Hack Lyon'22 Tutorial, 15th May 2022

Running complex applications

Gaulthier Gain <gaulthier.gain@uliege.be>

Engineering and
Physical Sciences
Research Council

Agenda

Time	Presentation	Presenter
9:30 – 10:15	Running (Complex) Application in Unikraft	Gaulthier (ULiège)
10:15 – 10:30	Overview of hackathon challenges (online)	Razvan (UPB)
10:30 – 11:30	Tutorial on porting applications (online)	Razvan (UPB)
11:30 – 13:00	Work on hackathon challenges (1)	
13:00 – 14:00	Lunch 🥖	
14:00 – 17:00	Work on hackathon challenges (2)	
17:00 – 17:30	Results, final remarks	

Organization

Run some real-world applications on top of Unikraft.



Copyright © Unikraft UG.

We will use some resources

git clone
https://github.com/unikraft/docs

SQLite: *script.sql*
Redis : *redis.conf*

The qemu-guest wrapper

Pre-Installed on the VM

```
qemu-guest -h
```

`qemu-guest` is wrapper script to make the use of qemu less painful

Copyright © Unikraft UG.

The qemu-guest wrapper

```
qemu-guest -e fs0 -k build/unikernel -m 100
```

=

```
qemu-system-x86_64 -m 100 -enable-kvm -cpu host -nographic -vga none -device
isa-debug-exit -fsdev local,id=myid,path=fs0,security_model=none -device
virtio-9p-pci,fsdev=myid,mount_tag=fs0,disable-modern=on,disable-legacy=off
-kernel build/unikernel_kvm-x86_64
```

Copyright © Unikraft UG.

