

## Network Compression Detection

Generated by Doxygen 1.8.6

Sun Dec 7 2014 00:40:19



# Contents

<b>1</b>	<b>NCD</b>	<b>1</b>
<b>2</b>	<b>Data Structure Index</b>	<b>3</b>
2.1	Data Structures . . . . .	3
<b>3</b>	<b>File Index</b>	<b>5</b>
3.1	File List . . . . .	5
<b>4</b>	<b>Data Structure Documentation</b>	<b>7</b>
4.1	pseudo_header Struct Reference . . . . .	7
4.1.1	Detailed Description . . . . .	7
4.1.2	Field Documentation . . . . .	7
4.1.2.1	dest . . . . .	7
4.1.2.2	len . . . . .	7
4.1.2.3	proto . . . . .	7
4.1.2.4	source . . . . .	7
4.1.2.5	zero . . . . .	8
<b>5</b>	<b>File Documentation</b>	<b>9</b>
5.1	client.c File Reference . . . . .	9
5.1.1	Detailed Description . . . . .	9
5.1.2	Function Documentation . . . . .	9
5.1.2.1	main . . . . .	9
5.2	ncd.c File Reference . . . . .	10
5.2.1	Function Documentation . . . . .	10
5.2.1.1	comp_det . . . . .	10
5.2.1.2	get_time . . . . .	10
5.2.1.3	ip_checksum . . . . .	10
5.2.1.4	main . . . . .	11
5.2.1.5	recv_data . . . . .	11
5.2.1.6	send_data . . . . .	11
5.3	ncd.h File Reference . . . . .	12
5.3.1	Macro Definition Documentation . . . . .	12

5.3.1.1	SIZE	12
5.3.2	Function Documentation	13
5.3.2.1	comp_det	13
5.3.2.2	ip_checksum	13
5.3.2.3	recv_data	13
5.3.2.4	send_data	13
5.4	ping.c File Reference	14
5.4.1	Macro Definition Documentation	14
5.4.1.1	SIZE	14
5.4.2	Function Documentation	15
5.4.2.1	debug_print	15
5.4.2.2	get_time	15
5.4.2.3	ip_checksum	15
5.4.2.4	main	15
5.4.2.5	sub_time	15
5.4.3	Variable Documentation	15
5.4.3.1	nsent	15
5.5	ping_ip.c File Reference	15
5.5.1	Macro Definition Documentation	16
5.5.1.1	SIZE	16
5.5.2	Function Documentation	16
5.5.2.1	debug_print	16
5.5.2.2	get_time	16
5.5.2.3	ip_checksum	16
5.5.2.4	main	16
5.5.2.5	sub_time	17
5.5.3	Variable Documentation	17
5.5.3.1	nsent	17
5.6	README.md File Reference	17
5.7	server.c File Reference	17
5.7.1	Detailed Description	17
5.7.2	Function Documentation	17
5.7.2.1	main	17
5.7.2.2	procs_msg	18
5.8	test.c File Reference	19
5.8.1	Macro Definition Documentation	19
5.8.1.1	SIZE	19
5.8.2	Function Documentation	19
5.8.2.1	debug_print	19
5.8.2.2	get_time	20

5.8.2.3	ip_checksum . . . . .	20
5.8.2.4	main . . . . .	20
5.8.2.5	sub_time . . . . .	20
5.8.3	Variable Documentation . . . . .	20
5.8.3.1	nsent . . . . .	20
5.9	tracert.c File Reference . . . . .	20
5.9.1	Macro Definition Documentation . . . . .	21
5.9.1.1	SIZE . . . . .	21
5.9.2	Function Documentation . . . . .	21
5.9.2.1	debug_print . . . . .	21
5.9.2.2	get_time . . . . .	21
5.9.2.3	ip_checksum . . . . .	21
5.9.2.4	main . . . . .	21
5.9.2.5	sub_time . . . . .	22
5.9.3	Variable Documentation . . . . .	22
5.9.3.1	nsent . . . . .	22
	<b>Index</b>	<b>23</b>



# Chapter 1

## NCD

### Network Compression Detection

A Project for Comp 429 @ CSUN, to detect compression along a network transmission path.

