



Full credit is given to the above companies including the OS that this PDF file was generated!

Rocky Enterprise Linux 9.2 Manual Pages on command 'dig.1'

\$ man dig.1

DIG(1) BIND 9 DIG(1)

NAME

dig - DNS lookup utility

SYNOPSIS

dig [@server] [-b address] [-c class] [-f filename] [-k filename] [-m]

[-p port#] [-q name] [-t type] [-v] [-x addr] [-y [hmac:]name:key] [

[-4] | [-6]] [name] [type] [class] [queryopt...]

dig [-h]

dig [global-queryopt...] [query...]

DESCRIPTION

dig is a flexible tool for interrogating DNS name servers. It performs DNS lookups and displays the answers that are returned from the name server(s) that were queried. Most DNS administrators use dig to troubleshoot DNS problems because of its flexibility, ease of use, and clarity of output. Other lookup tools tend to have less functionality than dig.

Although dig is normally used with command-line arguments, it also has a batch mode of operation for reading lookup requests from a file. A

brief summary of its command-line arguments and options is printed when the `-h` option is given. The BIND 9 implementation of `dig` allows multiple lookups to be issued from the command line.

Unless it is told to query a specific name server, `dig` tries each of the servers listed in `/etc/resolv.conf`. If no usable server addresses are found, `dig` sends the query to the local host.

When no command-line arguments or options are given, `dig` performs an NS query for "." (the root).

It is possible to set per-user defaults for `dig` via `$(HOME)/.digrc`.

This file is read and any options in it are applied before the command-line arguments. The `-r` option disables this feature, for scripts that need predictable behavior.

The IN and CH class names overlap with the IN and CH top-level domain names. Either use the `-t` and `-c` options to specify the type and class, use the `-q` to specify the domain name, or use "IN." and "CH." when looking up these top-level domains.

SIMPLE USAGE

A typical invocation of `dig` looks like:

```
dig @server name type
```

where:

`server` is the name or IP address of the name server to query. This can be an IPv4 address in dotted-decimal notation or an IPv6 address in colon-delimited notation. When the supplied server argument is a hostname, `dig` resolves that name before querying that name server.

If no server argument is provided, `dig` consults `/etc/resolv.conf`; if an address is found there, it queries the name server at that address. If either of the `-4` or `-6` options are in use, then only addresses for the corresponding transport are tried. If no usable addresses are found, `dig` sends the query to the local host. The reply from the name server that responds is displayed.

`name` is the name of the resource record that is to be looked up.

type indicates what type of query is required - ANY, A, MX, SIG, etc.

type can be any valid query type. If no type argument is supplied,

dig performs a lookup for an A record.

OPTIONS

-4 This option indicates that only IPv4 should be used.

-6 This option indicates that only IPv6 should be used.

-b address[#port]

This option sets the source IP address of the query. The address

must be a valid address on one of the host's network interfaces,

or "0.0.0.0" or ":::". An optional port may be specified by ap?

pending #port.

-c class

This option sets the query class. The default class is IN; other

classes are HS for Hesiod records or CH for Chaosnet records.

-f file

This option sets batch mode, in which dig reads a list of lookup

requests to process from the given file. Each line in the file

should be organized in the same way it would be presented as a

query to dig using the command-line interface.

-k keyfile

This option tells named to sign queries using TSIG using a key

read from the given file. Key files can be generated using

tsig-keygen. When using TSIG authentication with dig, the name

server that is queried needs to know the key and algorithm that

is being used. In BIND, this is done by providing appropriate

key and server statements in named.conf.

-m This option enables memory usage debugging.

-p port

This option sends the query to a non-standard port on the

server, instead of the default port 53. This option is used to

test a name server that has been configured to listen for

queries on a non-standard port number.

-q name

This option specifies the domain name to query. This is useful to distinguish the name from other arguments.

-r This option indicates that options from `$(HOME)/.digrc` should not be read. This is useful for scripts that need predictable behavior.

-t type

This option indicates the resource record type to query, which can be any valid query type. If it is a resource record type supported in BIND 9, it can be given by the type mnemonic (such as NS or AAAA). The default query type is A, unless the `-x` option is supplied to indicate a reverse lookup. A zone transfer can be requested by specifying a type of AXFR. When an incremental zone transfer (IXFR) is required, set the type to `ixfr=N`.

