ENDEAVOUR: Towards a flexible software-defined network ecosystem

ENDEAVOUR

Project name	ENDEAVOUR
Project ID	H2020-ICT-2014-1 Project No. 644960
Working Package Number	4
Deliverable Number	4.3
Document title	Design of Use Cases for Members of IXPs
Document version	1.5
Editor in Chief	Dietzel, DE-CIX
Authors	Dietzel, Bleidner, Kathareios, Chiesa, Castro, Abdellatif, Antichi, Bruyere, Fernandes, Owezarski
Date	28/01/2016
Reviewer	CNRS
Date of Review	20/01/2016
Status	*Public*

Revision History

Date	Version	Description	Author
08/12/15	0.1	Initial Version	Dietzel, Bleidner
17/12/15	0.2	Contribution Novel Services	Kathareios
22/12/15	0.3	Contribution Novel Services	Owezarski
23/12/15	0.4	First Draft	Dietzel, Bleidner
23/12/15	0.5	Virtualized Private Peering	Abdellatif
02/01/16	0.6	Contribution TE	Chiesa, Canini
05/01/16	0.7	General Improvements	Dietzel, Bleidner
05/01/16	0.8	Contribution Traffic Steering	Antichi
08/01/16	0.9	Virtual Peering Router	Fernandes,Bruyere
12/01/16	1.0	Revised all contributions	Dietzel, Bleidner
13/01/16	1.1	Added Intros and Summary	Dietzel, Bleidner
14/01/16	1.2	Added Exec. Summary	Dietzel, Bleidner
19/01/16	1.3	Revised full deliverable	Bleidner
22/01/16	1.4	Full Review	Owezarski,Bruyere
26/01/16	1.5	Final Changes	Dietzel, Bleidner

Executive Summary

IXPs are convergence points for inter-domain routing, making them an integral part of the rich Internet ecosystem. They are interconnecting a multitude of different network types and easing the setup of peering relations. ENDEAVOUR strives to transform this ecosystem with innovative and disruptive ideas through the introduction of the SDN technology at IXPs.

In this deliverable we demonstrate use cases which address the current limitations IXP members face today. Furthermore, we present solutions based on the programmability and flexibility SDN brings to the networking space. Based on the insights we obtained during several workshops, in discussion with members, or due to our experience as IXP operators we identify four areas where SDN can have significant impact on transforming operational tasks and new business opportunities.

We introduce use cases that allow members in a straight-forward manner to optimize the utilization of their IXP ports. Either by leveraging traffic engineering techniques or by sophisticated bandwidth management. This may also reduce the complexity members have to face as of today.

Currently, some protocols are banned legally through policies for all members at the peering platform. However, due to limited hardware capabilities of commodity switches this is not enforced. Also the omnipresent threat of DDoS attacks can be tackled more precisely and thus reduce the collateral damage.

IXPs' evolved from a relatively simple location to exchange layer 2 data frames to a more complex peering facility right in the core of the Internet ecosystem. They offer a multitude of value-added services, e.g., route servers, blackholing, or private peering VLANs. ENDEAVOUR proposes several novel services utilizing the feature set SDN has to offer. Most of these services are build on other use cases as primitives. We expect the suggested services to enrich the IXP environment further and contribute to a higher diversity in the inter-domain routing space.

ENDEAVOUR will evaluate the potential impact of the use cases provided in this deliverable. Based on this, we will make a selection of the most promising use cases to be considered for being implemented on top of the ENDEAVOUR architecture. Thus, a selection of use cases from this deliverable as well as from deliverable 4.2 will be implemented for demonstration purposes. This will allow ENDEAVOUR to show the practical impacts and relevance of SDN for both IXP operators and IXP members.

Contents

List of Figures and Tables

1 Introduction

While the Internet continues to evolve, today's applications constraint increasingly high demands in bandwidth, latency, and availability. Driven by such requirements two interesting aspects of the Internet ecosystem came into the focus of the research community in the past years, i.e., Software Defined Networking (SDN) and Internet eXchange Points (IXPs).

SDN is emphasized as the final break through for more programmable computer networks. To offer higher programmability the control plane and the data plane are separated. A logical centric entity controls multiple data plane devices inside a network. The OpenFlow protocol [49] is the most prevalent implementation of this concept. However, the practical impact falls behind the opportunities envisioned by academia. Most deployments of SDN technology occur in closed and controlled environments, e.g., data centers [69] or intra-domain routing [44]. We believe that SDN drives innovation and particularly for deployments beyond closed systems. Indeed, especially for dense inter-domain routing hotspots.

Presently, hundreds of IXPs allow thousands of ASes to peer with each other [31]. The largest among them carry about five Tbps and count over 600 member networks with a sustainable growth for the next years. Most IXPs operate route servers [64] to foster as much open peering relations as possible. However, BGP-based routing solely focuses on reachability and allows only a very myopic view of the data plane [12]. This constrains the ability of networks to route their traffic in a more effective manner and limits innovation potential for novel services.

Combining SDN as a powerful new technology with the rich inter-domain routing ecosystem at IXPs culminates in a hotbed of innovation. First, enhanced programmability even at a single IXP enables up to hundreds of Autonomous Systems (ASes) to innovate their peering strategies. Second, deploying SDN at IXPs is strategically sound because the network setups of IXPs itself are quite static which even scales with current SDN-capable switches.

ENDEAVOUR strives to impact the peering ecosystem at large by bringing SDN with practical use cases to IXPs. Fueled by numerous discussions, with input from workshops, a podium discussion and related work we present where exactly we expect SDN at IXPs to be beneficial. This deliverable reflects the current state of ENDEAVOUR use cases and its potential benefits for members of IXPs.

Section	Use Case Name	Category	Page
3.1	Inbound TE	Traffic Engineering	8
3.2	Outbound TE	Traffic Engineering	11
4.1	Control / Data Plane Consistency	Filtering	14
4.2	Advanced Blackholing	Filtering	15
5.1	Virtualized Private Peering	Bandwidth Mgmt.	17
5.2	Control Plane Traffic Protection	Bandwidth Mgmt.	19
6.1	Destination Port Congestion Awareness	Novel Services	21
6.2	IXP as Transport Marketplace	Novel Services	23
6.3	Service Chaining	Novel Services	24
6.4	Autonomous Anomaly Detection	Novel Services	25
6.5	Virtual Peering Router	Novel Services	29
6.6	Multi-Cloud and IXP as Cloud Broker	Novel Services	32
6.7	Losslessness as a Premium Service	Novel Services	34
7	Member Driven Monitoring	Monitoring	35

Table 1: Overview of use cases for members.

2 Outline

In this section we briefly introduce the structure of this deliverable. Each use case for IXP members outlined in this document is grouped into the following three paragraphs: *i*) First we provide an overview of the current situation and discuss its limitations. *ii*) We highlight the already available solutions and explain why they may not be sufficient. *iii*) Finally, we aim to sketch how a technical solution with SDN could be implemented and provide a brief description of the features we want to take advantage of.

Moreover, we present a comprehensive list of all use cases in Table 1. It lists the section in this document where the use case can be found, the name, its category, and on which page the description starts.

3 Traffic Engineering

The IXP environment is an arena of complex interactions among heterogeneous, possibly contrasting, non-coordinated economic entities. In this context, which encompasses both technical and economic factors, even a relatively simple task, like *Traffic-Engineering* (TE) operations, i.e., the steering of flows of data traffic along the best routing paths, becomes an unfeasible feat. In fact, by lacking a centralized control, a set of suitable routing tools, and a view of the global network state, Internet-wide TE is a notoriously difficult operation. This is in contrast to closed environments (e.g., data center networks), in which everything is controlled by a single administrative entity. As a result, while data center networks thrive from the recent advances in networking (e.g., SDN), inter-domain TE is still performed using the same mechanisms that were available more than a decade ago.

In this section, we explore and highlight the benefits that an SDN-enabled IXP can bring to the inter-domain ecosystem for both *inbound* and *outbound* TE, that is, how traffic enters and leaves an IXP member network, respectively. From the research perspective, inter-domain advanced fine-grained peering applications are begging for research. We believe that SDN has a great potential for fostering an enormous amount of novel ideas in the inter-domain routing areas. While SDN so far lacked a strong use case scenario, the recent growth of IXPs in the Internet has finally brought an ideal place for spurring inter-domain innovation. IXPs are thriving rich peering environments routing terabits of traffic per seconds. Its members continuously strive towards a better user experience by tweaking Border Gateway Protocol (BGP) configurations via indirect and limited mechanisms (e.g., as-path prepending, multi-exit-discriminator). The network topology is fairly static and any addition of a new device into the network has to coordinated with the IXP administrators.

From the operational perspective, we aim to cast our techniques into the use cases that we set forth in the deliverable 2.1 and 4.1.

3.1 Inbound TE

Current Situation

To support the increasing amount of high-volume traffic being exchanged, many IXP members need to connect to the IXP with multiple physical ports. These IXP members aim to achieve high port bandwidth utilization, that is, spreading the traffic that enters their networks (i.e., inbound traffic) through their multiple physical connections. Unfortunately, nowadays this operation is not an easy task mostly because of outdated inter-domain routing tools and a lack of an adequate monitoring infrastructure. BGP is the prevalent inter-domain routing protocol in today's Internet. The routing decision is based on the destination Internet Protocol (IP) address. Thus, ASes have limited control over how the traffic enters their networks. Mechanisms such as path prepending [58, 15], communities and selective announcements [59],

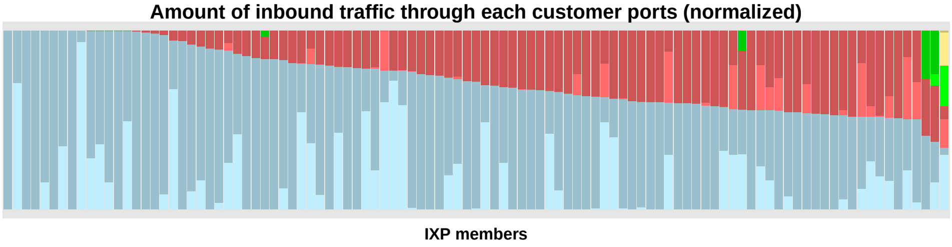

Figure 2: Inbound traffic load-balancing in a large IXP. Rach IXP member C with at least two physical ports connected to the IXP is represented by a vertical bar V. Each port of C is represented by a sub-bar of V coloured with a pair of light/dark color. Light colors are used for traffic that is received because of IP prefixes announced to the route server, while dark colors are used for the rest of the traffic. The size of the coloured sub-bar is proportional to the amount of inbound traffic that is routed through that peering port for a specific customer.

originally not part of the BGP routing protocol, have been widely adopted to fill this gap. Additionally, ASes originating traffic may have their own policies in place for outbound traffic [72], limiting the ability to control traffic based on inbound traffic engineering [58]. At very dense traffic convergence points such as IXPs, a wide range of independent and inconsistent peering policies clash [66].

