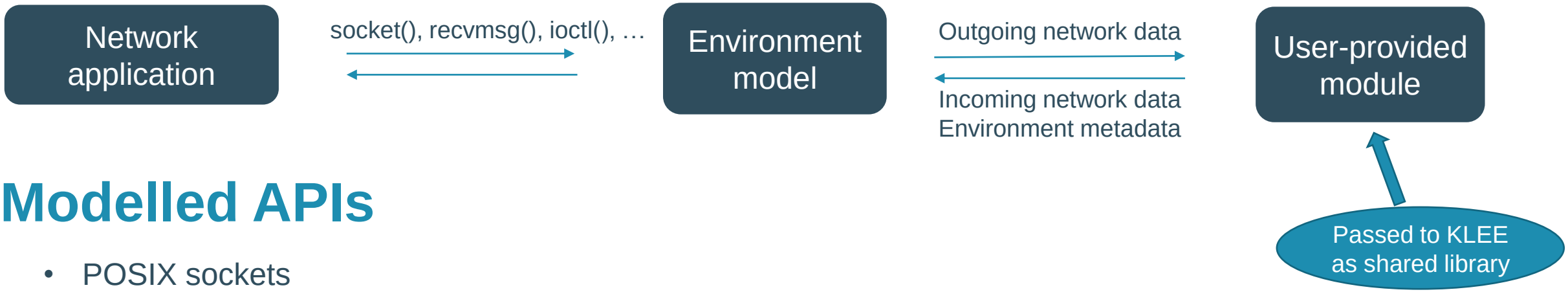
KU LEUVEN

Poster: Extending KLEE's POSIX environment model for networking

Jeroen Robben
Mathy Vanhoef

Overview

- Allow for testing real-world network software without altering application code
- Environment model for networking-related syscalls and APIs
- Network environment configurable by custom logic in user-provided *module*
  - Define incoming network data and metadata
  - (optional) Handle outgoing network data



Modelled APIs

- POSIX sockets
- Netlink (NETLINK_ROUTE)
- Some /proc and /sys entries
- Some ioctls

Reference module

- Declare network environment and incoming network data using configuration file

Network interface

```
device klee-eth0 {
    flags: 4163
    mac: 50:aa:00:00:00:00
    mac-broadcast:  ff:ff:ff:ff:ff:ff
    mtu: 1500
    address-ipv4: {
        address: 192.168.1.2
        netmask: 255.255.255.0
    }
    address-ipv6: {
        address: 2a02::0:ab43:c679
        prefix: 64
    }
}
```

Packet structure

```
packet-format packet1 {
    uint8: 0x8
    uint16:
}

packet-format packet2 {
    uint32: 123456
    char[10]:
}
```

Packet metadata

```
incoming-config queue-1 {
    device-name: klee-eth0
    domain: AF_INET
    type: SOCK_DGRAM
    protocol: IPPROTO_UDP
    receiver-ip: 0.0.0.0
    sender-ip: 192.168.1.4
    packet-queue: packet1 packet2
}
```

Preliminary results

- Dnsmasq
  - OK
  - Reproduced two **known** buffer overflow bugs due to crafted ICMPv6 or DHCPv6 packets
    - (CVE-2017-14492 and CVE-2017-14493)
- MiniUPnP
  - OK
- Radvd
  - OK (manual code edits needed to run in single-threaded mode)