PktSniffer

Release 0.1

Ethan Iannicelli

Jan 30, 2025

CONTENTS:

CHAPTER

ONE

PACKET SUMMARY

To get the summary of the ethernet header, you can use the `get_eth_summary()` function:

pktsniffer.**get_eth_summary**(*packet*)

Print selected ethernet header properties in a formatted manner

Parameters
packet (*pyshark packet*) – required packet to be summarized

To get the summary of the ip header, you can use the `get_ip_summary()` function:

pktsniffer.**get_ip_summary**(*packet*)

Print selected ip header properties in a formatted manner

Parameters
packet (*pyshark packet*) – required packet to be summarized

To get the summary of an encapsulated packet, you can use the `get_encapsulated_packets_summary()` function:

pktsniffer.**get_encapsulated_packets_summary**(*packet*)

Print any encapsulated packet(s) in a given packet (not specially formatted, does not extract specific properties)

Parameters
packet (*pyshark packet*) – required packet to be summarized

To get all available packet summaries, you can use the `get_packet_summary()` function:

pktsniffer.**get_packet_summary**(*packet*)

Print all available header summaries for a given packet

Parameters
packet (*pyshark packet*) – required packet to be summarized

To filter a list of packets by a host address, you can use the `filter_by_host()` function:

pktsniffer.**filter_by_host**(*packets*, *host*)

Filter all packets if they contain the host address in either the packet ip source property or the packet destination property

Parameters

- **packets** (*list[pyshark packet]*) – List of packets
- **host** (*MAC address*) – host to filter by

Returns
the filtered list

Return type
list[pyshark packet]

To filter a list of packets by a port, you can use the `filter_by_port()` function:

pktsniffer.**filter_by_port**(*packets*, *port*)

Filter all packets if they contain the port in either the encapsulated packet source property or encapsulated packet destination property

Parameters

- **packets** (*list[pyshark packet]*) – List of packets
- **port** – port to filter by

Returns
the filtered list

Return type
list[pyshark packet]

To check if a packet has a certain port number, you can use the `has_port()` function:

pktsniffer.**has_port**(*packet*, *port*)

Check if a packet has a port number in a encapsulated TCP or UDP packet at the source or destination property

Parameters

- **packet** (*pyshark packet*) – the packet to be checked
- **port** (*Int*) – the port number

Returns
the boolean value indicating if the packet has the port

Return type
boolean

To filter a list of packets by a ip version, you can use the `filter_by_ip()` function:

pktsniffer.**filter_by_ip**(*packets*, *ip*)

Filter all packets if they contain the ip version in the packet ip header

Parameters

- **packets** (*list[pyshark packet]*) – List of packets
- **ip** (*Int*) – ip version to filter by

Returns
the filtered list

Return type
list[pyshark packet]

To filter a list of packets by a net, you can use the `filter_by_net()` function:

pktsniffer.**filter_by_net**(*packets*, *net*)

Filter all packets if they contain an encapsulated icmp packet

Parameters
packets (*list[pyshark packet]*) – List of packets

Returns
the filtered list

Return type
list[pyshark packet]

To filter a list of packets by a tcp, you can use the `filter_by_tcp()` function:

pktsniffer.**filter_by_tcp**(*packets*)

Filter all packets if they contain the same address in either the packet ip source or destination property

Parameters

- **packets** (*list[pyshark packet]*) – List of packets
- **net** (*MAC address*) – net to filter by

Returns
the filtered list

Return type
list[pyshark packet]

To filter a list of packets by a udp, you can use the `filter_by_udp()` function:

pktsniffer.**filter_by_udp**(*packets*)

Filter all packets if they contain an encapsulated tcp packet

Parameters
packets (*list[pyshark packet]*) – List of packets

Returns
the filtered list

Return type
list[pyshark packet]

To filter a list of packets by a icmp, you can use the `filter_by_icmp()` function:

pktsniffer.**filter_by_icmp**(*packets*)

Filter all packets if they contain an encapsulated udp packet

Parameters
packets (*list[pyshark packet]*) – List of packets

Returns
the filtered list

Return type
list[pyshark packet]

To filter a list of packets by all filters, use the `filter_packets()` function:

pktsniffer.**filter_packets**(*packets*, *filters*)

This function uses all the filtering helper functions to filter a list of packets given a set of (active) filters

Parameters

- **packets** (*list[pyshark packet]*) – list of packets
- **filters** (*map<string, value>*) – the filters to use in filtering the packets

Returns
list of filtered packets

Return type
list[pyshark packet]

To initiate the program parser, you can use the `initialize_parser()` function:

pktsniffer.**initialize_parser**()

This function creates and defines the parser for the packet sniffer program, including file arguments, filtering arguments, and count arguments

Returns
the initialized parser

Return type
ArgParser