Despite these limitations, by analyzing real-traces of data traffic at one of the largest IXP (see Figure 2), it can be observed that inbound TE is a widely performed operation by IXP member network operators. For instance, the right-most bar in the graph represents a customer with four physical ports. The blue sub-bar (including both light and dark blue sub-bars) is associated with one of port that receives 40% of the traffic directed towards that customer. Since the light blue part of the sub-bar is large, most of this traffic is received because of IP prefixes announced directly to the route server.

Available Solutions

Current Inbound TE solutions at IXPs involve two parties : the IXP member that sends traffic and the IXP member that receives the traffic. Each of these two players can potentially influence how traffic enters the receiver network. As for the sender operational side, for example, it is known that some of the largest Content Delivery Networks (CDNs) that are connected to IXPs, continuously probe the quality of the communication through (some of) the IXP member ports and split their outgoing traffic based on these measurements. In this way, CDNs are performing inbound TE on behalf of the receiver of the traffic. As for the receiver operational side, current inter-domain routing protocols allow network operators to select a single IXP ingress port for each IP prefix. The operator is responsible for mapping each of its announced IP prefixes to one of its ingress port in order to steer inbound traffic. There are several drawbacks of this approach. The operator needs to measure the amount of traffic at the per-prefix granularity and then it has to carefully announce its IP prefixes from its BGP peerings in order to spread the incoming traffic among its ports. These operations are hindered by the lack of a global measurement infrastructure at the single IP prefix level.

Technical Description

There are three main advantages of SDN for TE purposes. First, it has a visibility of more information than any other IXP member about both control and data plane state. Second, it allows network operators to steer flow of traffic at a more fine-grained level of granularity than BGP. Some recent work showed how SDN can be used to steer inbound flow of traffic by means of Network Address Translation (NAT) mechanism [71]. Third, by providing a coherent view of the state of the network, SDN open doors for network operators to access a monitoring platform that supports TE applications. Monitoring operations can be outsourced to the IXP, who is then responsible for spreading inbound traffic among each members ports or communicating that information to (some of) the members. For example, the IXP can support the sender of traffic with a measurement of the IXP members port utilization. This solutions centralized monitors operation within the IXP, which are then available to all the IXP members.

If load balancing needs to be performed on the receiver side, SDN can be leveraged both in a static or dynamic way. The former consists of leveraging weighted hash-based per-flow load balancing mechanisms within the IXP network for balancing traffic among different ports, which has been standardized in OpenFlow 1.3. By being oblivious to the specific flow of traffic that are routed, this approach brings several benefits to inbound TE: it is static, it is more robust to per-prefix inter-domain routing changes and routing attacks. The inbound splitting ratios of each IXP member can be defined by the member itself via a proper configuration language. Unfortunately, weighted load balancing is not yet widely available on today's SDN switches. The latter operation (i.e., dynamic load balancing) requires to compute the traffic matrix within the IXP at suitable level of granularity, to estimate the IXP member ports utilization, and to balance the flows of traffic among the member ports by exploiting SDN fine-grained forwarding capabilities.

To summarize, an IXP represents the key place for centralizing these monitoring operations that would be beneficial for the whole set of IXP members. To be at the forefront of innovation, an IXP should exploit its superior vantage point at the intersection of the Internet traffic in order to improve cohesion of network monitoring functionalities by means of SDN support.

3.2 Outbound TE

Current Situation

Outbound TE, i.e., deciding where to send data traffic, is an extremely relevant operation for supporting high QoS-demanding services (e.g., Internet video broadcasting). Current mechanisms are limited by BGP, which is a per-destination routing protocol that works at the granularity of IP prefixes. Typical outbound TE tools encompass "BGP local-preference", which allows to explicitly prefer routes based (for instance) on the traversed ASes, and setting Interior Gateway Protocol (IGP) internal weights such that outgoing traffic flows are routed through their closest egress points, which is an indirect way of steering flows of traffic [51, 79]. Such mechanisms limit network operators ability to perform TE in a direct and more intuitive way. Instead of tweaking link weights, operators want to define routing policies in a more intuitive and direct way. In certain cases, operators also need to steer traffic flows at a finer granularity that the one of IP prefixes. On top of these limitations, the current IXP interconnection model poses even more challenges. Larger IXPs run a route server service which can be used to exchange control plane information among all the IXP members without any need to configure a BGP peering session with every single member. Members of the IXP need to peer with the route server, which will process all the BGP route announcements from its peerings and propagate the best routes (according to its perspective) towards its peerings. The main limitation of this mechanism is the lack of route visibility from the IXP members view-point. For instance, if a source of traffic wants to load balancing its outgoing traffic towards two ports of a single receiver IXP member, this operation cannot be performed using the route server as the only control plane IP reachability source of information since the route server will only send the best route, which identifies a single physical port, to the source of traffic. For this reason, many IXP members avoid using the route server, ending up in configuring a multitude of BGP peerings with most of the IXP members. This is particularly evident in Figure 2, where it can be seen that most of the bars are coloured with a dark tonality, which is associated to traffic that is not exchanged because of a peering with the route server.

As a side observation, the reader should observe that, as described in the inbound TE section, outbound TE sometimes reduces to inbound TE. This situation arises in those scenarios where the source of the traffic is limited in sending traffic to a unique IXP members with multiple ports. In that case, load balancing traffic among these ports can be both performed by the sender or the receiver of the traffic or by the IXP itself. Conversely, outbound TE includes scenarios in which a source of traffic estimates (e.g., by means of probing techniques) the quality of the paths through the different IXP members and through networks external to the IXP in order to determine the best outgoing communication paths. In that case, the routing decision is moved exclusively to the sender of traffic. The main limitation of current techniques is similar to the one already explained in the inbound TE section: It requires each source of traffic to build its own measurements infrastructure, without taking advantage of the fact that the IXP broad visibility of the state of the network.

Available Solutions

A plethora of academic efforts has been devoted to the intricate problem of tweaking inter- and intra-domain routing protocols in order to achieve certain performance goals [15, 34, 35, 36, 58, 59, 72]. In addition, network vendors provides simple heuristic to specify network configuration, which are however often oblivious to the traffic patterns and routing policies [20]. The most widely used tool for choosing the best outgoing route is the BGP local-preference. Unfortunately, the nature of BGP constrains operators to define their outbound policies on per-IP-destination basis, which is in certain cases an over-restricting limitation.

As for the lack of route visibility for those IXP members that peers with the route server, the networking community has repeatedly advocated for improvements in the BGP route propagation mechanism, i.e. the BGP-Add-Paths standard [62]. Unfortunately, most of the industrial route server services still do not implement this feature, leaving a huge disincentive for using the route server service.

Technical Description In the context of outbound TE, the benefits of equipping an IXP with SDN capabilities are twofold.

1. Fine-Grained Routing Policy Programmability. An SDN-enabled IXP provides each member AS with the abstraction of a dedicated switch that it can program using match-action policies to control traffic flows. Members may express SDN policies (i.e., fine-grained policies) on both their inbound and outbound traffic; the IXP controller ensures that no SDN policy results in traffic being forwarded to a neighboring AS that did not advertise a BGP route for the prefix that matches the packet's destination IP address. Each participant runs an SDN control application on the IXP controller and has its border router exchange BGP update messages with the IXP's route server. The SDN controller combines the SDN policies from all participants, reconciles the resulting policy with the BGP routing information, and computes and installs the resulting forwarding table entries in the IXP fabric. More details about the scalability issues that arise in this context are described in deliverable 2.2. The SDN controller does not suffer from the lack of visibility problem. Since it receives as input all the outbound policies and all the BGP routes of each IXP member, the best route is chosen according to the preference function of the IXP member.

2. Monitoring services for outbound TE. By being at the intersection of hundreds of ASes, an SDN-enable IXP is in a sweet spot for modularizing network monitoring operations and offering to its members an intuitive interface to access and control the measurements being performed. Such measurements can then be used as the input of any traffic engineering mechanism, especially outbound TE, in which the source of the traffic needs to estimate the state of the routing paths in order to better balance the outgoing flows.

4 Filtering

On account of the high bandwidth IXP members are connected the filtering capabilities are to somewhat limited. Especially because commodity switching hardware can not filter on arbitrary packet header fields on a sufficient large scale.

Service-Level Agreement (SLA) and other contractual constraints can not be enforced. Even more critical is the effective mitigation of Denial of Service (DoS) attacks. Defense techniques would benefit largely from fine-grained filtering capabilities.

Thus, this Section outlines use cases that address these shortcomings.

4.1 Control / Data Plane Consistency

Current Situation

Given the rich peering ecosystem at IXPs different network types (e.g., CDNs, Internet Service Providers (ISPs), or eyeball providers) have contrasting routing policies. CDNs are more likely to have an open peering policy. Thus, they establish peerings with any other member (multi and bi-lateral). In contrast, large ISPs prefer to peer with ASes about the same size. Otherwise, the larger party risks to lose a potential customer to whom they could sell transit.

ASes express their routing policies through BGP at IXPs. Hence, they limit their advertised prefixes towards other members if they rely on a strict peering policy.

Due to a multitude of reasons they are not able to verify or even filter if the received traffic on the data plane is consistent with the control plane, i.e., peering policy. Particularly, because mal-intended IXP members that know the Message Authentication Code (MAC) address of an ISP can configure a static BGP route and send traffic for not exported routes towards it. For instance, if the IXP is a Tier 1 provider it is very likely that the traffic is delivered to any destination. When IXP members apply this intended misconfiguration they can get transit without compensation at IXPs.

Available Solutions

Currently, the IXPs lacks insights into who is exporting which routes to whom. Hence, IXPs are not able to filter any traffic. However, we have anecdotal evidence from a large European ISP that reported they are not able to filter their ingress traffic themselves. To the best of our knowledge there is no solution available yet. Technical one could maintain filters based on source MACs. Yet, these lists need to be kept up to date. Depending on the size of an AS or the IXP this would pose significant management overhead.

