



# OVS

Open vSwitch

**High-performance packet parsing in OVS:  
Using AVX512 for Miniflow Extract & Protocol Specific Hashing**

**Kumar Amber  
Harry van Haaren  
Intel**

# OVERVIEW

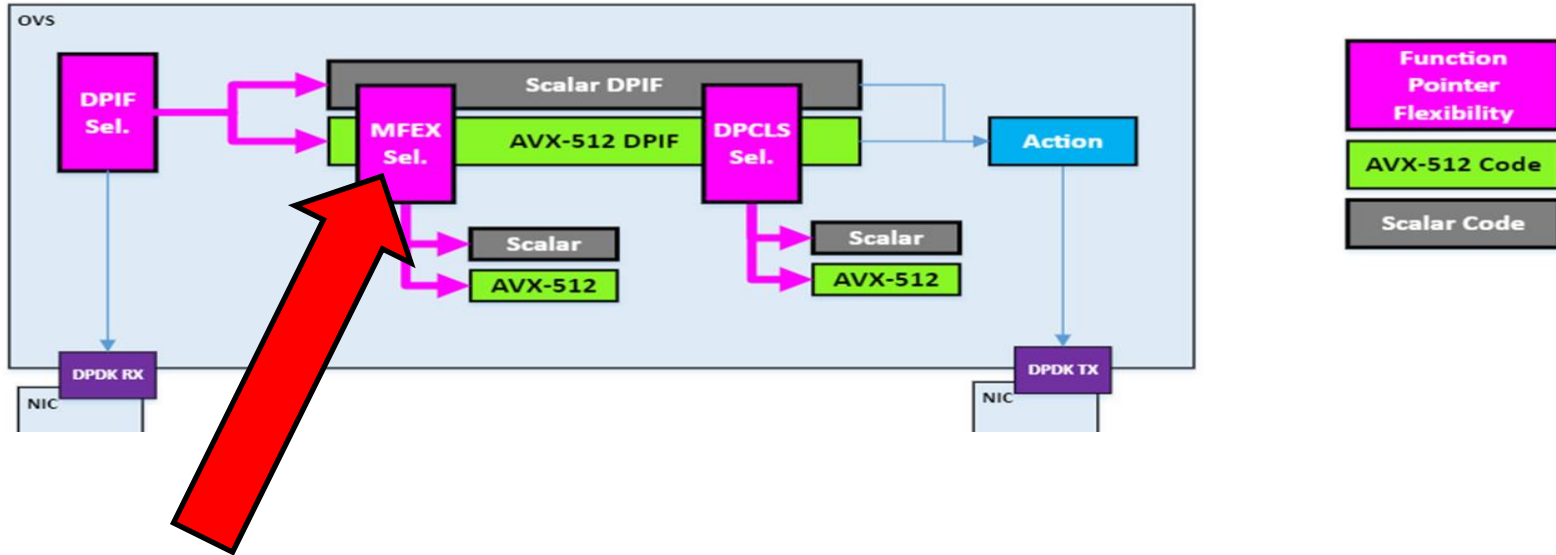
**Two optimizations discussed in this talk:**

- 1) Optimizing Miniflow Extract**
- 2) Protocol Aware Hashing**

# Miniflow Extract

- **What is “miniflow extract”?**
  - Parses packet, extracts metadata, builds miniflow
    - Represents all metadata OVS uses
    - Used in datapath for classifying packets/rules etc
- **Designing Optimized Code**
  - Optimizing MFEX with AVX512 SIMD Instructions
  - Scaling both Traffic Profiles and CPU ISA
- **Validation and Unit testing**
  - Automated approach to ensure correctness

