Utilizing Semantic Policies for Managing BGP Route Dissemination

Palanivel Kodeswaran, Sethuram Balaji Kodeswaran, Anupam Joshi
Department of Computer Science and Electrical Engineering
University of Maryland, Baltimore County
Baltimore, MD 21250, USA
Email:{palanik1,kodeswar,joshi}@cs.umbc.edu

Filip Perich
Shared Spectrum Company
Vienna, VA 22182, USA
Email: fperich@sharedspectrum.com

***Abstract*—Policies in BGP are implemented as routing configurations that determine how route information is shared among neighbors to control traffic flows across networks. This process is generally template driven, device centric, limited in its expressibility, time consuming and error prone which can lead to configurations where policies are violated or there are unintended consequences that are difficult to detect and resolve. In this paper, we propose an alternate mechanism for policy based networking that relies on using additional semantic information associated with routes expressed in an OWL ontology. Policies are expressed using SWRL to provide fine-grained control where by the routers can reason over their routes and determine how they need to be exchanged. In this paper, we focus on security related BGP policies and show how our framework can be used in implementing them. Additional contextual information such as affiliations and route restrictions are incorporated into our policy specifications which can then be reasoned over to infer the correct configurations that need to be applied, resulting in a process which is easy to deploy, manage and verify for consistency.**

I. Introduction

Border Gateway Protocol (BGP) was originally designed as a simple path vector protocol to share routing information between autonomous systems (AS) which has today, become the de-facto inter-domain routing protocol enabling the Internet. Autonomous systems (ISPs, enterprises etc) use policies, which are driven by various factors such as commercial peering agreements, security considerations, load balancing requirements etc., to define how the routes are to be shared and among which peers. These policies are then implemented in the network routers as configuration parameters that control the protocol behavior. One of the main challenges is in ensuring that network configuration settings are applied consistently throughout the network so that the correct actions are taken by the network devices both within an autonomous system and across boundaries. Current approaches to configuring BGP routers are operator dependent, device centric, and do not consider overall network objectives. Even under fairly static organizational policies, BGP misconfiguration has been the major cause for internet outage in recent years. Furthermore since routes are expressed as mere IP prefixes such as 127.0/16 with no additional metadata, there is an inherent inflexibility in specifying high level policies such as “Share route with tier I partners”. Furthermore, implementing these configuration changes requires time and highly skilled personnel and is not suitable for scenarios such as emergency response and army battlefield operations where minimizing deployment time and complexity is vital.

In this paper, we address the problem of secure route exchange among peers in a battlefield scenario where deployment time is critical and there are no guarantees of skilled network operators. We propose an alternate model to achieve policy based routing that can provide fine grained policy specification to automate network configuration and ease network management. Specifically, in this paper, we focus on import and export policies concerning route exchange among peers belonging to different Autonomous Systems (ASes). The model relies on two key components; namely a tagging mechanism that allows routes to convey higher level semantic information that can be used in conjunction with information about the participating BGP peers and a framework for specifying rules in an easy to use, formal model that can be checked for consistency. In our model, ASes encode routes that they originate with descriptions conveying semantics such as what this route represents, who this route can be shared with, traffic type limitations etc using RDF/OWL as a special option and transitive path attribute in BGP. Our motivation for using OWL [1] (specifically, OWL-DL), besides being a W3C standard, is mainly its capabilities for expressing formal semantics, defining class hierarchies and their relationships, associated properties, cardinality restrictions while still retaining decidability and computational completeness. Using OWL for ontology specification makes the framework generic, flexible and more scalable than using proprietary labeling schemes that raise interoperability issues.

Utilizing the framework, BGP speakers can run a reasoning engine that can reason over the RDF descriptions of the various routes and invoke rules depending on the correct set of actions that need to be enforced. Our framework utilizes SWRL [2] as the rule language which provides an easy to use mechanism for specifying event-condition-action rules which constitute the majority of rules envisioned for a typical network. Using this framework, we can control route exchanges at a finer granularity that also enables us to control the traffic flowing in the network.