Technical Description

SDN supports matching on the source MAC address. All member's MACs that do not peer with a given other member can be blacklisted. Thus, their packets would be dropped through an OpenFlow drop rule.

However, this assumes that all members disclose their peering policies to the IXP. Since this tend to be business critical information the IXP needs to assure a safe and sound processing.

4.2 Advanced Blackholing

Current Situation

Distributed Denial of Service (DDoS) attacks are and continue to be a serious threat to the Internet. Indeed, the intensity and the dimension of such attacks is still rising in particular due to amplification and reflection attacks [65, 26]. DDoS attacks impact not only edge networks but can also overwhelm cloud services [70] or congest backbone peering links at IXPs [57].

Available Solutions

Thus, IXPs have deployed blackholing as a service for their members [27]. Blackholing is an operational technique that allows a peer to announce a prefix via BGP to another peer, which then discards traffic destined for this prefix.

The IXP handles this IP address and resolves it by means of the Address Resolution Protocol (ARP) into a predefined blackholing MAC address. All Ethernet frames with this destination MAC are discarded via Access Control List (ACL) at the IXP layer 2 ingress switch interfaces. Note, this process is nontransparent for the traffic source, e.g., attacker. All other announced prefixes remain unaffected, but do not suffer from congestions anymore.

Despite its effectiveness in many situations, blackholing leaves still room for improvement [29]:

While blackholing at IXPs shields member networks and the links from congestions, it cannot distinguish between legitimate and malicious traffic. All packets destined for the defined IP prefix are dropped and, thus, it is not reachable from all upstream networks on the data path.

According to reports of operators obtained at the ENDEAVOUR workshops [14] and our RIPE plenary discussion [30] the granularity of blackholing is to coarse grained. Filtering on layer 4 ports is a common practice within several ASes. To provide this filtering tool to an IXP is desirable. It allows an AS to filter out the traffic before it may congests its own network or the IXP link.

Another limitation is that after detecting a massive DDoS attack the operator must trigger blackholing. This is a manual process where the router configuration must be adjusted in order to announce an IP prefix under attack via BGP. From now on the operators have neither insights in the traffic volumes (e.g., attack terminated), nor in the traffic patterns (e.g., port mix).

Technical Description

SDN and OpenFlow in particular allow to define very specific drop rules to discard packets. This be either, as offered by blackholing already, a certain prefix, or also more effective filter criteria such as Transport Control Protocol (TCP)/User Datagram Protocol (UDP) ports. For instance, some of the largest DDoS attacks solely use UDP. Hence, the IXP can provide an interface, e.g., Application Programming Interface (API) or website, to specify their own very precise drop rules executed by the IXP. Depending on the specification a drop rule must be installed in multiple different switches. Thus, the controller needs to calculate the required location for a flow rule and subsequently install it at those switches. The flow rule should remain their until the member revokes the dispensed blackholing rule.

Thus, fine grained blackholing is more effective to shield members from large volumetric attacks. However, some legitimate traffic may still suffer from the dropping of certain packets. SDN also comes with the feature to rate-limit traffic based on various header fields, e.g., transport layer port or transport layer protocol. Thus, traffic from selected members can be limited to a non-critical volume. This reduces the negative impact of attacks to a minimum, while legitimate traffic still has a chance to get trough.

The implementation of such an advanced blackholing through SDN does not require a router configuration change at member's edge. The benefits are that the risk of misconfiguration is lowered. Standardized flow rules are installed on the behalf of the network operator through a well-defined interface (e.g., API). Additionally, this allows operators to (semi-)automate the blackholing process and integrate it within their management environment.

First, the IXP can provide insights in the currently blackholed traffic by monitoring the corresponding flows, i.e., usage of OpenFlow flow counters. In accordance with deliverable 3.2 [13] an advanced monitoring system may allows to monitor traffic so that they issue warnings if traffic properties change significantly. This can be i) a very different traffic volume and pattern than normally; ii) during activated blackholing a traffic volume that is decreases to an acceptable level.

5 Bandwidth Management

The current setup of IXPs leverages a number of hardware features, e.g., protocols or capacities. This leaves space for integration of other services that can be virtualized. For instance, most members set up a dedicated private peering connection with each other. Such a service can also be offered by IXPs through virtualized private peerings. On the other hand it can be beneficial for members if their control plane traffic is protected, i.e., from congestions. The following Section elaborates on these approaches.

5.1 Virtualized Private Peering

Current Situation

The only way that IXP members have to express their transmission (i.e. bandwidth) needs is by choosing the transmission rate of their physical connections (i.e. port(s)) to the IXP. In fact, this transmission capacity is shared among all the traffic submitted by the IXP member whatever the next hop (BGP) member. As a consequence, an IXP member can not express a bandwidth requirement (or another/additional Quality of Service (QoS) requirement) for the traffic that it exchanges with a particular member.
IXP usually apply network resource over-provisioning to provide their member with a satisfactory QoS. In unusual situations (e.g. sustained overload at some IXP fabric locations or failures), no minimal performance can be guaranteed to the traffic exchanged between a pair of IXP members. Their traffic suffers from performance degradation until the IXP fabric regains its nominal state.

The basic idea of this use case is to provide some form of private peering (on the data plane not on the BGP level (advertisements)) between a couple of IXP member, i.e. a member-to-member virtual link that spans the IXP fabric with guaranteed QoS. This is what we refer to as virtualized private peering. Depending on business expectations, many options can be considered: *i)* either an on-demand version of the virtualized private peering with the ability for an IXP member to update on the fly (with a fews seconds of response time) the bandwidth needs or a conservative version established at subscription time with potentially time-based needs *ii)* the support of point-to-multipoint (or one-to-many) virtual links in addition to classical point-to-point links *iii)* the application of virtualized private peering to a group (> 2) of IXP members with an aggregate shared bandwidth.

Available Solutions

Provisioning an end-to-end (or a set of) virtual link on the IXP network goes through two steps: *i)* resource discovery to assess the available network resources, *ii)* resource allocation, which computes the necessary resources to provide the required QoS and virtual link(s) deployment on the network. Clearly, resource allocation, also known as virtual link/network resource embedding, is the most challenging step. It has attracted a lot of attention from the research community during the last few years [10]. Several optimization approaches were proposed, some follow an integrated strategy considering simultaneously the resource embedding of virtual nodes and links and, most, follow a split approach that considers separately and successively the two resource embedding problems. Whatever the approach, virtual link resource allocation is the core component of any virtual network resource embedding method.

Without being exhaustive, virtual link resource allocation methods can be classified with respect to the following criteria. Some are related to the characteristics of the virtual links being considered, namely: the type which is usually point-to-point but can also be point-to-multipoint, the QoS requirements with possibly a bandwidth, a delay and/or a loss rate requirement. The others are related to some features of the methods such as: *i)* the allocation process which can be online or offline (the requests are fully known in advance). *ii)* the general class to which a method belongs which can be heuristic or exact. *iii)* the followed approach which leads either to a stand-alone method exclusively concerned with virtual link resource embedding or a more general method that integrates and combines virtual node and virtual link resource allocation, *iv)* the network resources that are concerned by the allocation which can be the bandwidth of physical links and, possibly, the switching resources of nodes [1] (that are needed to forward the packets that belong to a virtual link); and (5) the potential support of techniques that contribute to improve the efficiency of the methods, such as path splitting and/or migration which allows the reallocation of resources to already admitted virtual links. Table 5.1 summarises the existing work according to these criteria (where P2P, P2M, BW and ILP respectively stand for point-to-point, point-to-multipoint, bandwidth and integer-linear Programming). All these methods were designed for legacy network infrastructures and hence do not benefit from the flexible flow-based forwarding brought by the SDN paradigm, which allows unprecedented control on network forwarding behaviour. Indeed, new methods that have the complete freedom in choosing the optimal physical paths and the associated resources that support the virtual links can be devised with no interference from any other network function (such as routing).

Technical Description

As explained in the previous section, setting up a virtualized private

[1] other node resources (typically, processor and memory resources) may be allocated to virtual nodes by the virtual node resource embedding algorithm

	VL type	VL QoS	allocation process	method	network resources	supported techniques
[54]	P2P	BW	offline	integrated	link	
[47]	P2P	BW delay	online	integrated	link	
[42]	P2P	BW	online	stand-alone	link	path-split migration
[17] [18]	P2P	BW	online	stand-alone	link	
[37]	P2P P2M	BW	offline	stand-alone	link	
[77]	P2P	BW	online	stand-alone	link	
[78]	P2P P2M	BW	offline	stand-alone	link	
[41]	P2P	BW	online offline	integrated	link	path-split
[50]	P2P	BW delay	online	integrated	link	
[60]	P2P	BW	offline online	integrated	link	

Table 3: Classification of virtual link resource allocation methods

peering mainly resorts to solving a virtual links embedding problem on an SDN/OpenFlow network. A portion (predefined slices) of the fabric's network resources can be dedicated for providing this IXP capability.

More effort is required to that take into account, on the one hand, some of the SDN/OpenFlow constraints (limited size of flow tables, meter tables and group tables entries) and, on the other hand, on the specificities of the IXP application context.

5.2 Control Plane Traffic Protection

Current Situation

Configuring and programming of the forwarding behavior of routers relies on control plane protocols (e.g., BGP or OSPF). For the member router in an IXP scenario, BGP is the predominant control plane protocol. BGP enables each member router to exchange reachability information which is essential for establishing peering connections over the IXP switching fabric. Hence, control plane packets must be delivered between the individual routers at any time.

Furthermore, IXP commonly operate route servers, with which a member router may establish a BGP session, to benefit from the rich public peering relations at IXPs. The IXP networks itself are usually highly overprovisioned and thus do not have to deal with congestions at all. However, the member's port can become a potential bottleneck. Since one member can receive traffic from multiple other members, a port can easily become congested. This is strengthen by the fact that members can operate ports with different port speeds, ranging from 1 Gbps to up to 100 Gbps. This asymmetry in port speeds of sender and receiver ports also boosts the potential of port congestions (See also Section 6.1).

Port congestions can jeopardize control plane packets. Thus, resulting in potential BGP session drops, due to the inability of receiving BGP packets from either another member or the route server.

Available Solutions

Unlike legacy network technologies such as ATM or ISDN, IP network deliver control plane packets and data plane packets within the same pipe. Thus, router vendors have already identified the importance of prioritizing control plane packets over data plane packets. Cisco routers tag by default outgoing control plane packets within the TOS field [7] of the IP header [2]. Thus, enabling the subsequent networking devices to prioritize those packets over other untagged packets.

