

Guide for Beginners **DNS** **Footprinting Tool**

DNS Footprinting



By:Souleiman Guedi

Content

Introduction

Basic DNS Queries

Subdomain Enumeration

DNS Zone Transfers (AXFR)

Reverse DNS Lookup

Mail Server Enumeration

DNS Brute-Forcing (Advanced)

Reporting & Documentation

Bonus: Automation & Scripting

Legal & Best Practices

Helpful Artifacts

Introduction

What is DNS Footprinting?

DNS Footprinting is the process of collecting **DNS records, subdomains, IP addresses, and mail servers** associated with a target domain. It helps attackers (or ethical hackers) map out the target's infrastructure.

Why is it Important?

- Reveals **hidden subdomains** (e.g., admin.example.com).
- Identifies **misconfigured DNS servers** (e.g., open zone transfers).
- Helps in **phishing attacks** (finding email servers).
- Assists in **network mapping** (IP ranges, services).

Legal & Ethical Considerations

- **Only perform on authorized targets** (your own domain, bug bounty programs, or with permission).
- **Avoid aggressive scanning** (rate limits can block you).

Basic DNS Queries

Using nslookup

- Checks DNS records interactively.



```
(sg-learning㉿kali)-[~] $ nslookup
> set type=any
>
```

```
set type=any
```

This tells nslookup to return any kind of DNS record it can find — like:

A (address) records

MX (mail) records

TXT (text) records

NS (name servers)

SOA (start of authority)

And more.

```
(sg-learning㉿SG)-[~]
$ nslookup
> set type=any
> example.com
```

(example.com) This is the domain you're querying. It tells nslookup: "Give me all the DNS records you can find for example.com." If type=any is set, you'll get multiple types of records in one response.

```
> server 8.8.8.8
Default server: 8.8.8.8
Address: 8.8.8.8#53
```

- **Records to check:**

- A (IPv4 address)
- AAAA (IPv6 address)
- MX (Mail servers)
- NS (Nameservers)
- TXT (SPF, DKIM, DMARC records)

Using dig (More Powerful)

```
(sg-learning㉿SG)-[~]
$ dig example.com ANY @8.8.8.8
```

DNS Footprinting

SG-Learning

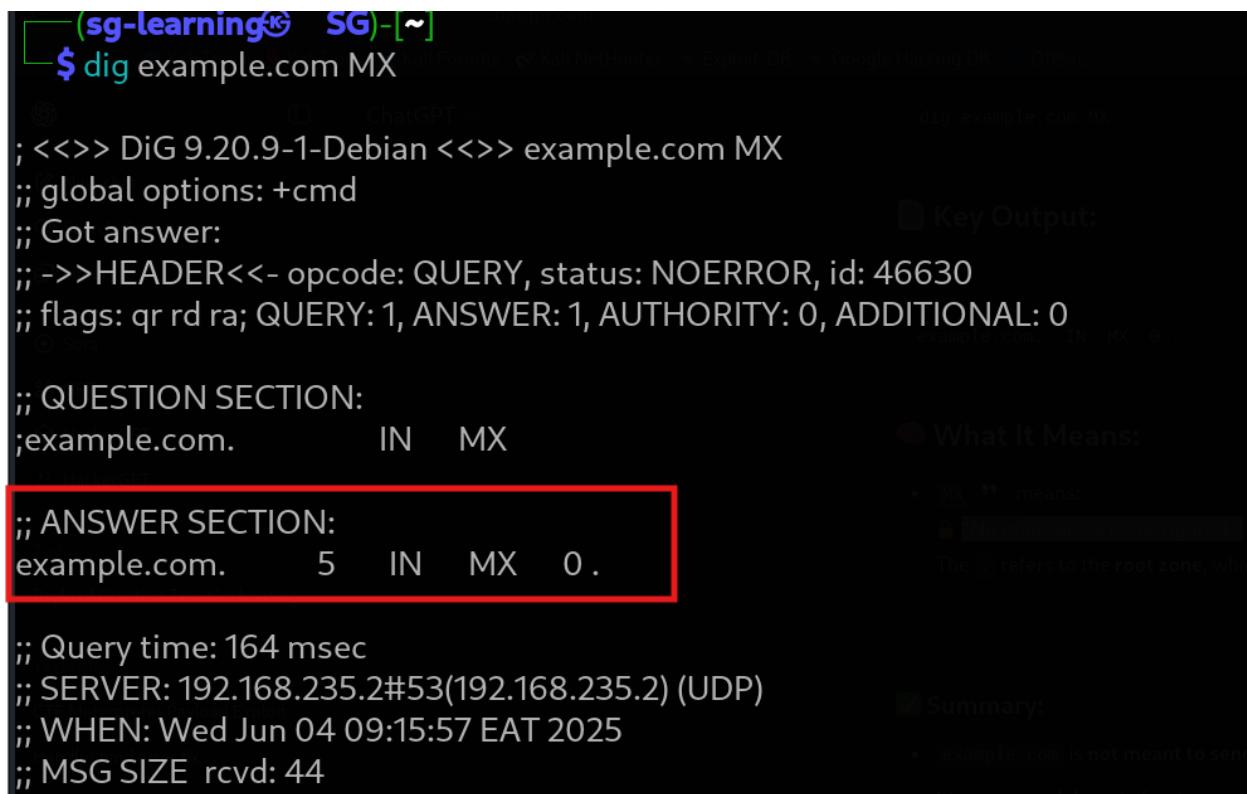
Dig example.com ANY @8.8.8.8 = All records

