

Trabalho de Redes - PUCRS

Gerado por Doxygen 1.7.1

Terça, 30 de Novembro de 2010 14:43:50

Sumário

1	Índice dos Módulos	1
1.1	Módulos	1
2	Índice dos Componentes	3
2.1	Lista de Componentes	3
3	Índice dos Arquivos	5
3.1	Lista de Arquivos	5
4	Módulos	7
4.1	Criação de sockets e envio de pacotes	7
4.1.1	Descrição Detalhada	7
4.1.2	Funções	7
4.1.2.1	get_promisc_socket	7
4.1.2.2	send_packet	7
4.2	Informação sobre os devices que serão utilizados.	8
4.2.1	Descrição Detalhada	8
4.2.2	Definições dos tipos	8
4.2.2.1	device_info	8
4.2.3	Funções	8
4.2.3.1	dump_device_info	8
4.2.3.2	load_device_info	8
4.2.4	Variáveis	9
4.2.4.1	device	9
4.3	Core do disturber.	9
4.3.1	Descrição Detalhada	9
4.3.2	Funções	9
4.3.2.1	main	9
4.3.2.2	packet_action	9

4.3.3	Variáveis	9
4.3.3.1	logfile	9
4.3.3.2	pid0	9
4.4	Log dos pacotes roubas em formato pcap.	9
4.4.1	Descrição Detalhada	10
4.4.2	Definições dos tipos	10
4.4.2.1	pcap_hdr_t	10
4.4.2.2	pcaprec_hdr_t	10
4.4.3	Funções	10
4.4.3.1	log_packet	10
4.5	Formação de pacotes.	10
4.5.1	Descrição Detalhada	11
4.5.2	Definições e macros	11
4.5.2.1	IP6_MIN_MTU	11
4.5.2.2	IPV6_VERSION	11
4.5.3	Definições dos tipos	11
4.5.3.1	pseudo_header	11
4.5.4	Funções	11
4.5.4.1	alloc_ndadvert	11
4.5.4.2	alloc_ndsolicit	11
4.5.4.3	alloc_pkt2big	12
4.5.4.4	debug_packet	12
4.5.4.5	fake_packet	12
4.6	Dados sobre as vítimas.	12
4.6.1	Descrição Detalhada	12
4.6.2	Definições dos tipos	13
4.6.2.1	victim	13
4.6.3	Funções	13
4.6.3.1	debug_victim	13
4.6.3.2	init_victim	13
4.6.3.3	populate_victim	13
4.6.3.4	victim_info_complete	13
4.6.4	Variáveis	13
4.6.4.1	cvictim	13
5	Classes	15
5.1	Referência da Estrutura device_info	15

5.1.1	Descrição Detalhada	15
5.1.2	Atributos	15
5.1.2.1	hwaddr	15
5.1.2.2	ifa_flags	15
5.1.2.3	index	15
5.1.2.4	ipv4	16
5.1.2.5	ipv6	16
5.1.2.6	name	16
5.2	Referência da Estrutura pcap_hdr_s	16
5.2.1	Atributos	16
5.2.1.1	magic_number	16
5.2.1.2	network	16
5.2.1.3	sigfigs	16
5.2.1.4	snaplen	16
5.2.1.5	thiszone	16
5.2.1.6	version_major	16
5.2.1.7	version_minor	16
5.3	Referência da Estrutura pcaprec_hdr_s	17
5.3.1	Atributos	17
5.3.1.1	incl_len	17
5.3.1.2	orig_len	17
5.3.1.3	ts_sec	17
5.3.1.4	ts_usec	17
5.4	Referência da Estrutura pseudo_header	17
5.4.1	Atributos	17
5.4.1.1	ph_dst	17
5.4.1.2	ph_len	17
5.4.1.3	ph_nxt	17
5.4.1.4	ph_src	17
5.4.1.5	ph_zero	17
5.5	Referência da Estrutura victim	18
5.5.1	Descrição Detalhada	18
5.5.2	Atributos	18
5.5.2.1	hwaddr	18
5.5.2.2	ipv4	18
5.5.2.3	ipv6	18

