Network Virtualization in Open flow Networks- A Review

Gowri Prasad, Swapna B Sasi

Abstract— This review paper presents a study about network virtualization in open flow based networks. It describes in detail about open flow networks and presents a detailed study of some of the existing approaches to network virtualization in open flow networks after investigating data path virtualization and control channel virtualisation. Different methods for resource allocation and separation in virtual networks are also covered as part of this paper.

Index Terms— Virtualization, Open Flow, Flowvisor,Network Hypervisor,ADVisor,Datapath Virtualization,Control Channel Virtualization

1 INTRODUCTION

Virtualization means creating virtual versions of some physically existing systems. Virtualization is employed in modern system design to decouple the system service model from its physical realization. Some examples of virtualization are virtualization of computing resources through the use of virtual machines and the virtualization of disks by presenting logical volumes as the storage interface. The insertion of these abstraction layers allows for achieving operational goals separated from the underlying physical infrastructure. Today, workloads can be migrated between physical servers and suspended if needed.

Network Virtualization virtualizes a network. Once network is virtualized successfully multiple customers and competitors can share a single physical network. Many techniques are employed to virtualize the network to improve network resource utilization through sharing, separation of traffic between different entities, and to simplify network management. While employing various techniques to virtualize a network the system should make sure that the traffic is separated between customers and no information should be able to leak between the virtual partitions. It should also enforce any established SLA`s and should be as flexible as possible to avoid costly interaction and coordination between the parties involved, reducing operational costs.

To better understand virtual networking, we have to look at computer virtualization. Computer virtualization's success is because of the presence of an abstraction layer i.e. the underlying hardware. This hardware abstraction permits slicing and sharing of resources among the operating systems. Thus, each OS believes that it has its own private hardware.Different hardware can be used below the abstraction layer so long as it can be mapped to the hardware abstraction layer. Thus different hardware can have instruction sets of their own.This leads to a more efficient system. Operating systems can be implemented for networks as well eg. NOX[7].Once the network is virtualized we should provide mechanisms for their resource allocation also since virtualization is mainly accomplished using sharing of resources. Mechanisms for resource allocation should adopt fair policy. That means different virtual networks have to get a fair share of resources of the network. Competition for resources occurs only at points in network called stress points where there is heavy traffic.

.

2 OPEN FLOW

Open flow[2] is based on an Ethernet switch with a flow table and interface to add and remove flow entries. Open flow switch provides an open protocol to program flow tables present in switches. Production traffic and Research traffic is partitioned. So research people can try their new routing protocols or security models without affecting production traffic. The datapath for an open flow switch consist of a flow table and an action associated with each flow table entry. All open flow switch supports a basic set of actions and is extensible. Open flow switch mainly consists of three parts:

1.) Flow Table with an action associated with each flow table entry.
2.) Secure Channel which connects switch to a controller.
3.) Open flow Protocol which is the standard way for communication between controller and switch.

Open flow helps in encouraging more research in the field of networking. It helps in running experiments in different heterogenous networks without writing specific software for each network.

.

- *Gowri Prasad is currently pursuing masters degree program in Computer Science andEngineering in Jyothi Engineering College Thrissur, India, PH-04872333629.*
  E-mail:gowri.0588@gmail.com
- *Swapna B Sasi is Assistant Professor in Computer Science And Engineering Jyothi Engineering College,Thrissur,India,PH-09447125373*
  E-mail: swapna@jecc.ac.in

3 NETWORK HYPERVISOR

Network Hypervisor[6] is a network wide software layer-which maps logical forwarding plane to the underlying physical hardware.Network Hypervisor is a network virtualization layer which virtualizes the forwarding plane of the network. The main components of a network includes control plane, Network Hypervisor, Logical forwarding plane and physical forwarding plane.

Logical forwarding plane consists of look up tables and ports. Look up tables consists of forwarding tables usually built around a pipeline of TCAMS with forwarding actions.. Ports can be logical ports which is bound to physical ports. In order to forward a packet each packet should pass through the following steps.

1) Map incoming packet to the correct logical context . It can be done by providing some identifying tag such as MPLS header.
2) Make a logical forwarding decision.
3) Map the logical forwarding decision back to physical plane.At the end of a logical forwarding it reaches one or more egress port on the logical network.
4) Physical Forwarding.

Network Hypervisor maps the logical forwarding plane to the underlying hardware by maintaining a global view of all physical resources in the network and all logical forwarding planes. It is built as a distributed system and it operates as an open flow controller. It maintains a network graph by connecting with every switch in the network. It provides an API which creates and maps logical forwarding elements to its physical network.

4 FLOWVISOR

Flowvisor[4] is a technique for control side virtualization. It is implemented as a protocol proxy that intercepts messages between open flow enabled switches and open flow controllers. Flowvisor layer is present between underlying physical hardware and software that controls it. It hosts multiple open flow controllers, one controller per slice to control that particular slice assigned to it.

