Simple iptables firewall

Network Startup Resource Center
www.nsrc.org

Linux has a built-in firewall

- A good way to limit access to your host
- Has been through many versions!
  - ipfwadm
  - ipchains
  - iptables
  - nftables
- Today most distributions use iptables

Packet flow in Netfilter and General Networking

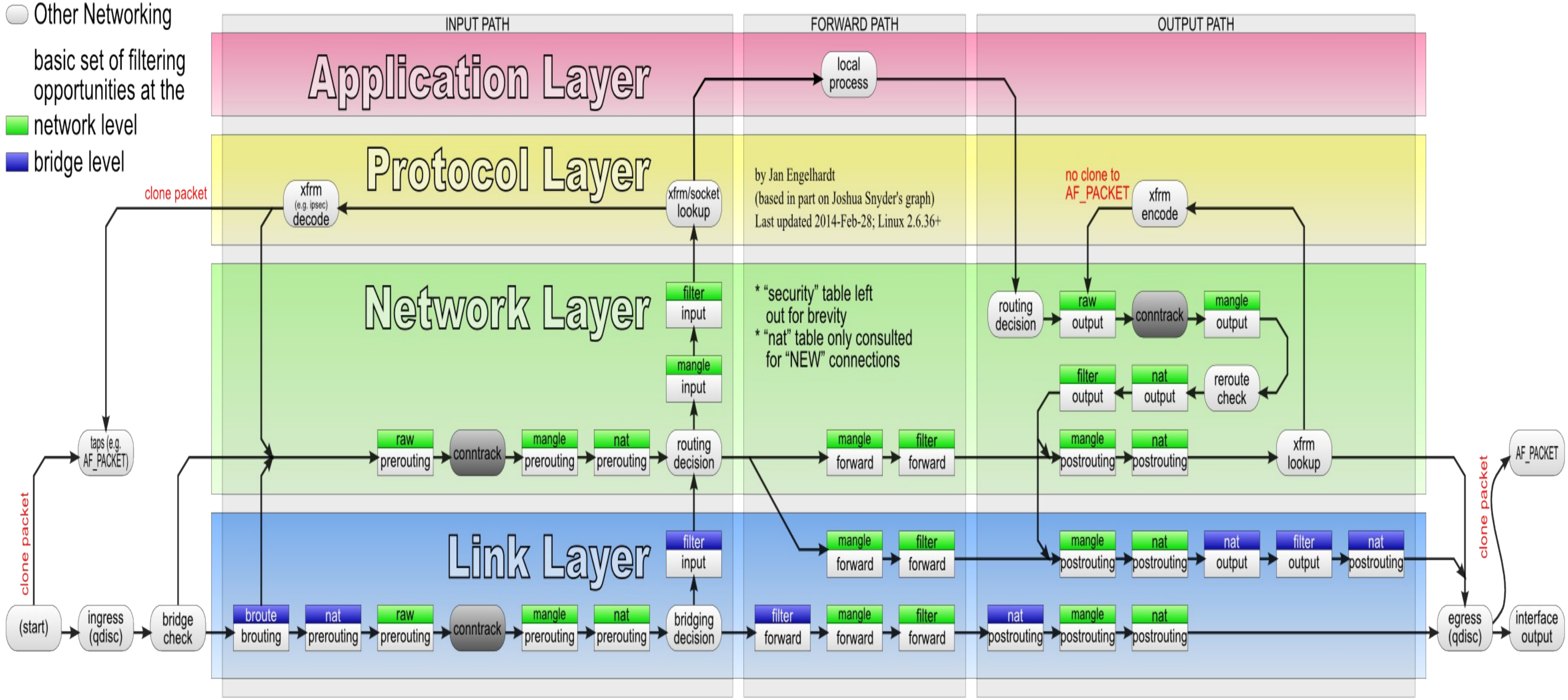
● Other NF parts

○ Other Networking

basic set of filtering opportunities at the

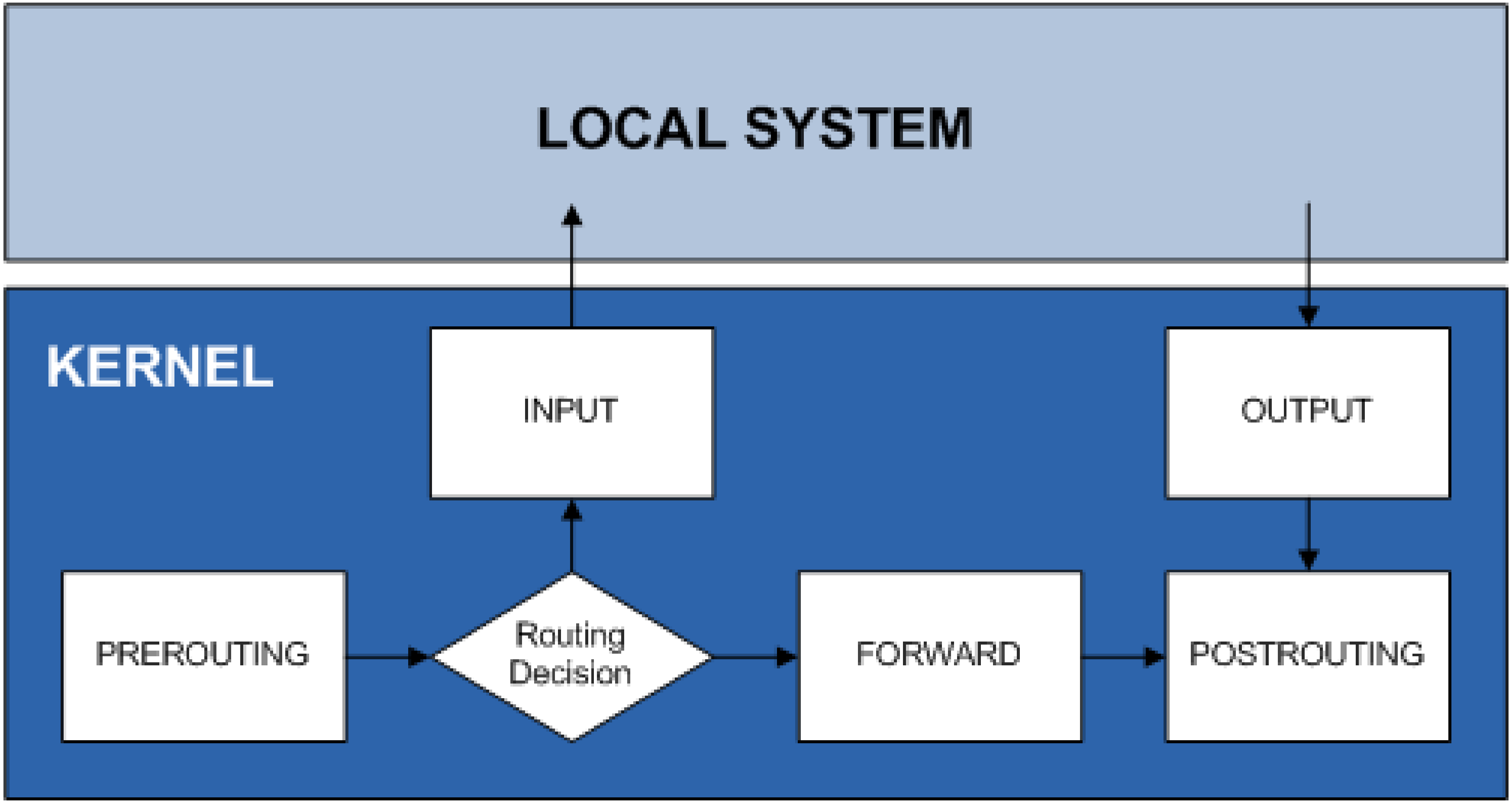
🟩 network level

🟦 bridge level



by Jan Engelhardt
(based in part on Joshua Snyder's graph)
Last updated 2014-Feb-28; Linux 2.6.36+

* "security" table left out for brevity
* "nat" table only consulted for "NEW" connections


LOCAL SYSTEM
KERNEL
INPUT
OUTPUT
PREROUTING
Routing Decision
FORWARD
POSTROUTING

iptables: viewing rules

- **`iptables -L -n -v`**
  - Show the FILTER rules
  - (L)ist rules, (n)o reverse DNS, (v)erbose
  - Separate rules INPUT (packets to the host), OUTPUT (packets from the host), and FORWARD (packets routed via the host)
- **`iptables -L -n -v -t nat`**
  - Show the NAT rules

iptables: writing rules

- It's very difficult to do it properly
  - lots of edge cases
  - certain ICMP packets are essential for correct network operation (e.g. path MTU discovery)
- Instead, use a simpler front-end
- For Ubuntu/Debian: look at **ufw** and **ferm**
- For CentOS/RedHAT: system-config-firewall-tui or firewall-cmd

ufw in operation

```
# install
apt-get install ufw

# show rules
ufw status

# allow all traffic from a given IP
ufw allow in from 128.223.157.19

# allow from all IPs to given service
# (note: "in" is assumed if not specified)
ufw allow proto tcp to 1.2.3.4 port 80
```

ufw configuration storage

- Stored in /etc/ufw/ and /etc/ufw/applications.d/
- The latter has pre-defined application rules
  - you can apply these rules if you want them
  - simplifies application configuration

```
# cat /etc/ufw/applications.d/openssh-server
[OpenSSH]
title=Secure shell server, an rshd replacement
description=OpenSSH is a free implementation of \
  the Secure Shell protocol.
ports=22/tcp
```

Test your firewall rules!

- Check that you *can* connect from authorised addresses and *cannot* connect to blocked ports from unauthorised addresses
- Decide on your firewall policy
  - Default accept: block only certain ports
  - Default deny: open only certain ports
- Beware locking yourself out
  - Have console access available

For more information

- https://help.ubuntu.com/community/UFW
- man ufw