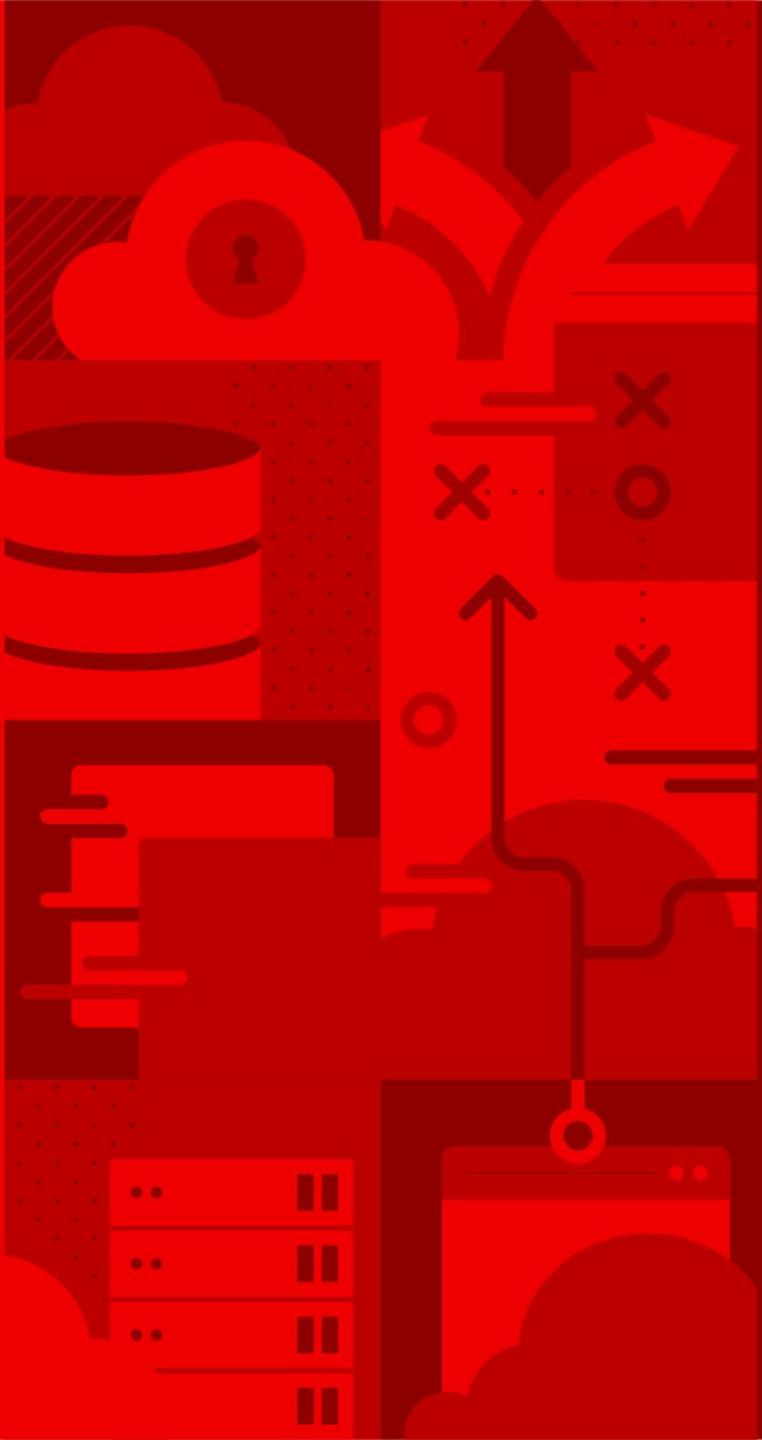
passt & pasta

Modern rootless networking for containers and VMs

David Gibson <dgibson@redhat.com>
Principal Software Engineer

Red Hat



Ancient History

It was the dawn of the third age of the internet, the year the browser war came upon us all.

This is the story of the first of the free L2/L4 bridges. The year is 1995 and the name of the software is SLiRP.

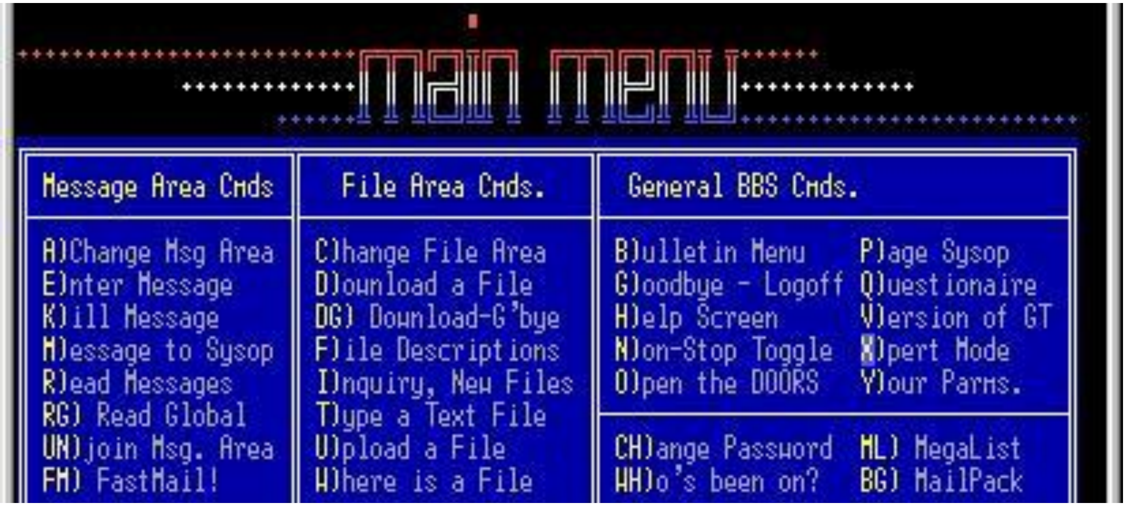
The Dial-up World

Of the mid-1990s

- Modems and home dial-up were established technology
- Internet access was common at universities
- Some large businesses and organizations also had internet
- Commercial ISPs were still uncommon and expensive
- However, many students and staff at universities had access to dial-up shell accounts

Dial-up without Internet

Bulletin Board Systems



```
main menu

Message Area Cmds     File Area Cmds.       General BBS Cmds.

A)Change Msg Area     C)hange File Area     B)ulletin Menu      P)age Sysop
E)nter Message        D)ownload a File      G)oodbye - Logoff   Q)uestionaire
K)ill Message         DG) Download-G'bye    H)elp Screen        V)ersion of GT
M)essage to Sysop     F)ile Descriptions    N)on-Stop Toggle    X)pert Mode
R)ead Messages        I)nquiry, New Files   O)pen the DOORS     Y)our Parms.
RG) Read Global       T)ype a Text File
UN)join Msg. Area     U)pload a File        CH)ange Password    ML) MegaList
FM) FastMail!         W)here is a File      WH)o's been on?     BG) MailPack
```

