Global Routing Operations
Internet-Draft
Intended status: Informational
Expires: 11 October 2025

T. Fiebig
MPI-INF

9 April 2025

Current Options for Securing Global Routing
draft-ietf-grow-routing-ops-sec-inform-00

Abstract

The Border Gateway Protocol (BGP) is the protocol is a critical component in the Internet to exchange routing information between network domains. Due to this central nature, it is an accepted best practice to ensure basic security properties for BGP and BGP speaking routers. While these general principles are outlined in BCP194, it does not provide a list of technical and implementation options for securing BGP.

This document lists available options for securing BGP, serving as a contemporary, non-exhaustive, repository of options and methods. The document explicitly does not make value statements on the efficacy of individual techniques, not does it mandate or prescribe the use of specific technique or implementations.

Operators are advised to carefully consider whether the listed methods are applicable for their use-case to ensure best current practices are followed in terms of which security properties need to be ensured when operating BGP speakers. Furthermore, the listed options in this document may change over time, and should not be used as a timeless ground-truth of applicable or sufficient methods.

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 https://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 11 October 2025.

Table of Contents

1. Introduction

The Border Gateway Protocol (BGP), specified in [RFC4271], is the protocol used in the Internet to exchange routing information between network domains. BGP does not directly include mechanisms that control whether the routes exchanged conform to the various guidelines defined by the Internet community. Besides, BGP itself, by its design, does not have any direct way to protect itself against possible security-related threats. This document intends to serve as a snapshot of currently available methods for ensuring BGP security.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Scope of the Document

The methods listed in this document are intended for BGP sessions carrying generic Internet routing information within the DFZ. It specifically does not cover security mechanics for other uses of BGP, e.g., when using BGP for NLRI exchange in a data-center context.

This document is a non-exhaustive and non-authoritative repository of available tools, methods, and techniques. When consulting this document, operators should consider that available tools and mechanisms, as well as the described circumstances and considerations may change over time.

The document does not make specific recommendations for the use of specific mechanisms, implementations, or configurations. Instead, operators are advised to carefully weigh the implications of listed methods and to apply their own judgement to assess whether these methods are appropriate for their network to ensure BGP security properties.

3. Protection of the BGP Speaker

The BGP speaker needs to be protected from external attempts to subvert the BGP session. Furthermore, access to management services of the BGP speaker should be limited to neighbors, as these services usually share resources with the control plane and, e.g., automated attacks on management ports may impact the BGP speaker’s ability to execute BGP related tasks.

3.1. BGP Network Layer Protection

To protect a BGP speaker on the network layer, the ability to connect to TCP port 179 on the local device should be restricted to known addresses that are permitted to become a BGP neighbor. Experience has shown that the natural protection TCP should offer is not always sufficient, as it is sometimes run in control-plane software. In the absence of ACLs, it is possible to attack a BGP speaker by simply sending a high volume of connection requests to it. This protection SHOULD be implemented by using an Access Control List (ACL) to limit access to TCP port 179 to authorized hosts.

If supported, an ACL specific to the control plane of the router SHOULD be used (receive-ACL, control-plane policing, etc.), to avoid configuration of data-plane filters for packets transiting through the router (and therefore not reaching the control plane). If the hardware cannot do that, interface ACLs can be used to block packets addressed to the local router.

Some routers automatically program such an ACL upon BGP configuration. On other devices, this ACL should be configured and maintained manually or using scripts.

In addition to strict filtering, rate-limiting MAY be configured for accepted BGP traffic. Rate-limiting BGP traffic consists in permitting only a certain quantity of bits per second (or packets per second) of BGP traffic to the control plane. This protects the BGP router control plane in case the amount of BGP traffic surpasses platform capabilities.

Furthermore, it is possible to use non-gloablly reachable addresses for BGP session links. Options include using IPv4 routes with an IPv6 next hop in IPv4 sessions (see [RFC9229]), using prefixes not advertised in the GRT ([TBD]), using unnumbered BGP/Link-Local addresses (also using [RFC9229]), or using [RFC1918] addresses for IPv4 sessions. Even though routing based network layer protection MAY be implemented, it SHOULD only be done in addition to deploying ACLs. If any of these approaches is utilized, it MUST be ensured that the BGP speaker originates Path MTU Discovery related packets

(see [RFC1191] for IPv4 and [RFC8201] for IPv6) from a globally reachable address, to ensure that Reverse Path Filtering of external parties does not interfere with PMTUD discovery for transiting traffic.