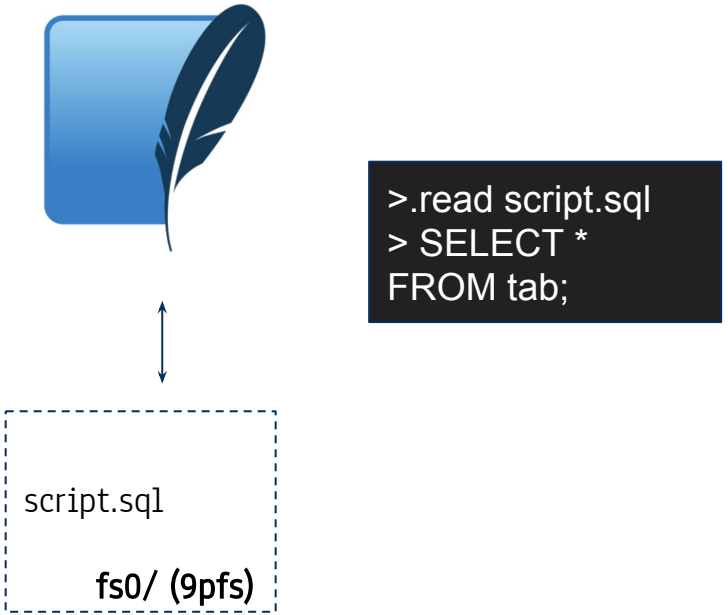
SQLite

SQLite is an embedded library that provides a lightweight database

Running SQLite unikernel

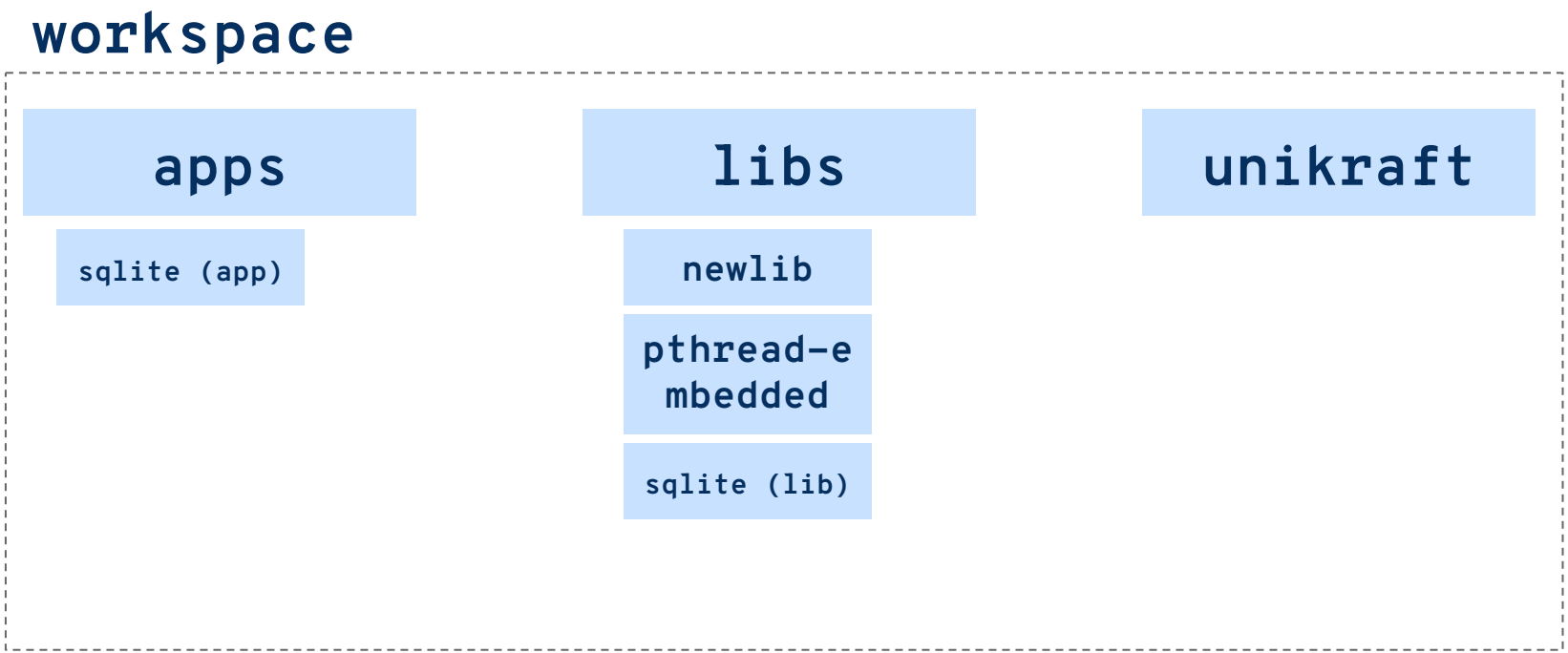
Overview:

- Use a shared folder which contains a SQL script (9pfs).
- Get a shell and execute the SQL script file to populate the database.
- Perform SQLite queries.



Copyright © Unikraft UG.

Running SQLite: workspace



Copyright © Unikraft UG.

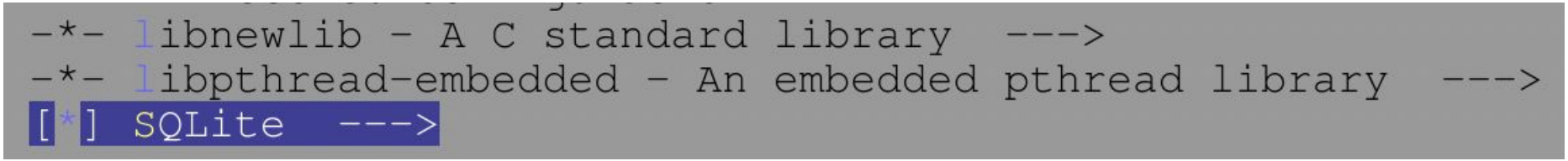
Adding a Makefile (sqlite app)

```
UK_ROOT ?= $(PWD)/../../unikraft
UK_LIBS ?= $(PWD)/../../libs
LIBS := $(UK_LIBS)/pthread-embedded:$(UK_LIBS)/newlib:$(UK_LIBS)/sqlite

all:
	@$(MAKE) -C $(UK_ROOT) A=$(PWD) L=$(LIBS)

$(MAKECMDGOALS):
	@$(MAKE) -C $(UK_ROOT) A=$(PWD) L=$(LIBS) $(MAKECMDGOALS)
```

