Transit Portal: BGP Connectivity as a Service

Vytautas Valancius, Hyojoon Kim, and Nick Feamster
School of Computer Science, Georgia Tech

Categories and Subject Descriptors

C.2 [**Network Protocols**]: Routing Protocols

General Terms

Design

1. INTRODUCTION

We demonstrate *Transit Portal*, a system that provides on-demand BGP Internet connectivity to multiple ISPs. Transit Portal provides connectivity to any virtual network or distributed service that needs to control its inbound and outbound route control. Examples of such services include virtual networks and distributed services in cloud computing environments (e.g., Amazon's EC2) that need to control inbound and outbound traffic.

Transit Portal offers a conventional BGP session interface to its clients and presents the appearance of a direct connection to one or more upstream Internet service providers, without requiring each client to establish an explicit contract with each upstream provider. Transit Portal aggregates client sessions and provides a single, stable BGP session to upstream providers. As shown in Figure 1, Transit Portal can be deployed in geographically distributed popular exchange points, close to the local ISPs. Downstream networks (e.g., researchers, experimenters, operators of cloud services) peer with these upstream ISPs though BGP sessions terminating on Transit Portal using tunnels. The main goal of this demonstration is to show the following three aspects of Transit Portal:

- **Upstream and downstream connectivity for virtual networks.** A Transit Portal in Atlanta will be connected to an upstream "provider" (i.e., the border routers of the campus at Georgia Tech). A downstream client network, connected to the Transit Portals via OpenVPN, with full BGP connectivity from both the Atlanta TP. This downstream connectivity feature is new and shows that *any* host that is capable of running an OpenVPN client can connect to the BGP Mux to get data plane connectivity.
- **Resource management across services.** One of the primary challenges with the design and implementation of the Transit Portal is to manage the limited set of resources across multiple downstream virtual networks. In particular, the system must keep track of which resources have been allocated to which downstream virtual network, as well as enforce those resource allocations and manage potential conflicts.
- **Cloud-based applications that can make use of Transit Portal.** We will show how various applications can make use of Transit Portal; in particular, we will demonstrate one or two cloud applications making use of Transit Portal to control its inbound traffic.

Differences from last year's demonstration. This demonstration significantly extends the demonstration of Transit Portal that we presented at *SIGCOMM 2009*. Last year, our demonstration focused only on showing that route control was possible; it did *not* demonstrate how the system could manage resources across multiple virtual networks simultaneously, nor did it demonstrate any applications using the Transit Portal. This year's demonstration will focus on these two aspects.

SIGCOMM'10, August 30–September 3, 2010, New Delhi, India.
ACM 978-1-4503-0201-2/10/08.

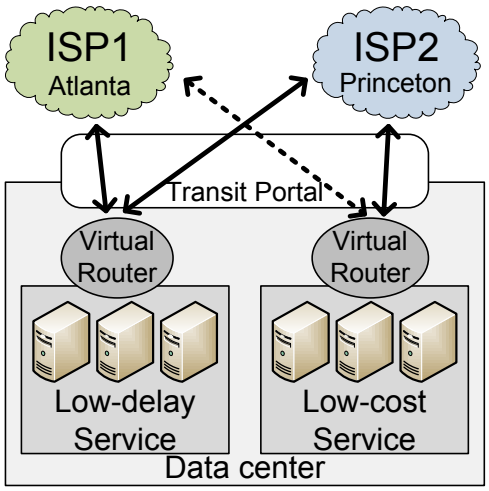

Figure 1: Transit Portal demonstration setup.

To attract more users, to encourage more sites to deploy Transit Portals on their networks, and to demonstrate TP features, we will run an interactive TP demonstration during *SIGCOMM 2010* conference. We plan to make this demonstration as interactive as possible, allowing the conference attendees themselves to use the TP in real time from example services running on their own laptops. Section 2 provides an overview of the design and implementation of Transit Portal, and Section 3 summarizes the demonstration.

2. DESIGN

We provide a brief overview of the Transit Portal design as it pertains to upstream and dowstream connectivity, resource management, and example applications. Our *USENIX* paper offers more details [5].

Upstream and downstream connectivity. *We will show how the Transit Portal can provide inbound and outbound traffic control to a client service.* This aspect of the demonstration will highlight two parts of Transit Portal's design: the control plane and the data plane. The *control plane* terminates BGP sessions from both the service providers and from Transit Portal users. Transit Portal presents the illusion of the direct connectivity to an upstream provider for each such session: the client sees the AS number of the provider it connects to and the updates from providers are propagated with minimum delay and no modifications except for the next-hop address. The *data plane* of the Transit Portal router has two components: (1) a mechanism for delivering the traffic from downstream network to Transit Portal router, and (2) a mechanism

for routing such traffic to the appropriate ISP. In our demonstration, we show the data plane with GRE tunneling; in this setup, Transit Portal forms BGP sessions with the downstream networks over GRE tunnels.

