Internet Engineering Task Force
Internet-Draft
Intended status: Experimental
Expires: March 3, 2016

W. Eddy
J. Dailey
G. Clark
MTI Systems

August 31, 2015

Experimental BGP Flow Specification Extensions
draft-eddy-idr-flowspec-exp-00

Abstract

This document discusses new extensions beyond the existing BGP mechanisms that support mitigation of Distributed Denial of Service (DDoS) attacks. The new extensions are focused on enabling an increase in collaborative inter-domain defenses involving multiple network providers, and enhancing the ability to describe desired filtering rules and actions. This document is primarily intended for discussion, and later on some ideas contained within it may be exported into other documents or specifications. In some cases, simple examples and proof-of-concept protocol mechanisms are described, but this document is not intended to become a standard or final protocol specification.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 3, 2016.

Table of Contents

1. Introduction

There is a long history of using BGP as a means to trigger filtering at upstream routers to defend against Distributed Denial of Service (DDoS) attacks. Destination-based Remote Triggered Black Hole (RTBH) Filtering is described in [RFC3882]. An advancement of the RTBH approach that additionally supports source-based filtering relying on unicast Reverse Path Forwarding (uRPF) support in routers is described in [RFC5635].

RTBH techniques for DDoS mitigation partly are motivated by the ability to use a router's packet forwarding hardware rather than its access control lists (ACL) features or other filtering mechanisms. This is desirable, because on some systems, use of ACLs can negatively impact performance, and operators may have a more difficult time in managing ACLs in comparison to routes. The downsides of RTBH techniques in comparison to alternatives like ACLs are that they can be slightly complex to setup and manage (though RFC 5635 is a very good guidebook for this), and are limited to making

decisions based on source and destination address fields. Additionally, RTBH mechanisms work within an attacked network or within an access provider's network used by a DDoS attack victim. On its own, RTBH can't directly operate further upstream closer to attack sources (where filtering would be most effective).

The Flow Specification (flowspec) [RFC5575] extension to BGP augments the information conveyed through BGP to better facilitate the use of ACLs and support more complex traffic signatures than just the destination and possibly source addresses that RTBH utilizes. This can somewhat reduce the management burden involved with ACLs, though there may still be performance impacts. Regardless, the flowspec mechanism offers a powerful means of signalling undesired traffic signatures upstream, where there is a possibility that packets can be more effectively filtered. There are basic validation procedures defined for processing messages containing flowspec entries, though these do require some trust as flowspec entries are not possible to cryptographically verify back to the origin AS. Presently, fielded systems such as UTRS [Kristoff15] rely on heuristics, such as checking for a 30-day history of originating routes, limiting to 25 routes at a time, and only working for IPv4 /32 routes, in order to provide some level of security.

In this document, we provide a number of improvements to the existing flowspec mechanism that may foster improved power and utility in inter-domain collaboration to mitigate DDoS:

Packet Rate Action - The existing flowspec standard supports traffic-rate limits conveyed only in bytes per second. In some cases, packets per second is a more relevant metric, and this document adds a packet-rate action that can be used to indicate this.

Tunnel Support - improving the ability for a flowspec to include information about both the outer and inner headers of tunneled traffic, beyond the existing capability that is mainly limited to the outer headers, and has ambiguous applicability in the specification for description of inner headers.

Flowspec Identification - currently, flow specifications are self-identifying, in that there is no shorter way to describe or refer to them other than to copy them. When an attack is detected and a response is initiated, this will often be tracked in one or more systems (e.g. through trouble tickets, etc), and it may be helpful when operators are coordinating responses or debugging in order to be able to refer to a flowspec by a shorter ID string.

Cryptographic Enhancement - the ability to use the Resource Public Key Infrastructure (RPKI) in order to sign BGP messages including flow specifications. This supports stronger verification upstream that the flowspec is genuine and current, since careless use of existing flowspec messages could enable new forms of routing system abuse.

Flow Delegation - using flowspec advertisements so that only traffic matching a particular flowspec is rerouted to a scrubbing center, rather than delegating entire prefixes to the scrubbing center. This can reduce the load on the scrubbing center or avoid performance penalties for other classes of traffic not specifically associated with an attack. Combined with the cryptographic authentication, the ability of scrubbing center services to "hijack" routes can now also be performed more securely. This may have advantages over using the flowspec redirect actions.