Makefile.uk

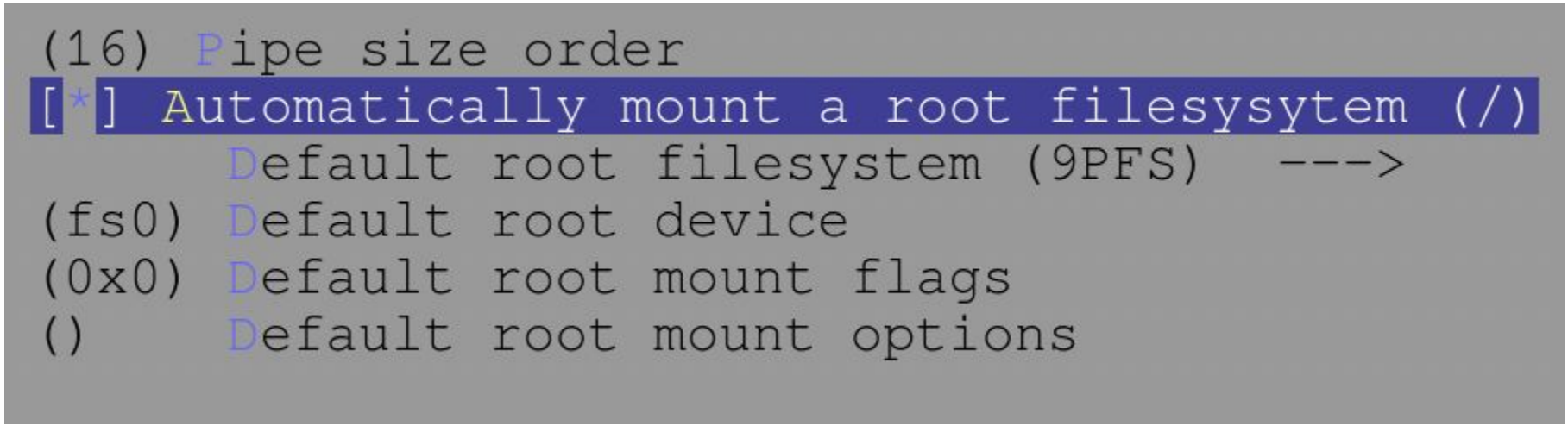
Copyright © Unikraft UG.

Select the external libraries



Copyright © Unikraft UG.

Configuring 9pfs



Library Configuration > vfscore: Configuration > 9pfs

Copyright © Unikraft UG.

Running the SQLite unikernel

```
make
```

```
qemu-guest     -k ./build/app-sqlite_kvm-x86_64 \
               -e ./fs0 \
               -m 500
```

```
SQLite version 3.30.1 2019-10-10 20:19:45
Enter ".help" for usage hints.
sqlite> .read script.sql

sqlite> select * from tab;
```

Copyright © Unikraft UG.

