Dynamic Connectivity Management with an Intelligent Route Service Control Point

J. Van der Merwe, A. Cepleanu, K. D'Souza, B. Freeman, A. Greenberg,
D. Knight, R. McMillan, D. Moloney, J. Mulligan, H. Nguyen, M. Nguyen,
A. Ramarajan, S. Saad, M. Satterlee, T. Spencer, D. Toll, S. Zelingher

AT&T Labs

ABSTRACT

Increased use of demanding network applications, as well as the increase of unwanted network traffic in the form of DDoS attacks, are putting new pressures on service providers to meet the expectations of customers in terms of network availability and performance. Providers are expected to deal with potential problems in near real-time fashion. Further, many of these demanding application, such as VoIP and online gaming, are very sensitivity to even small periods of disruption. In this work we therefore specifically focus on *dynamic connectivity management*, which we broadly define as the ability to dynamically manage how and where traffic flows across a network. Because it is intimately involved with how traffic flows through the network, BGP would be an ideal candidate for many of these management tasks. Unfortunately, BGP is itself a complicated protocol and up to now the prospect of using it to perform routine management tasks has not been considered a feasible approach. In this paper we show how the simplification introduced by a centralized Intelligent Route Service Control Point (IRSCP) that allows route selection to be performed outside the routers and also allows such route selection to be informed by external network intelligence, address this quandary. We present several examples of connectivity management tasks that can benefit from our approach. We describe our trial implementation of the IRSCP and show how our approach raise the level of abstraction, allowing operators to focus on *what* functions need to be performed, rather than getting bogged down with *how* to perform them.

Categories and Subject Descriptors

C.2.2 [**Network Protocols**]: Routing Protocols

General Terms

Design, Management

Keywords

routing, route control, connectivity management, BGP

SIGCOMM'06 Workshops September 11-15, 2006, Pisa, Italy.

1. INTRODUCTION

New wanted and (unfortunately) unwanted uses of the Internet put pressure on providers to improve network management operations. For example, the Internet is carrying more and more applications, such as voice over IP (VoIP) and online gaming, that are very sensitive to even short periods of loss of connectivity. Further, the increased occurrence of distributed denial of service (DDoS) attacks likewise require more sophisticated and responsive network management practices from providers. We broadly define this timely control of how traffic flows through a network as *dynamic connectivity management*.

BGP is used today to satisfy a variety of business or traffic management needs [5]. Current use of BGP is, however, typically limited to longer time scale policy realization. Because it is intimately involved with how traffic flows through the network, BGP also offers the ideal means to facilitate more dynamic connectivity management. Unfortunately, this potential has not been realized in practice because of a number of reasons. First, BGP configuration is very complex and distributed over tens to thousands of routers depending on the size of an ISP. Changing these configurations on-demand to perform dynamic management tasks is normally not considered a viable approach. Second, the lack of direct control over the route selection process means that BGP does not lend itself to the realization of common network management tasks. For example, using the IGP path cost to break ties between a prefix that is reachable via multiple egress points does not take into account either provider concerns (traffic engineering) or customer concerns (load balancing across multiple interfaces).

In this paper we present the Intelligent Route Service Control Point (IRSCP) as a platform for intelligent route control and show how it is used to perform a number of connectivity management tasks. The IRSCP is a logically centralized routing element, separate from routers which allows control of route selection in an IP/MPLS network [7, 4]. We show how this control can be used to perform the following connectivity management tasks:

- Selective blackholing of DDoS traffic: The IRSCP reduce the negative impact of this common ISP practice by allowing the operation to be performed in a surgical manner by only dropping packets on routers where attack traffic has been detected.
- Planned maintenance dryout: The IRSCP allows the operator to move traffic away from routers on which maintenance is to be performed, in a controlled manner, before such maintenance is performed, thus reducing the potential impact.
- VPN gateway selection: The IRSCP allows MPLS VPN customers with multiple Internet gateways to explicitly select

which VPN sites should use which gateways, rather than relying on default shortest path routing through the provider network.

