



















---

4.4.1.7	HAIL_WARN . . . . .	32
4.5	include/ncld.h File Reference . . . . .	33
4.5.1	Function Documentation . . . . .	34
4.5.1.1	ncld_close . . . . .	34
4.5.1.2	ncld_del . . . . .	34
4.5.1.3	ncld_get . . . . .	34
4.5.1.4	ncld_get_meta . . . . .	34
4.5.1.5	ncld_init . . . . .	34
4.5.1.6	ncld_open . . . . .	34
4.5.1.7	ncld_qlock . . . . .	34
4.5.1.8	ncld_read_free . . . . .	34
4.5.1.9	ncld_sess_close . . . . .	34
4.5.1.10	ncld_sess_open . . . . .	34
4.5.1.11	ncld_trylock . . . . .	34
4.5.1.12	ncld_unlock . . . . .	34
4.5.1.13	ncld_write . . . . .	34



# Chapter 1

## Data Structure Index

### 1.1 Data Structures

Here are the data structures with brief descriptions:

<a href="#">cld_dirent_cur</a>	5
<a href="#">cld_timer</a>	6
<a href="#">cld_timer_list</a>	7
<a href="#">cldc_call_opts</a> (Per-operation application options )	8
<a href="#">cldc_fh</a> (Open file handle associated with a session )	9
<a href="#">cldc_host</a> (Information for a single CLD server host )	10
<a href="#">cldc_msg</a> (Outgoing message, from client to server )	11
<a href="#">cldc_node_metadata</a>	12
<a href="#">cldc_ops</a> (Application-supplied facilities )	13
<a href="#">cldc_pkt_info</a>	14
<a href="#">cldc_session</a> (Single CLD client session )	15
<a href="#">cldc_udp</a> (A UDP implementation of the CLD client protocol )	17
<a href="#">hail_log</a>	18
<a href="#">nclد_fh</a>	19
<a href="#">nclد_read</a>	20
<a href="#">nclد_sess</a>	21



# Chapter 2

## File Index

### 2.1 File List

Here is a list of all files with brief descriptions:

<a href="#">include/cld-private.h</a>	23
<a href="#">include/cld_common.h</a>	24
<a href="#">include/cldc.h</a>	26
<a href="#">include/hail_log.h</a>	31
<a href="#">include/ncld.h</a>	33



## Chapter 3

# Data Structure Documentation

### 3.1 cld\_dirent\_cur Struct Reference

```
#include <cldc.h>
```

#### Data Fields

- const void \* [p](#)
- size\_t [tmp\\_len](#)

#### 3.1.1 Field Documentation

**3.1.1.1** const void\* cld\_dirent\_cur::p

**3.1.1.2** size\_t cld\_dirent\_cur::tmp\_len

The documentation for this struct was generated from the following file:

- include/[cldc.h](#)

## 3.2 cld\_timer Struct Reference

```
#include <cld_common.h>
```

### Data Fields

- bool [fired](#)
- bool [on\\_list](#)
- void(\* [cb](#))(struct [cld\\_timer](#) \*)
- void \* [userdata](#)
- time\_t [expires](#)
- char [name](#) [32]

### 3.2.1 Field Documentation

**3.2.1.1** void(\* [cld\\_timer::cb](#))(struct [cld\\_timer](#) \*)

**3.2.1.2** time\_t [cld\\_timer::expires](#)

**3.2.1.3** bool [cld\\_timer::fired](#)

**3.2.1.4** char [cld\\_timer::name](#)[32]

**3.2.1.5** bool [cld\\_timer::on\\_list](#)

**3.2.1.6** void\* [cld\\_timer::userdata](#)

The documentation for this struct was generated from the following file:

- include/[cld\\_common.h](#)



## 3.3 cld\_timer\_list Struct Reference

```
#include <cld_common.h>
```

### Data Fields

- void \* [list](#)

### 3.3.1 Field Documentation

#### 3.3.1.1 void\* cld\_timer\_list::list

The documentation for this struct was generated from the following file:

- include/[cld\\_common.h](#)

## 3.4 cldc\_call\_opts Struct Reference

per-operation application options

```
#include <cldc.h>
```

### Data Fields

- `int(* cb)`(struct `cldc_call_opts` \*, enum `cle_err_codes`)
- `void *` `private`
- `struct cld_msg_get_resp` `resp`

### 3.4.1 Detailed Description

per-operation application options

### 3.4.2 Field Documentation

**3.4.2.1** `int(* cldc_call_opts::cb)`(struct `cldc_call_opts` \*, enum `cle_err_codes`)

**3.4.2.2** `void* cldc_call_opts::private`

**3.4.2.3** `struct cld_msg_get_resp cldc_call_opts::resp`

The documentation for this struct was generated from the following file:

- `include/cldc.h`

## 3.5 cldc\_fh Struct Reference

an open file handle associated with a session

```
#include <cldc.h>
```

### Data Fields

- uint64\_t [fh](#)
- struct [cldc\\_session](#) \* [sess](#)
- bool [valid](#)

### 3.5.1 Detailed Description

an open file handle associated with a session

### 3.5.2 Field Documentation

**3.5.2.1**    `uint64_t cldc_fh::fh`

**3.5.2.2**    `struct cldc_session* cldc_fh::sess`

**3.5.2.3**    `bool cldc_fh::valid`

The documentation for this struct was generated from the following file:

- `include/cldc.h`

## 3.6 cldc\_host Struct Reference

Information for a single CLD server host.

```
#include <cldc.h>
```

### Data Fields

- unsigned int [prio](#)
- unsigned int [weight](#)
- char \* [host](#)
- unsigned short [port](#)

### 3.6.1 Detailed Description

Information for a single CLD server host.