We show how our architecture can be used to provide fine grained levels of control that is simple to implement and

easy to verify for correctness. We have developed a network ontology to be used to describe BGP protocol packets with attributes to describe the route meta-data and example policies to fine tune the BGP decision process.We have also developed a simulation toolkit in NS2 to implement aspects of our proposed architecture allowing us to simulate various scenarios and how policies can be expressed to offer desired behavior.

The rest of the paper is organized as follows. Section II describes related work, In section III we describe our policy based network and in section IV, we describe a typical use case and our simulation toolkit. We finally conclude in section V.

II. Related Work

There has been significant work in understanding and automating BGP network management. However, most of these approaches have been at the individual device level in terms of creating the right configuration files, and do not consider the high level network objectives. [3] provides a detailed description of the use of BGP policies in ISP networks. In an effort to better understand the dynamics of configuration management, [4] use a combination of TACAS logs, static configuration files and the router configuration of a Tier-1 ISP to build a Deterministic Finite Automaton (DFA) representation of network configuration. In the constructed DFA, each state represents the configured state of a router interface and the edges represent operations performed on the interface. All the router configuration commands are timestamped and accounted by the TACAS logs and thereby provide a chronological view of network configuration changes, which are captured by the constructed DFA. Also, the authors use the data to find correlated events in the network. [5] uses static analysis to detect configuration faults in BGP. Particularly, the authors derive configuration constraints from high level policy specifications and check BGP configurations against the derived constraints. Using this approach, they detect path visibility and route validity configuration faults. However, they do not deal with dynamic configuration changes. RPSL[6] is an object oriented language for specifying routing policies from which router configurations can be automatically generated. RPSL generated router configurations can aid in preventing internet router misconfigurations but it does not support inference and is limited in expressibility.

There has been considerable research on securing BGP. SBGP[7] proposes a comprehensive architecture for securing BGP using public key certificates. S-BGP uses a pair of PKIs, one for address authentication and the other for route validation. SoBGP[8] provides more flexibility compared to SBGP. In addition to the above PKIs, a third type of certificate is used which provides routing policy and local topology. When a route is received, it is compared for consistency with the topology database and dropped if found to be inconsistent. The architecture is more flexible as there are no fixed structures of authority and ASes can decide on their own for accepting routing announcements and policies. IRV[9] provides route validation through Interdomain Route Validation (IRV) servers running on the ASes. On receiving an UPDATE message, based on the local policy, the receiver can query the remote IRV server for veracity. The IRV servers in turn can enforce their local policies while responding to queries, thus providing control of data to the AS.

III. Semantics Driven Policy Based Networks

Policy based networks employ mechanisms that allow network operators to specify at a high level, rules defining how packet flows are handled within a network, how network resources are allocated, access control restrictions and levels of service. All these policies are then enforced by configuring the network devices with the requisite primitives so that the desired actions are performed on the data streams. For example, BGP allows specifying policies that decide whether a router can accept a route from a neighboring router or not. In previous work [10], [11], we have proposed an architecture for policy based networks that involves semantically tagging packets (in OWL/RDF) to convey higher level meta data about the content being carried in the packets. This semantic information can then be reasoned over at the network elements to provide specialized services in the network. Our policy based network is a multi-tier system with hierarchical policy enforcement with the highest level of the hierarchy being the central NOC for an ISP and the lowest level being an adaptation layer that is responsible for translating the high level policies into low level protocol specific configuration routines that can be applied to the various network elements being managed.

In this work, we have adapted the above framework to handle BGP interactions and use it to specify routing policies. We limit our discussion to how the various components of our general architecture work to drive the BGP decision process. More details on the architecture itself is available in [10], [11].

