# FreeBSD Jail
# Notes jotted on the prison vvall.

Bjorn A. Zeeb (bz@FreeBSD.org)

Zabbadoz.CoM

EuroBSDCon 201010

# Overview

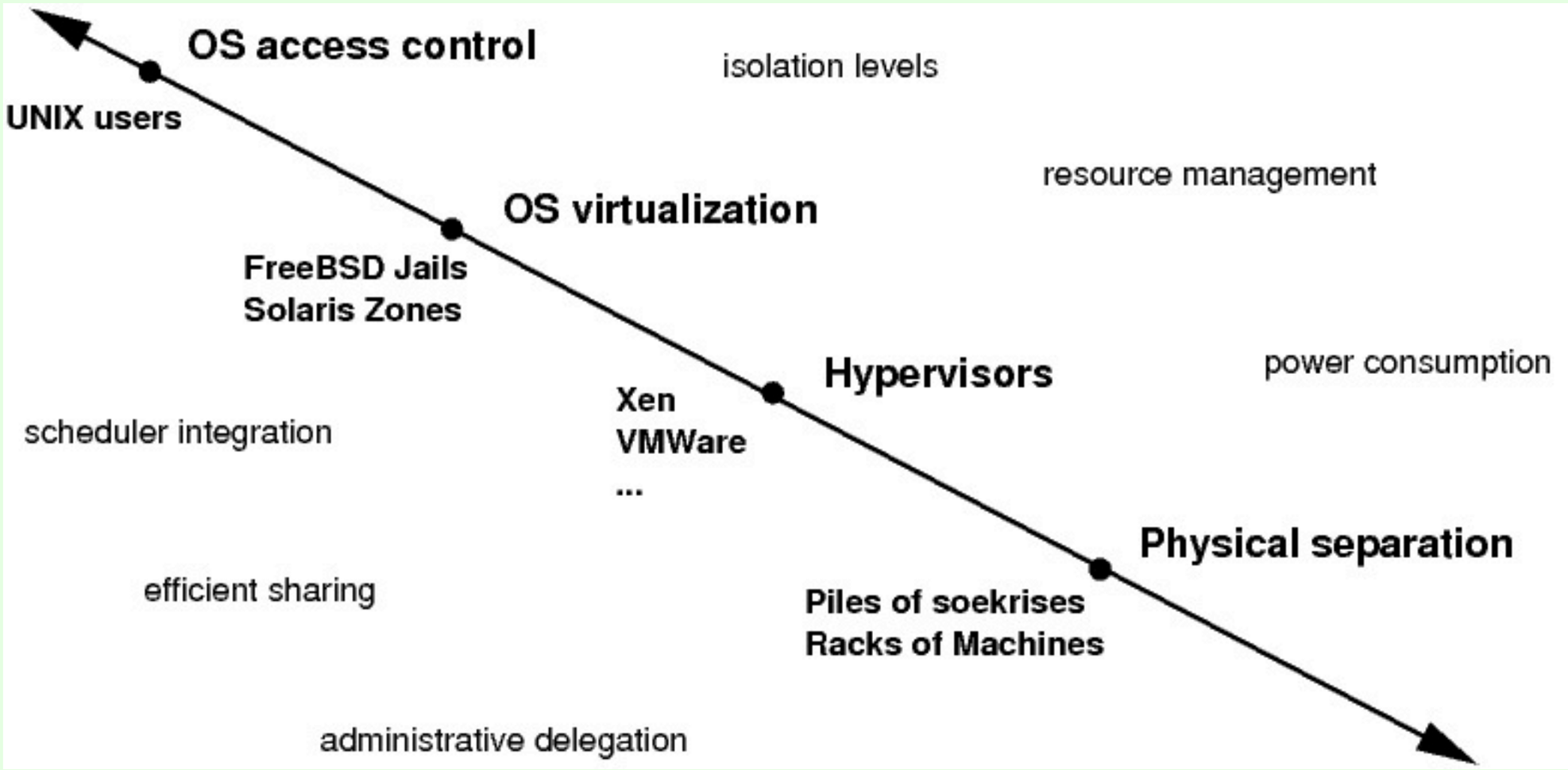
- The past.
- Why Jails on FreeBSD? When not?
- The present.
- Interesting on-going things.
- What people are doing with this.
- The future.
- What about you?