### 3.6.2 Field Documentation

**3.6.2.1** char\* cldc\_host::host

**3.6.2.2** unsigned short cldc\_host::port

**3.6.2.3** unsigned int cldc\_host::prio

**3.6.2.4** unsigned int cldc\_host::weight

The documentation for this struct was generated from the following file:

- include/[cldc.h](#)

## 3.7 cldc\_msg Struct Reference

an outgoing message, from client to server

```
#include <cldc.h>
```

### Data Fields

- uint64\_t [xid](#)
- enum [cld\\_msg\\_op](#) [op](#)
- struct [cldc\\_session](#) \* [sess](#)
- ssize\_t(\* [cb](#))(struct [cldc\\_msg](#) \*, const void \*, size\_t, enum [cle\\_err\\_codes](#))
- void \* [cb\\_private](#)
- struct [cldc\\_call\\_opts](#) [copts](#)
- bool [done](#)
- time\_t [expire\\_time](#)
- int [n\\_pkts](#)
- struct [cldc\\_pkt\\_info](#) \* [pkt\\_info](#) [0]

### 3.7.1 Detailed Description

an outgoing message, from client to server

### 3.7.2 Field Documentation

**3.7.2.1** `ssize_t(* cldc_msg::cb)(struct cldc_msg *, const void *, size_t, enum cle_err_codes)`

**3.7.2.2** `void* cldc_msg::cb_private`

**3.7.2.3** `struct cldc_call_opts cldc_msg::copts`

**3.7.2.4** `bool cldc_msg::done`

**3.7.2.5** `time_t cldc_msg::expire_time`

**3.7.2.6** `int cldc_msg::n_pkts`

**3.7.2.7** `enum cld_msg_op cldc_msg::op`

**3.7.2.8** `struct cldc_pkt_info* cldc_msg::pkt_info[0]`

**3.7.2.9** `struct cldc_session* cldc_msg::sess`

**3.7.2.10** `uint64_t cldc_msg::xid`

The documentation for this struct was generated from the following file:

- [include/cldc.h](#)

## 3.8 cldc\_node\_metadata Struct Reference

```
#include <cldc.h>
```

### Data Fields

- quad\_t [inum](#)
- quad\_t [vers](#)
- quad\_t [time\\_create](#)
- quad\_t [time\\_modify](#)
- int [flags](#)
- const char \* [inode\\_name](#)

### 3.8.1 Field Documentation

**3.8.1.1** int cldc\_node\_metadata::flags

**3.8.1.2** const char\* cldc\_node\_metadata::inode\_name

**3.8.1.3** quad\_t cldc\_node\_metadata::inum

**3.8.1.4** quad\_t cldc\_node\_metadata::time\_create

**3.8.1.5** quad\_t cldc\_node\_metadata::time\_modify

**3.8.1.6** quad\_t cldc\_node\_metadata::vers

The documentation for this struct was generated from the following file:

- [include/cldc.h](#)

## 3.9 cldc\_ops Struct Reference

application-supplied facilities

```
#include <cldc.h>
```

### Data Fields

- `bool(* timer_ctl )(void *private, bool add, int(*cb)(struct cldc\_session *, void *), void *cb_private, time_t secs)`
- `int(* pkt_send )(void *private, const void *addr, size_t addrlen, const void *buf, size_t buflen)`
- `void(* event )(void *private, struct cldc\_session *, struct cldc\_fh *, uint32_t)`

### 3.9.1 Detailed Description

application-supplied facilities

### 3.9.2 Field Documentation

**3.9.2.1** `void(* cldc_ops::event)(void *private, struct cldc\_session *, struct cldc\_fh *, uint32_t)`

**3.9.2.2** `int(* cldc_ops::pkt_send)(void *private, const void *addr, size_t addrlen, const void *buf, size_t buflen)`

**3.9.2.3** `bool(* cldc_ops::timer_ctl)(void *private, bool add, int(*cb)(struct cldc\_session *, void *), void *cb_private, time_t secs)`

The documentation for this struct was generated from the following file:

- `include/cldc.h`

## 3.10 cldc\_pkt\_info Struct Reference

```
#include <cldc.h>
```

### Data Fields

- int [pkt\\_len](#)
- int [hdr\\_len](#)
- int [retries](#)
- char [user](#) [CLD\_MAX\_USERNAME]
- char [data](#) [0]

### 3.10.1 Field Documentation

**3.10.1.1** char cldc\_pkt\_info::data[0]

**3.10.1.2** int cldc\_pkt\_info::hdr\_len

**3.10.1.3** int cldc\_pkt\_info::pkt\_len

**3.10.1.4** int cldc\_pkt\_info::retries

**3.10.1.5** char cldc\_pkt\_info::user[CLD\_MAX\_USERNAME]

The documentation for this struct was generated from the following file:

- include/[cldc.h](#)



## 3.11 cldc\_session Struct Reference

a single CLD client session

```
#include <cldc.h>
```

### Data Fields

- uint8\_t [sid](#) [CLD\_SID\_SZ]
- struct [cldc\\_ops](#) \* [ops](#)
- struct [hail\\_log](#) [log](#)
- void \* [private](#)
- uint8\_t [addr](#) [64]
- size\_t [addr\\_len](#)
- GArray \* [fh](#)
- GList \* [out\\_msg](#)
- time\_t [msg\\_scan\\_time](#)
- time\_t [expire\\_time](#)
- bool [expired](#)
- uint64\_t [next\\_seqid\\_in](#)
- uint64\_t [next\\_seqid\\_in\\_tr](#)
- uint64\_t [next\\_seqid\\_out](#)
- char [user](#) [CLD\_MAX\_USERNAME]
- char [secret\\_key](#) [CLD\_MAX\_SECRET\_KEY]
- bool [confirmed](#)
- enum [cld\\_msg\\_op](#) [msg\\_buf\\_op](#)
- unsigned int [msg\\_buf\\_len](#)
- char [msg\\_buf](#) [CLD\_MAX\_MSG\_SZ]
- char [payload](#) [CLD\_MAX\_PAYLOAD\_SZ]
- char [inode\\_name\\_temp](#) [CLD\_INODE\_NAME\_MAX]

### 3.11.1 Detailed Description

a single CLD client session

### 3.11.2 Field Documentation

- 3.11.2.1 `uint8_t cldc_session::addr[64]`
- 3.11.2.2 `size_t cldc_session::addr_len`
- 3.11.2.3 `bool cldc_session::confirmed`
- 3.11.2.4 `time_t cldc_session::expire_time`
- 3.11.2.5 `bool cldc_session::expired`
- 3.11.2.6 `GArray* cldc_session::fh`
- 3.11.2.7 `char cldc_session::inode_name_temp[CLD_INODE_NAME_MAX]`
- 3.11.2.8 `struct hail_log cldc_session::log`
- 3.11.2.9 `char cldc_session::msg_buf[CLD_MAX_MSG_SZ]`
- 3.11.2.10 `unsigned int cldc_session::msg_buf_len`
- 3.11.2.11 `enum cld_msg_op cldc_session::msg_buf_op`
- 3.11.2.12 `time_t cldc_session::msg_scan_time`
- 3.11.2.13 `uint64_t cldc_session::next_seqid_in`
- 3.11.2.14 `uint64_t cldc_session::next_seqid_in_tr`
- 3.11.2.15 `uint64_t cldc_session::next_seqid_out`
- 3.11.2.16 `struct cldc_ops* cldc_session::ops`
- 3.11.2.17 `GList* cldc_session::out_msg`
- 3.11.2.18 `char cldc_session::payload[CLD_MAX_PAYLOAD_SZ]`
- 3.11.2.19 `void* cldc_session::private`
- 3.11.2.20 `char cldc_session::secret_key[CLD_MAX_SECRET_KEY]`
- 3.11.2.21 `uint8_t cldc_session::sid[CLD_SID_SZ]`
- 3.11.2.22 `char cldc_session::user[CLD_MAX_USERNAME]`