However, since IXP networks operate on layer 2, their infrastructure does not take such tags into account. Layer 3 information is unavailable for the hardware deployed at IXPs today [28]. Therefore, prioritizing of control plane packets, even if they are tagged by the member's routers, is unrealizable.

Technical Description

OpenFlow defines an extensive set of matching fields, which implements header field matching for layer 2 up to layer 4. Thus, OpenFlow capable hardware deployed in an IXP network can be programmed to interpret the TOS field utilized by Cisco equipment. In case a router sends untagged control plane packets, we can craft flow rules to match on specific port numbers (e.g., TCP 179 for BGP [63]) to match those control plane packets.

While protecting control plane traffic from being dropped due to port congestion requires visibility of this traffic, it also requires mechanisms to prioritize those packets over other packets. OpenFlow supports queuing for implementing rate limits. Rate limits can implement a minimum and

[2] http://www.cisco.com/c/en/us/support/docs/quality-of-service-qos/qos-congestion-management-queueing/18664-rtgupdates.html

maximum rate for a given flow. Thus, control plane packets can be assigned with a minimum rate to ensure prioritized forwarding at the IXP egress port prior to other traffic.

Protecting control plane traffic at this level inside the IXP infrastructure will increase resilience of operation in case of port congestions for all member router without the need of changing alter their configuration.

6 Novel Services

The objective of ENDEAVOUR is to address current limitations of the Internet interconnection model, as well as to open the possibility for novel services. Thus, creating the possibility for new economic models around the created ecosystems. This Section suggests such novel services and sketches how they can be implemented.

6.1 Destination Port Congestion Awareness

Current Situation

The amount of traffic transferred through the Internet has steadily increased over the past years, and standard control plane mechanisms such as BGP have trouble coping with the dynamics of network interconnection.

Traffic is exchanged between members based on BGP reachability information at IXPs. However, the distributed control plane information is limited to reachability, i.e., if an IP prefix is reachable or not. Other valuable information for network operators, e.g., congestions or Round Trip Time (RTT) are not considered in BGP.

Content-heavy or latency-sensitive networks adopted to such conditions by implementing their own overlay networks, including powerful measurement infrastructures. Due to the nature of measurement-based reactions it is generally rather reactive than proactive. This conflicts with the goal to provide a maximized quality of experience at any times.

IXPs can provide detailed information on the current port utilization of other members. This allows to take informed decisions whether sending it to member A or rather to member B, while both announce the same prefix (e.g., available through private peering over the IXP).

Available Solutions

While there are available solutions to receive congestion notifications, they are not designed with IXPs in mind. Explicit Congestion Notification

(ECN) RFC3168 is designed to throttle TCP connections. Even if they lead to a resolution of the congestion, this might be not necessary since another AS at the same IXP would have provided enough spare capacity. Moreover, ECN throttles only already initiated data flows and is not compatible with UDP.

Technical Description

To provide congestion information two steps have to be performed, *information gathering* and then *information distribution* at the IXP.

The information gathering process shall collect the required utilization of all ports at all switches. This can be implemented by leveraging the statistic features of OpenFlow hardware. The OpenFlow standard defines counters for received and transmitted bytes per port. These counters can be periodically requested by the controller. The time interval between two subsequent requests depends on the required resolution. It also depends on the capabilities of the hardware switch to provide those counters. Instead of polling based statistics, those counters could also be send to the controller in a push based manner (available in OF 1.5.1). Those push based statistics are initiated by the switch based on predefined thresholds. In this case, the controller can be made aware of those statistics only if a congestion is present according to the predefined thresholds.

Access to the switch buffers can also provide an instant view on the current congestion status. Congestions on a port will cause the switch to fill up buffers. Unfortunately, per port buffers a rather rare. For instance as it is the case with Alcatel-Lucent hardware, ports of an individual line card share a common buffer pool. In addition, buffers might be kept for ingress ports rather than egress ports. Therefore, the utilization of switch buffers might not correlate to an individual egress port (member port).

After gathering the per port congestion information at the controller, it can be distributed to the individual members of an IXP. Several approaches are considered for implementation:

REpresentational State Transfer (REST) API: A member could query the port status of other members using a simple API provided by an IXP.

JSON Member List: IXPs already provide information about their member in a defined JSON format[3]. The per port congestion information can be included into this file. A member woudl need to poll this file periodically, in order to receive the latest information.

[3] https://github.com/euro-ix/json-schemas

BGP Communities: The route server can populate the congestion status via BGP communities added to a BGP message. One limitation of this approach is the frequency of BGP messages. BGP messages are only send by the route server if a routing entry has changed based on the AS path or other metrics. Since the congestion status might change more frequently, it would either increase the number of BGP update messages, or a member must rely on the congestion status reported by the latest BGP message.

Route Server: The per port congestion status information could also be taken into account by the route server itself, during the best path selection algorithm. In this case the information does not necessarily need to be revealed to the members.

6.2 IXP as Transport (e.g., Transit) Marketplace

Current Situation

ASes rely on transit providers and peering interconnections to attain universal Internet reachability. While IXPs enable and help to organize peering interconnections, the transit market bears no similarity.

Despite of a common billing practices where transit providers typically charge for the 95th percentile of the larger direction (down or upstream) short term traffic rates, negotiation of transit contracts is obscure and cumbersome: *i)* contracts are slowly negotiated and involve plenty of mails and discussions via telephone. As a result the transit market lacks transparency and automation is hard. *ii)* To establish a business relationship (i.e., transit) between two ASes a physical interconnection is required. This can either be a fiber within the same data center, a darkfiber through an metropolitan area, or a layer 2 transport or backbone provider. Unfortunately, this dedicated type of interconnection is rather inflexible. Given this two main reasons, this situation begs for an open market place with transparent pricing.

Available Solutions

Going beyond their original goal of being the central place for peering relationships, IXPs have expanded the transport opportunities: In remote peering, ASes offer layer 2 transport across IXPs, and there is anecdotal evidence of transit providers offering their transport services at IXPs. Building upon this increasing diversity IXPs could become a marketplace for transport in general. Benefiting from the co-location of ASes, the existing infrastructure and transparency standards, the IXP could help meet supply and demand of transport over the Internet in all of its different flavors.

Technical Description

While the business opportunity for this use case is clear, it lacks concrete vision on the implementation. As an ultimate goal we anticipate a solution that uses network layer protocols to distribute fees and negotiates contractual relations. Hereby, SDN can play a key-role since it comes with the desired flexibility. It can be designed to maintain the current routing state within a central entity that allows to faster shift the traffic between transit providers.

6.3 Service Chaining

Current Situation

Service Chaining is an emerging set of technologies and processes that enables operators to configure network services dynamically without having to make changes to the network at the hardware level. By routing traffic flows according to a "service graph", service chaining addresses the requirement for both optimization of the network (i.e., better utilization of resources) and monetization (i.e., provisioning of services that are tailored to the member context). The most common services include packet inspection (i.e., firewall, intrusion prevention systems), traffic optimization (i.e., traffic shaping), and protocol proxies (i.e., NAT, Domain Name System (DNS) cache, Session Initiation Protocol (SIP)) [38].

Available Solutions

Nowadays, network devices can be hardwired back-to-back to create a processing path, chaining of network functions in hardware. The challenge is that hardwired service chains are difficult to deploy and change. They are characterized by hand-crafted complexity, with life cycles that are long and static. Other proposed techniques are built upon BGP, Multiprotocol Label Switching (MPLS) control plane mechanisms to construct virtual topologies for service chaining. The virtual service topologies interconnect network zones and constrain the flow of traffic between these zones via a sequence of service nodes [61]. In this scenario, given the BGP control plane being used, the flow is recognized based on the only IP destination address limiting the overall service granularity.

Technical Description

In competitive markets, with rapid innovation at the application layer, this limits operators' ability to address emerging use cases and business models. In particular, an SDN approach, with its predominant realization, the OpenFlow protocol, open new scenarios in service chaining techniques given its centralized management (i.e., easier the configuration) and its enhanced granularity (i.e., flows can be selected based on layer 4 protocols). Especially, inter-domain routing convergence points such as IXPs appear as convenient locations for a central deployment of service chains.

6.4 Autonomous Anomaly Detection

Current Situation

Network anomaly detection has become a vital component of any network in today's Internet. Ranging from non-malicious unexpected events such as flash-crowds and failures, to network attacks such as DoS and network scans, network traffic anomalies can have serious detrimental effects on the performance and integrity of a network. The principal challenge in automatically detecting and characterizing traffic anomalies is that these are moving targets. It is difficult to precisely and permanently define the set of possible anomalies that may arise, especially in the case of network attacks, because new attacks as well as new variants to already known attacks are continuously emerging. A general anomaly detection system should therefore be able to detect a wide range of anomalies with diverse structures, using the least amount of previous knowledge and information, ideally none.

Available Solutions

The problem of network anomaly detection has been extensively studied during the last decade. Two different approaches are by far dominant in current research literature and commercial detection systems: signature-based detection and supervised-learning-based detection. Both approaches require some kind of guidance to work; hence they are generally referred to as supervised-detection approaches. Signature-based detection systems are highly effective to detect those anomalies that are programmed to alert on. When a new anomaly is discovered, generally after its occurrence, the associated signature is coded by human experts, which is then used to detect a new occurrence of the same anomaly. Such a detection approach is powerful and very easy to understand, because the operator can directly relate the detected anomaly to its specific signature. However, these systems cannot defend the network against new attacks, simply because they cannot recognize what they do not know. Furthermore, building new signatures is expensive, as it involves manual inspection by human experts.

On the other hand, supervised-learning-based detection uses labeled traffic data to train a baseline model for normal-operation traffic, detecting anomalies as patterns that deviate from this model. Such methods can detect new kinds of anomalies and network attacks not seen before, because they will naturally deviate from the baseline. Nevertheless, supervised-learning requires training, which is time-consuming and depends on the availability of purely anomaly-free traffic data-sets. Labeling traffic as anomaly-free is expensive and hard to achieve in practice, since it is difficult to guarantee that no anomalies are hidden inside the collected traffic. Additionally, it is not easy to maintain an accurate and up-to-date model for anomaly-free traffic, particularly when new services and applications are constantly emerging.

Apart from detection, operators need to analyze and characterize network anomalies, in order to take accurate countermeasures. The characterization of an anomaly can be a hard and time-consuming task. The analysis may become a particular bottleneck when new anomalies are detected, because the network operator has to manually dig into many traffic descriptors to understand its nature. In the current traffic scenario, even expert operators can be quickly overwhelmed if further information is not provided to prioritize the time spent on the analysis.