5.5.2.4	poisoned	18
6	Arquivos	19
6.1	Referência do Arquivo include/communication.h	19
6.2	Referência do Arquivo include/device.h	19
6.3	Referência do Arquivo include/disturber.h	19
6.4	Referência do Arquivo include/log.h	19
6.5	Referência do Arquivo include/packets.h	20
6.6	Referência do Arquivo include/victims.h	20
6.7	Referência do Arquivo src/communication.c	20
6.8	Referência do Arquivo src/device.c	21
6.9	Referência do Arquivo src/disturber.c	22
6.10	Referência do Arquivo src/log.c	23
6.11	Referência do Arquivo src/packets.c	23
6.12	Referência do Arquivo src/victims.c	24

Capítulo 1

Índice dos Módulos

1.1 Módulos

Esta é a lista com todos os módulos:

Criação de sockets e envio de pacotes	7
Informação sobre os devices que serão utilizados.	8
Core do disturber.	9
Log dos pacotes roubas em formato pcap.	9
Formação de pacotes.	10
Dados sobre as vítimas.	12

Capítulo 2

Índice dos Componentes

2.1 Lista de Componentes

Aqui estão as classes, estruturas, uniões e interfaces e suas respectivas descrições:

device_info	15
pcap_hdr_s	16
pcaprec_hdr_s	17
pseudo_header	17
victim	18

Capítulo 3

Índice dos Arquivos

3.1 Lista de Arquivos

Esta é a lista de todos os arquivos e suas respectivas descrições:

include/communication.h	19
include/device.h	19
include/disturber.h	19
include/log.h	19
include/packets.h	20
include/victims.h	20
src/communication.c	20
src/device.c	21
src/disturber.c	22
src/log.c	23
src/packets.c	23
src/victims.c	24

Capítulo 4

Módulos

4.1 Criação de sockets e envio de pacotes

Criação de sockets e envio de pacotes.

Funções

- int [send_packet](#) (char *pkt)
- int [get_promisc_socket](#) (char *dev_name)

4.1.1 Descrição Detalhada

Criação de sockets e envio de pacotes.

4.1.2 Funções

4.1.2.1 int [get_promisc_socket](#) (char * *dev_name*)

Cria um socket promisc.

Parâmetros

dev_name Interface que ficará promisc..

Retorna

Retorna socket em promisc mode.

4.1.2.2 int [send_packet](#) (char * *pkt*)

Envia pacote de dados desde os headers ethernet.

Parâmetros

pkt Pacote com dados desde os headers ethernet.

Retorna

Retorna > 0 em sucesso e < 0 em falha.

4.2 Informação sobre os devices que serão utilizados.

Informação sobre os devices que serão utilizados.

Componentes

- struct `device_info`

Funções

- void `load_device_info` (const char *dev_name)
- void `dump_device_info` (void)

- typedef struct `device_info` `device_info`
- struct `device_info` `device`

4.2.1 Descrição Detalhada

Informação sobre os devices que serão utilizados.

4.2.2 Definições dos tipos

4.2.2.1 typedef struct device_info device_info

Estrutura com a informação sobre o device utilizado.

4.2.3 Funções

4.2.3.1 void dump_device_info (void)

Debug das informações do device.

4.2.3.2 void load_device_info (const char * dev_name)

Carrega as informações do device.

Parâmetros

dev_name Interface do device a ser atrelado.

4.2.4 Variáveis

4.2.4.1 struct device_info device

4.3 Core do disturber.

Lógica de ataque e parser de opções.

Funções

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

Variáveis

- char * `logfile`
- pthread_t `pid0`
- void `packet_action` (char *packet, int len)

4.3.1 Descrição Detalhada

Lógica de ataque e parser de opções.