- Network-aware load balancing: Depending on the distribution of the offered (ingress) load, the coupling between IGP and BGP can cause the load on different egress points leading to the same destination to be completely unbalanced. Again we use the IRSCP capability of informing route selection with external information to perform load balancing across multiple egress points leading to the same destination.

Using a protocol, that operates at control time scales, to perform connectivity management tasks has the desirable properties that it enables fine grained, timely control of traffic flows. On the other hand, using a protocol that is inherently complex in itself to perform such tasks might appear to be counter productive. A major contribution of our work is to bridge this seeming inherent dilemma. Specifically, we raise the level of abstraction significantly by automating all of the details of the required protocol manipulation, allowing the operator to focus on the function to be performed, rather than how to do it. This enables a rigorous and concrete separation between policy and implementation. Our approach allows arbitrary external information to influence the route selection process. Our second contribution is illustrating the power of this ability by allowing route selection to be influenced by external information to realize common management tasks.

2. RELATED WORK

In the early 1980's the circuit switched voice network underwent a revolution with the introduction of a technology called the Network Control Point (NCP) [1]. Prior to the existence of the NCP, all call control was managed from the *internal* processor of the circuit switch that the call was passing through. Using the internal switch processor limited the call handling capacity because of (i) the limited processing power of the switch processor, (ii) the limited visibility into network resources that a single switch processor had and (iii) the limited amount of programming that could safely be accomplished on this processor. To address these problems, the NCP was introduced as an adjunct call processing platform, external to the circuit switch. The NCP became the basis on which many voice network features, still in use today, have been built (800-numbers, call centers, calling cards etc).

In the early 2000's, IP networks were faced with a very similar set of issues: (i) limited processing power on router controllers, (ii) limited network visibility adversely impacting routing and (iii) difficulty making configuration changes (let alone programming changes) to routers. This led researchers at AT&T to conceptualize an "IP-NCP" as a platform, separate from the routing infrastructure, in which route selection can be performed. This early work motivated our more recent work on the Route Control Platform (RCP) in which we developed a framework for such an approach [7]. We identified different evolutionary phases (only using iBGP, using iBGP and eBGP, and not using BGP at all), and showed that from a *routing protocol* perspective there are potential benefits for each step. In [4], we demonstrated the feasibility of this approach by prototyping an iBGP speaking RCP, working out the protocol details and showing that a scalable implementation was possible. In the work presented in this paper we show how this approach can enable external information to inform the route selection process, in much the same way that the NCP did for the circuit switched network. We have changed the name of our approach to Intelligent Route Service Control Point (IRSCP) to emphasize this ability.

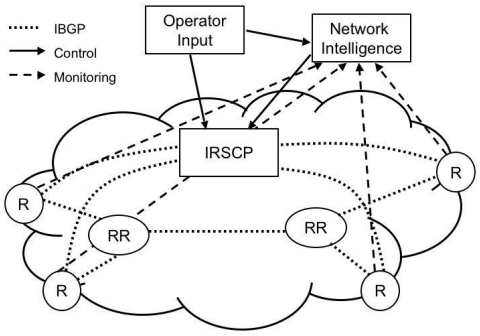

Figure 1: Intelligent Route Service Control Point (IRSCP)

A similar approach to ours has also been proposed in the IETF [3], and more recently a complete refactoring of the network architecture in the 4D project follows a similar separation of forwarding and decision planes [8]. Finally, the planned maintenance dryout approach presented in this paper functions at the IP layer only, and for multi-homed networks. A complimentary approach that works across both IP and transport layers allows near hitless planned maintenance to be performed for single-homed customers [2].

3. INTELLIGENT ROUTE SERVICE CONTROL POINT (IRSCP)

A high level view of the IRSCP in a network setting is depicted in Figure 1. The figure shows conventional network elements, routers (R) and route-reflectors (RR), as well as the IRSCP and associated functions. The IRSCP is a logically centralized network control element, i.e., it takes part in "control plane" functions but is not in the data path. In particular the IRSCP communicates with routers via iBGP: receiving routes from routers, performing route selection on behalf of each router and communicating the selected routes back to the routers (i.e., "phase one" as defined in [7]). The IRSCP also makes use of an interior gateway protocol (IGP), like OSPF, to perform per-router route selection and to break ties as part of the normal BGP route selection process [4]. In this deployment scenario, where the IRSCP is only part of the internal BGP (iBGP) process the IRSCP can not control all route selection in the network. Specifically, routers will still make their own route selection decisions based on routes learned via eBGP[1]. However, as we will show in this paper, the capabilities enabled by this limited form of route control is enough to warrant the deployment of IRSCP functionality in a production network, thus taking a small but significant step towards the more ambitious overhaul of the Internet routing infrastructure.