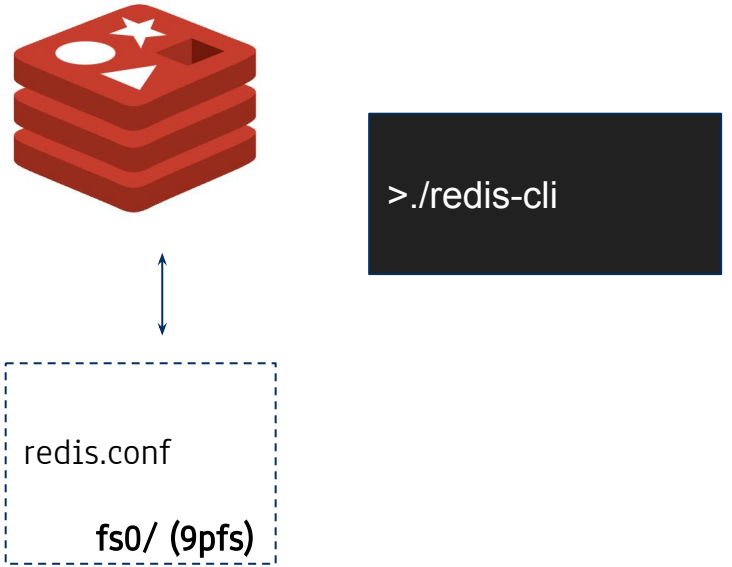
Redis

Redis is a key-value database which stores data in memory

Running Redis unikernel

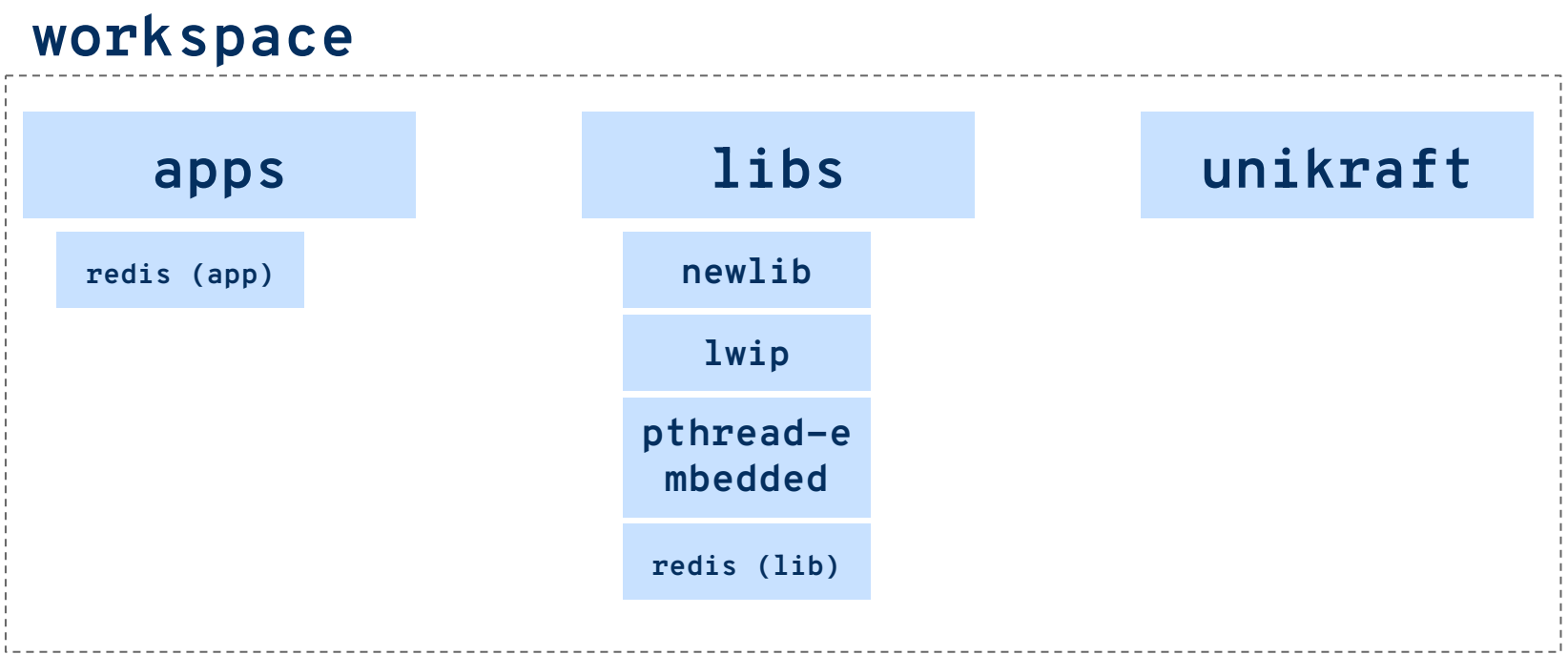
Overview:

- Use a shared folder which contains a redis conf (9pfs).
- Run redis server with an ip address (lwip) - bridge + dhcp
- Run another terminal and use redis-client to communicate with the server.



Copyright © Unikraft UG.

Running Redis: workspace



Copyright © Unikraft UG.

