Virtual Network Overlays Product / RFI Requirements

Version 1.0

A white paper from the ONUG Overlay Working Group

October, 2014

VIRTUAL NETWORK
OVERLAY WORKING GROUP
2014
Open Networking
USER GROUP

Definition of Open Networking

Open networking is a suite of interoperable software and/or hardware that delivers choice and design options to IT business leaders, service and cloud providers. At its core, open networking is the separation or decoupling of specialized network hardware and software - all in an effort to give IT architects options in the way in which they choose to design, provision, and manage their networks. These technologies must be based on industry standards. The standards can be de-facto as adopted by a large consortium of the vendor community, open in the sense that they are community based, or defined as standards by the prevailing standards bodies. Open networking hopes to deliver on two promises:

1) Decoupling of network hardware and software which mitigates vendor lock-in and shifts network architecture structure options to users
2) Significant reduction of the total cost of ownership model, especially operational expense

VIRTUAL NETWORK OVERLAY WORKING GROUP
2014
Open Networking
USER GROUP

Executive Summary

This document defines a common set of functional solution requirements for one of the open networking use cases, virtual overlay networking, identified by the ONUG community. The content of this document is intended as general guidelines for IT enterprise end-users to compare vendor solutions and develop formal RFI specifications, and for IT vendors to develop and align product requirements. Defining a common set of solution requirements aligns with ONUG's goal to drive the IT vendor community to deliver open interoperability networking solutions in order to provide IT end users maximum choice and flexibility in deploying open networking solutions.

The expectation is that this document provides a common baseline, covering the majority of enterprise deployment requirements for network overlay virtualization solutions. The assumption is being made that this set of requirements will be completed by enterprise-specific requirements to meet specific deployment needs.

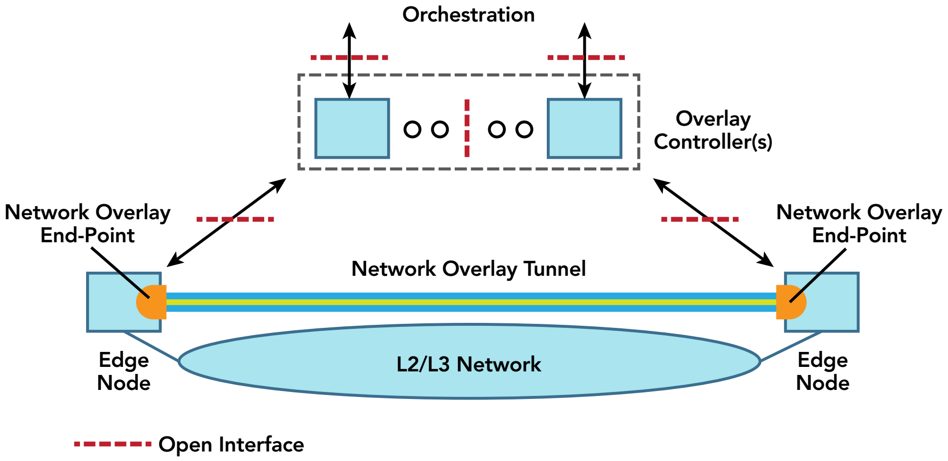
Finally, the expectation is that the scope of requirements defined in this document will evolve. Hence, the versioning of this document.

Virtual Networks/Overlays

Virtual overlays are networking solutions characterized by one or more packet encapsulation techniques in order to forward end-to-end service traffic, independent of underlying transport network technology and architecture. Early successful implementations of this type of architecture were MPLS layer-3 and layer 2 VPNs (Virtual Private Networks), which were introduced in the late 90's. Since then, vendors and industry bodies have introduced various other network overlay solutions, some based on open and some based on proprietary technologies. The scope of network overlay technology solutions in this document is based on data center deployments and IP-based networks. Note that similar network overlay techniques can be applied to other networking segments such as branch, WAN and campus. These scenarios are out of scope for this document at this point. The same applies for network overlays for layer-2 networks; requirements for these types of network deployment scenarios are not being covered at this point.