# The past

- 1999 Jails introduced.
- 2002 M. Zec 'network stack virtualization'.
- Since:
  - ZFS support.
  - Multi IPv4/v6/no-IP jails.
  - Cpuset support.
  - Flexible jail command (new syscalls).
  - Persistence and hierarchy support.
  - ...

# Why Jails on FreeBSD?

## When not?



# Why Jails on FreeBSD?

## When not?

- Not for "unnamed commercial OS".
- Not if we cannot run it.
- Not if you have too many machines anyway ..
  .. well .. stay here and listen ..
- Not if you want live migration.

# Why Jails on FreeBSD?

## When not?

- Lightweight and fast.
- Secure.
- Simple.
- ...

And we want to keep it that way.

# Why Jails on FreeBSD?

## When not?

- Does not depend on special hardware support.
- Works across all architectures.
- 3rd party features included like:
  - ZFS and DTrace (user space support coming).
- Linux support.
- Ports collection.
- freebsd-update support.

Not convinced yet? There's more ...

# Why Jails on FreeBSD?

## When not?

- Lightweight
  - Lots of jails on one box.
  - We could give you six 9s.
  - As low as 2MB + user data per virtual instance using ZFS or nullfs based techniques.
  - Classic jail w/o processes uses about 5k of memory on amd64.

# Why Jails on FreeBSD?

## When not?

- Secure
  - Save super user delegation with restrictions.
  - Less bad security press than most hypervisors.
  - *There is no escape -- as the T-shirts say.*

# Why Jails on FreeBSD?

## When not?

- Simple
  - jail(8) to start, modify and stop.
  - jls(8) to list.
  - jexec to attach to a jail.
  - One could still do it by hand, but ...

# The present

- All formerly mentioned things still work (mostly).
- FreeBSD 7.2 and later ship with multi-IP jails.
- FreeBSD 8.0 and later ships with
  - new, flexible jail command (new syscalls),
  - hierarchy support and
  - experimental virtual network stack support.

# Interesting on-going things.

- Generalized VIMAGE virtualization framework.
- Virtualized IPC (2 patches),
  - e.g. better PostgreSQL support with jails.
- Virtual network stack support (vnet).
- New configuration (Jamie's talk after this one).
- Hierarchical Resource Limits.