4.3.2 Funções

4.3.2.1 int main (int *argc*, char ** *argv*)

4.3.2.2 void packet_action (char * *packet*, int *len*)

4.3.3 Variáveis

4.3.3.1 char* logfile

4.3.3.2 pthread_t pid0

4.4 Log dos pacotes roubas em formato pcap.

Log dos pacotes roubas em formato pcap.

Componentes

- struct `pcap_hdr_s`

Funções

- void `log_packet` (const char *packet, const char *logfile)
- typedef struct `pcap_hdr_s` `pcap_hdr_t`
- typedef struct `pcaprec_hdr_s` `pcaprec_hdr_t`

4.4.1 Descrição Detalhada

Log dos pacotes roubas em formato pcap.

4.4.2 Definições dos tipos

4.4.2.1 typedef struct `pcap_hdr_s` `pcap_hdr_t`

4.4.2.2 typedef struct `pcaprec_hdr_s` `pcaprec_hdr_t`

4.4.3 Funções

4.4.3.1 void `log_packet` (const char * *packet*, const char * *logfile*)

Loga os pacotes em formato pcap.

Parâmetros

packet Pacote que será logado.

logfile Arquivo no qual será logado.

4.5 Formação de pacotes.

Formação de pacotes.

Componentes

- struct `pseudo_header`

Definições e Macros

- #define `IPV6_VERSION` 6 << 4
- #define `IP6_MIN_MTU` 1280

Definições de Tipos

- typedef struct `pseudo_header` `pseudo_header`

Funções

- char * `alloc_pkt2big` (struct `victim` *svic, struct `victim` *dvic, struct ip6_hdr *pkt)
- char * `alloc_ndsolicit` (struct in6_addr *addr)
- char * `alloc_ndadvert` (struct `victim` *svic, struct `victim` *dvic)
- void `fake_packet` (char *packet, struct `victim` *dvic)

- void `debug_packet` (char *packet)

4.5.1 Descrição Detalhada

Formação de pacotes.

4.5.2 Definições e macros

4.5.2.1 `#define IP6_MIN_MTU 1280`

4.5.2.2 `#define IPV6_VERSION 6 << 4`

4.5.3 Definições dos tipos

4.5.3.1 `typedef struct pseudo_header pseudo_header`

4.5.4 Funções

4.5.4.1 `char* alloc_ndadvert (struct victim * svic, struct victim * dvic)`

Aloca um pacote de "neighbor advertisement".

Parâmetros

svic Vítima origem.

dvic Vítima destino.

Retorna

Retorna o pacote de "neighbor advertisement".

4.5.4.2 `char* alloc_ndsolicit (struct in6_addr * addr)`

Aloca um pacote de "neighbor solicitation".

Parâmetros

addr Endereço a ser solicitado.

Retorna

Retorna o pacote de "neighbor solicitation".

4.5.4.3 char* alloc_pkt2big (struct victim * *svic*, struct victim * *dvic*, struct ip6_hdr * *pkt*)

Aloca um pacote de "packet too big".

Parâmetros

svic Vítima origem.

dvic Vítima destino.

pkt Pacote que supostamente teria gerado a mensagem.

Retorna

Retorna o pacote de "packet too big".

4.5.4.4 void debug_packet (char * *packet*)

4.5.4.5 void fake_packet (char * *packet*, struct victim * *dvic*)

Altera o mac destino de um pacote.

Parâmetros

packet Pacote que será modificado.

dvic Vítima destino.

4.6 Dados sobre as vítimas.

Dados sobre as vítimas.

Componentes

- struct [victim](#)

Funções

- void [debug_victim](#) (struct [victim](#) *cli)
- int [victim_info_complete](#) (struct [victim](#) *vic)
- void [populate_victim](#) (char *pkt)
- void [init_victim](#) (struct [victim](#) *vic, const char *address)

- typedef struct [victim](#) [victim](#)
- struct [victim](#) svictim [cvictim](#)

4.6.1 Descrição Detalhada

Dados sobre as vítimas.

4.6.2 Definições dos tipos

4.6.2.1 typedef struct victim victim

Estrutura com a informação sobre uma vítima.

