IDR Working Group
Internet-Draft
Expires: February 15, 2008

P. Marques
N. Sheth
R. Raszuk
Juniper Networks
B. Greene
Cisco Systems, Inc.
J. Mauch
NTT/Verio
D. McPherson
Arbor Networks

August 14, 2007

Dissemination of flow specification rules
draft-ietf-idr-flow-spec-00

Status of this Memo

Copyright Notice

Abstract

This document defines a new BGP NLRI encoding format that can be used to distribute traffic flow specifications. This allows the routing system to propagate information regarding more-specific components of the traffic aggregate defined by an IP destination prefix.

Additionally it defines two applications of that encoding format. One that can be used to automate inter-domain coordination of traffic filtering, such as what is required in order to mitigate (distributed) denial of service attacks. And a second application to traffic filtering in the context of a BGP/MPLS VPN service.

The information is carried via the Border Gateway Protocol (BGP), thereby reusing protocol algorithms, operational experience and administrative processes such as inter-provider peering agreements.

Table of Contents

1. Introduction

Modern IP routers contain both the capability to forward traffic according to aggregate IP prefixes as well as to classify, shape, limit filter or redirect packets based on administratively defined policies.

While forwarding information is, typically, dynamically signaled across the network via routing protocols, there is no agreed upon mechanism to dynamically signal flows across autonomous-systems.

For several applications, it may be necessary to exchange control information pertaining to aggregated traffic flow definitions which cannot be expressed using destination address prefixes only.

An aggregated traffic flow is considered to be an n-tuple consisting of several matching criteria such as source and destination address prefixes, IP protocol and transport protocol port numbers.

The intention of this document is to define a general procedure to encode such flow specification rules as a BGP [2] NLRI which can be reused for several different control applications. Additionally, we define the required mechanisms to utilize this definition to the problem of immediate concern to the authors: intra and inter provider distribution of traffic filtering rules to filter (Distributed) Denial of Service (DoS) attacks.

By expanding routing information with flow specifications, the routing system can take advantage of the ACL/firewall capabilities in the router's forwarding path. Flow specifications can be seen as more specific routing entries to an unicast prefix and are expected to depend upon the existing unicast data information.

A flow specification received from a external autonomous-system will need to be validated against unicast routing before being accepted. If the aggregate traffic flow defined by the unicast destination prefix is forwarded to a given BGP peer, then the local system can safely install more specific flow rules which result in different forwarding behavior, as requested by this system.

The choice of BGP as the carrier of this control information is also justifiable by the fact that the key issues in terms of complexity are problems which are common to unicast route distribution and have already been solved in the current environment.

From an algorithmic perspective, the main problem that presents itself is the loop-free distribution of <key, attribute> pairs from one originator to N ingresses. The key, in this particular instance,

being a flow specification.

From an operational perspective, the utilization of BGP as the carrier for this information, allows a network service provider to reuse both internal route distribution infrastructure (e.g.: route reflector or confederation design) and existing external relationships (e.g.: inter-domain BGP sessions to a customer network).

While it is certainly possible to address this problem using other mechanisms, the authors believe that this solution offers the substantial advantage of being an incremental addition to deployed mechanisms.

2. Flow specifications

A flow specification is an n-tuple consisting on several matching criteria that can be applied to IP traffic. A given IP packet is said to match the defined flow if it matches all the specified criteria.

A given flow may be associated with a set of attributes, depending on the particular application, such attributes may or may not include reachability information (i.e. NEXT_HOP). Well-known or AS-specific community attributes can be used to encode a set of predeterminate actions.

A particular application is identified by a specific (AFI, SAFI) pair [3] and corresponds to a distinct set of RIBs. Those RIBs should be treated independently from each other in order to assure non-interference between distinct applications.

BGP itself treats the NLRI as an opaque key to an entry in its databases. Entries that are placed in the Loc-RIB are then associated with a given set of semantics which is application dependent. This is consistent with existing BGP applications. For instance IP unicast routing (AFI=1, SAFI=1) and IP multicast reverse-path information (AFI=1, SAFI=2) are handled by BGP without any particular semantics being associated with them until installed in the Loc-RIB.

