BPF Code Inspection

FreeBSD Developer Summit
Ottawa, Canada
May, 2007

BPF's Purpose

- Provide a raw interface to data link layers in a protocol independent fashion
  - Being able to capture incoming and outgoing packets off the wire on several layers
  - Being able to transmit raw packets to a network: commonly Ethernet, could be others
- Common uses, tcpdump, dhclient…

Code Organization

- Main BPF subsystem code:
  - **sys/net/bpf.[ch]**
  - **sys/net/bpfdesc.h**
  - **sys/net/bpf_compat.h**
- BPF filter machine
  - **sys/net/bpf_jitter.[ch]**
  - **sys/net/bpf_filter.c**
  - **sys/$arch/$arch/bpf_jit_machdep.[ch]**
    - (Only amd64 and i386 are supported currently)
- VLAN tapping functions
  - **sys/net/if_ethersubr.c**
  - **sys/net/ethernet.h**

FreeBSD®

Main Code Paths

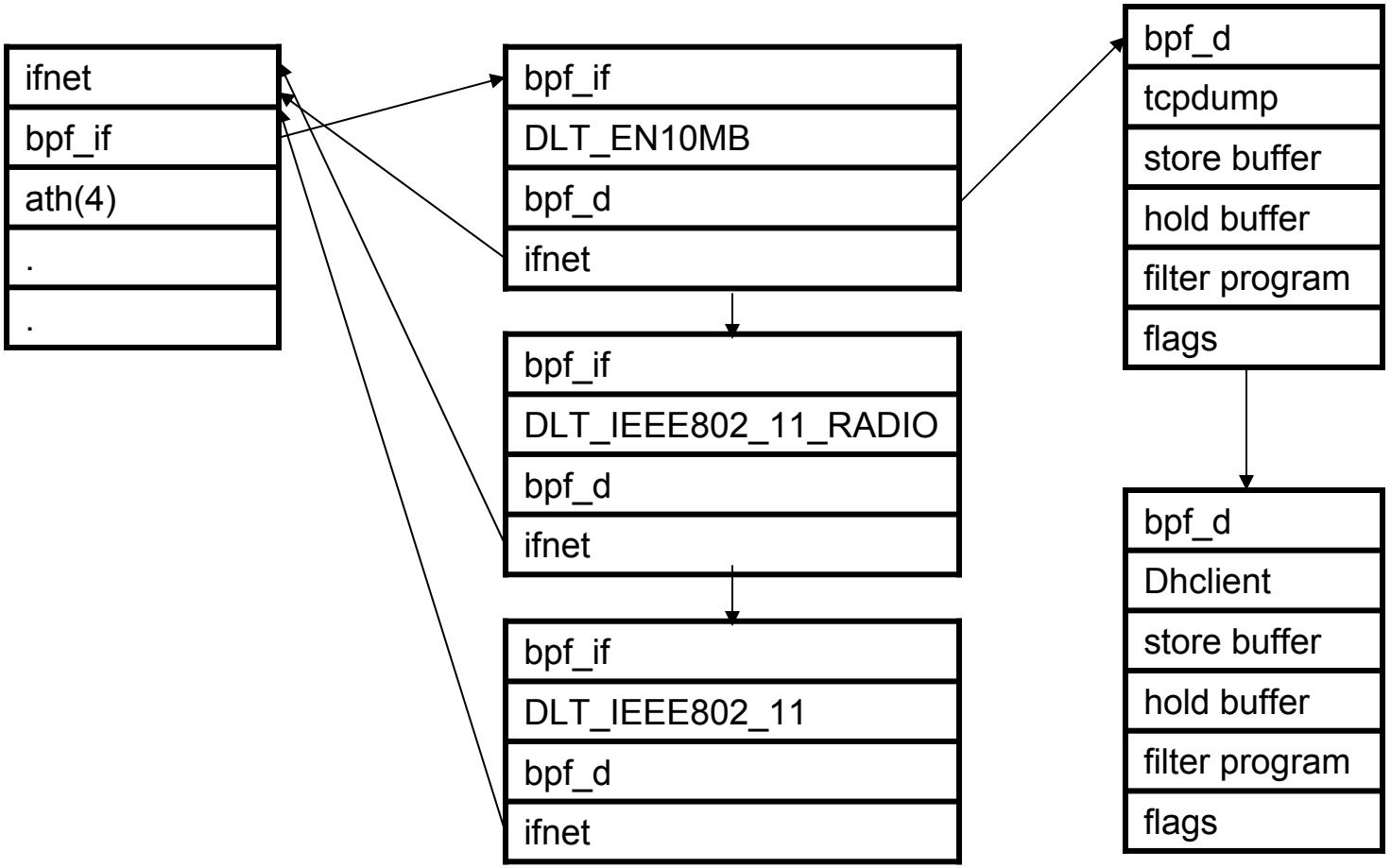
- Tapping of incoming and outgoing packets:
  - Incoming, typically done in ether_input()
  - Outgoing, usually in driver's if_start() routine
- Transmission of raw packets to the network: writes to /dev/bpfX (bpfwrite())

Main BPF Structures

- Two main structures
- One which represents hardware interface, which peers are associated with:
  - **`struct bpf_if (sys/net/bpfdesc.h)`**
- Another which represents a BPF peer or descriptor:
  - **`struct bpf_d (sys/net/bpfdesc.h)`**
  - **`struct xbpf_d (sys/net/bpfdesc.h)`**

FreeBSD®

BPF Object Layout



FreeBSD

Code To Review

- Tapping bpf.h bpf.c:
  - ETHER_BPF_MTAP()/BPF_MTAP()
  - bpf_mtap()
  - catchpacket()
- Packet injection bpf.c:
  - bpfwrite()
  - bpf_movein()