

# A Universal Measurement Endpoint Interface

Tzu-Bin Yan with

Michael Chen, Lamyia Alowain,

Kirill Levchenko, Amogh Dhamdhere, Bradley Huffaker,

kc claffy, Mark Allman, Vern Paxson



# Quick Recap

- A measurement endpoint **interface** design unifying access to different end host based measurement platform
  - e.g. CAIDA ark, ICSI Netalyzer, RIPE Atlas ... etc.
- **PacketLab endpoint  $\approx$  VPN endpoint with network measurement support**
- Experiment logic relocated to experimenter side

# Quick Recap

- Access control/delegation?
  - Certificates
- Endpoint fine-grain control?
  - Monitors



Cpf

# Where We Are Now

- Basic endpoint implementation
- pktlab C library
  - Message, time, byte order utilities
- pyPL – Python wrapper for C library
- Filled in some missing design pieces
  - Endpoint virtual memory layout
  - Security between endpoint and controller
  - Fields in each certificate
  - Message protocol

# Today's Demo

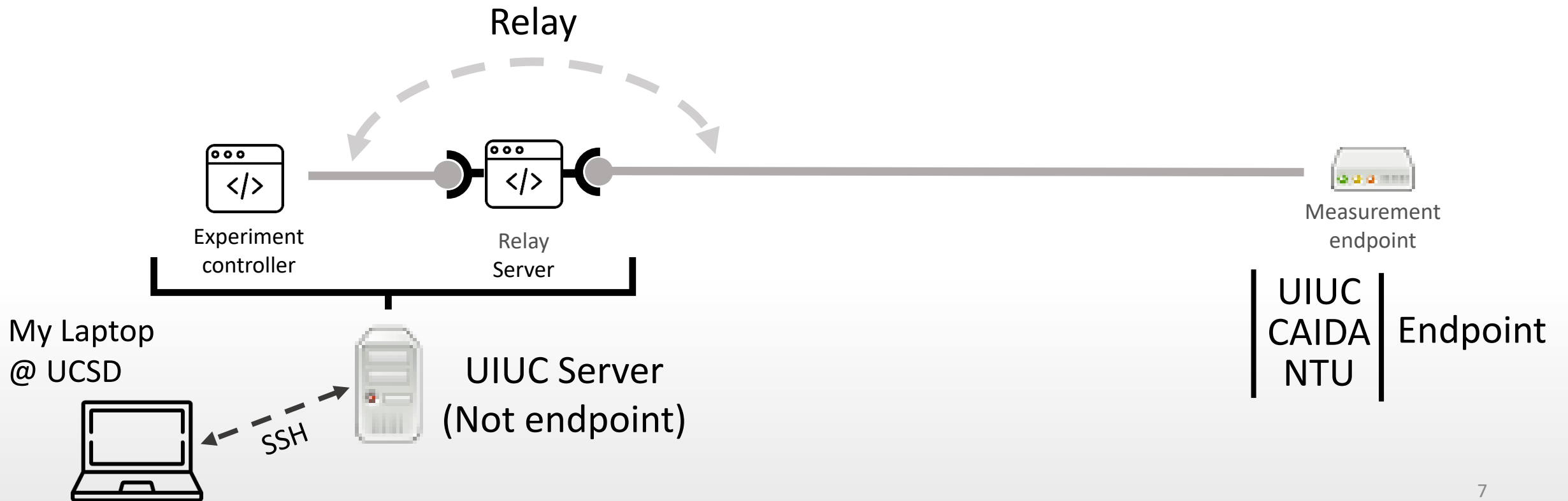
- `pl_ping`, `pl_dig`
  - Respective tools modified to use PacketLab endpoint “sockets”