The incremental zone transfer contains all changes made to the zone since the serial number in the zone's SOA record was N.

All resource record types can be expressed as `TYPEnn`, where `nn` is the number of the type. If the resource record type is not supported in BIND 9, the result is displayed as described in RFC 3597.

-u This option indicates that print query times should be provided in microseconds instead of milliseconds.

-v This option prints the version number and exits.

-x addr

This option sets simplified reverse lookups, for mapping addresses to names. The `addr` is an IPv4 address in dotted-decimal notation, or a colon-delimited IPv6 address. When the `-x` option is used, there is no need to provide the name, class, and type arguments. `dig` automatically performs a lookup for a name like `94.2.0.192.in-addr.arpa` and sets the query type and class to PTR and IN respectively. IPv6 addresses are looked up using nibble format under the IP6.ARPA domain.

-y [hmac:]keyname:secret

This option signs queries using TSIG with the given authenticator.

tion key. keyname is the name of the key, and secret is the base64-encoded shared secret. hmac is the name of the key algorithm; valid choices are hmac-md5, hmac-sha1, hmac-sha224, hmac-sha256, hmac-sha384, or hmac-sha512. If hmac is not specified, the default is hmac-md5; if MD5 was disabled, the default is hmac-sha256.

NOTE:

Only the -k option should be used, rather than the -y option, because with -y the shared secret is supplied as a command-line argument in clear text. This may be visible in the output from ps1 or in a history file maintained by the user's shell.

QUERY OPTIONS

dig provides a number of query options which affect the way in which lookups are made and the results displayed. Some of these set or reset flag bits in the query header, some determine which sections of the answer get printed, and others determine the timeout and retry strategies.

Each query option is identified by a keyword preceded by a plus sign (+). Some keywords set or reset an option; these may be preceded by the string no to negate the meaning of that keyword. Other keywords assign values to options, like the timeout interval. They have the form +keyword=value. Keywords may be abbreviated, provided the abbreviation is unambiguous; for example, +cd is equivalent to +cdflag. The query options are:

+[no]aaflag

This option is a synonym for +[no]aaonly.

+[no]aaonly

This option sets the aa flag in the query.

+[no]additional

This option displays [or does not display] the additional section of a reply. The default is to display it.

+[no]adflag

This option sets [or does not set] the AD (authentic data) bit

in the query. This requests the server to return whether all of the answer and authority sections have been validated as secure, according to the security policy of the server. AD=1 indicates that all records have been validated as secure and the answer is not from a OPT-OUT range. AD=0 indicates that some part of the answer was insecure or not validated. This bit is set by default.

`+noall`

This option sets or clears all display flags.

`+noanswer`

This option displays [or does not display] the answer section of a reply. The default is to display it.

`+noauthority`

This option displays [or does not display] the authority section of a reply. The default is to display it.

`+nobadcookie`

This option retries the lookup with a new server cookie if a BADCOOKIE response is received.

`+nobesteffort`

This option attempts to display the contents of messages which are malformed. The default is to not display malformed answers.

`+bufsize[=B]`

This option sets the UDP message buffer size advertised using EDNS0 to B bytes. The maximum and minimum sizes of this buffer are 65535 and 0, respectively. `+bufsize=0` disables EDNS (use `+bufsize=0 +edns` to send an EDNS message with an advertised size of 0 bytes). `+bufsize` restores the default buffer size.

`+nocdflag`

This option sets [or does not set] the CD (checking disabled) bit in the query. This requests the server to not perform DNSSEC validation of responses.

`+noclass`

This option displays [or does not display] the CLASS when print?

ing the record.

`+[no]cmd`

This option toggles the printing of the initial comment in the output, identifying the version of dig and the query options that have been applied. This option always has a global effect; it cannot be set globally and then overridden on a per-lookup basis. The default is to print this comment.

`+[no]comments`

This option toggles the display of some comment lines in the output, with information about the packet header and OPT pseudo-section, and the names of the response section. The default is to print these comments.