Figure 1 shows two forms of input into the IRSCP. First is direct operator input, for example when a task like blackholing of DDoS traffic is performed. The second IRSCP input is what we broadly call "network intelligence" and represents the fact that the IRSCP platform allows external information to directly impact the routing process. We present two examples of such network intelligence in Section 4. For VPN gateway selection the intelligence might simply be in the form of customer preference. Alternatively, for both VPN gateway selection and network aware load-balancing the intelligence can be based on actual network monitoring. In either case though, the fact is that routing is informed in dynamic fashion by external information.

While we envisage in the long run that the IRSCP will be the sole

[1] These issues will be resolved in a "phase-two" deployment where the IRSCP is also eBGP capable.

route selection and distribution function in a network [7], Figure 1 shows the IRSCP being deployed in parallel with a regular route-reflector (RR) hierarchy. This is an important pragmatic approach which allowed us to deploy the IRSCP in a production network with minimal risk involved. A parallel deployment strategy such as this has some limitation, e.g., the IRSCP can not prevent routes from being distributed via the route reflectors, but can dictate the relative preference of routes distributed by the IRSCP itself. However, as we will show in later sections, for the network management IRSCP functions described in this paper this is not a limitation.

4. IRSCP CONNECTIVITY MANAGEMENT

4.1 Selective DDoS Blackholing

Blackholing of DDoS traffic is unfortunately a common management task performed by operators. The method consists of a two step process. First a static route to a pre-defined "blackhole destination" is configured on all edge routers in the network. This static route is set up such that any traffic sent to this destination will be dropped on the edge router. The second step of the process is invoked when a DDoS attack against a specific target prefix is detected in the network: A BGP speaking entity in the network (i.e., a router or in our case the IRSCP), generates a more specific route (called the blackhole-route), for the target destination and sets the next-hop attribute of this blackhole-route to point to the previously configured blackhole destination. At this point, all traffic destined to the target destination will therefore be dropped on entry in the network.

Since most DDoS attacks target specific IP addresses, the blackhole route would only cover the corresponding /32 prefix and other traffic going to the less specific site-prefix is allowed to pass through unhindered. While blackholing clearly does mitigate the DDoS problem, the approach has a very significant and obvious drawback. Once invoked on a particular router, *all* traffic towards the destination passing through that router will be dropped, thus in effect fulfilling the intent of the attacker because the destination is now unreachable through that router. This is especially a concern when the blackhole-route is injected into the network by a router that connects to a route-reflector hierarchy, because in this case the black-hole route will be distributed to all edge routers thus prohibiting all communication to the target destination.

The IRSCP, on the other hand, can selectively send the blackhole-route only to those edge routers that carry DDoS traffic or carry a significant portion of DDoS traffic. In practice this is a critically important advantage because DDoS attacks are in fact not that distributed. For example a recent study [9] showed that for DDoS attacks observed in an ISP network, over a four week period, only 0.1% of ingress interfaces contributed more than 90% of the DDoS traffic volume. This means that significant mitigation can be realized by blackholing traffic on a small number of edge routers.

