Use Case: Modify Packets
Scenario Name: Modify Packets
Description: An analyst makes use of the functionalities the system provides to perform a series of operations on the packets captured.
Actors: Analyst, Packet Replayer, and Fuzzer.
Pre-condition: The System has intercepted packets to be modified.
Trigger-condition: Analyst has selected a packet and has accessed the option that deals with packet modification.
Flow of events:
Step 1: Analyst selects the functionality called “package options”.
Step 2: System responds with a dialog that asks the user to select from the enqueued packets which will be modified.
Step 3: The Analyst acknowledges the dialog.
Step 4: Analyst chooses the packets that will be modified from the queue of packets.
Step 5: The System displays the information contained in the captured packet on an editor as a command line representation.
Step 6: The Analyst performs intended modifications on the packets using a specified view (raw, hex, or decoded).
Step 7: The Analyst saves the modifications made to the packet.
Step 8: The System overwrites the captured packet.
Step 9: Extend << Send Packet>>
Step 10: Extend <<Drop Packet>>
Step 11: Extend <<Fuzz Packet>>
Step 12: End of use case.

Scenario Name: Drop Packet
Description: The Analyst can decide to drop a packet
Actors: Analyst, Enqueuing Tool
Pre-condition: The Analyst has entered to the modification packets options and a packet has been intercepted previously.
Trigger-condition: Analyst has choose the drop option while modifying a packet.
Flow of events:
Step 1: The Analyst choose a packet and as a modification option the Analyst choose to drop it.
Step 2: The System communicates to the Enqueuing tool which packet to drop from the transmission queue.
Step 3: The Enqueueing tool sets a rule blocks that specific packet from being sent.
Step 4: The Enqueuing tool provides a response to the input of the System.
Step 5: The System communicates the Analyst the status of the dropping.
Step 6: End of use case.

Scenario Name: Fuzz Packet
Description: Modify the packet using the functionality of the Fuzzer.
Actors: Analyst, Fuzzer
Pre-condition: The Analyst has entered to the modification packets options and a packet has been intercepted previously.
Trigger-condition: The Analyst choose to modify the contents of a packet in the queue of packets by making use of the functionality of the Fuzzer.
Flow of events:
Step 1: The Analyst selects the option to modify the packet by making use of the Fuzzer.
Step 2: The System displays a list of possible sections the system can fuzz within the packet.

Step 3: The System sends the information to the Fuzzer that contains the files and the System executes the necessary scripts to prepare the Fuzzer's wrapper to fuzz the information.
Step 4: The Fuzzer creates the fuzzed results and saves it to a specific directory.
Step 5: The System then retrieves the fuzzed packet.
Step 6: The System saves a copy of the fuzzed packet into the archives of the system.
Step 7: End of use case.

Scenario Name: Send Packet
Description: A packet can be retransmitted after being intercepted.
Actors: Packet Replayer and Analyst
Pre-condition: A packet has been intercepted previously
Trigger-condition: The Analyst activates the send functionality of the modification of packets while targeting a specific number of packets.
Flow of events:
Step 1: The Analyst activates the transmission of packets a selection of intercepted packages.
Step 2: The System communicates a created PCAP file based on the packets selected by the Analyst to the Packet Replayer.
Step 3: The Packet Replayer receives the PCAP files and produces a communication instruction that recreates the flow of information instructed in the packets contained within the PCAP files.
Step 4: A confirmation of the send procedure is read by the System from the Packet Replayer.
Step 5: The System notify the results of transmitting the packets to the Analyst.
Step 6: End of use case.