Figure 1 shows a general framework illustrating the main architecture components of a virtual network overlay architecture solution. Solution requirements are organized along these functional building blocks in the following sections.

Figure 1. Functional Framework for Virtual Network Overlays



Open Networking User Group (ONUG)

ONUG is one of the largest industry user groups in the networking and storage sectors. Its board is made up exclusively of IT business leaders, with representation from Fidelity Investments, FedEx, Bank of America, UBS, Cigna, Pfizer, JPMorgan Chase, Citigroup, Credit Suisse, Gap, Inc., and Symantec. The ONUG mission is to guide and accelerate the adoption of open networking solutions that meet user requirements as defined through use cases, proof of concepts, hackathons, and deployment examples to ensure open networking promises are kept.

The ONUG community is led by IT business leaders and aims to drive industry dialogue to set the technology direction and agenda with vendors. To that end, ONUG hosts two major conferences per year where use cases are defined and members vote to establish a prioritized list of early adopter, open networking projects that communicate propensity to buy and budget development. The vendor community stages proof of concepts based upon ONUG Use Cases, while standards and open source organizations prioritize their initiatives and investments based upon them. ONUG also hosts user summits and smaller, regional user-focused Fireside Chat Meet-Ups through the year.

ONUG defines six architectural areas that will open the networking industry and deliver choice and design options. To enable an open networking ecosystem, a common multivendor approach is necessary for the following six architecture components:

1) Device discovery, provisioning, and asset registration for physical and virtual devices

2) Automated "no hands on keyboards" configuration and change management tools that align DevOps and NetOps

3) A common controller and control protocol for both physical and virtual devices

4) A baseline policy manager that communicates to the common controller for enforcement

5) A mechanism for sharing (communicating or consuming) network state and a unified network state database that collects, at a minimum, MAC and IP address forwarding tables automatically

6) Integrated monitoring of overlays and underlays

In the virtual network overlay framework, three main functional areas can be identified: a data plane, control plane and a management plane.

Network overlay tunnels are part of the data plane and carry end-to-end data traffic between edge nodes, connected to a common underlying layer-2/3 network infrastructure, also referred to as underlay. Edge nodes can be physical devices or virtual networking functionality supported in software. Examples are Top-of-Rack switches (ToRs), network services appliances (e.g., firewalls and load balancers), hypervisors and OS containers. Overlay tunnels are supported via packet encapsulation techniques. At ingress of a tunnel, traffic is encapsulated with a new additional packet header, and this header is removed at the egress end-point of the tunnel.

The control plane is responsible for placement and configuration of overlay tunnel end-points. The assumption is being made that control plane functionality is implemented by one or more overlay controller software components, which are deployed in some cluster and/or distributed fashion for resiliency and scalability purposes. The control plane also provides programmatic access to upper layer orchestration software, responsible for end-to-end provisioning across technology domains, e.g., compute, network and storage.

The management plane consists of all functions needed to configure, monitor and troubleshooting of virtual network overlays, including overlay end-points and end-to-end connectivity.

Figure 1 shows various areas in the network overlay architecture where there is a requirement for open interfaces, both for overlay provisioning and between overlay control plane components. These areas will be addressed in more detail in the following sections of this document.

Problem Statement

A brief description of the problems and challenges being addressed by network virtualization and overlays is provided in the [ONUG use case whitepaper] and were discussed within separate user discussion groups at the recent ONUG May 2014 conference. These problems can be summarized as follows:

1. **Network Segmentation:** Multi-tenant data center networks often have a requirement for separate virtualized workloads and separate connectivity for each tenant to meet security and compliance guidelines. Using conventional centralized network security techniques to meet these workload specific security requirements is challenging and often not feasible.