4.2 Planned Maintenance Dryout

ISPs routinely perform planned maintenance on routers to replace faulty hardware or install new router software. In instances where alternate paths are available to the prefixes advertised by the router to be taken down (the dryout-router), the IRSCP can be used to move traffic away from this router. Figure 2 shows two instances where the IRSCP can be utilized in this manner. First, when customer-edge (CE) routers are dual homed to two provider-edge (PE) routers, e.g., CE to PE_1 and PE_2 in Figure 2. This arrangement is typical for larger customers and standard practice in data centers. The second, is in the case of peering routers where prefixes available in the other ISP is normally available via all peering

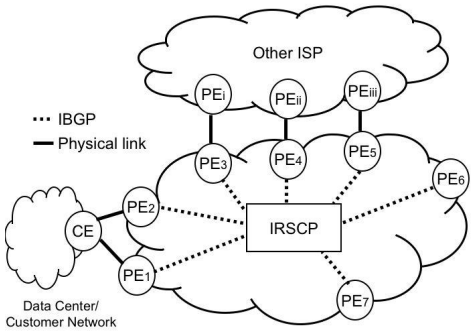

Figure 2: Planned Maintenance Dryout

routers, PE_3, PE_4 and PE_5 in the figure. Note that in both cases moving traffic away from the dryout-router involves traffic in **two** directions namely traffic entering and leaving the IRSCP-enabled ISP network.

For traffic leaving the ISP, dryout involves identifying the alternate PE(s) that are advertising the same prefixes as the dryout-PE and making routes from those PEs more preferable. For example, if PE_2 is to be dried out, the IRSCP should ensure that all other PEs in the network (i.e., PE_3 to PE_7) prefer the route via PE_1 to reach this network. The IRSCP can do this by increasing the *local preference* attribute of the routes received from PE_1 before distributing the advertised route to the other PEs in the network. The general IRSCP rule to realize this part of the dryout operation is therefore: for all prefixes advertised by the dryout-router, if those prefixes are available from another router, make them more preferred.

The same mechanism is used in the case where a peering PE router is dried out. In this case, however, more than one alternative path might be available as shown in the example in Figure 2. This offers the opportunity to refine the dryout operation by spreading the load across all the alternate available paths. For example, if PE_5 is to be dried out, the operator might prefer to send some of the traffic via PE_3 and some via PE_4 to ensure that these alternative paths are not overloaded because of the shift in traffic. Splitting traffic between possible egress points can in the first instance be done by simply proportional allocation of prefixes to the alternate egress points, but more ideally would make use of actual traffic loads to load balance the traffic as is outlined in a later section. The ability to split the redirected traffic in a controlled and informed manner differentiates the use of the IRSCP for this function from more conventional approaches. For example, common practice today to realize dryout is to change the IGP weight of selected links in the network to force traffic off the dryout router. This is a very indirect approach to the problem at hand and does not allow the operator control over where the shifted traffic should exit the network. For example, an IGP induced dryout of PE_5 in our example network would likely dump most of that traffic onto PE_4, possibly causing overload conditions on that peering link. In addition, since an IGP weight change will propagate throughout the ISP network, it can also trigger other (unintended) route changes and resulting traffic shifts [11]. IGP induced dryout also does not deal at all with traffic in the other direction, i.e., traffic coming into the ISP.

To move traffic coming into the ISP network off the dryout-router involves influencing the routing decision in neighboring networks and therefore requires cooperation from those networks. In the data center scenario this is easily done by pre-configuring a policy on the CE such that routes with a certain community value will be less preferred than routes without this value. To initiate dryout, the IRSCP then adds the special community value to all routes sent