Adding a Makefile (Redis app)

```
UK_ROOT ?= $(PWD)/../../unikraft
UK_LIBS ?= $(PWD)/../../libs
LIBS := $(UK_LIBS)/pthread-embedded:$(UK_LIBS)/newlib:$(UK_LIBS)/lwip:$(UK_LIBS)/redis

all:
	@$(MAKE) -C $(UK_ROOT) A=$(PWD) L=$(LIBS)

$(MAKECMDGOALS):
	@$(MAKE) -C $(UK_ROOT) A=$(PWD) L=$(LIBS) $(MAKECMDGOALS)
```

Makefile.uk

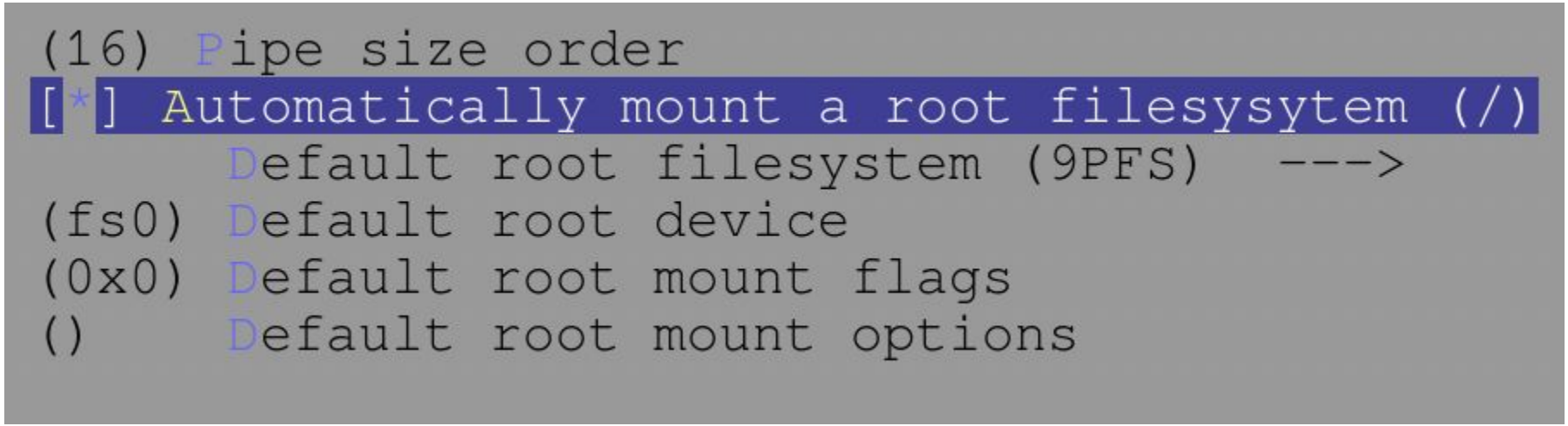
Copyright © Unikraft UG.

Select the external libraries

```
-*- libpthread-embedded - An embedded pthread library  --->
-*- libnewlib - A C standard library  --->
-*- lwip - Lightweight TCP/IP stack  --->
[*] Redis  --->

        [*]   Redis server
        [*]     Provide main function
        [ ]   Redis client
        [*]   Use internal Lua implementation
        -*-   Use internal Hiredis implementation
```

Copyright © Unikraft UG.