The project will proceed in phases, and incrementally build to be an application suitable to an uncoperative environment.

We will use low and high entropy data sent along the network to detect if compression occurs. If a node compresses the data, the low entropy transmission times will differ significantly from the high entropy data transmission times, and give us a reliable way to detect compression.





## Chapter 2

# Data Structure Index

### 2.1 Data Structures

Here are the data structures with brief descriptions:

<a href="#">pseudo_header</a>	
Struct for udp pseudo header . . . . .	??



## Chapter 3

# File Index

### 3.1 File List

Here is a list of all files with brief descriptions:

<a href="#">client.c</a>	.....	??
<a href="#">ncd.c</a>	.....	??
<a href="#">ncd.h</a>	.....	??
<a href="#">ping.c</a>	.....	??
<a href="#">ping_ip.c</a>	.....	??
<a href="#">server.c</a>	.....	??
<a href="#">test.c</a>	.....	??
<a href="#">tracert.c</a>	.....	??



## Chapter 4

# Data Structure Documentation

### 4.1 pseudo\_header Struct Reference

struct for udp pseudo header

```
#include <ncd.h>
```

#### Data Fields

- `u_int32_t` [source](#)
- `u_int32_t` [dest](#)
- `u_int8_t` [zero](#)
- `u_int8_t` [proto](#)
- `uint16_t` [len](#)

#### 4.1.1 Detailed Description

struct for udp pseudo header

Definition at line 32 of file `ncd.h`.

#### 4.1.2 Field Documentation

##### 4.1.2.1 `u_int32_t pseudo_header::dest`

Definition at line 34 of file `ncd.h`.

##### 4.1.2.2 `uint16_t pseudo_header::len`

Definition at line 37 of file `ncd.h`.

##### 4.1.2.3 `u_int8_t pseudo_header::proto`

Definition at line 36 of file `ncd.h`.

##### 4.1.2.4 `u_int32_t pseudo_header::source`

Definition at line 33 of file `ncd.h`.

#### 4.1.2.5 `u_int8_t pseudo_header::zero`

Definition at line 35 of file `ncd.h`.

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

- [ncd.h](#)
- [tracert.c](#)

## Chapter 5

# File Documentation

### 5.1 client.c File Reference

```
#include <ctype.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <netinet/udp.h>
#include <stdio.h>
#include <stdlib.h>
#include <arpa/inet.h>
#include <string.h>
#include <fcntl.h>
#include "icmp.h"
```

#### Functions

- `int main (int argc, char *argv[])`

*Detects compression by sending a stream of low entropy and high entropy packets, and comparing their transmission time to determine if compression occurs along the transmission path.*

#### 5.1.1 Detailed Description

##### Author

: Paul Kirth

Comp 429 Project Phase I

Definition in file [client.c](#).

#### 5.1.2 Function Documentation

##### 5.1.2.1 `int main ( int argc, char * argv[] )`

Detects compression by sending a stream of low entropy and high entropy packets, and comparing their transmission time to determine if compression occurs along the transmission path.

[1] Destination IP address [2] Number of data packets to send in each train [3] TIME\_WAIT

Definition at line 29 of file client.c.

## 5.2 ncd.c File Reference

```
#include "ncd.h"
```

### Functions

- double [get\\_time](#) (void)
- int [comp\\_det](#) (char \*address, char \*port, char hl, size\_t data\_size, size\_t num\_packets, ushort ttl, size\_t time\_wait, int n\_tail)  
*Determines if compression occurs along the current transmission path to host by sending data to a remote location.*
- int [send\\_data](#) (char \*address, char \*port\_name, char hl, size\_t data\_size, size\_t num\_packets, ushort ttl, size\_t time\_wait, int n\_tail)  
*sends data train to the end host with leading and trailing icmp timestamps*
- int [recv\\_data](#) (double \*time)  
*Receives ICMP responses from end host and records times.*
- uint16\_t [ip\\_checksum](#) (void \*vdata, size\_t length)  
*calculates the ip checksum for some buffer of size length*
- int [main](#) (int argc, char \*argv[])  
*Main function only calls comp\_detection()*