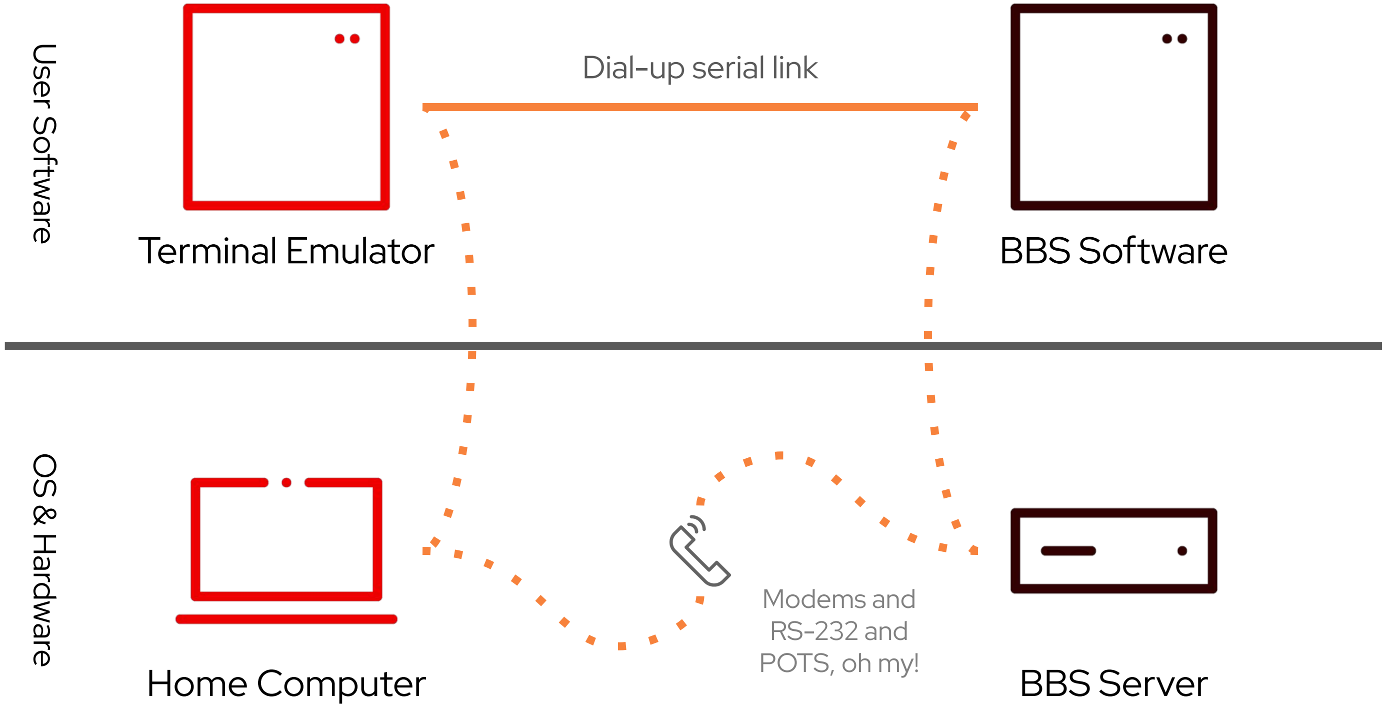
Dial-up text systems

- Text interface
  - Usually menu driven
- Exchange messages with other users
- Upload and download files (slowly)
- Play games
- Various other things

