Università degli Studi Roma Tre
Dipartimento di Ingegneria
Computer Networks Research Group

Kathará

A container-based framework for experimenting computer networking

Version	2.0
Author(s)	L. Ariemma, G. Bonofiglio, G. Di Battista, V. Iovinella, G. Lospoto, M. Patrignani, M. Pizzonia, M. Rimondini
E-mail	contact@kathara.org
Web	http://www.kathara.org/
Description	An introduction to the architecture, setup and usage of kathará – based on a similar presentation of Netkit

Copyright notice

About computer networks

- computer networks are quite complex
  - several devices (computers, routers, etc.)
  - several interfaces
  - several protocols running
  - physical interconnections originate complex topologies

How to perform experiments?

- performing experiments may be unfeasible
- the currently used network cannot be exploited for experiments
  - it hosts services that are critical for the company
  - it would be necessary to coordinate different departments of the company
- network equipment is expensive
  - sometimes, even for performing simple experiments, several equipment should be available in the same test bed

Simulation vs. emulation

- emulation and simulation systems put at user's disposal a virtual environment that can be exploited for tests, experiments, measures
- simulation systems aim at reproducing the performance of a real-life system (latency time, packet loss, etc.)
  - e.g.: ns, real, ...
- emulation systems aim at accurately reproducing the functionalities of a real-life system (configurations, architectures, protocols), with limited attention to performance

Kathará

a system for emulating computer networks

Emulating a network

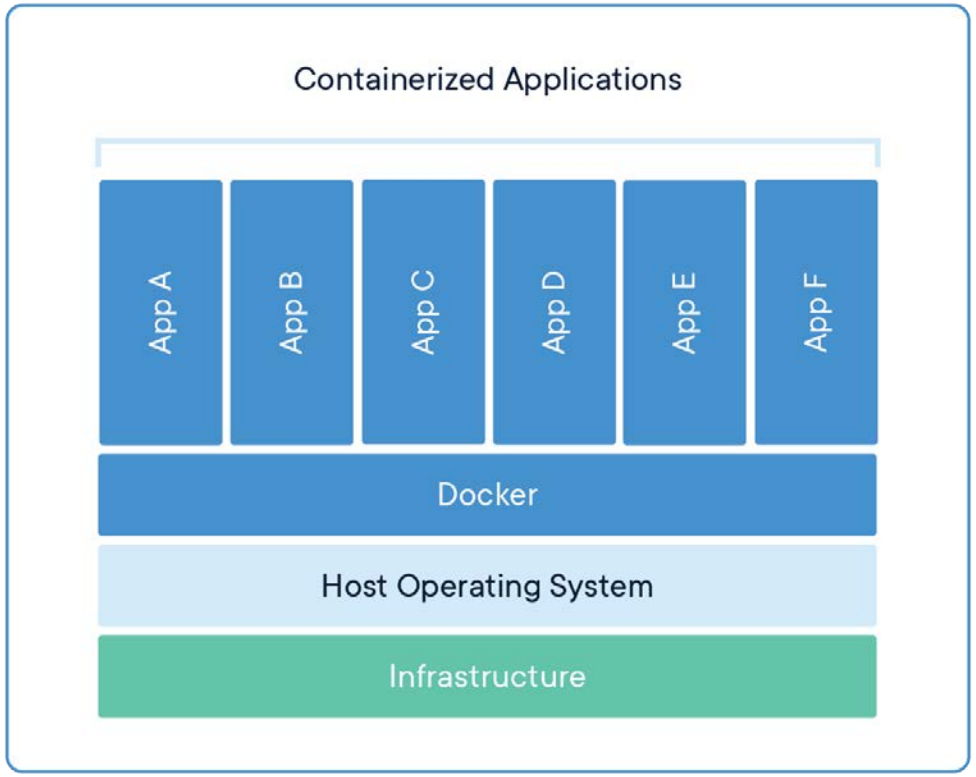
- basic idea:
  - several containers are created inside a single host machine
  - containers are connected to virtual collision domains and thus can communicate with each other
- each container can be configured as a device that plays the role of a regular host, of a router, of a switch, ...