Flowspec Feedback - the ability to signal back to a flowspec originator that the flowspec is being honored in a particular AS, and optionally to indicate properties of the filtering activity (e.g. number of packets dropped within a time interval, indication of the previous-hop AS, etc.). This can support a form of telemetry or monitoring capability for flow specifications that have been distributed across multiple cooperating networks.

The enhanced flowspec validation procedure described in [I-D.ietf-idr-bgp-flowspec-oid] is not necessary when cryptographic means can be used to validate a received flowspec. That enhanced validation procedure can be applied in conjunction with the one described in this document.

There is an IETF effort (DOTS) currently working to standardize some signalling for DDoS mitigation [I-D.teague-open-threat-signaling]. This could likely be used in conjunction with the BGP extensions discussed in this document, but we are not attempting to address the DOTS problem space or scenarios directly.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Protocol Extensions

This document section contains descriptions of protocol message modifications that support the additional features described in this document. In some cases, it does not fully describe all the details, but only a sketch of proposed path in order to generate discussion. It is possible that some extensions are more favorable to the community than others, and could be factored out into separate specifications in order to progress as standards.

2.1. Packet Rate Action

In network equipment, it may be easier, faster, or more convenient to perform accounting or decision-making based on quantities of packets rather than quantities of bytes. Additionally, many attacks are effective due to the number of packets per second, and not necessarily due to the number of bytes per second. It is desirable to be able to specify rate limits in terms of the number of packets per second, and not just the number of bytes per second.

Traffic filtering actions pertaining to a matched flow specification are indicated using BGP extended communities. Particular extended community values are defined in RFC 5575 for a small number of possible actions. New types of actions can be defined using additional extended community values.

The existing type 0x8006 ("traffic-rate") extended community value carries a 2-octet ID value with only informational value, and a 4-byte floating point value indicating the number of bytes per second that traffic should be limited to.

We propose to allocate an additional type number (TBD assigned by IANA) called "packet-rate". Similar to the traffic-rate action, this will carry an informational-only ID value, followed by a 4-byte floating point value indicating the number of packets per second that traffic should be limited to.

Although a floating-point value for packets per second may seem odd or unnatural compared to an integer value, the motivations for this are:

- The maximum value that a 32-bit unsigned integer could hold would limit to specifying under 2.15 Gpps (2.15 billion packets per second). For large or high-performance networks especially in the future, this may not be sufficient. The maximum floating point value is much higher (on the order of 10^38) and should be future-proof.

The reduced precision of the floating-point limit that can be specified compared to an integer encoding does not seem to be a major concern.

This maintains consistency with the present syntax for bytes per second rate limits.

Please note that this is a transitive community type, as explained in [RFC7153] and not a non-transitive type as mentioned narratively in the RFC 5575 description of the traffic-rate action.

2.2. Tunnel Description

The existing flow specification format in RFC 5575 includes a mixture of both IP-layer and TCP or UDP-layer fields used to describe a flow or the packet signature to be matched. However, there is no means to nest or scope the flow specification values so that they can be describe flows that are carried within tunnels of various types. For instance, a particular TCP flow within an IP-in-IP tunnel or a particular UDP flow in a GRE tunnel would not be possible to describe. Only values in the "outer" headers can be conveyed, and not the inner headers.

To solve this, we propose to add two additional component types to the flowspec mechanism. The first is necessary in order to separate between outer and inner components of a flow specification. This new "Tunnel Separator" component can be included multiple times in order to select protocol headers nested several layers deep within a tunnel. The second is necessary to compare arbitrary fields and values at given offsets within a packet. This "Offset-Value-Mask" component permits filtering of packets containing inner protocols that upstream routers do not even need to parse or support themselves.

This new separator component will be encoded as a single type byte (1 octet) followed by a tunnel-header-length 2-byte unsigned integer value. The flowspec components preceding a tunnel header apply to outer headers, and components following the separator apply to the inner nested headers.

The tunnel-header-length is used when there are multiple nestings, in order to indicate how much of the packet is within scope of the current section defined by the separator, and to be matched before the next separator is processed. For the final separator in a flowspec, this can be set to 0. The value 0 indicates that no further separators will be present.