The *Network Ontology* (NetOnto) is the OWL ontology that we define to mark up the routes being exchanged. By using OWL rather than simple XML, the language is semantically richer and highly extensible which is very important especially when we have interdomain interactions (such as peering arrangements, SLAs etc). Policies are written using the concepts defined in NetOnto using SWRL as the rule language. OWL has axiomatic and model-theoretic semantics, which allows for verification of knowledge expressed in OWL constructs. OWL + SWRL can be used to define ontologies, using which one can declaratively define facts, policies, and rules in terms of what needs to be true or false for a policy to hold. The route descriptions are carried in the BGP updates as optional transitive attributes either as directly embedded in a bit efficient format, or contain a URL to the description or use UUIDs that imply a certain well known description. A Policy Enforcement Point (PEP) extracts this description and adds to it, any extra contextual information including aspects such as peer identity, network state (congestion, link failures etc), network technology (wired, hybrid, MANET, cellular) etc. This information is then sent to the Policy Decision Point (PDP) for reasoning. The response back from the PDP will cause specific configurations to be installed by the PEP on the

device (in this work, as we are dealing with import/export policies, the PDP filters appropriately the routes that are exchanged).

IV. Securing BGP through Route Filtering - A Use Case

In this section we describe how our framework can be used to secure BGP route exchange through appropriate import and export policies. To apply the above framework to provide BGP route dissemination that takes into account the security credentials and external relationships, we needed to make two modifications to the protocol. The first modification is aimed at establishing the identity of the BGP peers in a secure and verifiable manner. For this purpose, we assume the BGP session establishment process is extended to include the sharing of signed credentials to validate the identity of the BGP peers and their affiliations. Prior work such as S-BGP [7] have shown that this is feasible using a public key infrastructure and signed certificates. This modification is necessary as it is important for a BGP router to establish the identity of its peer so that the routes learned from and advertised to this peer can be handled correctly. The second modification is to include with the route advertisement in the BGP update messages, an additional optional and transitive attribute that conveys semantic meta-data about that NLRI. The intent here is for the originating AS to provide this meta-data so that other nodes can handle the route appropriately. The interim routers are also allowed to add to this description as necessary (keeping the original intact) in a manner that is secure and cannot be repudiated. In this work, we are concerned about the import/export policies in use in the BGP decision process. The modifications allow nodes in our framework to, for each route that is being advertised to or learned from, contact a PDP that will reason over the semantic information provided for that route and the policies that need to be enforced, and communicate to the node whether or not, the route can be shared or accepted.

1) Use Case: The use case we consider in this paper is that of a secure version of BGP where there are constraints on route exchanges between BGP peers. As with the real Internet, BGP nodes are owned by different agencies that have different affiliations. During the initial session establishment, nodes exchange their identity information to indicate the agencies to which they belong. These agencies or organizations have external socio-economic, political or financial relationships that will influence the BGP nodes in their exchanges. Routes advertised by each AS are tagged with additional semantic information that describe aspects such as its confidentiality, sharing restrictions etc. For such a use case, the following policies would be appropriate:

- Routes marked as "ShareWithFriendly" can only be exchanged between routers that belong to organizations that have a collaborative relationship
- Routes marked as "Restricted" can only be shared between nodes that belong to the same parent organization (even if they are different divisions of that organization)
- Routes marked to be used only for data backup traffic are installed only during non-peak hours
- Allow a route to be used only for data traffic that has a specified or higher clearance level.

2) Simulation Toolkit: We used the ns-BGP [12] extension to NS2 to implement our framework. The network topology considered is a linear network as shown in Figure 1 with nodes grouped into various ASes. Each node is initialized with credentials that specify what organization the node belongs to. We modified the BGP session establishment process to allow the exchange of these credentials so that the BGP nodes can establish the identity and affiliation of the peers with which they are interacting with. We added an additional optional transitive attribute to the BGP update messages to convey additional semantic information about the route. For the network ontology, we used Protege as the editor for specifying our ontology. Jess was used as the reasoning engine. The choice of Jess was mainly motivated by its easy integration with Protege. Other reasoning engines can be used as a replacement if desired.

