

# Dig Command

## Objective:

Learn how to use the Dig command to gather DNS information.

## Purpose:

Dig stands for Domain Information Groper. It is a tool for querying DNS nameservers for information about host addresses, mail exchanges, nameservers and related information.

## Tool:

Kali Linux.

## Topology:

You can use Kali Linux for this lab.

## Walkthrough:

### Task 1:

Dig is a tool which can be used on either Linux or Mac OS. Dig comes pre-installed on Kali Linux and you can check its version using the following command:

```
dig -v
```

The dig syntax looks like the following:

```
Dig [server] [name] [type]
```

We will begin by performing a simple dig command. Type the following into a terminal:

```
dig google.com
```

## Task 2:

The above command will include several information. There may be a time when you only want the

```
(root@kali)-[~]
# dig google.com statistics
;; global options: +cmd
;; Got answer:
;; -->HEADER<-- opcode: QUERY, status: NOERROR, id: 62136
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
100 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.041 ms
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;google.com.                IN      A
100 bytes from mad41s13-in-f14.1e100.net (142.250.200.110): icmp_seq=1 ttl=11
113 ms
100 bytes from mad41s13-in-f14.1e100.net (142.250.200.110): icmp_seq=2 ttl=11
113 ms
;; ANSWER SECTION:
google.com. 4 received, 0% packet loss, time 3057ms
google.com. 11 1/0/0 IN 0.15 A 0.042 142.250.185.14

;; Query time: 8 msec
;; SERVER: 172.20.10.1#53(172.20.10.1) (UDP)
;; WHEN: Mon Jan 27 08:00:18 EST 2025 128 bytes of data.
;; MSG SIZE rcvd: 55
100 bytes from mad41s13-in-f14.1e100.net (142.250.200.110): icmp_seq=1 ttl=11
113 ms
100 bytes from mad41s13-in-f14.1e100.net (142.250.200.110): icmp_seq=2 ttl=11
113 ms
```

result of the query. This can be achieved in dig with the following command:

dig google.com +short

```
(root@kali)-[~]
# dig google.com +short
google.com. 49 IN A 142.250.184.174
100 bytes from mad41s13-in-f14.1e100.net (142.250.200.110): icmp_seq=1 ttl=11
113 ms
100 bytes from mad41s13-in-f14.1e100.net (142.250.200.110): icmp_seq=2 ttl=11
113 ms
```

As you can see, there can be more than one IP for a host record.

## Task 3:

This next command will get rid of all information before the answer section, for easier reading. We can specify this using the following command:

```
(root@kali)-[~]
# dig google.com +noall +answer
google.com. 49 IN A 142.250.184.174
100 bytes from mad41s13-in-f14.1e100.net (142.250.200.110): icmp_seq=1 ttl=11
113 ms
100 bytes from mad41s13-in-f14.1e100.net (142.250.200.110): icmp_seq=2 ttl=11
113 ms
```

## Task 4:

We can also specify the nameservers we wish to query using the following command:

```
(root@kali)-[~]
# dig @8.8.8.8 google.com

; <<>> DiG 9.19.21-1+b1-Debian <<>> @8.8.8.8 google.com 7003ms
; (1 server found)
;; global options: +cmd
;; Got answer:
;; -->HEADER<-- opcode: QUERY, status: NOERROR, id: 7754
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
100 bytes from localhost (::1): icmp_seq=1 ttl=64 time=0.041 ms
;; OPT PSEUDOSECTION: host (::1): icmp_seq=2 ttl=64 time=0.098 ms
; EDNS: version: 0, flags:; udp: 512_seq=3 ttl=64 time=0.076 ms
;; QUESTION SECTION: host (::1): icmp_seq=4 ttl=64 time=0.158 ms
;google.com.                IN      A
100 bytes from localhost (::1): icmp_seq=1 ttl=64 time=0.041 ms
;; ANSWER SECTION:
google.com. 187 0.0 IN 0.15 A 142.250.200.142

;; Query time: 200 msec
;; SERVER: 8.8.8.8#53(8.8.8.8) (UDP)
;; WHEN: Mon Jan 27 08:03:35 EST 2025 128 bytes of data.
;; MSG SIZE rcvd: 55
100 bytes from google.com (142.250.200.110): icmp_seq=1 ttl=112 time=297 ms
100 bytes from google.com (142.250.200.110): icmp_seq=2 ttl=112 time=297 ms
```

This command queries the “google.com” record from the Name Server with IP address 8.8.8.8.