### 5.2.1 Function Documentation

**5.2.1.1** int [comp\\_det](#) ( char \* *address*, char \* *port*, char *hl*, size\_t *data\_size*, size\_t *num\_packets*, ushort *ttl*, size\_t *time\_wait*, int *n\_tail* )

Determines if compression occurs along the current transmission path to host by sending data to a remote location.

two data trains are sent each with leading and trailing ICMP timestamp messages

the first data train will be low entropy data, to encourage compression the second train will have high entropy data, which should not be compressed if the times are significantly different, we have reasonable evidence compression exists along this path.

#### Parameters

<i>address</i>	the address of the end host stored in a char array (cstring)
<i>port</i>	the port number or service name
<i>num_packets</i>	the number of packets in each data train
<i>time_wait</i>	the wait between trains

#### Returns

0 success, 1 error/failure

Definition at line 12 of file ncd.c.

**5.2.1.2** double [get\\_time](#) ( void )

Definition at line 3 of file ncd.c.

**5.2.1.3** uint16\_t [ip\\_checksum](#) ( void \* *vdata*, size\_t *length* )

calculates the ip checksum for some buffer of size length



## Parameters

<i>vdata</i>	a pointer to the buffer to be checksummed
<i>length</i>	the length in bytes of the data to be checksummed

## Returns

the IP checksum of the buffer

Definition at line 323 of file ncd.c.

## 5.2.1.4 int main ( int argc, char \* argv[] )

Main function only calls comp\_detection()

[1] Destination IP address [2] Port Number [3] High or low entropy data 'H' or 'L' [4] Size of udp data packet [5] Number of packets in UDP Data Train [6] Time to Live [7] Wait time in milliseconds [8] number of tail ICMP messages

Definition at line 391 of file ncd.c.

## 5.2.1.5 int recv\_data ( double \* time )

Receives ICMP responses from end host and records times.

## Parameters

<i>time</i>	returns the time in ms between head echo response and first processed tail echo response to a resolution of microseconds ( $10^{-6}$ sec)
-------------	---

## Returns

0 success, -1 error/failure

Definition at line 244 of file ncd.c.

## 5.2.1.6 int send\_data ( char \* address, char \* port, char hl, size\_t data\_size, size\_t num\_packets, ushort ttl, size\_t time\_wait, int n\_tail )

sends data train to the end host with leading and trailing icmp timestamps

sets up a connection using raw sockets and sends a head ICMP echo request followed by a UDP data train. After the data train is sent, a series of ICMP echo responses is sent.

## Parameters

<i>address</i>	the address of the end host stored in a char array (cstring)
<i>port</i>	the port number or service name
<i>hl</i>	the entropy of the data (either 'H' for high entropy or 'L' for low entropy)
<i>data_size</i>	the size in bytes of the data to be sent in the udp train
<i>num_packets</i>	the number of packets in each data train
<i>ttl</i>	the time to live for each packet (max size 255)
<i>time_wait</i>	the wait between ICMP tail messages
<i>n_tail</i>	the number of ICMP tail messages to be sent

## Returns

0 success, -1 error/failure

Definition at line 45 of file ncd.c.

## 5.3 ncd.h File Reference

```
#include <ctype.h>
#include <sys/types.h>
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <time.h>
#include <errno.h>
#include <sys/socket.h>
#include <netinet/udp.h>
#include <netinet/in.h>
#include <netinet/ip.h>
#include <netinet/ip_icmp.h>
#include <netdb.h>
#include <arpa/inet.h>
#include <signal.h>
#include <fcntl.h>
```