The documentation for this struct was generated from the following file:

- `include/cldc.h`

## 3.12 cldc\_udp Struct Reference

A UDP implementation of the CLD client protocol.

```
#include <cldc.h>
```

### Data Fields

- `uint8_t addr` [64]
- `size_t addr_len`
- `int fd`
- `struct cldc_session * sess`
- `int(* cb)(struct cldc_session *, void *)`
- `void * cb_private`

### 3.12.1 Detailed Description

A UDP implementation of the CLD client protocol.

### 3.12.2 Field Documentation

**3.12.2.1** `uint8_t cldc_udp::addr[64]`

**3.12.2.2** `size_t cldc_udp::addr_len`

**3.12.2.3** `int(* cldc_udp::cb)(struct cldc_session *, void *)`

**3.12.2.4** `void* cldc_udp::cb_private`

**3.12.2.5** `int cldc_udp::fd`

**3.12.2.6** `struct cldc_session* cldc_udp::sess`

The documentation for this struct was generated from the following file:

- `include/cldc.h`

## 3.13 hail\_log Struct Reference

```
#include <hail_log.h>
```

### Data Fields

- void(\* [func](#) )(int prio, const char \*fmt,...) ATTR\_PRINTF(2
- void(\*) boo [debug](#) )
- bool [verbose](#)

### 3.13.1 Field Documentation

**3.13.1.1 void(\*) boo hail\_log::debug)**

**3.13.1.2 void(\* hail\_log::func)(int prio, const char \*fmt,...) ATTR\_PRINTF(2**

**3.13.1.3 bool hail\_log::verbose**

The documentation for this struct was generated from the following file:

- include/[hail\\_log.h](#)

## 3.14 ncld\_fh Struct Reference

```
#include <ncld.h>
```

### Data Fields

- struct [ncld\\_sess](#) \* [sess](#)
- struct [cldc\\_fh](#) \* [fh](#)
- bool [is\\_open](#)
- int [errc](#)
- int [nios](#)
- unsigned int [event\\_mask](#)
- void(\* [event\\_func](#))(void \*, unsigned int)
- void \* [event\\_arg](#)

### 3.14.1 Field Documentation

**3.14.1.1 int ncld\_fh::errc**

**3.14.1.2 void\* ncld\_fh::event\_arg**

**3.14.1.3 void(\* ncld\_fh::event\_func)(void \*, unsigned int)**

**3.14.1.4 unsigned int ncld\_fh::event\_mask**

**3.14.1.5 struct cldc\_fh\* ncld\_fh::fh**

**3.14.1.6 bool ncld\_fh::is\_open**

**3.14.1.7 int ncld\_fh::nios**

**3.14.1.8 struct ncld\_sess\* ncld\_fh::sess**

The documentation for this struct was generated from the following file:

- include/[ncld.h](#)

## 3.15 nclد\_read Struct Reference

```
#include <nclد.h>
```

### Data Fields

- const void \* [ptr](#)
- long [length](#)
- struct [cldc\\_node\\_metadata](#) [meta](#)
- struct [nclد\\_fh](#) \* [fh](#)
- bool [is\\_done](#)
- int [errc](#)

### 3.15.1 Field Documentation

**3.15.1.1** int nclد\_read::errc

**3.15.1.2** struct nclد\_fh\* nclد\_read::fh

**3.15.1.3** bool nclد\_read::is\_done

**3.15.1.4** long nclد\_read::length

**3.15.1.5** struct cldc\_node\_metadata nclد\_read::meta

**3.15.1.6** const void\* nclد\_read::ptr

The documentation for this struct was generated from the following file:

- [include/nclد.h](#)

## 3.16 ncld\_sess Struct Reference

```
#include <ncld.h>
```

### Data Fields

- char \* [host](#)
- unsigned short [port](#)
- GMutex \* [mutex](#)
- GCond \* [cond](#)
- GThread \* [thread](#)
- bool [is\\_up](#)
- bool [open\\_done](#)
- int [errc](#)
- GList \* [handles](#)
- int [to\\_thread](#) [2]
- struct [cldc\\_udp](#) \* [udp](#)
- struct [cld\\_timer\\_udp\\_timer](#)
- struct [cld\\_timer\\_list](#) [tlist](#)
- void(\* [event](#) )(void \*, unsigned int)
- void \* [event\\_arg](#)

### 3.16.1 Field Documentation

- 3.16.1.1 `GCond* ncld_sess::cond`
- 3.16.1.2 `int ncld_sess::errc`
- 3.16.1.3 `void(* ncld_sess::event)(void *, unsigned int)`
- 3.16.1.4 `void* ncld_sess::event_arg`
- 3.16.1.5 `GList* ncld_sess::handles`
- 3.16.1.6 `char* ncld_sess::host`
- 3.16.1.7 `bool ncld_sess::is_up`
- 3.16.1.8 `GMutex* ncld_sess::mutex`
- 3.16.1.9 `bool ncld_sess::open_done`
- 3.16.1.10 `unsigned short ncld_sess::port`
- 3.16.1.11 `GThread* ncld_sess::thread`
- 3.16.1.12 `struct cld_timer_list ncld_sess::tlist`
- 3.16.1.13 `int ncld_sess::to_thread[2]`
- 3.16.1.14 `struct cldc_udp* ncld_sess::udp`
- 3.16.1.15 `struct cld_timer ncld_sess::udp_timer`

The documentation for this struct was generated from the following file:

- [include/ncld.h](#)



## Chapter 4

# File Documentation

### 4.1 include/cld-private.h File Reference

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

```
#include <glib.h>
```

## 4.2 include/cld\_common.h File Reference

```
#include <stdint.h>
#include <stdbool.h>
#include <string.h>
#include <time.h>
#include <openssl/sha.h>
#include <cld_msg_rpc.h>
```

### Data Structures

- struct [cld\\_timer\\_list](#)
- struct [cld\\_timer](#)

