Using netfilter

Basic configuration for a simple packet filtering implementation

Architectue

- Uses the kernel’s netfilter interface

- Rules are grouped into chains
- Chains are grouped into tables
- Each table is assigned to a family

Sample nftables.conf accepting all packets

```
table inet filter {
        chain input {
                type filter hook input priority filter; policy accept;
        }

        chain forward {
                type filter hook forward priority filter; policy accept;
        }

        chain output {
                type filter hook output priority filter; policy accept;
        }
}
```

Sample nftables.conf accepting just ssh

```
table inet filter {
	chain input {
		type filter hook input priority filter; policy drop;

		tcp dport 22 accept
	}

	chain forward {
		type filter hook forward priority filter; policy accept;
	}

	chain output {
		type filter hook output priority filter; policy accept;
	}
}
```

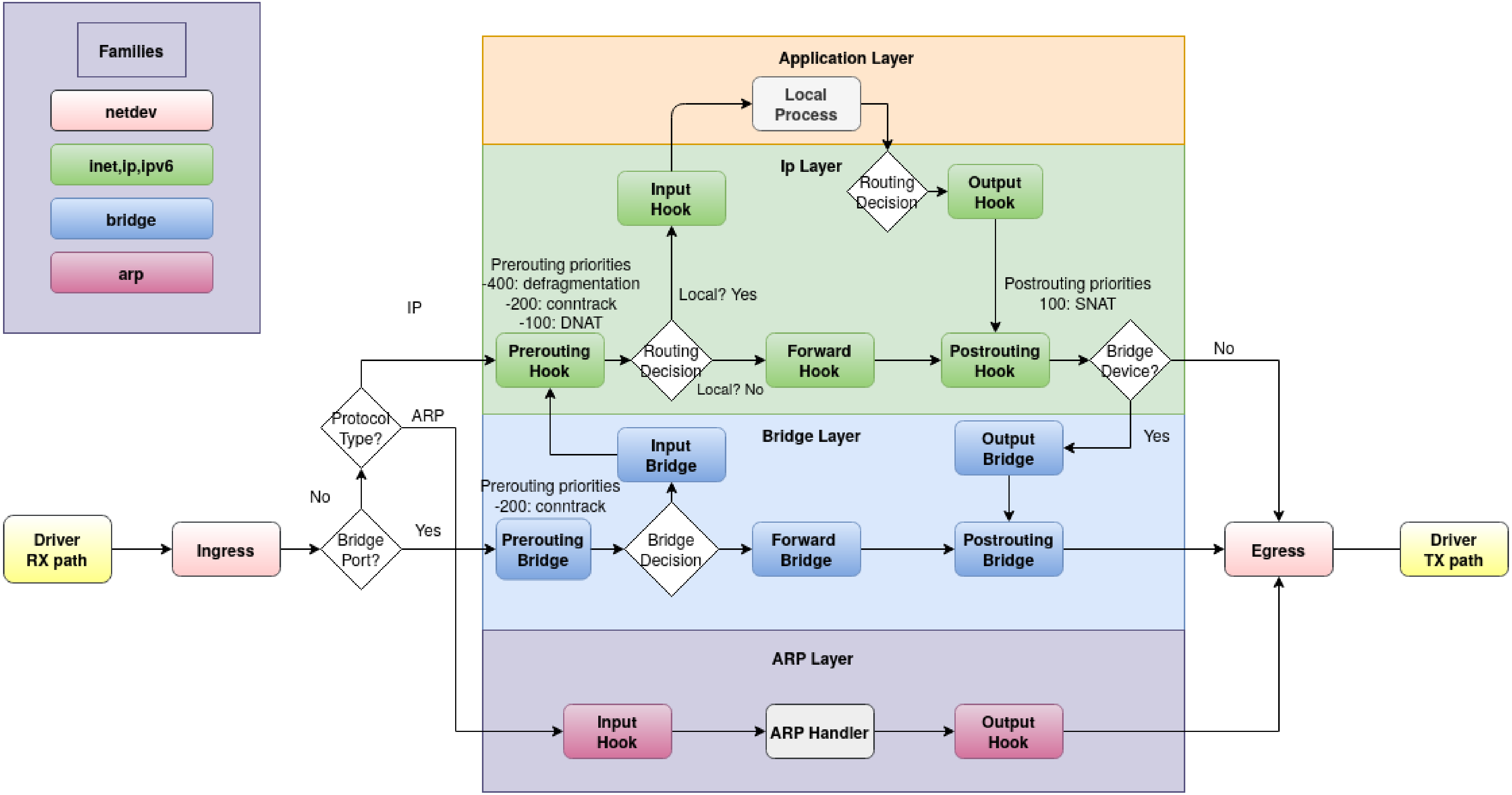
Chain evaluation

- Chains evaluated in order
- Rules evaluated left-to-right
- Rules are a mix of ‘matches’ and actions
- If a match is true, evaluation continues
- Actions executed as encountered

Sample rule

```
tcp dport 22 iif enp0s8 log accept
```

Hooks



Families

- 🟥 netdev
- 🟩 inet,ip,ipv6
- 🟦 bridge
- 🟪 arp

Application Layer

Local Process

Ip Layer

Input Hook

Routing Decision

Output Hook

Prerouting priorities
-400: defragmentation
-200: conntrack
-100: DNAT

Local? Yes

Postrouting priorities
100: SNAT

IP

Prerouting Hook

Routing Decision

Forward Hook

Postrouting Hook

Bridge Device?

No

Local? No

Protocol Type?

ARP

Bridge Layer

Input Bridge

Output Bridge

Yes

Prerouting priorities
-200: conntrack

No

Driver RX path

Ingress

Bridge Port?

Yes

Prerouting Bridge

Bridge Decision

Forward Bridge

Postrouting Bridge

Egress

Driver TX path

ARP Layer

Input Hook

ARP Handler

Output Hook

Examples

Walk through Lab 06b - Firewall