The file waitpacket.c contains functions related to handling and processing incoming network packets.

recv_beep: This function checks if the opt_beep option is set, and if so, it prints a beep sound to the console. This function is called for every matching packet received.

wait_packet: This function waits for a packet to be received. It reads the packet, checks if it matches certain criteria, and then processes it accordingly. If the packet is an ICMP, UDP, or TCP packet, it is passed to the corresponding function for further processing. If the packet is an HCMP packet, it is passed to the handle_hcmp function.

log_ip: This function logs information about an IP packet, including its length, source IP, TTL, ID, and other details.

log_icmp_ts: This function logs information about an ICMP timestamp.

log_icmp_addr: This function logs information about an ICMP address mask.

recv_icmp: This function processes an ICMP packet. It checks the type of the ICMP packet and handles it accordingly.

recv_udp: This function processes a UDP packet. It checks if the packet matches certain criteria, and if so, it updates the RTT and logs the packet information.

recv_tcp: This function processes a TCP packet. It checks if the packet matches certain criteria, and if so, it updates the RTT, logs the packet information, and handles TCP timestamps if the opt_tcp_timestamp option is set.

read_packet: This function reads a packet from the network using the pcap_recv function.

hex_dump: This function prints a hex dump of a packet.

human_dump: This function prints a human-readable dump of a packet.

handle_hcmp: This function handles an HCMP packet. It checks for the reverse signature in the packet and processes the packet accordingly. The handle_hcmp function is used to handle the HCMP (Hping Control Message Protocol) for almost safe file transfer with hping.

icmp_unreach_rtt: The  icmp_unreach_rtt function is used to extract information about the original packet from the ICMP error to obtain the round time trip.

The print_tcp_timestamp function is used to print the TCP timestamp option if present.

Clock_skew: The clock_skew function in the waitpacket.c file is used to detect and calculate the clock skew between the local host and the remote host. Clock skew refers to the difference in the perceived time between different nodes in a network.

The function takes three parameters: hz (the frequency of the remote host's clock in Hertz), tstamp (the timestamp received from the remote host), and rttms (the round-trip time of the packet in milliseconds).

The function works by collecting a series of samples of the time difference between the local host and the remote host. Each sample is based on a certain number of packets (cs_vector_len). For each packet, the function calculates the time difference between the local host and the remote host, corrected for the round-trip time. These time differences are stored in the deltavect array, along with the corresponding round-trip times in the deltartt array.

Once cs_vector_len packets have been captured, the function calculates a sample of the clock skew. This is done by averaging the time differences in deltavect, but discarding any packets with a round-trip time that is significantly larger than the minimum round-trip time. This is because packets with a high round-trip time can introduce errors into the calculation.

The calculated sample is then stored in the sample array, along with the corresponding local time in the samplelocal array.

If enough samples have been collected, the function calculates the clock skew based on these samples. This is done by calculating the skew for multiple pairs of samples with a given time distance (window), and then averaging these skews to get a more accurate estimate.

The function then prints out various information about the clock skew, such as the estimated skew in nanoseconds per second, the local time difference used to calculate the skew, and the number of deltas collected for the next sample.