Standard BGP policy mechanisms, such as UPDATE filtering by NLRI prefix and community matching, SHOULD apply to the newly defined NLRI-type. Network operators can also control propagation of such routing updates by enabling or disabling the exchange of a particular (AFI, SAFI) pair on a given BGP peering session.

3. Dissemination of Information

We define a "Flow Specification" NLRI type that may include several components such as destination prefix, source prefix, protocol, ports, etc. This NLRI is treated as an opaque bit string prefix by BGP. Each bit string identifies a key to a database entry which a set of attributes can be associated with.

This NLRI information is encoded using MP_REACH_NLRI and MP_UNREACH_NLRI attributes as defined in RFC4760 [3]. Whenever the corresponding application does not require Next Hop information, this shall be encoded as a 0 octet length Next Hop in the MP_REACH_NLRI attribute and ignored on receipt.

The NLRI field of the MP_REACH_NLRI and MP_UNREACH_NLRI is encoded as a 1 or 2 octet NLRI length field followed by a variable length NLRI value. The NLRI length is expressed in octets.

```
+-------------------------------+
|    length (0xnn or 0xfn nn)   |
+-------------------------------+
|    NLRI value  (variable)     |
+-------------------------------+
```

flow-spec NLRI

If the NLRI length value is smaller than 240 (0xf0 hex), the length field can be encoded as a single octet. Otherwise, it is encoded as a extended length 2 octet value in which the most significant nibble of the first byte is all ones.

The Flow Specification NLRI-type consists of several optional subcomponents. A specific packet is considered to match the flow specification when it matches the intersection (AND) of all the components present in the specification.

The following component types are defined:

Type 1 - Destination Prefix

Encoding: <type (1 octet), prefix length (1 octet), prefix>

Defines the destination prefix to match. Prefixes are encoded as in BGP UPDATE messages, a length in bits is followed by enough octets to contain the prefix information.

Type 2 - Source Prefix

Encoding: <type (1 octet), prefix-length (1 octet), prefix>

Defines the source prefix to match.

Type 3 - IP Protocol

Encoding: <type (1 octet), [op, value]+>

Contains a set of {operator, value} pairs that are used to match IP protocol value byte in IP packets.

The operator byte is encoded as:

```
  7   6   5   4   3   2   1   0
+---+---+---+---+---+---+---+---+
| e | a |  len  | 0 |lt |gt |eq |
+---+---+---+---+---+---+---+---+

          Numeric operator
```

+ End of List bit. Set in the last {op, value} pair in the list.

+ And bit. If unset the previous term is logically ORed with the current one. If set the operation is a logical AND. It should be unset in the first operator byte of a sequence. The AND operator has higher priority than OR for the purposes of evaluating logical expressions.

+ The length of value field for this operand is given as (1 << len).

+ Lt - less than comparison between data and value.

+ gt - greater than comparison between data and value.

+ eq - equality between data and value.

The bits lt, gt, and eq can be combined to produce "less or equal", "greater or equal" and inequality values.

Type 4 - Port

Encoding: <type (1 octet), [op, value]+>

Defines a list of {operation, value} pairs that matches source OR destination TCP/UDP ports. This list is encoded using the numeric operand format defined above. Values are encoded as 1 or 2 byte quantities.

Type 5 - Destination port

Encoding: <type (1 octet), [op, value]+>

Defines a list of {operation, value} pairs used to match the destination port of a TCP or UDP packet. Values are encoded as 1 or 2 byte quantities.

Type 6 - Source port

Encoding: <type (1 octet), [op, value]+>

Defines a list of {operation, value} pairs used to match the source port of a TCP or UDP packet. Values are encoded as 1 or 2 byte quantities.

Type 7 - ICMP type

Encoding: <type (1 octet), [op, value]+>

Defines a list of {operation, value} pairs used to match the type field of an icmp packet. Values are encoded using a single byte.

Type 8 - ICMP code

Encoding: <type (1 octet), [op, value]+>

Defines a list of {operation, value} pairs used to match the code field of an icmp packet. Values are encoded using a single byte.

Type 9 - TCP flags

Encoding: <type (1 octet), [op, bitmask]+>

Bitmask values are encoded using a single byte, using the bit definitions specified in the TCP header format [1].

