

Testbed for header compression implementation

Mario Mornar , Alen Pezelj* , Julije Ozegovic***

*HT-TKC Split

**University of Split, FESB Split

Split, Croatia

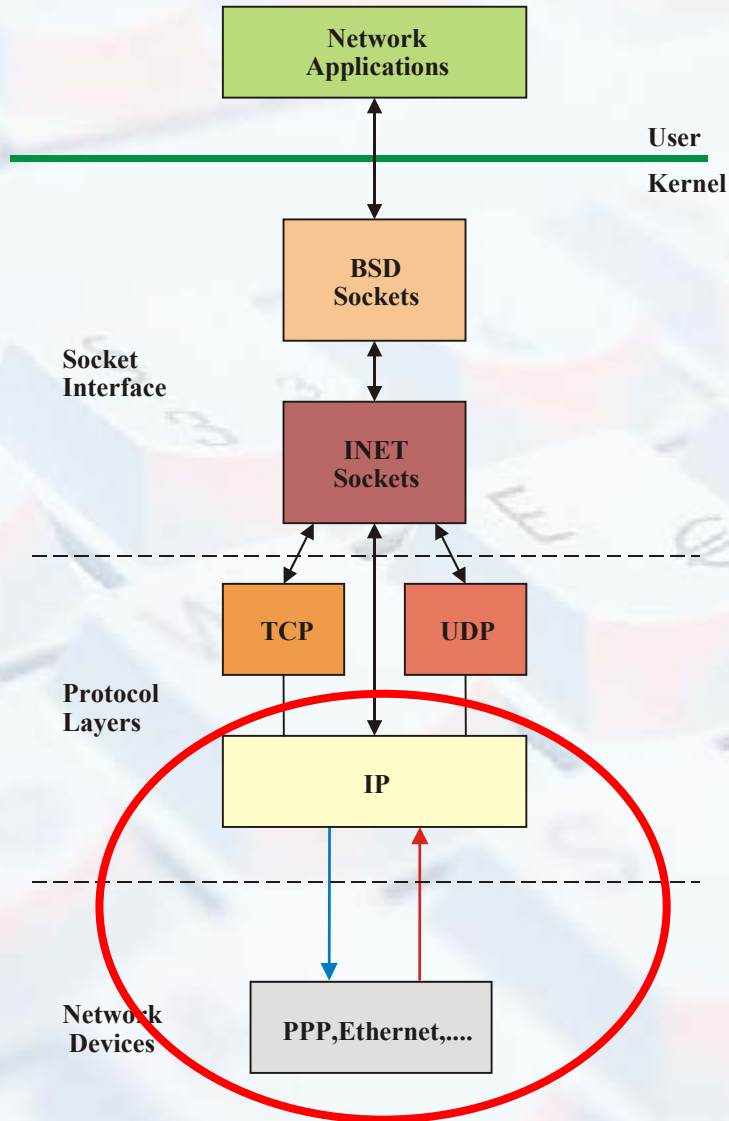
E-mail: mario.mornar@ht.hr, alen.pezelj@ht.hr,
julije.ozegovic@fesb.hr

Introduction

EPIC-Lite compression scheme implementation required a flexible test platform with following requirements:

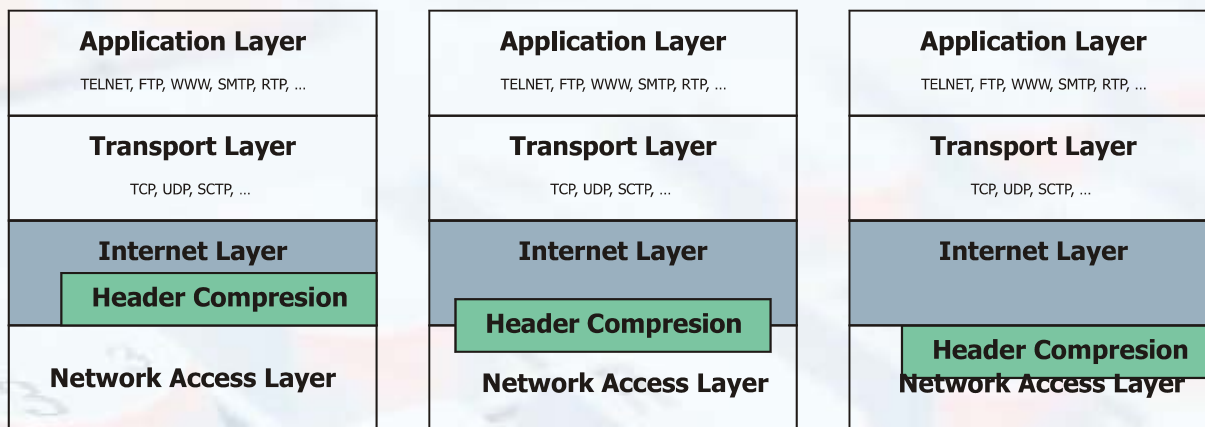
- *Functionality*
 - *Flexibility*
 - *Implementability*
 - *Transparency*
-
- To achieve that goal a new hybrid testbed architecture was introduced