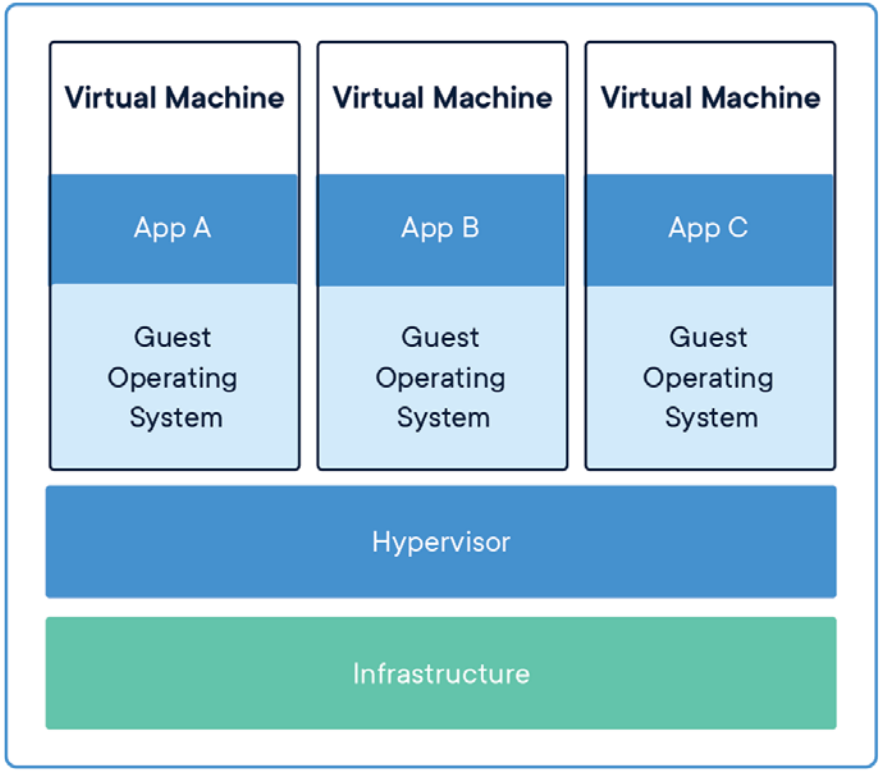
Kathará

- based on Docker
- each emulated network device (in what follows *device*) is a container
- note: several container images available, e.g.:
  - Base (DNS, Web Server, network utilities)
  - Quagga (standard routing protocols)
  - FRRouting (standard routing protocols + EVPN + MPLS)
  - Open vSwitch (Open Flow enabled switch)
  - Behavioral Model (software implementation of a P4 switch)

Docker and containers

A container is a standard unit of software that packages up code and all its dependencies, so the application runs quickly and reliably from one computing environment to another.