This type uses the bitmask operand format, which differs from the numeric operator format in the lower nibble.

```
  7   6   5   4   3   2   1   0
+---+---+---+---+---+---+---+---+
| e | a |  len  | 0 | 0 |not| m |
+---+---+---+---+---+---+---+---+
```

+ Top nibble: (End of List bit, And bit and Length field), as defined for in the numeric operator format.

+ Not bit. If set, logical negation of operation.

+ Match bit. If set this is a bitwise match operation defined as "(data & value) == value"; if unset (data & value) evaluates to true if and of the bits in the value mask are set in the data.

Type 10 - Packet length

Encoding: <type (1 octet), [op, value]+>

Match on the total IP packet length (excluding L2 but including IP header). Values are encoded using as 1 or 2 byte quantities.

Type 11 - DSCP

Encoding: <type (1 octet), [op, value]+>

Defines a list of {operation, value} pairs used to match the IP TOS octet.

Type 12 - Fragment

Encoding: <type (1 octet), [op, bitmask]+>

Uses bitmask operand format defined above.

Bitmask values:

+ Bit 0 - Dont fragment

+ Bit 1 - Is a fragment

+ Bit 2 - First fragment

+ Bit 3 - Last fragment

Flow specification components must follow strict type ordering. A given component type may or may not be present in the specification,

but if present it MUST precede any component of higher numeric type value.

If a given component type within a prefix in unknown, the prefix in question cannot be used for traffic filtering purposes by the receiver. Since a Flow Specification as the semantics of a logical AND of all components, if a component is FALSE by definition it cannot be applied. However for the purposes of BGP route propagation this prefix should still be transmitted since BGP route distribution is independent on NLRI semantics.

Flow specification components are to be interpreted as a bit match at a given packet offset. When more than one component in a flow specification tests the same packet offset the behavior is undetermined.

The <type, value> encoding is chosen in order to account for future extensibility.

An example of a Flow Specification encoding for: "all packets to 10.0.1/24 and TCP port 25".

destination	proto	port
0x01 18 0a 00 01	03 81 06	04 81 19

Decode for protocol:

Value		
0x03	type	
0x81	operator	end-of-list, value size=1, =
0x06	value	

An example of a Flow Specification encoding for: "all packets to 10.0.1/24 from 192/8 and port {range [137, 139] or 8080}".

destination	source	port
0x01 18 0a 01 01	02 08 c0	04 03 89 45 8b 91 1f 90

Decode for port:

Value		
0x04	type	
0x03	operator	size=1, >=
0x89	value	137
0x45	operator	&, value size=1, <=
0x8b	value	139
0x91	operator	end-of-list, value-size=2, =
0x1f90	value	8080

This constitutes a NLRI with an NLRI length of 16 octets.

Implementations wishing to exchange flow specification rules MUST use BGP's Capability Advertisement facility to exchange the Multiprotocol Extension Capability Code (Code 1) as defined in RFC4760 [3]. The (AFI, SAFI) pair carried in the Multiprotocol Extension capability MUST be the same as the one used to identify a particular application that uses this NLRI-type.

4. Traffic filtering

Traffic filtering policies have been traditionally considered to be relatively static.

The popularity of traffic-based denial of service (DoS) attacks, which often requires the network operator to be able to use traffic filters for detection and mitigation, brings with it requirements that are not fully satisfied by existing tools.

Increasingly, DoS mitigation, requires coordination among several Service Providers, in order to be able to identify traffic source(s) and because the volumes of traffic may be such that they will otherwise significantly affect the performance of the network.

Several techniques are currently used to control traffic filtering of DoS attacks. Among those, one of the most common is to inject unicast route advertisements corresponding to a destination prefix being attacked. One variant of this technique marks such route advertisements with a community that gets translated into a discard next-hop by the receiving router. Other variants, attract traffic to a particular node that serves as a deterministic drop point.

Using unicast routing advertisements to distribute traffic filtering information has the advantage of using the existing infrastructure and inter-as communication channels. This can allow, for instance, for a service provider to accept filtering requests from customers for address space they own.

There are several drawbacks, however. An issue that is immediately apparent is the granularity of filtering control: only destination prefixes may be specified. Another area of concern is the fact that filtering information is intermingled with routing information.

The mechanism defined in this document is designed to address these limitations. We use the flow specification NLRI defined above to convey information about traffic filtering rules for traffic that should be discarded.