A slice may be defined as a set of flows through a set of switches. Flowvisor makes sure that each controller can control only the switches it is supposed to control. It partitions the flow table such that it can differentiate the flow entries of different guest controllers. A slice of a network can be considered as a subspace of the entire geometric space of all the packet headers. In open flow each of these flow entries will be made based on a 10 field packet header which is 256 bits long. So there will be 2 pow 256 points in a 256 dimensional space. We can define various regions as a subset of space using bit masks if we define a header using 256-k bits(k denotes bit mask) then it has a k dimensional region. A slice can be defined by a set of regions, where each region is a subset of entire geometric space and is composed of a set of points. This set of regions can be considered as a slice`s flowspace.Flowvisor acts as a transparent proxy between guest controllers and switch thus ensuring transparency and isolation between slices by the process of inspecting,rewriting and policing open flow messages as they pass.Flowvisor can control multiple switches and can virtualize another virtual network.

5 ADVISOR

Advanced Flowvisor[3] is similar to flowvisor except that in ADVisor virtual topologies for each virtual networks are no restricted by the underlying physical topology. It is present between the physical network and the controllers. It can slice a virtual topology repeatedly and can directly reply to open flow network. Various components like virtual nodes are represented as a set of tuples and using these tuples Virtual topologies are identified.[5]

Main parts of an ADVisor includes:

1.) Topology Monitor which identifies whether the switch generating the open flow protocol message is at the endpoint of a link or a part of physical link.
2.) Link Broker which controls the switches which are present as part of a virtual link. Packets sent by these switches are controlled.
3.) Port mapper which edits the action field.
4.) Flowvisor which slices the nework.

6 NETWORK VIRTUALIZATION

As discussed above network virtualization can be accomplished through either datapath virtualization[1] or through control channel virtualization[1]. Whatever may be the case we have to ensure proper separation between different virtual networks and fair resource allocation. In case of datapath virtualization separation should be ensured at link level and for flow tables. A link level separation can be done using two approaches viz partitioning and encapsulation. In partitioning the link is separated by splitting it into multiple partitions by slicing the total flow space, assigning one partition per virtual network. In encapsulation each Virtual network is assigned a link local encapsulation ID, called the virtual network ID to which the VN traffic will be mapped. In case of flow table separation can be done using two approaches viz flowspace partitioning and table partitioning. A specific flow space is assigned to each VN. Each VN can enter only those flow table entries which match the specific flow space assigned to that particular VN. In table partitioning the flow table is divided into multiple tables logically or physically and each VN is assigned a group of distinct flow tables. To ensure fair resource allocation at link level we can use the classic QoS tools like classification, metering, coloring, policing etc.

We have to ensure fair allocation of resources on the control channel i.e. the network that connects controller to the switches. If the control channels for different virtual networks are multiplexed over a single tcp connection it is impossible to differentiate control channels and to enforce QoS .If switches allow different controllers to connect and control the VN`s using different source or destination IP addresses or port numbers, the control traffic can easily distinguish between different connections and thus enforce QoS policies.

Control channel may be implemented as an out of band or in band control channel. In an out of band control channel the control signals and data traffic pass through different channels. While in case of an in band control channel the data and control traffic pass through the same channel. Out of band channels are simpler and easier to design in a reliable fashion. But it is more expensive due to an entire extra network and extra ports on hardware. In an In band control channel the channel is vulnerable to misconfigured flow table entries since all packets, including control packets traverse them .If sufficient QoS support is provided to control traffic entering. the switch and outgoing traffic , In band channel will be more advantageous to implement.

7 CONCLUSION

Virtualizing a network helps in regulating the competition for resources, improving resource utilisation, implementing a checkpoint for network changes and also helps in experimenting new network protocols. We have reviewed different virtualization systems and virtualization models. We also saw how to enforce isolation and separation between different virtual network in both datapath and control channels. As part of future work we can look forward in implementing another efficient virtualization system.

REFERENCES

[1] Pontus Skoldstorm and Kiran Yedavalli "Network Virtualization and Resource allocation in open flow based wide area networks."

[2] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson,J. Rexford, S. Shenker, and J. Turner, "OpenFlow: enabling innovation in campus networks," ACM SIGCOMM Computer Communication Review,vol. 38, no. 2, pp. 69–74, 2008.

[3] M. Gerola, "Enabling Network Virtualization in OpenFlow Networksthrough Virtual Topologies Generalization."

[4] R. Sherwood, G. Gibb, K. Yap, G. Appenzeller, M. Casado, N. McKeown,and G. Parulkar, "Flowvisor: A network virtualization layer,"OpenFlow Switch Consortium, Tech. Rep, 2009.

[5] E. Salvadori, R. Corin, M. Gerola, A. Broglio, and F. De Pellegrini,"Demonstrating generalized virtual topologies in an openflow network,"in Proceedings of the ACM SIGCOMM 2011 conference on SIGCOMM.ACM, 2011, pp. 458–459.

[6] M. Casado, T. Koponen, R. Ramanathan, and S. Shenker, "Virtualizingthe network forwarding plane," in Proceedings of the Workshop onProgrammable Routers for Extensible Services of Tomorrow. ACM,2010, p. 8.

[7] Natasha Gude,Teemu Koponen, Justin Pettit, Ben Pfaff, Marin Casado, Nick McKeown, Scott Shenker NOX:Towards an Operating System for Networks.