Security companies such as Norton, Arbor Networks, Symantec, Avast, etc., to quote a few, mostly propose solutions based on misuse detection. These systems rely on a database consisting of signatures of the known attacks or intrusion attempts. Every time a new attack is discovered, the signatures database is updated. This procedure is generally costly. So is the security business model, as security companies rely on selling the results of the skills of their engineers and security experts. This is a very slow, inefficient and ineffective process, leaving systems and networks unprotected for long periods. This is however a very good source of revenue for security companies, at the expense of badly protected networked systems. These security companies also provide security tools based on anomaly detection, which relies on supervised learning. Such a strategy also requires the skills of security experts for providing the normal or anomalous models. Therefore, they still follow the same business model.

The main drawback of the current security model, because of its cost and sluggishness, is that it does not enforce a fully secure digital world, first because it is reactive and second because it requires the full adhesion to these concepts of the entire actors of the ecosystem that need to buy and install immediately any security software update. Unprotected or badly protected systems can then be an easy target for hackers, who can corrupt them and use them for perpetrating massive and more dangerous attacks.

This has to be added to the fact that such reactive security systems cannot detect unknown attacks.

Technical Description

This use case deals with designing an autonomous anomaly detection system that does not rely on previous acquired knowledge, i.e. which does not need known attack signatures, labeled traffic, training, etc. It also aims at autonomously triggering proper countermeasures when attacks are detected among the legitimate traffic classes. The result of this use case cannot be fully stated at this point of the project. At this stage the directions we are proposing for this use case are:

- To take advantage of an unsupervised clustering approach to detect and characterize network anomalies, without relying on signatures, statistical training, or labeled traffic, which represents a significant step towards the autonomy of networks.
- To propose for accomplishing unsupervised detection some robust (i.e. that converge to the same result when applied to similar cases) data-clustering techniques to avoid general clustering drawbacks, such as sensitivity to initial conditions, course of dimensionality, cluster correlation, etc..
- To use the clustering results for issuing traffic characteristics and especially the rules characterizing the anomalies, so that they could be used as filtering rules.

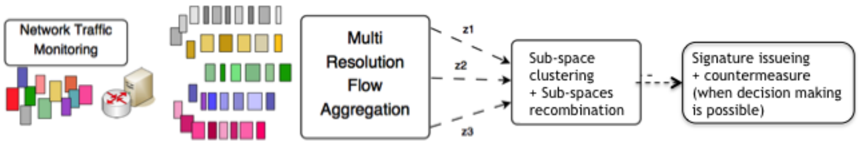
Network Traffic Monitoring
Multi Resolution Flow Aggregation
z1
z2
z3
Sub-space clustering + Sub-spaces recombination
Signature issueing + countermeasure (when decision making is possible)

Figure 4: Functional three stages architecture for anomaly detection system

Autonomously detecting anomalies in network traffic is a complex process that consists of several tasks: First, the network traffic is monitored. Second, it is classified into flows according to the features under consideration. The flows are in particular aggregated several times with different time bins and address prefix sizes in order to cope with all kinds of anomalies whatever their structure and strategies are for remaining undetected. The third step deals with the sub-space clustering algorithm ? the unsupervised machine learning techniques that we designed for this use case (see [48]). Of course, as we split the problem in a complete set of sub-spaces with an exhaustive combination of n features (n being small, we recommend less than 5 for good performances), it also includes recombining the results got in the different sub-spaces. For this purpose we apply evidence accumulation, inter-clustering result association, and correlation, as described in [48]. Last, based on the characteristics of the anomalies detected in the third step, it is possible to issue an accurate signature and an abnormality score. Based on the abnormality score, the signature can be directly exported to network security devices (if the anomaly has been classified as malicious), or sent to the network operator if it is not possible to make autonomously the decision.

Figure 4 exhibits the draft architecture for the anomaly detection process as it is currently defined. A detailed description of the algorithm is nevertheless provided in [48].

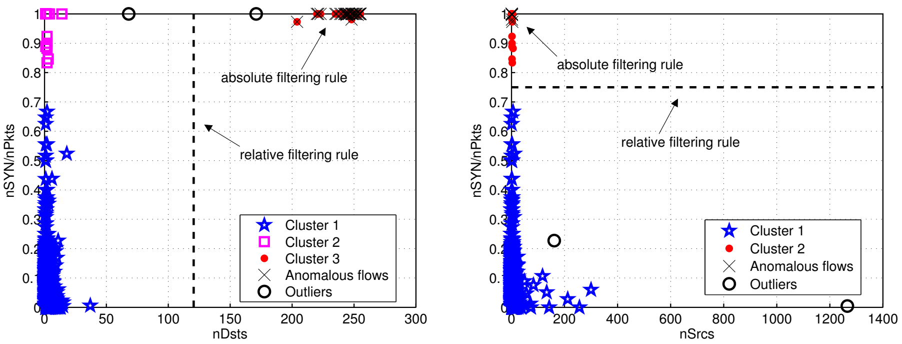

Figure 5: Filtering rules for characterization of a found network scan.

Example: Figures 5.(a,b) depicts the results of the characterization phase for a network scan anomaly. Each sub-figure represents a partition P_n for which filtering rules were found. They involve the number of IP sources and destinations, and the fraction of SYN packets. Combining them produces a signature that can be expressed as (nSrcs == 1) $\wedge$ (nDsts $> \lambda_1$) $\wedge$ (nSYN/nPkts $> \lambda_2$), where λ_1 and λ_2 are two thresholds obtained by separating normal and anomalous clusters at half distance. This signature makes perfect sense: the network scan uses SYN packets from a single attacking host to a large number of victims. The main advantage of the unsupervised approach relies on the fact that this new signature has been produced without any previous information about the attack or the baseline traffic.

IXPs are strongly concerned by the detection of anomalous and malicious traffic, as it can dramatically impact the performance of the exchange network by consuming uselessly large amounts of resources. A strong issue when deploying anomaly detection algorithms and associated tools is related to the distribution of this functionality on the full IXP network: as it works on top of the monitoring service (i.e., usually on a single link), an instance of the tool should be installed on each network device (switch, or dedicated machine on each link). This would represent a strong investment and would increase the complexity of the network. In addition, the correlation of all results of the anomaly detection entities infers extra communications on the data plane, possibly leading to general communication performance decrease, up to congestions.

SDN in such context provides strong advantages for deploying the anomaly detection functionality. Thanks to the function virtualization concept, the anomaly detection function can be virtualized on all the network, on top of the virtualized monitoring function. The SDN virtualization concept makes easy the distribution of the anomaly detection function: a single entity is required (for instance on the manager machine) and can be virtualized on any link or switch of the network thanks to the SDN principles.

Another strong benefit in using the SDN technology for anomaly detection function is linked to the enforcement of countermeasures. Actually, most anomalous traffic and all malicious traffic has to be discarded from the network as soon as it is detected., i.e., ideally when it reaches the ingress border switches of the IXP. The proposed anomaly detection algorithm is able to autonomously generate signatures, that are basically the filtering rules for configuring a security device (e.g., firewall, filtering router or switch). SDN basically leverages such rules. Then, thanks to the SDN concept, the security filtering rules can be easily and quickly deployed thanks to the control plan of the SDN based network. The only constraint for the anomaly detection algorithm deals with providing the signature (i.e., the filtering rules) using the SDN controller syntax.

6.5 Virtual Peering Router

Current Situation

Network virtualization is not a recent trend in network. Since the beginning VLANs were being used to create multiple logical networks in a single physical infrastructure. Nowadays, thanks to the emergence the SDN, network virtualization is becoming a norm for data centers, like server virtualization had become years ago. The flexibility and the agility provided by SDN networks are fostering the development of software switches and protocol stacks for virtualization platforms. While the data center is the environment to which these technologies were firstly designed, they seem to be general enough for applications outside of its original scope.

A virtual router is a software only implementation of a hardware router protocols and forwarding behavior. Apart from the clearly performance difference, due to speed limitations of a pure software approach, virtual routers work exactly like the physical version. It seems the perfect case for SDN because it splits packet forwarding from the path calculation. In this scenario, it is possible to execute control plane protocols and algorithms in the virtual router to calculate the network routes. The result is then pushed into the data plane as forwarding rules. Also, the control plane configuration (e.g: BGP configuration) is kept at the virtual routers. Because of the flexibility and programmability of these virtual instances, provisioning a new router in the network might be faster and easier. These benefits make the case for virtual peering routers at an IXP. Connecting members via virtual routers, maintained by the IXP [46], could ease the connection setup and also protect the fabric and members from accidental BGP configurations [46] .

Available Solutions

The closest solution to virtual peering at the IXPs are route servers. These servers act as a central BGP router and usually run an open source protocol stack to establish sessions with the IXP members. It releases the members from the burden to configure multiple BGP sessions with its peering partners. While route servers have been used with success on IXPs, there are some disadvantages.

- Data link failure cannot be detected by the BGP control plane.
- Route servers with policy control per client might cause path hiding. [45]
- Route servers do not avoid BGP configuration errors from the members routers. (e.g., accidentally overwriting the next-hop address of prefixes received by the route server can put it in a black hole).

Technical Description

In order to mitigate possible problems in the IXP fabric and leverage the services offered to members, the use case of virtual peering router could be implemented through the use of Virtual Machine (VM) running a virtual BGP-router. Each VM maps to a member router and the BGP configuration is performed by the IXP. The member policies would be specified by some type of policy configuration language as RPSL [5] or the most convenient way, defined by the IXP. Moreover, these virtual routers will interact with an SDN controller giving the IXP operator a global view of the control plane.

For the data plane, the use case still requires the collocation of a member equipment at the IXP. However, because of the separation between forwarding and the network intelligence provided by SDN, peers have the option to choose cheaper and simpler switches. Two possible network equipments to implement the use case are:

Traditional routers with a programmable Forwarding Information Base (FIB). Some routers have API that enables the insertion of static routes in the FIB. In this case the virtual routers learn routes and hold the Routing Information Base (RIB). Then, the routes can be pushed into the physical router FIB. A solution that leverages the opportunity to add entries to the router's FIB is the SDN Internet router [9]. It is an agent written in the Python programming language. The agent can be installed on a router with support of Python. It allows an external controller to collect the RIB from a router and then perform optimizations to install only the most used routes in the FIB.

