Intent-based Analysis of Network-wide Routing Policy Configuration

Kyriaki Levanti
ECE Department
Carnegie Mellon University
klevanti@andrew.cmu.edu

Hyong S. Kim
ECE Department
Carnegie Mellon University
kim@ece.cmu.edu

Tina Wong
ECE Department
Carnegie Mellon University
tinawong@cmu.edu

ABSTRACT

Routing policy configuration is a very important aspect of network operations because it affects the network's profit, performance and security. Network operators implement low-level routing policies according to their high-level objectives. In this paper, we propose a set of techniques for analyzing network-wide routing policies. First, we interpret the routing policies relevant to a single neighbor. Then, we classify all neighbors into groups which express common intent. Classification is done by generating and comparing *update patterns*. We validate our approach by experimenting with the router configuration files of a Tier-1 ISP. Our techniques classify neighbors according to their type (customer/peer/transit), highlight neighbors which deviate from the norm and reveal possible mistakes. Consequently, our network-wide analysis seems to be promising for automating the translation of routing policy configuration into initial intent.

Categories and Subject Descriptors

C.2.3 [**Computer-Communication Networks**]: Network Operations – *Network Management*

General Terms

Management

Keywords

Routing policies, Router configuration, BGP

1. INTRODUCTION

Among the various tasks that a network operator performs is the routing policy configuration of the network's routers. Currently, network operators translate their set of high-level objectives into low-level statements with the guidance of the configuration manuals of the router vendor. However, there is no efficient way for network operators to "decode" the configuration files into their initial intentions. After configuring a router, network operators mostly depend on their memory in order to extract the intention behind their configuration choices. Also, given the turnover of engineering personnel understanding the enforced routing policies of a network by simply reading the configuration files is a difficult and error-prone task. It would be extremely useful to develop an analysis that provides a high-level view of the routing policies enforced in a network and validates intentions with low-level implementations.

INM'07, August 27–31, 2007, Kyoto, Japan.

In this paper, we propose a set of techniques for network-wide routing policy analysis. We focus on routing policies supported by the Border Gateway Protocol (BGP), the dominant path-vector protocol that implements policy-based routing. BGP supports a number of routing policy mechanisms, i.e. route-maps, prefix-lists and filter-lists. Given these low-level mechanisms, we analyze the routing policies affecting a single neighbor. We represent the impact of the routing policy mechanisms with *update patterns* and proceed to rule-based mapping in order to extract the intention. We continue our analysis by developing a hierarchical model of neighbor groups which represent distinct intents. Our techniques have been implemented for routers running CISCO IOS but our approach is applicable to other types of routers. To validate our approach, we experiment with the routing policy configuration of a Tier-1 ISP. Our results reveal the network's common practices in a compact way. Finally, there is significant relevant work [1,2,3,4] which we have to omit due to space limitations.

2. ROUTING POLICY ANALYSIS

2.1 Per Neighbor Routing Policy Analysis

In order to represent the routing policy actions affecting the update messages originating from one neighbor (*import neighbor*) until they are exported to another neighbor (*export neighbor*), we define the *update pattern*. The update pattern summarizes the characteristics of an update message once it has passed through a routing policy component. The update pattern has six attributes: the update attributes which are mostly manipulated by routing policy components (i.e. advertised prefix, AS path, communities, local preference and Multi-Exit-Discriminator) and the number of times that the AS path has been prepended. We add this last attribute because prepending an AS path expresses a different intent than filtering an AS path. Figure 1 illustrates an example of an *update's journey* through two routing policy components.

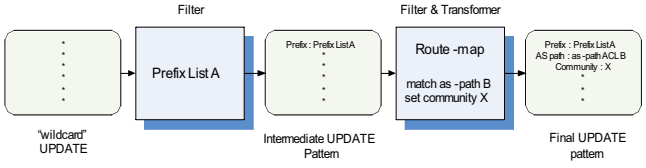

Figure 1. Example: update's journey through two routing policy components.

Obviously, an update's journey is determined by BGP's route selection process which in turn is determined by the static BGP configuration of the network. Previous work presents a model that predicts the outcome of the BGP route selection process in a single AS [4]. We are currently not using this model in our

routing policy analysis because dynamic data are not available. However, it can be easily incorporated once such data are available.

Given an import neighbor we generate the final update patterns exported to all export neighbors. Our goal is to interpret the meaning of the final update patterns concentrating on common routing policy practices as mentioned in [1]. Our intention analysis is based on hard-coded mappings from low-level commands to high-level objectives. We define interpretations, such as the more times the AS path of an update is prepended, the less preferred is the route advertised by this update. Given these interpretations, which we omit due to space limitations, we extract the meaning of each attribute of an update pattern.