dig = DNS query tool

example.com = domain to query

ANY = request all DNS record types

@8.8.8.8 = use Google DNS server



```
(sg-learning) SG-[~]
$ dig example.com MX

; <>> DiG 9.20.9-1-Debian <>> example.com MX
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 46630
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;example.com.           IN    MX
;; ANSWER SECTION:
example.com.      5    IN    MX    0.

;; Query time: 164 msec
;; SERVER: 192.168.235.2#53(192.168.235.2) (UDP)
;; WHEN: Wed Jun 04 09:15:57 EAT 2025
;; MSG SIZE rcvd: 44
```

MX 0. means: "No mail server is configured."

The . refers to the root zone, which effectively means no email is accepted for example.com.

Only IPs (faster)

```
(sg-learning㉿ SG)-[~]
$ dig example.com +short
96.7.128.175
96.7.128.198
23.192.228.80
23.192.228.84
23.215.0.136
23.215.0.138
```

Only IPs (faster) = **dig example.com +short**

- Advanced uses:

- +trace → Follows DNS resolution path
- x → Reverse DNS lookup (IP → Domain)

```
(sg-learning㉿ SG)-[~]
$ dig example.com +trace
; <>> DiG 9.20.9-1-Debian <>> example.com +trace
;; global options: +cmd
.          5    IN   NS   b.root-servers.net.
.          5    IN   NS   d.root-servers.net.
.          5    IN   NS   e.root-servers.net.
.          5    IN   NS   h.root-servers.net.
.          5    IN   NS   g.root-servers.net.
.          5    IN   NS   j.root-servers.net.
.          5    IN   NS   f.root-servers.net.
.          5    IN   NS   a.root-servers.net.
.          5    IN   NS   c.root-servers.net.
.          5    IN   NS   m.root-servers.net.
.          5    IN   NS   i.root-servers.net.
.          5    IN   NS   l.root-servers.net.
.          5    IN   NS   k.root-servers.net.
```

Meaning:

- These are A records — IPv4 addresses that
- It looks like Akamai CDN IPs (content delivery network) routing traffic through Akamai
- +short gives you clean output, just the IPs

You can use this to test or script DNS lookups

dig example.com +trace

This command performs a full DNS resolution trace, starting from the root DNS servers all the way down to the authoritative nameserver for example.com.

DNS Footprinting

SG-Learning

`dig -x 8.8.8.8` only works with IP addresses, not domain names.

```
(sg-learning) SG-[~]
$ dig -x 8.8.8.8

; <>> DiG 9.20.9-1-Debian <>> -x 8.8.8.8
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 46692
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; QUESTION SECTION:
;8.8.8.8.in-addr.arpa.      IN    PTR
;; ANSWER SECTION:
8.8.8.8.in-addr.arpa. 5    IN    PTR    dns.google.

;; Query time: 8 msec
;; SERVER: 192.168.235.2#53(192.168.235.2) (UDP)
;; WHEN: Wed Jun 04 09:28:42 EAT 2025
;; MSG SIZE rcvd: 62

dig -x 8.8.8.8
  ↪ Why It Doesn't Work:
  ↪ The -x flag is for reverse DNS lookups. If you want to find the domain name for an IP address, use dig -t A instead.
  ↪ So, dig -x shows an error or does nothing.

  ↪ Correct Usage of -x:
  ↪ If you want to find the domain name for an IP address, use dig -t A instead.
  ↪ This asks: "What domain name is associated with the IP address 8.8.8.8?"
```

Subdomain Enumeration

A. Passive Methods (No Direct Queries)

- Tools:

- Sublist3r (scrapes Google, DNSdumpster, Netcraft)

```
(sg-learning) SG-[~]
$ sublist3r -d example.com -o subdomains.txt
```