### Defines

- #define [CLD\\_ALIGN8](#)(n) ((8 - ((n) & 7)) & 7)
- #define [SIDFMT](#) "%016lX"
- #define [SIDARG](#)(sid) cld\_sid2llu(sid)
- #define [CLD\\_PKT\\_FTR\\_LEN](#) sizeof(struct cld\_pkt\_ftr)  
*Length of the packet footer.*
- #define [PKT\\_HDR\\_TO\\_STR\\_SCRATCH\\_LEN](#) 128

### Functions

- void [cld\\_timer\\_add](#) (struct [cld\\_timer\\_list](#) \*tlist, struct [cld\\_timer](#) \*timer, time\_t expires)
- void [cld\\_timer\\_del](#) (struct [cld\\_timer\\_list](#) \*tlist, struct [cld\\_timer](#) \*timer)
- time\_t [cld\\_timers\\_run](#) (struct [cld\\_timer\\_list](#) \*tlist)
- unsigned long long [cld\\_sid2llu](#) (const uint8\_t \*sid)
- void [\\_\\_cld\\_rand64](#) (void \*p)
- const char \* [cld\\_errstr](#) (enum cle\_err\_codes ecode)
- int [cld\\_readport](#) (const char \*fname)
- int [\\_\\_cld\\_authcheck](#) (struct [hail\\_log](#) \*log, const char \*key, const void \*buf, size\_t buf\_len, const void \*sha)
- int [\\_\\_cld\\_authsign](#) (struct [hail\\_log](#) \*log, const char \*key, const void \*buf, size\_t buf\_len, void \*sha)
- const char \* [\\_\\_cld\\_opstr](#) (enum cld\_msg\_op)
- const char \* [\\_\\_cld\\_pkt\\_hdr\\_to\\_str](#) (char \*scratch, const char \*pkt\_hdr, size\_t pkt\_len)
- void [\\_\\_cld\\_dump\\_buf](#) (const void \*buf, size\_t len)
- struct [\\_\\_attribute\\_\\_](#) ((packed)) cld\_pkt\_ftr  
*Footer that appears at the end of each packet.*

## 4.2.1 Define Documentation

**4.2.1.1** `#define CLD_ALIGN8(n) ((8 - ((n) & 7)) & 7)`

**4.2.1.2** `#define CLD_PKT_FTR_LEN sizeof(struct cld_pkt_ftr)`

Length of the packet footer. This size is fixed

**4.2.1.3** `#define PKT_HDR_TO_STR_SCRATCH_LEN 128`

**4.2.1.4** `#define SIDARG(sid) cld_sid2llu(sid)`

**4.2.1.5** `#define SIDFMT "%016llx"`

## 4.2.2 Function Documentation

**4.2.2.1** `struct __attribute__((packed)) [read]`

Footer that appears at the end of each packet.

< packet sequence ID

< packet signature

**4.2.2.2** `int __cld_authcheck (struct hail_log * log, const char * key, const void * buf, size_t buf_len, const void * sha)`

**4.2.2.3** `int __cld_authsign (struct hail_log * log, const char * key, const void * buf, size_t buf_len, void * sha)`

**4.2.2.4** `void __cld_dump_buf (const void * buf, size_t len)`

**4.2.2.5** `const char* __cld_opstr (enum cld_msg_op)`

**4.2.2.6** `const char* __cld_pkt_hdr_to_str (char * scratch, const char * pkt_hdr, size_t pkt_len)`

**4.2.2.7** `void __cld_rand64 (void * p)`

**4.2.2.8** `const char* cld_errstr (enum cle_err_codes ecode)`

**4.2.2.9** `int cld_readport (const char * fname)`

**4.2.2.10** `unsigned long long cld_sid2llu (const uint8_t * sid)`

**4.2.2.11** `void cld_timer_add (struct cld_timer_list * tlist, struct cld_timer * timer, time_t expires)`

**4.2.2.12** `void cld_timer_del (struct cld_timer_list * tlist, struct cld_timer * timer)`

**4.2.2.13** `time_t cld_timers_run (struct cld_timer_list * tlist)`

## 4.3 include/cldc.h File Reference

```
#include <sys/types.h>
#include <stdbool.h>
#include <glib.h>
#include <cld_msg_rpc.h>
#include <cld_common.h>
#include <hail_log.h>
```

### Data Structures

- struct [cldc\\_call\\_opts](#)  
*per-operation application options*
- struct [cldc\\_node\\_metadata](#)
- struct [cldc\\_pkt\\_info](#)
- struct [cldc\\_msg](#)  
*an outgoing message, from client to server*
- struct [cldc\\_fh](#)  
*an open file handle associated with a session*
- struct [cldc\\_ops](#)  
*application-supplied facilities*
- struct [cldc\\_session](#)  
*a single CLD client session*
- struct [cldc\\_host](#)  
*Information for a single CLD server host.*
- struct [cldc\\_udp](#)  
*A UDP implementation of the CLD client protocol.*
- struct [cld\\_dirent\\_cur](#)

### Functions

- int [cldc\\_receive\\_pkt](#) (struct [cldc\\_session](#) \*sess, const void \*net\_addr, size\_t net\_addrlen, const void \*buf, size\_t buflen)  
*Packet received from remote host.*
- void [cldc\\_init](#) (void)
- int [cldc\\_new\\_sess](#) (const struct [cldc\\_ops](#) \*ops, const struct [cldc\\_call\\_opts](#) \*copts, const void \*addr, size\_t addr\_len, const char \*user, const char \*secret\_key, void \*private, struct [cldc\\_session](#) \*\*sess\_out)
- void [cldc\\_kill\\_sess](#) (struct [cldc\\_session](#) \*sess)

