Layer-Two Peering across SAVI and GENI Testbeds using HyperExchange

Saeed Arezoumand, Hadi Bannazadeh, and Alberto Leon-Garcia
Department of Electrical and Computer Engineering
University of Toronto, Toronto, ON, Canada
Email: s.arezoumand@mail.utoronto.ca, hadi.bannazadeh@utoronto.ca, alberto.leongarcia@utoronto.ca

***Abstract*—We demonstrate the peering of virtual networks between the SAVI and GENI testbeds using HyperExchange [1], a software-defined exchange fabric. The exchange is deployed between the physical networks of the two testbeds. Specifically, a layer-two WAN including nodes in SAVI testbed is peered with a VLAN in GENI testbed without using encapsulation and overlays. Each of these testbeds has a different logic to create and manage layer-two networks, so this demonstration shows how the HyperExchange is protocol-agnostic and allows tenants to create networks across dissimilar networks.**

***Keywords*—Federation, Software Defined Exchange, Software Defined Infrastructure, Orchestration**

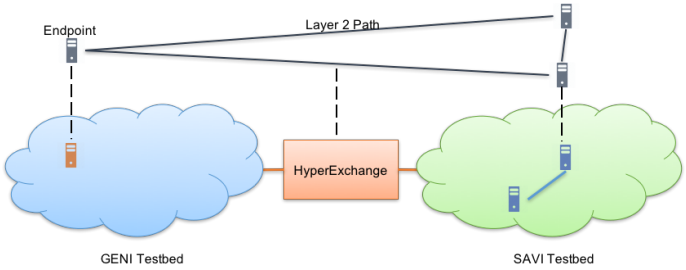

Fig. 1. The overall topology of demonstration

I. Introduction

A. Background

Software Defined Infrastructures promise to redefine the network foundation of carrier networks and Internet Service Providers (ISPs). SDI drives this foundation towards integrated, multi-tenant clouds that offer programmable and fine-grained resources, including but not limited to virtual networks. In this new model, the traditional role of ISPs can be separated into two roles: Infrastructure Providers (InPs), that provide virtualizable network infrastructure; and Service Providers (i.e. tenants) that use virtual networks to provide services for end-users [1]. In this new model, there is now an opportunity fora Service Provider to deploy its Virtual Network over multiple InPs, thus achieving greater geographic coverage and creating new cost optimization strategies.

The SAVI [2] and GENI [3] testbeds are two examples of the Infrastructure Providers [A, B]. In both of these testbeds, heterogeneous resources can be provisioned on-demand and applications can be deployed dynamically across multiple geographically distributed data-centers. Each of the these InPs provides the capability of provisioning layer-two networks and allows a tenant to stitch layer-two paths between resources using orchestration tools. However, since each of these testbeds uses a different logic to provision virtual networks, there has been no capability to enable inter-networking of VNs across both testbeds, other than using overlays. The problem with overlays is that they continue the ossification of the current Internet and provide a narrow solution for specific use cases, not a holistic uniform approach for network federation of testbeds. In this demonstration, we present cases of layer-two peering using HyperExchange to establish networks that span over both testbeds.

[1]The main paper of this work is presented in the main IFIP/IEEE IM 2017 conference.

II. Demonstration Architecture

A. SAVI Testbed

SAVI, Smart Applications on Virtual Infrastructure, is a distributed research testbed in Canada and with an architecture based on a Software-Defined Infrastructure (SDI). The SDI model in SAVI testbed integrates both SDN and traditional cloud computing by providing an abstraction from the underlying infrastructure resources, exposing control over compute, network, storage, and other resources via a programmatic interface. SDN applications are enabled on SDI using the Network Control Module, a technology-agnostic SDN platform with access to infrastructure-wide multi-resource information. The design of a SAVI node leverages the open-source cloud computing platform OpenStack, and the de facto standard SDN protocol, OpenFlow. Each SAVI node is built around a network fabric that utilizes OpenFlow-enabled switches, and so it is possible to leverage an OpenFlow controller for controlling all the traffic within a node.

The layer-two networks in overall SAVI testbed are managed by an end-to-end orchestrator that operates on top of multiple SDI nodes. In this architecture, each region is an SDI node with an autonomous SDI manager, dubbed JANUS, and a single testbed-wide centralized orchestrator transforms user requests for layer-two network management to a set of API calls to the SDI manager of each region. The isolation between layer-two networks is defined based on MAC-based access controls in OpenFlow switches.

