Imp Points

# Intro:-

1:-Traffic Increase   
2:- Line Speed Packet Processing  
3:- Software Optimization, Testing and Hardware Optimization and Testing   
4:-The final outcome come from the comparative analysis of different experiment.

5:- **XDP ----------------------> Firewall**

**DPDK ----------------> Router**

# Problem:-

1:- Device are available in the Market that can process the packet but   
 Operating Systems cannot meet with the speed due to

**Overheads due to the Architecture of Network Stack**

# DPDK:-

Kernel bypass approach, which needs control of a NIC to user\_space programs to greatly reduce the overhead introduced into the kernel by things like   
  
1:- Context Switching  
2:- Network Layer Processing  
3:-Interruptions

When we work on Higher Networking Speeds ( 10 GBS or Higher) , these things all become the relevant.

**Size of the Packets:- Min :- 64 Bytes Max:- 1500 Bytes**

**Q: - Why XDP is better than DPDK?**

Ans:- XDP is prominent over the bypassing techniques because XDP offer all others features that always compile with networking.   
 These feature include retaining Kernel security and management compatibility, selectively utilization existing kernel stack features as needed.  
 Providing a stable programming interface and complete transparency to application.

Imp devices information   
   
**1:-** **PCIE: - Peripheral Component Interconnect Express.**  
**2:- CPU:- CPU Caches , Throughput of an Ethernet wire.  
3:- SMP:- Symmetric Multiprocessing.**   
 Involves two or more processing unit on a single system which will run the same operating system , share a common memory and input/output devices for example hard drivers and devices.

# NUMA Node:- ( Non Uniform Memory Access)

It is designed in SMP architecture which states that CPU should have dedicated space in the memory which can be accessed much faster than the other due to it’s proximity.

# DMA :- (Direct Memory Access)

Direct Memory Access is a technique to avoid having to make a CPU intervene between an input/output device and memory to copy data from an another.

# Profiling:-

Getting record of a system.  
 **Perf, mpstate, PCM (Process Control Monitor)**

Standards approaches of Data Validation and Data Evolution.  
  
1:- Reprouctiablity   
2:- Granulation   
3:- Interpretation

# Q:- How XDP Works?

Ans:- We attach ebpf program to an interface (that allows the custom programming) to execute the program when required. The interface is called the **Hook.**

The registration of certain events can be allowed by the hooks.

Here we are looking for the kernel XDP hook to which an ebpf program can be attached; ebpf program executes when that event occur which is register in parallel to the program. So in the computive networking common event are receiving and sending packet but we are mainly focus on the receiving.

**XDP progress returns an action, which represents what happened with the packet after program exitsts.**

XDP\_ABORT -----> Error Drop Tables   
XDP\_DROP -------> Drop Packets   
XDP\_PASS --------> Allow further processing by the kernel stack.  
XDP\_TX ----------> Transmit to the interface it comes from.  
XDP\_REDIRECT -----> Transmit packet to another interface.  
  
  
XDP Modes of Operations

**XDP\_NATIVE:-** XDP programs are executed a re directly by the driver.  
**XDP\_GENERIC:-** Operating system itself do the XDP execution, emulating native execution.   
**XDP\_OFFLOAD:-** The ebpf program off-loaded to compatible program NIC’S.

# DPDK: - (Data Plane Development Kit)

Set of libraries for implementing totally user space drivers for Network Interface Controller. It provides with a framework and common api for archiving higher performance.

DPDK has five core components

**1:-** **EAL (Environment Abstraction Layer):-**   
 For accessing the low level resources like core, memory.  
**2:- MBUF:** -  
 A data structure carry network packets.  
**3:- MEMPOOL: -**  
 The Library for creating memory for packet allocation.  
**4:-RING: -**  
 The library manage messages between threads, cores etc.  
**5:-TIMER:-**  
 For asynchronies call back functions.

***DUT: - Device Under Testing***

# Data Evaluation Rules:-

**Reproducibility: -** The experiment must yield the same results by being ran over the same setting.

**Granularity: -**  
 The level of details in the data must balance between being detailed enough to detect meaningful changes and general enough to complete experiments in a reasonable frame.

**Interpretation: -** Extra profiling tests are conducted to ensure results are conducted to ensure results are accurately interpreted and not compromised by complicating program.