NAME

dig – send domain name query packets to name servers

SYNOPSIS

dig [**@***server*] *domain* [⟨*query-type*⟩] [⟨*query-class*⟩] [**+**⟨*query-option*⟩] [**–**⟨*dig-option*⟩] [*%comment*]

DESCRIPTION

Dig (domain information groper) is a flexible command line tool which can be used to gather information from the Domain Name System servers. **Dig** has two modes: simple interactive mode for a single query, and batch mode which executes a query for each in a list of several query lines. All query options are accessible from the command line.

The usual simple use of **dig** will take the form:

dig @*server domain query-type query-class*

where:

server may be either a domain name or a dot-notation Internet address. If this optional field is omitted, **dig** will attempt to use the default name server for your machine.

Note: If a domain name is specified, this will be resolved using the domain name system resolver (i.e., BIND). If your system does not support DNS, you may *have* to specify a dot-notation address. Alternatively, if there is a server at your disposal somewhere, all that is required is that `/etc/resolv.conf` be present and indicate where the default name servers reside, so that *server* itself can be resolved. See `resolver`(5) for information on `/etc/resolv.conf`. **WARNING:** Changing `/etc/resolv.conf` will affect both the standard resolver library and (potentially) several programs which use it. As an option, the user may set the environment variable `LOCALRES` to name a file which is to be used instead of `/etc/resolv.conf` (`LOCALRES` is specific to the **dig** resolver and is not referenced by the standard resolver). If the `LOCALRES` variable is not set or the specified file is not readable, then `/etc/resolv.conf` will be used.

domain is the domain name for which you are requesting information. See the **–x** option (documented in the **OTHER OPTIONS** subsection of this section) for convenient way to specify inverse address query.

query-type

is the type of information (DNS query type) that you are requesting. If omitted, the default is "*a*" (`T_A = address`). The following types are recognized:

a	`T_A`	network address
any	`T_ANY`	all/any information about specified domain
mx	`T_MX`	mail exchanger for the domain
ns	`T_NS`	name servers
soa	`T_SOA`	zone of authority record
hinfo	`T_HINFO`	host information
axfr	`T_AXFR`	zone transfer (must ask an authoritative server)
txt	`T_TXT`	arbitrary number of strings

(See RFC 1035 for the complete list.)

query-class
is the network class requested in the query. If omitted, the default is "*in*" (`C_IN = Internet`). The following classes are recognized:

in	C_IN	Internet class domain
any	C_ANY	all/any class information

(See RFC 1035 for the complete list.)

Note: "*Any*" can be used to specify a *class* and/or a *type* of query. **Dig** will parse the first occurrence of "*any*" to mean *query-type* = T_ANY. To specify *query-class* = C_ANY, you must either specify "any" twice, or set *query-class* using the **-c** option (see below).

OTHER OPTIONS

%ignored-comment
"*%*" is used to included an argument that is simply not parsed. This may be useful if running **dig** in batch mode. Instead of resolving every *@server-domain-name* in a list of queries, you can avoid the overhead of doing so, and still have the domain name on the command line as a reference. Example:

dig @128.9.0.32 %venera.isi.edu mx isi.edu

-⟨*dig option*⟩
"**-**" is used to specify an option which affects the operation of **dig**. The following options are currently available (although not guaranteed to be useful):

-x *dot-notation-address*
Convenient form to specify inverse address mapping. Instead of "**dig 32.0.9.128.in-addr.arpa**", one can simply "**dig -x 128.9.0.32**".

-f *file* File for **dig** batch mode. The file contains a list of query specifications (**dig** command lines) which are to be executed successively. Lines beginning with ';', '#', or '\n' are ignored. Other options may still appear on command line, and will be in effect for each batch query.

-T *time* Time in seconds between start of successive queries when running in batch mode. Can be used to keep two or more batch **dig** commands running roughly in sync. Default is zero.

-p *port* Port number. Query a name server listening to a non-standard port number. Default is 53.

-P[*ping-string*]
After query returns, execute a `ping`(8) command for response time comparison. This rather unelegantly makes a call to the shell. The last three lines of statistics is printed for the command:

ping -s -server_name -56 -3

If the optional "*ping_string*" is present, it replaces "**ping -s**" in the shell command.

-t *query-type*
Specify type of query. May specify either an integer value to be included in the type field or use the abbreviated mnemonic as discussed above (i.e., *mx* = T_MX).

-c *query-class*
Specify class of query. May specify either an integer value to be included in the class field or use the abbreviated mnemonic as discussed above (i.e., in = C_IN).

-k *keydir:keyname*
Sign the query with the TSIG key named keyname that is in the directory keydir.

-envsav This flag specifies that the **dig** environment (defaults, print options, etc.), after all of the arguments are parsed, should be saved to a file to become the default environment. This is useful if you do not like the standard set of defaults and do not desire to include a large number of options each time **dig** is used. The environment consists of resolver state variable flags, timeout, and retries as well as the flags detailing **dig** output (see below). If the shell environment variable LOCALDEF is set to the name of a file, this is where the default **dig** environment is saved. If not, the file "DiG.env" is created in the current working directory.

Note: LOCALDEF is specific to the **dig** resolver, and will not affect operation of the standard resolver library.