2.2 Network-wide Routing Policy Analysis

The proposed network-wide analysis is based on a hierarchical model of neighbor groups. The representation of routing policy behaviors in different granularities, i.e. levels, makes this model particularly expressive. Generic groups populate the first level of the model. Two import neighbors belong in the same generic group when the network exhibits similar export routing policy behavior towards both of them. Intuitively, this should be the case for two customers whose advertisements are treated similarly by the network. A generic group is a group of neighbors with the same *export groups* in terms of membership.

An export group (EG) of an import neighbor is a group of export neighbors with similar characteristics. It represents a distinct intention on behalf of the network operator for updates originating from the import neighbor and advertised to the members of the group. Similar update patterns express a single intention. Therefore, we generate the export groups by comparing the final update patterns of all export neighbors. Specifically, two export neighbors belong in the same export group when the following update pattern attributes are identical: prefix, AS path, communities, prepending. Local preference is not included because it does not express any intention at the export side. We also exclude the MED value because it is usually set to a different value for each export neighbor. Therefore, it would prevent the forming of groups.

However, even if two neighbors belong to the same generic group, this does not mean that the intention expressed by each of their export groups is the same. For instance, two customers, *A* and *B*, share the members of an export group. But updates from A get prepended when exported to the members of the export group, while updates from *B* do not. This is possible if updates from *A* carry a community value which acts as a flag for prepending. For this reason, the second level of the hierarchy differentiates the impact of the import routing policies from that of the export routing policies in the generic group formation. The subgroups of a generic group are populated by the members of a generic group which belong to the same *import group*.

An import group (IG) is a group of import neighbors with similar characteristics. It represents a distinct intention on behalf of the network for updates originating from the members of the group. We obtain the import update pattern by feeding the "wildcard" update to the import policy of each neighbor. Specifically, two import neighbors belong in the same import group when the following update pattern attributes are identical: local preference, communities, prepending. The advertised prefix or AS path are often manipulated by import routing policies in order to perform ingress filtering. In this case they are neighbor-specific and they cannot be used for grouping because we would get as many import groups as neighbors. In the case where they are not neighbor-specific (most likely one of the two will not be), they are used for comparing import update patterns. The MED value is again excluded as already mentioned.

3. EVALUATION

In order to check the validity of our approach we experimented with a snapshot of a Tier-1 ISP's router configuration files. The 13 routers of this network run CISCO IOS and "speak" BGP. The number of neighbors with distinct import and export routing policies is 64. The results of our analysis are summarized in Table 1. Our approach synopsizes the network-wide routing policies of this ISP in a meaningful and compact way which can be directly used by network operators. The groups expressing common practices have many members. The deviant neighbors belong to groups with few members and thus, they can be the result of misconfiguration. We are currently talking with the ISP's network operators in order to verify our results. Finally, our analysis highlights the differences in routing policy configuration between customers, peers and transits.

Table 1. Results from routing policy analysis of a Tier-1 ISP

GG description	#members	#EG	#IG
transits + peers	31	5	12
customers connected to router X	4	10	3
customer A (multiple sessions)	5	12	2
customer B (multiple sessions)	4	17	1
rest of the customers	10	10	4
imported updates all dropped	10	-	-

4. CONCLUSIONS

In this paper we propose a set of techniques for network-wide routing policy analysis with the goal of extracting the network's high-level objectives. Our analysis is based on the similarity of update patterns. First, we analyze the routing policies relevant to a single neighbor. Then, we develop a hierarchical model of neighbor groups which represent different routing policies. The characteristics of each group can be mapped to a high-level intent. This is ongoing work and initial results seem promising towards the automatic mapping of low-level commands to high-level intents.

5. REFERENCES

[1] M. Caesar and J. Rexford. BGP routing Policies in ISP networks. UC Berkeley Technical report, UCB/ CSD-05-1377, 2005.

[2] N. Feamster. Practical Verification Techniques for Wide-Area Routing. In Proc. of HotNets, 2003.

[3] N. Feamster and H. Balakrishnan. Detecting BGP configuration faults with static analysis. In Proc. of NSDI, 2005.

[4] N. Feamster, J. Winick, and J. Rexford. A model of BGP routing for network engineering. In Proc. of ACM SIGMETRICS, June 2004.

[5] G. Xie, et al. Routing design in operational networks: a look from the inside. In Proc. of SIGCOMM, 2004.