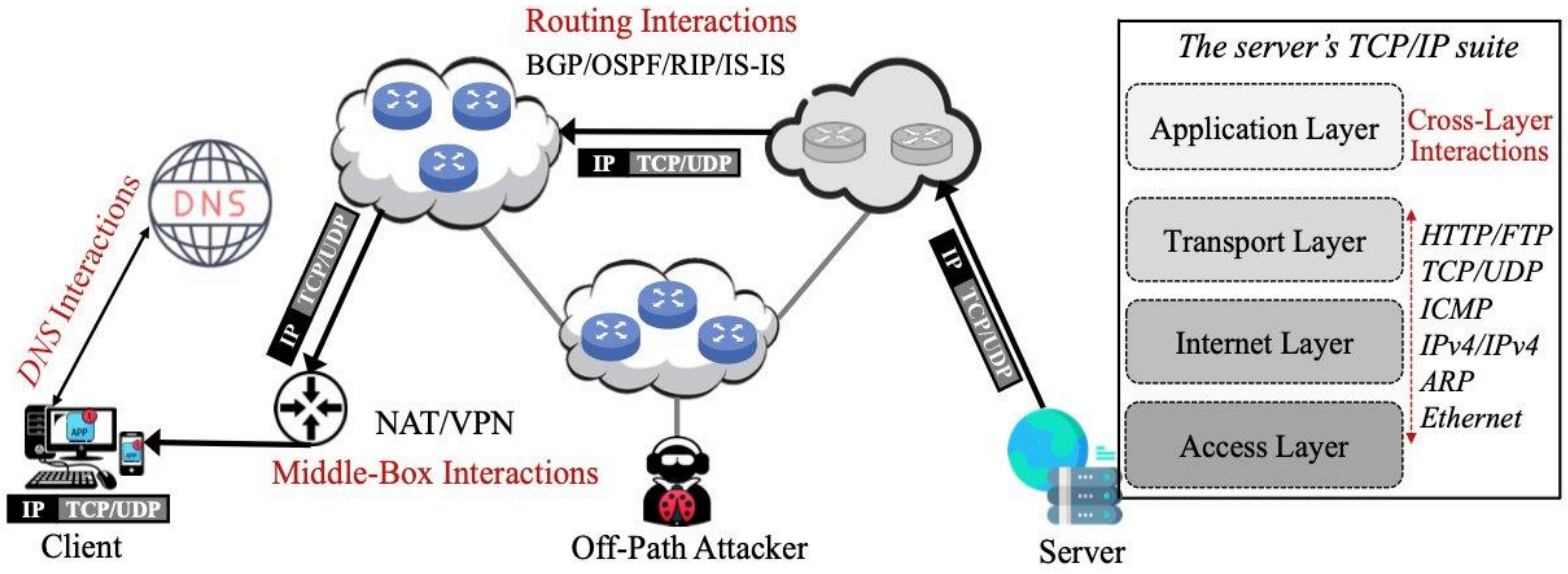
IETF 124 Hackathon

PacketScope

PacketScope: LLM-Powered eBPF for Protocol Stack Defense

What problems are we working on?

- The traditional network stack is a **black box**, leading to two major pain points:
  - **Difficult to Diagnose**:
    - Performance bottlenecks (latency, packet loss) are hard to pinpoint.
    - Packet paths within the kernel are unclear.
  - **Difficult to Defend**:
    - Hard to Detect: Modern attacks (e.g., cross-protocol, semantic) are stealthy. Static rules and traditional IDS miss them.
    - Hard to Block: A gap exists between detection and response. Admins must manually write and deploy complex rules, which is too slow.