2. **Workload Mobility:** In order to optimize asset utilization, there is a need to seamlessly move workloads across server clusters and, this way, move workloads to different server PoDs. Certain vendor cluster solutions require layer-2 connectivity between cluster nodes and moving workloads between clusters requires extending layer-2 domains between clusters. Extending layer-2 connectivity using conventional layer-2 VLAN configuration requires multiple network touch points, which is error prone and resource intensive.

3. **Workload Migration:** physical server (rack) upgrades sometimes require migration (temporarily) of workloads to a different server PoD within the same or between data centers. To ensure continued reachability and operation of these workloads, there is a requirement for the workload to keep the same IP addressing (and other network configuration). With today's conventional networking

techniques, this typically involves reconfigurations in the multiple parts of the data center network, which often requires manual configuration, which is error prone and resource intensive.

Virtual network overlays address these problems by decoupling network configuration of virtual workloads from the physical network and support automated API-based networking configuration in server virtualization software. In addition, data traffic from workloads is encapsulated and, this way, logically separated from physical network traffic.

Virtual network overlays enable decoupling of edge networking from physical network underlays and, this way, limit or even eliminate any vendor lock-in. Virtual network overlays can run over any (IP-based) physical underlay network. So, end-users have the ability to buy virtual network overlay and physical network underlay solutions from different vendors, if desired. In addition, virtualizing edge networking enables hiding large number of MAC and IP addresses, consumed by VMs (Virtual Machines) or containers, from the physical edge switches, which simplifies the networking (scale) requirements for these devices. This, in turn, makes it easier for end-users to buy networking switching hardware from any vendor, including the option to use low-cost white box switches.

Requirements

The syntax of a requirement definition in the following sections is defined as follows:

R-10 < Requirement text/definition >

Priority: < High, Medium or Low >

Requirements in this document are numbered using increments of 10. Where needed, related sub-requirements are numbered using increments of 1. In addition, for each requirement, a priority is assigned using the following guidelines:

High: Functionality that must be supported at day one and is critical for baseline deployment.

Medium: Functionality that must be supported, but is not mandatory for initial baseline deployment.

Low: Desired functionality, which should be supported, but can be phased in as part of longer term solution evolution.

Data Plane Requirements

[The network overlay data plane is expected to provide seamless end-to-end connectivity among a heterogeneous set of virtual and physical end-points, which will evolve over time. End-to-end overlay connectivity scenarios include (in order of expected deployment scenario):

Virtual-to-virtual: network overlay connectivity between virtualized host end-point instances, such as hypervisors and Linux containers, referred to Network Virtualization Edge (NVE).

Virtual-to-physical: network overlay connectivity between a NVE and physical network end-point, such as a ToR switch or network service appliance.

Physical-to-physical: network overlay connectivity between physical network end-points, such as ToR switches and/or network service appliances.

Different encapsulation techniques are being supported by vendor network overlay solutions at this time of writing. In general, ONUG's preferred approach is that a single published and agreed-upon encapsulation technique is being used in network overlay solutions by both software and hardware vendors. This way, cross-vendor interoperability can be achieved, as network overlay deployments evolve over time.

General Data Plane Requirements

The following general requirements are defined for network overlay packet forwarding operations:

R-10 Network overlay packet processing and forwarding at a NVE (e.g., hypervisor) should be supported via common open mechanisms defined by [Open Virtual Switch].

Priority: High

Network Overlay Encapsulation

The following requirements are defined for network overlay encapsulation:

R-20 CPU processing overhead, as a result of overlay packet processing, on server host systems should not exceed more than five percent (5%).

Priority: High

R-30 Any specific features required for hardware offload of network overlay packet processing (for meeting the CPU processing overhead requirements) should be documented clearly. For example, use of TSO or VXLAN offload on NICs. Similarly, any requirement for enabling or disabling certain (e.g., Linux) OS kernel modules must be clearly documented.

Priority: High

R-40 Characteristics of CPU processing overhead, as a result of overlay packet processing, on server host systems should be documented and made available to end-users. For example, incremental CPU load as a function of throughput of overlay data traffic.