3.2. BGP Speaker Management Interface Protection

Usually, a BGP speaker's management interface is also reachable in-band, i.e., via the default routing domain / VRF (Virtual Routing Fabric) of the control plane. To make it easier to separate BGP and management related control plane traffic, management traffic SHOULD be exclusively handled via dedicated out-of-band management. This network SHOULD be protected from unauthorized connections by ACLs not handled on the BGP speaker itself to ensure that the control plane cannot be overloaded by attacks on the management interfaces of the BGP speaker.

Please note that, in general, filtering and rate-limiting of control-plane traffic is a wider topic than "just for BGP". For further recommendations on how to protect the router's control plane, see [RFC6192])

4. Protection of the BGP Sessions

Current security issues of TCP-based protocols (therefore including BGP) have been documented in [RFC6952]. The following subsections list the major points raised in this document and give the best practices related to TCP session protection for BGP operation.

4.1. Protection of TCP Sessions Used by BGP

Attacks on TCP sessions used by BGP (aka BGP sessions), for example, sending spoofed TCP RST packets, could bring down a BGP session. Following a successful ARP spoofing attack (or other similar man-in-the-middle attack), the attacker might even be able to inject packets into the TCP stream (routing attacks).

BGP sessions can be secured with a variety of mechanisms.

4.1.1. Integrity Verification and Authentication

MD5 protection of the TCP session header, described in [RFC5925], was the first available mechanism to protect the integrity of a BGP session. It has been obsoleted by the TCP Authentication Option (TCP-AO; [RFC5925]), which offers stronger protection. While MD5 is still the most used mechanism due to its availability in vendors' equipment, TCP-AO SHOULD be preferred when implemented by both sides of a session.

Optionally, if TCP-AO is not supported, while both sides of the BGP session can support a stronger authentication algorithm than MD5, such as SHA-1 or SHA-256, using the stronger method SHOULD be considered. Aside from that, using keychain-based cryptographic keys lifecycle management, as suggested in [RFC6518] is highly RECOMMENDED.

Additionally, IPsec could also be used for session protection. At the time of publication, there has been no wide-spread adoption of using IPsec for BGP sessions, and further analysis is required to define guidelines.

The drawback of TCP session protection is additional configuration and management overhead for the maintenance of authentication information (for example, MD5 passwords). In either case, protection of TCP sessions used by BGP SHOULD be enabled when BGP sessions are established over shared networks where the risk of spoofing is high (like IXPs). Operators are also RECOMMENDED to consider the trade-offs and apply BGP session protection on all other external BGP sessions as well.

Aside of this, most vendors use simple, reverse-decryptable password hash algorithm to store shared secrets keys for BGP (and other routing protocols) in devices' configuration files. While this practice simplifies password management tasks, since the passwords can always easily be deciphered, it carries the risk of leaking this information if a configuration is shared, e.g., with a vendor for a support case, or if the device is decommissioned and later resold without having been wiped. Hence, if a device offers more secure storage mechanisms for secrets, these SHOULD be used.

Furthermore, operators SHOULD block spoofed packets (packets with a source IP address not belonging to their IP address space) at all edges of their network (see [RFC2827] and [RFC3704]). This protects the TCP session used by Internal BGP (iBGP) from attackers outside the Autonomous System. Similarly, the considerations for using non globally reachable addresses for links handling BGP sessions from Section 3.1 apply accordingly.

Furthermore, as an additional security measure, iBGP sessions SHOULD also be protected using the authentication mechanisms discussed above.

4.1.2. Defending Against PMTUD Related Attacks

In 2018 an attack on BGP was described in the literature which claims to enable BGP route injection without Layer 2 adjacency by leveraging PMTUD, see ([FENG-22]). The attack leverages packet fragmentation to bypass standard TCP protection mechanisms, so routes can be injected into an established BGP session. While the attack would be mitigated by the integrity mechanisms suggested in Section 4.1.1, operators SHOULD additionally take precautions to defend against these attacks, especially if authentication mechanisms are not in use. To mitigate this attack, BGP speakers should not allow packet fragmentation on the control plane for BGP traffic between themselves and their neighbors. This is feasible, as even on multi-hop sessions, the path MTU should be known to the operators, meaning that it can be statically and consistently configured for both speakers involved in a session to prevent the need for fragmentation. Hence, operators SHOULD ensure that fragmentation is neither allowed nor necessary for BGP packets between two BGP speakers. If this is not possible, a strict lower limit for the MTU SHOULD be configured. This is usually done for TCP packets like those for a BGP session using MSS (Maximum Segment Size) clamping. Given that IPv6 requires an MTU of at least 1280b [RFC8200], and to keep clamping consistent between IPv4 and IPv6, an MTU of 1280b, i.e., an MSS of 1240b for IPv4 and 1220b for IPv6, is the RECOMMENDED minimum.

