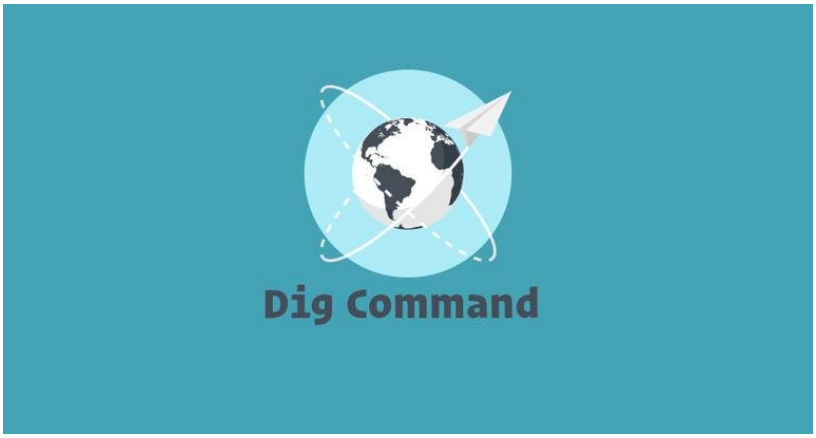
Dig Command in Linux (DNS Lookup)

Updated Feb 26, 2020 • 8 min read



Dig (Domain Information Groper) is a powerful command-line tool for querying DNS name servers.

The `dig` command, allows you to query information about various DNS records, including host addresses, mail exchanges, and name servers. It is the most commonly used tool among system administrators for troubleshooting DNS problems because of its flexibility and ease of use.

This tutorial explains how to use the `dig` utility through practical examples and detailed explanations of the most common `dig` options.

Installing `dig`

To check if the `dig` command is available on your system type:

```
$ dig -v
```

The output should look something like this:

```
Output
DiG 9.11.3-1ubuntu1.1-Ubuntu
```

If `dig` is not present on your system, the command above will print “dig: command not found”. The `dig` tool can be installed using the distro’s package manager.

Install `dig` on Ubuntu and Debian

```
$ sudo apt update && sudo apt install dnsutils
```

Install `dig` on CentOS and Fedora

```
$ sudo yum install bind-utils
```

RELATED TUTORIALS

Linux File Command
How to Create Bash Aliases
How to Rename Files and Directories in Linux
Scheduling Cron Jobs with Crontab
Linux Time Command
How to Create Users in Linux (useradd Command)
Chown Command in Linux (File Ownership)

Install `dig` on Arch Linux

```
$ sudo pacman -S bind-tools
```

Understanding the `dig` Output

In its simplest form, when used to query a single host (domain) without any additional options, the `dig` command is pretty verbose.

In the following example, we're performing on the `linux.org` domain:


```
$ dig linux.org
```

The output should look something like this:

```
; <<>> DiG 9.13.3 <<>> linux.org
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 37159
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2, ADDITIONAL: 5

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;linux.org.                     IN      A

;; ANSWER SECTION:
linux.org.              300     IN      A       104.18.59.123
linux.org.              300     IN      A       104.18.58.123

;; AUTHORITY SECTION:
linux.org.              86379   IN      NS      lia.ns.cloudflare.com.
linux.org.              86379   IN      NS      mark.ns.cloudflare.com.

;; ADDITIONAL SECTION:
lia.ns.cloudflare.com.  84354   IN      A       173.245.58.185
lia.ns.cloudflare.com.  170762  IN      AAAA    2400:cb00:2049:1::adf5:3ab9
mark.ns.cloudflare.com. 170734  IN      A       173.245.59.130
mark.ns.cloudflare.com. 170734  IN      AAAA    2400:cb00:2049:1::adf5:3b82

;; Query time: 58 msec
;; SERVER: 192.168.1.1#53(192.168.1.1)
;; WHEN: Fri Oct 12 11:46:46 CEST 2018
;; MSG SIZE  rcvd: 212
```

Let's go section by section and explain the output of the `dig` command:

01. The first line of the output prints the installed `dig` version, and the queried domain name. The second line shows the global options (by default, only cmd).

    ```
    ; <<>> DiG 9.13.3 <<>> linux.org
    ;; global options: +cmd
    ```

    If you don't want those lines to be included in the output, use the `+nocmd` option. This option must be the very first one after the `dig` command.

02. The next section includes technical details about the answer received from the requested authority (DNS server). The header shows the opcode (the action performed by `dig`) and the status of the action. In this example, the status is `NOERROR`, which means that the requested authority served the query without any issue.