- int `cldc_end_sess` (struct `cldc_session` \*sess, const struct `cldc_call_opts` \*copts)
- int `cldc_nop` (struct `cldc_session` \*sess, const struct `cldc_call_opts` \*copts)
- int `cldc_del` (struct `cldc_session` \*sess, const struct `cldc_call_opts` \*copts, const char \*pathname)
- int `cldc_open` (struct `cldc_session` \*sess, const struct `cldc_call_opts` \*copts, const char \*pathname, uint32\_t open\_mode, uint32\_t events, struct `cldc_fh` \*\*fh\_out)
- int `cldc_close` (struct `cldc_fh` \*fh, const struct `cldc_call_opts` \*copts)
- int `cldc_unlock` (struct `cldc_fh` \*fh, const struct `cldc_call_opts` \*copts)
- int `cldc_lock` (struct `cldc_fh` \*fh, const struct `cldc_call_opts` \*copts, uint32\_t lock\_flags, bool wait\_for\_lock)
- int `cldc_put` (struct `cldc_fh` \*fh, const struct `cldc_call_opts` \*copts, const void \*data, size\_t data\_len)
- int `cldc_get` (struct `cldc_fh` \*fh, const struct `cldc_call_opts` \*copts, bool metadata\_only)
- int `cldc_dirent_count` (const void \*data, size\_t data\_len)
- int `cldc_dirent_first` (struct `cld_dirent_cur` \*dc)
- int `cldc_dirent_next` (struct `cld_dirent_cur` \*dc)
- void `cldc_dirent_cur_init` (struct `cld_dirent_cur` \*dc, const void \*buf, size\_t buflen)
- void `cldc_dirent_cur_fini` (struct `cld_dirent_cur` \*dc)
- char \* `cldc_dirent_name` (struct `cld_dirent_cur` \*dc)
- void `cldc_copts_get_data` (const struct `cldc_call_opts` \*copts, char \*\*data, size\_t \*data\_len)
- void `cldc_copts_get_metadata` (const struct `cldc_call_opts` \*copts, struct `cldc_node_metadata` \*md)
- void `cldc_udp_free` (struct `cldc_udp` \*udp)
- int `cldc_udp_new` (const char \*hostname, int port, struct `cldc_udp` \*\*udp\_out)
- int `cldc_udp_receive_pkt` (struct `cldc_udp` \*udp)
- int `cldc_udp_pkt_send` (void \*private, const void \*addr, size\_t addrlen, const void \*buf, size\_t buflen)
- int `cldc_getaddr` (GList \*\*host\_list, const char \*thishost, struct `hail_log` \*log)
- int `cldc_saveaddr` (struct `cldc_host` \*hp, unsigned int priority, unsigned int weight, unsigned int port, unsigned int nlen, const char \*name, struct `hail_log` \*log)



### 4.3.1 Function Documentation

- 4.3.1.1 `int cldc_close (struct cldc_fh *fh, const struct cldc_call_opts *copts)`
- 4.3.1.2 `void cldc_copts_get_data (const struct cldc_call_opts *copts, char **data, size_t *data_len)`
- 4.3.1.3 `void cldc_copts_get_metadata (const struct cldc_call_opts *copts, struct cldc_node_metadata *md)`
- 4.3.1.4 `int cldc_del (struct cldc_session *sess, const struct cldc_call_opts *copts, const char *pathname)`
- 4.3.1.5 `int cldc_dirent_count (const void *data, size_t data_len)`
- 4.3.1.6 `void cldc_dirent_cur_fini (struct cld_dirent_cur *dc)`
- 4.3.1.7 `void cldc_dirent_cur_init (struct cld_dirent_cur *dc, const void *buf, size_t buflen)`
- 4.3.1.8 `int cldc_dirent_first (struct cld_dirent_cur *dc)`
- 4.3.1.9 `char* cldc_dirent_name (struct cld_dirent_cur *dc)`
- 4.3.1.10 `int cldc_dirent_next (struct cld_dirent_cur *dc)`
- 4.3.1.11 `int cldc_end_sess (struct cldc_session *sess, const struct cldc_call_opts *copts)`
- 4.3.1.12 `int cldc_get (struct cldc_fh *fh, const struct cldc_call_opts *copts, bool metadata_only)`
- 4.3.1.13 `int cldc_getaddr (GList **host_list, const char *thishost, struct hail_log *log)`
- 4.3.1.14 `void cldc_init (void)`
- 4.3.1.15 `void cldc_kill_sess (struct cldc_session *sess)`
- 4.3.1.16 `int cldc_lock (struct cldc_fh *fh, const struct cldc_call_opts *copts, uint32_t lock_flags, bool wait_for_lock)`
- 4.3.1.17 `int cldc_new_sess (const struct cldc_ops *ops, const struct cldc_call_opts *copts, const void *addr, size_t addr_len, const char *user, const char *secret_key, void *private, struct cldc_session **sess_out)`
- 4.3.1.18 `int cldc_nop (struct cldc_session *sess, const struct cldc_call_opts *copts)`
- 4.3.1.19 `int cldc_open (struct cldc_session *sess, const struct cldc_call_opts *copts, const char *pathname, uint32_t open_mode, uint32_t events, struct cldc_fh **fh_out)`
- 4.3.1.20 `int cldc_put (struct cldc_fh *fh, const struct cldc_call_opts *copts, const void *data, size_t data_len)`
- 4.3.1.21 `int cldc_receive_pkt (struct cldc_session *sess, const void *net_addr, size_t net_addrlen, const void *buf, size_t buflen)`

Packet received from remote host. Called by app when a packet is received from a remote host over the network

---

**Parameters:**

*sess* Session associated with received packet  
*net\_addr* Opaque network address  
*net\_addrlen* Size of opaque network address  
*buf* Pointer to data buffer containing packet  
*buflen* Length of received packet

**Returns:**

Zero for success, non-zero on error

**4.3.1.22** `int cldc_saveaddr (struct cldc_host * hp, unsigned int priority, unsigned int weight, unsigned int port, unsigned int nlen, const char * name, struct hail_log * log)`

**4.3.1.23** `void cldc_udp_free (struct cldc_udp * udp)`

**4.3.1.24** `int cldc_udp_new (const char * hostname, int port, struct cldc_udp ** udp_out)`

**4.3.1.25** `int cldc_udp_pkt_send (void * private, const void * addr, size_t addrlen, const void * buf, size_t buflen)`

**4.3.1.26** `int cldc_udp_receive_pkt (struct cldc_udp * udp)`

**4.3.1.27** `int cldc_unlock (struct cldc_fh * fh, const struct cldc_call_opts * copts)`



## 4.4 include/hail\_log.h File Reference

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

### Data Structures

- struct [hail\\_log](#)

### Defines

- #define [ATTR\\_PRINTF](#)(x, y)
- #define [HAIL\\_VERBOSE](#)(log,...)  
*Print out a CLD session debug message if enabled.*
- #define [HAIL\\_DEBUG](#)(log,...)  
*Print out an application debug message if enabled.*
- #define [HAIL\\_INFO](#)(log,...) (log)->func(LOG\_INFO, \_\_VA\_ARGS\_\_)  
*Print out an informational log message.*
- #define [HAIL\\_WARN](#)(log,...) (log)->func(LOG\_WARNING, \_\_VA\_ARGS\_\_)  
*Print out a warning message.*
- #define [HAIL\\_ERR](#)(log,...) (log)->func(LOG\_ERR, \_\_VA\_ARGS\_\_)  
*Print out an error message.*
- #define [HAIL\\_CRIT](#)(log,...) (log)->func(LOG\_CRIT, \_\_VA\_ARGS\_\_)  
*Print out a critical warning message.*

### 4.4.1 Define Documentation

#### 4.4.1.1 #define ATTR\_PRINTF(x, y)

#### 4.4.1.2 #define HAIL\_CRIT(log, ...) (log)->func(LOG\_CRIT, \_\_VA\_ARGS\_\_)

Print out a critical warning message.