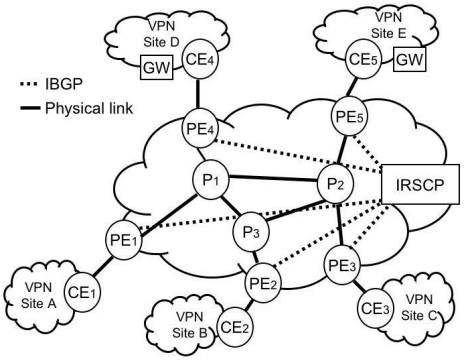

Figure 3: VPN Gateway Selection

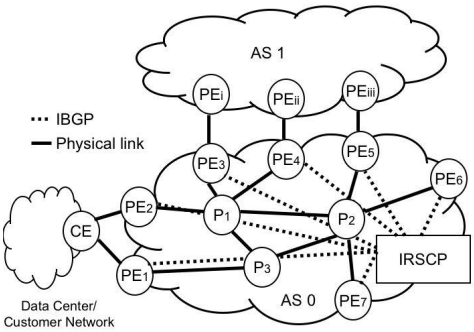

Figure 4: Network Aware Load-balancing

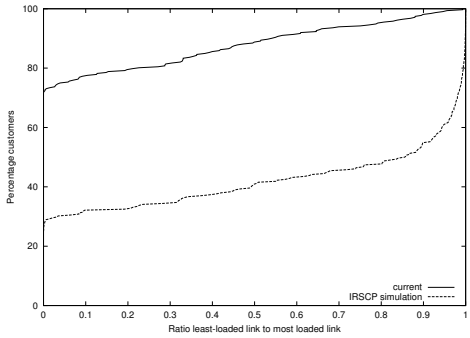

Figure 5: Traffic ratio: Least-loaded to most loaded link

to the dryout-router, which in turn will send the routes on to the CE to trigger the change in preference[2]. For example for the network in Figure 2, and assuming that PE_2 is to be dried out, the IRSCP will add the special community value to routes it receives from all other PEs before sending it to PE_2.

The same approach can be utilized in the peering scenario, although using the BGP MED attribute is generally a simpler approach. Naturally, this approach requires a peering arrangement where the peers agree to honor MEDs. Using MEDs, the IRSCP advertises selective routes with lower MEDs to the PEs where it wants to move the traffic to. Alternatively, the IRSCP can advertise routes with higher MED values via the dryout PE, leaving it up to the peer network to decide where the traffic will be moved to. As before, however, such a coarse grained approach might end up overloading peering links. Note that in the deployment scenario described in this paper, the IRSCP can only modify MEDs for iBGP routes. Such an iBGP speaking IRSCP will likely not fully dry out a peering router as some locally learned eBGP routes will not be affected. However, significant benefit is still provided to routes that do pass through the IRSCP, most notably, routes of customers and data centers in the provider network.

4.3 VPN Gateway Selection

Figure 3 shows a simple example of a typical MPLS VPN scenario. The example shows a single VPN consisting of five different sites ($A - E$). Connectivity between the sites are provided by the MPLS provider network. We show a possible internal topology for the provider network in the figure.

In our example we assume that sites D and E have gateways that collectively provide Internet connectivity for the VPN, and that the VPN customer wants to load balance traffic across the two gateways. The provider network has no knowledge of these customer goals and simply route traffic across the backbone network according to default shortest path behavior. Assuming that all IGP link weights are the same, this means that in our example traffic from CE_1/PE_1 will exit the network at PE_4/CE_4 via P_1 and traffic from CE_3/PE_3 will exit at PE_5/CE_5 via P_2. For traffic from CE_2/PE_2 two equal cost paths exist, namely via P_3 and P_1 to exit at PE_4/CE_4, or via P_3 and P_2 to exit at PE_5/CE_5. According to normal BGP tie-breaking rules (and assuming that the router-ID of PE_4 is smaller than that of PE_5), the path via PE_4 will be selected and the traffic will exit the network at PE_4/CE_4. In this simple example, this would be a fine choice if the traffic from sites A and B roughly equal that from site C. If that is not the case, which happens very frequently in practice, the Internet traffic will be unbalanced across the gateways.

[2]The PE-CE eBGP session has to be configured to allow community values to be passed through.

The IRSCP solution to this problem is to allow the customer to dictate the egress selection for traffic from different sites, thus allowing the default behavior to be overridden if required. Again the IRSCP can achieve this by increasing the preference (e.g., by assigning a higher local preference value) of VPN routes received from PE_5 when the routes are sent to PE_2. Conventional solutions to this problem involve the creation of appropriate policies on the PEs themselves. For example, in our example a policy can be installed on PE_2 such that routes from PE_5 is preferred over those from PE_4. The key differentiator of the IRSCP approach is that it is fairly simple to put these controls directly in the hands of customers through an appropriate interface, e.g., a Web portal, to the IRSCP.

