

From textbook 'Conceptual Computer Networks'

© 2016-2021 by José María Foces Morán & José María Foces Vivancos. All rights reserved.

PF_PACKET SOCKETS

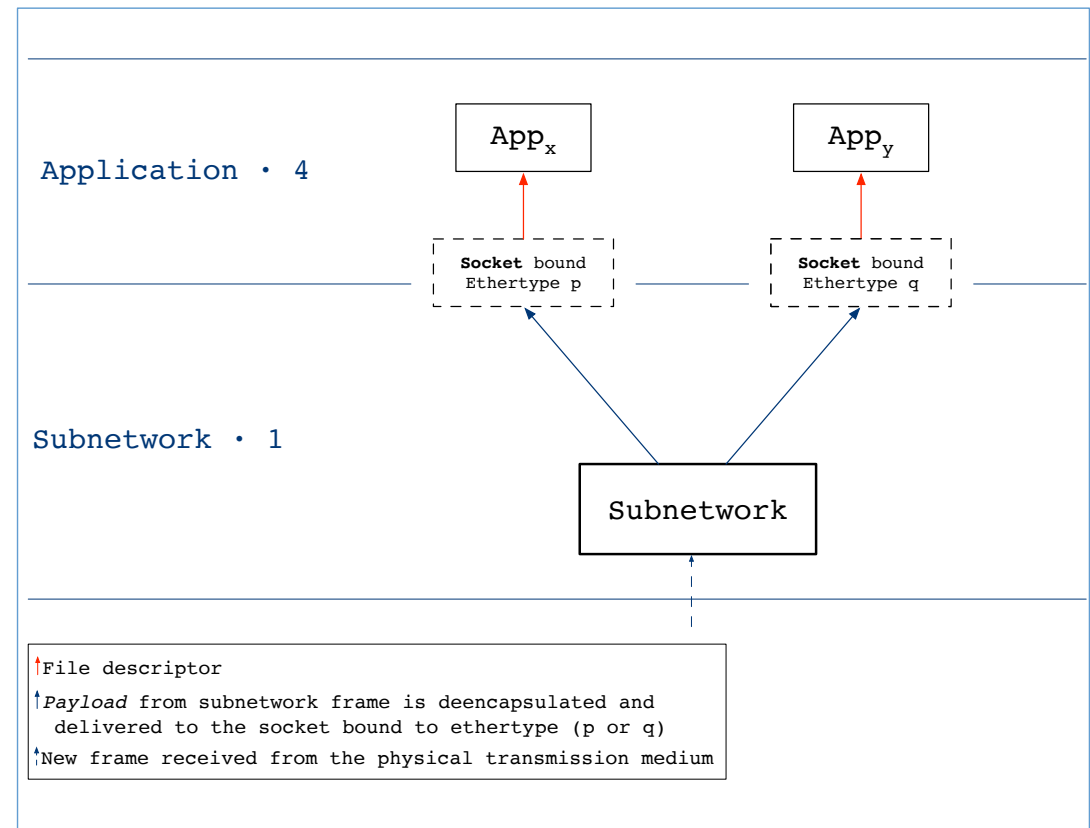
V 1.1

Next Generation Internet, Universidad de León, 2018

PF_PACKET Sockets

2

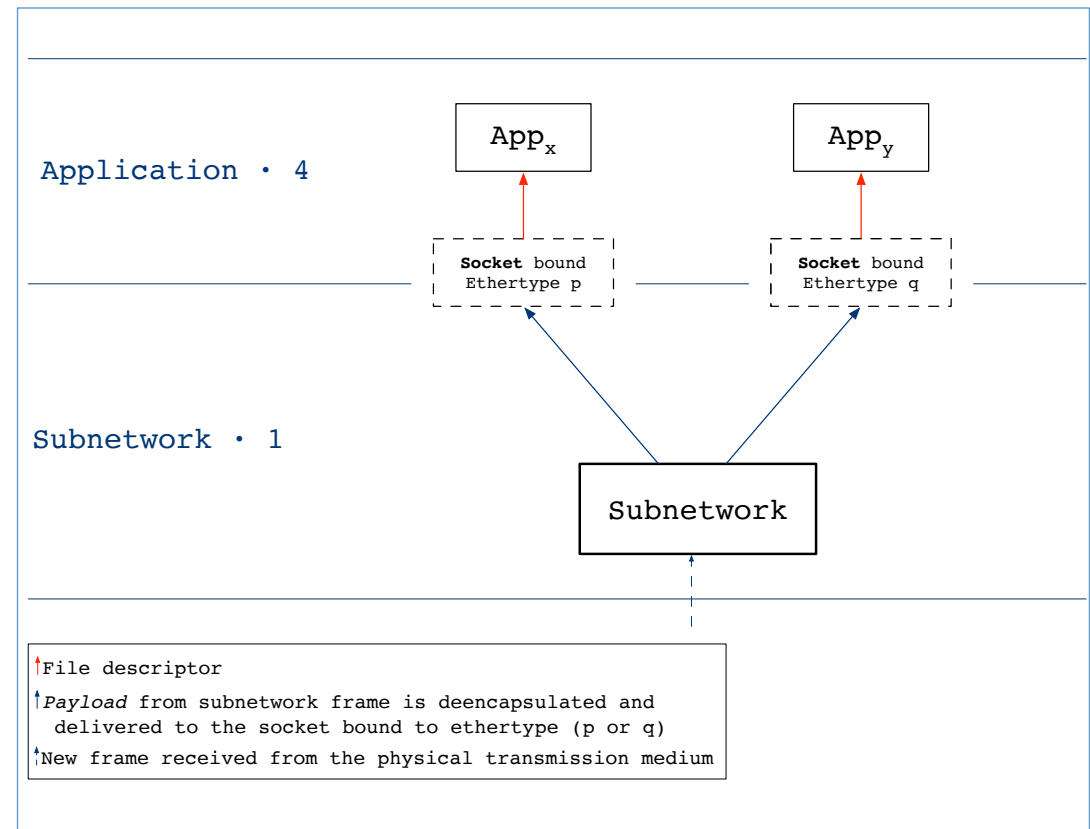