#### 4.4.1.3 #define HAIL\_DEBUG(log, ...)

##### Value:

```
if ((log)->debug) { \
    (log)->func(LOG_DEBUG, __VA_ARGS__); \
}
```

Print out an application debug message if enabled.

**4.4.1.4 #define HAIL\_ERR(log, ...) (log)->func(LOG\_ERR, \_\_VA\_ARGS\_\_)**

Print out an error message.

**4.4.1.5 #define HAIL\_INFO(log, ...) (log)->func(LOG\_INFO, \_\_VA\_ARGS\_\_)**

Print out an informational log message.

**4.4.1.6 #define HAIL\_VERBOSE(log, ...)**

**Value:**

```
if ((log)->verbose) { \
    (log)->func(LOG_DEBUG, __VA_ARGS__); \
}
```

Print out a CLD session debug message if enabled.

**4.4.1.7 #define HAIL\_WARN(log, ...) (log)->func(LOG\_WARNING, \_\_VA\_ARGS\_\_)**

Print out a warning message.

## 4.5 include/ncld.h File Reference

```
#include <stdbool.h>
#include <glib.h>
#include <cldc.h>
```

### Data Structures

- struct [ncld\\_sess](#)
- struct [ncld\\_fh](#)
- struct [ncld\\_read](#)

### Functions

- struct [ncld\\_sess](#) \* [ncld\\_sess\\_open](#) (const char \*host, int port, int \*error, void(\*event)(void \*, unsigned int), void \*ev\_arg, const char \*cld\_user, const char \*cld\_key, struct [hail\\_log](#) \*log)
- struct [ncld\\_fh](#) \* [ncld\\_open](#) (struct [ncld\\_sess](#) \*s, const char \*fname, unsigned int mode, int \*error, unsigned int events, void(\*event)(void \*, unsigned int), void \*ev\_arg)
- int [ncld\\_del](#) (struct [ncld\\_sess](#) \*nsess, const char \*fname)
- struct [ncld\\_read](#) \* [ncld\\_get](#) (struct [ncld\\_fh](#) \*fh, int \*error)
- struct [ncld\\_read](#) \* [ncld\\_get\\_meta](#) (struct [ncld\\_fh](#) \*fh, int \*error)
- void [ncld\\_read\\_free](#) (struct [ncld\\_read](#) \*rp)
- int [ncld\\_write](#) (struct [ncld\\_fh](#) \*, const void \*data, long len)
- int [ncld\\_trylock](#) (struct [ncld\\_fh](#) \*)
- int [ncld\\_qlock](#) (struct [ncld\\_fh](#) \*)
- int [ncld\\_unlock](#) (struct [ncld\\_fh](#) \*)
- void [ncld\\_close](#) (struct [ncld\\_fh](#) \*)
- void [ncld\\_sess\\_close](#) (struct [ncld\\_sess](#) \*s)
- void [ncld\\_init](#) (void)

## 4.5.1 Function Documentation

4.5.1.1 void `nclد_close` (struct `nclد_fh` \*)

4.5.1.2 int `nclد_del` (struct `nclد_sess` \* *nsess*, const char \* *fname*)

4.5.1.3 struct `nclد_read`\* `nclد_get` (struct `nclد_fh` \* *fh*, int \* *error*) [read]

4.5.1.4 struct `nclد_read`\* `nclد_get_meta` (struct `nclد_fh` \* *fh*, int \* *error*) [read]

4.5.1.5 void `nclد_init` (void)

4.5.1.6 struct `nclد_fh`\* `nclد_open` (struct `nclد_sess` \* *s*, const char \* *fname*, unsigned int *mode*, int \* *error*, unsigned int *events*, void(\*) (void \*, unsigned int) *event*, void \* *ev\_arg*) [read]

4.5.1.7 int `nclد_qlock` (struct `nclد_fh` \*)

4.5.1.8 void `nclد_read_free` (struct `nclد_read` \* *rp*)

4.5.1.9 void `nclد_sess_close` (struct `nclد_sess` \* *s*)

4.5.1.10 struct `nclد_sess`\* `nclد_sess_open` (const char \* *host*, int *port*, int \* *error*, void(\*) (void \*, unsigned int) *event*, void \* *ev\_arg*, const char \* *cld\_user*, const char \* *cld\_key*, struct `hail_log` \* *log*) [read]

4.5.1.11 int `nclد_trylock` (struct `nclد_fh` \*)

4.5.1.12 int `nclد_unlock` (struct `nclد_fh` \*)

4.5.1.13 int `nclد_write` (struct `nclد_fh` \*, const void \* *data*, long *len*)