4.6.3 Funções

4.6.3.1 void debug_victim (struct victim * *cli*)

4.6.3.2 void init_victim (struct victim * *vic*, const char * *address*)

Popula a estrutura de uma vítima baseado em seu endereço endereço.

Parâmetros

vic Vítima que sera iniciada.

address Endereço da vítima.

4.6.3.3 void populate_victim (char * *pkt*)

Popula os dados de uma vítima baseado em um pacote recebido.

Parâmetros

pkt Pacote recebido.

4.6.3.4 int victim_info_complete (struct victim * *vic*)

Verifica se uma estrutura de vítima está completa.

Parâmetros

vic Vítima a ser verificada.

Retorna

1 se estiver completa e 0 caso não esteja.

4.6.4 Variáveis

4.6.4.1 struct victim svictim cvictim

Capítulo 5

Classes

5.1 Referência da Estrutura `device_info`

```
#include <device.h>
```

Atributos Públicos

- int `index`
- char `name` [IFNAMSIZ]
- unsigned int `ifa_flags`
- struct in_addr `ipv4`
- struct in6_addr `ipv6`
- struct ether_addr `hwaddr`

5.1.1 Descrição Detalhada

Estrutura com a informação sobre o device utilizado.

5.1.2 Atributos

5.1.2.1 `struct ether_addr device_info::hwaddr`

MAC address do device

5.1.2.2 `unsigned int device_info::ifa_flags`

Flags do device

5.1.2.3 `int device_info::index`

Índice do device

5.1.2.4 struct in_addr device_info::ipv4

Endereço IPv4 do device

5.1.2.5 struct in6_addr device_info::ipv6

Endereço IPv6 do device

5.1.2.6 char device_info::name[IFNAMSIZ]

Nome da interface

A documentação para esta estrutura foi gerada a partir do seguinte arquivo:

- [include/device.h](#)

5.2 Referência da Estrutura pcap_hdr_s

```
#include <log.h>
```

Atributos Públicos

- uint32_t [magic_number](#)
- uint16_t [version_major](#)
- uint16_t [version_minor](#)
- int32_t [thiszone](#)
- uint32_t [sigfigs](#)
- uint32_t [snaplen](#)
- uint32_t [network](#)

5.2.1 Atributos

5.2.1.1 uint32_t pcap_hdr_s::magic_number

5.2.1.2 uint32_t pcap_hdr_s::network

5.2.1.3 uint32_t pcap_hdr_s::sigfigs

5.2.1.4 uint32_t pcap_hdr_s::snaplen

5.2.1.5 int32_t pcap_hdr_s::thiszone

5.2.1.6 uint16_t pcap_hdr_s::version_major

5.2.1.7 uint16_t pcap_hdr_s::version_minor

A documentação para esta estrutura foi gerada a partir do seguinte arquivo:

- [include/log.h](#)

5.3 Referência da Estrutura pcaprec_hdr_s

```
#include <log.h>
```

Atributos Públicos

- uint32_t [ts_sec](#)
- uint32_t [ts_usec](#)
- uint32_t [incl_len](#)
- uint32_t [orig_len](#)

5.3.1 Atributos

5.3.1.1 uint32_t pcaprec_hdr_s::incl_len

5.3.1.2 uint32_t pcaprec_hdr_s::orig_len

5.3.1.3 uint32_t pcaprec_hdr_s::ts_sec

5.3.1.4 uint32_t pcaprec_hdr_s::ts_usec

A documentação para esta estrutura foi gerada a partir do seguinte arquivo:

- [include/log.h](#)

5.4 Referência da Estrutura pseudo_header

Atributos Públicos

- struct in6_addr [ph_src](#)
- struct in6_addr [ph_dst](#)
- u_int32_t [ph_len](#)
- u_int8_t [ph_zero](#) [3]
- u_int8_t [ph_nxt](#)