To begin, we defined an ontology [13] to use for our BGP example. We modeled the various BGP protocol messages and constructs. Since we are dealing with import/export policies, we modeled special instances of classes representing the various actions that a BGP router (PEP) should take such as whether a route should be advertised or not, whether a route should be accepted or not etc. These special instances contain the low level primitive commands that need to be invoked to realize the necessary behavior. In our case, we implemented handlers in the NS2 implementation to handle the response coming back from the reasoner to determine whether a route should be included in an advertisement or whether a route that was received, should be accepted (these commands are expressed as snippets of Tcl code that are evaluated by NS2).

Using this framework, we implemented our typical use case scenario focusing on the import/export policies for BGP. For our example, we consider a network of four autonomous domains with five BGP routers. The Autonomous Domain AS0 belongs to UK forces. The Autonomous Domains AS1 and AS2 belong to two organizations within the US military. Finally, the last Autonomous Domain AS3 belongs to Russian military. During the initial BGP session establishment, the identity of each of the peers is established. This indicates the organization that the router belongs to (US_{Milcom}, UK_{Milcom}, Russian_{Milcom} etc) which is tracked in the"owner" property of the network devices. Some of these organizations have external relationships (such as NATO to which US_{Milcom} and UK_{Milcom} belong). Such external relationships are modeled through OWL restrictions on properties. For example, a device that is part of NATO is modeled as one where there is a necessary and sufficient constraint that the owner is either an instance of US_{Milcom}, UK_{Milcom} or France_{Milcom}. Each router that originates a route includes a description that at the least, indicates the sharing restrictions for that route. In the current version, we have values such as None (which is similar to the "internet" community attribute in BGP), Restricted and

ShareWithFriendly as examples. The intention here is that a route marked as "ShareWithFriendly" can only be shared with a peer who can be considered friendly. For example, if we considered forces within NATO to be friendly, a SWRL policy to permit the routes marked as "ShareWithFriendly" to be exchanged could be written as:

```
BGP_Update(?adv) ∧
interimRouter(?adv, ?routeadvertising) ∧
dest(?adv, ?peer) ∧
NATO_Forces(?routeadvertising) ∧
NATO_Forces(?peer) ∧
routeRestriction(?adv, ?restriction) ∧
ShareWithFriendly(?restriction) ∧
AllowRouteAdvertisement(?allow)
    → inferredAction(?adv, ?allow)
```

Once the simulation starts, each router advertises its routes with its peers in order to compute its routing table. The simulation proceeds until all routes are computed and the routers converge on their tables. Note that when two routers belonging to UK_{Milcom} and US_{Milcom} (AS0 and AS1) are in a BGP session and while none of the routers have explicitly been identified as belonging to NATO, the reasoner can deduce this relationship and allow route exchanges between them. Similarly the reasoner can deduce that the route exchange cannot be allowed between AS2 and AS3 as they do not have an explicit relationship that permits this. Figure 2 is a snapshot of the system with the nodes contacting the reasoner to determine if routes can be exchanged and the responses received.

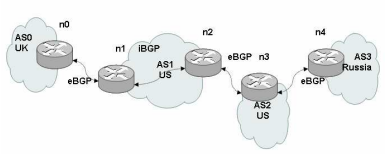

Fig. 1. Topology

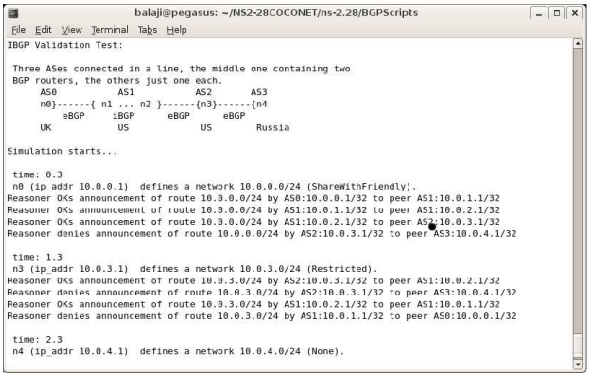

Fig. 2. Simulation Output

In this manner, we can now setup arbitrary relationships between routers and can specify policies through higher level rule based mechanisms to implement fine grained control over the protocol. This example can be easily extended to scenarios where the relationships are short lived and arbitrary such as in emergency response scenarios (where organizations may temporarily want to share information for providing quick response), application need driven (such as for supporting live event feeds) etc. by extending on the ontology and defining the desired policies.