# Index

- `__attribute__`
    - `cld_common.h`, [25](#)
  - `__cld_authcheck`
    - `cld_common.h`, [25](#)
  - `__cld_authsign`
    - `cld_common.h`, [25](#)
  - `__cld_dump_buf`
    - `cld_common.h`, [25](#)
  - `__cld_opstr`
    - `cld_common.h`, [25](#)
  - `__cld_pkt_hdr_to_str`
    - `cld_common.h`, [25](#)
  - `__cld_rand64`
    - `cld_common.h`, [25](#)
- `addr`
  - `cldc_session`, [16](#)
  - `cldc_udp`, [17](#)
- `addr_len`
  - `cldc_session`, [16](#)
  - `cldc_udp`, [17](#)
- `ATTR_PRINTF`
  - `hail_log.h`, [31](#)
- `cb`
  - `cld_timer`, [6](#)
  - `cldc_call_opts`, [8](#)
  - `cldc_msg`, [11](#)
  - `cldc_udp`, [17](#)
- `cb_private`
  - `cldc_msg`, [11](#)
  - `cldc_udp`, [17](#)
- `CLD_ALIGN8`
  - `cld_common.h`, [25](#)
- `cld_common.h`
  - `__attribute__`, [25](#)
  - `__cld_authcheck`, [25](#)
  - `__cld_authsign`, [25](#)
  - `__cld_dump_buf`, [25](#)
  - `__cld_opstr`, [25](#)
  - `__cld_pkt_hdr_to_str`, [25](#)
  - `__cld_rand64`, [25](#)
  - `CLD_ALIGN8`, [25](#)
  - `cld_errstr`, [25](#)
  - `CLD_PKT_FTR_LEN`, [25](#)
  - `cld_readport`, [25](#)
  - `cld_sid2llu`, [25](#)
  - `cld_timer_add`, [25](#)
  - `cld_timer_del`, [25](#)
  - `cld_timers_run`, [25](#)
  - `PKT_HDR_TO_STR_SCRATCH_LEN`, [25](#)
  - `SIDARG`, [25](#)
  - `SIDFMT`, [25](#)
- `cld_dirent_cur`, [5](#)
  - `p`, [5](#)
  - `tmp_len`, [5](#)
- `cld_errstr`
  - `cld_common.h`, [25](#)
- `CLD_PKT_FTR_LEN`
  - `cld_common.h`, [25](#)
- `cld_readport`
  - `cld_common.h`, [25](#)
- `cld_sid2llu`
  - `cld_common.h`, [25](#)
- `cld_timer`, [6](#)
  - `cb`, [6](#)
  - `expires`, [6](#)
  - `fired`, [6](#)
  - `name`, [6](#)
  - `on_list`, [6](#)
  - `userdata`, [6](#)
- `cld_timer_add`
  - `cld_common.h`, [25](#)
- `cld_timer_del`
  - `cld_common.h`, [25](#)
- `cld_timer_list`, [7](#)
  - `list`, [7](#)
- `cld_timers_run`
  - `cld_common.h`, [25](#)
- `cldc.h`
  - `cldc_close`, [29](#)
  - `cldc_copts_get_data`, [29](#)
  - `cldc_copts_get_metadata`, [29](#)
  - `cldc_del`, [29](#)
  - `cldc_dirent_count`, [29](#)
  - `cldc_dirent_cur_fini`, [29](#)
  - `cldc_dirent_cur_init`, [29](#)
  - `cldc_dirent_first`, [29](#)
  - `cldc_dirent_name`, [29](#)
  - `cldc_dirent_next`, [29](#)

- cldc\_end\_sess, 29
- cldc\_get, 29
- cldc\_getaddr, 29
- cldc\_init, 29
- cldc\_kill\_sess, 29
- cldc\_lock, 29
- cldc\_new\_sess, 29
- cldc\_nop, 29
- cldc\_open, 29
- cldc\_put, 29
- cldc\_receive\_pkt, 29
- cldc\_saveaddr, 30
- cldc\_udp\_free, 30
- cldc\_udp\_new, 30
- cldc\_udp\_pkt\_send, 30
- cldc\_udp\_receive\_pkt, 30
- cldc\_unlock, 30
- cldc\_call\_opts, 8
  - cb, 8
  - private, 8
  - resp, 8
- cldc\_close
  - cldc.h, 29
- cldc\_copts\_get\_data
  - cldc.h, 29
- cldc\_copts\_get\_metadata
  - cldc.h, 29
- cldc\_del
  - cldc.h, 29
- cldc\_dirent\_count
  - cldc.h, 29
- cldc\_dirent\_cur\_fini
  - cldc.h, 29
- cldc\_dirent\_cur\_init
  - cldc.h, 29
- cldc\_dirent\_first
  - cldc.h, 29
- cldc\_dirent\_name
  - cldc.h, 29
- cldc\_dirent\_next
  - cldc.h, 29
- cldc\_end\_sess
  - cldc.h, 29
- cldc\_fh, 9
  - fh, 9
  - sess, 9
  - valid, 9
- cldc\_get
  - cldc.h, 29
- cldc\_getaddr
  - cldc.h, 29
- cldc\_host, 10
  - host, 10
  - port, 10
  - prio, 10
  - weight, 10
- cldc\_init
  - cldc.h, 29
- cldc\_kill\_sess
  - cldc.h, 29
- cldc\_lock
  - cldc.h, 29
- cldc\_msg, 11
  - cb, 11
  - cb\_private, 11
  - copts, 11
  - done, 11
  - expire\_time, 11
  - n\_pkts, 11
  - op, 11
  - pkt\_info, 11
  - sess, 11
  - xid, 11
- cldc\_new\_sess
  - cldc.h, 29
- cldc\_node\_metadata, 12
  - flags, 12
  - inode\_name, 12
  - inum, 12
  - time\_create, 12
  - time\_modify, 12
  - vers, 12
- cldc\_nop
  - cldc.h, 29
- cldc\_open
  - cldc.h, 29
- cldc\_ops, 13
  - event, 13
  - pkt\_send, 13
  - timer\_ctl, 13
- cldc\_pkt\_info, 14
  - data, 14
  - hdr\_len, 14
  - pkt\_len, 14
  - retries, 14
  - user, 14
- cldc\_put
  - cldc.h, 29
- cldc\_receive\_pkt
  - cldc.h, 29
- cldc\_saveaddr
  - cldc.h, 30
- cldc\_session, 15
  - addr, 16
  - addr\_len, 16
  - confirmed, 16
  - expire\_time, 16
  - expired, 16

- fh, 16
- inode\_name\_temp, 16
- log, 16
- msg\_buf, 16
- msg\_buf\_len, 16
- msg\_buf\_op, 16
- msg\_scan\_time, 16
- next\_seqid\_in, 16
- next\_seqid\_in\_tr, 16
- next\_seqid\_out, 16
- ops, 16
- out\_msg, 16
- payload, 16
- private, 16
- secret\_key, 16
- sid, 16
- user, 16
- cldc\_udp, 17
  - addr, 17
  - addr\_len, 17
  - cb, 17
  - cb\_private, 17
  - fd, 17
  - sess, 17
- cldc\_udp\_free
  - cldc.h, 30
- cldc\_udp\_new
  - cldc.h, 30
- cldc\_udp\_pkt\_send
  - cldc.h, 30
- cldc\_udp\_receive\_pkt
  - cldc.h, 30
- cldc\_unlock
  - cldc.h, 30
- cond
  - ncld\_sess, 22
- confirmed
  - cldc\_session, 16
- copts
  - cldc\_msg, 11
- data
  - cldc\_pkt\_info, 14
- debug
  - hail\_log, 18
- done
  - cldc\_msg, 11
- errc
  - ncld\_fh, 19
  - ncld\_read, 20
  - ncld\_sess, 22
- event
  - cldc\_ops, 13
  - ncld\_sess, 22
- event\_arg
  - ncld\_fh, 19
  - ncld\_sess, 22
- event\_func
  - ncld\_fh, 19
- event\_mask
  - ncld\_fh, 19
- expire\_time
  - cldc\_msg, 11
  - cldc\_session, 16
- expired
  - cldc\_session, 16
- expires
  - cld\_timer, 6
- fd
  - cldc\_udp, 17
- fh
  - cldc\_fh, 9
  - cldc\_session, 16
  - ncld\_fh, 19
  - ncld\_read, 20
- fired
  - cld\_timer, 6
- flags
  - cldc\_node\_metadata, 12
- func
  - hail\_log, 18
- HAIL\_CRIT
  - hail\_log.h, 31
- HAIL\_DEBUG
  - hail\_log.h, 31
- HAIL\_ERR
  - hail\_log.h, 31
- HAIL\_INFO
  - hail\_log.h, 32
- hail\_log, 18
  - debug, 18
  - func, 18
  - verbose, 18
- hail\_log.h
  - ATTR\_PRINTF, 31
  - HAIL\_CRIT, 31
  - HAIL\_DEBUG, 31
  - HAIL\_ERR, 31
  - HAIL\_INFO, 32
  - HAIL\_VERBOSE, 32
  - HAIL\_WARN, 32
- HAIL\_VERBOSE
  - hail\_log.h, 32
- HAIL\_WARN
  - hail\_log.h, 32

