

**NAME**

ldns-testns - simple fake nameserver tool

**SYNOPSIS**

**ldns-testns** [ *OPTION* ] *datafile*

**DESCRIPTION**

**ldns-testns** can be used to provide answers to DNS queries for testing. The answers are premade, and can be tailored to testing needs. The answers can be wildly invalid or unparsable.

This program is a debugging aid. It is not efficient, especially with a long config file, but it can give any reply to any query. This can help the developer pre-script replies for queries.

It listens to IP4 UDP and TCP by default. You can specify a packet RR by RR with header flags to return.

ldns-testns is not meant for production use.

**OPTIONS**

- r** Listens to a random port. The port number is printed to stdout.
  
- p** *port*  
Listens to the specified port.
  
- f** *num*  
Forks this number of additional instances that serve the same ports and same datafile. They do not exit; printed is 'forked pid: <num>' and you have to kill them yourself.
  
- v** Outputs more debug information. It is possible to give this option multiple times to increase verbosity level.
  
- 6** Bind to IP6 address instead of IP4. Use together with -p.

**datafile**

The data file is read on start up. It contains queries and the packets that should be sent in answer to those queries. The data file format is explained below.

**DATA FILE FORMAT**

The data file format has ';' to denote comment. A number of entries are processed first to last. The first matching entry is used to answer the query with. This is a line based format. DNS resource records are entered in zone-file format.

You can use \$ORIGIN and \$TTL directives. Zone file '(' and ') ' to span multiple lines are not allowed.

```
$ORIGIN origin $TTL default_ttl
```

**ENTRY\_BEGIN**

; first give MATCH lines, that say what queries are matched ; by this entry. ; 'opcode' makes the query match the opcode from the reply ; if you leave it out, any opcode matches this entry. ; 'qtype' makes the query match the qtype from the reply ; 'qname' makes the query match the qname from the reply ; 'serial=1023' makes the query match if ixfr serial is 1023.

```
MATCH [opcode] [qtype] [qname] [serial=<value>] MATCH [UDP|TCP] MATCH ...
```

; Then the REPLY header is specified.

```
REPLY opcode, rcode or flags.
```

```
(opcode) QUERY IQUERY STATUS NOTIFY UPDATE
(rcode) NOERROR FORMERR SERVFAIL NXDOMAIN NOTIMPL YXDOMAIN
        YXRRSET NXRRSET NOTAUTH NOTZONE
(flags) QR AA TC RD CD RA AD
```

```
REPLY ...
```

; any additional actions to do.

```
ADJUST copy_id ; 'copy_id' copies the ID from the query to the answer.
```

```
; 'sleep=10' sleeps for 10 seconds before giving the answer (TCP is open)
```

```
ADJUST [sleep=<num>] ; sleep before giving any reply ADJUST [packet_sleep=<num>] ; sleep
```

before this packet in sequence

SECTION QUESTION <RRs, one per line> ; the RRcount is determined automatically.

SECTION ANSWER <RRs, one per line>

SECTION AUTHORITY <RRs, one per line>

SECTION ADDITIONAL <RRs, one per line>

EXTRA\_PACKET ; follow with SECTION, REPLY for more packets.

HEX\_ANSWER\_BEGIN ; follow with hex data

; this replaces any answer packet constructed

; with the SECTION keywords (only SECTION QUERY

; is used to match queries). If the data cannot

; be parsed, ADJUST rules for the answer packet

; are ignored

HEX\_ANSWER\_END

ENTRY\_END

## **AUTHOR**

Written by the ldns team as an example for ldns usage, and for testing purposes.

## **REPORTING BUGS**

Report bugs to <ldns-team@nlnetlabs.nl>.

## **COPYRIGHT**

Copyright (C) 2006-2008 NLnet Labs. This is free software. There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.