

---

# **PacketAnalyser**

***Release 1.0.0***

**Akira Takeuchi**

**Feb 08, 2025**



**CONTENTS:**

<b>1</b>	<b>pktsniffer module</b>	<b>3</b>
<b>2</b>	<b>filter_packet module</b>	<b>5</b>
<b>3</b>	<b>addres_conv module</b>	<b>7</b>
<b>4</b>	<b>parse_mssg module</b>	<b>9</b>
	<b>Python Module Index</b>	<b>11</b>
	<b>Index</b>	<b>13</b>



Add your content using reStructuredText syntax. See the [reStructuredText](#) documentation for details.



## PKTSNIFFER MODULE

`pktsniffer.main()`

Main function to parse command line arguments and process pcap file

`pktsniffer.process_pcap(file_path, filter_opts, count)`

Process the pcap file and print packet details

**Parameters**

**name** – input file, filter\_opts, count

**Returns**

N/A

**Return type**

N/A





## FILTER\_PACKET MODULE

`filter_packet.filter_packet(ip, filter_opts)`

Filter packets based on the provided options

**Parameters**

**name** – ip, filter\_opts

**Returns**

bool (if the packet passes filter or not)

**Return type**

bool



## ADDRES\_CONV MODULE

`address_conv.ip6_addr(address)`

Convert IPv6 address to human-readable format

**Parameters**

**name** – address

**Returns**

ip6 address

**Return type**

str

`address_conv.ip_addr(address)`

Convert IP address to human-readable format

**Parameters**

**name** – address

**Returns**

ip address

**Return type**

str

`address_conv.mac_addr(address)`

Convert MAC address to human-readable format

**Parameters**

**name** – address

**Returns**

mac address

**Return type**

str



## PARSE\_MSSG MODULE

`parse_mssg.parse_ethernet(packet)`

Parse Ethernet header

`parse_mssg.parse_icmp icmp`

Parse ICMP header

**Parameters**

**name** – icmp

**Returns**

N/A

**Return type**

N/A

`parse_mssg.parse_ip(ip)`

Parse IP header

**Parameters**

**name** – ip

**Returns**

ip

**Return type**

ip packet

`parse_mssg.parse_ip6(ip6)`

Parse IPv6 header

**Parameters**

**name** – ip6

**Returns**

ip6

**Return type**

ip6 packet

`parse_mssg.parse_tcp(tcp)`

Parse TCP header

**Parameters**

**name** – tcp

**Returns**

N/A

**Return type**

N/A

`parse_mssg.parse_udp(udp)`

Parse UDP header

**Parameters**

**name** – udp

**Returns**

N/A

**Return type**

N/A

## PYTHON MODULE INDEX

### a

`addres_conv`, 7

### f

`filter_packet`, 5

### p

`pktsniffer`, 3





## INDEX

### A

`adres_conv`  
    module, 7

### F

`filter_packet`  
    module, 5  
`filter_packet()` (in module *filter\_packet*), 5

### I

`ip6_addr()` (in module *adres\_conv*), 7  
`ip_addr()` (in module *adres\_conv*), 7

### M

`mac_addr()` (in module *adres\_conv*), 7  
`main()` (in module *pktsniffer*), 3  
module  
    *adres\_conv*, 7  
    *filter\_packet*, 5  
    *parse\_mssg*, 9  
    *pktsniffer*, 3

### P

`parse_ethernet()` (in module *parse\_mssg*), 9  
`parse_icmp()` (in module *parse\_mssg*), 9  
`parse_ip()` (in module *parse\_mssg*), 9  
`parse_ip6()` (in module *parse\_mssg*), 9  
`parse_mssg`  
    module, 9  
`parse_tcp()` (in module *parse\_mssg*), 9  
`parse_udp()` (in module *parse\_mssg*), 10  
`pktsniffer`  
    module, 3  
`process_pcap()` (in module *pktsniffer*), 3