Other types of comments in the output are not affected by this option, but can be controlled using other command-line switches.

These include `+[no]cmd`, `+[no]question`, `+[no]stats`, and `+[no]rrcomments`.

`+[no]cookie=####`

This option sends [or does not send] a COOKIE EDNS option, with an optional value. Replaying a COOKIE from a previous response allows the server to identify a previous client. The default is `+cookie`.

`+cookie` is also set when `+trace` is set to better emulate the default queries from a nameserver.

`+[no]crypto`

This option toggles the display of cryptographic fields in DNSSEC records. The contents of these fields are unnecessary for debugging most DNSSEC validation failures and removing them makes it easier to see the common failures. The default is to display the fields. When omitted, they are replaced by the string [omitted] or, in the DNSKEY case, the key ID is displayed as the replacement, e.g. [key id = value].

`+[no]defname`

This option, which is deprecated, is treated as a synonym for

`+[no]search.`

`+[no]dnssec`

This option requests that DNSSEC records be sent by setting the DNSSEC OK (DO) bit in the OPT record in the additional section of the query.

`+domain=somename`

This option sets the search list to contain the single domain somename, as if specified in a domain directive in `/etc/resolv.conf`, and enables search list processing as if the `+search` option were given.

`+dscp=value`

This option sets the DSCP code point to be used when sending the query. Valid DSCP code points are in the range `[0..63]`. By default no code point is explicitly set.

`+[no]edns[=#]`

This option specifies the EDNS version to query with. Valid values are 0 to 255. Setting the EDNS version causes an EDNS query to be sent. `+noedns` clears the remembered EDNS version. EDNS is set to 0 by default.

`+[no]ednsflags[=#]`

This option sets the must-be-zero EDNS flags bits (Z bits) to the specified value. Decimal, hex, and octal encodings are accepted. Setting a named flag (e.g., DO) is silently ignored. By default, no Z bits are set.

`+[no]ednsnegotiation`

This option enables/disables EDNS version negotiation. By default, EDNS version negotiation is enabled.

`+[no]ednsopt[=code[:value]]`

This option specifies the EDNS option with code point code and an optional payload of value as a hexadecimal string. code can be either an EDNS option name (for example, NSID or ECS) or an arbitrary numeric value. `+noednsopt` clears the EDNS options to be sent.

`+[no]expire`

This option sends an EDNS Expire option.

`+[no]fail`

This option indicates that named should try [or not try] the next server if a SERVFAIL is received. The default is to not try the next server, which is the reverse of normal stub resolver behavior.

`+[no]header-only`

This option sends a query with a DNS header without a question section. The default is to add a question section. The query type and query name are ignored when this is set.

`+[no]identify`

This option shows [or does not show] the IP address and port number that supplied the answer, when the `+short` option is enabled. If short form answers are requested, the default is not to show the source address and port number of the server that provided the answer.

`+[no]idnin`

This option processes [or does not process] IDN domain names on input. This requires IDN SUPPORT to have been enabled at compile time.

The default is to process IDN input when standard output is a tty. The IDN processing on input is disabled when dig output is redirected to files, pipes, and other non-tty file descriptors.

`+[no]idnout`

This option converts [or does not convert] puny code on output.

This requires IDN SUPPORT to have been enabled at compile time.

The default is to process puny code on output when standard output is a tty. The puny code processing on output is disabled when dig output is redirected to files, pipes, and other non-tty file descriptors.

`+[no]ignore`

This option ignores [or does not ignore] truncation in UDP re?

sponses instead of retrying with TCP. By default, TCP retries are performed.

`+[no]keepalive`

This option sends [or does not send] an EDNS Keepalive option.

`+[no]keepopen`

This option keeps [or does not keep] the TCP socket open between queries, and reuses it rather than creating a new TCP socket for each lookup. The default is `+nokeepopen`.

`+[no]mapped`

This option allows [or does not allow] mapped IPv4-over-IPv6 addresses to be used. The default is `+mapped`.

`+[no]multiline`

This option prints [or does not print] records, like the SOA records, in a verbose multi-line format with human-readable comments. The default is to print each record on a single line to facilitate machine parsing of the dig output.