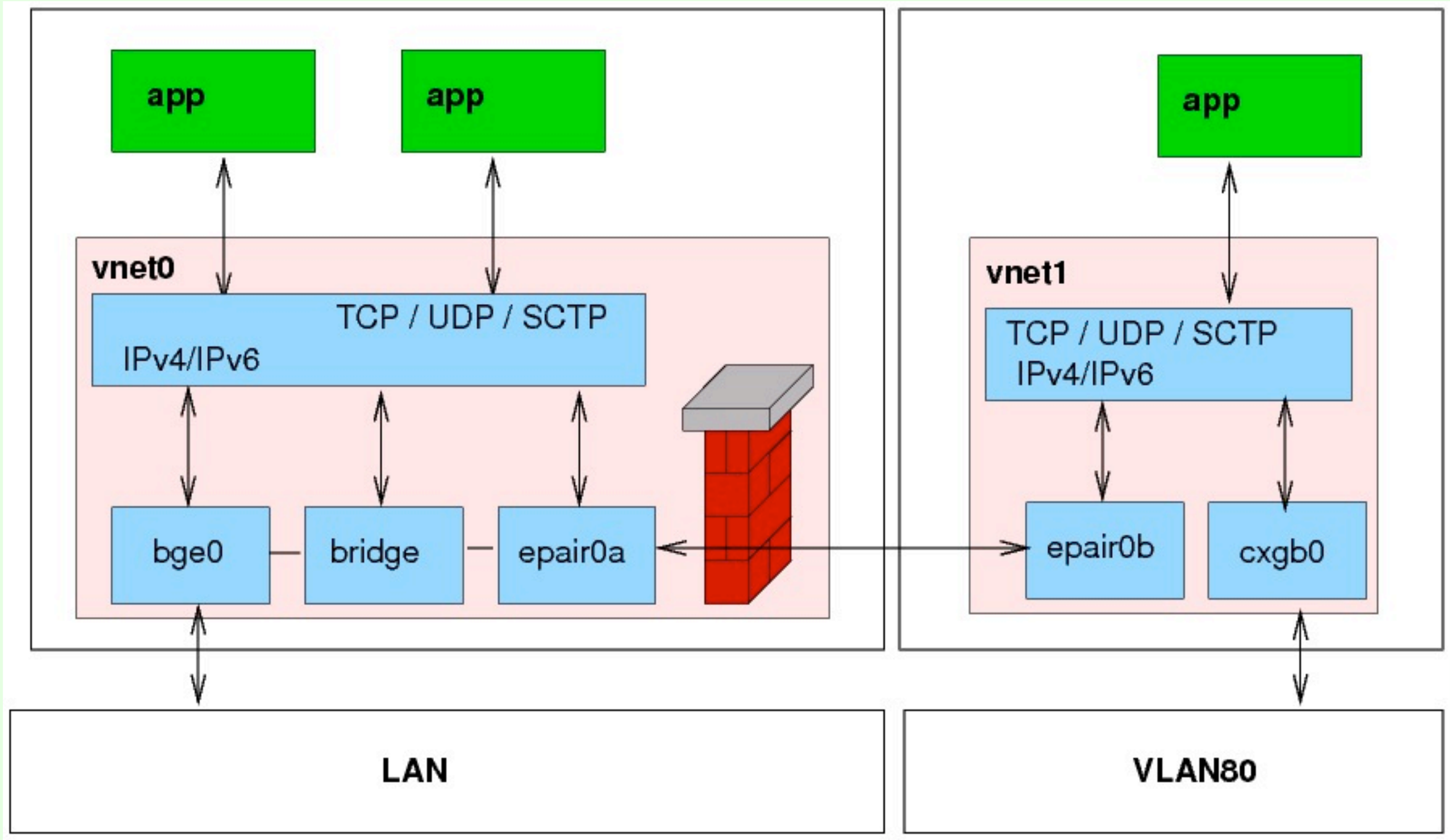
# Hierarchical Resource Limits

- Started as Summer of Code.
- FreeBSD Foundation sponsored project.
- Limits on:
  - CPU,
  - memory,
  - number of processes and threads,
  - number of file descriptors,
  - SYSV
- All applicable to jail as well.