5.4.1 Atributos

5.4.1.1 struct in6_addr pseudo_header::ph_dst

5.4.1.2 u_int32_t pseudo_header::ph_len

5.4.1.3 u_int8_t pseudo_header::ph_nxt

5.4.1.4 struct in6_addr pseudo_header::ph_src

5.4.1.5 u_int8_t pseudo_header::ph_zero[3]

A documentação para esta estrutura foi gerada a partir do seguinte arquivo:

- [src/packets.c](#)

5.5 Referência da Estrutura victim

```
#include <victims.h>
```

Atributos Públicos

- struct ether_addr [hwaddr](#)
- struct in_addr [ipv4](#)
- struct in6_addr [ipv6](#)
- int [poisoned](#)

5.5.1 Descrição Detalhada

Estrutura com a informação sobre uma vítima.

5.5.2 Atributos

5.5.2.1 struct ether_addr victim::hwaddr

MAC da vítima

5.5.2.2 struct in_addr victim::ipv4

Endereço IPv4 da vítima

5.5.2.3 struct in6_addr victim::ipv6

Endereço IPv6 da vítima

5.5.2.4 int victim::poisoned

Flag que indica se a vítima já foi "poisoned"

A documentação para esta estrutura foi gerada a partir do seguinte arquivo:

- [include/victims.h](#)

Capítulo 6

Arquivos

6.1 Referência do Arquivo include/communication.h

Funções

- int `get_promisc_socket` (char *dev_name)
- int `send_packet` (char *pkt)

6.2 Referência do Arquivo include/device.h

```
#include <net/if.h>
```

Componentes

- struct `device_info`
- typedef struct `device_info` `device_info`
- struct `device_info` `device`
- void `load_device_info` (const char *dev_name)
- void `dump_device_info` (void)

6.3 Referência do Arquivo include/disturber.h

Funções

- void `packet_action` (char *packet, int len)

6.4 Referência do Arquivo include/log.h

Componentes

- struct `pcap_hdr_s`

- struct [pcaprec_hdr_s](#)
- typedef struct [pcap_hdr_s](#) [pcap_hdr_t](#)
- typedef struct [pcaprec_hdr_s](#) [pcaprec_hdr_t](#)
- void [log_packet](#) (const char *packet, const char *logfile)

6.5 Referência do Arquivo include/packets.h

```
#include <netinet/ip6.h>
#include <netinet/icmp6.h>
#include <victims.h>
```

Funções

- void [debug_packet](#) (char *packet)
- char * [alloc_pkt2big](#) (struct [victim](#) *svic, struct [victim](#) *dvic, struct ip6_hdr *pkt)
- char * [alloc_ndsolicit](#) (struct in6_addr *addr)
- char * [alloc_ndadvert](#) (struct [victim](#) *svic, struct [victim](#) *dvic)
- void [fake_packet](#) (char *packet, struct [victim](#) *dvic)

6.6 Referência do Arquivo include/victims.h

```
#include <pthread.h>
#include <netinet/in.h>
#include <netinet/ether.h>
```

Componentes

- struct [victim](#)
- typedef struct [victim](#) [victim](#)
- struct [victim](#) svictim [cvictim](#)
- void [init_victim](#) (struct [victim](#) *vic, const char *address)
- void [debug_victim](#) (struct [victim](#) *cli)
- int [victim_info_complete](#) (struct [victim](#) *vic)
- void [populate_victim](#) (char *pkt)

6.7 Referência do Arquivo src/communication.c

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

```
#include <errno.h>
#include <netdb.h>
#include <ifaddrs.h>
#include <unistd.h>
#include <sys/socket.h>
#include <sys/ioctl.h>
#include <linux/if_packet.h>
#include <netinet/if_ether.h>
#include <device.h>
#include <packets.h>
```

Funções

- int [send_packet](#) (char *pkt)
- int [get_promisc_socket](#) (char *dev_name)