`+ndots=D`

This option sets the number of dots (D) that must appear in name for it to be considered absolute. The default value is that defined using the `ndots` statement in `/etc/resolv.conf`, or 1 if no `ndots` statement is present. Names with fewer dots are interpreted as relative names, and are searched for in the domains listed in the `search` or `domain` directive in `/etc/resolv.conf` if `+search` is set.

`+[no]nsid`

When enabled, this option includes an EDNS name server ID request when sending a query.

`+[no]nssearch`

When this option is set, dig attempts to find the authoritative name servers for the zone containing the name being looked up, and display the SOA record that each name server has for the zone. Addresses of servers that did not respond are also printed.

`+[no]onesoa`

When enabled, this option prints only one (starting) SOA record when performing an AXFR. The default is to print both the starting and ending SOA records.

`+[no]opcode=value`

When enabled, this option sets (restores) the DNS message opcode to the specified value. The default value is QUERY (0).

`+padding=value`

This option pads the size of the query packet using the EDNS Padding option to blocks of value bytes. For example, `+padding=32` causes a 48-byte query to be padded to 64 bytes. The default block size is 0, which disables padding; the maximum is 512. Values are ordinarily expected to be powers of two, such as 128; however, this is not mandatory. Responses to padded queries may also be padded, but only if the query uses TCP or DNS COOKIE.

`+[no]qr`

This option toggles the display of the query message as it is sent. By default, the query is not printed.

`+[no]question`

This option toggles the display of the question section of a query when an answer is returned. The default is to print the question section as a comment.

`+[no]raflag`

This option sets [or does not set] the RA (Recursion Available) bit in the query. The default is `+noraflag`. This bit is ignored by the server for QUERY.

`+[no]rdflag`

This option is a synonym for `+[no]recurse`.

`+[no]recurse`

This option toggles the setting of the RD (recursion desired) bit in the query. This bit is set by default, which means dig normally sends recursive queries. Recursion is automatically

disabled when the `+nssearch` or `+trace` query option is used.

`+retry=T`

This option sets the number of times to retry UDP and TCP queries to server to `T` instead of the default, 2. Unlike `+tries`, this does not include the initial query.

`+[no]rrcomments`

This option toggles the display of per-record comments in the output (for example, human-readable key information about DNSKEY records). The default is not to print record comments unless multiline mode is active.

`+[no]search`

This option uses [or does not use] the search list defined by the `searchlist` or `domain` directive in `resolv.conf`, if any. The search list is not used by default.

`ndots` from `resolv.conf` (default 1), which may be overridden by `+ndots`, determines whether the name is treated as relative and hence whether a search is eventually performed.

`+[no]short`

This option toggles whether a terse answer is provided. The default is to print the answer in a verbose form. This option always has a global effect; it cannot be set globally and then overridden on a per-lookup basis.

`+[no]showsearch`

This option performs [or does not perform] a search showing intermediate results.

`+[no]sigchase`

This feature is now obsolete and has been removed; use `delv` instead.

`+split=W`

This option splits long hex- or base64-formatted fields in source records into chunks of `W` characters (where `W` is rounded up to the nearest multiple of 4). `+nosplit` or `+split=0` causes fields not to be split at all. The default is 56 characters, or

44 characters when multiline mode is active.

`+[no]stats`

This option toggles the printing of statistics: when the query was made, the size of the reply, etc. The default behavior is to print the query statistics as a comment after each lookup.

`+[no]subnet=addr[/prefix-length]`

This option sends [or does not send] an EDNS CLIENT-SUBNET option with the specified IP address or network prefix.

`dig +subnet=0.0.0.0/0`, or simply `dig +subnet=0` for short, sends an EDNS CLIENT-SUBNET option with an empty address and a source prefix-length of zero, which signals a resolver that the client's address information must not be used when resolving this query.

`+[no]tcflag`

This option sets [or does not set] the TC (TrunCation) bit in the query. The default is `+notcflag`. This bit is ignored by the server for QUERY.

`+[no]tcp`

This option uses [or does not use] TCP when querying name servers. The default behavior is to use UDP unless a type any or `ixfr=N` query is requested, in which case the default is TCP.