Priority: Medium

R-50 Network overlay traffic encapsulation, as defined in Section 5 "VXLAN Frame Format" of [draft-mahalingam-dutt-dcops-vxlan], must be supported.

Priority: High

R-60 Network overlay traffic encapsulation, as defined in Section 3.2 "Network virtualization frame format" of [draft-sridharan-virtualization-nvgre], must be supported.

Priority: Medium

In addition to VXLAN and NVGRE, several other proprietary traffic encapsulation types are being used by vendor solutions today. Until these alternative encapsulation types are clearly documented, publicly available and have cross-vendor support, these encapsulation techniques will not be considered and/or promoted by ONUG at this point of time.

ONUG strongly encourages that agreed-upon virtual network overlay encapsulation techniques are being pushed by end-users and vendors through the various standard committees, such that these encapsulation techniques formally get ratified to approved standards.

Network Overlay Termination

The following requirements are defined for network overlay end-point termination:

R-70 The ability for virtual network overlays to terminate on a software hypervisor and have guest VMs connected to a specific virtual network overlay end-point on the hypervisor must be supported.

Priority: High

R-80 The ability for virtual network overlays to terminate on a server host OS with container configured and have container instance connected to a specific virtual network overlay end-point must be supported.

Priority: High

R-90 The ability for virtual network overlays to terminate on a physical network switch or router must be supported. The virtual network overlay end-point should be configurable, similar to virtual/logical ports.

Priority: High

R-100 The ability for virtual network overlays to terminate on a physical network services appliances (e.g., Firewalls, Load Balancers) must be supported. The virtual network overlay end-point should be configurable, similar to virtual/logical ports.

Priority: Medium

Network Overlay End-Point Traffic Policies

The following requirements are defined for policies to steer and manipulate traffic to and from network overlay end-points:

R-110 Virtual (e.g., hypervisor) or physical (e.g., ToR switch) edge nodes must support mapping of layer-2 traffic onto and from virtual network overlays. The later requires the edge nodes to support layer-2 forwarding lookup and overlay header encap/decap operations.

Priority: High

R-120 Virtual (e.g., hypervisor) or physical (e.g., ToR switch) edge nodes must support mapping of layer-3 traffic onto and from virtual network overlays. The later requires the edge nodes, in addition to layer-2, to support layer-3 forwarding lookup and overlay header encap/decap operations.

Priority: High

R-130 In addition to mapping of layer-2 and layer-3 traffic onto network overlays, the ability to do layer-2 switching and layer-3 routing within a network overlay must be supported as well. The assumption is being made that layer-2/3 networking functionality is fully distributed across the network overlay end-points and, this way, avoids hair-pinning of traffic between subnets with routed traffic having to travel to a centralized gateway.

Priority: High

R-140 Virtual or physical edge nodes, supporting network overlay end-point termination, must support configurable QoS mapping(s) as part of traffic overlay header encapsulation operations.

Priority: Medium

R-150 Virtual or physical edge nodes, supporting network overlay end-point termination, must support configurable access control lists for filtering ingress and egress traffic of virtual network.

Priority: Medium

Control Plane Requirements

The network overlay control plane is expected to support placement and provisioning of overlay tunnel end-points and apply ingress/egress traffic policies on these end-points. End-point forwarding state needs to be configured in order to direct traffic onto (on ramp) and from (off ramp) network overlays, which requires communication between the controller(s) and overlay end-points. In addition, exchange of forwarding state between network overlay controllers may be needed when separate network overlay controller functions are deployed for different network domains (e.g., separate data centers). Finally, virtual network overlays provide connectivity between compute, storage and end-users applications. Coordination and placement of network connectivity among these domains typically involves higher layer orchestration functions, which need to interact with the network overlay controller(s).

The various interfaces for the virtual network overlay control plane are depicted in Figure 1 and can be summarized as follows:

Controller-to-End-Point (South-Bound) interface: for configuration of overlay end-points and communication of forwarding state information that needs to be configured at overlay end-points.