Source:
https://commons.wikimedia.org/wiki/File:Koala_Country_BBS_Main_Menu.jpg

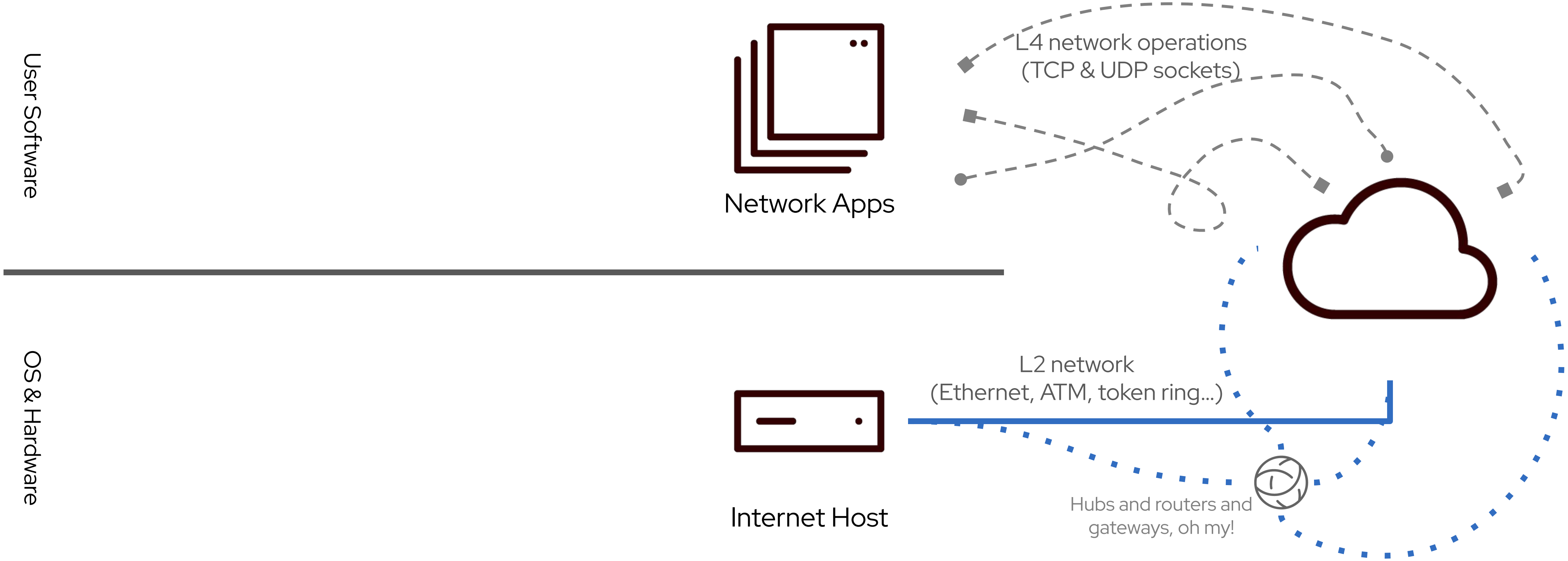
Dial-up without Internet

Bulletin Board Systems



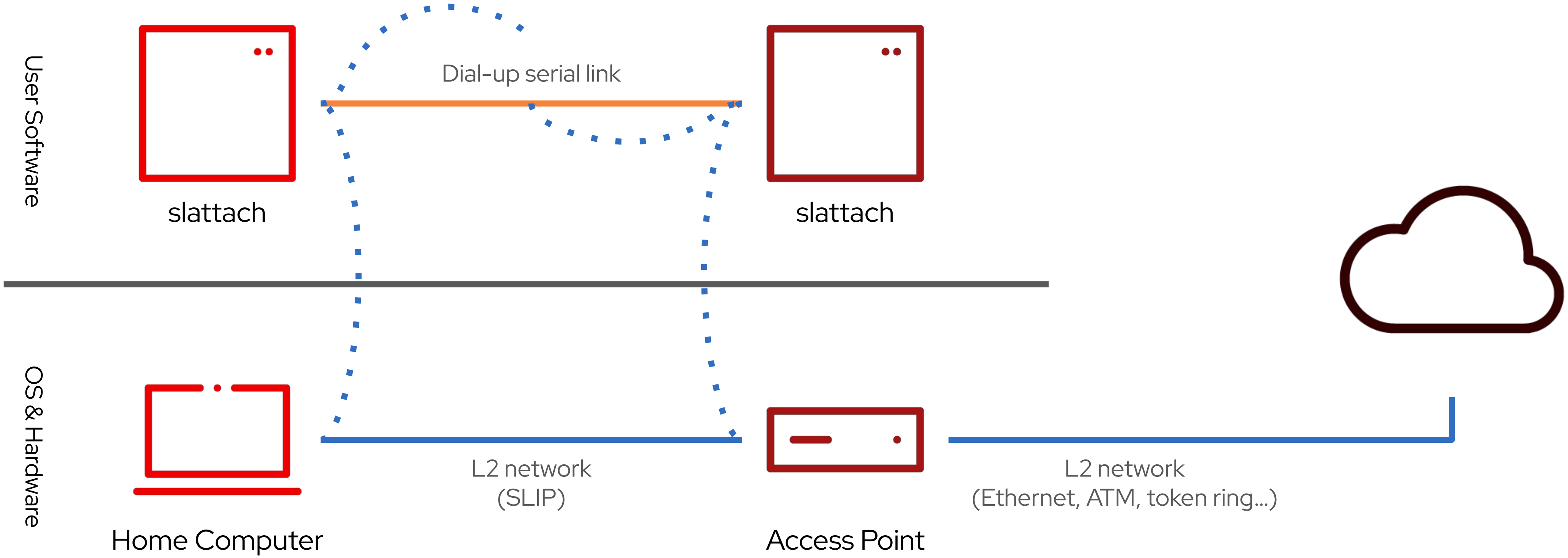
Internet without Dial-up

At your local university



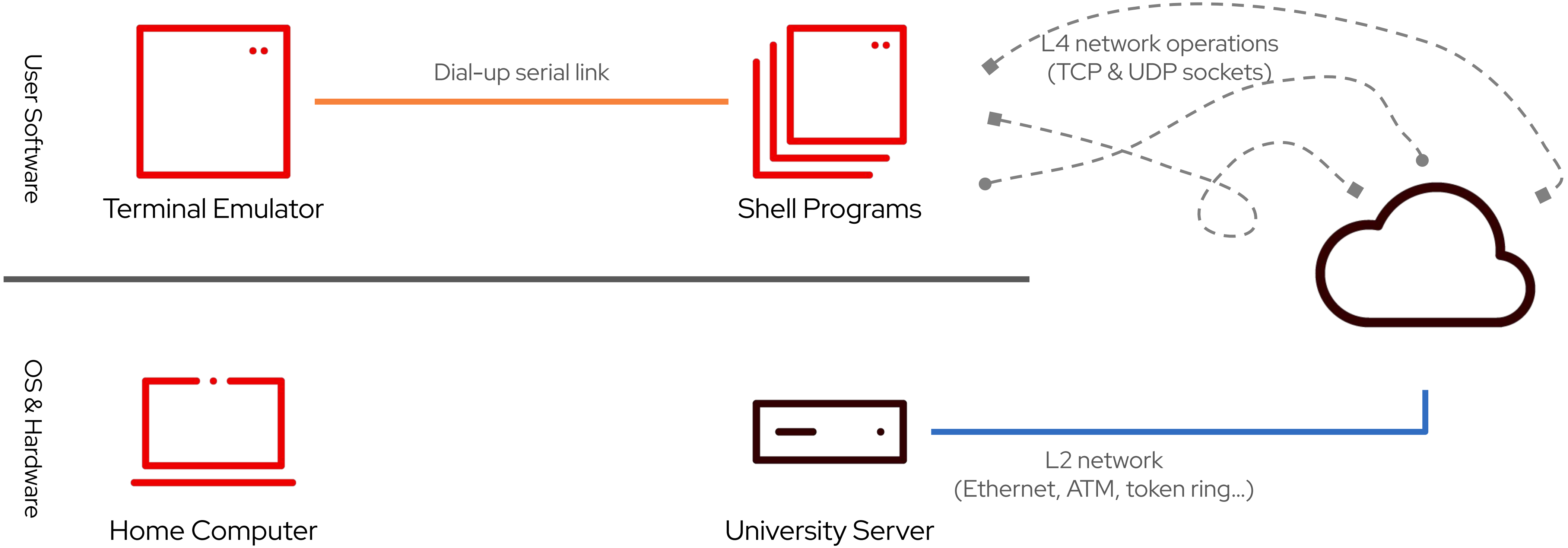
Dial-up Internet

For the lucky few who can get it



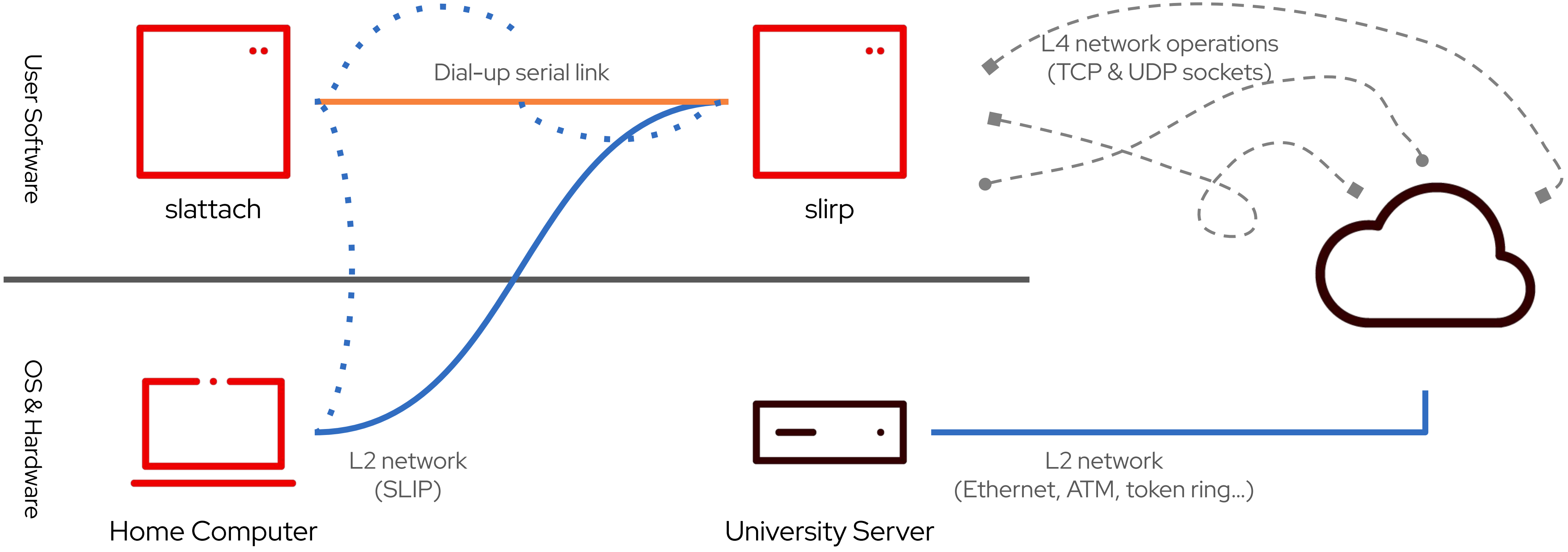
Dial-up Shell Accounts

root here, connectivity over there



Slirp

