

Controlling Real Cloud Experiments, from BGP to the Server (and Back) with PEERING

Todd Arnold[†], Brandon Schlinker^{*}, Italo Cunha^{‡†}, and Ethan Katz-Bassett[†]
Columbia University[†], University of Southern California^{*}, Universidade Federal de Minas Gerais[‡]

What is the PEERING Platform?

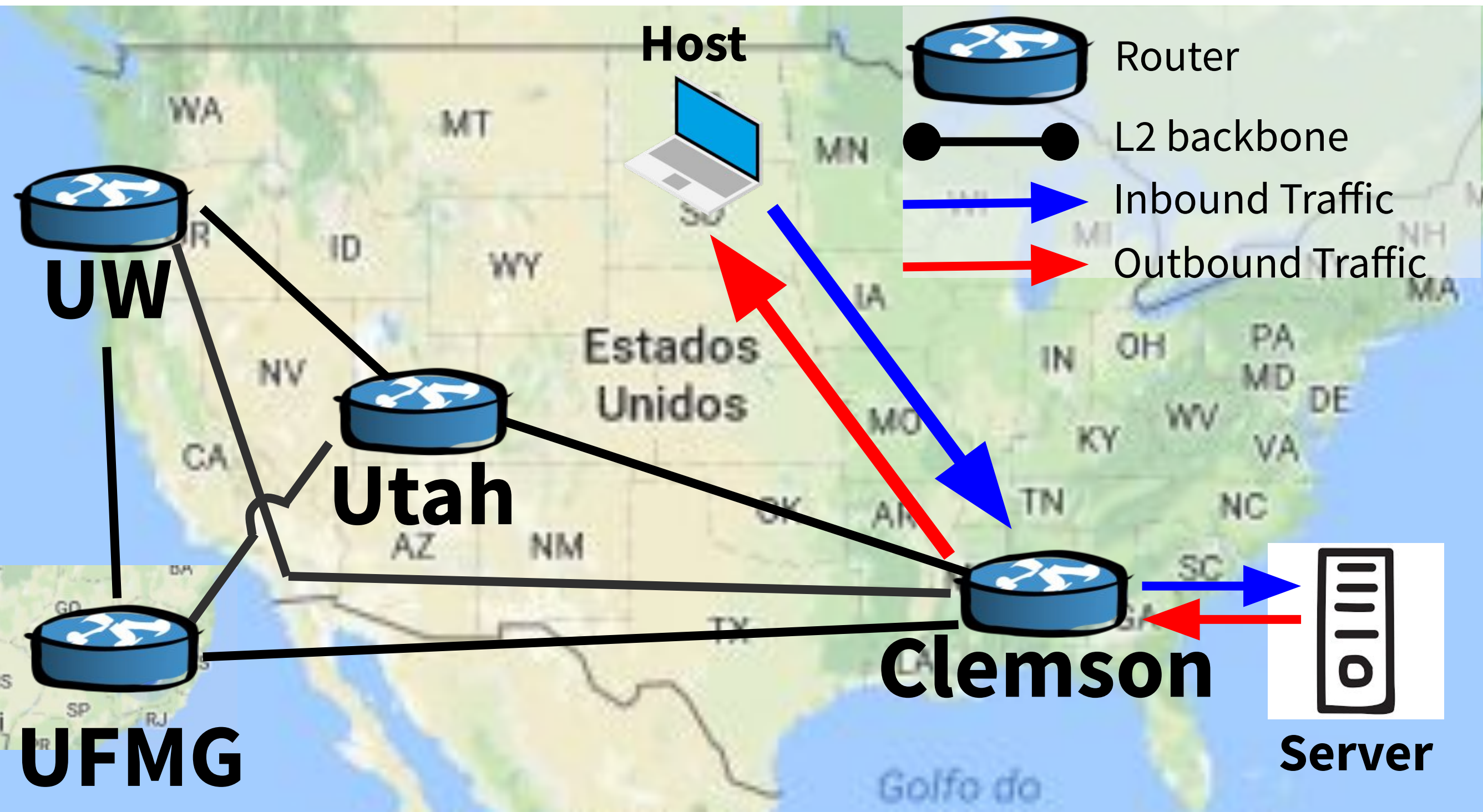
A testbed that provides researchers control over an Autonomous System (AS) with the ability to send/receive traffic and routes on the (real) Internet.

Available Resources and Capabilities

- Originate advertisements from eight ASes
- IPv4 and IPv6 address space
- 900+ peers at 12 points of presence (PoPs)
- Ability to run arbitrary client code and send/receive arbitrary traffic from within PEERING address space

New Functionality

- Federated with CloudLab, for research coupling cloud/data-center experiments with Internet route control (like Espresso/Edge Fabric, SIGCOMM'17)
- Backbone connectivity between PoPs at CloudLab locations and additional universities
 - Selectively advertise prefixes to specific PoPs and to specific peers at those PoPs across the backbone
 - Direct traffic to specific peer at remote PoP
- Support for new types of experiments, including emulating various components of cloud provider networks (e.g. traffic engineering, load balancing, anycast, etc.)



Demo Scenario

- By manipulating routing, the experiment is able to direct inbound and outbound traffic towards different PEERING PoPs
- An experiment using the PEERING client software is configured as a web **server** in the Clemson Datacenter and receives BGP tables from each PoP
 - The experiment advertises its network, 184.164.230.0/24, to Clemson so all traffic to the server enters at Clemson
 - A **host** on the Internet requests data from the server, generating ~7Mbps of traffic

Shifting Traffic

- The server decides to shift the traffic away from the Clemson PoP (for example, for TE, load balancing, congestion, etc.)
- PEERING's backbone gives the server paths to the host's network through UW & Utah
- The server shifts the traffic to the host to exit via PEERING's UW PoP
- The server then advertises its network out of the Utah PoP and withdraws it from the Clemson PoP
- All traffic to the server now enters at Utah

