

NAME

ldns_zone, ldns_zone_new, ldns_zone_free, ldns_zone_deep_free, ldns_zone_new_frm_fp, ldns_zone_new_frm_fp_l, ldns_zone_print, ldns_zone_print_fmt - ldns_zone creation, destruction and printing

SYNOPSIS

```
#include <stdint.h>
#include <stdbool.h>

#include <ldns/ldns.h>

ldns_zone * ldns_zone_new(void);

void ldns_zone_free(ldns_zone *zone);

void ldns_zone_deep_free(ldns_zone *zone);

ldns_status ldns_zone_new_frm_fp(ldns_zone **z, FILE *fp, const ldns_rdf *origin, uint32_t ttl,
ldns_rr_class c);

ldns_status ldns_zone_new_frm_fp_l(ldns_zone **z, FILE *fp, const ldns_rdf *origin, uint32_t ttl,
ldns_rr_class c, int *line_nr);

void ldns_zone_print(FILE *output, const ldns_zone *z);

void ldns_zone_print_fmt(FILE *output, const ldns_output_format *fmt, const ldns_zone *z);
```

DESCRIPTION

ldns_zone

DNS Zone

A list of RR's with some extra information which comes from the SOA RR

Note: nothing has been done to make this efficient (yet).

struct ldns_struct_zone

```
{
```

the soa defines a zone:

```
ldns_rr *_soa;
```

```
/* basically a zone is a list of rr's */
```

```

    ldns_rr_list    *_rrs;
    /* we could change this to be a b-tree etc etc todo */
};
typedef struct ldns_struct_zone ldns_zone;

```

ldns_zone_new() create a new *ldns_zone* structure

Returns a pointer to a *ldns_zone* structure

ldns_zone_free() Frees the allocated memory for the zone, and the *rr_list* structure in it

zone: the zone to free

ldns_zone_deep_free() Frees the allocated memory for the zone, the soa rr in it, and the *rr_list* structure in it, including the rr's in that. etc.

zone: the zone to free

ldns_zone_new_frm_fp() Create a new zone from a file

z: the new zone

***fp:** the filepointer to use

***origin:** the zones' origin

ttl: default ttl to use

c: default class to use (IN)

Returns *ldns_status* mesg with an error or LDNS_STATUS_OK

ldns_zone_new_frm_fp_l() Create a new zone from a file, keep track of the line numbering

z: the new zone

***fp:** the filepointer to use

***origin:** the zones' origin

ttl: default ttl to use

c: default class to use (IN)

line_nr: used for error msg, to get to the line number

Returns *ldns_status* mesg with an error or LDNS_STATUS_OK

ldns_zone_print() Print a zone structure * to output. Note the SOA record is included in this output

output: the fd to print to

z: the zone to print

ldns_zone_print_fmt() Print a zone structure * to output. Note the SOA record is included in this output

output: the fd to print to

fmt: format of the textual representation

z: the zone to print

AUTHOR

The ldns team at NLnet Labs.

REPORTING BUGS

Please report bugs to ldns-team@nlnetlabs.nl or in our bugzilla at <http://www.nlnetlabs.nl/bugs/index.html>

COPYRIGHT

Copyright (c) 2004 - 2006 NLnet Labs.

Licensed under the BSD License. There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

SEE ALSO

`perldoc Net::DNS`, **RFC1034**, **RFC1035**, **RFC4033**, **RFC4034** and **RFC4035**.

REMARKS

This manpage was automatically generated from the ldns source code.