4.2. BGP TTL Security (GTSM)

BGP sessions can be made harder to spoof with the Generalized TTL Security Mechanisms (GTSM aka TTL security), defined in [RFC5082]. Instead of sending TCP packets with TTL value of 1, the BGP speakers send the TCP packets with TTL value of 255, and the receiver checks that the TTL value equals 255. Since it’s impossible to send an IP packet with TTL of 255 to an IP host that is not directly connected, BGP TTL security effectively prevents all spoofing attacks coming from third parties not directly connected to the same subnet as the BGP-speaking routers. Operators SHOULD implement TTL security on directly connected BGP neighbors.

GTSM could also be applied to multi-hop BGP session as well. To achieve this, TTL needs to be configured with a proper value depending on the distance between BGP speakers (using the principle described above). Nevertheless, it is not as effective because anyone inside the TTL diameter could spoof the TTL.

Like MD5 protection, TTL security has to be configured on both ends of a BGP session.

5. Static Prefix Filtering

The main aspect of securing BGP resides in controlling the prefixes that are received and advertised on the BGP session. Prefixes exchanged between BGP neighbors are controlled with inbound and outbound filters that can match on well-known/statically typed IP prefixes (as described in this section), a combination of Prefix and AS paths (see), BGP roles as (see Section 6.5.1), or any other attributes of a BGP prefix (for example, BGP communities, as described in Section 6.5.2).

This section lists the most commonly used static prefix filters. We define static prefixes as prefixes that are published via an authoritative list which changes, on average, not more frequently than every 12 months. We will utilize these definitions of static prefixes in Section 8 to clarify where and how these filters should be applied.

5.1. Special-Purpose Prefixes

5.1.1. IPv4 Special-Purpose Prefixes

The IANA IPv4 Special-Purpose Address Registry [IANAv4Spec] maintains the list of IPv4 special-purpose prefixes and their routing scope, and it SHOULD be used for prefix-filter configuration. Prefixes with value "False" in column "Global" SHOULD be discarded on Internet BGP sessions (eBGP).

5.1.2. IPv6 Special-Purpose Prefixes

The IANA IPv6 Special-Purpose Address Registry [IANAv6Spec] maintains the list of IPv6 special-purpose prefixes and their routing scope, and it SHOULD be used for prefix-filter configuration. Only prefixes with value "False" in column "Global" SHOULD be discarded on Internet BGP sessions.

5.2. Unallocated Prefixes

IANA allocates prefixes to RIRs that in turn allocate prefixes to LIRs (Local Internet Registries). While it is in general sensible to not accept routing table prefixes that are not allocated by IANA and/or RIRs, it is important to understand that filtering unallocated prefixes requires constant updates, as prefixes are continually allocated. Therefore, automation of such prefix filters is key for the success of this approach. Operators SHOULD NOT consider solutions described in this section if they are not capable of maintaining updated prefix filters: the damage would probably be worse than the intended security policy. In this section we focus on

IP address space allocated to RIRs by IANA. Allocations by RIRs are generally more dynamic. Therefore, we will discuss using RIR level data in Section 6.1.

5.2.1. IANA-Allocated Prefix Filters

IANA has allocated all the IPv4 available space. Therefore, there is no reason why operators would keep checking that prefixes they receive from BGP neighbors are in the IANA-allocated IPv4 address space [IANAv4Reg]. No specific filters need to be put in place by operators who want to make sure that IPv4 prefixes they receive in BGP updates have been allocated by IANA.

For IPv6, given the size of the address space, it can be seen as wise to accept only prefixes derived from those allocated by IANA. Operators can dynamically build this list from the IANA- allocated IPv6 space [IANAv6Reg]. As IANA keeps allocating prefixes to RIRs, the aforementioned list should be checked regularly against changes, and if they occur, prefix filters should be computed and pushed on network devices. The list could also be pulled directly by routers when they implement such mechanisms. As there is delay between the time an RIR receives a new prefix and the moment it starts allocating portions of it to its LIRs, there is no need for doing this step quickly and frequently. However, operators SHOULD ensure that all IPv6 prefix filters are updated within a maximum of one month after any change in the list of IPv6 prefixes allocated by IANA.

If the process in place (whether manual or automatic) cannot guarantee that the list is updated regularly, then it’s better not to configure any filters based on allocated networks. The IPv4 experience has shown that many network operators implemented filters for prefixes not allocated by IANA but did not update them on a regular basis. This created problems for the latest allocations, and required extra work for RIRs that had to "de-bogonize" the newly allocated prefixes. (See [RIPE-351] for information on de-bogonizing.)