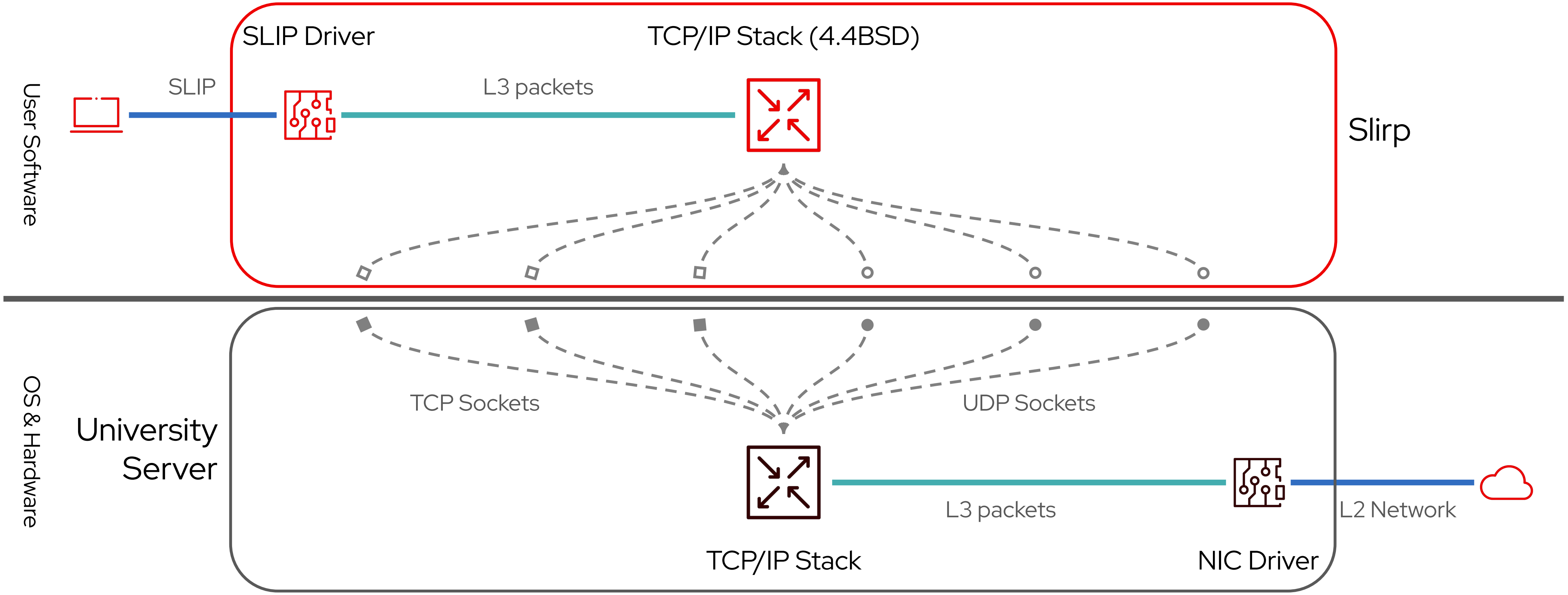
a.k.a. SLiRP, slirp, SLuRP or slurp



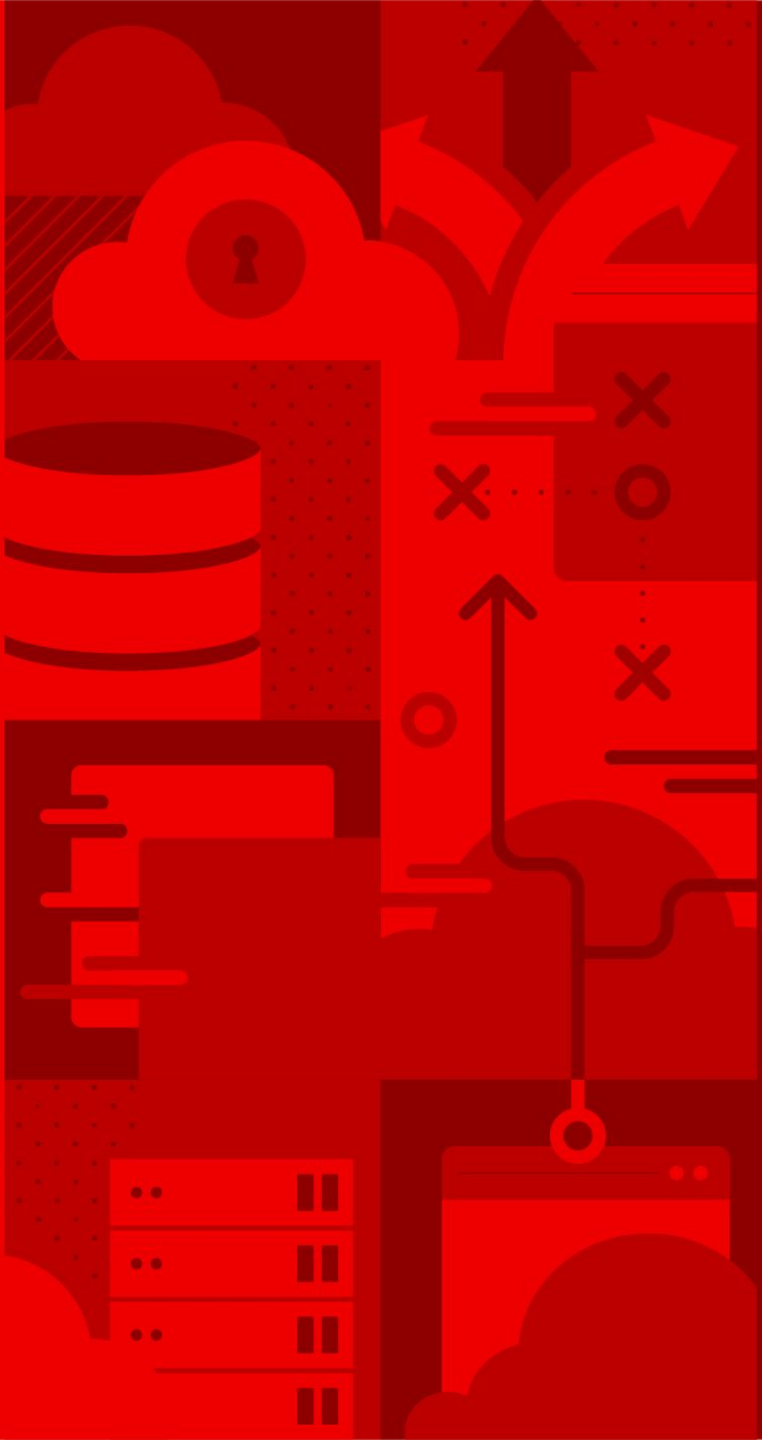
https://groups.google.com/g/comp.os.linux.announce/c/iAl4BcnnzJU/m/zpF2D-cxa-8J

Slirp

A network stack in userspace



Source:
https://groups.google.com/g/comp.os.linux.announce/c/iAI4BcnnzJU/m/zpF2D-cxa-8J

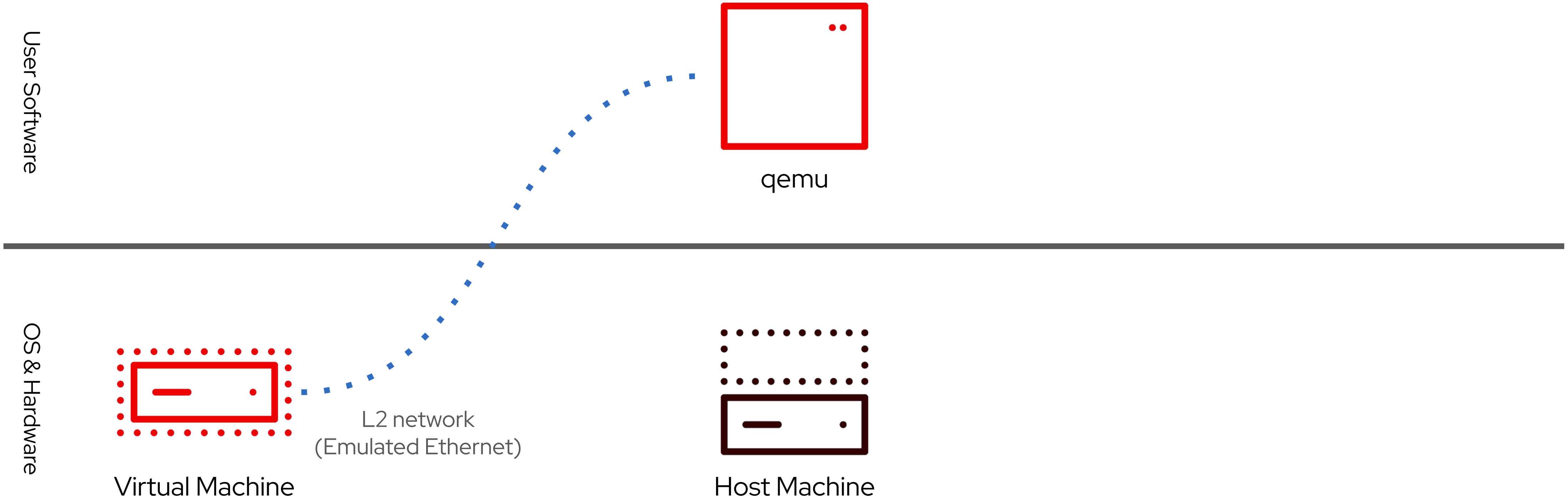


Modern History

ISPs became common and cheap.
PPP replaced SLIP.
Broadband replaced dial-up.
So much for Slirp... right?