B. GENI Testbed

GENI, the Global Environment for Networking Innovation, is a distributed testbed sponsored by the U.S. National Science Foundation (NSF) for the development of future-oreinted network experiments. GENI is built on federation of data-centers

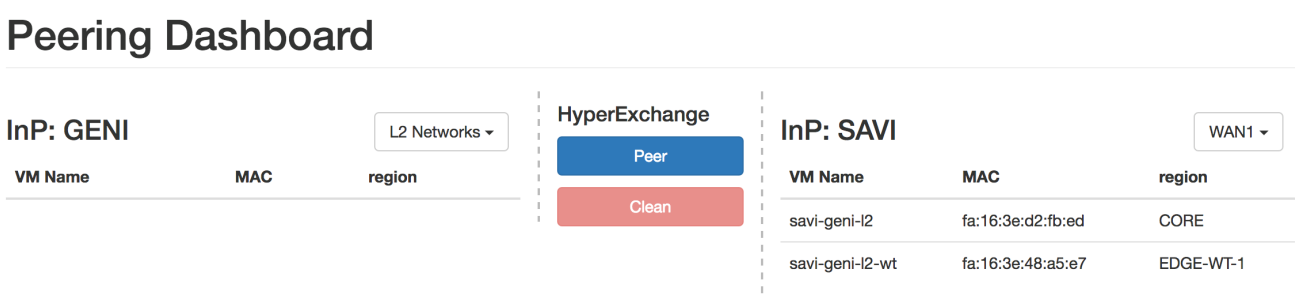

Fig. 2. Peering Dashboard, GUI of HyperExchange

Fig. 3. WAN Dashboard, GUI of SAVI orchestrator

across multiple sites and universities in the U.S. Unlike SAVI, layer-two networks in GENI are isolated using VLAN tags. To setup a layer-two connection between VMs, a user must define a topology in the form of an Rspec and pass it as a request to the Stitching tool. This tool, acting as an orchestrator, will then call Aggregate Managers involved in the topology to request the necessary resources, including VMs and VLAN tags.

C. HyperExchange

HyperExchange is a protocol-agnostic and software defined exchange fabric for peering of InPs and their hosted virtual networks [4]. It provides inter-domain tenant authentication and authorization for network control and is architected as an autonomous SDI node that inter-connects participating networks. A novel data model is provided by HyperExchange to specify heterogeneous networks in a uniform way. Once networks are specified, a tenant can define policies to allow traffic exchange and enable peering of networks.

III. DEMONSTRATION DESCRIPTION

A prototype of HyperExchange has been implemented and deployed between SAVI and GENI testbeds. This prototype is an extension of the SDI-manager reference model, written in Python. In the current implementation, the user can specify a VN using a network specification API. The authorization module uses a private API to get VN attributes and to authorize the specification. The current implementation supports a combination of OpenFlow actions as the policies described by the user. The authorization module uses remote APIs for network specification requests, and it uses local specifications of network domains for policy flow entries. The prototype of HyperExchange is built using a hardware switch between the underlying interconnection of SAVI and GENI testbeds. Each of the SAVI and GENI networks are connected to the exchange point via a single dedicated physical port.

During the demonstration, we first create a layer-two WAN in the SAVI testbed using the WAN dashboard shown in Figure 2. Layer-two networks in SAVI are established by end-to-end path stitching on OpenFlow switches using the MAC addresses of endpoints. Next we create a VLAN in the GENI testbed using its Stitching tool. Each of these InPs has a different logic to realize its own layer-two network, but the HyperExchange node nevertheless peers the newly-created networks into a single end-to-end layer-two network. We check the TTL values in packet exchanges to verify that connectivity is at layer 2.

REFERENCES

[1] N. M. K. Chowdhury and R. Boutaba, "Network virtualization: state of the art and research challenges," *IEEE Communications magazine*, vol. 47, no. 7, pp. 20–26, 2009.

[2] J.-M. Kang, H. Bannazadeh, and A. Leon-Garcia, "SAVI testbed: Control and management of converged virtual ICT resources," in *2013 IFIP/IEEE International Symposium on Integrated Network Management (IM 2013)*. IEEE, pp. 664–667.

[3] M. Berman, J. S. Chase, L. Landweber, A. Nakao, M. Ott, D. Raychaudhuri, R. Ricci, and I. Seskar, "GENI: A federated testbed for innovative network experiments," vol. 61, pp. 5–23.

[4] S. Arezoumand, H. Bannazadeh, and A. Leon-Garcia, "Hyperexchange: A protocol-agnostic exchange fabric enabling peering of virtual networks," in *Integrated Network Management (IM), 2017 IFIP/IEEE International Symposium on*. IEEE, 2017.