# Virtual Network Stacks

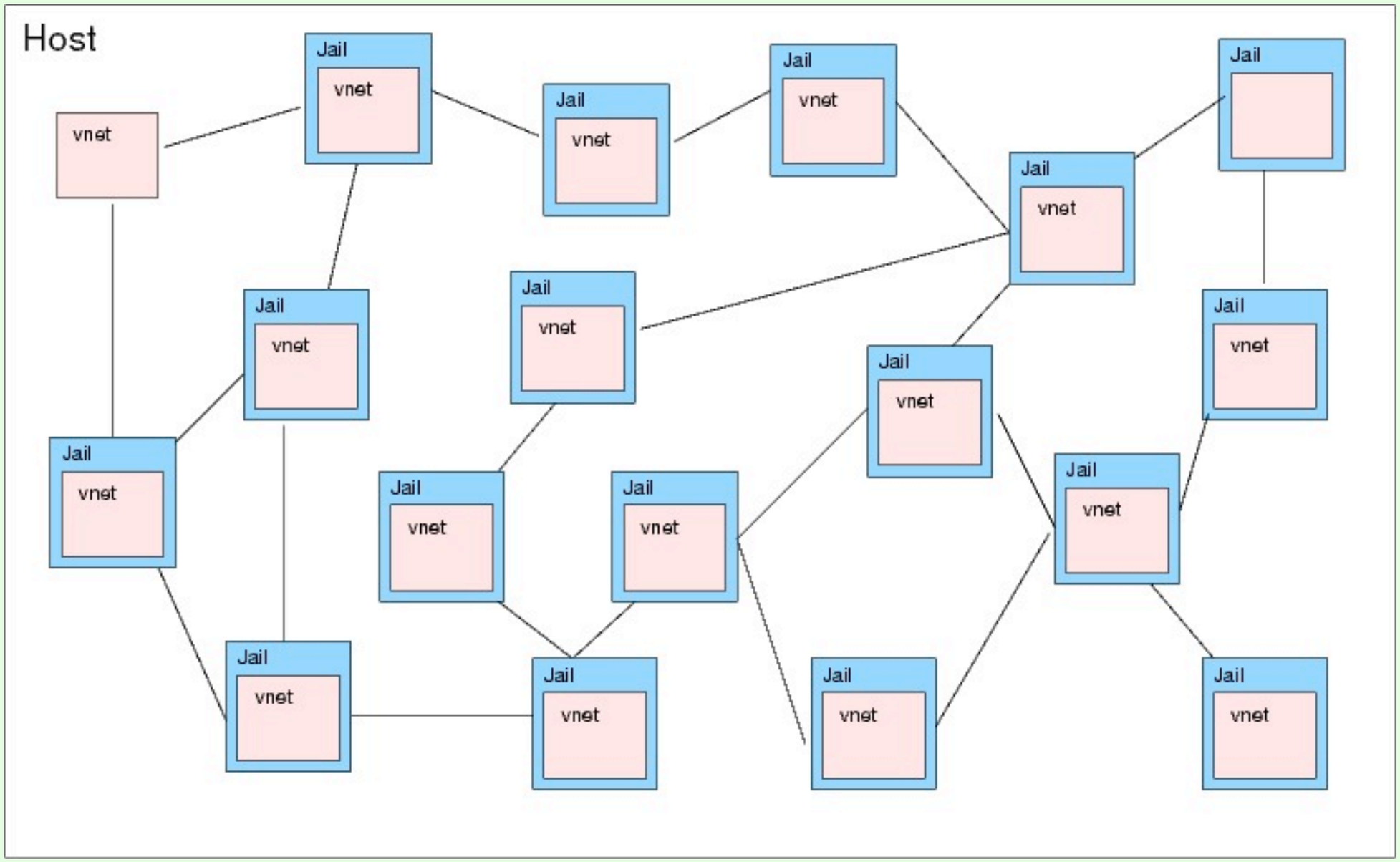
- What is this?
  - Experimental Feature.
  - Jails with their own network stack.
  - TCP/IP socket binding.
  - Own routing table.
  - Own IPsec, Firewalls.
  - Arbitrary topologies are OK.

# Virtual Network Stacks




Sample jail setup.

# Virtual Network Stacks




Arbitrary topology sample

# Virtual Network Stacks

- What's the problem it's taking so long?
- What's cooking?
  - pf support?
  - Cloned interfaces like carp, vlan, ...
  - USB Ethernet and Cardbus.
  - Top-Down teardown and with that general kernel enhancements.

# What are people doing?

## Development

- Network protocol development.
  - link layer,
  - UDP, TCP, SCTP, ..
  - IPsec,
  - Application protocol level.
- IPv6 only networks.
- Bug hunting.
- Regression testing.

