PacketLab - Tools Alpha Release and Demo

Tzu-Bin Yan*, Yuxuan Chen*, Anthea Chen*, Zesen Zhang†, Bradley Huffaker†‡, Ricky Mok†‡, Kirill Levchenko*, kc claffy†‡

*University of Illinois at Urbana-Champaign, †UC San Diego, ‡CAIDA

INTRODUCTION

PacketLab is a network measurement endpoint interface design to help researchers overcome the barrier of vantage point sharing, such as required experiment porting across measurement platforms, lack of endpoint operator incentives to support new experiments, and lack of experimenter trust.

YOU BRING THE EXPERIMENT,
WE BRING THE ACCESS

For attendees interested in
trying PacketLab,
see our website
@ pktlab.github.io,
for more instructions.

TRY PKTLAB FIRSTHAND!

Software package: The PacketLab software package is now in alpha! Including within: the **reference endpoint "`pktendpt`"** and the **experiment manager utility "`pktxpmgr`"** for experiment publication and execution. Example measurements supported: DNS A record lookup, HTTP GET request issuing, ICMP echo, and traceroute.

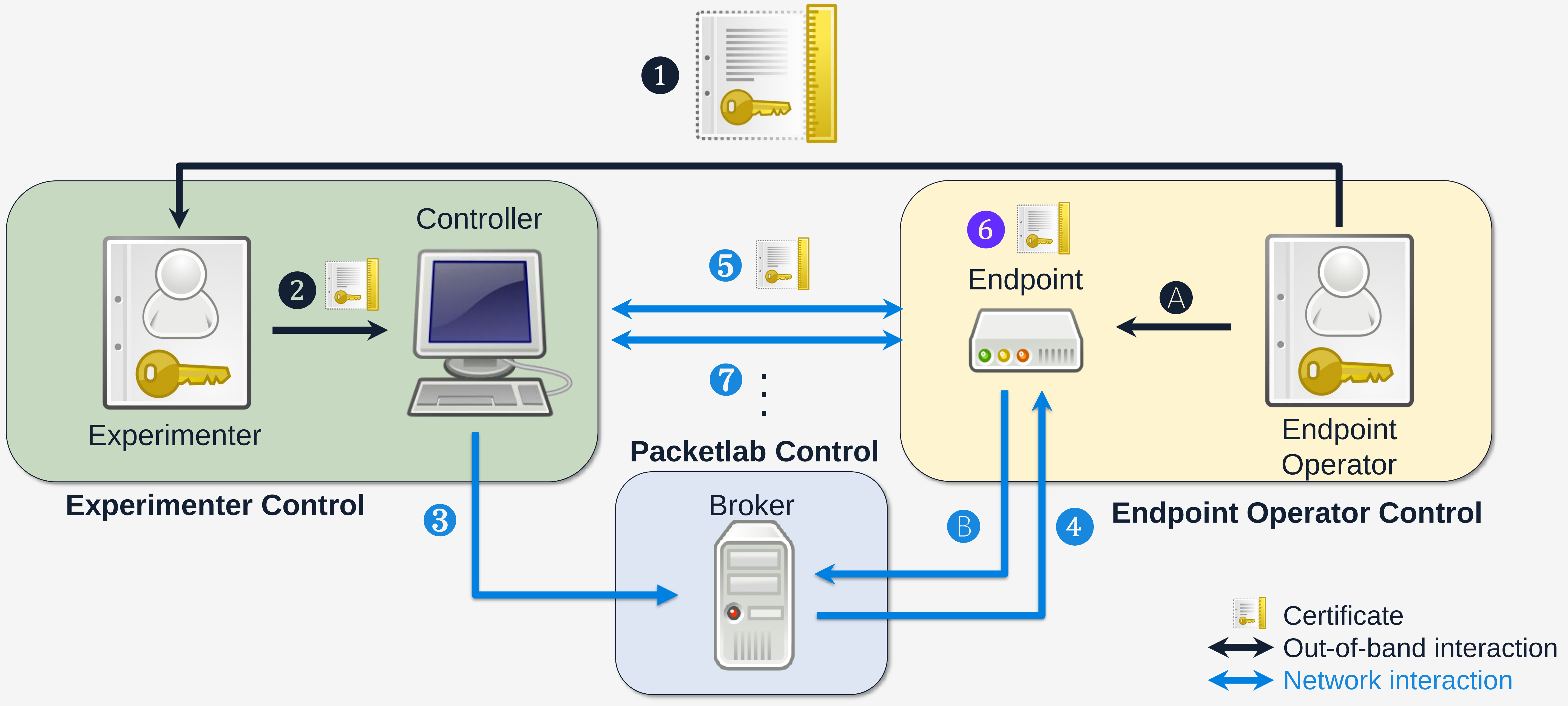
Readily available PacketLab endpoints: Cooperating with the EdgeNet team, we have deployed PacketLab endpoints on the EdgeNet cluster. Researchers can now request experiment privilege certificates from our website and set up **`pktxpmgr`** to run measurements from distributed locations.

- **`nopen(sktid,prfram,proto,rbufsz,locaddr,…)`**
  Open a socket on the endpoint with given protocol family, protocol (UDP/TCP …), and additional parameters.
- **`nsend(sktid,prfram,proto,sndtime,tag,…,data)`**
  Schedule to send data out of opened socket at specific time. "**`tag`**" is used to propagate result back to controller.
- **`ncap(sktid,prfam,proto,endtime,recvfilter)`**
  Schedule to receive "**`recvfilter`**"-matched data from an opened raw socket until a specific time. "**`recvfilter`**" is a SOCKET_FILTER-type eBPF program.
- **`nclose(sktid)`**
  Close a socket on the endpoint.

Table 1: Excerpt of protocol requests defined by the PacketLab protocol

HOW IS AN EXPERIMENT RUN IN PKTLAB?

PacketLab contains three types of agents (programs): **measurement endpoints**, **experiment controllers**, and **brokers**, as well as mainly two types of principals (human operators): **endpoint operators**, and **experimenters**. A network measurement experiment is carried out in PacketLab as follows:



Endpoint Setup:
Ⓐ: Endpoint operator sets up endpoint for experiment subscription (including endpoint operator public key)
Ⓑ: Endpoint contacts broker to subscribe for experiments

Controller Setup:
❶: Experimenter obtains experiment privilege certificate (containing restrictions) from endpoint operator
❷: Experimenter sets up controller for experiment execution (including provision of obtained experiment privilege certificate)
❸: Controller publishes an experiment to the broker

Broker Notification:
❹: Broker matches between publication and subscription, and whenever a match is found, forward the publication information to the subscribing endpoint

Experiment Execution:
❺: Endpoint contacts controller and between the two a TLS handshake is performed, during which the experiment privilege certificate is submitted to endpoint
❻: Endpoint verifies submitted certificate (with endpoint operator public key), and knows that the experimenter running the controller is authorized to do experiments
❼: Endpoint carries out the experiment via serving PacketLab protocol requests (see Table 1) from the controller, applying restrictions included within the supplied experiment privilege certificate

Acknowledgments: We would like to thank EdgeNet (https://www.edge-net.org/) for providing the computing cluster for the demo. We would also like to thank Margie Ruffin, Kevin Liu, Richard Masser-Frye, Patrick Marschoun, and Tabish Shaik for their help on the demo setup. This work is supported by NSF grants CNS-1764055 and CNS-1903612.