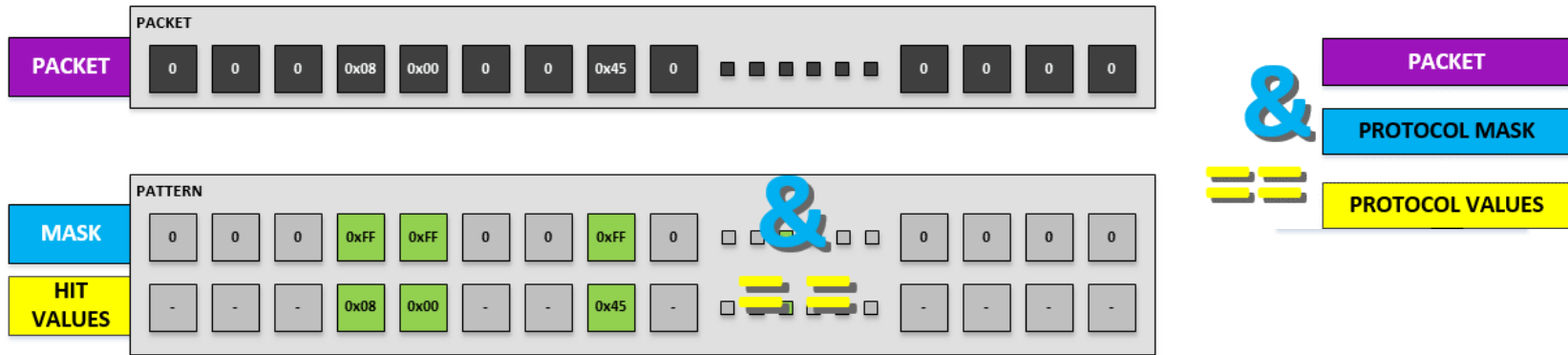
# Using the Optimized MFEX



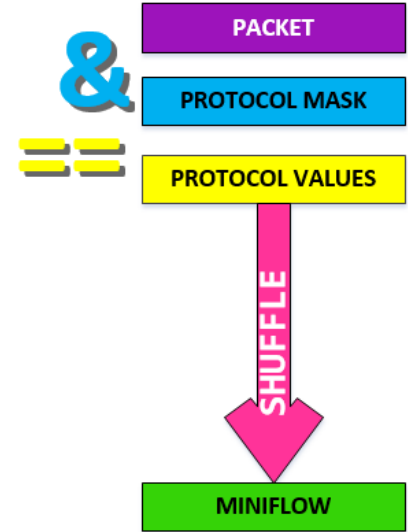
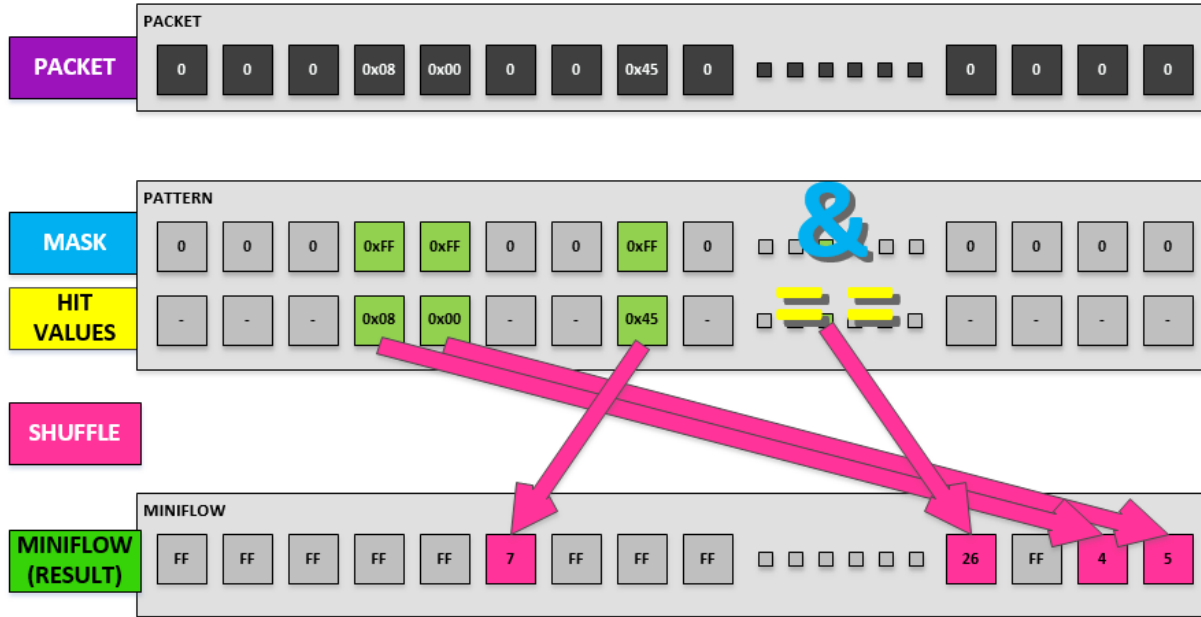
```
$ dpif-netdev/miniflow-parser-set scalar  
$ dpif-netdev/miniflow-parser-set avx512  
$ dpif-netdev/miniflow-parser-set autovalidator
```

```
# default  
# only works if CPU ISA is available  
# validate implementations are ==
```