4.4 Network Aware Load-balancing

In Figure 4 we show the setting for a similar problem in the Internet environment. Again the problem stems from the coupling between BGP and IGP in the BGP decision process. For example, let's assume that a significant portion of the traffic destined to the data center (or customer network) is entering the IRSCP-enabled network from AS 1. Assuming that all IGP links weights are the same, both PE_3 and PE_4 will prefer to reach the data center using the routes advertised by PE_2. PE_5 will use the router-IDs of PE_1 and PE_2 to break the tie and might therefore also select to use the path through PE_2 (if the router ID of PE_2 is smaller than that of PE_1). Either way the net result is that the link between PE_2 and CE will carry most of the traffic while the link between PE_1 and CE will be mostly idle.

This is a common problem for providers and customers alike. For example, Figure 5 shows a CDF of the traffic ratio between the most loaded link and the least loaded link for each multi-homed customer in a large ISP over a typical day. The top curve ("current") presents the actual ratios observed on that day based on sampled Netflow records collected across the ISP network. For 71.8% of the customers, this ratio is zero, showing complete imbalance with

Function	Commands	Significant Parameters
Selective Blackholing	addblackhole	–routerlist
	delblackhole	–prefix
Dryout	adddryout	–dryout
	deldryout	–backup
VPN Gateway Selection/ Load-balancing	addgroup	–ingress
	delgroup	–group
		–vpn
	addpolicy	–egress
	delpolicy	–pref
		–group
		–prefix
		–vpn

Table 1: IRSCP connectivity management primitives

the most loaded link carrying all the traffic and the least loaded link carrying no traffic at all.

We can use the IRSCP to address this problem by basing the routing decision at the ingress routers on the historic offered load towards the multi-homed customer. For example, in Figure 4 we could monitor the traffic load at all ingress routers (PE_3 to PE_7), towards the data center prefixes behind egress routers PE_1 and PE_2. This information can be harvested in a straight forward manner from, for example, Netflow data [6]. Using this offered load information, the IRSCP redirect traffic by making the route from the appropriate egress router more preferred (increasing the local preference attribute), on a per-ingress router basis. In our example network in Figure 4, the IRSCP might direct traffic from both PE_4 and PE_5 to egress PE_1, thus overriding the default IGP based selection.

The bottom curve ("IRSCP simulation") in Figure 5 shows simulation results of the same offered traffic load as before, but in this case showing the effect of IRSCP-based load balancing as described above. Only 25% of the customers still have an unbalanced ratio of zero. The data for this graph is for a single day and since our approach load balance at the granularity of an ingress router, it is quite possible that all traffic to a particular prefix enters through a single ingress router, thus not offering the possibility to balance the load. None the less the improvement of this approach is evident from the graph, e.g., 50% of the customers now achieve a ratio of 0.87 or better.

We present network aware load-balancing as a solution to a specific problem, i.e., unbalanced customer links, which occurs frequently in practice. We leave for future work to understand how this specific solution fits in with more general work on satisfying network wide traffic engineering objectives [10].

5. IMPLEMENTATION

The implementation used in our current trial deployment make use of enhancements to the Quagga open source protocol suite. Specifically we use a modified version of Quagga's BGP implementation. A collection of Perl scripts are used to automate the configuration details on the IRSCP and to present the operator with a higher level functional control interface. Table 1 shows the pertinent IRSCP primitives and the most important parameters for the connectivity management functions described in this paper[3]. In all cases the interface provides an appropriate "add/del" type command to initiate or terminate the selected connectivity management function. The required parameters for selective blackholing is simply the prefix to be blackholed and a list of routers that should be blackholing traffic towards the prefix. The dryout function take two router IPs as parameters namely the dryout router and the "backup" router where traffic should be moved to. Finally, for

[3] Note that in all cases several auxiliary commands exist which are not described here.

```
access-list 1 permit 10.1.1.1
access-list 2 permit 10.1.1.2
!
route-map outip-b permit 1
 match ip peeraddress 1
 set community 0:99
 on-match next
!
route-map outip-b permit 2
!
route-map inip-b permit 1
 match ip next-hop 2
 set local-preference 110
 on-match next
!
route-map inip-b permit 2
!
```

Figure 6: Route-map section of dryout configuration

both the VPN gateway selection and the load-balancing functions our implementation involves two primitives, which respectively require the ingress or egress routers to be specified. An optional prefix parameter can be specified to make the executed function more specific.