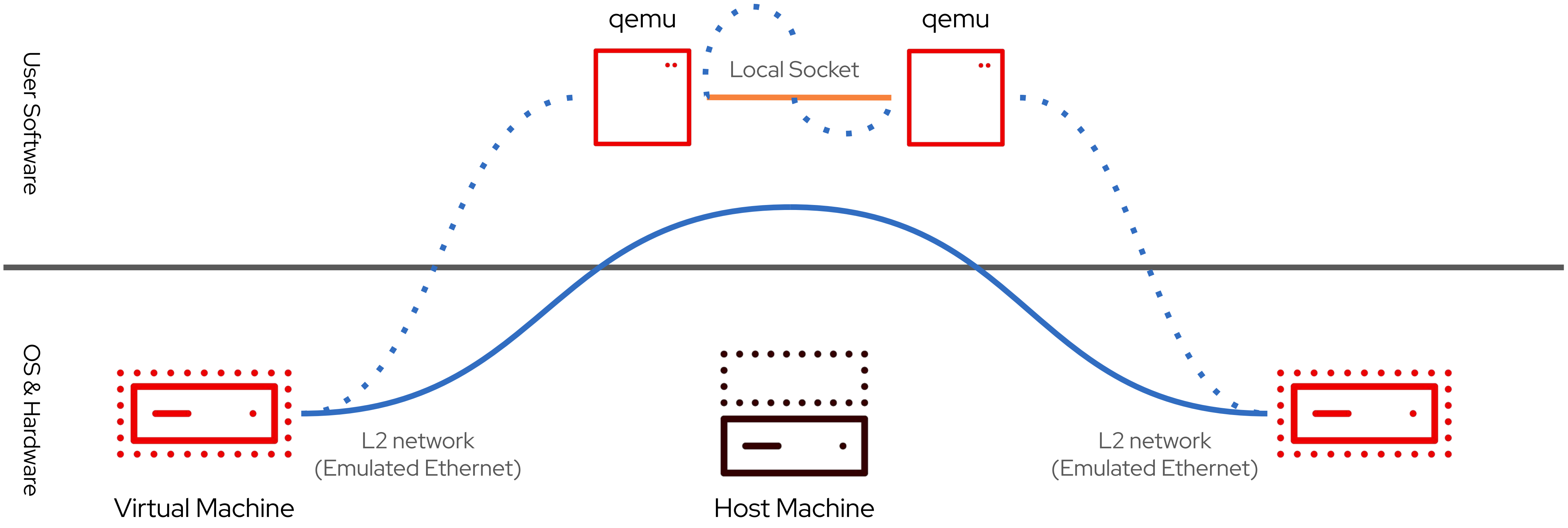
Virtual Machines

Emulated Network Interface



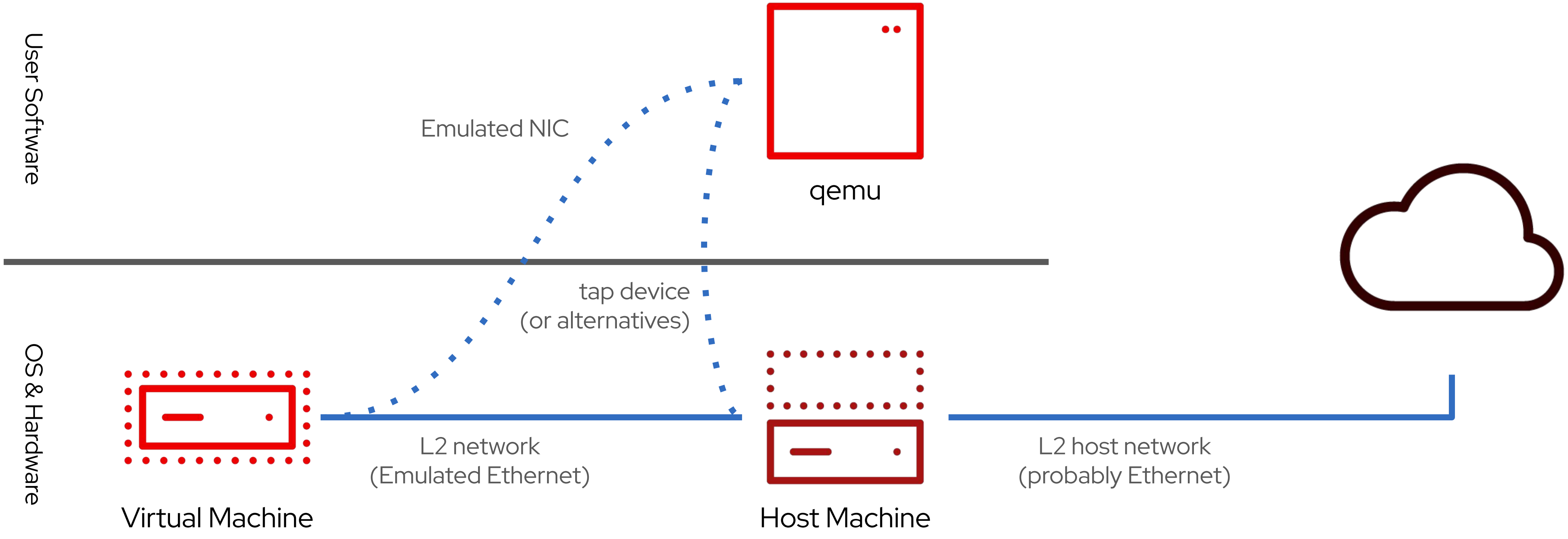
Virtual Machines

QEMU `-net socket` / `-net stream`



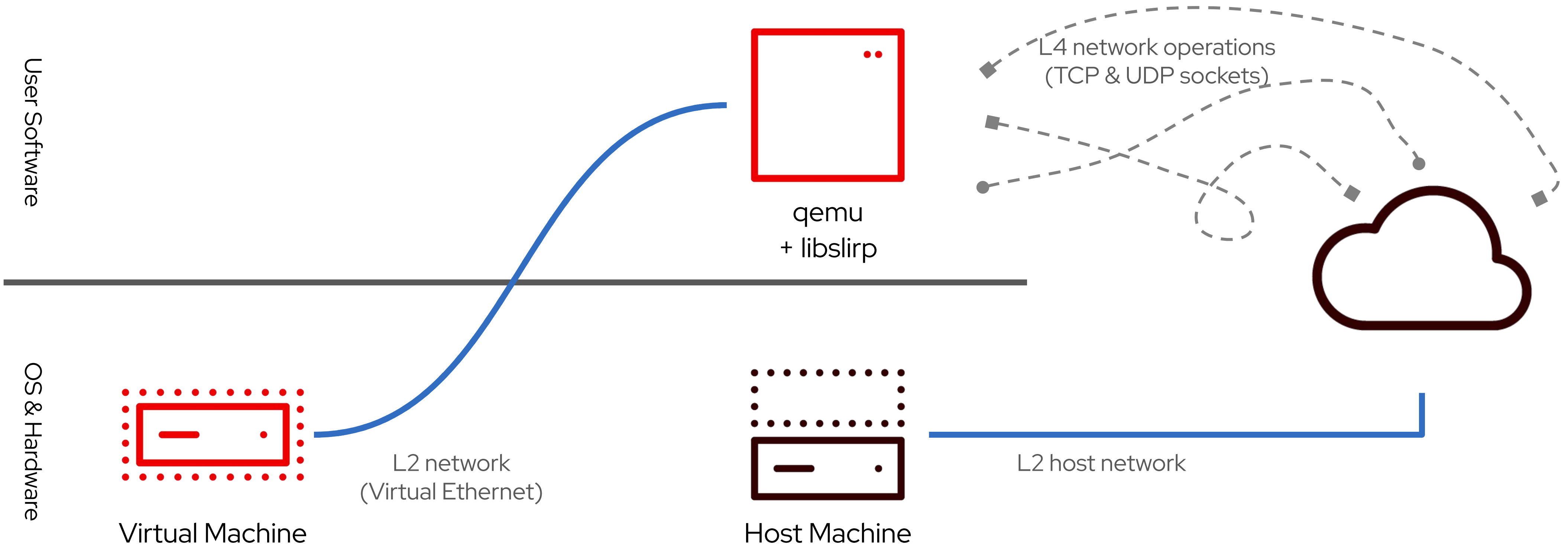
Virtual Machines

QEMU `-net tap`



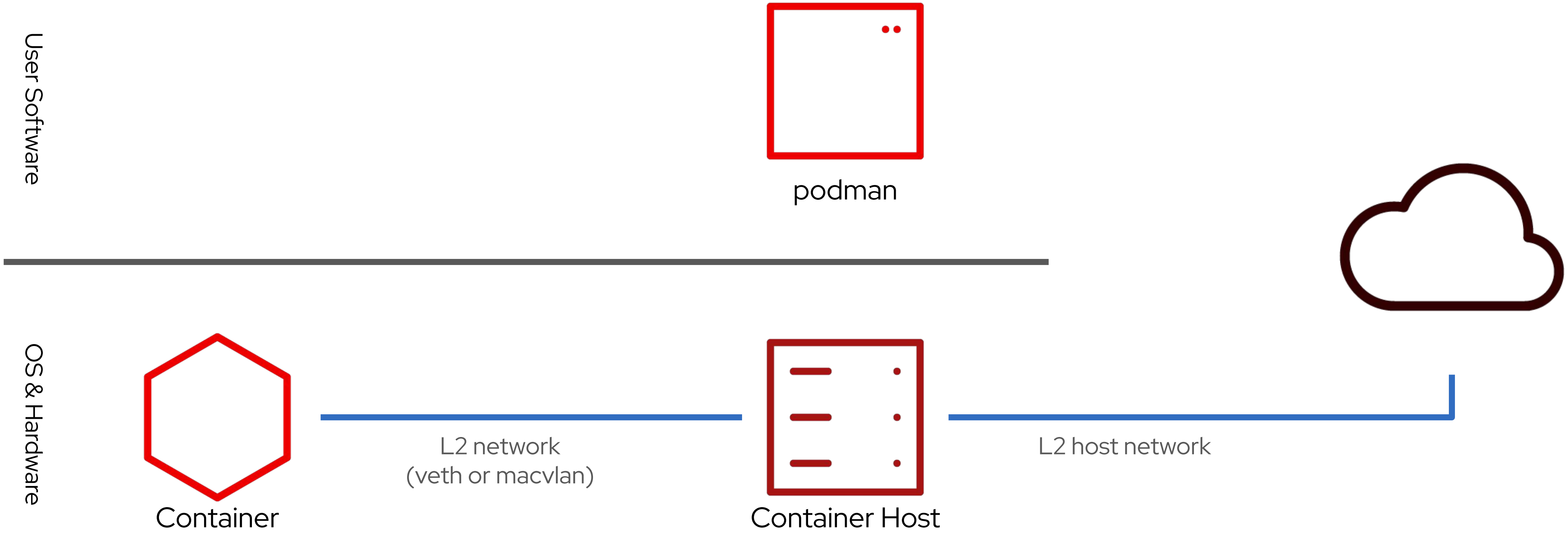
Virtual Machines

QEMU `-net user`



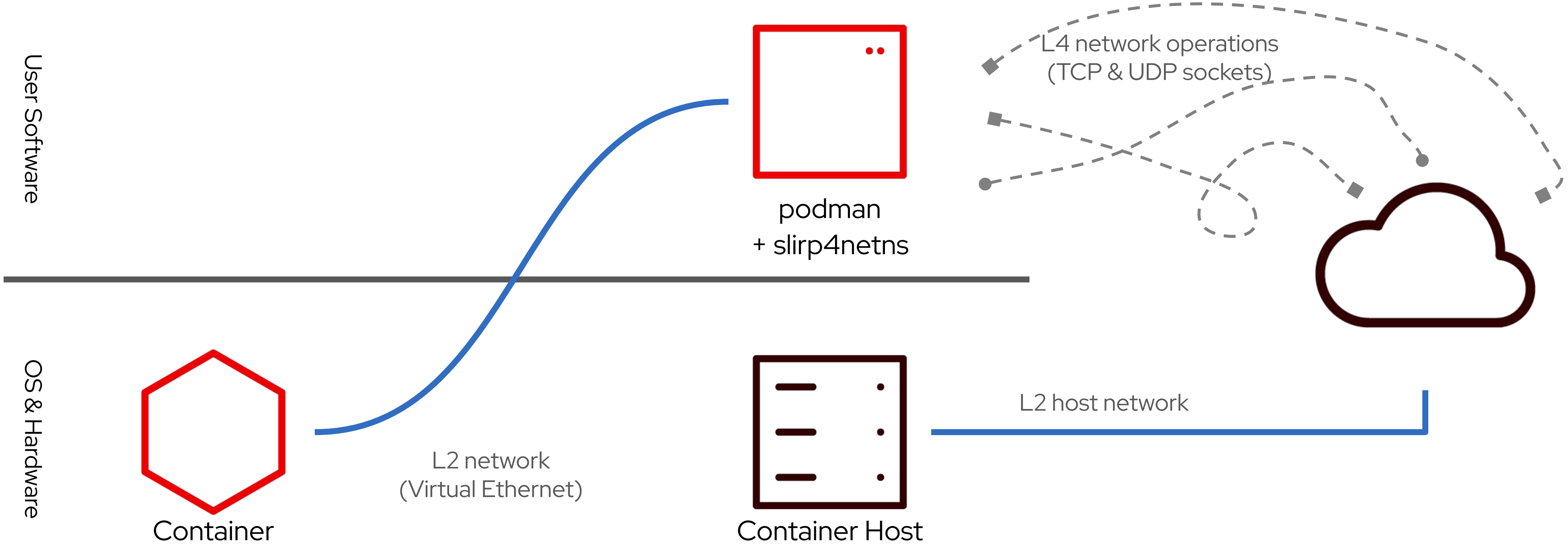
Containers

Kernel interface networking



Containers

rootless networking



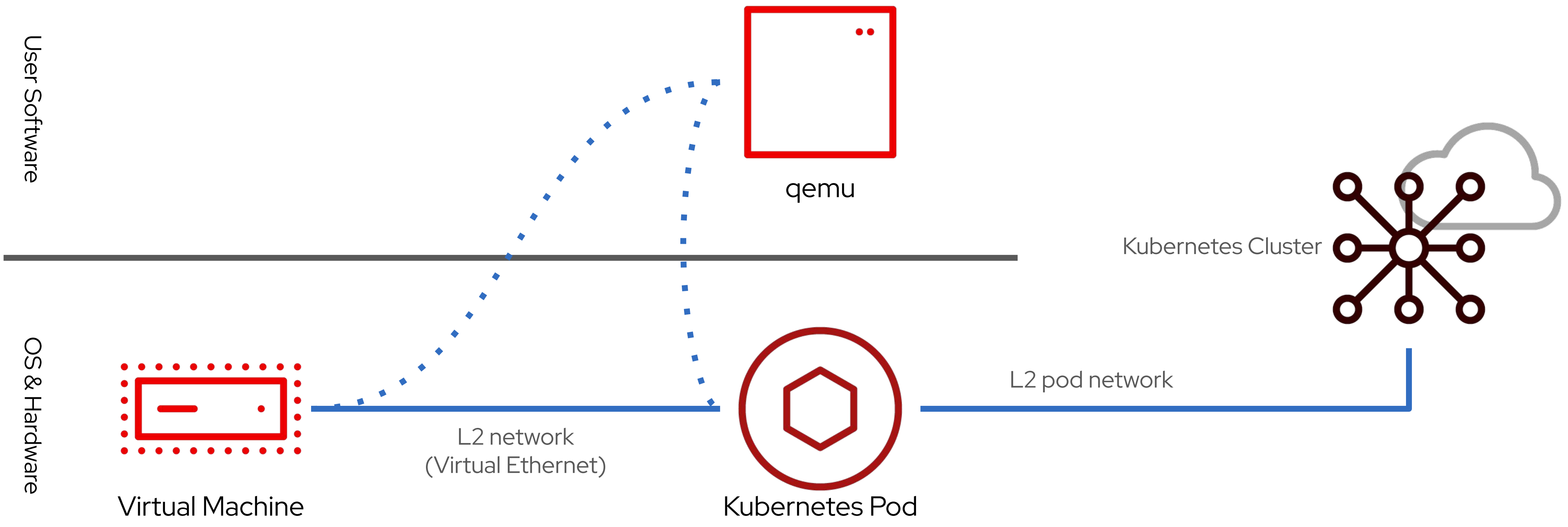
KubeVirt

Containers and VMs and Kubernetes, together at last

- Runs VM workloads within a Kubernetes cluster
  - Useful for moving legacy workloads to cloud
- Does this by running QEMU in a “launcher” pod
- VMs need to interact with the rest of the cluster
  - Such as components of the same app in containers
  - So VM networking needs to integrate with k8s network

KubeVirt

Networking



KubeVirt

Networking modes

Bridge Mode

- VM takes over pod IP address
- Other containers in the pod have no IP
  - Which breaks “sidecar” containers

Masquerade Mode

- Kernel NAT connects VM to pod interface
- Guest doesn’t see the “real” pod IP address
  - Kubernetes apps don’t like that