# What are people doing?

## Simulations - Integrated Multiprotocol Emulator/Simulator

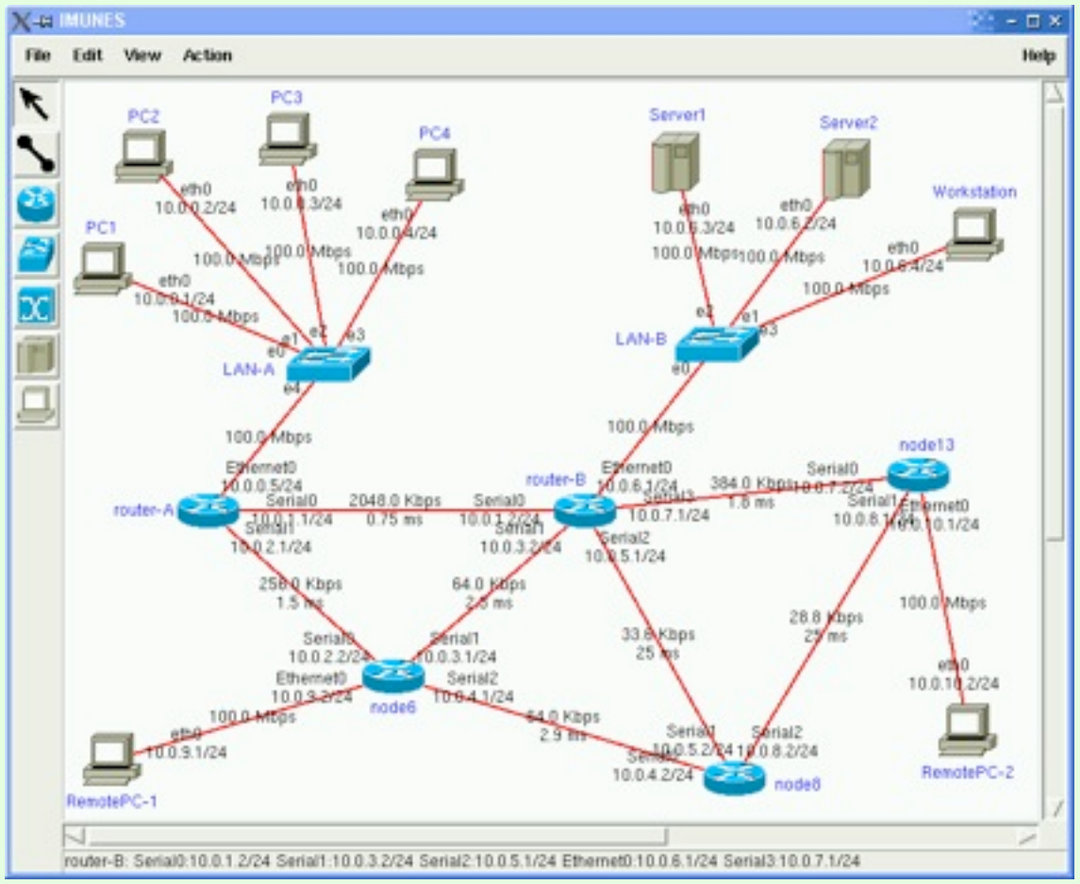



Image from old.tel.fer.hr/imunes/GUI-normal.gif

- New version of Imunes to come.
  - More easy integration of private "nodes".
  - Documentation.
- http://imunes.tel.fer.hr/