- Applications can register for receiving frames directly from the subnetwork layer
- Protocol family PF_PACKET is exclusive of Linux



Two types of PF_PACKET sockets

3

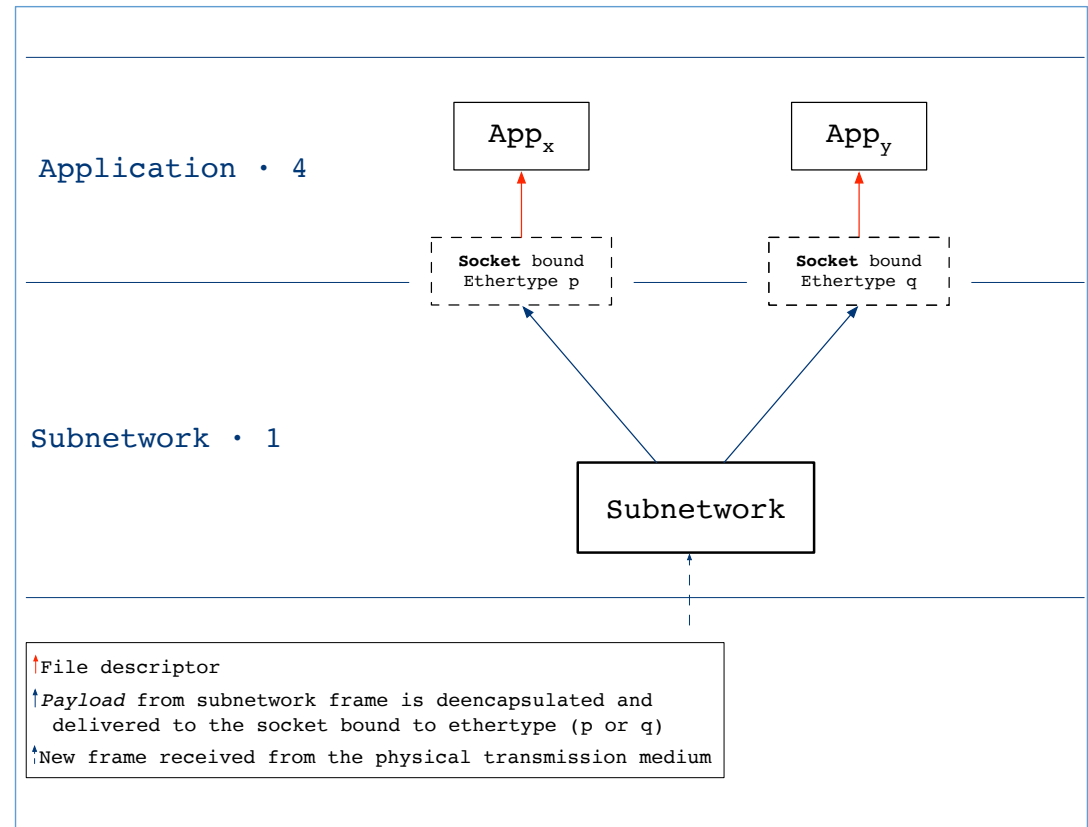
- Datagram (SOCK_DGRAM)
 - Only the frame payload is delivered to the application
- Raw (SOCK_RAW)
 - Both, the frame header and the frame payload are delivered to the application



