# Practical experiments about IP networking and routing

Franco CALLEGATI – Walter CERRONI

AA 2019-2020

Experiments with networking

- Practical experiments in networking with physical infrastructure are complex to organize and expensive

- In the last decade:
  - Virtualization technologies in ICT gives new opportunities

- Question?
  - Can we run a whole network «inside» a single physical machine, experimenting all protocols and features «as if» in a real environment

Single physical machine
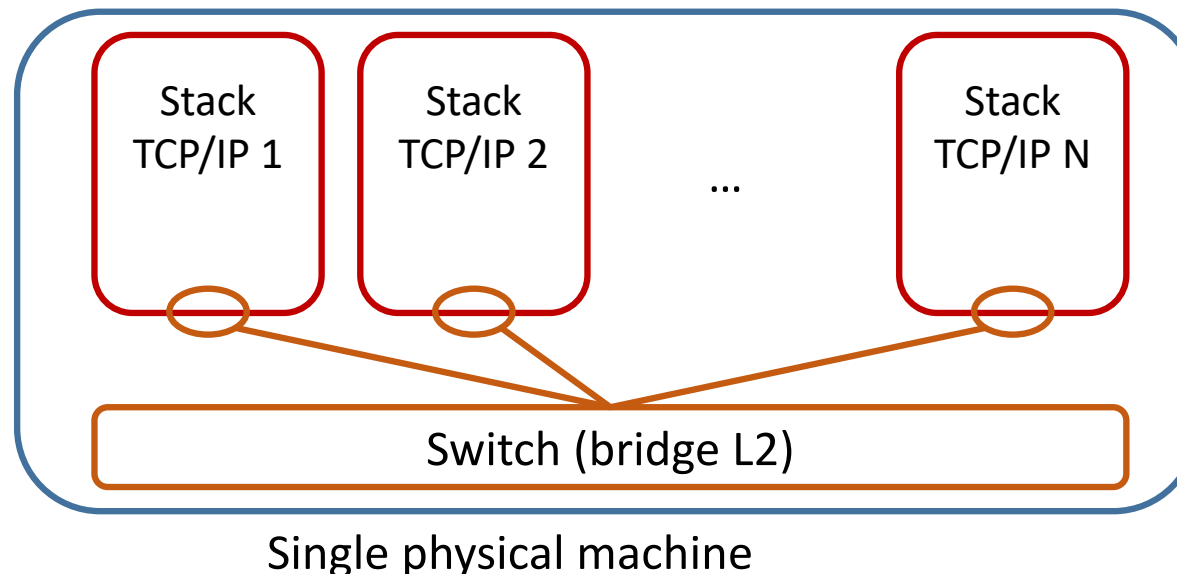
Example

- Networking with Linux namespaces
  - Run different network protocol stacks on the same machine
    - Different IP addresses
    - Different routing tables
    - Different forwarding strategies

  - Need a bridge to interconnect them

How to build LANs in the virtual space

- **Networking with Linux namespaces**
  - Run multiple and separate network protocol stacks on the same machine
    - Different IP addresses
    - Different routing tables
    - Different forwarding strategies

  - The IP network is built with a switch interconnecting protocol stacks by means of their interfaces
  - Need for a software switch or bridge

The key commands (1)

- `ip nets`
  - Network namespace management
    - `ip netns add NAMESPACE`
      - Create a namespace
    - `ip netns del NAMESPACE`
      - Delete a namespace
    - `ip netns list`
      - List the existing namespaces
    - `ip netns exec NAME COMMAND`
      - Execute a specific COMMAND in namespace NAME
    - `ip netns id`
      - Gives the ID of the current namespace

The key commands (2)

- `ip link`
  - Manage and configure network devices
    - `ip link add NAME type TYPE`
      - NAME is the symbolinc name of the device
      - TYPE is the type of network device
        - Veth = virtual ethernet interface
        - Bridge = Ethernet bridge (switch)

    - `ip link add NAME1 type veth peer name NAME2`
      - Create a ethernet link between two logical interfaces named NAME1 and NAME2s

    - `ip link set NAME`
      - Configure properties of NAME

The key commands (3)

- `bridge`
  - Manage bridge addressing
    - `bridge link`
      - Set properties of interfaces to links
    - `bridge fdb`
      - Manage bridge forwarding database