- Online Tools:

- [DNSdumpster](#)
 - [Crt.sh](#) (Finds subdomains via SSL certificates)

DNS Footprinting

SG-Learning

The screenshot shows the DNSDumpster.com website. At the top, there's a navigation bar with links for Learn, Defend, API, FAQ, Membership, Login, and a mail icon. Below the header, a sub-header reads "dns recon & research, find & lookup dns records". A main input field is labeled "Enter a Domain to Test" with "example.com" typed into it. Below the input field is a green "Start Test!" button. A red arrow points from the bottom right towards this button. At the bottom of the page, a note states: "DNSDumpster.com is a FREE domain research tool that can discover hosts related to a domain. Finding visible hosts from the attackers perspective is an important part".

crt.sh Certificate Search

Enter an **Identity** (Domain Name, Organization Name, etc),
a **Certificate Fingerprint** (SHA-1 or SHA-256) or a **crt.sh ID**:

Search [Advanced...](#)

© Sectigo Limited 2015-2025. All rights reserved.



B. Active Methods (Brute-Forcing)

- Wordlist-based attacks:

```
(sg-learning㉿ SG-[~])  
$ gobuster dns -d example.com -w /usr/share/wordlists/seclists/Discovery/DNS/subdomains-top1million-5000.txt -t 50
```

Uses wordlists (/usr/share/wordlists/seclists/Discovery/DNS/subdomains-top-5000.txt)

DNS Zone Transfers (AXFR)

What is a Zone Transfer?

- A DNS server **sends all records** to another server (misconfigured if public).
- **Exploitable if allowed:**

```
(sg-learning㉿ SG)-[~]
$ dig @a.iana-servers.net example.com AXFR
```

Reverse DNS Lookup

- Finds **domains linked to an IP range**.

```
(sg-learning㉿ SG)-[~]
$ for ip in $(seq 1 254); do host 192.168.1.$ip:done | grep -v 1.1.168.192.in-addr.arpa domain name
```

- Useful for **finding hidden services**.

Mail Server Enumeration

- Find **MX records**:

```
(sg-learning㉿ SG)-[~]
$ dig example.com MX +short
0 .
```

- Check for **misconfigurations**:

- Open relays
- Weak SPF/DKIM/DMARC policies

DNS Brute-Forcing (Advanced)

- Using `dnsenum`:

DNS Footprinting

SG-Learning

The screenshot shows a terminal session on a Linux system named 'sg-learning'. The user runs the command '\$ dnsenum example.com --threads 10 -o report.xml'. The output includes the version of dnsenum (1.3.1), a section for 'Host's addresses:' listing various IP addresses for the domain, and a section for 'Name Servers:'.

Host's addresses:

Address	TTL	Type	Value
example.com.	5	IN A	23.215.0.136
example.com.	5	IN A	96.7.128.175
example.com.	5	IN A	23.215.0.138
example.com.	5	IN A	23.192.228.84
example.com.	5	IN A	23.192.228.80
example.com.	5	IN A	96.7.128.198

Name Servers:

- Combines **Google scraping, brute-forcing, and zone transfer checks**.
- dnsenum** = DNS enumeration tool to gather info about domains.

example.com = target domain.

--threads 10 = run with 10 concurrent threads (faster scanning).

-o report.xml = save the output/results in XML format to report.xml.

Reporting & Documentation

- Structure your findings:**

```
# DNS Footprinting Report - example.com
#
## Subdomains Found
- admin.example.com
- dev.example.com
#
## Vulnerabilities
- Open AXFR on ns1.example.com
```

- Tools for automation:**

- Metasploit (auxiliary/gather/dns_info)
- DNSRecon (XML/CSV reports)

Bonus: Automation & Scripting

Bash Script Example

```
#!/bin/bash
domain=$1
echo "[+] Running DNS recon on $domain"
dig $domain ANY +noall +answer
dnsrecon -d $domain -t std,axfr -j report.json
```

- **Scheduled monitoring:**
 - Use cron to run weekly scans.

Legal & Best Practices

- **Always get permission** before scanning.
- **Use rate-limiting** to avoid detection.
- **Avoid illegal activities** (unauthorized hacking is a crime).

Helpful Artifacts

Cheat Sheet

1. `dig example.com ANY` → All records
2. `dnsrecon -d example.com` → Subdomains
3. Check AXFR: `dig @ns1 example.com AXFR`

Checklist

- Basic DNS queries (nslookup, dig)
- Subdomain enumeration (Sublist3r, dnsenum)
- Mail server checks (MX, TXT)
- Report findings