For example, to describe a TCP flow to port X inside an IPv4-in-IPv4 tunnel to a particular outer prefix Pout and inner prefix Pin, the flowspec components list would be:

1. Destination Prefix: Pout

2. IP Protocol: 4 (IP-in-IP encapsulation)

3. Tunnel Separator, tunnel-header-length: 0 (final separator)

4. Destination Prefix: Pin

5. IP Protocol: 6 (TCP)

6. Destination Port: X

For IPv4 as an inner protocol, the interpretation of the nested flowspec is obvious per RFC 5575. For IPv6 as an inner protocol, the interpretation of the nested flowspec is also direct per [I-D.ietf-idr-flow-spec-v6]. For other inner protocols, we propose a generic "Offset-Value-Mask" component in order to match particular bits. This could be used, for instance, to match the Protocol Type field of a GRE header along with IP address and TCP port bytes within the inner protocol.

The format for an Offset-Value-Mask component is:

Encoding: <type (1 octet), offset (2 octets), value-len (1 octet), value, mask

offset - indicates a number of leading bytes to be skipped into the packet section being described

value-len - indicates the length in bytes of the following "value" field

value - conveys a number of bytes to be checked at the indicated offset within the packet

mask - indicates a bitmask applied to the value being checked; a 1 bit at a given position mean that the bit value must match in the packet, and 0 bits mean that the value does not need to match in that bit position; the mask must be the same length as the value

As an example, the components that would describe an IPv4 flow to address A within a GRE flow to address Pout would be:

1. Destination Prefix: Pout

2. IP Protocol: 47 (GRE)

3. Tunnel Separator, tunnel-header-length: 0 (final separator)

4. Offset-Value-Mask: 2 (offset 2 bytes into packet), 2 (compare value 2 bytes long), 4 (IPv4 protocol type), 0xFFFF (match all bits)

5. Offset-Value-Mask: 24 (offset 24 bytes into the packet --- 8 bytes for GRE plus 16 bytes into the IPv4 header for the destination address), 4 (compare value 4 bytes long), A, 0xFFFFFFFF (match all 32 bits)

For description of nested IP and transport protocol headers, having several Offset-Value-Mask components is less desirable than just being able to use normal flowspec components on the innermost set of headers, however, since the length of tunnel headers is not always known, this may not be possible in general. In cases where the length of the tunnel header is known, this can be conveyed in the tunnel separator component. As an example, the previous example can be encoded as:

1. Destination Prefix: Pout

2. IP Protocol: 47 (GRE)

3. Tunnel Separator, tunnel-header-length: 8 (GRE header is 8 bytes long)

4. Offset-Value-Mask: 2 (offset 2 bytes into packet), 2 (compare value 2 bytes long), 4 (IPv4 protocol type), 0xFFFF (match all bits)

5. Tunnel Separator, tunnel-header-length: 0 (final separator)

6. Destination Prefix: A

While a set of Offset-Value-Mask rules can be created in order to check multiple nested protocol layers, this can become tedious and error-prone. Instead, when flowspec components can be used to describe an inner header, use of non-zero tunnel-header-length values allows constructions that are more human-intelligible than Offset-Value-Mask permits.

When tunnel flow specifications are used, the validity checking should be applied to the outer destination prefix only (appearing first within the flowspec NLRI).

2.3. Flowspec Identifier

BGP update messages are not uniquely identified in any way. If there is a desire to link specific updates to the events that they relate to, or to refer to specific updates in conversation or collaboration between operators, then a BGP attribute that identifies the messages may be useful.

For instance, "Are you implementing the flowspec 0xA1B2C3D4 from AS N?" (where 0xA1B2C3D4 is a Flowspec Identifier) is likely to be an easier question to communicate than trying to identify it by the full NLRI binary string. It should also be easier to tie unambiguously to particular database entries for active ACLs, attacks being tracked, incident management records, etc. The same flowspec NLRI values may end up being reused between different incidents if they share similar attack signatures.

This attribute could be unique to updates containing the flow specification SAFI, or might be more generally useful in other BGP usages, but we do not have an opinion on that.

For flowspec messages enhanced to use BGPsec, it is possible that the originator's signature (or some truncation of it) could act as an identifier, as long as it covers some field guaranteed to change over time between DDoS incidents, like the timestamp discussed later.

The value of the attribute can be set by the system generating the message, and might be treated as an opaque set of bytes elsewhere in the network. A 32-bit value is likely to be sufficient for any AS to uniquely identify its flowspec messages. This could be generated by the originator in a number of ways, such as via a hash function over some set of fields, a counter, or any other means.

In other ASes, assumptions about the identifier value should not be made, such as about its uniqueness or reuse by the originating AS. It is a purely informational value, not intended to be computed on directly.