# What are people doing?

## Simulations - Common Open Research Emulator (CORE)

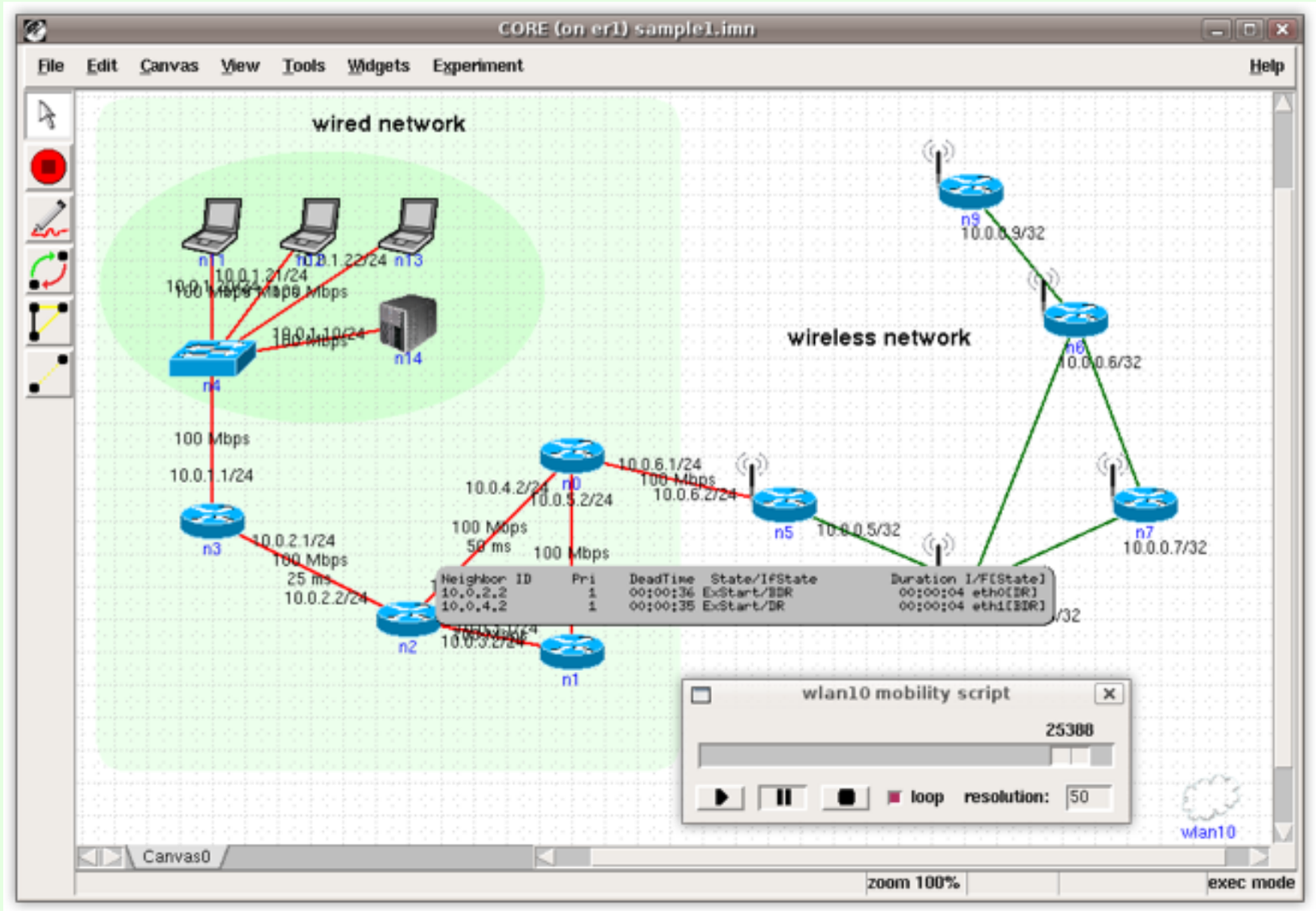



Image from cs.itd.navy.mil/work/core

- Also has MANET/WLAN support.
- http://cs.itd.nrl.navy.mil/work/core/index.php

# What are people doing?

## Server consolidation, SMB setups.

- Move n old installations into a jail on a new box.
- Internal / external jails.
  - Webserver / DB.
  - Mail relay / mail filtering, virus scanner / IMAP.
- Use carp and storage/mirroring for redundancy.

# What are people doing?

## ISPs / Appliance builders

- Lots of VLANs to the box, per customer FW.
- L2TP concentrator with per customer fan-out and RADIUS.
- IPsec gateways.
- All with add-on services (application level gateways).
- Different "zones" in one appliance.
- Virtualized overlay networks.

# What are people doing?

## Hosting

- Lots and lots and more jails on one machine.
  - 100s and 1000s of classic jails.
  - Couple of thousand jails with vnet.
- Have Debian in a jail (Debian GNU kFreeBSD).
- Run Linux binaries inside jails.
- Systems supporting vnets already exist http://www.ispsystem.com/ .

# The future.

- Virtualized SYSV and Posix IPC.
- Docs.
- (jail)init.
- per-jail audit support.
- VPROC.
- Console (kernel messages and kind of getty).
- priv(9) management.

# Conclusions

- Virtual kernel subsystem like vnet become reality.
- Prototype increasingly stable.
- Performs and scales really well.
- Adds to the virtualization menu of FreeBSD combined with other techniques like Xen.
- Coming soon(sih).

# What about you?

- What do you want to see happen?
- Can you contribute to it?
- freebsd-virtualization@freebsd.org.
- Questions?