OpenFlow switches. OpenFlow is a protocol which enables programmability in network switches. Flow rues can match and forward packets based on arbitrary fields of the packet, rather than only on the IP destination of traditional routers. It provides much more flexibility and opportunities for new applications at the IXPs fabric. In the virtual peering router scenario, the SDN controller translates the virtual routers' RIB into OpenFlow flows. Currently, there is a number of SDN applications enabling BGP routing in OpenFlow networks, such as RouteFlow [52] and the ONOS's Atrium peering controller [11].

In both cases BGP packets need to be forwarded to the data plane in order to reach the member's AS. For this reason, forwarding entries should also be added to the data plane in order to guarantee the correct operation of control plane protocols.

Overall, the use case of Virtual Peering Routers, can also raise some privacy concerns in the member side. However, a neutral entity as the IXP seems to be the perfect environment to apply and leverage a routing outsourcing model. Moreover, we belive that the benefits of agility and automation, plus the creation of new services, overcome the possible drawbacks.

6.6 Multi-Cloud and IXP as Cloud Broker

Current Situation

With the advent of layer 3 dataceneter networks [76, 33, 39, 68] and the standardization of Ethernet Virtual Private Network (EVPN) [43], Cloud operators and sophisticated members will ask for the ability to scatter/gather physical and virtual machines, storage and fabrics into virtual Performance Optimized Datacenters (PoDs)/ clusters / datacenters. Albeit the physical resources remain fixed in their current geo-locations (scatter), they could be arbitrarily re-clustered (gather) with similar resources from other locations, and thus constitute a new virtual Datacenter (DC)). The need to build such multi-clouds in the form of continental-scale, distributed and unified (single-roof illusion) DCs has already been described in literature [73, 19, 81, 16], but no concrete solutions have been presented.

Available Solutions

The SDN-enabled IXP would be indispensable in a multi-cloud environment by providing three main services: interconnection, interfacing and brokering.

An SDN-enabled IXP will be perfectly equipped to act as an interconnection point for the scattered PoDs, providing the management, security, traffic engineering and load balancing capabilities offered to every member. In the case of the multi-cloud architecture, the IXP can become invaluable by offering more than just the interconnection, namely, providing an interface services for the inevitably heterogeneous set of PoDs.

Furthermore, vendor- and platform-independent multi-cloud environments [16], can be enabled by an SDN-based IXP that acts as a cloud broker, encompassing the functionalities of a service broker, controller and super-overlay network manager. Ideally, a Cloud-as-a-Service (CaaS) client would be able to select the cloud vendors he desires from a service marketplace managed by the controller of the IXP, and would be able to distribute any kind of cloud application to them, in load-balanced and/or reliable N+1 failover configurations. Cloud brokering at the IXP offers significant benefits to the CaaS client. First and foremost, the client is provided with vendor-independence, allowing him to easily migrate from one provider to another according to the current prices/QoS/personal preferences. In addition, the

CaaS member is provided with geo-distributed load balancing across the multi-cloud (moving the service/application to the PoD closest to the load), and reliability in the face of failures, natural disasters or malicious attacks by distributing the DC across the globe.

Technical Description

The ability to run and manage multi-cloud systems (i.e., applications targeting multiple private, public, or hybrid clouds) allows benefiting from the distinct characteristics of each individual PoD, enabling the optimization of performance, availability, and cost of the applications [32]. The main contribution of the IXP in a multi-cloud environment, would be the responsibility to extend and manage the layer 3 DC networks via established protocols, e.g., EVPN, Virtual Extensible LAN (VXLAN), coordinated from the IXP's control center.

Key features needed in an multi-cloud environment that should be supported by the IXPs that interconnect them are: heterogeneity, portability, interoperability and geo-diversity [55]. Arguably, the most severe of them is heterogeneity, both in terms of services provided by the single cloud and in terms of the layer 2/3 characteristics and protocols. The provided features of today's cloud solutions are often incompatible, as each vendor provides services with distinct characteristics. This diversity hinders the proper exploitation of the potential of cloud computing at extremely large scales, prevents interoperability and promotes vendor lock-in [32]. The following three are the main cases of heterogeneity between DCs:

- Private and public clouds: VMs/applications should migrate freely and seamlessly between them.

- Bare-metal (non-virtualized) and virtualized (both hypervisor and container) DCs: The IXP would ideally provide support (or even handle) protocol translation and packet encapsulation/decapsulation to realize the interconnection and provide the basis for application migration. The IXP's centralized controller, would also provide support and interfacing of the variety of distributed centralized controllers [49, 22] of individual PoDs.

- Converged Enhanced Ethernet (CEE)-enabled and non-CEE enabled DCs (linked to 6.7): The IXP should preserve and support Random Early Detection (RED)/ECN, possibly layer 2 Quantized Congestion Notification (QCN) congestion notifications (non-routable today) and

InfiniBand ECN markings across the heterogeneous multi-Cloud, beyond a single DC scope.

The IXP as a cloud broker would be responsible for managing the advertisement, use, performance and delivery of member clouds and helping negotiate relationships between cloud providers and cloud members [53], or even managing them transparently on behalf of any of the two parties. The main requirements of such a functionality would include the support of complex constraints, the inherent dynamism of ever-evolving PoDs and applications, as well as maintaining the QoS and enforcing SLAs. The cloud brokering problem becomes even more complex when taking into account the layers of cloud computing services, i.e., application- / platform- / infrastructure-as-a-service [75]. This high degree of dynamism, heterogeneity and complexity of the parties involved in the cloud brokering problem render an SDN-enabled IXP. The best candidate for providing a solution, with the distributed cloud management entities (individual SDN controllers or otherwise) offloading both decision-making and interfacing processes to the IXP's centralized controller.

6.7 Losslessness as a Premium Service

Current Situation

Most big data applications and DC/Cloud workloads such as Hadoop, Spark, Kafka, Hadoop distributed file system, noSQL etc. are sensitive to the long-tailed distributions of their flow completion times, as has been demonstrated in recent studies [6, 25]. The primary cause is TCP's sensitivity to packet loss and the consequent timeouts and retransmissions [80, 74]. Additionally, natively lossless distributed storage, MPI and RDMA protocols, originating from High-Performance Computing (HPC), have strong assumptions about a reliable layer 2 fabric, e.g. Peripheral Component Interconnect Express (PCIe) or InfiniBand [21, 8, 40].

This issue has been brought forth by the Converged Enhanced Ethernet (CEE) technology / 802 Datacenter Bridging [3, 4, 2], which introduces losslessness on the DC network. However, currently there exists no possibility to extend the losslessness property between two or more IXP-interconnected DCs that operate on a multi-cloud environment.

Available Solutions

An SDN-enabled IXP would be in position to offer lossless interconnection by multiplexing lossless and lossy flows on the same interconnection fabric. A centralized SDN controller could be able to route designated flows through losslessness-enabled switches or enable it on-the-go. Alternatively, in the case that the IXP does not implement the losslessness property, the controller can ensure that the relevant flows are always given the required bandwidth, through strict traffic engineering and load balancing.

Technical Description

Traditionally, switched point-to-point Ethernet has been lossy: Frames are dropped whenever a receive buffer had reached its capacity, under the generally accepted end-to-end assumption [67] that an upper layer protocol such as TCP will take the corrective steps to recover. Such a lossy network does not properly meet the semantics of the converged datacenter applications such as Fibre Channel over Ethernet (FCoE) [1] or Remote Direct Memory Access (RDMA) over Ethernet [21].

This mismatch has been recently corrected in CEE, that segregates Ethernet frames into 8 different layer 2 priorities. Each priority may be configured as either lossy or lossless. Within a lossless priority, Priority Flow Control (PFC) [4] acts as the earlier 802.3x PAUSE, preventing buffer overflows in a hop-by-hop manner - except that a paused priority does not affect other priorities.

Besides enabling network convergence, prior work has demonstrated that lossless Ethernet clusters can improve the performance of soft real-time, scale-out applications, that harness big-data. In particular, lossless fabrics avoid TCP incast throughput collapse, and can reduce the completion times by up to an order of magnitude [56, 25].

Thus arises the need for tunnels with either credit exchange across an IXP that extends the local PoD flow controls (PFC, InfiniBand or PCIe credits). This capability is not possible now, except some exploratory research [24, 25, 23], but can be provided as a premium service on top of the rest - as an opportunity to bootstrap new Cloud and HPC applications [40].

7 Member Driven Monitoring

Current Situation

Deliverable 3.2 [13] summarizes the current monitoring practices at IXPs and highlights several limitations: *i*) Counter based monitoring – The set of retrievable information is limited to the number of packets and packet size distribution per interfaces. *ii*) Flow based monitoring – The data plane is sampled at a specific rate and transmitted to an external collector, e.g., a commodity server. Due to the large amount of traffic exchanged at IXPs, flow based monitoring requires extensive external resources, such as storage and processing power for implementing lookups within the datasets. To limit the storage requirements flow based monitoring is usually applied in sampling mode. Thus, the data is not fully accurate due to inherent statistical shortcomings.

Available Solutions

Yet, IXPs exploit flow based monitoring to provide their member extended insights into the peering traffic. However, due to the enormous amount of data to be processed, members can only access a predefined view on the traffic data (e.g., traffic exchanged per peer). Trouble shooting or managing of peering relations requires members to have a more detailed view on the traffic data (e.g., layer 4 port distribution). Thus, making it challenging for IXPs to provide the relevant data on request. This usually includes manual work, which is time-consuming and error-prone. Therefore, it is desirable for IXP members to specify their own requirements on which packets should be monitored at the IXP.

Technical Description

An OpenFlow enabled switch can store per flow-rule counters. Since the flow rule matching allows fine-grained header field matching, packets can be matched on any desired granularity. Furthermore, OpenFlow rules can be defined as match-only, if they are defined without an associated action. Thus, they do not influence the forwarding behavior of switches.

Leveraging those match-only flow rules, an IXP can allow members to specify the matching part of a flow rule and access the counters associated with this flow rule periodically. This would enable members to monitor traffic on any granularity on demand in a fully automated fashion. However, privacy restrictions have to be taken into account. Hence, the IXP operator would only install flow rules after carefully validating their rightfulness.

The use case of member driven monitoring expands the visibility for members on the IXP fabric. According to specific member monitoring desires the IXP can provide specific information on demand. Therefore, a member shall define a set of monitoring rules. By leveraging OpenFlow, this allows to create specific counters directly on the data plane. This bears multiple advantages: *i*) no or limited involvement of IXP operators. *ii*) Computing and storage resources are saved at both sides. *iii*) Available monitoring data on different granularity.

