

Linux Administration

Firewalls

Xavier Belanger

**This work is licensed under
a Creative Commons Attribution-ShareAlike 4.0 International License.**

<https://creativecommons.org/licenses/by-sa/4.0/>

You are free to:

- **Share** — copy and redistribute the material in any medium or format
- **Adapt** — remix, transform, and build upon the material for any purpose, even commercially.

The licensor cannot revoke these freedoms as long as you follow the license terms.

Under the following terms:

- **Attribution** — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
- **ShareAlike** — If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.
- **No additional restrictions** — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

Why using a firewall?

- The network where your system is hosted may be considered as “hostile”. The public Internet is definitely hostile.
- Some applications may not provide an easy way to restrict network accesses.
- A local firewall provides an additional layer of protection.
- You may need to use firewall rules for some specific functions (redirection, logging, network address translation, ...).

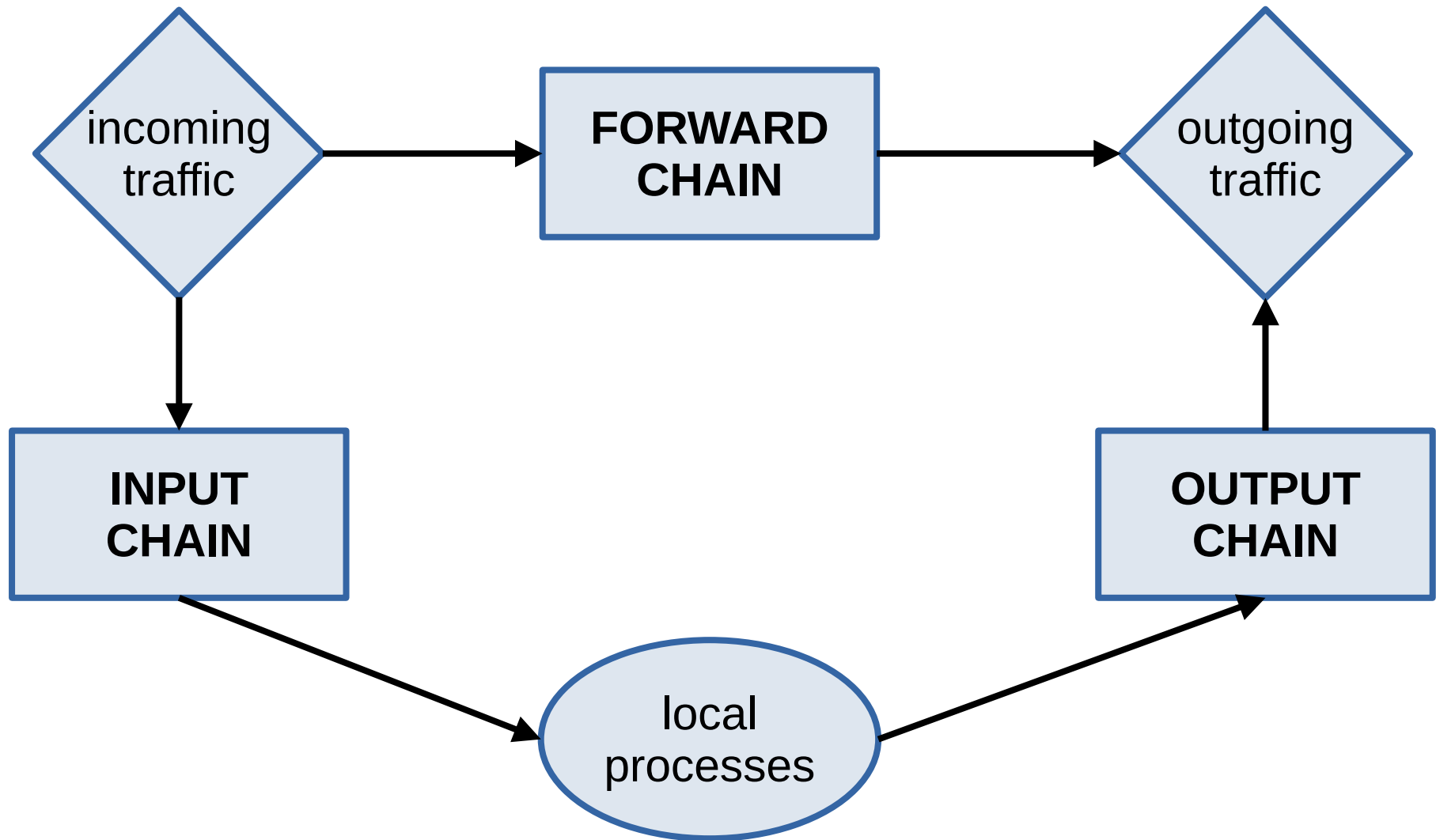
The Netfilter project

- Network filtering and related applications for the Linux system are developed under the umbrella of the Netfilter project.
- <https://netfilter.org/>

iptables and nftables

- iptables was the main network filtering tool available on Linux until the release of version 3.13 of the kernel, in 2014.
- Since then, nftables is used instead of iptables.
- A translation mechanism exists to keep iptables rules working with nftables, but long term management requires to work with the new tools.

Basic filter flow



iptables

- It relies on various network tables; each table contains chains, with a default policy (accept or drop).
- Filtering rules can be added or removed in each chain.
- 'Filter' is the default table, with three chains: input, forward and output.

nftables

Some of the main differences with iptables are:

- nftables doesn't provide pre-build tables
- the syntax is different; you can use the iptables-translate command to convert iptables scripts
- a rule can perform multiple actions at once (blocking and logging for instance)
- the same rules can be used for both IPv4 and IPv6
- performances have been improved

The nft command

- **Manipulating tables:**
nft { add | delete | list | flush } table { ipv4 | ipv6 | inet } table *table_name*
- **Manipulating chains:**
nft { add | create | delete | rename | list | flush } chain *table_name chain_name* <options>
- **Manipulating rules:**
nft { add | insert | replace | delete } rule <options>
- **Reviewing rules:**
nft list ruleset
- **Deleting all rules:**
nft flush ruleset

Firewall script

- Firewall rules can be set or modified manually, but it is strongly recommended to apply them with a script, during the boot process.
- Depending on your distribution, some scripts or tools may already be provided (*firewalld* on Red Hat Enterprise Linux, *ufw* on Ubuntu).
- Test your script when you are not relying on network connectivity!

Creating your first rule

- A rule usually match one or more of the following criteria:

Source IP Address

Source Port

Destination IP Address

Destination Port

- You can use the commands below to track the services running on the system and create the appropriate rules:
 - `sudo ss -antup | grep LISTEN`
 - `sudo lsof -i -P | grep LISTEN`
- Other criteria are available: network interface, connection status, ...

Modifying firewall rules

- Confirm that you have a script with a set of working rules (or to delete all existing rules).
- Schedule a job (with “at” or “cron”) to apply that rule set in 15 minutes.
- Make your firewall rule change.
- If you’ve lost access, wait up to 15 minutes to reconnect to the system.
- If the new rules are working properly, delete the scheduled job before it gets executed.