Potential approaches for adding an identifier to BGP flowspec messages include:

- New Attribute - Defining a new generic Identifier or "Flowspec Identifier" BGP attribute is a simple approach but requires a new attribute to be defined, understood, and conveyed by BGP speakers.

- BGPsec Signatures - The advantage to this approach is that no new fields are necessary. The disadvantage is that it assumes a new form of BGPsec for flowspec messages is in use.

ASN+NLRI - Simply composing the originator's AS number and the NLRI, or hashing over those with an agreed function, is another option that could be used to compute "mostly unique" identifiers for flowspec messages, without needing to add any fields. A timestamp from the BGP-TS option could be included within the hashed fields in order to help make the result vary when similar NLRI are re-used between event mitigations.

If there is agreement that identifying flowspecs is useful, then one of these (or some other) mechanism can be selected later.

2.4. Cryptographic Enhancement

BGP by default does not include cryptographic protection, and there are various attacks possible on BGP flows and BGP speakers. Misuse or spoofing of flow specifications by an attacker could create new ways to inflict abuse that is difficult for a victim to detect and debug. Because significant work has gone into designing, developing, and beginning to deploy BGPsec, it will be highly desirable to use BGPsec as a means of protecting and validating flow specifications.

Existing BGPsec signatures cover the Network Layer Reachability Information (NLRI) and AS numbers in BGP updates:

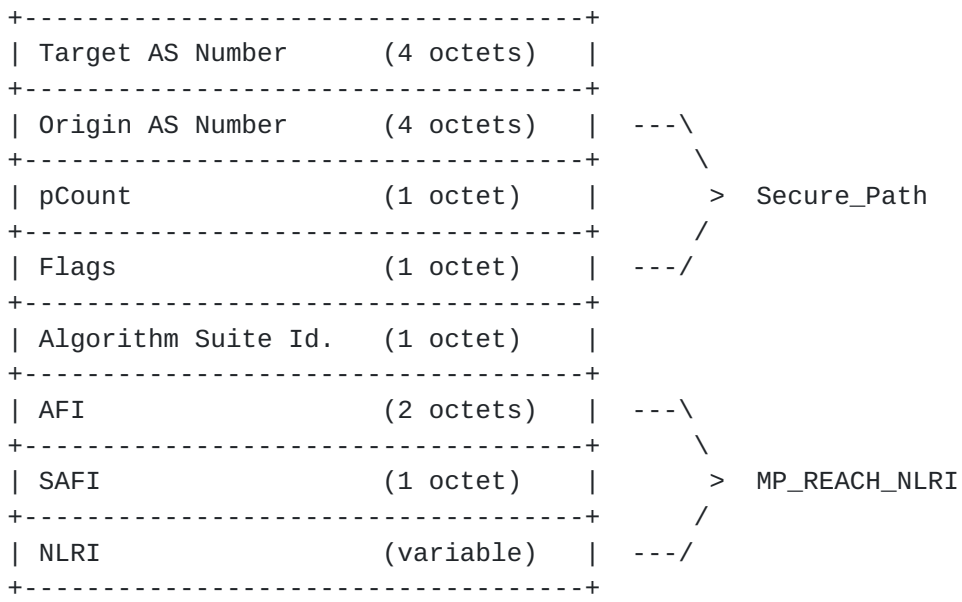


Figure from draft-ietf-sidr-bgpsec-protocol-13

The existing BGPsec specification [I-D.ietf-sidr-bgpsec-protocol] covers AFI values for IPv4 and IPv6, but does not place an explicit restriction on the SAFI. However, BGPsec does assume that the NLRI

has a destination prefix field corresponding to a Route Origination Authorization (ROA). This may not be true for flowspec NLRI (e.g. SAFI 133 for IPv4), which are not strictly required to have a destination prefix.

It appears to us that this is an ommission in the BGPsec checks, since mostly BGPsec is targeting IPv4 and IPv6 unicast forwarding only. BGPsec should only be applicable when the SAFI also matches the AFI, as other SAFI values may also not meet the assumption that NLRI will contain a destination prefix corresponding to a ROA.