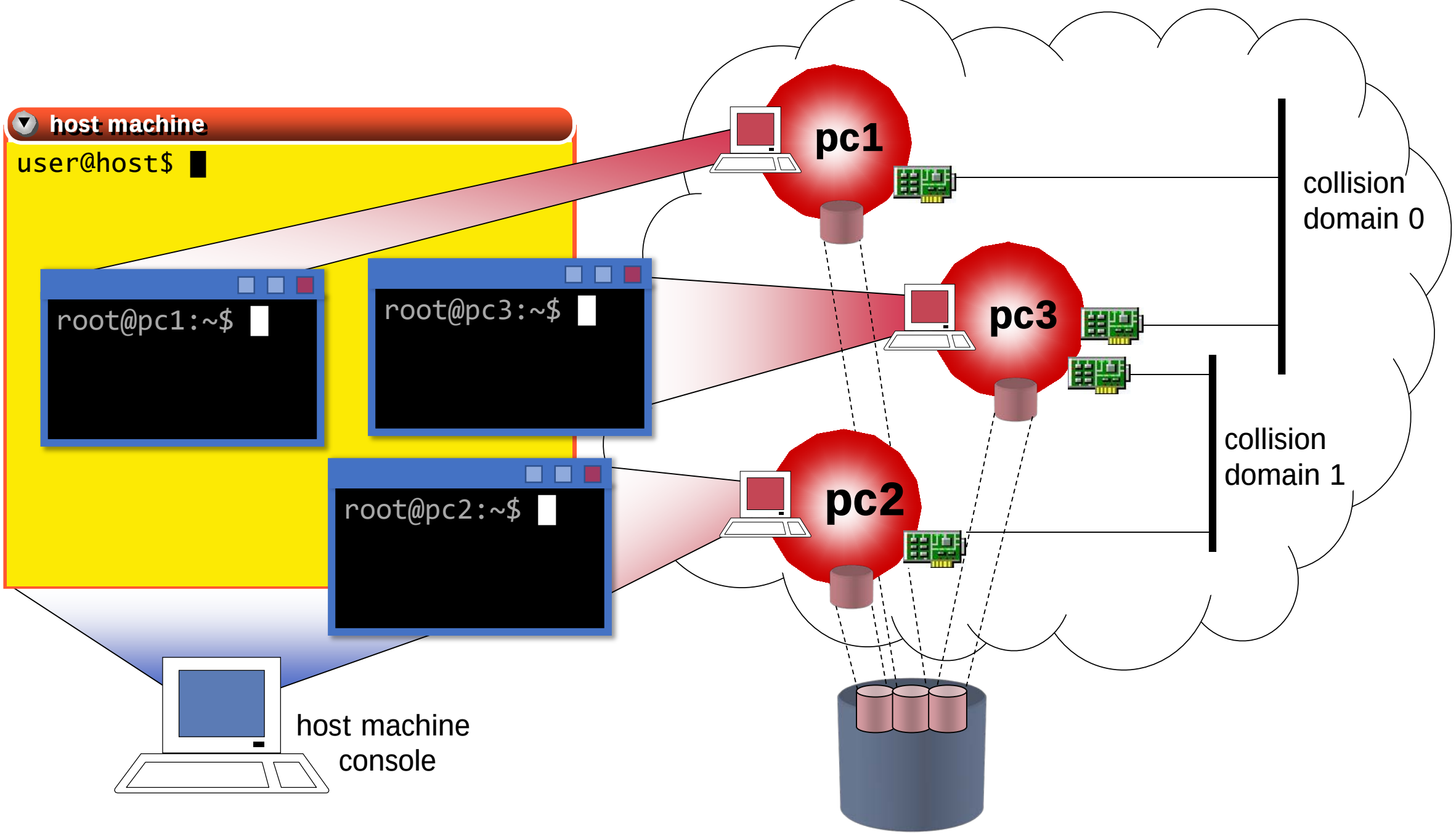




https://www.docker.com/resources/what-container

Emulated network devices

- each device has:
  - a console (a terminal window)
  - a memory
  - a filesystem
  - (zero, one or more) network interfaces
- each network interface can be connected to a (virtual) collision domain
- each virtual collision domain can be connected to several interfaces



Setting up Kathará

Setup manual

Setting up Kathará

- available for:
  - Windows 10 Pro
  - Linux
  - MacOS
- download at http://www.kathara.org/
- follow the wiki

Using Kathará

Kathará commands

- kathará provides users with three sets of commands
  - v-prefixed commands (v-commands)
  - l-prefixed commands (l-commands)
  - global commands
- v-commands act as low-level tools for configuring and starting up a single device
- l-commands provide an easier-to-use environment to set up complex labs consisting of several devices
- global commands are mainly management commands

Kathará v-commands

- allow to startup a single device with arbitrary configurations (network interfaces, etc.)
  - vstart: starts a new device
  - vconfig: attaches network interfaces to a running device
  - vclean: halts a device

Kathará l-commands

- ease setting up complex labs consisting of several virtual machines
  - lstart: starts a Kathará lab
  - lclean: halts all the devices of a lab
  - lrestart: halts all the devices of a lab and start them again
  - linfo: provides information about a lab

Kathará global commands

- management commands
  - check: Check your system environment
  - connect: Connect to a running Kathará machine
  - list: Show all running Kathará machines of the current user
  - settings: Show and edit Kathará settings
  - wipe: Delete all Kathará machines and links, optionally also delete settings

Share files between the host and the devices

- there are two ways to share files between the host filesystem and the device filesystem:
  - the **`/shared`** directory inside a device directly points to the **`shared`** directory inside the lab
    - by default it is **ENABLED**, you can disable it in the settings
    - read/write access is allowed
  - the **`/hosthome`** directory inside a device directly points to the **home directory** of the current user of the host
    - by default it is **DISABLED**, you can enable it in the settings
    - read/write access is allowed

Testing Kathará

Testing Kathará

- To test if your setup works correctly run:
  - `kathara check`
    - This command will ran automatic tests to your environment
  - `kathara vstart -n pc1 --eth 0:A`
    - This command will start a new device called **`pc1`** and connected to the virtual collision domain **`A`**
    - A terminal window will open allowing to run commands inside the device
  - `kathara vclean -n pc1`
    - This command will stop the previous started device