### Data Structures

- struct [pseudo\\_header](#)  
*struct for udp pseudo header*

### Macros

- #define [SIZE](#) 1500  
*maximum ip packet size*

### Functions

- int [comp\\_det](#) (char \*address, char \*port, char hl, size\_t data\_size, size\_t num\_packets, ushort ttl, size\_t time\_wait, int n\_tail)  
*Determines if compression occurs along the current transmission path to host by sending data to a remote location.*
- int [send\\_data](#) (char \*address, char \*port, char hl, size\_t data\_size, size\_t num\_packets, ushort ttl, size\_t time\_wait, int n\_tail)  
*sends data train to the end host with leading and trailing icmp timestamps*
- int [recv\\_data](#) (double \*time)  
*Receives ICMP responses from end host and records times.*
- uint16\_t [ip\\_checksum](#) (void \*vdata, size\_t length)  
*calculates the ip checksum for some buffer of size length*

#### 5.3.1 Macro Definition Documentation

##### 5.3.1.1 #define SIZE 1500

maximum ip packet size

Definition at line 26 of file ncd.h.

### 5.3.2 Function Documentation

**5.3.2.1** `int comp_det ( char * address, char * port, char hl, size_t data_size, size_t num_packets, ushort ttl, size_t time_wait, int n_tail )`

Determines if compression occurs along the current transmission path to host by sending data to a remote location.

two data trains are sent each with leading and trailing ICMP timestamp messages

the first data train will be low entropy data, to encourage compression the second train will have high entropy data, which should not be compressed if the times are significantly different, we have reasonable evidence compression exists along this path.

#### Parameters

<i>address</i>	the address of the end host stored in a char array (cstring)
<i>port</i>	the port number or service name
<i>num_packets</i>	the number of packets in each data train
<i>time_wait</i>	the wait between trains

#### Returns

0 success, 1 error/failure

Definition at line 12 of file ncd.c.

**5.3.2.2** `uint16_t ip_checksum ( void * vdata, size_t length )`

calculates the ip checksum for some buffer of size length

#### Parameters

<i>vdata</i>	a pointer to the buffer to be checksummed
<i>length</i>	the length in bytes of the data to be checksummed

#### Returns

the IP checksum of the buffer

Definition at line 323 of file ncd.c.

**5.3.2.3** `int recv_data ( double * time )`

Receives ICMP responses from end host and records times.

#### Parameters

<i>time</i>	returns the time in ms between head echo response and first processed tail echo response to a resolution of microseconds ( $10^{-6}$ sec)
-------------	---

#### Returns

0 success, -1 error/failure

Definition at line 244 of file ncd.c.

**5.3.2.4** `int send_data ( char * address, char * port, char hl, size_t data_size, size_t num_packets, ushort ttl, size_t time_wait, int n_tail )`

sends data train to the end host with leading and trailing icmp timestamps

sets up a connection using raw sockets and sends a head ICMP echo request followed by a UDP data train. After the data train is sent, a series of ICMP echo responses is sent.

#### Parameters

<i>address</i>	the address of the end host stored in a char array (cstring)
<i>port</i>	the port number or service name
<i>hl</i>	the entropy of the data (either 'H' for high entropy or 'L' for low entropy)
<i>data_size</i>	the size in bytes of the data to be sent in the udp train
<i>num_packets</i>	the number of packets in each data train
<i>ttl</i>	the time to live for each packet (max size 255)
<i>time_wait</i>	the wait between ICMP tail messages
<i>n_tail</i>	the number of ICMP tail messages to be sent

#### Returns

0 success, -1 error/failure

Definition at line 45 of file ncd.c.

## 5.4 ping.c File Reference

```
#include <stdlib.h>
#include <stdio.h>
#include <errno.h>
#include <unistd.h>
#include <sys/time.h>
#include <string.h>
#include <netinet/ip.h>
#include <netinet/ip_icmp.h>
#include <netdb.h>
```

#### Macros

- `#define SIZE 1024`

#### Functions

- double `get_time` (void)
- double `sub_time` (struct timeval \*out, struct timeval \*in)
- uint16\_t `ip_checksum` (void \*vdata, size\_t length)  
*calculates the ip checksum for some buffer of size length*
- void `debug_print` (struct icmp \*icmp, ssize\_t n)
- int `main` (int argc, char \*argv[])

#### Variables

- size\_t `nsent`

### 5.4.1 Macro Definition Documentation

#### 5.4.1.1 #define SIZE 1024

Definition at line 96 of file ping.c.