---

- handles
  - ncld\_sess, 22
- hdr\_len
  - cldc\_pkt\_info, 14
- host
  - cldc\_host, 10
  - ncld\_sess, 22
- include/cld-private.h, 23
- include/cld\_common.h, 24
- include/cldc.h, 26
- include/hail\_log.h, 31
- include/ncld.h, 33
- inode\_name
  - cldc\_node\_metadata, 12
- inode\_name\_temp
  - cldc\_session, 16
- inum
  - cldc\_node\_metadata, 12
- is\_done
  - ncld\_read, 20
- is\_open
  - ncld\_fh, 19
- is\_up
  - ncld\_sess, 22
- length
  - ncld\_read, 20
- list
  - cld\_timer\_list, 7
- log
  - cldc\_session, 16
- meta
  - ncld\_read, 20
- msg\_buf
  - cldc\_session, 16
- msg\_buf\_len
  - cldc\_session, 16
- msg\_buf\_op
  - cldc\_session, 16
- msg\_scan\_time
  - cldc\_session, 16
- mutex
  - ncld\_sess, 22
- n\_pkts
  - cldc\_msg, 11
- name
  - cld\_timer, 6
- ncld.h
  - ncld\_close, 34
  - ncld\_del, 34
  - ncld\_get, 34
  - ncld\_get\_meta, 34
  - ncld\_init, 34
  - ncld\_open, 34
  - ncld\_qlock, 34
  - ncld\_read\_free, 34
  - ncld\_sess\_close, 34
  - ncld\_sess\_open, 34
  - ncld\_trylock, 34
  - ncld\_unlock, 34
  - ncld\_write, 34
- ncld\_close
  - ncld.h, 34
- ncld\_del
  - ncld.h, 34
- ncld\_fh, 19
  - errc, 19
  - event\_arg, 19
  - event\_func, 19
  - event\_mask, 19
  - fh, 19
  - is\_open, 19
  - nios, 19
  - sess, 19
- ncld\_get
  - ncld.h, 34
- ncld\_get\_meta
  - ncld.h, 34
- ncld\_init
  - ncld.h, 34
- ncld\_open
  - ncld.h, 34
- ncld\_qlock
  - ncld.h, 34
- ncld\_read, 20
  - errc, 20
  - fh, 20
  - is\_done, 20
  - length, 20
  - meta, 20
  - ptr, 20
- ncld\_read\_free
  - ncld.h, 34
- ncld\_sess, 21
  - cond, 22
  - errc, 22
  - event, 22
  - event\_arg, 22
  - handles, 22
  - host, 22
  - is\_up, 22
  - mutex, 22
  - open\_done, 22
  - port, 22
  - thread, 22



- tlist, 22
- to\_thread, 22
- udp, 22
- udp\_timer, 22
- nclد\_sess\_close
  - nclد.h, 34
- nclد\_sess\_open
  - nclد.h, 34
- nclد\_trylock
  - nclد.h, 34
- nclد\_unlock
  - nclد.h, 34
- nclد\_write
  - nclد.h, 34
- next\_seqid\_in
  - cldc\_session, 16
- next\_seqid\_in\_tr
  - cldc\_session, 16
- next\_seqid\_out
  - cldc\_session, 16
- nios
  - nclد\_fh, 19
- on\_list
  - clد\_timer, 6
- op
  - cldc\_msg, 11
- open\_done
  - nclد\_sess, 22
- ops
  - cldc\_session, 16
- out\_msg
  - cldc\_session, 16
- p
  - clد\_dirent\_cur, 5
- payload
  - cldc\_session, 16
- PKT\_HDR\_TO\_STR\_SCRATCH\_LEN
  - clد\_common.h, 25
- pkt\_info
  - cldc\_msg, 11
- pkt\_len
  - cldc\_pkt\_info, 14
- pkt\_send
  - cldc\_ops, 13
- port
  - cldc\_host, 10
  - nclد\_sess, 22
- prio
  - cldc\_host, 10
- private
  - cldc\_call\_opts, 8
  - cldc\_session, 16
- ptr
  - nclد\_read, 20
- resp
  - cldc\_call\_opts, 8
- retries
  - cldc\_pkt\_info, 14
- secret\_key
  - cldc\_session, 16
- sess
  - cldc\_fh, 9
  - cldc\_msg, 11
  - cldc\_udp, 17
  - nclد\_fh, 19
- sid
  - cldc\_session, 16
- SIDARG
  - clد\_common.h, 25
- SIDFMT
  - clد\_common.h, 25
- thread
  - nclد\_sess, 22
- time\_create
  - cldc\_node\_metadata, 12
- time\_modify
  - cldc\_node\_metadata, 12
- timer\_ctl
  - cldc\_ops, 13
- tlist
  - nclد\_sess, 22
- tmp\_len
  - clد\_dirent\_cur, 5
- to\_thread
  - nclد\_sess, 22
- udp
  - nclد\_sess, 22
- udp\_timer
  - nclد\_sess, 22
- user
  - cldc\_pkt\_info, 14
  - cldc\_session, 16
- userdata
  - clд\_timer, 6
- valid
  - cldc\_fh, 9
- verbose
  - hail\_log, 18
- vers
  - cldc\_node\_metadata, 12
- weight

cldc\_host, [10](#)

xid

cldc\_msg, [11](#)