AXFR queries always use TCP.

`+timeout=T`

This option sets the timeout for a query to T seconds. The default timeout is 5 seconds. An attempt to set T to less than 1 is silently set to 1.

`+[no]topdown`

This feature is related to `dig +sigchase`, which is obsolete and has been removed. Use `delv` instead.

`+[no]trace`

This option toggles tracing of the delegation path from the root name servers for the name being looked up. Tracing is disabled by default. When tracing is enabled, `dig` makes iterative queries

to resolve the name being looked up. It follows referrals from the root servers, showing the answer from each server that was used to resolve the lookup.

If @server is also specified, it affects only the initial query for the root zone name servers.

+dnssec is also set when +trace is set, to better emulate the default queries from a name server.

+tries=T

This option sets the number of times to try UDP and TCP queries to server to T instead of the default, 3. If T is less than or equal to zero, the number of tries is silently rounded up to 1.

+trusted-key=####

This option formerly specified trusted keys for use with dig +sigchase. This feature is now obsolete and has been removed; use delv instead.

+[no]ttlid

This option displays [or does not display] the TTL when printing the record.

+[no]ttlunits

This option displays [or does not display] the TTL in friendly human-readable time units of s, m, h, d, and w, representing seconds, minutes, hours, days, and weeks. This implies +ttlid.

+[no]unexpected

This option accepts [or does not accept] answers from unexpected sources. By default, dig will not accept a reply from a source other than the one to which it sent the query.

+[no]unknownformat

This option prints all RDATA in unknown RR type presentation format (RFC 3597). The default is to print RDATA for known types in the type's presentation format.

+[no]vc

This option uses [or does not use] TCP when querying name servers. This alternate syntax to +[no]tcp is provided for back?

wards compatibility. The `vc` stands for "virtual circuit."

`+[no]yaml`

When enabled, this option prints the responses (and, if `+qr` is in use, also the outgoing queries) in a detailed YAML format.

`+[no]zflag`

This option sets [or does not set] the last unassigned DNS header flag in a DNS query. This flag is off by default.

MULTIPLE QUERIES

The BIND 9 implementation of `dig` supports specifying multiple queries on the command line (in addition to supporting the `-f` batch file option). Each of those queries can be supplied with its own set of flags, options, and query options.

In this case, each query argument represents an individual query in the command-line syntax described above. Each consists of any of the standard options and flags, the name to be looked up, an optional query type and class, and any query options that should be applied to that query.

A global set of query options, which should be applied to all queries, can also be supplied. These global query options must precede the first tuple of name, class, type, options, flags, and query options supplied on the command line. Any global query options (except `+[no]cmd` and `+[no]short` options) can be overridden by a query-specific set of query options. For example:

```
dig +qr www.isc.org any -x 127.0.0.1 isc.org ns +noqr
```

shows how `dig` can be used from the command line to make three lookups:

an ANY query for `www.isc.org`, a reverse lookup of `127.0.0.1`, and a query for the NS records of `isc.org`. A global query option of `+qr` is applied, so that `dig` shows the initial query it made for each lookup.

The final query has a local query option of `+noqr` which means that `dig` does not print the initial query when it looks up the NS records for `isc.org`.

IDN SUPPORT

If `dig` has been built with IDN (internationalized domain name) support,

it can accept and display non-ASCII domain names. dig appropriately converts character encoding of a domain name before sending a request to a DNS server or displaying a reply from the server. To turn off IDN support, use the parameters `+noidnin` and `+noidnout`, or define the `IDN_DISABLE` environment variable.

RETURN CODES

dig return codes are:

- 0 DNS response received, including NXDOMAIN status
- 1 Usage error
- 8 Couldn't open batch file
- 9 No reply from server
- 10 Internal error

FILES

`/etc/resolv.conf`

`${HOME}/.digrc`

SEE ALSO

`delv(1)`, `host(1)`, `named(8)`, `dnssec-keygen(8)`, RFC 1035.

BUGS

There are probably too many query options.

AUTHOR

Internet Systems Consortium

COPYRIGHT

2021, Internet Systems Consortium

9.16.23-RH

DIG(1)