# Flexible Internet Routing for Cloud Tenants and Cloud Researchers

Ethan Katz-Bassett, Assistant Professor, University of Southern California

**Abstract:** While cloud service providers (CSPs) offer compute and storage resources customized for each tenant's needs, they apply a *one size fits all* approach to interdomain networking, using a single policy for all traffic to the broader Internet. As a result, today's CSPs cannot provide an *economy service* for tenants hosting non-critical applications or a *premium service* for tenants willing to pay more for improved performance, even though CSPs maintain extensive peering with networks around the world. In addition, because CSPs are unaware of their tenant's often volatile traffic needs, they must substantially over provision their network infrastructure to handle bursts in traffic and inevitable failures.

In an analogous fashion, CloudLab offers tremendous flexibility to researchers to configure an experimental cloud's software stack, intra-data-center topology and routing, and inter-data-center topology and routing, but it currently provides no interdomain route control or flexibility to researchers. While CloudLab is suitable for a wide range of innovative research, the current design will not expose researchers to the rich interdomain connectivity that today's real world clouds offer, or allow researchers to experiment with future cloud architectures that take advantage of this interconnectivity.[1]

In this document, we first propose to design PATHS, a system that enables CSPs to provide tenants with interdomain routing tailored to their needs. Second, we will describe the additional functionality that CloudLab needs to support a PATHS prototype. Third, we will argue that this functionality would be useful for a broad class of CloudLab-based research. Finally, we will describe how we can enable this functionality on CloudLab by integrating it with our NSF CRI-funded BGP testbed called PEERING.[2]

## PATHS: Flexible Internet Routing for Cloud Tenants

Today, a CSP routes all interdomain traffic from tenants equivalently, regardless of whether it is a jitter sensitive VoIP call or an email. Cloud tenants have less control than in a colocation environment, in stark contrast to the gain in flexibility from the cloud's elastic compute and storage.

We intend to define a PATHS API that enables tenants to inform the CSP of their interdomain networking needs within a given time window by expressing the desired bandwidth, latency, path stability, resiliency, and budget for a specific class of traffic, such as traffic flowing to a chosen metropolitan region and/or associated with an application identifier. The API will benefit tenants *and* CSPs: tenants express desired properties, and CSPs gain insight into workloads, enabling them to efficiently allocate resources and adjust prices, balancing bandwidth supply and tenant demand.

The PATHS API will have to balance a number of concerns. First, the API must allow tenants to specify needs without requiring networking sophistication, and without requiring a tight feedback loop in which tenants must constantly update due to dynamic changes in their load and the CSP's available capacity, performance, and pricing. Second, PATHS must meet the needs both of tenants that simply want to express performance-sensitive versus cost-sensitive traffic and tenants that want fine-grained control. Third, PATHS must allow the CSP enough freedom in meeting tenant requests to take advantage of its upstream interconnectivity to maximize revenue while reducing costs paid to upstreams.

---

[1] Valancius, Vytautas, et al. "Wide-Area Route Control for Distributed Services." *USENIX Annual Technical Conference 2010.*

[2] Schlinker, Brandon, et al. "PEERING: An AS for Us." *ACM HotNets 2014.*

We will develop a system to transform API requests into CSP routing policies. It will take as input tenant API requests; routing, capacity, and pricing information from peers; and performance measurements of tenant traffic. The system will assign traffic to routes across and out of the CSP network, based on historical and real-time performance measurements of flows on candidate paths. To map from tenant requests to route assignments, the system will balance the utility of each assignment (relative to API requests) with the desire to minimize the costs incurred by the CSP, subject to capacity constraints.

**CloudLab + PEERING: Flexible Internet Routing for Cloud Researchers**

To build a prototype PATHS-Enabled CSP on today's Internet, we will need to emulate a CSP with control of the following: (1) tenant and hypervisor network stack, to enable marking of traffic according to tenant demands; (2) intra-cloud network, to honor markings and prioritize traffic according to the demands; (3) inter-DC and inter-PoP wide-area routing, to honor markings and carry traffic to the right egress to the public Internet; (4) interdomain routing, to control outgoing and incoming traffic to meet tenant requests. In addition, we will need (5) rich interconnectivity to ISPs around the world, to emulate the connectivity of real cloud providers. CloudLab offers flexibility to configure and control (1) and (2), and its federation with Internet2's Advanced Layer2 Service (AL2S) offers the ability to control (3). However, the current proposed infrastructure does not provide (4) or (5).

In fact, these capabilities will make CloudLab suitable for an even wider range of research and help achieve its goal of providing researchers with full control over a slice, including networking. With more and more of today's services running in public and private clouds, the interface between clouds and the Internet is an important aspect of what clouds offer, with modern cloud providers having route options at multiple points-of-presence around the world, and hundreds or thousands of peer ISPs at each location. In contrast, while CloudLab provides rich control within the cloud, it is only going to give researchers access to the Internet via default routing at the local campus or via research and educational networks.

To allow us to build a PATHS prototype--and, more generally, to allow researchers on CloudLab to experiment with interdomain routing--we propose to federate our existing PEERING[3] BGP testbed with CloudLab. PEERING has an ASN and an IP prefix, and it peers with hundreds of real ISPs, allowing it to control routing and exchange routes and traffic. We plan to expand into more Internet exchange points around the world, to approximate the rich connectivity of modern cloud providers. A researcher who acquires a PEERING slice obtains full control of its Internet routing, allowing experiments to make independent routing decisions. Combining the testbeds, we will be able to emulate a PATHS-enabled cloud using CloudLab, then connect it to PEERING to exchange routes and traffic with the public Internet.

Federation between PEERING and CloudLab will further the goals of both projects, enabling our PATHS' research and others' varied cloud research. By coupling the testbeds, researchers can emulate a modern cloud provider similar to Google or Amazon, with full control over the data center software stacks and networking, the inter-data center routing, and the peering and interdomain traffic. PEERING already has routers at Wisconsin and Clemson, two of the CloudLab sites. We imagine working with the CloudLab team to integrate the testbeds, which could be done using the GENI APIs, with a specification similar to CloudLab profiles to configure PEERING for an experiment, and AL2S to connect sites.

---

[3] **P**airing **E**mulated **E**xperiments with **R**eal **I**nterdomain **N**etworking **G**ateways, also known as Transit Portal and BGP-Mux.