8 Summary

In this deliverable we collect 15 use cases, which show the potential of SDN to simplify, secure, and enhance the connection model for IXP members. It clearly shows, that a single SDN deplyoment can enabling benefits for a large number of networks at once by deploying it at the convergence points of a multitude of network interconnections, namely an IXP.

Section 3 outlines identified shortcomings in current TE solutions. We describe scenarios in which we see a benefit of employing SDN to implement TE decisions, e.g., to load balance traffic for customers with more than one IXP port.

In Section 4, we highlight opportunities that come with the flexibility of SDN to define very specific filters. This either helps to diminish the negative impact of DoS attacks or enforces contractual policies among peers.

The use cases summarized in Section 5 explain how we anticipate added values for the IXP members by managing the bandwidth according to certain constraints. For instance, we describe measures to protect IXP members's control plane traffic.

One of the visions of ENDEAVOUR is to initiate the emerging of novel services for IXP members. Section 6 summarizes all potential novel services we could perceive so far. For each use case we outline the current limitation in the market and then highlight how to implement a new service for members on this basis.

The last use case in Section 7 discusses the flexible monitoring on demand on the behalf of a member. A set of monitoring properties is defined and data is collected through SDN monitoring rules. This increases the insights members can have into the IXPs peering platform.

9 Outlook

ENDEAVOUR anticipates more innovation and development at IXPs and therefore at the core of the Internet. Introducing SDN will allow IXPs to innovate on a higher frequency than today. One critical advantage is the increased control over the software stack of their networks. While this innovation will allow them to simplify overall operations, it will also lead to innovative and novel features for the IXP members.

With insights into the operation of a large IXP, such as DE-CIX, ENDEAVOUR will further work on foster incentives for IXP operators to deploy SDN. We will work on implementing those use cases as a prototype in order to show their potential for the IXP community in practice.

10 Acronyms

SDN Software Defined Networking

RIB Routing Information Base

BGP Border Gateway Protocol

ISP Internet Service Provider

IXP Internet eXchange Point

CDN Content Delivery Network

QoS Quality of Service

SLA Service-Level Agreement

AS Autonomous System

IP Internet Protocol

DDoS Distributed Denial of Service

DoS Denial of Service

TE Traffic Engineering

DNS Domain Name System

FIB Forwarding Information Base

VM Virtual Machine

RTT Round Trip Time

TCP Transport Control Protocol

UDP User Datagram Protocol

EVPN Ethernet Virtual Private Network

DC Datacenter

PoD Performance Optimized Datacenter

VXLAN Virtual Extensible LAN

ARP Address Resolution Protocol

ACL Access Control List

API Application Programming Interface

MPLS Multiprotocol Label Switching

MAC Message Authentication Code

ECN Explicit Congestion Notification

REST REpresentational State Transfer

NAT Network Address Translation

SIP Session Initiation Protocol

IGP Interior Gateway Protocol

CaaS Cloud-as-a-Service

CEE Converged Enhanced Ethernet

QCN Quantized Congestion Notification

HPC High-Performance Computing

RDMA Remote Direct Memory Access

FCoE Fibre Channel over Ethernet

PFC Priority Flow Control

RED Random Early Detection

PCIe Peripheral Component Interconnect Express

References

[1] Fabric Convergence with Lossless Ethernet and Fibre Channel over Ethernet (FCoE), 2008.

[2] 802.1Qau - Virtual Bridged Local Area Networks - Amendment: Congestion Notification. Technical report, 2010.

[3] P802.1Qaz/D2.5 - Virtual Bridged Local Area Networks - Amendment: Enhanced Transmission Selection for Bandwidth Sharing Between Traffic Classes. Draft standard, 2011.

[4] P802.1Qbb/D2.3 - Virtual Bridged Local Area Networks - Amendment: Priority-based Flow Control. Technical report, 2011.

[5] C. Alaettinoglu, C. Villamizar, E. Gerich, D. Kessens, D. Meyer, T. Bates, D. Karrenberg, and M. Terpstra. Routing Policy Specification Language (RPSL). RFC 2622, RFC Editor, June 1999.

[6] M. Alizadeh, A. Greenberg, D. A. Maltz, J. Padhye, P. Patel, B. Prabhakar, S. Sengupta, and M. Sridharan. DCTCP: Efficient Packet Transport for the Commoditized Data Center. In *Proc. ACM SIGCOMM 2010 Conference on Data Communication*, New Delhi, India, August 2010.

[7] P. Almquist. RFC 1349 Type of Service in the Internet Protocol Suite, 1992.

[8] H. Ballani, P. Costa, T. Karagiannis, and A. Rowstron. Towards Predictable Datacenter Networks. In *Proc. ACM SIGCOMM 2011*, Toronto, Canada, August 2011.

[9] D. Barroso. SDN Internet Router (sir) Documentation. Technical report, November 2015.

[10] A. Belbekkouche, M. M. Hasan, and A. Karmouch. Resource Discovery and Allocation in Network Virtualization. *EEE Communications Surveys and Tutorials*, 14(4), 2012.

[11] P. Berde, M. Gerola, J. Hart, Y. Higuchi, M. Kobayashi, T. Koide, B. Lantz, B. O'Connor, P. Radoslavov, W. Snow, and G. Parulkar. ONOS: Towards an Open, Distributed SDN OS. In *Proceedings of the Third Workshop on Hot Topics in Software Defined Networking*, HotSDN '14, pages 1–6, New York, NY, USA, 2014. ACM.

[12] R. Bush, O. Maennel, M. Roughan, and S. Uhlig. Internet Optometry: Assessing the Broken Glasses in Internet Reachability. In *ACM IMC*, pages 242–253. ACM, 2009.

[13] Castro et al. D.3.2: Primitives for flexible and scalable monitoring. *ENDEAVOUR*, 2016.

[14] I. Castro, C. Dietzel, S. Uhlig, and T. King. D.5.3: Report from IXP member workshops. *ENDEAVOUR*, 2015.

[15] R. K. Chang and M. Lo. Inbound Traffic Engineering for Multihomed ASs using AS Path Prepending. *Network, IEEE*, 19(2):18–25, 2005.

[16] C. Chen, C. Liu, P. Liu, B. T. Loo, and L. Ding. A Scalable Multi-datacenter Layer-2 Network Architecture. In *Proceedings of the 1st ACM SIGCOMM Symposium on Software Defined Networking Research*, SOSR '15, pages 8:1–8:12, New York, NY, USA, 2015. ACM.

[17] N. Chowdhury, M. Rahman, and R. Boutaba. Virtual Network Embedding with Coordinated Node and Link Mapping. In *INFOCOM 2009, IEEE*, pages 783–791, April 2009.

[18] N. Chowdhury, M. Rahman, and R. Boutaba. ViNEYard: Virtual Network Embedding Algorithms With Coordinated Node and Link Mapping. *Networking, IEEE/ACM Transactions on*, 20(1):206–219, Feb 2012.

[19] K. Church, A. G. Greenberg, and J. R. Hamilton. On Delivering Embarrassingly Distributed Cloud Services. In *HotNets*, pages 55–60. Citeseer, 2008.

[20] Cisco. OSPF Design Guide, 2011. http://www.cisco.com/image/gif/paws/7039/1.pdf.

[21] D. Cohen, T. Talpey, A. Kanevsky, et al. Remote Direct Memory Access over the Converged Enhanced Ethernet Fabric: Evaluating the Options. In *Proc. HOTI 2009*, New York, NY, August 2009.

[22] R. Cohen, K. Barabash, B. Rochwerger, L. Schour, D. Crisan, R. Birke, C. Minkenberg, M. Gusat, R. Recio, and V. Jain. An intent-based approach for network virtualization. In *Integrated Network Management (IM 2013), 2013 IFIP/IEEE International Symposium on*, pages 42–50. IEEE, 2013.

[23] D. Crisan, A. S. Anghel, R. Birke, C. Minkenberg, and M. Gusat. Short and Fat: TCP Performance in CEE Datacenter Networks. In *Proc. 19th Symposium on High-Performance Interconnects (HOTI 2011)*, Santa Clara, CA, August 2011.

[24] D. Crisan, R. Birke, N. Chrysos, C. Minkenberg, and M. Gusat. zFabric: How to virtualize lossless ethernet? In *Cluster Computing (CLUSTER), 2014 IEEE International Conference on*, pages 75–83. IEEE, 2014.

[25] D. Crisan, R. Birke, G. Cressier, C. Minkenberg, and M. Gusat. Got Loss? Get zOVN! In *Proc. ACM SIGCOMM 2013*, Hong Kong, China, August 2013.

[26] J. Czyz, M. Kallitsis, M. Gharaibeh, C. Papadopoulos, M. Bailey, and M. Karir. Taming the 800 Pound Gorilla: The Rise and Decline of NTP DDoS Attacks. In *ACM IMC*, 2014.

[27] DE-CIX. DE-CIX Blackholing Support. `www.de-cix.net/products-services/de-cix-frankfurt/blackholing/`.

[28] C. Dietzel and S. Bleidner. Design of Use Cases for IXP operator, 2016.

[29] C. Dietzel, A. Feldmann, and T. King. Blackholing at IXPs: On the Effectiveness of DDoS Mitigation in the Wild. To appear at PAM, 2016.

[30] ENDEAVOUR Consortium. RIPE 71 BoF: ”ENDEAVOUR: Towards a Flexible Software-Defined Network Ecosystem”. RIPE 71 Bucharest, November 2015. `https://ripe71.ripe.net/programme/meeting-plan/bof/`.

[31] EURO-IX. European Internet Exchange Association. `https://www.euro-ix.net/`.

[32] N. Ferry, A. Rossini, F. Chauvel, B. Morin, and A. Solberg. Towards model-driven provisioning, deployment, monitoring, and adaptation of multi-cloud systems. In *Proceedings of the 2013 IEEE Sixth International Conference on Cloud Computing*, CLOUD ’13, pages 887–894, Washington, DC, USA, 2013. IEEE Computer Society.

[33] D. Fisher, D. Maltz, A. Greenberg, X. Wang, H. Warncke, G. Robertson, M. Czerwinski, et al. Using visualization to support network and application management in a data center. In *Internet Network Management Workshop, 2008. INM 2008. IEEE*, pages 1–6. IEEE, 2008.