5.3. Prefixes That Are Too (Un)Specific

Most ISPs will not accept advertisements beyond a certain level of specificity (and in return, they do not announce prefixes they consider to be too specific). That acceptable specificity is decided for each session between two BGP neighbors. Some ISP communities have tried to document acceptable specificity. This document does not make any judgement on what the best approach is, it just notes that there are existing practices on the Internet and recommends that the reader refer to them. As an example, the RIPE community has documented that, at the time of writing of this document, IPv4

prefixes longer than /24 and IPv6 prefixes longer than /48 are generally neither announced nor accepted in the Internet [RIPE-399] [RIPE-532]. These values may change in the future.

Some operators MAY choose to allow customers to additionally announce more specifics than commonly used on the Internet (see Section 5.3). This can be to allow customers more fine-grained traffic steering in case of multiple BGP sessions between the AS and its customer in multiple locations, and/or to sub-delegate IPv4 address space smaller than a /24 from the AS' allocation to the customer.

In that case, the operators SHOULD add a specific accept rule for these exact prefixes before Rule 11. Routes of this type SHOULD be annotated in away that ensures they are not re-exported to other neighbors (see Section 6.5.2). Furthermore, in case of using more specifics for traffic steering, the customer SHOULD also announce at least the covering /24 to ensure global reachability of the prefix and prevent issues with uRPF (see also [RFC8704] and Section 6.6.3).

Similar to too specific routes, most ISPs will not accept advertisements beyond a certain level of aggregation. The general guideline here are the least specific allocations commonly handed out by RIRs to LIRs. At the moment, the largest allocations for IPv4 are continuous /8. For IPv6, one /13 allocation exists, followed by several LIRs holding /19. Several operators currently limit the smallest prefix size for IPv6 to /16. This document does not make any judgement on what the best approach is, it just notes that there are existing practices on the Internet and recommends that the reader refer to them. These values may change in the future.

5.4. The Default Route

5.4.1. IPv4

Typically, the 0.0.0.0/0 prefix is not intended to be accepted or advertised except in specific customer/provider configurations; general filtering outside of these is RECOMMENDED.

5.4.2. IPv6

Typically, the ::/0 prefix is not intended to be accepted or advertised except in specific customer/provider configurations; general filtering outside of these is RECOMMENDED.

6. Dynamic Prefix Filtering

In this section, we discuss dynamic prefix filters, i.e., filters that decide whether a prefix should be im- or exported or not based on frequently changing parameters and external resources.

6.1. Prefix Filters Created from Internet Routing Registries (IRRs)

A more precise check can be performed when one would like to make sure that received prefixes are being originated or transited by Autonomous Systems (ASes) entitled to do so. It has been observed in the past that an AS could easily advertise someone else's prefix (or more specific prefixes) and create black holes or security threats. To partially mitigate this risk, administrators would need to make sure BGP advertisements correspond to information located in the existing registries.

An Internet Routing Registry (IRR) is a database containing Internet routing information, described using Routing Policy Specification Language objects as described in [RFC4012]. Operators are given privileges to describe routing policies of their own networks in the IRR, and that information is published, usually publicly. A majority of Regional Internet Registries do also operate an IRR and can control whether registered routes conform to the prefixes that are allocated or directly assigned. However, it should be noted that the list of such prefixes is not necessarily a complete list, and as such the list of routes in an IRR is not the same as the set of RIR-allocated prefixes. Furthermore, especially IRRs not operated by RIRs regularly list conflicting information, see Section 6.1.

6.1.1. Route Objects

The corner stone of IRR based information are ROUTE (IPv4) and ROUTE6 (IPv6) objects. These document, for a given prefix, the AS/ASes allowed to originate the prefix. Note that for a given prefix also more specific objects may exist. However, technically, the semantic of a ROUTE/ROUTE6 object is that of an exact match.

Operators SHOULD create ROUTE/ROUTE6 objects for all prefixes they do or do plan to originate.

6.1.2. AS-SETs

An AS-SET is an object that contains AS numbers or other AS-SETs. The purpose of AS-SETs is creating a recursively queryable structure documenting the cone of an AS. An operator may create an AS-SET defining all AS numbers of its customers. A transit provider might create an AS-SET listing the AS numbers or AS-SETS of those ASes it provides upstream to. In turn, these ASes describe the AS numbers/ AS-SETS of their customers, etc. Using recursion, it is possible to retrieve from an AS-SET the complete list of AS numbers that the neighbor is likely to announce. For each of these AS numbers, it is also easy to look in the corresponding IRR for all associated prefixes.