Resource Management. *We will show how the Transit Portal can manage resources across multiple simultaneous running client networks.* Transit Portal management plane takes user requests and provisions necessary control plane and data plane resources. We implement the management plane in the spirit of the Slice-based Facility Architecture (SFA) [1]. This management plane approach is actively developed by projects in the GENI [3] initiative, such as ProtoGENI [4]. The primary components of the SFA are the Component Manager (CM) and Aggregate Manager (AM). CM instances reside on each TP node and is managed by centralized AM. The AM is collocated with a Web interface, using which users can request BGP resources.

Example Applications. To demonstrate the utility of Transit Portal, *we will show how TP can be used to provide route control for two example applications*:

- **Reliable, low-latency distributed services.** A service provider that hosts some service for some domain may wish to provision both hosting and anycast connectivity in locations that are close to the clients for that domain. In our demonstration, *we will show how Transit Portal can provide route control to an anycasted DNS service like DONAR* [2].
- **Service migration.** A service provider might wish to migrate a service from one data center to another to cope with fluctuations in demand. Today, service providers must use DNS for such migration, which can degrade performance and does not permit the migration of a running service. Using Transit Portal, a provider can use route control to migrate a service and re-route traffic on the fly, taking DNS out of the loop and enable migration of running services. In our demonstration, we will show how Transit Portal can facilitate *live service migration from one data center to another*.

3. DEMONSTRATION

Goals. The primary goal of the demonstration is to *show how any client network that has the appropriate credentials can establish Internet transit peering through Transit Portal*. A second goal of the demonstration are to show applications running on the Transit Portal. A third goal of the demonstration will be to have conference attendees actually use Transit Portal from their own machines and get them familiar with the interface for configuring connectivity. We will show how different clients can connect to Transit Portal with different tunneling technologies. We will show how clients can connect to the Transit Portal with a variety of tunneling technologies (e.g., OpenVPN, GRE tunneling). The final goal will be to show various features of the Transit Portal such as automated prefix withdrawals upon session failure, and stable connections to upstream ISPs even when the downstream connections fluctuate.

The demonstration will show how downstream networks connect to ISPs though Transit Portal and how they can control these connections in real time. We will use a poster to explain the topology setup. The Transit Portal will be set up as shown in Figure 1, with the portal routers in Atlanta and Princeton. Both portal routers use local Internet service providers for upstream connectivity. The demonstration will have a virtual network on Emulab that is connected to service providers though Transit Portal routers in both locations. The Transit Portal client at the conference venue will be also emulated on a laptop.

Outline. We provide an outline of the demonstration, which will consist of three parts: (1) showing how clients can use Transit Portal to achieve transparent upstream connectivity; (2) showing how virtual networks that use Transit Portal can take advantage of IP routing to respond to failures and balance traffic loads; (3) showing the example applications that we described in Section 2.

Part 1: Upstream Connectivity and Resource Management. To demonstrate how a client can use Transit Portal to establish upstream connectivity—and how the Transit Portal manages resources as new downstream clients claim resources—the demonstration will proceed as follows:

- A client logs on to Transit Portal Web site and requests the access to the Internet at all service locations. Using this interface, a user selects the tunneling technology and enters the type of his connection. Users can use their own AS numbers and prefixes, or they can use private AS numbers and "lease" IP address space from Transit Portal.
- The client sets up the tunnels and BGP session to Transit Portal, as shown in Figure 1. The client also announces an IP prefix via its BGP session.
- To demonstrate control and data plane connectivity we will show client's BGP advertisements and *traceroutes* to new prefix through "looking-glass" servers around the U.S.

Part 2: IP Routing. When the session between virtual networks and Internet service providers is established, we will demonstrate session transparency to upstream ISPs and the availability features of Transit Portal as follows:

- The client re-announces one of the prefixes on the session to Atlanta Internet service provider with a longer AS path. This announcement will cause more traffic to arrive over the session to the Princeton Internet service provider.
- The client shuts down the tunnel to the Atlanta service provider. Transit Portal detects the tunnel failure and withdraws the east coast prefix. When the client session is shut down, the session to the service provider in Atlanta is unaffected, and the traffic is rerouted to the Seattle router.

Part 3: Example Applications. We will show two example applications running on the Transit Portal. For the IP anycast experiment, we will run an anycasted DNS service at both the Princeton and Atlanta TP sites and show how traceroutes from different locations arrive at different service replicas. For the service migration experiment, we will show how a live TCP connection running between a client and a service in one data center can be migrated to the service in the other data center without interruption.

Acknowledgments

This work was funded by NSF award CNS-0626950, NSF CAREER award 0694974, and a grant from the GENI project office "Bringing Experimenters and External Connectivity to GENI".

REFERENCES

[1] Slice-based facility architecture. `http://www.cs.princeton.edu/~llp/arch_abridged.pdf`, 2010.

[2] DONAR. `http://sns.cs.princeton.edu/projects/donar/`.

[3] GENI: Global Environment for Network Innovations. `http://www.geni.net/`.

[4] ProtoGENI. `http://www.protogeni.net/`, 2010.

[5] V. Valancius, N. Feamster, J. Rexford, and A. Nakao. Wide-Area Route Control for Distributed Services. In *Proc. USENIX Annual Technical Conference*, Boston, MA, June 2010.