[34] B. Fortz and M. Thorup. Internet Traffic Engineering by Optimizing OSPF Weights. In *Proc. INFOCOM*, 2000.

[35] B. Fortz and M. Thorup. Increasing Internet Capacity Using Local Search. *Comp. Opt. and Appl.*, 29(1):13–48, 2004.

[36] B. Fortz and M. Thorup. Optimizing OSPF/IS-IS Weights in a Changing World. *IEEE J.Sel. A. Commun.*, 20(4):756–767, Sept. 2006.

[37] C. Frei and B. Faltings. Resource Allocation in Networks Using Abstraction and Constraint Satisfaction Techniques. In J. Jaffar, editor, *Principles and Practice of Constraint Programming – CP'99*, volume 1713 of *Lecture Notes in Computer Science*, pages 204–218. Springer Berlin Heidelberg, 1999.

[38] Gabriel Brown. Service Chaining in Carrier Networks. White Paper on behalf of Qosmos, February 2015. `http://www.qosmos.com/wp-content/uploads/2015/02/Service-Chaining-in-Carrier-Networks_WP_Heavy-Reading_Qosmos_Feb2015.pdf`.

[39] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel. The cost of a cloud: research problems in data center networks. *ACM SIGCOMM computer communication review*, 39(1):68–73, 2008.

[40] Q. He, S. Zhou, B. Kobler, D. Duffy, and T. McGlynn. Case study for running HPC applications in public clouds. In *Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing*, pages 395–401. ACM, 2010.

[41] I. Houidi, W. Louati, W. B. Ameur, and D. Zeghlache. Virtual network provisioning across multiple substrate networks. *Computer Networks*, 55(4), 2011.

[42] W.-H. Hsu, Y.-P. Shieh, C.-H. Wang, and S.-C. Yeh. Virtual Network Mapping through Path Splitting and Migration. In *Advanced Information Networking and Applications Workshops (WAINA), 2012 26th International Conference on*, pages 1095–1100, March 2012.

[43] A. Isaac, N. Bitar, J. Uttaro, R. Aggarwal, and A. Sajassi. BGP MPLS-Based Ethernet VPN. Technical report, 2015.

[44] S. Jain, A. Kumar, S. Mandal, J. Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, M. Zhu, et al. B4: Experience with a globally-deployed software defined WAN. In *ACM SIGCOMM Computer Communication Review*, volume 43, pages 3–14. ACM, 2013.

[45] E. Jasinska, N. Hilliard, R. Raszuk, and N. Bakker. Internet Exchange Route Server. Internet-Draft draft-ietf-idr-ix-bgp-route-server-03, IETF Secretariat, August 2013. `http://www.ietf.org/internet-drafts/draft-ietf-idr-ix-bgp-route-server-03.txt`.

[46] V. Kotronis, X. Dimitropoulos, and B. Ager. Outsourcing the Routing Control Logic: Better Internet Routing Based on SDN Principles. In *Proceedings of the 11th ACM Workshop on Hot Topics in Networks*, HotNets-XI, pages 55–60, New York, NY, USA, 2012. ACM.

[47] J. Lischka and H. Karl. A Virtual Network Mapping Algorithm Based on Subgraph Isomorphism Detection. In *Proceedings of the 1st ACM Workshop on Virtualized Infrastructure Systems and Architectures*, VISA '09, pages 81–88, New York, NY, USA, 2009. ACM.

[48] J. Mazel, P. Casas, Y. Labit, and P. Owezarski. Sub-space clustering, inter-clustering results association and anomaly correlation for unsupervised network anomaly detection. In *International Conference on Network and Service Management (CNSM?2011)*, Paris, France, October 2011.

[49] 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*, 38(2):69–74, 2008.

[50] M. Melo, S. Sargento, U. Killat, A. Timm-Giel, and J. Carapinha. Optimal Virtual Network Embedding: Node-Link Formulation. *Network and Service Management, IEEE Transactions on*, 10(4):356–368, December 2013.

[51] J. Moy. OSPF Version 2. RFC 2328, 1998. http://www.ietf.org/rfc/rfc2328.txt.

[52] M. R. Nascimento, C. E. Rothenberg, M. R. Salvador, C. N. A. Corrêa, S. C. de Lucena, and M. F. Magalhães. New York, NY, USA.

[53] L. D. Ngan and R. Kanagasabai. Owl-s based semantic cloud service broker. In *Web Services (ICWS), 2012 IEEE 19th International Conference on*, pages 560–567. IEEE, 2012.

[54] J. Nogueira, M. Melo, J. Carapinha, and S. Sargento. Virtual network mapping into heterogeneous substrate networks. In *Computers and Communications (ISCC), 2011 IEEE Symposium on*, pages 438–444, June 2011.

[55] F. Paraiso, N. Haderer, P. Merle, R. Rouvoy, and L. Seinturier. A Federated Multi-cloud PaaS Infrastructure. In *Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on*, pages 392–399, June 2012.

[56] A. Phanishayee, E. Krevat, et al. Measurement and Analysis of TCP Throughput Collapse in Cluster-Based Storage Systems. In *FAST'08*, February 2008.

[57] M. Prince. The DDoS That Almost Broke the Internet, March 2013. `www.blog.cloudflare.com/the-ddos-that-almost-broke-the-internet/`.

[58] B. Quoitin, C. Pelsser, O. Bonaventure, and S. Uhlig. A Performance Evaluation of BGP-based Traffic Engineering. *International Journal of Network Management*, 15(3):177–191, 2005.

[59] B. Quoitin, C. Pelsser, L. Swinnen, O. Bonaventure, and S. Uhlig. Interdomain Traffic Engineering with BGP. *IEEE Communications Magazine*, 41(5):122–128, 2003.

[60] G. R., T. R., V. I., and D. Z. A novel approach to virtual networks embedding for SDN management and orchestration. *IEEE Network Operations and Management Symposium*, 2014.

[61] D. Rao, R. Fernando, L. Fang, M. Napierala, and A. F. N. So. Virtual Topologies for Service Chaining in BGP IP MPLS VPNs. Internet-Draft, IETF, 2013.

[62] R. Raszuk, R. Fernando, K. Patel, D. McPherson, and K. Kumaki. Distribution of Diverse BGP Paths. RFC 6774, 2012. http://www.ietf.org/rfc/rfc6774.txt.

[63] Y. Rekhter, T. Li, and S. Hares. RFC 4271 A Border Gateway Protocol 4 (BGP-4), 2006.

[64] P. Richter, G. Smaragdakis, A. Feldmann, N. Chatzis, J. Boettger, and W. Willinger. Peering at Peerings: On the Role of IXP Route Servers. In *ACM IMC*, pages 31–44. ACM, 2014.

[65] C. Rossow. Amplification Hell: Revisiting Network Protocols for DDoS Abuse. In *NDSS*, 2014.

[66] P. S. Ryan and J. Gerson. A primer on Internet Exchange Points for Policymakers and Non-Engineers. 2012.

[67] J. H. Saltzer, D. P. Reed, and D. D. Clark. End-to-End Arguments in System Design. *ACM Transactions on Computer Systems*, 2(4):277–288, November 1984.

[68] A. Shieh, S. Kandula, A. Greenberg, C. Kim, and B. Saha. Sharing the Data Center Network. In *Proc. 8th USENIX Symposium on Networked Systems Design and Implementation (NSDI 2011)*, Boston, MA, April 2011.

[69] A. Singh, J. Ong, A. Agarwal, G. Anderson, A. Armistead, R. Bannon, S. Boving, G. Desai, B. Felderman, P. Germano, A. Kanagala, J. Provost, J. Simmons, E. Tanda, J. Wanderer, U. Hlzle, S. Stuart, and A. Vahdat. Jupiter Rising: A Decade of Clos Topologies and Centralized Control in Google's Datacenter Network. *SIGCOMM Comput. Commun. Rev.*, 45(5):183–197, Aug. 2015.

[70] Sipgate. The Sipgate DDoS Story, October 2014. `https://medium.com/@sipgate/ddos-attacke-auf-sipgate-a7d18bf08c03`.

[71] P. Sun, L. Vanbever, and J. Rexford. Scalable Programmable Inbound Traffic Engineering. In *Proceedings of the 1st ACM SIGCOMM Symposium on Software Defined Networking Research*, SOSR '15, pages 12:1–12:7, New York, NY, USA, 2015. ACM.

[72] S. Uhlig and O. Bonaventure. Designing BGP-based Outbound Traffic Engineering Techniques for Stub ASes. 34(5):89–106, October 2004. ACM Computer Communication Review.

[73] V. Valancius, N. Laoutaris, L. Massoulié, C. Diot, and P. Rodriguez. Greening the internet with nano data centers. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, pages 37–48. ACM, 2009.

[74] V. Vasudevan, A. Phanishayee, H. Shah, E. Krevat, D. Andersen, G. Ganger, G. Gibson, and B. Mueller. Safe and effective fine-grained TCP retransmissions for datacenter communication. In *ACM SIGCOMM computer communication review*, volume 39, pages 303–314. ACM, 2009.

[75] D. Villegas, N. Bobroff, I. Rodero, J. Delgado, Y. Liu, A. Devarakonda, L. Fong, S. M. Sadjadi, and M. Parashar. Cloud federation in a layered service model. *Journal of Computer and System Sciences*, 78(5):1330–1344, 2012.

[76] G. Wang and T. S. E. Ng. The Impact of Virtualization on Network Performance of Amazon EC2 Data Center. In *Proc. 29th Conference on Computer Communications (INFOCOM 2010)*, San Diego, CA, March 2010.

[77] Y. Wei, J. Wang, C. Wang, and X. Hu. Bandwidth Allocation in Virtual Network Based on Traffic Prediction. In *Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on*, pages 1–4, Sept 2010.

[78] Y. Xi and E. Yeh. Distributed Algorithms for Minimum Cost Multicast With Network Coding. *Networking, IEEE/ACM Transactions on*, 18(2):379–392, April 2010.

[79] Y. Rekhter and T. Li and S. Hares. BGP Best Path Selection Algorithm. RFC 4271, January 2006. `https://tools.ietf.org/html/rfc4271`.

[80] D. Zats, T. Das, P. Mohan, D. Borthakur, and R. Katz. DeTail: reducing the flow completion time tail in datacenter networks. *ACM SIGCOMM Computer Communication Review*, 42(4):139–150, 2012.

[81] Q. Zhang, L. Cheng, and R. Boutaba. Cloud computing: state-of-the-art and research challenges. *Journal of internet services and applications*, 1(1):7–18, 2010.