    ```
    ;; Got answer:
    ;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 37159
    ```

```
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 2, ADDITIONAL: 5
```

This section can be removed using the `+nocomments` option, which also disables some other section's headers.

03. The "OPT" pseudo section is shown only in the newer versions of the `dig` utility. You can read more about the Extension mechanisms for DNS (EDNS) here .

```
;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
```

To exclude this section from the output, use the `+noedns` option.

04. In the "QUESTION" section `dig` shows the query (question). By default, `dig` requests the A record.

```
;; QUESTION SECTION:
;linux.org.         IN  A
```

You can disable this section using the `+noquestion` option.

05. The "ANSWER" section provides us with an answer to our question. As we already mentioned, by default `dig` will request the A record. Here, we can see that the domain `linux.org` points to the `104.18.59.123` IP address.

```
;; ANSWER SECTION:
linux.org.      300 IN  A   104.18.59.123
linux.org.      300 IN  A   104.18.58.123
```

Usually, you do not want to turn off the answer, but you can remove this section from the output using the `+noanswer` option.

06. The "AUTHORITY" section tells us what server(s) are the authority for answering DNS queries about the queried domain.

```
;; AUTHORITY SECTION:
linux.org.      86379   IN  NS  lia.ns.cloudflare.com.
linux.org.      86379   IN  NS  mark.ns.cloudflare.com.
```

You can disable this section of the output using the `+noauthority` option.

07. The "ADDITIONAL" section gives us information about the IP addresses of the authoritative DNS servers shown in the authority section.

```
;; ADDITIONAL SECTION:
lia.ns.cloudflare.com.  84354   IN  A   173.245.58.185
lia.ns.cloudflare.com.  170762  IN  AAAA    2400:cb00:2049:1::adf5:3ab9
mark.ns.cloudflare.com. 170734  IN  A   173.245.59.130
mark.ns.cloudflare.com. 170734  IN  AAAA    2400:cb00:2049:1::adf5:3b82
```

The `+noadditional` option disables the additional section of a reply.

08. The last section of the `dig` output includes statistics about the query.

```
;; Query time: 58 msec
;; SERVER: 192.168.1.1#53(192.168.1.1)
;; WHEN: Fri Oct 12 11:46:46 CEST 2018
;; MSG SIZE  rcvd: 212
```

You can disable this part with the `+nostats` option.

Printing Only the Answer

Generally, you would want to get only a short answer to your `dig` query.

1. Get a Short Answer

To get a short answer to your query, use the `+short` option:

```
$ dig linux.org +short
```

```
Output
104.18.59.123
104.18.58.123
```

The output will include only the IP addresses of the A record.

2. Get a Detailed Answer

For more a detailed answer, turn off all the results using the `+noall` options and then turn on only the answer section with the `+answer` option.

```
$ dig linux.org +noall +answer
```

```
Output
; <<>> DiG 9.13.3 <<>> linux.org +noall +answer
;; global options: +cmd
linux.org.      67  IN  A   104.18.58.123
linux.org.      67  IN  A   104.18.59.123
```

Query Specific Name Server

By default, if no name server is specified, `dig` uses the servers listed in `/etc/resolv.conf` file.

To specify a name server against which the query will be executed, use the `@` (at) symbol followed by the name server IP address or hostname.

For example, to query the Google name server (8.8.8.8) for information about the `linux.org` domain you would use:

```
$ dig linux.org @8.8.8.8
```

```
Output
; <<>> DiG 9.13.3 <<>> linux.org @8.8.8.8
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 39110
```

```
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;linux.org.         IN  A

;; ANSWER SECTION:
linux.org.      299 IN  A   104.18.58.123
linux.org.      299 IN  A   104.18.59.123

;; Query time: 54 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Fri Oct 12 14:28:01 CEST 2018
;; MSG SIZE  rcvd: 70
```

Query a Record Type

Dig allows you to perform any valid DNS query by appending the record type to the end of the query. In the following section, we will show you examples of how to search for the most common records, such as A (the IP address), CNAME (canonical name), TXT (text record), MX (mail exchanger), and NS (name servers).

1. Querying A records

To get a list of all the address(es) for a domain name, use the `a` option:

```
$ dig +nocmd google.com a +noall +answer
```

```
Output
google.com.     128 IN  A   216.58.206.206
```

As you already know, if no DNS record type is specified, `dig` will request the A record. You can also query the A record without specifying the `a` option.