Please note that different IRR may provide conflicting data, especially on AS-SETs. Recently, an attack was observed where a malicious party created an empty AS-SET for a large transit provider (see [NLNOG-22]). As it was created in an RIR database often taking precedent over other IRR sources, several ASes imported this empty AS-SET, and hence filtered all prefixes advertised by this transit provider. To mitigate this issue, hierarchical AS-SETs reside in the IRR of the RIR and explicitly list the ASN to which they pertain, e.g., AS65536:AS-EXAMPLE. Additionally, the IRR source may also be referenced: RIPE::AS65536:AS-EXAMPLE.

Operators SHOULD create a hierarchical AS-SET representing their cone. If AS-SETs are included in another AS-SET, they SHOULD be hierarchical.

6.1.3. Recursively Computing Filters

Using AS-SETs and ROUTE/ROUTE6 objects, it is possible to use the IRR information to build, for a given neighbor AS, a list of prefixes the neighbor is authorized to originated or transited. This can be done relatively easily using scripts and existing tools capable of retrieving this information from the registries. This approach is exactly the same for both IPv4 and IPv6.

The macro-algorithm for the script is as follows. For the neighbor that is considered, the distant operator has provided the AS and may be able to provide a hierarchically named AS-SET object (aka AS-MACRO). With these two mechanisms, a script can build, for a given neighbor, that lists allowed prefixes and the AS number from which they should be originated. One could decide not to use the origin information and only build monolithic prefix filters from fetched data combining prefixes a neighbor is authorized to transit and originate.

As prefixes, AS numbers, and AS-SETs may not all be under the same RIR authority, it is difficult to choose for each object the appropriate IRR to poll. Some IRRs have been created and are not restricted to a given region or authoritative RIR. They allow RIRs to publish information contained in their IRR in a common place. They also make it possible for any subscriber (probably under contract) to publish information too. When doing requests inside such an IRR, it is possible to specify the source of information in order to have the most reliable data. One could check a popular IRR containing many sources (such as RADb [RADb], the Routing Assets Database) and only select as sources some desired RIRs and trusted major ISPs (Internet Service Providers).

As objects in IRRs may frequently vary over time, it is important that prefix filters computed using this mechanism are refreshed regularly. Refreshing the filters on a daily basis SHOULD be considered because routing changes must sometimes be done in an emergency and registries may be updated at the very last moment. Note that this approach significantly increases the complexity of the router configurations, as it can quickly add tens of thousands of configuration lines for some important neighbors, e.g., large peers or downstreams. To manage this complexity, operators could use, for example, bgpq4 [bgpq4], a set of tools making it possible to simplify the creation of automated filter configuration from policies stored in an IRR.

6.2. SIDR - Secure Inter-Domain Routing: RPKI and ASPA

SIDR (Secure Inter-Domain Routing), described in [RFC6480], has been designed to secure Internet advertisements. Even though technically incorrect, as it is only the name of an important component, the use of techniques entailed in SIDR is commonly referred to as RPKI (Resource Public Key Infrastructure).

There are basically two services that SIDR offers:

* Origin validation, described in [RFC6811], seeks to make sure that attributes associated with routes are correct, see Section 6.2.1. (The major point is the validation of the AS number originating a given route.) Origin validation is now operational (Internet registries, protocols, implementations on some routers), and in theory it can be implemented knowing that the number of signed resources is still low at the time of writing this document.

* Path validation provided by BGPsec [RFC7353] seeks to make sure that no one announces fake/wrong BGP paths that would attract traffic for a given destination; see [RFC7132]. Even though the work on BGPsec has been concluded, adoption is still limited.

Instead, the use of ASPA (Autonomous System Provider Authorization) [I-D.ietf-sidrops-aspa-verification] objects, see also Section 6.2.2, is likely to be more relevant within the context of BGP. Even though the draft on ASPA has not been finalized, first adoption of ASPA can be observed, and first implementations started to support it, following the development of [I-D.ietf-sidrops-aspa-verification], see [OpenBGPd] and [rpki-client].

Implementing SIDR mechanisms is expected to solve many of the BGP routing security problems in the long term, but it may take time for deployments to be made and objects to become signed. Also, note that the SIDR infrastructure is complementing (not replacing) the security best practices listed in this document. Therefore, operators SHOULD implement any SIDR proposed mechanism (for example, route origin validation) on top of the other existing mechanisms even if they could sometimes appear to be targeting the same goal.

6.2.1. Route Origin Validation (ROV)

If route origin validation is implemented, the reader SHOULD refer to the rules described in [RFC7115]. In short, each external route received on a router SHOULD be checked against the Resource Public Key Infrastructure (RPKI) data set:

* If a corresponding ROA (Route Origin Authorization) is found and is valid, then the prefix SHOULD be accepted.

* If the ROA is found and is INVALID, then the prefix SHOULD be discarded.

* If a ROA is not found, then the prefix SHOULD be accepted, but the corresponding route SHOULD be given a low preference.