Controller-to-Controller (East-West) interface: for exchange of network topology, state and policy information, and network overlay end-point forwarding state information in order to enable end-to-end network overlay connectivity across a single or multiple network domains.

Controller-to-Orchestration (North-Bound) interface: for orchestration systems to communicate high-level application connectivity policy information to network overlay controllers and for network overlay controller to expose logical network topology and policy information to north-bound orchestration systems.

Network overlay solutions available at the time of this writing support either all or a subset of the network control plane interfaces listed above. There are a variety of protocols/mechanisms being used to implement each of these interfaces. Since we are still in the early stages of the network overlay technology adoption cycle, it is not always possible to identify clear winners of specific technology options for network overlay control plane interfaces. Instead, rather than specific technologies, functional requirements are being defined for the various interfaces, which should be used as guidelines for development of technologies and standards related to these interfaces. A general guideline, ONUG's preferred approach is that published, well-defined and agreed-upon (across vendors) protocol mechanisms are being used for implementing the various network overlay control plane interfaces. This way, cross-vendor interoperability can be achieved, providing maximum flexibility for end-users, as network overlay deployments evolve over time.

For completeness, where applicable, technology options currently in use by existing network overlay solutions will be mentioned for the associated requirements sections. The fact that these technology options are being listed does not necessarily mean that these options are being supported by ONUG (that is, is an explicit ONUG requirement) at this point of time.

General Overlay Control Plane Requirements

The following general requirements are defined for a virtual network overlay control plane in a large scale deployment scenario:

R-160 The control plane of a virtual network overlay solution must support state and policy provisioning of at least 10,000 end-points in a given (single) network domain. The assumption is being made that scale is being addressed via either a multi-node cluster or federated control plane architecture.

Priority: High

R-170 The control plane of a virtual network overlay solution must support state and policy provisioning of up to 100,000 end-points in a given (single) network domain. The assumption is being made that scale is being addressed via either a multi-node cluster or federated control plane architecture.

Priority: Medium

R-180 The control plane of a virtual network overlay deployment scenario, defined in R-170 and R-180, must able to recover (re-converge) from failures within an acceptable timeframe. The assumption is being made that the duration of failure recovery will depend on the number of supported end-points. Worst-case scenario is assumed to be when one or controller nodes fail or lose network connectivity and need to completely resynchronize/rebuild. In these cases, the re-converge/rebuild time should not take more than 15 minutes. The assumption is being made that aforementioned-control-plane failures will not result into any data plane forwarding interruptions.

Priority: High

The following general requirements are defined for a virtual network overlay control plane in a small-to-medium scale (e.g., PoD, data center) deployment scenario:

R-190 The control plane of a virtual network overlay solution must support state and policy provisioning of at least 1,000 end-points in a given (single) network domain. The assumption is being made that, if needed, scale is being addressed via either a multi-node cluster or federated control plane architecture.

Priority: High

R-200 The control plane of a virtual network overlay deployment scenario, defined in R-200, must able to recover (re-converge) from failures within an acceptable timeframe. The assumption is being made that the duration of failure recovery will depend on the number of supported end-points. Worst-case scenario is assumed to be when one or controller nodes fail or loose network connectivity and need to completely resynchronized/rebuild. In these cases, the re-converge/rebuild time should not take more than five minutes. The assumption is being made that aforementioned-control-plane failures will not result into any data plane forwarding interruptions.

Priority: High

For both large and small scale deployment scenarios, the following general virtual network overlay control plane requirements are defined:

R-210 The control plane of a virtual network overlay solution must be able to handle connectivity and reachability failures with one or more end-points (e.g., server/host resets or NIC failures). After these types of events, the control plane of a virtual network overlay solution should be able to automatically re-discover and restore the virtual network end-point state and configuration.

Priority: High

R-220 Failure of one or more control plane component in one given domain should not affect overall operation of the system and/or control plane components in other adjacent domains (i.e., faults should be contained in one single overlay control plane domain).