Linux TCP/IP network architecture



- Network applications communicate with the protocol stack using sockets
- Multiple network devices can be attached as well as multiple protocol stacks

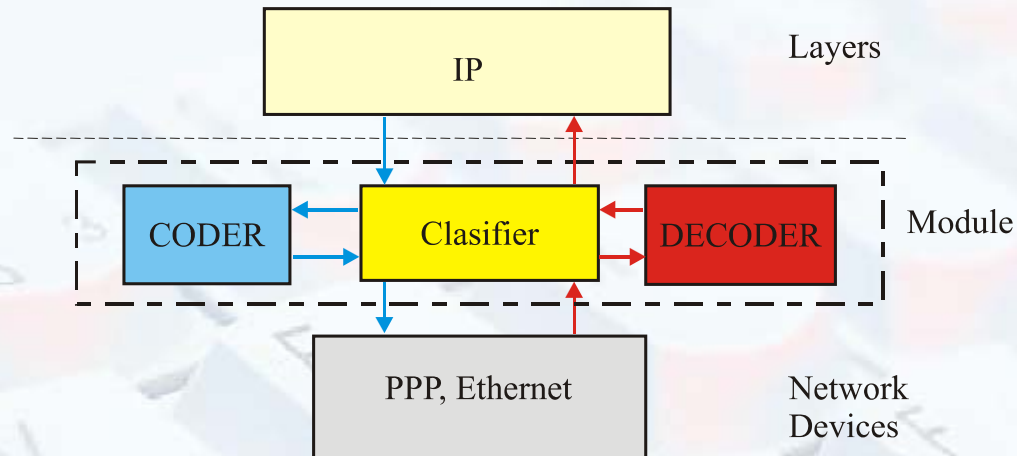
Protocol issues



Position of header compression function

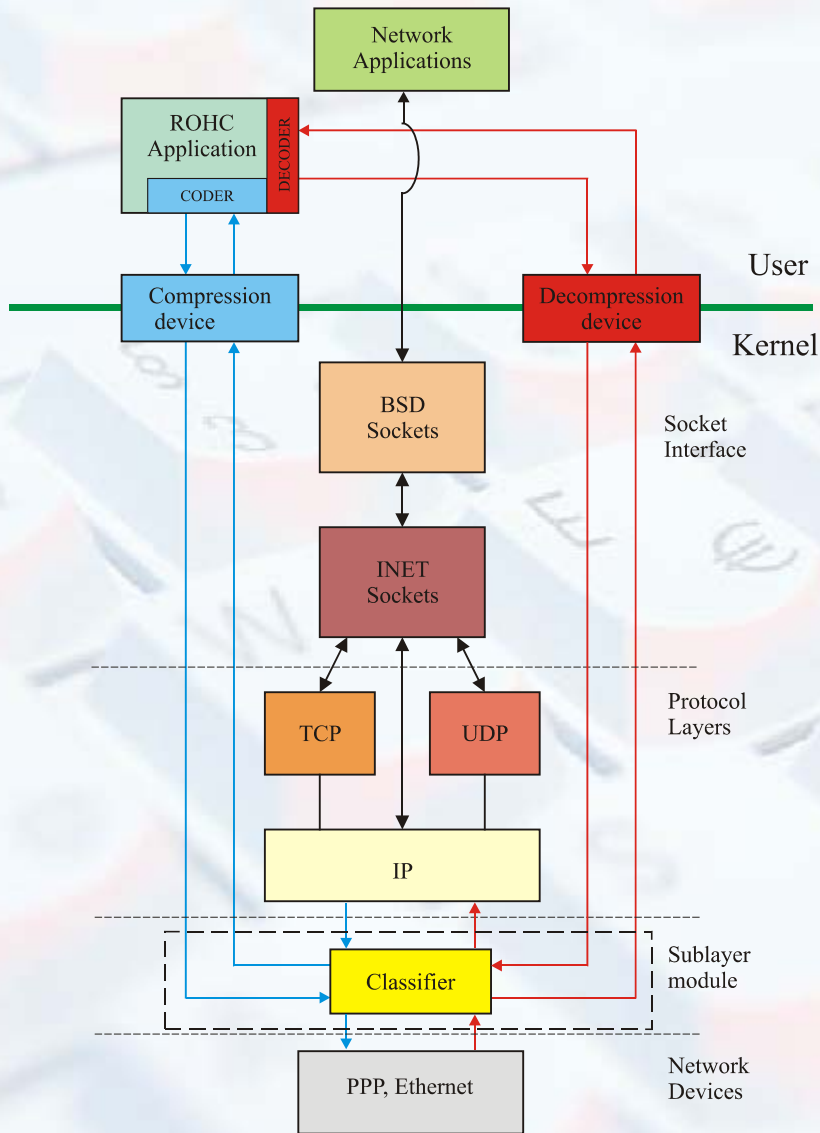
- integrate compression inside the protocol stack
- put compression inside a sublayer between network and data link layer
- integrate compression into device driver module