## 5.4.2 Function Documentation

### 5.4.2.1 void debug\_print ( struct icmp \* *icmp*, ssize\_t *n* )

Definition at line 199 of file ping.c.

### 5.4.2.2 double get\_time ( void )

Definition at line 19 of file ping.c.

### 5.4.2.3 uint16\_t ip\_checksum ( void \* *vdata*, size\_t *length* )

calculates the ip checksum for some buffer of size length

Parameters

<i>vdata</i>	a pointer to the buffer to be checksummed
<i>length</i>	the length in bytes of the data to be checksummed

Returns

the IP checksum of the buffer

Definition at line 38 of file ping.c.

### 5.4.2.4 int main ( int *argc*, char \* *argv*[] )

Definition at line 98 of file ping.c.

### 5.4.2.5 double sub\_time ( struct timeval \* *out*, struct timeval \* *in* )

Definition at line 28 of file ping.c.

## 5.4.3 Variable Documentation

### 5.4.3.1 size\_t *nsent*

Definition at line 92 of file ping.c.

## 5.5 ping\_ip.c File Reference

```
#include <stdlib.h>
#include <stdio.h>
#include <errno.h>
#include <unistd.h>
#include <sys/time.h>
#include <string.h>
#include <netinet/ip.h>
#include <netinet/ip_icmp.h>
#include <netdb.h>
```

## Macros

- `#define SIZE 1024`

## Functions

- double `get_time` (void)
- double `sub_time` (struct timeval \*out, struct timeval \*in)
- uint16\_t `ip_checksum` (void \*vdata, size\_t length)  
*calculates the ip checksum for some buffer of size length*
- int `main` (int argc, char \*argv[])
- void `debug_print` (struct icmp \*icmp, ssize\_t n)

## Variables

- size\_t `nsent`

### 5.5.1 Macro Definition Documentation

#### 5.5.1.1 `#define SIZE 1024`

Definition at line 98 of file ping\_ip.c.

### 5.5.2 Function Documentation

#### 5.5.2.1 void `debug_print` ( struct icmp \* *icmp*, ssize\_t *n* )

Definition at line 251 of file ping\_ip.c.

#### 5.5.2.2 double `get_time` ( void )

Definition at line 19 of file ping\_ip.c.

#### 5.5.2.3 uint16\_t `ip_checksum` ( void \* *vdata*, size\_t *length* )

calculates the ip checksum for some buffer of size length

##### Parameters

<i>vdata</i>	a pointer to the buffer to be checksummed
<i>length</i>	the length in bytes of the data to be checksummed

##### Returns

the IP checksum of the buffer

Definition at line 38 of file ping\_ip.c.

#### 5.5.2.4 int `main` ( int *argc*, char \* *argv*[] )

Definition at line 100 of file ping\_ip.c.

#### 5.5.2.5 double sub\_time ( struct timeval \* out, struct timeval \* in )

Definition at line 28 of file ping\_ip.c.

### 5.5.3 Variable Documentation

#### 5.5.3.1 size\_t nsent

Definition at line 92 of file ping\_ip.c.

## 5.6 README.md File Reference

## 5.7 server.c File Reference

```
#include <ctype.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <netinet/udp.h>
#include <stdio.h>
#include <stdlib.h>
#include <arpa/inet.h>
#include <string.h>
#include <time.h>
```

### Functions

- clock\_t [procs\\_msg](#) (size\_t num\_msg, int sockfd, struct sockaddr\_in client)
- int [main](#) (int argc, char \*argv[])

*sets up a server whos clients will send a series of data packages so that compression along the transmission path can be detected.*

#### 5.7.1 Detailed Description

Author

: Paul Kirth

Comp 429 Project Phase I

Definition in file [server.c](#).

#### 5.7.2 Function Documentation

##### 5.7.2.1 int main ( int argc, char \* argv[] )

sets up a server whos clients will send a series of data packages so that compression along the transmission path can be detected.

Definition at line 32 of file server.c.