Priority: High

R-230 Failures in one given domain should be signaled to and detected by other adjacent domains and may result in updates of overlay traffic forwarding policies in adjacent control plane domains.

Priority: Medium

R-240 Scale and re-converge results should be clearly documented and be accessible to end-users (e.g., when an NDA is put in place).

Priority: High

Controller-to-End-Point (South-Bound) Interface Requirements

The following functional requirements are defined for the interface between the network overlay controller and NVEs and physical end-points:

R-250 Common mechanisms and bi-directional transport should be used for exchange of configuration and forwarding state associated with network overlay end-points.

Priority: High

R-260 The information syntax and semantics, together with the data transport procedures, for exchange of configuration and forwarding state associated with network overlay end-points should be clearly documented and publicly accessible.

Priority: High

[OVSDB] and several other open and proprietary mechanisms are currently in use by various network overlay solutions on the market today. In addition, new generic mechanisms, such as [OPFLEX], are being proposed for exchange of overlay end-point policy information. None of these mechanisms entirely meets the requirements stated above due to either (1) lack of complete publicly available documentation (i.e., there are undocumented proprietary extensions), and/or (2) lack of common mechanisms for both exchange of state and configuration information of network overlay end-points.

Controller-to-Controller (East-West) Interface Requirements

The following functional requirements are defined for the interface between network overlay controller instances:

R-270 The ability for separate network overlay controllers, each managing a separate non-overlapping network domains with inter-connectivity, to communicate with each other exchange overlay end-point configuration and forwarding state information in order to establish end-to-end virtual overlay connectivity across these separate network domains must be supported.

Priority: High

R-280 The information syntax and semantics, together with the data transport procedures, for exchange of configuration and forwarding state associated with network overlay end-points between network overlay controllers must be clearly documented and publicly accessible.

Priority: High

A select number of networking overlay solutions today support federation of controllers. Some use proprietary mechanisms and some use a combination of [BGP] plus protocol extensions. If these BGP extensions currently in use for overlay controller federation would be formally published and get widely support by the vendor community, this would meet the functional ONUG requirements stated above.

R-290 The exchange of reachability information associated with overlay end-points across network overlay domains between federated network overlay controllers via [BGP] should be supported.

Priority: Medium

Controller-to-Orchestration (North-Bound) Interface Requirements

The following functional requirements are defined for the interface between network overlay controller(s) and upstream cross-domain orchestration systems:

R-300 Logical topology, policy and network state information should be accessible via a RESTful API for upstream applications and orchestration systems. The assumption is being made that a subset of information will be read-only and another subset will be read-write, especially for network policy configuration and control.

Priority: High

R-310 The information elements, made available via a RESTful API for upstream applications and orchestration systems, must be clearly documented, under change control and publicly accessible.

Priority: High

R-320 Provisioning of virtual network topology objects, via the RESTful API for upstream applications and orchestration systems, must support the virtual network overlay control plane scale numbers defined by R-150, R-160 and R-180.

Priority: High

R-330 The overlay controller API must support a robust Role Based Access Control (RBAC) mechanism for authentication and access control of API requests.

Priority: High

R-340 Support for RBAC-based change management and audit information provisioning request via the north-bound API must be supported. Information and logs on type and who made changes, and possible validation of certain conditions after changes, must be supported.

Priority: High

The OpenStack [Neutron API] specification meets the functional ONUG requirements stated above and could be considered as a baseline reference for a north-bound API for network overlay controllers.

Management Plane Requirements

The following functional requirements are defined for the management of network overlay end-points on edge nodes:

R-350 Interface traffic statistics and configuration parameters, defined in [IF-MIB], should be supported for virtual network end-points.

Priority: High

R-360 Administrative and status Up/Down events, as defined in [IF-MIB], should be supported for virtual network end-points.

Priority: High

R-370 Events related to state (Up/Down) and status of a session between a virtual network overlay end-point and associated controller(s) must be supported.