Proposed elementary architecture



- Module sublayer provides elementary mechanism to integrate header compression
- Coder should change the protocol ID, ROHC framework is not assigned one

Advanced sublayer architecture



- Interfaces toward user space added
- Compress/Decompress run in user space
- Very limited functionality needed in sublayer module

Implementation issues

- Sublayer module performs complex connections towards protocol stack, network module and device interface
- Standard attachment functions had to be modified:

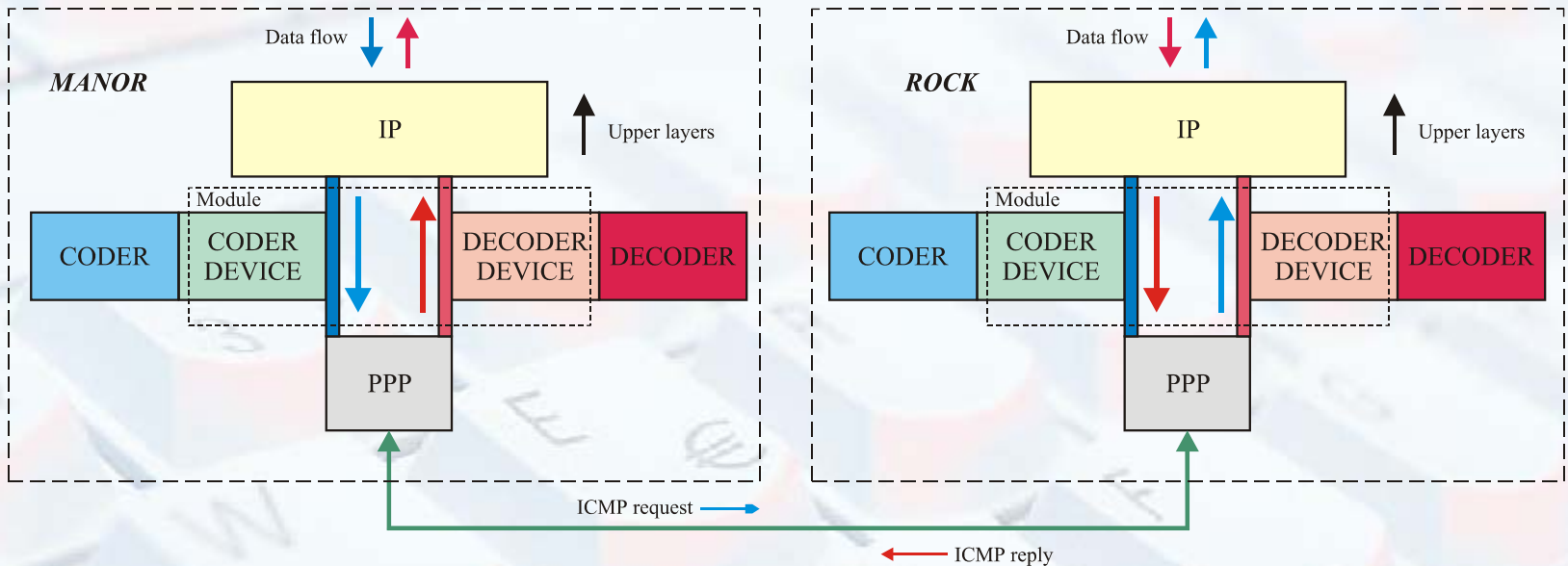
init_module() - *starting all necessary initializations*

hard_start_xmit() - *send packet to network interface card*

netif_rx() - *send data from NIC to upper network layers*

query_module() - *get table of exported kernel symbols*

Experimental hybrid solution deployment



- Coder and decoder devices initialized
- Simple *Coder* and *Decoder* applications provided, started on A
- PING utility started on A, ICMP requests sent to B
- Packets discarded on B
- *Coder/Decoder* started on B, PING receives ICMP replies from B

Conclusion

- The developed hybrid solution is especially feasible for real network protocol tests
- There is no need to provide complex negotiation procedures
- Test application run in user space, various compression profiles can be used
- EPIC-Lite functionality can be approved in file to file compression and decompression of captured flows, and then quickly attached to the testbed for real network experiments.