6.8 Referência do Arquivo src/device.c

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <errno.h>
#include <netdb.h>
#include <ifaddrs.h>
#include <unistd.h>
#include <sys/socket.h>
#include <sys/ioctl.h>
#include <net/if.h>
#include <linux/if_packet.h>
#include <arpa/inet.h>
#include <netinet/in.h>
#include <netinet/if_ether.h>
#include <netinet/ip6.h>
#include <netinet/icmp6.h>
#include <netinet/tcp.h>
#include <netinet/ether.h>
#include <packets.h>
```

```
#include <device.h>
```

Funções

- void `load_device_info` (const char *dev_name)
- void `dump_device_info` (void)

6.9 Referência do Arquivo src/disturber.c

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <errno.h>
#include <unistd.h>
#include <pthread.h>
#include <signal.h>
#include <getopt.h>
#include <sys/socket.h>
#include <linux/if_packet.h>
#include <arpa/inet.h>
#include <netinet/if_ether.h>
#include <netinet/ip6.h>
#include <netinet/icmp6.h>
#include <netinet/tcp.h>
#include <netinet/ether.h>
#include <device.h>
#include <packets.h>
#include <communication.h>
#include <log.h>
```

Funções

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

Variáveis

- char * `logfile`
- pthread_t `pid0`

6.10 Referência do Arquivo src/log.c

```
#include <stdio.h>
#include <unistd.h>
#include <time.h>
#include <string.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <netinet/ether.h>
#include <netinet/ip6.h>
#include <log.h>
```

Funções

- void [log_packet](#) (const char *packet, const char *logfile)

6.11 Referência do Arquivo src/packets.c

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <netinet/ether.h>
#include <netinet/ip6.h>
#include <netinet/icmp6.h>
#include <netinet/tcp.h>
#include <arpa/inet.h>
#include <disturber.h>
#include <device.h>
#include <victims.h>
```

Componentes

- struct [pseudo_header](#)

Definições e Macros

- #define `IPV6_VERSION` 6 << 4
- #define `IP6_MIN_MTU` 1280

Definições de Tipos

- typedef struct `pseudo_header` `pseudo_header`

Funções

- char * `alloc_pkt2big` (struct `victim` *svic, struct `victim` *dvic, struct `ip6_hdr` *pkt)
- char * `alloc_ndsolicit` (struct `in6_addr` *addr)
- char * `alloc_ndadvert` (struct `victim` *svic, struct `victim` *dvic)
- void `fake_packet` (char *packet, struct `victim` *dvic)

- void `debug_packet` (char *packet)

6.12 Referência do Arquivo src/victims.c