Current non-BGPsec flowspec validation procedures are intended to verify that the neighboring AS advertising a flowspec is the same AS advertising the best unicast route for the flowspec destination. However, since the destination prefix field is an optional component, this may not always even be possible. In this case, it might be sufficient to ensure that the flowspec only is applied to traffic that would be sent through the advertising AS and subject to their downstream filtering anyways. For instance, if AS X sends a flowspec to AS Y indicating that all ICMP of some type will be filtered, then this can only be safely applied to packets on egress from AS Y to AS X, but not immediately on ingress to AS Y until the next AS hop can be determined to be AS X. It might be possible to compute all destination prefixes that will be routed to AS X at any time and create ingress ACLs to cover only these, but that set of destination prefixes may need to be recomputed continuously or cause other problems (such as exceeding limitations in the number of ACLs possible).

For these reasons (flowspec validation in general, as well as potential compatibility with BGPsec), it seems highly recommended that flow specifications that are intended to travel multiple AS hops into the Internet should explicitly include a destination prefix. For a tunnel flowspec, there must be a destination prefix in the outermost portion of the specification.

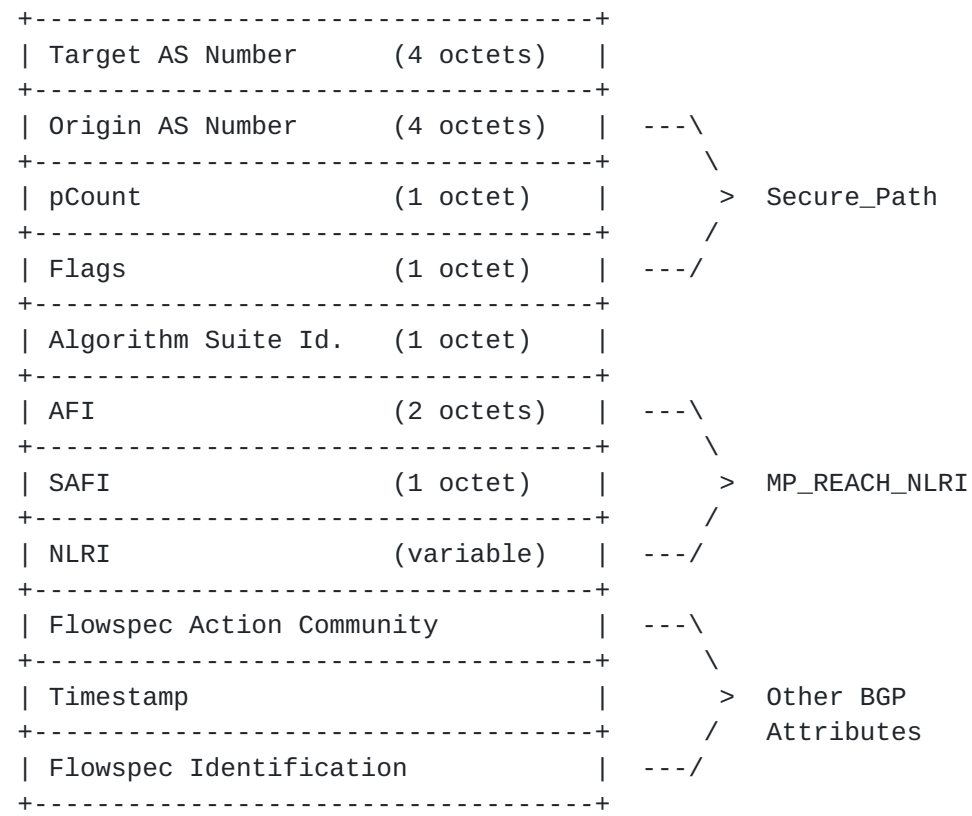
Beyond this, to use BGPsec for flowspec protection, there is a need to also include the community attributes conveying the flowspec actions within the fields covered by the signature. For instance, when using the IP redirection capability [I-D.ietf-idr-flowspec-redirect-ip] a community attribute holds the IPv4 or IPv6 addresses that packets matching the flowspec are either to be forwarded or copied to. Even if BGPsec was used to cover the flowspec AFI, the signatures would not cover these redirection target IP addresses, and they could be altered while in transit, or changed to other types of actions (e.g. rate limits, etc).

Additionally, since DDoS attacks are dynamic and redirection or filtering of a flow may only be necessary for some short time, and be undesirable at other times, it seems useful for a secure flowspec message to include a timestamp as part of the data protected by a signature. Otherwise a replay could be used to re-initiate filtering or redirection actions that would cause performance issues or packet loss. Received flowspecs could be verified to have been timestamped within some window of time (e.g. several minutes or hours), and discarded if they are too stale.