# MFEX Basics : The AVX512 Shuffle



# MFEX Basics : The AVX512 Shuffle



# The AVX512 MFEX : Static Data

- **Processing Ether/IPv4/UDP**
  - Protocol Specific Data
    - **mask**, **value**, and **shuffle**
  - Extra Metadata
    - **Miniflow bits**
      - known from Ether/IPv4/UDP
    - **DP Packet properties**
      - Packet Lengths
      - L2/L3 Offsets

PROTOCOL MASK

PROTOCOL VALUES

SHUFFLE

# The AVX512 MFEX : Static Data

- **Processing Ether/IPv4/UDP**

- Protocol Specific Data
  - **mask**, **value**, and **shuffle**
- Extra Metadata
  - **Miniflow bits**
    - known from Ether/IPv4/UDP
  - **DP Packet properties**
    - Packet Lengths
    - L2/L3 Offsets

} **STATIC DATA!**  
(Per Traffic Profile)

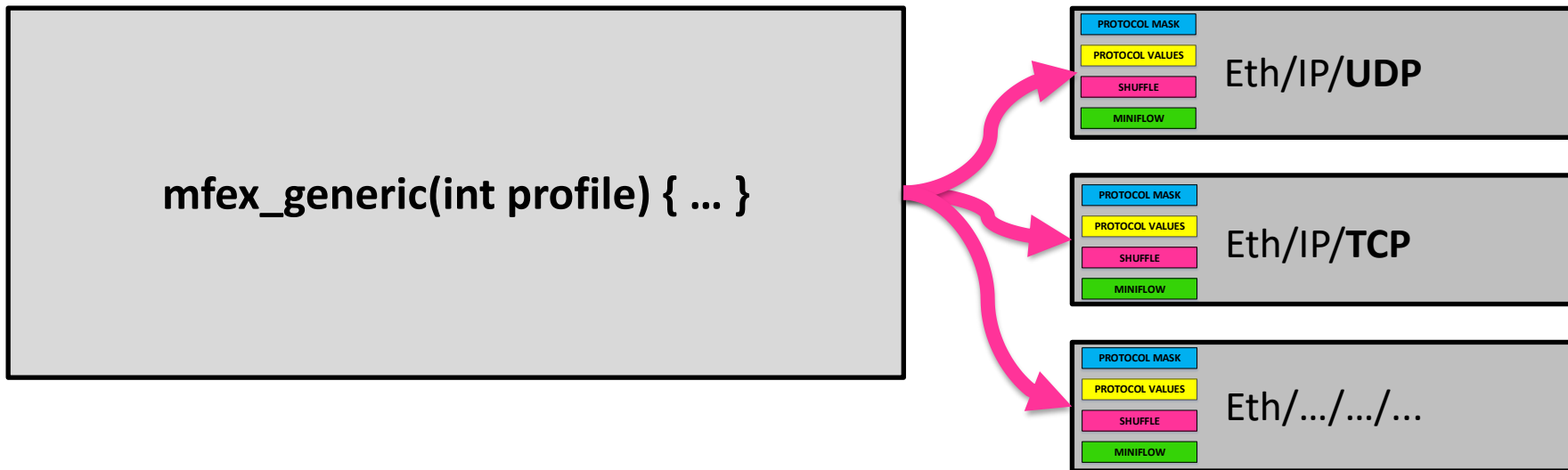
PROTOCOL MASK

PROTOCOL VALUES

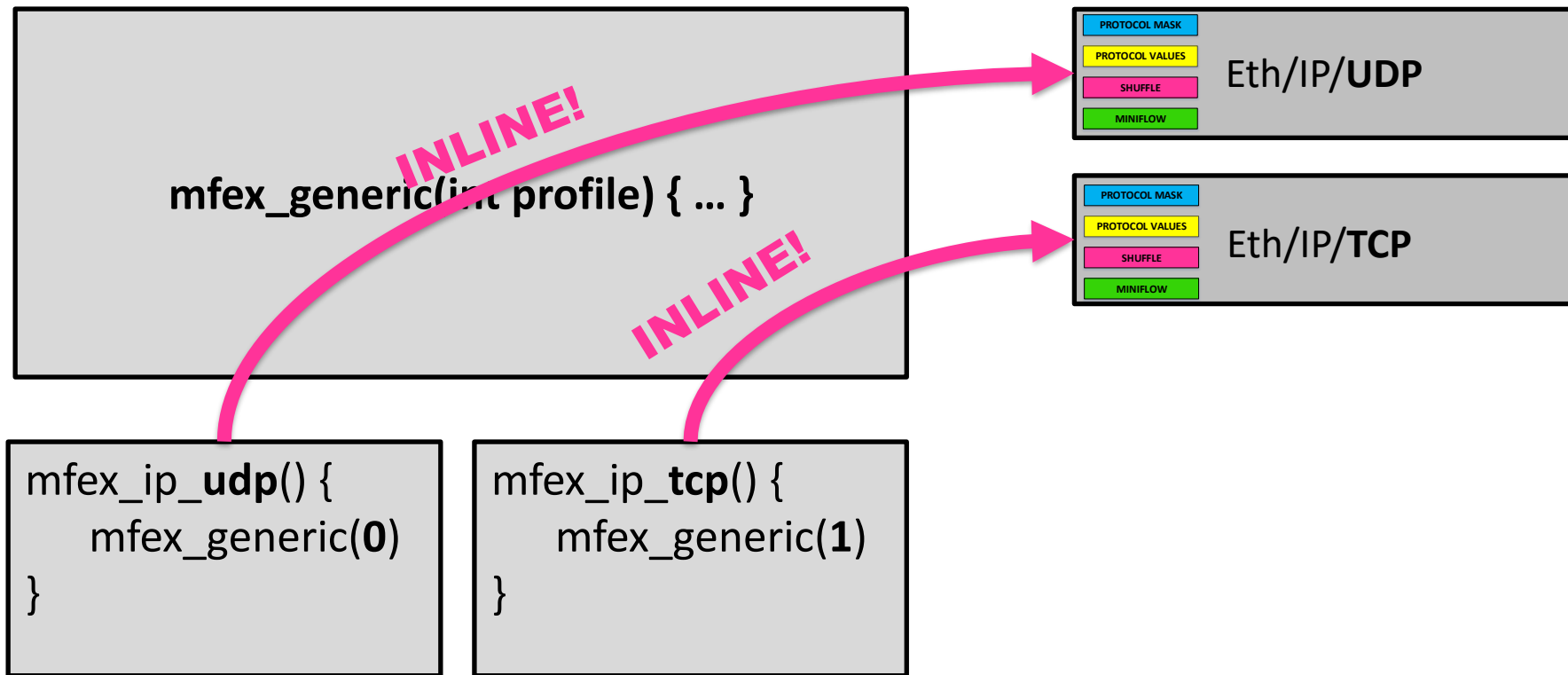
SHUFFLE



# Performance & Scaling AVX512 MFEX



# Performance & Scaling AVX512 MFEX



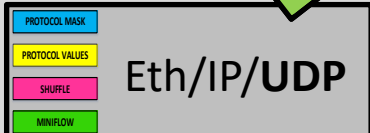
# Performance & Scaling AVX512 MFEX

  
mfex\_generic(profile) { ... }

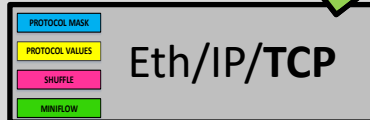


Generic Function **DOES NOT**  
execute on datapath!