```
#include <stdio.h>
#include <unistd.h>
#include <pthread.h>
#include <stdlib.h>
#include <string.h>
#include <errno.h>
#include <arpa/inet.h>
#include <netinet/in.h>
#include <netinet/if_ether.h>
#include <netinet/ip6.h>
#include <netinet/icmp6.h>
#include <netinet/tcp.h>
#include <netinet/ether.h>
#include <disturber.h>
#include <packets.h>
#include <device.h>
#include <communication.h>
```

Funções

- void `debug_victim` (struct `victim` *cli)
- int `victim_info_complete` (struct `victim` *vic)

- void `populate_victim` (char *pkt)
- void `init_victim` (struct `victim` *vic, const char *address)

Índice Remissivo

- alloc_ndadvert
 - packets, [11](#)
- alloc_ndsolicit
 - packets, [11](#)
- alloc_pkt2big
 - packets, [11](#)
- communication
 - get_promisc_socket, [7](#)
 - send_packet, [7](#)
- Core do disturber., [9](#)
- Criação de sockets e envio de pacotes, [7](#)
- cvictim
 - victims, [13](#)
- Dados sobre as vítimas., [12](#)
- debug_packet
 - packets, [12](#)
- debug_victim
 - victims, [13](#)
- device
 - device, [9](#)
 - device_info, [8](#)
 - dump_device_info, [8](#)
 - load_device_info, [8](#)
- device_info, [15](#)
 - device, [8](#)
 - hwaddr, [15](#)
 - ifa_flags, [15](#)
 - index, [15](#)
 - ipv4, [15](#)
 - ipv6, [16](#)
 - name, [16](#)
- disturber
 - logfile, [9](#)
 - main, [9](#)
 - packet_action, [9](#)
 - pid0, [9](#)
- dump_device_info
 - device, [8](#)
- fake_packet
 - packets, [12](#)
- Formação de pacotes., [10](#)
- get_promisc_socket
 - communication, [7](#)
- hwaddr
 - device_info, [15](#)
 - victim, [18](#)
- ifa_flags
 - device_info, [15](#)
- incl_len
 - pcaprec_hdr_s, [17](#)
- include/communication.h, [19](#)
- include/device.h, [19](#)
- include/disturber.h, [19](#)
- include/log.h, [19](#)
- include/packets.h, [20](#)
- include/victims.h, [20](#)
- index
 - device_info, [15](#)
- Informação sobre os devices que serão utilizados., [8](#)
- init_victim
 - victims, [13](#)
- IP6_MIN_MTU
 - packets, [11](#)
- ipv4
 - device_info, [15](#)
 - victim, [18](#)
- ipv6
 - device_info, [16](#)
 - victim, [18](#)
- IPV6_VERSION
 - packets, [11](#)
- load_device_info
 - device, [8](#)
- log
 - log_packet, [10](#)
 - pcap_hdr_t, [10](#)
 - pcaprec_hdr_t, [10](#)
- Log dos pacotes roubas em formato pcap., [9](#)
- log_packet
 - log, [10](#)
- logfile
 - disturber, [9](#)

- magic_number
 - pcap_hdr_s, 16
- main
 - disturber, 9
- name
 - device_info, 16
- network
 - pcap_hdr_s, 16
- orig_len
 - pcaprec_hdr_s, 17
- packet_action
 - disturber, 9
- packets
 - alloc_ndadvert, 11
 - alloc_ndsolicit, 11
 - alloc_pkt2big, 11
 - debug_packet, 12
 - fake_packet, 12
 - IP6_MIN_MTU, 11
 - IPV6_VERSION, 11
 - pseudo_header, 11
- pcap_hdr_s, 16
 - magic_number, 16
 - network, 16
 - sigfigs, 16
 - snaplen, 16
 - thiszone, 16
 - version_major, 16
 - version_minor, 16
- pcap_hdr_t
 - log, 10
- pcaprec_hdr_s, 17
 - incl_len, 17
 - orig_len, 17
 - ts_sec, 17
 - ts_usec, 17
- pcaprec_hdr_t
 - log, 10
- ph_dst
 - pseudo_header, 17
- ph_len
 - pseudo_header, 17
- ph_nxt
 - pseudo_header, 17
- ph_src
 - pseudo_header, 17
- ph_zero
 - pseudo_header, 17
- pid0
 - disturber, 9
- poisoned
 - victim, 18
- populate_victim
 - victims, 13
- pseudo_header, 17
 - packets, 11
 - ph_dst, 17
 - ph_len, 17
 - ph_nxt, 17
 - ph_src, 17
 - ph_zero, 17
- send_packet
 - communication, 7
- sigfigs
 - pcap_hdr_s, 16
- snaplen
 - pcap_hdr_s, 16
- src/communication.c, 20
- src/device.c, 21
- src/disturber.c, 22
- src/log.c, 23
- src/packets.c, 23
- src/victims.c, 24
- thiszone
 - pcap_hdr_s, 16
- ts_sec
 - pcaprec_hdr_s, 17
- ts_usec
 - pcaprec_hdr_s, 17
- version_major
 - pcap_hdr_s, 16
- version_minor
 - pcap_hdr_s, 16
- victim, 18
 - hwaddr, 18
 - ipv4, 18
 - ipv6, 18
 - poisoned, 18
 - victims, 13
- victim_info_complete
 - victims, 13
- victims
 - cvictim, 13
 - debug_victim, 13
 - init_victim, 13
 - populate_victim, 13
 - victim, 13
 - victim_info_complete, 13