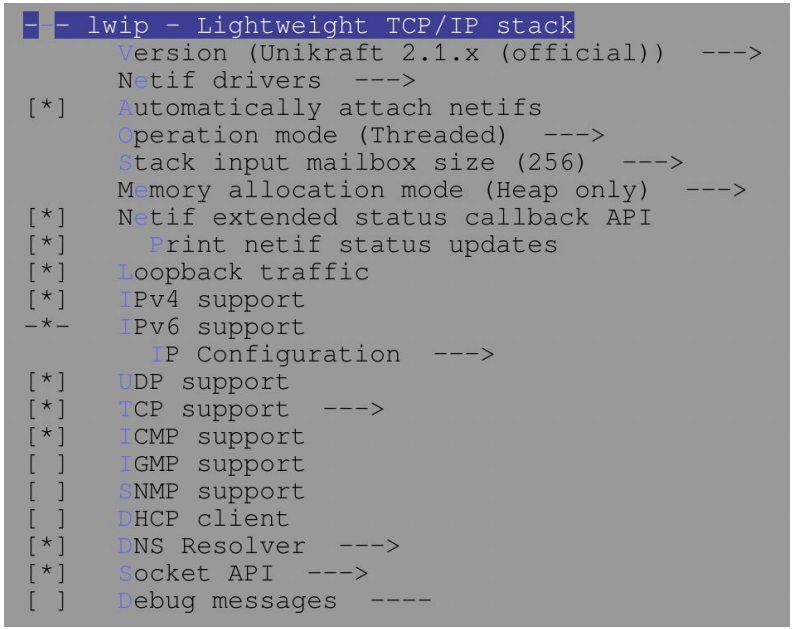
Configuring 9pfs



Library Configuration > vfscore: Configuration > 9pfs

Copyright © Unikraft UG.

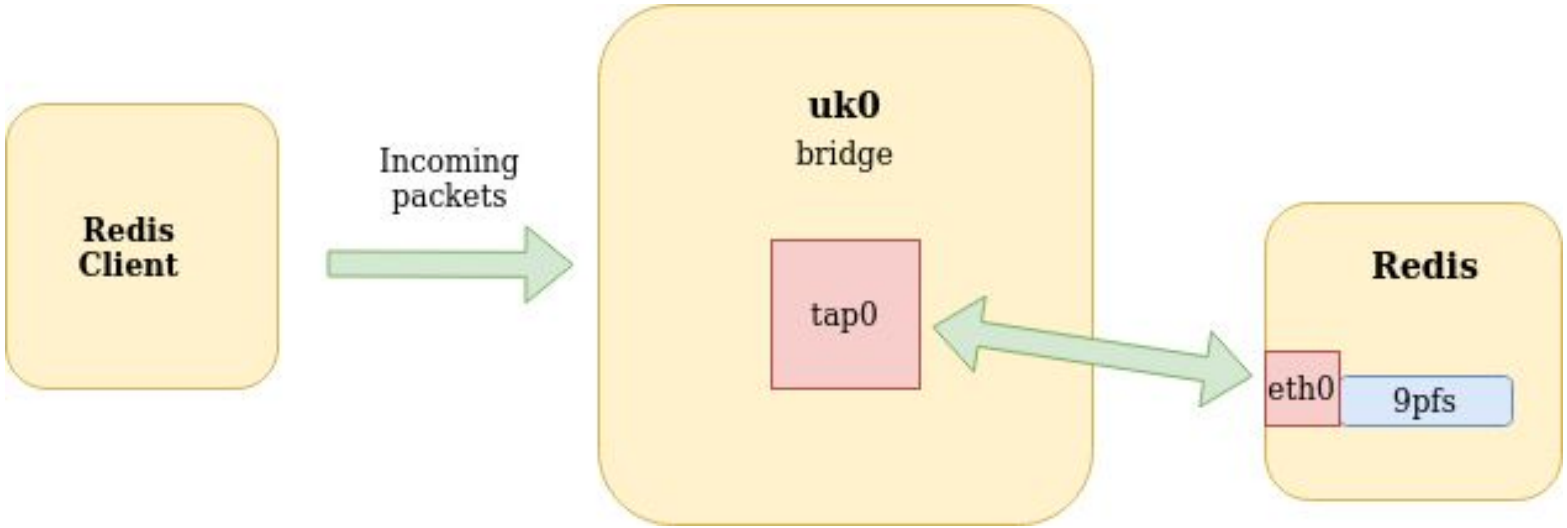
Configuring lwip



Library Configuration > lwip: Configuration

Copyright © Unikraft UG.

Setting up the network



We will create a bridge (with some specific commands) and a network interface to assign an ip to the Redis server and then we will run the redis-cli to communicate with the Unikraft Redis server.

Copyright © Unikraft UG.

Running the Redis application

```
make
```

```
qemu-guest      -k ./build/redis_kvm-x86_64 \
                -a "/redis.conf" \
                -b kraft0 \
                -e ./fs0 -m 100
```

```
$ ./redis-cli -h 172.88.0.76 -p 6379

172.88.0.2:6379> PING

PONG

172.88.0.2:6379>
```

Copyright © Unikraft UG.