- Also breaks service meshes
  - Proxy side car expects traffic from userspace

KubeVirt

A different approach to networking

- Plus… getting privilege in a pod is pretty awkward in k8s
  - KubeVirt uses a bunch of tricks

- We want VM traffic to appear as if it’s coming from pod userspace
  - In other words, an L2 to L4 bridge
  - So, Slirp again, right?
    - Alas, no…

Slirp's Deficiencies

Part 1

Network Address Translation

- Guest and host on a private (10.0.0.0/8) network
  - Internally NATs to outside world
- Tends to break Kubernetes apps
  - Expects pod IPs to be cluster-global
  - Communicated between pods

Security

- Poor track record
  - Especially for resource leaks
- Fairly large attack surface
  - Complete TCP/IP stack
  - Extra features
- Old, difficult to maintain codebase
- libslirp shares process context with qemu
  - No isolation

Slirp’s Deficiencies

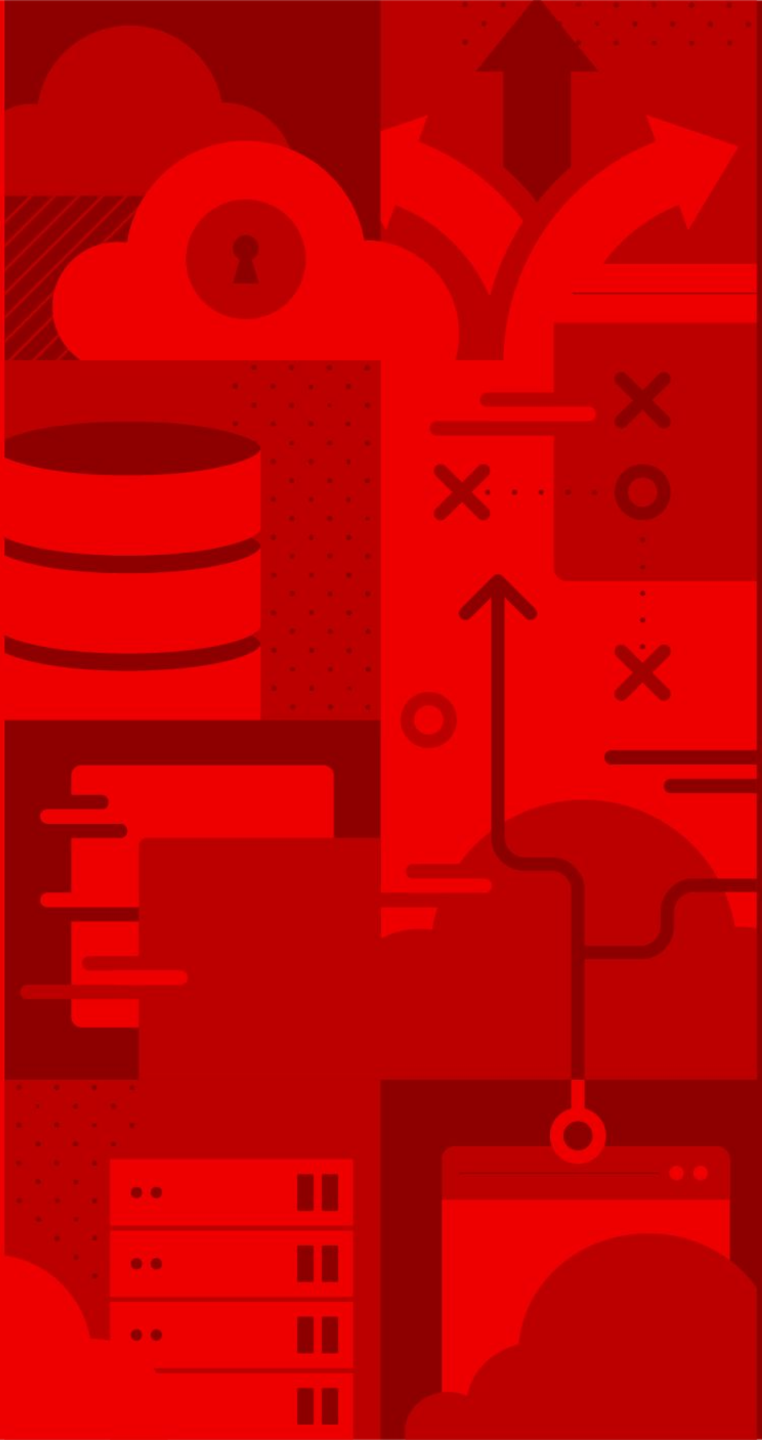
Part 2

Performance

- Not built for performance
  - Easy to keep up with dial-up speeds
  - Little effort to batch syscalls
- Modern systems need different techniques
- No support for TCP Window Scaling
  - Maximum 64KiB “in flight” without ACK