The BGP Timestamp (BGP-TS) attribute [I-D.litkowski-idr-bgp-timestamp] has other purposes (e.g. performance measurements), but may be possible to use for this purpose. A simpler construction intended only to include a single and not-necessarily high-precision timestamp would also suffice for the purposes described in this document, of limiting the potential to replay flowspec messages.

If the Flowspec Identifier is used, then it also should be included within the data covered by a signature.

For BGPsec use with the flowspec SAFI, we propose that the signatures contained in the Signature_Blocks of the BGPsec_Path attribute be computed over:



Suggested Signature Coverage for BGP use with Flowspec

A flow specification destination address may be narrower in scope than the prefix included in a ROA. For instance, a single /32 may be the destination address under attack, and which redirection or filtering is requested for. ROA objects contain ROAIPAddress values including an optional maxLength element. If the destination address field within a flowspec NLRI are considered equivalent to the NLRI of the normal IPv4 and IPv6 unicast SAFIs, then validation should check this against the maxLength value in relevant ROAs. This will mean that ASes intending to use fairly specific flow specifications will need to produce ROAs with permissive maxLength values. If maxLength is not present, the normal ROA validation requires the destination prefix to match exactly with the prefix indicated in the ROA.

2.5. Flow Delegation

The RPKI enables the legitimate holder of an IP prefix to authorize other ASes to originate routes to that prefix. ROA objects in the RPKI express this authorization. ROAs are currently defined to include AS numbers and IP address blocks [RFC6842]. ROAs are not able to cover other types of NLRI, such as a flow specification.

Some current DDoS mitigation systems involve the attack victim allowing another service provider with better connectivity to hijack their prefix(es), filter or scrub the traffic to reduce the volume of attack packets, and re-route the clean traffic to the original destination. With BGPsec in use, the victim would have to publish ROAs for any service provider ASes that they utilize for traffic redirection, but otherwise the practice would still function.

However, since this redirection works at prefix granularity rather than flow-level granularity, all traffic for the victim is impacted, and there may be significant performance impacts for latency and jitter-sensitive flows. This is generally better than leaving the attack unmitigated, but can still impact the business operations of the victim. Flow-level redirection, as enabled by BGP flowspec advertisements would be superior to redirecting all traffic for the prefix.

Additionally, it may reduce the load and burden on the scrubbing center's resources if a majority of the acceptable traffic never has to even be redirected through it, because only questionable traffic identified by a flow specification needs to be rerouted. This could help this mitigation approach to scale to even larger attack volumes than currently seen, even when the heuristics used to distinguish good from bad packets become exceedingly complex.

In order to allow a scrubbing center provider to advertise flow specifications rather than entire prefixes, an additional type of ROA will need to be defined, containing a list of the flowspec NLRI entries that they're authorized to scrub, rather than the simple IP prefix list currently in a ROA. We assume that the default action (of forwarding matching traffic) will be used, and so the flowspec action extended communities do not need to be included in the ROA for the service provider, but this should be a topic for discussion.

This differs from having the victim AS originate a flowspec route with an IP redirect action towards the mitigation service provider. First, if the victim has been knocked completely offline or if some facet of a coordinated attack also impacts their BGP infrastructure, that may not even be possible. Second, it may offer scrubbing center providers more flexibility in how their services are implemented, by not requiring them to specify a single IP address in an IP redirect action. This could aid in maintaining the scalability of this type of mitigation.

2.6. Feedback

Currently, BGP flowspec updates can be sent, but there is no feedback explicit in order to indicate whether the flow specifications are being put into action or to monitor their effectiveness. Implicitly, a reduction in attack traffic volume reaching the victim may suggest that the flowspec is being honored, but this can also happen simply because the attack is subsiding.

As attacks become more prevalent, more persistent, and more advanced in their tactics, signatures, and dynamics, it will be useful for coordinated defense efforts to be monitored and for the attack volume at different vantage points within the network to be synthesized.

In order to enable this, we propose to add a feedback message to BGP allowing individual ASes participating in attack mitigation to optionally advertise this fact, and provide basic information about their status.

Feedback messages should include fields for:

1. Reporting AS

2. Flowspec Identification

3. Report Time Interval (start and end timestamps)

4. (optional) Ingress AS List

5. (optional) List of Matched Packets Counter (within the time interval)

The Ingress AS List identifies where the attack traffic matching the flowspec is coming from, which may be multiple points. This can be useful in tracing back the source of an attack, especially when IP addresses are spoofed, and ingress filtering has either not been properly implemented or has been defeated somehow (e.g. through tunneling, or other abuses). The List of Match Packet Counters can convey the volume of traffic coming from each AS in the ingress AS list.

