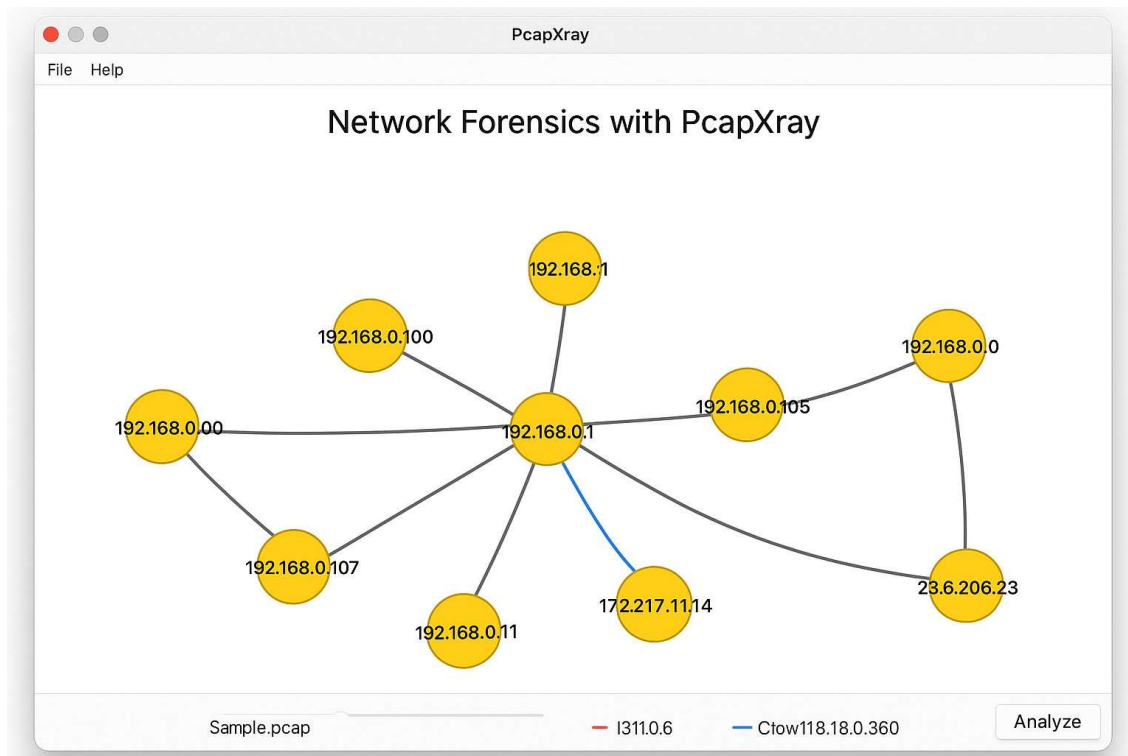


EXPT: 15 DEMONSTRATE NETWORK FORENSICS USING PCAPXRAY TOOLS**Aim:**

To analyze captured network traffic using PcapXray and identify hosts, traffic patterns, and suspicious network activities for forensic investigation.

Algorithm:

1. Install prerequisites:
 - Install Python 3, pip, Graphviz, Tkinter, and required libraries.
 - Clone the PcapXray repository and install dependencies using `pip install -r requirements.txt`.
2. Prepare input:
 - Obtain a .pcap file containing network traffic to be analyzed.
 - Ensure the PCAP is from a safe/testing source for learning purposes.
3. Launch PcapXray:
 - Open main.py in the repository using Python.
 - Load the selected .pcap file via the GUI.
4. Analyze traffic:
 - Observe the network graph of hosts (nodes) and connections (edges).
 - Filter traffic based on Web, Tor, Malicious, DNS, or ICMP.
 - Click on nodes/edges to view traffic details, HTTP requests, or extracted payloads.
5. Record observations:
 - Note suspicious hosts, unusual ports, or Tor traffic.
 - Check extracted files or payloads for anomalies.
 - Optionally, cross-verify suspicious IPs with WHOIS or threat intelligence sources.
6. Document results:
 - Capture screenshots of network diagrams and significant flows.
 - Summarize the suspicious activities identified during analysis.

Output:

- Graphical visualization of network hosts and flows.
- Reports listing:
 - Host IPs
 - Connection types
 - Protocols used
 - Extracted payloads
 - Flags for Tor/malicious traffic
- Optional JSON or text files summarizing traffic analysis.

Result :

- Hosts with the most connections were identified as central nodes.
- Web traffic, Tor traffic, and DNS requests were visualized clearly.
- Suspicious or unusual traffic flows were highlighted for further investigation.
- Payload extraction revealed potential files or URLs of interest.
- PcapXray provided a clear, interactive overview of network activity, making it easier to identify anomalies or malicious patterns compared to raw packet inspection in Wireshark.