Each time **dig** is executed, it looks for "./DiG.env" or the file specified by the shell environment variable LOCALDEF. If such file exists and is readable, then the environment is restored from this file before any arguments are parsed.

-envset This flag only affects batch query runs. When "**-envset**" is specified on a line in a **dig** batch file, the **dig** environment after the arguments are parsed becomes the default environment for the duration of the batch file, or until the next line which specifies "**-envset**".

-[**no**]**stick**
This flag only affects batch query runs. It specifies that the **dig** environment (as read initially or set by "**-envset**" switch) is to be restored before each query (line) in a **dig** batch file. The default "**-nostick**" means that the **dig** environment does not stick, hence options specified on a single line in a **dig** batch file will remain in effect for subsequent lines (i.e. they are not restored to the "sticky" default).

+⟨*query-option*⟩
"+" is used to specify an option to be changed in the query packet or to change **dig** output specifics. Many of these are the same parameters accepted by nslookup(8). If an option requires a parameter, the form is as follows:

+ *keyword* [=*value*]

Most keywords can be abbreviated. Parsing of the "+" options is very simplistic — a value must not be separated from its keyword by white space. The following keywords are currently available:

Keyword	Abbrev.	Meaning [default]
[**no**]**debug**	(**deb**)	turn on/off debugging mode [**deb**]
[**no**]**d2**		turn on/off extra debugging mode [**nod2**]
[**no**]**recurse**	(**rec**)	use/don't use recursive lookup [**rec**]
retry=#	(**ret**)	set number of retries to # [4]
time=#	(**ti**)	set timeout length to # seconds [4]

[**no**]**ko**		keep open option (implies vc) [**noko**]
[**no**]**vc**		use/don't use virtual circuit [**novc**]
[**no**]**defname**	(**def**)	use/don't use default domain name [**def**]
[**no**]**search**	(**sea**)	use/don't use domain search list [**sea**]
domain=*NAME*	(**do**)	set default domain name to *NAME*
[**no**]**ignore**	(**i**)	ignore/don't ignore trunc. errors [**noi**]
[**no**]**primary**	(**pr**)	use/don't use primary server [**nopr**]
[**no**]**aaonly**	(**aa**)	authoritative query only flag [**noaa**]
[**no**]**cmd**		echo parsed arguments [**cmd**]
[**no**]**stats**	(**st**)	print query statistics [**st**]
[**no**]**Header**	(**H**)	print basic header [**H**]
[**no**]**header**	(**he**)	print header flags [**he**]
[**no**]**ttlid**	(**tt**)	print TTLs [**tt**]
[**no**]**cl**		print class info [**nocl**]
[**no**]**qr**		print outgoing query [**noqr**]
[**no**]**reply**	(**rep**)	print reply [**rep**]
[**no**]**ques**	(**qu**)	print question section [**qu**]
[**no**]**answer**	(**an**)	print answer section [**an**]
[**no**]**author**	(**au**)	print authoritative section [**au**]
[**no**]**addit**	(**ad**)	print additional section [**ad**]
pfdef		set to default print flags
pfmin		set to minimal default print flags
pfset=*#*		set print flags to # (# can be hex/octal/decimal)
pfand=*#*		bitwise and print flags with #
pfor=*#*		bitwise or print flags with #

The **retry** and **time** options affect the retransmission strategy used by the resolver library when sending datagram queries. The algorithm is as follows:

```
for i = 0 to retry - 1
    for j = 1 to num_servers
        send_query
        wait((time * (2**i)) / num_servers)
    end
end
```

(Note: **dig** always uses a value of 1 for "num_servers".)

DETAILS

Dig once required a slightly modified version of the BIND resolver(3) library. As of BIND 4.9, BIND's resolver has been augmented to work properly with **dig**. Essentially, **dig** is a straight-forward (albeit not pretty) effort of parsing arguments and setting appropriate parameters. **Dig** uses resolver(3) routines **res_init**(), **res_mkquery**(), **res_send**() as well as accessing the *_res* structure.

ENVIRONMENT

LOCALRES	file to use in place of Pa /etc/resolv.conf
LOCALDEF	default environment file

See also the explanation of the **-envsav**, **-envset**, and **-**[**no**] **stick** options, above.

FILES

`/etc/resolv.conf`	initial domain name and name server addresses
`./DiG.env`	default save file for default options

SEE ALSO

`named`(8), `resolver`(3), `resolver`(5), `nslookup`(8).

STANDARDS

RFC 1035.

AUTHOR

Steve Hotz hotz@isi.edu

ACKNOWLEDGMENTS

`Dig` uses functions from `nslookup`(8) authored by Andrew Cherenson.

BUGS

`Dig` has a serious case of "creeping featurism" -- the result of considering several potential uses during it's development. It would probably benefit from a rigorous diet. Similarly, the print flags and granularity of the items they specify make evident their rather ad hoc genesis.

`Dig` does not consistently exit nicely (with appropriate status) when a problem occurs somewhere in the resolver (**NOTE:** most of the common exit cases are handled). This is particularly annoying when running in batch mode. If it exits abnormally (and is not caught), the entire batch aborts; when such an event is trapped, **`dig`** simply continues with the next query.