  
mfex\_ip\_udp()



  
mfex\_ip\_tcp()



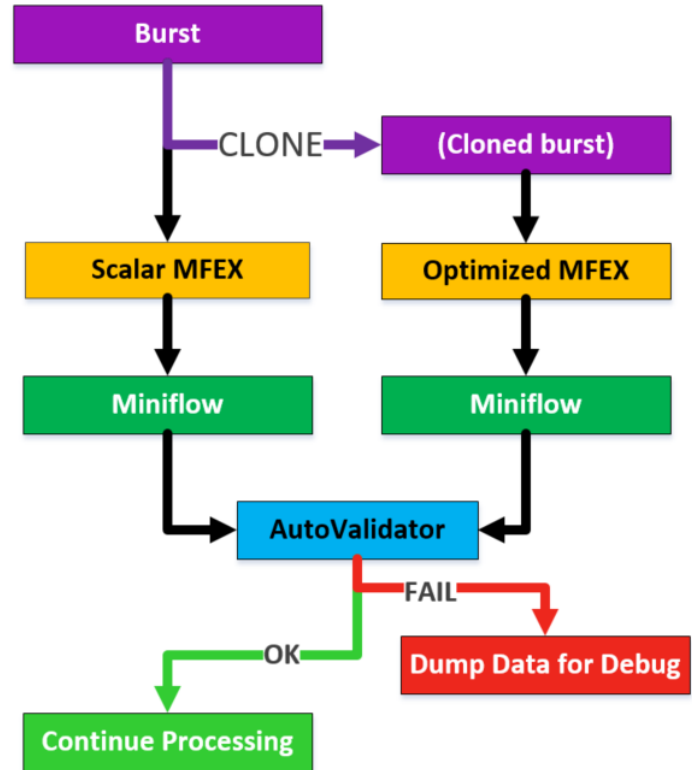
Inlined version **DOES**  
execute on datapath

- **Specialized to profile!**
- **Specialized to CPU ISA!**

# Validation And Testing

- **Autovalidator Compares Miniflows**
  - Ensure Miniflow is always same
  - Confidence in all implementations
- **Fuzz testing**
  - Try make the autovalidator fail!

```
$ dpif-netdev/miniflow-parser-set autovalidator
```

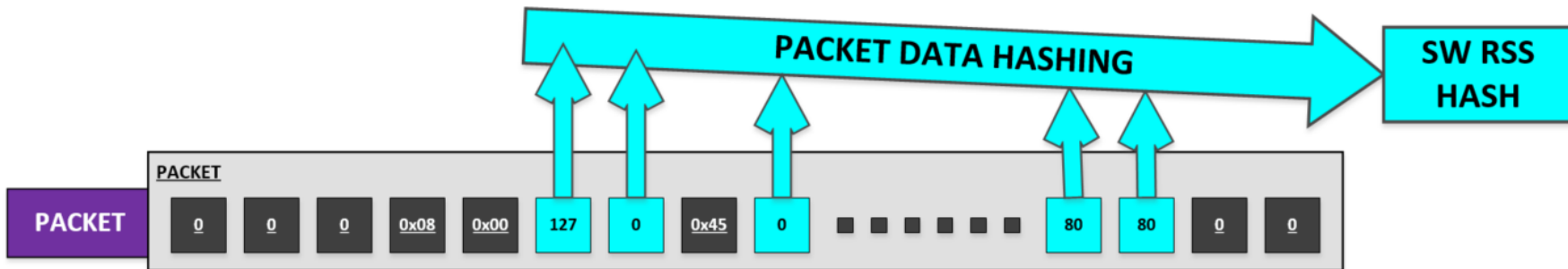


# Hashing : Overview

- **What hashing is done? Software RSS**
  - L3 IP src/dst
  - L4 UDP/TCP src/dst ports
- **When does this need to be done?**
  - Guest to Network (southbound vhost)
- **What is Protocol Aware Hashing?**
  - Re-use MFEX profile knowledge
  - AVX512 “knowledge” to Optimize Hashing