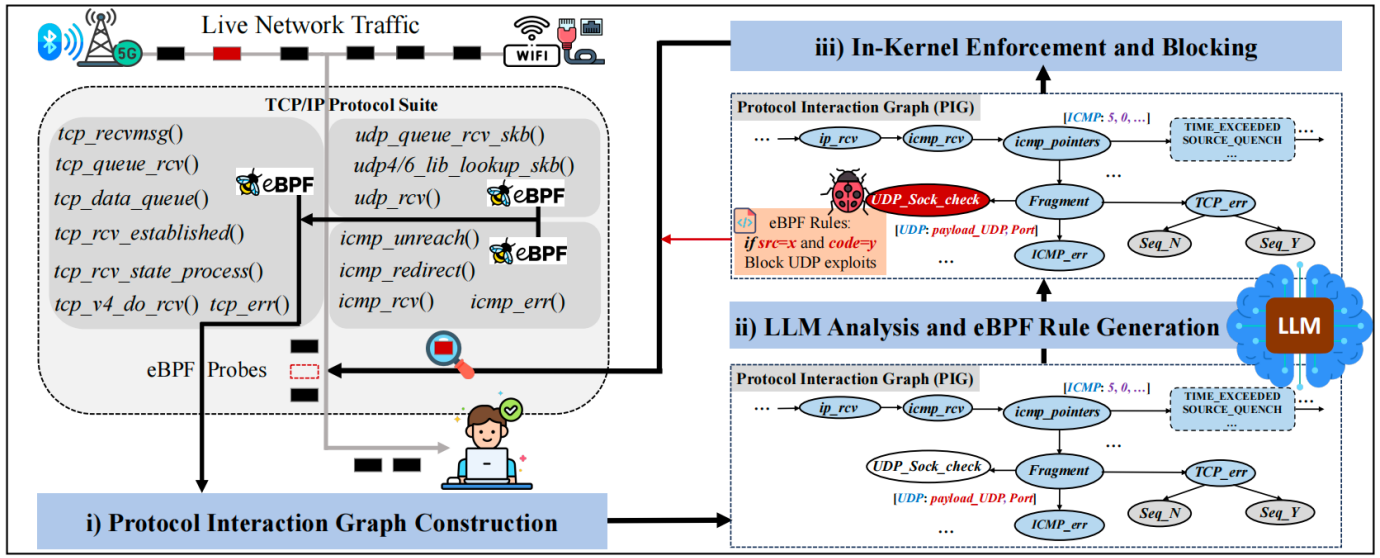


Threat Model

Demo: http://82.156.141.213:4173/

What have we achieved?

- We successfully deployed PacketScope on our server.
- Deployed & Monitored: Deployed in the test network, monitoring and analyzing live traffic (Analyzer).
  - Validated the Guarder Module: Achieved a seamless integration of eBPF and LLMs.
    - eBPF (XDP): Provides zero-copy, zero-latency data collection in the kernel.
    - LLM: Uses long-context reasoning on eBPF data to identify stealthy attacks.
  - Built an "Intelligent Defense Loop":
    - LLM analyzes threats and auto-generates eBPF filtering rules.
    - Rules are hot-swapped into the kernel's XDP hook.
    - We successfully used this to auto-detect and block an ICMP Flooding attack in our demo.



Design of PacketScope

Demo: http://82.156.141.213:4173/

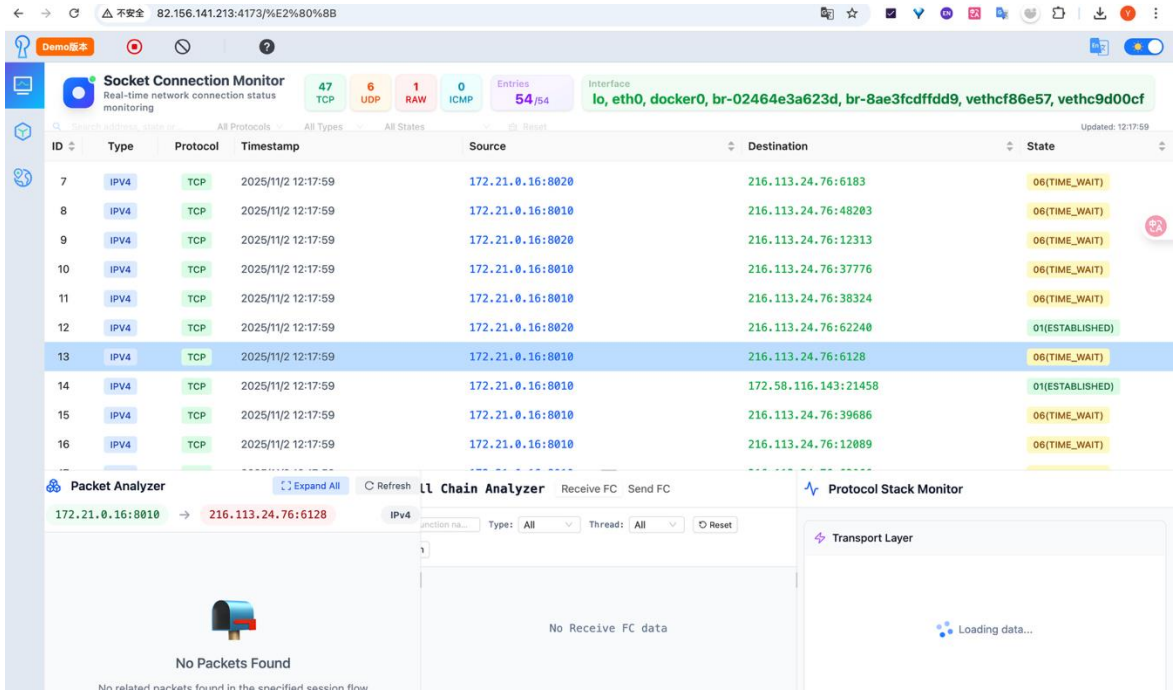
Highlights

- Lessons Learned:
  - "LLM + eBPF" is a new paradigm for network defense. It shifts security from "passive response" to "intelligent, real-time blocking."
  - LLMs solve the "Hard to Block" problem by auto-generating precise rules, drastically lowering the operational burden.
  - XDP provides "zero-latency" defense by dropping malicious packets before they enter the stack.
- Feedback to working group:
  - We hope to see more flexible eBPF hooks for modern protocols (e.g., QUIC, HTTP/3).

Demo: http://82.156.141.213:4173/

Conclusion & Future Work

- Future Work:
  - Extend tracing to application-layer protocols (e.g., HTTP/2, QUIC).
  - Build cross-host correlation for distributed tracing and analysis.
  - Using BTF CO-RE to enhance compatibility
- GitHub: Internet-Architecture-and-Security/PacketScope



Demo: http://82.156.141.213:4173/