When a command is invoked, the current IRSCP configuration is read, parsed and interpreted to establish which parts of the configuration pertains to the current operation and to verify that everything is in place to support the operation. For example, ensuring that peering sessions are in place with the router(s) involved with the operation. Next, the IRSCP configuration is automatically updated to reflect changes related to the operation. For example, when a `adddryout --dryout 10.1.1.1 --backup 10.1.1.2` primitive is invoked the in and out route-maps of the IRSCP will be automatically updated as shown in Figure 6. The out route-map will set the community value when sending routes to the dryout router while the in route-map will ensure that routes from the backup router gets a higher local preference attribute assigned (as explained in Section 4.2). Similarly, invoking the `deldryout` version of the command will remove the appropriate route-map and access-list clauses.

Raising the level of abstraction in this manner is even more important in the case of VPN gateway selection or load-balancing, as the IRSCP logic needed to realize these functions is significantly more involved. As shown in Table 1, these functions are realized through two main primitives. The first, `add/delgroup`, associates a particular ingress router with a group of such routers that will all be receiving the same route. The second, `add/delpolicy` specifies the relative preference of a route received from a particular egress router and states the relative preference to associate with that route when it is passed to a previously defined group of routers.

For example, in Figure 3, suppose that for the VPN shown, we would like PE_1 to have a higher preference for routes received from PE_4 than those received from PE_5, and for PE_2 to have the reverse preference. Because we want to treat PE_1 and PE_2 differently, the first step would be to associate them with different groups: `addgroup --ingress PE1 -vpn VPNA --group 1` and `addgroup --ingress PE2 --vpn VPNA --group 2`. Figure 7 shows the essence of the IRSCP outgoing route-map that gets generated to as a result of running these commands[4]. Note that there are three sections to this route-map. First, is the "VPN Selection" section, which match against routes that belong to the VPN in question (i.e., based on the route-target extended community values as-

[4] Note that the route-map is slightly simplified and annotated to aid readability.

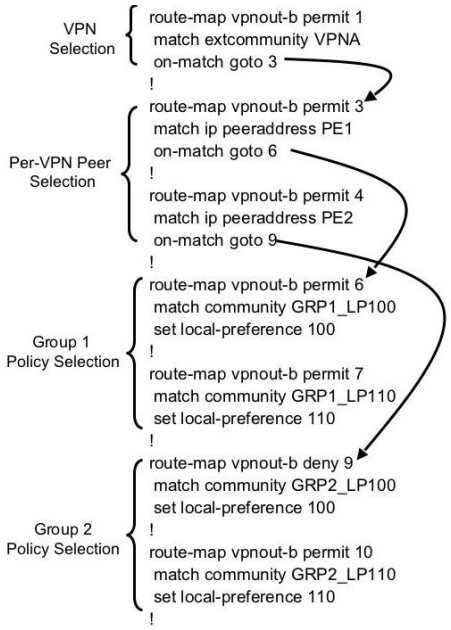

Figure 7: Out route-map section of gateway selection configuration

```
route-map vpnin-a permit 1
 match ip next-hop PE4
 match extcommunity VPNA
 match ip address prefix-list DEFAULT
 set community GRP1_LP110 additive
 on-match next
!
route-map vpnin-a permit 1
 match ip next-hop PE4
 match extcommunity VPNA
 match ip address prefix-list DEFAULT
 set community GRP2_LP100 additive
 on-match next
!
```

Figure 8: In route-map section of gateway selection configuration

sociated with the VPN). Routes that match this section would jump to the "Per-VPN Peer Selection" section. Each VPN in which gateway selection is to be performed, will have its own VPN and Per-VPN Peer selection sections. A "peeraddress" match in the Per-VPN Peer section (i.e., the IRSCP is about to send a route to the matching peer), will cause a jump in the route-map to the appropriate "Group Policy Section" where the per-group policies are applied before routes are sent to the respective peers.