Sending with PF_PACKET sockets

4

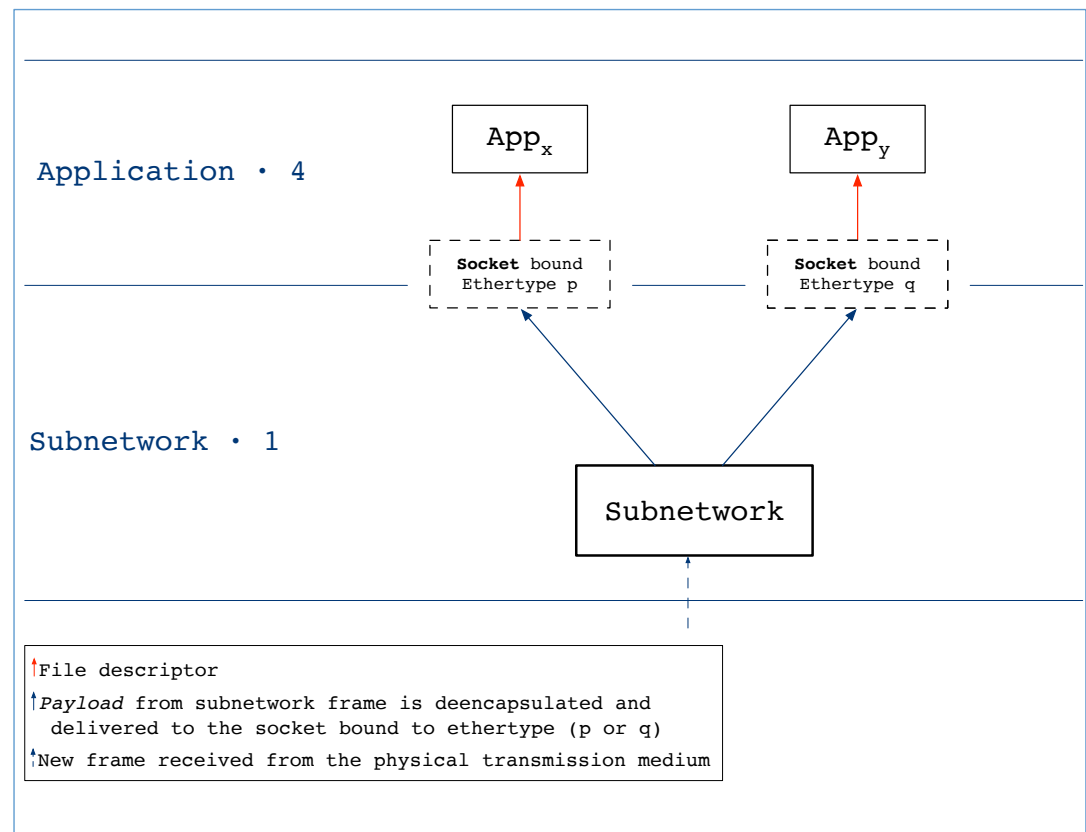
- Application can only send the full frame:
 - ▣ The header
 - ▣ And the payload



Link-layer header for PF_PACKET sockets

5

- Application must send the full frame:
 - ▣ Sending through this type of socket includes a **link-layer header** which fields are **derived** from the `sockaddr_ll` passed on to the write call made by the application.



Link-layer header for PF_PACKET sockets

6

- Assume a PF_PACKET socket is created; sockaddr_ll fields meaning follows:
- **sll_family** must be assigned PF_PACKET
- **sll_protocol** is the link-layer protocol field value (Type or Ethertype in the case of an Ethernet frame) that we wish to have demultiplexed. If the value of sll_protocol is htons(ETH_P_ALL), then all protocols encapsulated in the link-layer frame will be accepted by the socket. This precludes the socket having IP_HDRINCL option set, in which case, receiving all IP's protocols is not supported. Standard values for sll_protocol are declared in /usr/include/linux/if_ether.h.
- **sll_ifindex** the underlying network interface index
- **sll_hatype** (For ARP)
- **sll_pkttype**
- **sll_halen** (For ARP)
- **sll_addr[8]** is used for storing the interface's MAC address htons() function call reorders the bytes of a short int to the network byte ordering format (Network byte order). This byte-ordering is usually applied in networking and is the same as the big-endian ordering of MIPS and other microprocessors.

```
struct sockaddr_ll {  
  
    unsigned short    sll_family;  
    __be16           sll_protocol;  
    int              sll_ifindex;  
    unsigned short    sll_hatype;  
    unsigned char     sll_pkttype;  
    unsigned char     sll_halen;  
    unsigned char     sll_addr[8];  
  
};
```