This mechanism is designed to, primarily, allow an upstream autonomous system to perform inbound filtering, in their ingress routers of traffic that a given downstream AS wishes to drop.

In order to achieve that goal, we define an application specific NLRI identifier (AFI=1, SAFI=133) along with specific semantic rules.

BGP routing updates containing this identifier use the flow specification NLRI encoding to convey particular aggregated flows

that require special treatment.

Flow routing information received via this (afi, safi) pair is subject to the validation procedure detailed bellow.

4.1. Order of traffic filtering rules

With traffic filtering rules, more than one rule may match a particular traffic flow. Thus it is necessary to define the order at which rules get matched and applied to a particular traffic flow. This ordering function must be such that it must not depend on the arrival order of the flow specifications rules and must be constant in the network.

We choose to order traffic filtering rules such that the order of two flow specifications is given by the comparison of NLRI key byte strings as defined by the memcmp() function is the ISO C standard.

Given the way that flow specifications are encoded this results in a flow with a less-specific destination IP prefix being considered less-than (and thus match before) a flow specification with a more-specific destination IP prefix.

This matches an application model where the user may want to define a restriction that affects an aggregate of traffic and a subsequent rule that applies only to a subset of that.

A flow-specification without a destination IP prefix is considered to match after all flow-specifications that contain an IP destination prefix.

5. Validation procedure

Flow specifications received from a BGP peer and which are accepted in the respective Adj-RIB-In are used as input to the route selection process. Although the forwarding attributes of two routes for the same Flow Specification prefix may be the same, BGP is still required to perform its path selection algorithm in order to select the correct set of attributes to advertise.

The first step of the BGP Route Selection procedure (section 9.1.2) is to exclude from the selection procedure routes that are considered non-feasible. In the context of IP routing information this step is used to validate that the NEXT_HOP attribute of a given route is resolvable.

The concept can be extended, in the case of Flow Specification NLRI, to allow other validation procedures.

A flow specification NLRI must be validated such that it is considered feasible if and only if:

a) The originator of the flow specification matches the originator of the best-match unicast route for the destination prefix embedded in the flow specification.

b) There are no more-specific unicast routes, when compared with the flow destination prefix, that have been received from a different neighboring AS than the best-match unicast route, which has been determined in step a).

By originator of a BGP route, we mean either the BGP originator path attribute, as used by route reflection, or the transport address of the BGP peer, if this path attribute is not present.

The underlying concept is that the neighboring AS that advertises the best unicast route for a destination is allowed to advertise flow-spec information that conveys a more or equally specific destination prefix. This, as long as there are no more-specific unicast routes, received from a different neighbor AS, which would be affected by that filtering rule.

The neighboring AS is the immediate destination of the traffic described by the Flow Specification. If it requests these flows to be dropped that request can be honored without concern that it represents a denial of service in itself. Supposedly, the traffic is being dropped by the downstream autonomous-system and there is no added value in carrying the traffic to it.

6. Traffic Filtering Actions

This specification defines a minimum set of filtering actions that it standardizes as BGP extended community values [4]. This is not ment to be an inclusive list of all the possible actions but only a subset that can be interpreted consistently across the network.

Implementations should provide mechanisms that map an arbitrary bgp community value (normal or extended) to filtering actions that require different mappings in different systems in the network. For instance, providing packets with a worse than best-effort per-hop behavior is a functionality that is likely to be implemented differently in different systems and for which no standard behavior is currently known. Rather than attempting to define it here, this can be accomplished by mapping a user defined community value to platform / network specific behavior via user configuration.

The default action for a traffic filtering flow specification is to accept IP traffic that matches that particular rule.

The following extended community values can be used to specify particular actions.

type	extended community	encoding
0x8006	traffic-rate	2-byte as#, 4-byte float
0x8007	traffic-action	bitmask
0x8008	redirect	6-byte Route Target

Traffic-rate The traffic-rate extended community uses the same encoding as the "Link Bandwidth" [4] extended community. The rate is is expressed as 4 octets in IEEE floating point format, units being bytes per second. A traffic-rate of 0 should result on all traffic for the particular flow to be discarded.

Traffic-action The traffic-action extended community consists of 6 bytes of which only the 2 least significant bits of the 6th byte (from left to right) are currently defined.