5.7.2.2 `clock_t` `procs_msg` ( `size_t` *num\_msg*, `int` *sockfd*, `struct sockaddr_in` *client* )



## Parameters

<i>num_msg</i>	the number of messages to expect
<i>sockfd</i>	the socket file descriptor to expect communication from
<i>client</i>	a data structure to hold client information

## Returns

the time spent gathering packets in the train

Definition at line 140 of file server.c.

## 5.8 test.c File Reference

```
#include <stdlib.h>
#include <stdio.h>
#include <errno.h>
#include <unistd.h>
#include <sys/time.h>
#include <string.h>
#include <netinet/ip.h>
#include <netinet/ip_icmp.h>
#include <netdb.h>
```

## Macros

- `#define SIZE 1024`

## Functions

- double `get_time` (void)
- double `sub_time` (struct timeval \*out, struct timeval \*in)
- uint16\_t `ip_checksum` (void \*vdata, size\_t length)  
*calculates the ip checksum for some buffer of size length*
- void `debug_print` (struct icmp \*icmp, ssize\_t n)
- int `main` (int argc, char \*argv[])

## Variables

- size\_t `nsent`

### 5.8.1 Macro Definition Documentation

#### 5.8.1.1 `#define SIZE 1024`

Definition at line 96 of file test.c.

### 5.8.2 Function Documentation

#### 5.8.2.1 void `debug_print` ( struct icmp \* *icmp*, ssize\_t *n* )

Definition at line 192 of file test.c.

### 5.8.2.2 double get\_time ( void )

Definition at line 19 of file test.c.

### 5.8.2.3 uint16\_t ip\_checksum ( void \* vdata, size\_t length )

calculates the ip checksum for some buffer of size length

#### Parameters

<i>vdata</i>	a pointer to the buffer to be checksummed
<i>length</i>	the length in bytes of the data to be checksummed

#### Returns

the IP checksum of the buffer

Definition at line 38 of file test.c.

### 5.8.2.4 int main ( int argc, char \* argv[] )

Definition at line 98 of file test.c.

### 5.8.2.5 double sub\_time ( struct timeval \* out, struct timeval \* in )

Definition at line 28 of file test.c.

## 5.8.3 Variable Documentation

### 5.8.3.1 size\_t nsent

Definition at line 92 of file test.c.

## 5.9 tracert.c File Reference

```
#include <stdlib.h>
#include <stdio.h>
#include <errno.h>
#include <unistd.h>
#include <sys/time.h>
#include <string.h>
#include <netinet/ip.h>
#include <netinet/ip_icmp.h>
#include <netinet/udp.h>
#include <netdb.h>
#include <fcntl.h>
```

### Data Structures

- struct [pseudo\\_header](#)  
*struct for udp pseudo header*

## Macros

- `#define SIZE 1500`

## Functions

- double `get_time` (void)
- double `sub_time` (struct timeval \*out, struct timeval \*in)
- uint16\_t `ip_checksum` (void \*vdata, size\_t length)  
*calculates the ip checksum for some buffer of size length*
- void `debug_print` (struct icmp \*icmp, ssize\_t n)
- int `main` (int argc, char \*argv[])

## Variables

- size\_t `nsent`

### 5.9.1 Macro Definition Documentation

#### 5.9.1.1 `#define SIZE 1500`

Definition at line 95 of file tracert.c.

### 5.9.2 Function Documentation

#### 5.9.2.1 void `debug_print` ( struct icmp \* *icmp*, ssize\_t *n* )

Definition at line 355 of file tracert.c.

#### 5.9.2.2 double `get_time` ( void )

Definition at line 18 of file tracert.c.

#### 5.9.2.3 uint16\_t `ip_checksum` ( void \* *vdata*, size\_t *length* )

calculates the ip checksum for some buffer of size length

##### Parameters

<i>vdata</i>	a pointer to the buffer to be checksummed
<i>length</i>	the length in bytes of the data to be checksummed

##### Returns

the IP checksum of the buffer

Definition at line 37 of file tracert.c.

#### 5.9.2.4 int `main` ( int *argc*, char \* *argv*[] )

Definition at line 106 of file tracert.c.