2. Querying CNAME records

To find the alias domain name use the `cname` option:

```
$ dig +nocmd mail.google.com cname +noall +answer
```

```
Output
mail.google.com.    553482  IN  CNAME   googlemail.l.google.com.
```

3. Querying TXT records

Use the `txt` option to retrieve all the TXT records for a specific domain:

```
$ dig +nocmd google.com txt +noall +answer
```

```
Output
google.com.     300 IN  TXT "facebook-domain-verification=22rm551cu4k0ab0bxs
google.com.     300 IN  TXT "v=spf1 include:_spf.google.com ~all"
google.com.     300 IN  TXT "docusign=05958488-4752-4ef2-95eb-aa7ba8a3bd0e"
```

4. Querying MX records

To get a list of all the mail servers for a specific domain use the `mx` option:

```
$ dig +nocmd google.com mx +noall +answer
```

```
Output
google.com.     494 IN  MX  30 alt2.aspmx.l.google.com.
google.com.     494 IN  MX  10 aspmx.l.google.com.
google.com.     494 IN  MX  40 alt3.aspmx.l.google.com.
google.com.     494 IN  MX  50 alt4.aspmx.l.google.com.
google.com.     494 IN  MX  20 alt1.aspmx.l.google.com.
```

5. Querying NS records

To find the authoritative name servers for our specific domain use the `ns` option:

```
$ dig +nocmd google.com ns +noall +answer
```

```
Output
google.com.     84527   IN  NS  ns1.google.com.
google.com.     84527   IN  NS  ns2.google.com.
google.com.     84527   IN  NS  ns4.google.com.
google.com.     84527   IN  NS  ns3.google.com.
```

6. Querying All Records

Use the `any` option to get a list of all DNS records for a specific domain:

```
$ dig +nocmd google.com any +noall +answer
```

```
Output
google.com.     299 IN  A   216.58.212.14
google.com.     299 IN  AAAA    2a00:1450:4017:804::200e
google.com.     21599   IN  NS  ns2.google.com.
google.com.     21599   IN  NS  ns1.google.com.
google.com.     599 IN  MX  30 alt2.aspmx.l.google.com.
google.com.     21599   IN  NS  ns4.google.com.
google.com.     599 IN  MX  50 alt4.aspmx.l.google.com.
google.com.     599 IN  MX  20 alt1.aspmx.l.google.com.
google.com.     299 IN  TXT "docusign=05958488-4752-4ef2-95eb-aa7ba8a3bd0e"
google.com.     21599   IN  CAA 0 issue "pki.goog"
google.com.     599 IN  MX  40 alt3.aspmx.l.google.com.
google.com.     3599    IN  TXT "facebook-domain-verification=22rm551cu4k0ab
google.com.     21599   IN  NS  ns3.google.com.
google.com.     599 IN  MX  10 aspmx.l.google.com.
google.com.     3599    IN  TXT "v=spf1 include:_spf.google.com ~all"
google.com.     59  IN  SOA ns1.google.com. dns-admin.google.com. 216967258
```

Reverse DNS Lookup

To query the hostname associated with a specific IP address use the `-x` option.

For example, to perform a reverse lookup on `208.118.235.148` you would type:

```
$ dig -x 208.118.235.148 +noall +answer
```

As you can see from the output below the IP address `208.118.235.148` is associated with the hostname `wildebeest.gnu.org`.

```
Output
; <<>> DiG 9.13.3 <<>> -x 208.118.235.148 +noall +answer
;; global options: +cmd
148.235.118.208.in-addr.arpa. 245 IN    PTR wildebeest.gnu.org.
```

Bulk Queries

If you want to query a large number of domains, you can add them in a file (one domain per line) and use the `-f` option followed by the file name.

In the following example, we are querying the domains listed in the `domains.txt` file.

domains.txt

```
lxer.com
linuxtoday.com
tuxmachines.org
```

```
$ dig -f domains.txt +short
```

```
Output
108.166.170.171
70.42.23.121
204.68.122.43
```

The .digrc File

The `dig` command's behavior can be controlled by setting up per-user options in the `${HOME}/.digrc` file.

If the `.digrc` file is present in the user's home directory, the options specified in it are applied before the command line arguments.

For example, if you want to display only the answer section, open your text editor and create the following `~/.digrc` file:

~/.digrc

```
+nocmd +noall +answer
```

Conclusion

`dig` is a command-line tool for querying DNS information and troubleshooting DNS related issues.

If you have any questions or feedback, feel free to leave a comment.

dig terminal

Related Articles

DEC 21, 2018
Linux File Command

DEC 20, 2018
How to Create Bash Aliases

DEC 19, 2018
How to Rename Files and Directories in Linux