text file containing possible subdomain names).

For example:

```
dnsenum -f list.txt -r google.com
```



Breakdown:

- `-f list.txt` → tells dnsenum to use your own file of subdomain guesses
- `-r` → means it will recursively try them
- `google.com` → still your target