In addition to this, operators SHOULD sign their routing objects so their routes can be validated by other networks running origin validation. Please note that, when signing routing objects, operators SHOULD strive to create minimally covering ROAs for their intended announcements, see [RFC7115] and [RFC9319], to reduce the attack surface of forged-origin hijacks and attempts to exhaust routers' route processing capacity in terms of memory and CPU [KIRIN-22]. For example, if an operator received a /29 allocation and intends to announce it in a deaggregation of /32, the corresponding ROA should cover the /29 with a longest allowed prefix of /32, instead of signing for a deaggregation up until /48.

One should understand that the RPKI model brings new, interesting challenges. The paper "On the Risk of Misbehaving RPKI Authorities" [hotRPKI] explains how the RPKI model can impact the Internet if authorities don't behave as they are supposed to. Further analysis is certainly required on RPKI, which carries part of BGP security.

6.2.2. Autonomous System Provider Authorization (ASPA)

If autonomous system provider authorization is implemented, the reader SHOULD refer to the rules described in [I-D.ietf-sidrops-aspa-verification]. In short, each external route received on a router SHOULD be checked against the ASPA record found in the Resource Public Key Infrastructure (RPKI) based on the relationship to the neighbor.

In [I-D.ietf-sidrops-aspa-verification], see following sections based on the neighbor relationship:

* Section 6.1 for routes received from customer, lateral peer, by an RS from an RS-client, or by an RS-client from an RS.

* Section 6.2 for routes received from an upstream or mutual-transit neighbor.

ASPA validation can result in one of three outcomes, VALID, INVALID, and UNKNOWN.

* If a route's AS_PATH is evaluated as VALID, then the prefix SHOULD be accepted.

* If a route's AS_PATH is evaluated as INVALID, then the prefix SHOULD be discarded, and the event logged.

* If a route's AS_PATH is evaluated as UNKNOWN, then the prefix SHOULD be accepted, and the event logged. The corresponding route MAY be given a low preference.

6.2.3. Running an RPKI Validator

A key component of RPKI ROV is a validator that collates ROAs from the RIR TAs and distributes this information to routes (via the RTR protocol, or others per operator preference). Operators SHOULD run their own validator and SHOULD NOT outsource the collection and validation of ROAs to a third party.

6.3. Inbound Filtering Prefixes Belonging to the Local AS

A network SHOULD filter its own prefixes on BGP sessions with all its neighbors (inbound direction). This prevents local traffic (from a local source to a local destination) from leaking over an external BGP session, in case someone else is announcing the prefix over the Internet. This also protects the infrastructure that may directly suffer if the backbone's prefix is suddenly preferred over the Internet.

In some cases, for example, multihoming scenarios, such filters SHOULD NOT be applied, as this would break the desired redundancy.

6.4. Inbound Filtering Prefixes Belonging to Downstreams

Filtering prefixes belonging to multi-homed downstreams on sessions with other ASes is NOT RECOMMENDED. This practice may lead to blackholing of traffic if the filter is semi-statically configured, i.e., not removed upon withdrawal of the specific prefix by a downstream. Downstreams may choose to not advertise prefixes to an upstream for a variety of reasons, including traffic engineering and Denial-of-Service attack response. Instead, operators SHOULD assign downstreams' prefixes learned from other neighbors a lower priority than those routes directly learned from downstreams. This can be done, e.g., by adding additional path prepends or using local preference settings. Please note, though, that using local preferences for this purpose may lead to a situation where a downstream is unable to perform traffic engineering apart from withdrawing a route towards its upstream in case of, e.g., a congested link in a multi-homed setup.

Even though filtering prefixes belonging to single-homed downstreams on sessions with other ASes carries less risk of immediate negative impact, it is crucial that operators coordinate closely with their downstream if such practices are applied. Otherwise, if a downstream becomes multi-homed connectivity issues may appear. Hence, assuming that other appropriate filters are in place ensuring, e.g., validity of the announcing AS and the AS-PATH, see Section 8.2, not filtering prefixes originated by downstreams on sessions with other ASes solely based on the prefix is NOT RECOMMENDED.

6.5. Outbound Filtering Prefixes Based on Learned-From

TODO: Make more general about annotating routes, also include BGP neighbor roles.

Prefixes learned from BGP neighbors may technically conform to static metrics and filter types discussed above. For example, when learning prefixes from peers and/or upstreams which have been originally announced by downstreams of an AS, it is crucial to not leak these routes to upstreams and peers in case they are preferred over those learned directly from a downstream. This may occur, for example, if a downstream uses path prepending with an upstream, while the upstream has a peering session with another AS which is also an upstream of said downstream. With the route advertised by the peer being shorter, the AS may export the learned route via the peer if:

* The outbound filter only checks whether a prefix is in a prefix list.