5.9.2.5 `double sub_time ( struct timeval * out, struct timeval * in )`

Definition at line 27 of file `tracert.c`.

### 5.9.3 Variable Documentation

5.9.3.1 `size_t nsent`

Definition at line 91 of file `tracert.c`.

# Index

- client.c, [9](#)
  - main, [9](#)
- comp\_det
  - ncd.c, [10](#)
  - ncd.h, [13](#)
- debug\_print
  - ping.c, [15](#)
  - ping\_ip.c, [16](#)
  - test.c, [19](#)
  - tracert.c, [21](#)
- dest
  - pseudo\_header, [7](#)
- get\_time
  - ncd.c, [10](#)
  - ping.c, [15](#)
  - ping\_ip.c, [16](#)
  - test.c, [19](#)
  - tracert.c, [21](#)
- ip\_checksum
  - ncd.c, [10](#)
  - ncd.h, [13](#)
  - ping.c, [15](#)
  - ping\_ip.c, [16](#)
  - test.c, [20](#)
  - tracert.c, [21](#)
- len
  - pseudo\_header, [7](#)
- main
  - client.c, [9](#)
  - ncd.c, [11](#)
  - ping.c, [15](#)
  - ping\_ip.c, [16](#)
  - server.c, [17](#)
  - test.c, [20](#)
  - tracert.c, [21](#)
- ncd.c, [10](#)
  - comp\_det, [10](#)
  - get\_time, [10](#)
  - ip\_checksum, [10](#)
  - main, [11](#)
  - recv\_data, [11](#)
  - send\_data, [11](#)
- ncd.h, [12](#)
  - comp\_det, [13](#)
  - ip\_checksum, [13](#)
  - recv\_data, [13](#)
  - SIZE, [12](#)
  - send\_data, [13](#)
- nsent
  - ping.c, [15](#)
  - ping\_ip.c, [17](#)
  - test.c, [20](#)
  - tracert.c, [22](#)
- ping.c, [14](#)
  - debug\_print, [15](#)
  - get\_time, [15](#)
  - ip\_checksum, [15](#)
  - main, [15](#)
  - nsent, [15](#)
  - SIZE, [14](#)
  - sub\_time, [15](#)
- ping\_ip.c, [15](#)
  - debug\_print, [16](#)
  - get\_time, [16](#)
  - ip\_checksum, [16](#)
  - main, [16](#)
  - nsent, [17](#)
  - SIZE, [16](#)
  - sub\_time, [16](#)
- procs\_msg
  - server.c, [17](#)
- proto
  - pseudo\_header, [7](#)
- pseudo\_header, [7](#)
  - dest, [7](#)
  - len, [7](#)
  - proto, [7](#)
  - source, [7](#)
  - zero, [7](#)
- README.md, [17](#)
- recv\_data
  - ncd.c, [11](#)
  - ncd.h, [13](#)
- SIZE
  - ncd.h, [12](#)
  - ping.c, [14](#)
  - ping\_ip.c, [16](#)
  - test.c, [19](#)
  - tracert.c, [21](#)
- send\_data
  - ncd.c, [11](#)
  - ncd.h, [13](#)

- server.c, [17](#)
  - main, [17](#)
  - procs\_msg, [17](#)
- source
  - pseudo\_header, [7](#)
- sub\_time
  - ping.c, [15](#)
  - ping\_ip.c, [16](#)
  - test.c, [20](#)
  - tracert.c, [21](#)
- test.c, [19](#)
  - debug\_print, [19](#)
  - get\_time, [19](#)
  - ip\_checksum, [20](#)
  - main, [20](#)
  - nsent, [20](#)
  - SIZE, [19](#)
  - sub\_time, [20](#)
- tracert.c, [20](#)
  - debug\_print, [21](#)
  - get\_time, [21](#)
  - ip\_checksum, [21](#)
  - main, [21](#)
  - nsent, [22](#)
  - SIZE, [21](#)
  - sub\_time, [21](#)
- zero
  - pseudo\_header, [7](#)