* Terminal action (bit 0). When this bit is set the traffic filtering engine will apply any subsequent filtering rules (as defined by the ordering procedure). If not set the evaluation of the traffic filter stops when this rule is applied.

* Sample (bit 1). Enables traffic sampling and logging for this flow specification.

Redirect The redirect extended community allows the traffic to be redirected to a VRF routing instance that list the specified route-target in its import policy. If several local instances match this criteria, the choice between them is a local matter (for example, the instance with the lowest Route Distinguisher value can be elected). The traffic marking extended community instruct a system to modify the DSCP bits of a transiting IP packet to the corresponding value. This extended community is encoded as a sequence of 5 zero bytes followed by the DSCP value.

7. Traffic filtering in RFC2547bis networks

Provider-based layer 3 VPN networks, such as the ones using an BGP/ MPLS IP VPN [5] control plane, have different traffic filtering requirements than internet service providers.

In these environments, the VPN customer network often has traffic filtering capabilities towards their external network connections (e.g. firewall facing public network connection). Less common is the presence of traffic filtering capabilities between different VPN attachment sites. In an any-to-any connectivity model, which is the default, this means that site to site traffic is unfiltered.

In circumstances where a security threat does get propagated inside the VPN customer network, there may not be readily available mechanisms to provide mitigation via traffic filter.

This document proposes an additional BGP NLRI type (afi=1, safi=134) value, which can be used to propagate traffic filtering information in a BGP/MPLS VPN environment.

The NLRI format for this address family consists of a fixed length Route Distinguisher field (8 bytes) followed by a flow specification, following the encoded defined in this document. The NLRI length field shall includes the both 8 bytes of the Route Distinguisher as well as the subsequent flow specification.

Propagation of this NLRI is controlled by matching Route Target extended communities associated with the BGP path advertisement with the VRF import policy, using the same mechanism as described in "BGP/ MPLS IP VPNs" [5] .

Flow specification rules received via this NLRI apply only to traffic that belongs to the VRF(s) in which it is imported. By default, traffic received from a remote PE is switched via an mpls forwarding decision and is not subject to filtering.

Contrary to the behavior specified for the non-VPN NLRI, flow rules are accepted by default, when received from remote PE routers.

8. Monitoring

Traffic filtering applications require monitoring and traffic statistics facilities. While this is an implementation specific choice, implementations SHOULD provide:

- o A mechanism to log the packet header of filtered traffic,

- o A mechanism to count the number of matches for a given Flow Specification rule.

9. Security considerations

Inter-provider routing is based on a web of trust. Neighboring autonomous-systems are trusted to advertise valid reachability information. If this trust model is violated, a neighboring autonomous system may cause a denial of service attack by advertising reachability information for a given prefix for which it does not provide service.

As long as traffic filtering rules are restricted to match the corresponding unicast routing paths for the relevant prefixes, the security characteristics of this proposal are equivalent to the existing security properties of BGP unicast routing.

Where it not the case, this would open the door to further denial of service attacks.

10. Acknowledgments

The authors would like to thank Yakov Rekhter, Dennis Ferguson and Chris Morrow for their comments.

Chaitanya Kodeboyina helped design the flow validation procedure.

Steven Lin and Jim Washburn ironed out all the details necessary to produce a working implementation.

11. Normative References

[1] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, September 1981.

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

[3] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, January 2007.

[4] Sangli, S., Tappan, D., and Y. Rekhter, "BGP Extended Communities Attribute", RFC 4360, February 2006.

[5] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, February 2006.

Authors' Addresses

Pedro Marques
Juniper Networks
1194 N. Mathilda Ave.
Sunnyvale, CA 94089
US

Email: roque@juniper.net

Nischal Sheth
Juniper Networks
1194 N. Mathilda Ave.
Sunnyvale, CA 94089
US

Email: nsheth@juniper.net

Robert Raszuk
Juniper Networks
1194 N. Mathilda Ave.
Sunnyvale, CA 94089
US

Email: raszuk@juniper.net

Barry Greene
Cisco Systems, Inc.
170 West Tasman Dr
San Jose, CA 95134
US

Email: bgreene@cisco.com

Jared Mauch
NTT/Verio
8285 Reese Lane
Ann Arbor, MI 48103-9753
US

Danny McPherson
Arbor Networks

Email: danny@arbor.net

Full Copyright Statement

Intellectual Property

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).