Other

- Limited IPv6 support
  - No NDP
  - No DHCPv6
  - No port mapping



The Present

Something like Slirp that's not Slirp

passt

A modern from-scratch L2/L4 bridge

Trivia

- “Plug a Simple Socket Transport”
  - (and wordplay in German)
- Originally written by Stefano Brivio
  - Starting late 2020
- I joined project May 2022
  - Second major contributor
- Written in C
  - ~9,500 LoC (excluding comments)

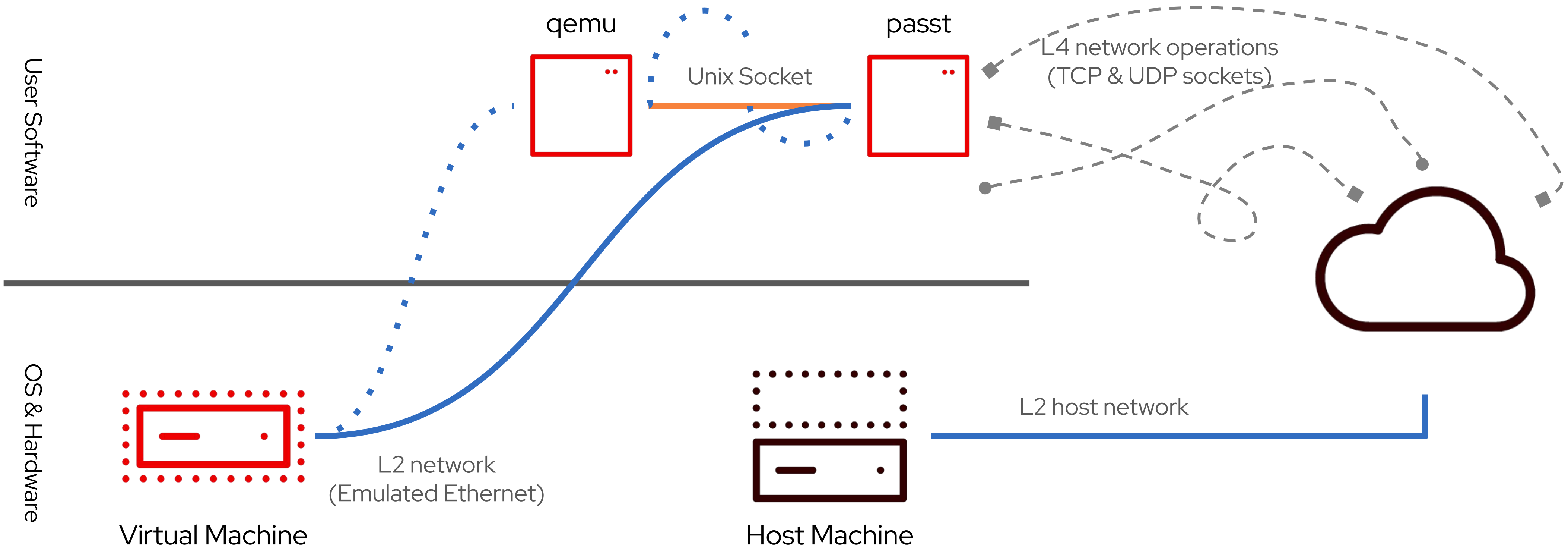
Design Goals

- No dependencies (except libc & kernel)
- No NAT
  - Well… minimal NAT
- No dynamic memory allocation
- Reasonably performant
- Security conscious
- IPv6 as first class citizen

Source:
https://passt.top

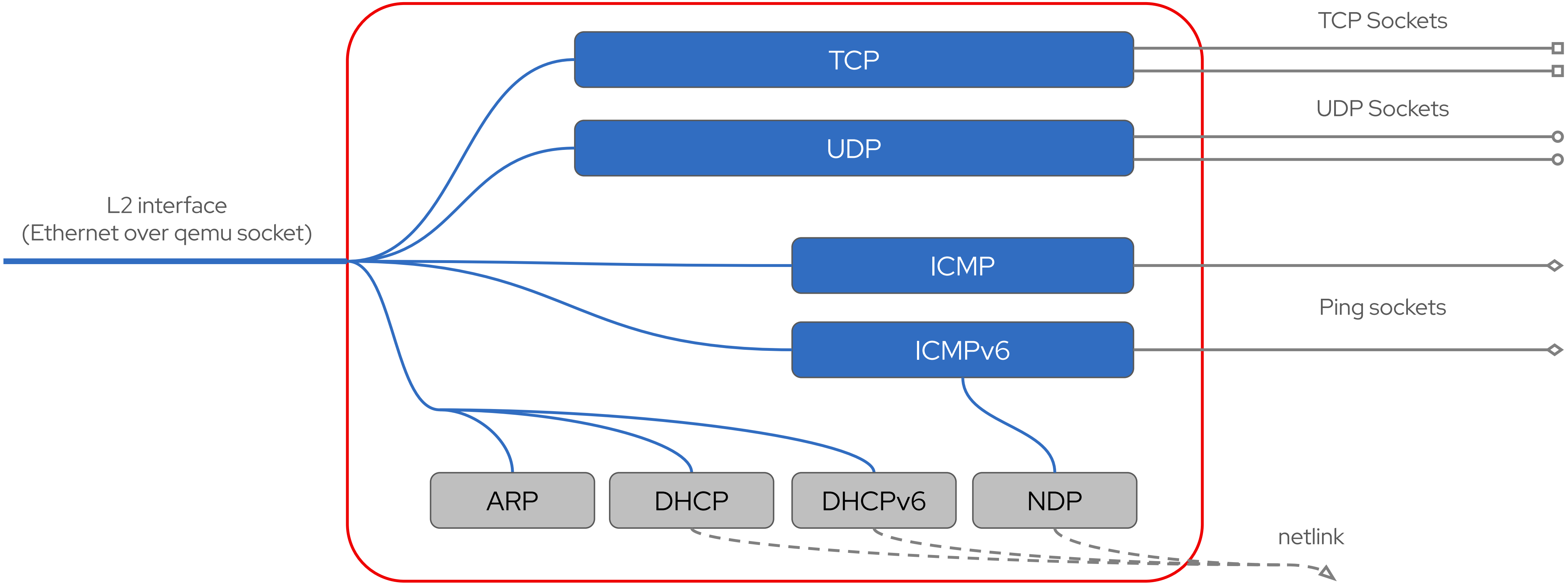
passt

With QEMU `-net stream`



passt

Design Outline



passt

Avoiding NAT

- Guest given same IP address as host
  - Including for IPv6
  - Supplied by in-built NDP / DHCP / DHCPv6
  - Taken from host interface with default route
- Works seamlessly with connections to the outside world
  - But can’t address the host itself
- Special case NATs for certain addresses
  - One guest visible IP address is mapped to 127.0.0.1 on the host
  - Another special case mapping for DNS queries

passt

Security

- Simplified TCP state machine
  - Using some newer socket APIs
  - Smaller attack surface
- No dynamic allocation
  - Enforced with `seccomp()`
- Static checkers used
  - `cppcheck` and `clang-tidy`
  - Coverity Scan, periodically
- No external dependencies
  - Other than standard C library

- Self-isolate with namespaces
- `pivot_root()` to empty tmpfs
  - No access to host filesystem
- `seccomp()` filter
  - Only 26 syscalls on x86_64
- Drop capabilities
  - In both original and isolated namespace
  - Won’t run as root
- AppArmor and SELinux profiles

passt

Performance

- TCP
  - 64kiB MTU advertised to guest
  - Segments coalesced and batched
- UDP
  - `sendmmsg()` / `recvmmsg()`
- AVX2 checksum routines (on x86_64)
- Buffer pools with partially pre-generated headers
  - Improves data locality

- Exact numbers still in flux
- Much faster than Slirp
- Comparable to single queue tap
- Slower than multi-queue tap
  - For now
  - Unlikely to ever exceed this, but we hope to be respectably competitive

passt

Miscellany

- Full IPv6 support
  - Network Discovery Protocol (NDP)
  - DHCPv6
- Flexible inbound port forwarding
  - Selected, ranges or all host side ports