## Task 5:

If we want to query all DNS record types, we can use the “ANY” option. This will display all the available record types in the output:

```
(root@kali)-[~]
# dig google.com ANY

;; communications error to 172.20.10.1#53: timed out
;; communications error to 172.20.10.1#53: timed out
;; communications error to 172.20.10.1#53: timed out

; <<>> DiG 9.19.21-1+b1-Debian <<>> google.com ANY
;; global options: +cmd
;; no servers could be reached
100 bytes from google.com (142.250.200.110): icmp_seq=1 ttl=112 time=297 ms
```

## Task 6:

We can also look up a specific record. For example, if we want to get only the mail exchange section associated with a domain, we can use the following command:

```
dig google.com MX
```

We can query a number of specific record types using the following tags in place of MX:

TXT, CNAME, NS, A

```
(root@kali)-[~]
# dig google.com MX statistics --
;; packet transmitted, 4 received, 0% packet loss, time 7003ms
; <<>> DiG 9.19.21-1+b1-Debian <<>> google.com MX 905 ms
;; global options: +cmd
;; Got answer:
;; -->HEADER<-- opcode: QUERY, status: NOERROR, id: 9319
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;; from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.041 ms
;; OPT PSEUDOSECTION: opt 1:1: icmp_seq=2 ttl=64 time=0.098 ms
;; EDNS: version: 0, flags:; udp: 4096 seq=3 ttl=64 time=0.076 ms
;; QUESTION SECTION:
;; 127.0.0.1: icmp_seq=4 ttl=64 time=0.158 ms
google.com.                IN      MX
-- localhost ping statistics --
;; ANSWER SECTION:
google.com. 374 IN 0.15 MX 042 10 smtp.google.com.

;; Query time: 2177 msec
;; SERVER: 172.20.10.1#53(172.20.10.1) (UDP)
;; WHEN: Mon Jan 27 08:07:28 EST 2025 126 bytes of data
;; MSG SIZE rcvd: 60
127.0.0.1: icmp_seq=1 ttl=11 time=0.041 ms
```

## Task 7:

```

NS i.root-servers.net. from server 172.20.10.1 in 1328 ms.
NS j.root-servers.net. from server 172.20.10.1 in 1328 ms.
NS m.root-servers.net. from server 172.20.10.1 in 1328 ms.
NS e.root-servers.net. from server 172.20.10.1 in 1328 ms.
;; UDP setup with 2001:501:b1f9::30#53(2001:501:b1f9::30) for microsoft.com failed: network unreachable.
;; no servers could be reached

;; UDP setup with 2001:501:b1f9::30#53(2001:501:b1f9::30) for microsoft.com failed: network unreachable.
;; communications error to 192.52.178.30#53: timed out
;; UDP setup with 2001:502:7094::30#53(2001:502:7094::30) for microsoft.com failed: network unreachable.
;; UDP setup with 2001:503:231d::2:30#53(2001:503:231d::2:30) for microsoft.com failed: network unreachable.
;; UDP setup with 2001:502:1ca1::30#53(2001:502:1ca1::30) for microsoft.com failed: network unreachable.
;; UDP setup with 2001:503:d414::30#53(2001:503:d414::30) for microsoft.com failed: network unreachable.
;; UDP setup with 2620:1ec:8ec:10::27#53(2620:1ec:8ec:10::27) for microsoft.com failed: network unreachable.
A 20.70.246.20 from server 13.107.206.39 in 280 ms.
A 20.76.201.171 from server 13.107.206.39 in 280 ms.
A 20.112.250.133 from server 13.107.206.39 in 280 ms.
A 20.231.239.246 from server 13.107.206.39 in 280 ms.
A 20.236.44.162 from server 13.107.206.39 in 280 ms.

```

We can trace the DNS path, similar to traceroute, using the following command:

### Task 8:

```

(olalekan@kali)-[~]
$ dig -x 142.250.75.238
;; communications error to 172.20.10.1#53: timed out
;; communications error to 172.20.10.1#53: timed out
;; communications error to 172.20.10.1#53: timed out

; <<>> DiG 9.19.21-1+b1-Debian <<>> -x 142.250.75.238
;; global options: +cmd
;; no servers could be reached

```

It is also possible to make DNS queries for IP addresses.

### Task 9:

Dig has a useful feature which allows you to perform a number of DNS lookups for a list of domains instead of doing the same for each one individually. This can be done by performing a lookup using a file:

```
dig -f domain_names.txt +short
```

### Task 10:

```
(olalekan@kali)-[~]  
$ dig +short TXT hackaday.com  
"facebook-domain-verification=ie1lkz19o2lsbploq4owagf1snbzsy"  
"google-site-verification=1Xv4FJCKt039C05Cy0mNT4j9zLRbWS03GIicV4x-iQg"  
"v=spf1 include:aspmx.googlemail.com include:mailer.postageapp.com include:ma  
ilgun.org include:servers.mcsv.net ~all"  
"projects google-site-verification=RjppnbZuuM-LhJ6Xb1EG0vnZeM6xvkkMxBxGmOm7ek  
Q"  
"ZOOM_verify_cuY_AVoeSBi4AAVJQvMu-A"
```

It is possible to access domain verification data by making a DNS TXT query.

Dig is a tool with multiple uses and can be very useful for gathering a broad range of DNS information about a target site.