Preparing a Kathará lab

Kathará lab

- a Kathará lab is a set of preconfigured devices that can be started and halted together
- a basic Kathará lab is a directory tree containing:
  - a lab.conf file describing the network topology
  - a set of subdirectories that contain the configuration settings for each device
  - <device_name>.startup files that describe actions performed by devices when they are started

lab.conf

- describes:
  - the settings of the devices that make up the lab
  - the topology of the network that interconnects the lab's devices
- contain a list of **`machine[arg]=value`** lines where:
  - **`machine`** is the name of the device (e.g. pc1)
  - if **`arg`** is a number, then **`value`** is the name of a collision domain to which eth**`arg`** should be attached
  - if **`arg`** is not a number, then it must be an option and **`value`** the argument

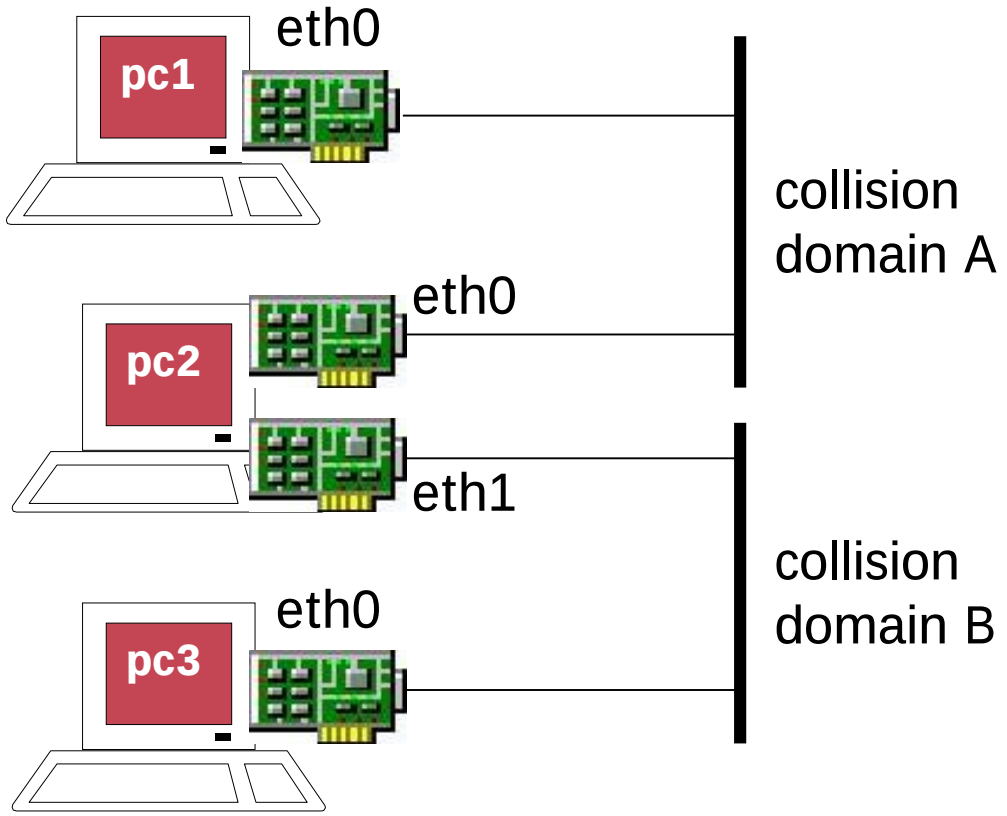
lab.conf

- example

```
pc1[0]=A

pc2[0]=A
pc2[1]=B

pc3[0]=B
```



lab subdirectories

- Kathará starts a device for every device specified in lab.conf file and every subdirectory of the lab folder
- the contents of subdirectory **`device`** are copied into the root (**`/`**) of the device named **`device`** filesystem
  - for example, **`pc1/foo/file.txt`** is copied to **`/foo/file.txt`** inside the device **`pc1`**

startup files

- shell scripts that are executed inside a device right after its startup
- typical usage of a **`.startup`** file is to configure network interfaces and/or start network services
  - for example:

```
ifconfig eth0 10.0.0.1/24 up
/etc/init.d/zebra start
```

launching/halting a lab

- open a terminal
- enter the lab directory (**`cd lab_directory`**)
- launch a Kathará l-command
  - where l-command could be one of the following
    - **`kathara lstart`**, to start the lab
    - **`kathara lclean`**, to stop the lab
    - **`kathara lrestart`**, to restart the lab

more information

- further information can be found:
  - on GitHub's Kathará wiki
  - on the official website http://www.kathara.org
  - inside Kathará man pages
    - you can start from **`man kathara`**
    - available only on Linux and MacOS