Since some of this information may be difficult to collect and synthesize, it is marked as optional. At a basic level, the feedback that a flowspec is being implemented by an upstream AS is useful to the victim.

These messages could be signed, potentially reusing certificates from the RPKI, in order to avoid potential abuse of the feedback mechanism and to discourage fraudelent reporting of incorrect information.

Due to the number of ASes on the Internet, any given AS will need to be very judicious about how it generates and propagates these feedback indications. For instance, there may not be a problem one AS-hop away from the victim to provide these reports on 1-minute intervals, but deeper into the AS graph, generating and propagating the feedback could become overwhelming. When responding to particular attacks, and coordinating across provider on specific attacks, it could be possible to enable reporting and tweak the time intervals for reporting based on a particular Flowspec Identification value.

Other heuristics will also help to reduce the volume of feedback, such as using a small reporting interval for "fresh" flowspec advertisements and backing off the reporting interval over time either based on the age of the flowspec or the volume of matching traffic observed. There are many algorithms that can be employed to keep the feedback manageable, and these do not need to be uniform across ASes. The mere presence and generation of any feedback adds utility not present in the existing system.

Since Identification values might be reused over time by the originating AS (either accidentally or on purpose) this could lead to ambiguity in feedback or issues in accounting by other ASes). Since this is likely to lead to poor utility for the originating AS, it should be highly encouraged that when ASes generate flowspec messages, that they select fresh Identifiers that do not collide with other values that they have recently used. In the case that another AS detects an apparent collision in its systems that utilize received flow specifications or account on status of implemented flow specifications, then the receiving AS is free to choose any reasonable action that it pleases (e.g. suspending implementation of the flowspecs, suspending reporting on the flowspecs, using only the most recent flowspec, etc).

3. Other Discussion

The existing RFC 5575 flowspec definition can be translated easily into Access Control List (ACL) rules in order to perform hardware-based filtering on many platforms. The proposal to include tunnel specifications in this document may not be as easily or directly transformable into ACLs across such a wide range of systems. Future platforms might be more capable.

Some of the flowspec components are able to specify comparison operators (e.g. less-than or greater-than) that can be applied to packet fields. This is useful for indicating a range of port numbers, for instance. The Offset-Value-Mask component specified in this document only includes a bitmask to check against, so is weaker than existing flowspec components for checking some fields. We thought that this generic bitwise comparison would be more easily supported in hardware than a more complete set of comparison operators that might apply to different sized fields at different packet offsets. We hope to receive more feeback from the community of implementers on this topic in particular.

Proper BGP operation is critical to the Internet's stability and changes and extensions to BGP must be carefully deployed. The new mechanisms described in this document are no exception. The text in RFC 5575 which added the Flow Specification NLRI to BGP is also applicable here:

> As with previous extensions to BGP, this specification makes it possible to add additional information to Internet routers. These are limited in terms of the maximum number of data elements they can hold as well as the number of events they are able to process in a given unit of time. The authors believe that, as with previous extensions, service providers will be careful to keep information levels below the maximum capacity of their devices.

Certain types of DDoS attacks are still not possible to easily indicate in flowspec messages. For instance, a "crossfire" attack where traffic is directed at multiple destinations that share the same tight link as the actual victim can be difficult to securely signal even with the extensions described in this document. Other types of attacks focused on particular application properties can also be difficult to capture in packet-level flowspec logic. Future work might address this limitation.

In some cases, the victims of an attack may be reticent to have knowledge or acknowledgement of the attack on them propagated. Their providers may be able to mitigate attacks within their networks, but forbidden from working with other providers further upstream to more efficiently mitigate the attack. This makes it very challenging to apply collaborative defenses to aid these types of attack victims. Future work is possible that considers this and whether there are improvements to the flowspec construction that would support dealing with these situations.

3.1. Other Extensions

IP options currently defy description within a flow specification, and further complicate the parsing and processing of transport headers in general and inner protocols in the case of tunnels. Other future work may be possible in order to better deal with this.

4. Acknowledgements

Work on some of the material discussed in this document was sponsored by the United States Department of Homeland Security (contract HSHQDC-15-C-00017), but it does not necessarily reflect the position or the policy of the Government and no official endorsement should be inferred. Support and feedback from Dan Massey was helpful in formulating ideas in this document.

