Wireshark: Packet Operations

### 🧠 **Wireshark: Packet Operations - Notes**

#### **Task 1: Introduction**

* This module builds on Wireshark basics.
* Focuses on using Wireshark for deeper protocol analysis, statistics, and advanced filtering.

#### **Task 2: Statistics | Summary**

* **Statistics > Summary** gives a high-level overview:
  + Packet counts
  + File size
  + Capture duration
  + Average packets per second
* Useful for quick capture insights.

#### **Task 3: Statistics | Protocol Details**

* **Statistics > Protocol Hierarchy**:
  + Shows percentage of traffic per protocol (Ethernet, IP, TCP, DNS, etc.)
  + Helps identify unusual or unexpected protocols.
* **Conversations and Endpoints**:
  + Found under **Statistics > Conversations/Endpoints**
  + Breaks down communication pairs and stats by IP, TCP/UDP sessions.

#### **Task 4: Packet Filtering | Principles**

* Wireshark uses **Display Filters** (not capture filters here).
* Basic structure:  
  field operator value  
  e.g., ip.addr == 192.168.1.1
* Use logical operators:
  + and, or, not
* Combine filters:
  + (ip.src == 10.0.0.1) and (tcp.port == 80)

#### **Task 5: Packet Filtering | Protocol Filters**

* Filter by protocol:
  + http, dns, ftp, tcp, udp, etc.
* Example:
  + http.request → shows HTTP GET/POST
  + dns.qry.name == "example.com"
* Use autocomplete feature (Ctrl+Space) in the filter bar.

#### **Task 6: Advanced Filtering**

* Filters for complex conditions:
  + Filter by flags: tcp.flags.syn == 1
  + Filter by length: frame.len > 1000
  + Filter by time: frame.time >= "Aug 8, 2025 10:00:00"
* Use **"Apply as Filter"** and **"Prepare a Filter"** from right-click context menu.

#### **Task 7: Conclusion**

* Wireshark’s filtering and statistical tools make it powerful for threat detection and protocol analysis.
* Regular use of filters and statistics streamlines investigations.