* The outbound filter only checks whether a prefix is in a prefix list and has been originated by the downstream AS.

To counteract this issue, outbound filtering should consider the source type, i.e., relationship to the neighbor from whom a route was originally learned.

6.5.1. Outbound Filtering Prefixes Using BGP Roles

To ensure that no prefixes leak via AS relationships (routes learned from peers or upstreams to other peers or upstreams), [RFC9234] introduces BGP roles and the BGP Only to Customer (OTC) attribute. The OTC attribute forms a tandem with ASPA, see Section 6.2.2. Operators SHOULD configure appropriate roles according to Section 3 of [RFC9234] to enable prefix filtering based on BGP relationships. Furthermore, for prefixes imported from upstreams, the OTC attribute SHOULD be set and evaluated according to [RFC9234], Section 5:

6.5.1.1. Route Import

When OTC is being used, and a route is received, it should be handled as follows:

* If OTC is set, and it is received from a customer or RS-Client, it is a routeleak and MUST be discarded.

* If OTC is set, and it is received from a peer and its value is not equal to the peer’s AS number, it is a routeleak and MUST be discarded.

* If OTC is not set, and it is received from an upstream, a peer, or an RS, it MUST be set to the AS number of the remote AS.

6.5.1.2. Route Export

* When advertising a route to a customer, peer, or (as a Routeserver) to an RS-Client, the OTC attribute MUST be set if it is not already present.

* Routes that have the OTC attribute set MUST NOT be exported to upstreams, peers, or routeservers.

6.5.2. Outbound Filtering Using Large-Communities

Despite a fall-back mechanism being implemented to support one-sided BGP roles, they must be supported by both neighbors in a BGP session to be fully effective. To completely cover an AS, all neighbors should utilize BGP roles on their sessions. Hence, if at least one neighbor does not yet utilize BGP roles, or if the operator cannot deploy BGP roles and/or use the OTC attribute on their own infrastructure, operators SHOULD additionally utilize BGP large-communities to annotate where they learned prefixes and filter accordingly on sessions where they re-announce these prefixes, see [RFC8195]. While technically possible, standard BGP communities (see [RFC1997]) SHOULD NOT be used for this purpose due to the prevalence of 32bit ASNs which can only be represented in large-communities (see [RFC8092]).

Operators SHOULD designate a large community namespace for each neighbor relationship, for example, OPERATOR_ASN:100:NEIGHBOR_ASN for upstreams, OPERATOR_ASN:101:NEIGHBOR_ASN for peers, OPERATOR_ASN:102:NEIGHBOR_ASN for downstreams, etc. These communities SHOULD cover all relationships documented in Section 3 of [RFC9234]. Additionally, if operators allow downstreams to announce more specifics than generally accepted in the GRT (see [CCR-22]), they should dedicate a large-community list to that purpose as well, to ensure they can effectively prevent re-announcements of these prefixes.

For information on how these annotations SHOULD be included in filter sets, please see Section 8.

6.6. IXP LAN Prefixes

6.6.1. IXP LAN Prefix Filtering

Within the IXP community, most IXPs prefer the IXP LAN prefix to not be advertised to the GRT ([TBD]). While some IXPs may opt to advertise the IXP LAN prefix, e.g., with the route server's ASN, operators present on an IXP MUST respect the choice of the IXP regarding the advertisement state of the IXP LAN prefix. Furthermore, e.g., the RIPE region now reached consensus on reducing the initial IXP allocation size for IPv4 (see [RIPE-804]) above their own limits on maximum prefix lengths acceptable in the GRT (see [RIPE-399] and [RIPE-532]). When a network is present on an IXP and has sessions with other IXP members over a common subnet (IXP LAN prefix), it SHOULD NOT accept exact matches or more-specific prefixes for the IXP LAN prefix from any of its external BGP neighbors. Accepting these routes may create a black hole for connectivity to the IXP LAN. To reduce the risk of accidental route leaks of IXP LAN prefixes for which the corresponding IXP opted to not have them in the GRT, operators MAY choose to use "BGP next-hop-self" on all routes learned on that IXP to not be required to distribute the IXP LAN Prefix within their IGP. Furthermore, IXPs may opt to create ROAs indicating AS0 as the only valid origin AS if they want to prevent their prefixes from being announced on the Internet.

If the IXP LAN prefix is accepted at all, it SHOULD only be accepted from the ASes that the IXP authorizes to announce it -- this will usually be automatically achieved by filtering announcements using RPKI and/or IRR database.