5. IANA Considerations

If accepted for publication, IANA will need to allocate a BGP extended community value for the packet-rate action from the "Generic Transitive Experimental Use Extended Community Sub-Types" registry.

6. Security Considerations

This document provides enhanced capabilities to defend against DDoS attacks. In doing so, there is an intent to not add any additional vulnerabilities that could be exploited.

Because the mechanisms described in this document are conveyed over BGP, they are subject to the risks posed by the underlying BGP connection's configuration and its ability to implement security features.

The extensions described in this document are not intended to reduce the security properties of the BGP flowspec mechanism originally defined in RFC 5575. By offering a means of cryptographic protection for authenticating flowspec messages, it should provide an improvement over the security properties of the basic RFC 5575 signalling. There is a dependancy on use of the RPKI in order to obtain this improvement.

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

7.2. Informative References

[I-D.ietf-idr-bgp-flowspec-oid]
Uttaro, J., Filsfils, C., Smith, D., Alcaide, J., and P. Mohapatra, "Revised Validation Procedure for BGP Flow Specifications", draft-ietf-idr-bgp-flowspec-oid-02 (work in progress), January 2014.

[I-D.ietf-idr-flow-spec-v6]
Raszuk, R., Pithawala, B., McPherson, D., and A. Andy, "Dissemination of Flow Specification Rules for IPv6", draft-ietf-idr-flow-spec-v6-06 (work in progress), November 2014.

[I-D.ietf-idr-flowspec-redirect-ip]
Uttaro, J., Haas, J., Texier, M., Andy, A., Ray, S., Simpson, A., and W. Henderickx, "BGP Flow-Spec Redirect to IP Action", draft-ietf-idr-flowspec-redirect-ip-02 (work in progress), February 2015.

[I-D.ietf-sidr-bgpsec-protocol]
Lepinski, M., "BGPsec Protocol Specification", draft-ietf-sidr-bgpsec-protocol-13 (work in progress), July 2015.

[I-D.litkowski-idr-bgp-timestamp]
Litkowski, S., Patel, K., and J. Haas, "Timestamp support for BGP paths", draft-litkowski-idr-bgp-timestamp-02 (work in progress), March 2015.

[I-D.teague-open-threat-signaling]
Teague, N., "Open Threat Signaling using RPC API over HTTPS and IPFIX", draft-teague-open-threat-signaling-01 (work in progress), July 2015.

[Kristoff15]
Kristoff, J., "An Internet-wide BGP RTBH Service", April 2015.

2015 IAB CARIS workshop

[PS15] Pinkerton, S. and C. Strasburg, "Coordinating Attack Response at Internet Scale", April 2015.

[RFC3882] Turk, D., "Configuring BGP to Block Denial-of-Service Attacks", RFC 3882, DOI 10.17487/RFC3882, September 2004, <http://www.rfc-editor.org/info/rfc3882>.

[RFC5575] Marques, P., Sheth, N., Raszuk, R., Greene, B., Mauch, J., and D. McPherson, "Dissemination of Flow Specification Rules", RFC 5575, DOI 10.17487/RFC5575, August 2009, <http://www.rfc-editor.org/info/rfc5575>.

[RFC5635] Kumari, W. and D. McPherson, "Remote Triggered Black Hole Filtering with Unicast Reverse Path Forwarding (uRPF)", RFC 5635, DOI 10.17487/RFC5635, August 2009, <http://www.rfc-editor.org/info/rfc5635>.

[RFC6842] Swamy, N., Halwasia, G., and P. Jhingran, "Client Identifier Option in DHCP Server Replies", RFC 6842, DOI 10.17487/RFC6842, January 2013, <http://www.rfc-editor.org/info/rfc6842>.

[RFC7153] Rosen, E. and Y. Rekhter, "IANA Registries for BGP Extended Communities", RFC 7153, DOI 10.17487/RFC7153, March 2014, <http://www.rfc-editor.org/info/rfc7153>.

[SSBP15] Steinberger, J., Sperotto, A., Baier, H., and A. Pras, "Exchanging Security Events of flow-based Intrusion Detection Systems at Internet Scale", April 2015.

2015 IAB CARIS workshop

Authors' Addresses

Wesley Eddy
MTI Systems

Email: wes@mti-systems.com

Justin Dailey
MTI Systems

Email: justin@mti-systems.com

Gilbert Clark
MTI Systems

Email: gclark@mti-systems.com