

DNS RESOLVER USING AF_XDP SOCKET

PROJECT PROPOSAL - TEAM 13

Guntas Singh Saran^{*} & Hrriday V. Ruparel[†] & Kishan Ved[‡] & Pranav Patil[§]

Department of Computer Science and Engineering
Indian Institute of Technology Gandhinagar
Palaj, GJ 382355, India

Prof. Sameer Kulkarni

Department of Computer Science and Engineering
Indian Institute of Technology Gandhinagar
Palaj, GJ 382355, India

1 INTRODUCTION

This project focuses on building a DNS resolver that uses the AF_XDP socket to bypass the traditional Linux kernel networking stack, enabling high-throughput and low-latency DNS query processing. By leveraging XDP programs, the resolver will filter DNS packets directly at the NIC level, extract domain names, forward queries to a recursive resolver (e.g., Google Public DNS), and send responses back to clients. The system aims to handle packets with reduced CPU usage, making it ideal for high-performance networking scenarios.

2 OBJECTIVES

- Implement an XDP program to filter incoming DNS packets (on port 53) and redirect them to an AF_XDP socket.
- Design a user-space application to process DNS queries, perform lookups using nslookup or similar tools, and construct DNS responses.
- Optimize packet processing using zero-copy memory access via Direct Memory Access (DMA) to avoid CPU intervention.
- Benchmark the resolver's performance in terms of throughput, latency, and CPU usage compared to traditional DNS resolvers that use Linux Kernel Network Stack.
- So in all the XDP programs:
 - Filter incoming packets on UDP port 53.
 - Extract DNS queries and forward them to a recursive resolver.
 - Construct DNS responses and send them back to clients.

3 PROPOSED WORKFLOW

Tentatively, every header is planned to span for 2 weeks. Initial weeks would involve reading and understanding core concepts.

1. Packet Filtering with XDP

- Write an XDP program to identify DNS query packets destined for port 53.

^{*}Guntas Singh Saran - 22110089 - guntassingh.saran@iitgn.ac.in

[†]Hrriday V. Ruparel 22110099 - hrriday.ruparel@iitgn.ac.in

[‡]Kishan Ved - 22110122 - kishan.ved@iitgn.ac.in

[§]Pranav Patil - 22110199 - pranav.patil@iitgn.ac.in

*Order based on Roll Numbers Only.

- Redirect these packets from the NIC directly to an AF_XDP socket.
- 2. User-Space Packet Processing
 - Bind an AF_XDP socket to the NIC RX queue.
 - Extract domain names from incoming packets and forward queries to a recursive resolver using nslookup.
 - Construct valid DNS response packets and send them back through the TX queue.
- 3. Performance Optimization
 - Enable zero-copy mode for efficient memory access.
 - Minimize context switching between user space and kernel space.
- 4. Benchmarking
 - Evaluate throughput, latency, and CPU usage using tools like dnsperf.
 - Compare performance against traditional resolvers.

4 TOOLS & RESOURCES

1. Development Tools
 - C for XDP program and AF_XDP integration.
 - nslookup, dig, ping, wireshark, tcpdump.
2. Hardware & Software Requirements
 - NIC with support for XDP zero-copy mode.
 - A Linux VM (to avoid complications on our active host machine).
 - Linux kernel with XDP support.
 - Virtualization tools like Libvirt or QEMU for testing in isolated environments.

5 EXPECTED OUTCOMES

The final system will be a high-performance XDP-based DNS resolver with reduced latency and CPU usage compared to conventional approaches. This project will demonstrate the practical application of AF_XDP sockets in enhancing network performance while providing insights into its scalability and limitations through benchmarking.

6 ACKNOWLEDGMENTS

We would like to sincerely thank **TA Naveen Tiwari** for his valuable guidance and much-needed insight for approaching the project in a structured manner by setting up a clear workflow for the same. We plan to coordinate our progress with him on a weekly basis. We are grateful to him for taking out time to meet us and share useful resources.

REFERENCES

- [1] Linux Kernel Documentation, “AF_XDP Official Documentation.” https://www.kernel.org/doc/html/latest/networking/af_xdp.html.
- [2] H. P. N. Programming, “Recapitulating af_xdp.” <https://medium.com/high-performance-network-programming/recapitulating-af-xdp-ef6c1ebeat8>.
- [3] XDP Project, “The eXpress Data Path (XDP) inside the Linux kernel.” <https://github.com/xdp-project>.
- [4] DNS-OARC, “dnsperf: Gather accurate latency and throughput metrics for Domain Name Service (DNS).” <https://github.com/DNS-OARC/dnsperf>.
- [5] Libvirt Project, “libvirt: An Open-Source API, Daemon, and Management Tool for Platform Virtualization.” <https://libvirt.org/>.