Priority: High

R-380 SNMP access to virtual network end-point interface traffic statistics and configuration parameters and events must be supported.

Priority: High

R-390 API-based access to virtual network end-point interface traffic statistics and configuration parameters and events must be supported

Priority: High

R-400 Syslog access to events related to virtual network end-point and session between virtual network end-points and controllers should be supported.

Priority: Medium

R-410 SFlow, IPFIX or other flow monitoring/sampling technologies to collect ingress and egress NVE traffic statistics must be supported.

Priority: High

VIRTUAL NETWORK OVERLAY WORKING GROUP 2014
Open Networking USER GROUP

End-to-end Network Overlay Monitoring

The following functional requirements are defined for end-to-end virtual network management:

R-420 Automated reachability validation, leveraging heartbeat messages, between any given set of end-points within a given virtual network overlay must be supported. The assumption is being made that the frequency of heartbeat message and the threshold number of unsuccessful heartbeats before reporting a reachability failure event is configurable.

Priority: Medium

R-430 Traffic flow monitoring between any given set of end-points within a given virtual network overlay must be supported. The assumption is being made that the sampling frequency is configurable.

Priority: Medium

Network Overlay-Underlay Monitoring

The following functional requirements are defined for correlating physical underlay network state and events with corresponding network overlays:

R-440 The ability to correlate status and failure events in physical network underlays with network overlays must be supported.

Priority: High

Recommendations for Standards

For completeness, the specific areas in virtual network overlays solutions where there is a need for open well-defined and vendor-agreed-on technology standards can be summarized as follows:

- **Virtual network controller-to-end-point interface:** open south-bound interface for configuration of overlay end-points and communication of forwarding state information that needs to be configured at overlay end-points.
- **Virtual network controller-to-controller interface:** open east-west interface for exchange of network topology, state and policy information, and network overlay end-point forwarding state information in order to enable end-to-end network overlay connectivity across a single or multiple network domains.
- **Virtual network controller-to-orchestration interface:** open north-bound interface for orchestration systems to communicate high-level application connectivity policy information to network overlay controllers, and for network overlay controller to expose logical network topology and policy information to north-bound orchestration systems.

References

[ONUG web site] - ONUG white paper, “Open Networking Challenges and Opportunities.”

[Open Virtual Switch] – Apache project Open vSwitch.

[draft-mahalingam-dutt-dcops-vxlan] - IETF Informational RFC, “VXLAN: A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks.”

[draft-sridharan-virtualization-nvgre] - IETF Informational RFC, “NVGRE: Network Virtualization using Generic Routing Encapsulation.”

[OVSDB] - IETF Informational RFC, “The Open vSwitch Database Management Protocol.”

[draft-smith-opflex] - IETF Informational RFC, “OpFlex Control Protocol.”

[draft-ietf-l2vpn-evpn-07] - IETF Draft RFC, “BGP MPLS Based Ethernet VPN.”

[Networking API v2.0 (CURRENT)] - OpenStack Networking API specification.

[RFC2233] - IETF Draft Standard RFC, “The Interfaces Group MIB.”

ONUG Overlay Working Group

Harmen Van der Linde, Co-Chairman	citi	Neal Secher	Morgan Stanley
Carlos Matos, Co-Chairman	Fidelity INVESTMENTS	Srini Seetharaman	T Deutsche Telekom
Mike Cohen	CISCO	Dimitri Stiliadis	nuagenetworks
Kyle Forster	big switch networks	Francois Tallet	vmware
Piyush Gupta	JPMorgan Chase & Co.	Sunay Tripathi	PLURIBUS NETWORKS
Terry McGibbon	BARCLAYS	Jaiwant Virk	DELL
Thomas Nadeau	BROCADE	Sean Wang	UBC THE UNIVERSITY OF BRITISH COLUMBIA

VIRTUAL NETWORK
OVERLAY WORKING GROUP
2014
Open Networking
USER GROUP