Having dealt with the selective treatment of PE_1 and PE_2 in our example, the next step is to apply policies to routes received from PE_4 and PE_5. Here we only show the example for routes received from PE_4. Specifically, the commands `addpolicy --egress PE4 --vpn VPNA --prefix DEFAULT --pref 110 --group 1`, would result in the IRSCP in route-map shown in the top part of Figure 8. In essence the three match statements ensures that this statement would only be applied to routes from PE_4, that belong to the VPN in question and match the DEFAULT prefix list. A route that does match all these criteria on entry to the IRSCP, will have a special community value set that identifies it as requiring its local preference to be set to 110 when it is sent to any PE in group 1 (GRP1_LP110). Referring back to the out route-map shown in Figure 7, routes with this community value set will have their local preference set to 110. The bottom part of Figure 8 shows the result of the command `addpolicy --egress PE4 --vpn VPNA --prefix DEFAULT --pref 100 --group 2`. Since the community values are set in an "additive" fashion, a route from PE_4 that matches all the criteria will be "tagged" twice on entry to the IRSCP so that the correct part of the out route-map is triggered to realize our objective.

6. CONCLUSION

We have presented the IRSCP as the means to dynamically control the BGP protocol to realize connectivity management functions. The use of a control protocol, by necessity, brings complexity, and raising the level of abstraction and automating much of the detailed mechanics of what needs to be done, are therefor key aspects to the success of our approach. Another key contribution of our work is allowing external information to inform route selection, whether that be by making use of network load conditions or by providing customers direct control about how their traffic is routed through the network. In the work presented in this paper we have clearly just scratched the surface in terms of the total set of connectivity management tasks and service/feature applications that are enabled by the IRSCP. The IRSCP realizes a new paradigm in the continued evolution of IP backbone networks, from dumb IP transports to networks with dynamic informed connectivity management that meet the needs of demanding customers and applications.

7. REFERENCES

[1] Special Issue on Stored Program Controlled Network. The Bell System Technical Journal, September 1982. Volume 61, Number 7, Part 3.

[2] M. Agrawal, S. Bailey, A. Greenberg, J. Pastor, P. Sebos, S. Seshan, J. Van der Merwe, and J. Yates. RouterFarm: Towards a Dynamic, Manageable Network Edge. SIGCOMM Workshop, INM, Sept 2006.

[3] O. Bonaventure, S. Uhlig, and B. Quoitin. The Case for More Versatile BGP Route Reflectors. Internet Draft draft-bonaventure-bgp-route-reflectors-00.txt, July 2004.

[4] M. Caesar, D. Caldwell, N. Feamster, J. Rexford, A. Shaikh, and J. van der Merwe. Design and implementation of a Routing Control Platform. In *Proc. NSDI*, 2005.

[5] M. Caesar and J. Rexford. Bgp policies in isp networks. IEEE Network Magazine, November 2005.

[6] N. Duffield and C. Lund. Predicting Resource Usage and Estimation Accuracy in an IP Flow Measurement Collection Infrastructure. In *SIGCOMM IMC*, 2003.

[7] N. Feamster, H. Balakrishnan, J. Rexford, A. Shaikh, and J. van der Merwe. The Case for Separating Routing from Routers. FDNA Workshop, Aug 2004.

[8] A. Greenberg, G. Hjalmtysson, D. A. Maltz, A. Myers, J. Rexford, G. Xie, H. Yan, J. Zhan, and H. Zhang. A clean slate 4D approach to network control and management. *SIGCOMM Comput. Commun. Rev.*, 35(5), 2005.

[9] Z. M. Mao, V. Sekar, O. Spatscheck, J. van der Merwe, and R. Vasudevan. Analyzing Large DDoS Attacks Using Multiple Data Sources. SIGCOMM Workshop, LSAD, Sept 2006.

[10] R. Teixeira, T. G. Griffin, M. G. C. Resende, and J. Rexford. TIE breaking: Tunable interdomain egress selection. Proc. CoNEXT, October 2005.

[11] R. Teixeira, A. Shaikh, T. Griffin, and J. Rexford. Dynamics of hot-potato routing in ip networks. ACM SIGMETRICS, June 2004.