**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.

**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:

1. **Endpoint Setup:**
   - nopen(sktd,prfram,proto,rbufsz,locaddr,...)
     - Open a socket on the endpoint with given protocol family, protocol (UDP/TCP ...), and additional parameters.
   - nsendl(sktd,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(sktd,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(sktd)
     - Close a socket on the endpoint.

2. **Controller Setup:**
   - pktxpmgr
     - Experimenters set up controller for experiment execution (including provision of obtained experiment privilege certificate)

3. **Experimenter Control:**
   - Broker Notification:
     - Broker matches between publication and subscription, and whenever a match is found, forward the publication information to the subscribing endpoint

4. **Experiment Execution:**
   - Endpoint operator sets up endpoint for experiment subscription (including endpoint operator public key)

5. **Capabilities:**
   - Endpoints run measurement tasks from the controller and propagate result back to controller.

6. **TRY PKTLAB FIRSTHAND!**

   - Certificate
   - Out-of-band interaction
   - Network interaction

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

<table>
<thead>
<tr>
<th>Protocol Request</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sktid,prfram,proto,rbufsz,locaddr,...</td>
<td>Open a socket on the endpoint with given protocol family, protocol (UDP/TCP ...), and additional parameters.</td>
</tr>
<tr>
<td>sktd,proto,endtime,tag,...,data</td>
<td>Schedule to send data out of opened socket at specific time. “tag” is used to propagate result back to controller.</td>
</tr>
<tr>
<td>sktd,proto,endtime,recvfilter</td>
<td>Schedule to receive “recvfilter”-matched data from an opened raw socket until a specific time.</td>
</tr>
<tr>
<td>sktd,proto,endtime,sktid,prfram,proto,sndtime,tag,...,data</td>
<td>Schedule to send data out of opened socket at specific time. “tag” is used to propagate result back to controller.</td>
</tr>
</tbody>
</table>

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-1764053 and CNS-1903612.