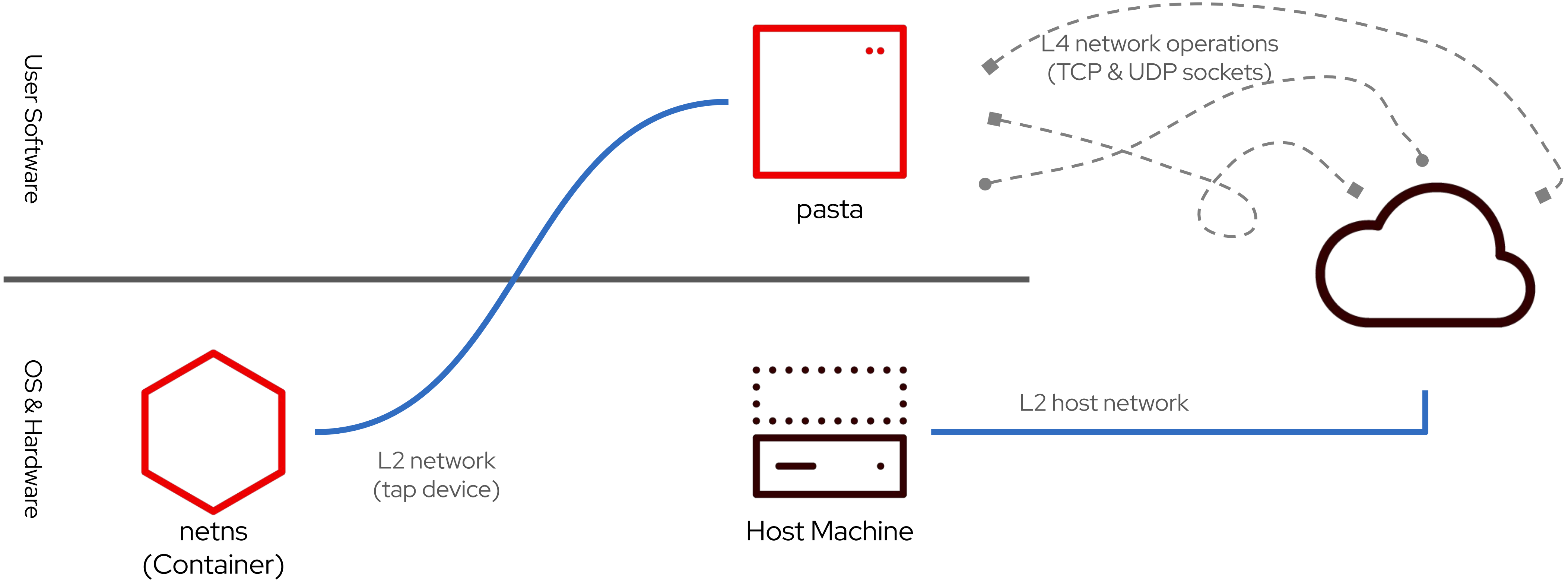
pasta

Pack a Subtle Tap Abstraction

- slirp4netns equivalent
  - tap device in network namespace instead of qemu socket
  - Requires privilege in the netns, but not on the host
- Same binary as passt
- Accelerated path for local to local communication
  - Uses `splice()` for TCP (very high performance)
- Outbound port forwarding
- Automatic port forwarding
  - By polling active sockets in `/proc`

pasta

Rootless Namespace Networking



passt & pasta

Availability

- Official packages for:
  - Fedora 35+
  - CentOS Stream 9 / RHEL9.2+
  - Debian 12+
  - Ubuntu 23.04+
- Unofficial packages for:
  - Arch Linux, openSUSE
- Underway for:
  - Alpine Linux, Void Linux

- Architecture support
  - Mostly ISA neutral, non-x86 less tested
  - Some testing on: ARM, ppc64, s390
- libc support
  - GNU libc
  - musl patches under review
- Linux only (for now)
  - Uses Linux specific kernel features
    - Alternatives exist on some other OS, but not drop-in

Source
https://repology.org/project/passt/packages
https://buildd.debian.org/status/package.php?p=passt

passt & pasta

Integration

libvirt

passt as network configuration option

Added in libvirt 9.0.0 (with SELinux bugs)

Should be fixed in libvirt 9.2.0

podman

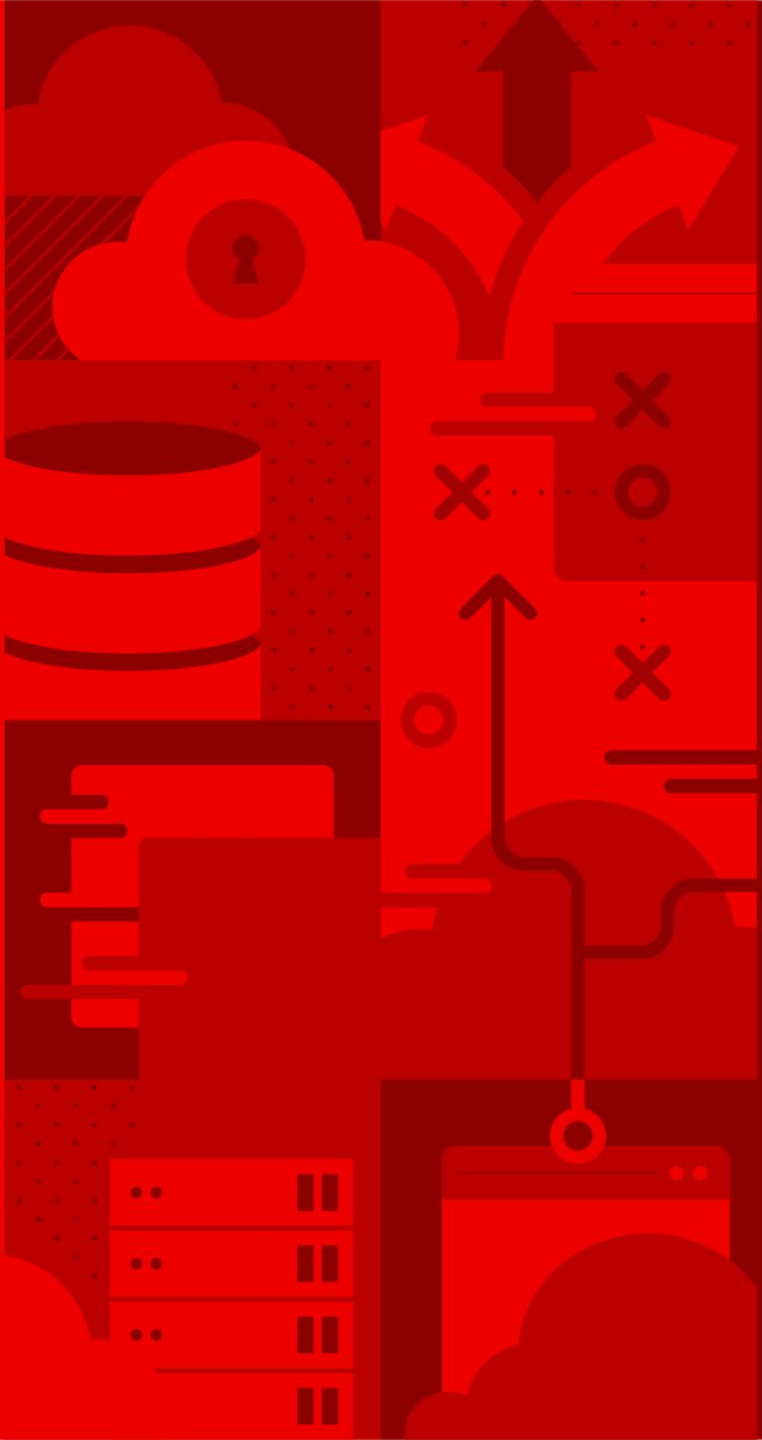
pasta as alternative to slirp4netns for rootless networking

Added in Podman 4.4

KubeVirt

passt available as a network mode

Added in 0.56 (but some complications)



The Future

There's always more to do

Future Work

Addressing some flaws

- NAT
  - Current NAT cases are inflexible with some weird edge cases
- Kernel memory usage
  - Forwarding all ports for TCP & UDP, IPv4 & IPv6:
    - ~200,000 listening sockets
- Testing / CI improvements
  - Too dependent on host’s network configuration
  - Fragile, and doesn’t test enough scenarios

Future Work

Features & improvements

- vhost-user
  - QEMU socket requires an extra copy, which vhost can avoid
  - Probably needs multithreading to take full advantage
- Fuzzing
  - Theoretically straightforward, but fiddly
- Portability
  - BSD & Darwin are the most likely candidates
- Use Rust in places?
  - Non-trivial due to low-level operations

Future Work

Additional use cases

- QEMU
  - Possible `-net user` replacement
  - Theoretically simple, but not started yet
- Kata Containers
  - Might get Kata closer to running without root
- CLAT (RFC6145, RFC6877)
- Your use case here

Contributing

...and further information

- Website:
  - https://passt.top
- Mailing lists:
  - `passt-user@passt.top`
  - `passt-dev@passt.top`
- Vulnerability reports:
  - `passt-sec@passt.top`
- Bugzilla
  - https://bugs.passt.top

- An earlier KVM Forum presentation
  - Video: https://red.ht/3T05Txu
  - Slides: https://red.ht/3JjK3Si

Credits and Questions

(and maybe a demo)

Thanks to:

- Stefano Brivio
  - Reviewed this presentation
  - ...as well as creating the project
- Laurent Vivier
  - Fixed blocking qemu bugs
  - Working on vhost-user
- Andrea Bolognani
- Alona Paz

https://passt.top

Source:
https://passt.top

Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

in linkedin.com/company/red-hat

facebook.com/redhatinc

youtube.com/user/RedHatVideos

twitter.com/RedHat

Red Hat