# Protocol Aware Hashing

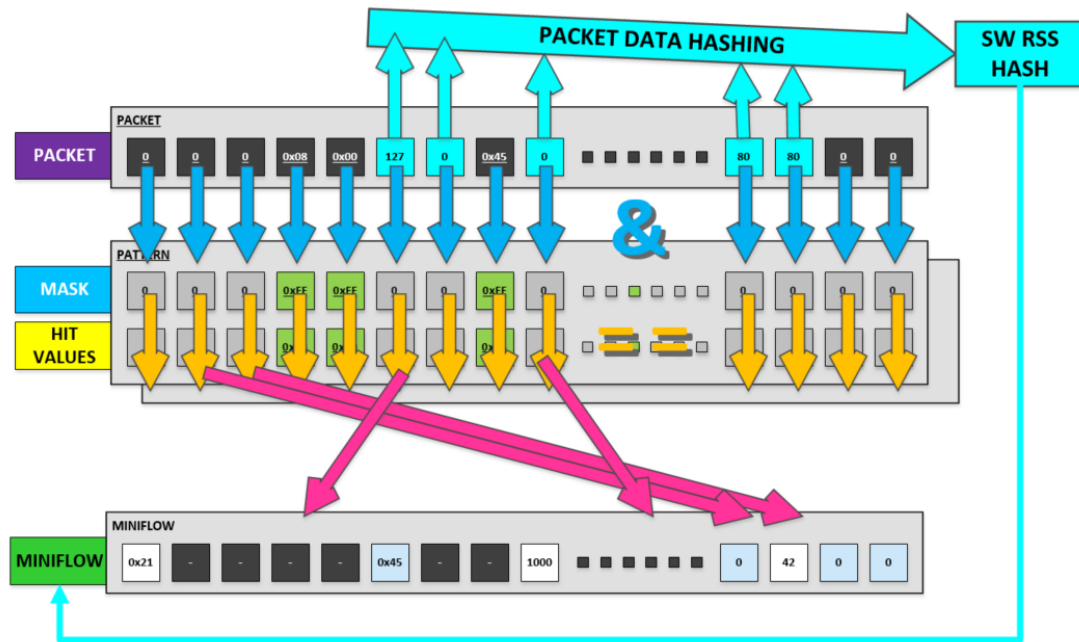
- **Known Packet Data Layout**
  - From AVX512 Profile Match
  - Load Hash data *directly* from packet  
(not from miniflow like scalar)




# Performance Benefits : Overall

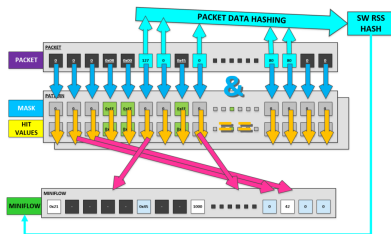
## Instruction Parallelism


- AVX512 SIMD
- Packet Hashing



# Code for 2.17!

 **Open vSwitch**  
2.16



 **Open vSwitch**  
2.17

## MFEX IPv6 & Protocol Aware Hashing

<http://patchwork.ozlabs.org/project/openvswitch/list/?series=263303>

Reviews & Comments welcomed





**! Thank You !**  
**? Questions ?**

Kumar Amber [kumar.amber@intel.com](mailto:kumar.amber@intel.com)

Harry van Haaren [harry.van.haaren@intel.com](mailto:harry.van.haaren@intel.com)