# Geographical Setup

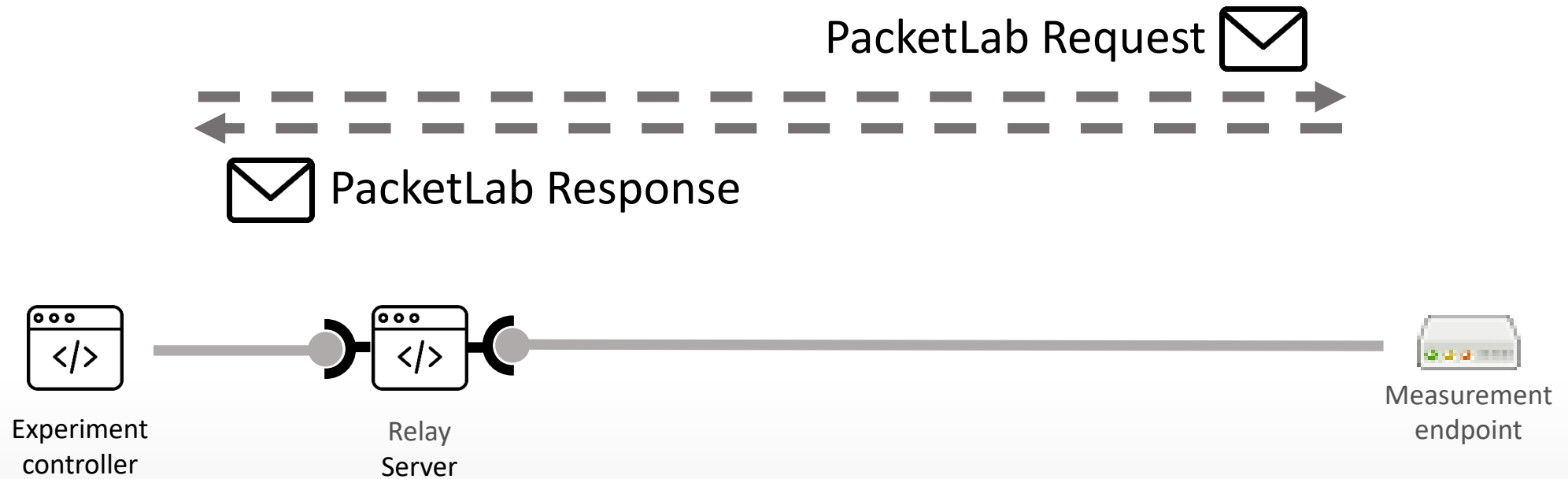


# Experiment Topology

- Relay mode



# Experiment process





# Demo in Action!!

- `pl_ping`
  - ping from different endpoint to 8.8.8.8 and see latency
- `pl_dig`
  - DNS query from different endpoint to 8.8.8.8 and see query time

Relay Port	Endpoint
10001	UIUC
10002	NTU
10003	CAIDA

# Endpoint?

- No changes to endpoint for both `pl_ping`, `pl_dig`!
  - Endpoint already done when doing `pl_ping`, `pl_dig`
  - No cost to endpoint operator for new experiments!

# Some Problems - Still Exploring

- Usability vs Simplicity
  - More or less “transport layer” socket types?
- Sharing model
  - Allow sharing of endpoint at the same time?
  - Suspend and resume of connection (with priority)?
  - Preemption?
- Other operation modes
  - Relay mode
  - Direct endpoint connect mode
  - Short lived experiment mode
- Approach: come up with new experiments and see impact of design

# Future Roadmap



## **Basic Cont'd**

1. Raw socket impl.
2. Trans-socket design & impl.

## **Sharing**

- Sharing model design and impl.

## **Monitor and Filter**

- Monitor & filter design and impl.

## **Certificates**

1. Certificate design and impl.
2. Rendezvous server impl.

Thank you!

Questions?

# Icon & Picture source

- Laptop, certificate, man icon by “Freepik” @ [www.flaticon.com](http://www.flaticon.com)
- Email icon by “Those Icons” @ [www.flaticon.com](http://www.flaticon.com)
- Program icon by “Icongeek26” @ [www.flaticon.com](http://www.flaticon.com)
- World map @ favpng.com