V. Conclusion

In this paper we have proposed a new approach for specifying and managing BGP Routing policies. We build on top of our semantically tagged policy based network and show how our approach can be used to manage BGP route filtering policies. Our approach varies from traditional device centric approaches in that we can now focus on the high level policies while the network reconfigures itself automatically. We have developed and implemented our system in a simulation framework that validates our approach.

Acknowledgment

This work was supported in part by DARPA under contract W31P4Q-06-C-0395.

References

[1] D. L. McGuinness and F. van Harmelen, "Owl web ontology language overview," W3C Recommendation 10 February 2004, Tech. Rep., 2004. [Online]. Available: http://www.w3.org/TR/owl-features/

[2] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean, "Swrl: A semantic web rule language combining owl and ruleml," W3C Member submission 21 may 2004, Tech. Rep., 2004. [Online]. Available: http://www.w3.org/Submission/SWRL/

[3] M. Caesar and J. Rexford, "Bgp routing policies in isp networks," 2005. [Online]. Available: citeseer.ist.psu.edu/caesar05bgp.html

[4] X. Chen, Z. M. Mao, and J. van der Merwe, "Towards automated network management: network operations using dynamic views," in *INM '07: Proceedings of the 2007 SIGCOMM workshop on Internet network management*. New York, NY, USA: ACM, 2007, pp. 242–247.

[5] N. Feamster and H. Balakrishnan, "Detecting bgp configuration faults with static analysis," in *NSDI'05: Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation*. Berkeley, CA, USA: USENIX Association, 2005, pp. 43–56.

[6] C. Alaettinoglu, C. Villamizar, E. Gerich, D. Kessens, D. Meyer, T. Bates, D. Karrenberg, and M. Terpstra, "Routing Policy Specification Language (RPSL)," Internet Engineering Task Force: RFC 2622, June 1999.

[7] S. Kent, C. Lynn, and K. Seo, "Secure border gateway protocol(s-bgp)," *IEEE Journal on Selected Areas in Communication*, vol. 18, pp. 582–592, 2000.

[8] "Secure Origin BGP (SoBGP) Certificates. Internet Research Task Force, June 2003. (draft-weis-sobgp-certificates-00.txt)."

[9] G. Goodell, W. Aiello, T. Griffin, J. Ioannidis, P. McDaniel, and A. Rubin, "Working around bgp: An incremental approach to improving security and accuracy in interdomain routing," in *Proc. ISOC Network and Distributed System Security (NDSS '03)*, San Diego, CA, February 2003. [Online]. Available: citeseer.ist.psu.edu/goodell03working.html

[10] S. B. Kodeswaran, O. Ratsimor, A. Joshi, and F. Perich, "Utilizing Semantic Tags for Policy Based Networking," in *Globecom 2007 (accepted for publication)*, November 2007.

[11] S. B. Kodeswaran and A. Joshi, "Content and context aware networking using semantic tagging," in *ICDEW '06: Proceedings of the 22nd International Conference on Data Engineering Workshops (ICDEW'06)*. Washington, DC, USA: IEEE Computer Society, 2006, p. 77.

[12] T. Feng, R. Ballantyne, and L. Trajkovic, "Implementation of bgp in a network simulator," in *Proc. Applied Telecommunications Symposium, ATS'04*, April 2004, pp. 149–154.

[13] "http://www.cs.umbc.edu/ kodeswar/ontologies/NetworkOnto.owl." [Online]. Available: http://www.cs.umbc.edu/ kodeswar/ontologies/NetworkOnto.owl