6.6.2. Prefixes on Routers Connected to an IXP

It is suggested (see also [APNICTRN-17]), that operators dedicate routers for connections to an IXP that SHOULD only carry routes from the ASes cone, and not a full-table or default-route. This reduces the chance of accidental route leaks and prevents other IXP members from pointing default routes via the IXP LAN to such a router. Alternative, MAY use a separate routing context (e.g. VRF) for IXP peerings, which only containins routes form the local AS cone.

6.6.3. PMTUD and the Loose uRPF Problem

Originally, in order to have PMTUD working in the presence of loose uRPF, it would be necessary that all the networks that may source traffic that could flow through the IXP have a route for the IXP LAN prefix. This relates to "packet too big" ICMP messages sent by IXP members' routers potentially being sourced using an address of the IXP LAN prefix. In the presence of loose uRPF, this ICMP packet is dropped if there is no route for the IXP LAN prefix or a less specific route covering the IXP LAN prefix.

Hence, similar to considerations in Section 3 regarding non globally routable transit networks, IXP members SHOULD ensure that "packet too big" ICMP messages sent by their routers have a source address in IP address space advertised to the GRT, e.g., the router's loopback address. Note that this issue causes service interruption in case of lost "packet too big" messages, but may also reduce debuggability in, e.g., traceroutes. If they decide to implement this behavior for all ICMP messages, operators SHOULD ensure that this address is only used for ICMP messages egressing via the interface connected to the IXP LAN. Otherwise, readability of traceroutes will be significantly reduced, as the specific interface a packet passed through is no longer visible in traceroutes.

7. Filtering and Cleaning Based on Other BGP Aspects

7.1. BGP Route Flap Dampening

The BGP route flap dampening mechanism makes it possible to give penalties to routes each time they change in the BGP routing table [RFC2439]. Initially, this mechanism was created to protect the entire Internet from multiple events that impact a single network. Studies have shown that implementations of BGP route flap dampening could cause more harm than benefit; therefore, in the past, the RIPE community has recommended against using BGP route flap dampening [RIPE-378]. Later, studies were conducted to propose new route flap dampening thresholds in order to make the solution "usable"; see [RFC7196] and [RIPE-580] (in which RIPE reviewed its recommendations). Following IETF and RIPE recommendations and using BGP route flap dampening with the adjusted configured thresholds is RECOMMENDED.

7.2. Maximum Prefixes

A spike in the number of received and imported prefixes can be a threat to the availability of a BGP speaker. Furthermore, a significant increase in the number of prefixes received from a neighbor might indicate a misconfiguration, e.g., a failure in outbound filtering for the advertising neighbor, or a failure in inbound filtering in the ingesting neighbor. Finally, it is important to limit the overall GRT growth given theoretical attacks utilizing deaggregation of IPv6 prefixes to globally exhaust routers' memory and CPU capacity (see [KIRIN-22]), the number of prefixes accepted to be originated by a neighboring AS across all BGP sessions should be limited.

7.2.1. Maximum Prefixes on a Single Session

It is RECOMMENDED to configure a limit on the number of routes to be accepted from a neighbor. The following rules are generally RECOMMENDED:

* For peers, it is RECOMMENDED to have a limit lower than the number of routes in the Internet. This will shut down the BGP session if the neighbor suddenly advertises the full table. Operators can also configure different limits for each neighbor to which they have a peering relationshipt, according to the number of routes they are supposed to advertise, plus some headroom to permit growth. However, please note that these limits may change over time. Hence, it is RECOMMENDED that both neighbors clearly communicate the number of prefixes they expect to announce to each other and agree on a way to automatically update this information in the future. At the time of writing, ([PeeringDB]) is a common source for programmatically obtaining suggested prefix limits for neighbors. In the absence of communicated prefix limits, the number of expected prefixes can be inferred from the AS-SET, see Section 6.1.2. It is RECOMMENDED to include additional headroom of 20% when utilizing an inferred prefix limit.

* From upstreams that provide full routing, it is RECOMMENDED to have a limit higher than the number of routes in the Internet. A limit is still useful in order to protect the network (and in particular, the routers' memory) if too many routes are sent by the upstream. The limit should be chosen according to the number of routes that can actually be handled by routers.

It is important to regularly review the limits that are configured as the Internet can quickly change over time. Some vendors propose mechanisms to have two thresholds: while the higher number specified will shut down the session, the first threshold will only trigger a log and can be used to passively adjust limits based on observations made on the network.

7.2.2. Continuously Monitoring Prefix Limits

When enforcing limits on the number of prefixes sent by neighbors, including upstreams, an operator may lose connectivity to one or multiple peers if, e.g., the GRT or the number of routes in the peer's cone suddenly increases. Such a sudden growth might occur due to organic effects, but could also be triggered by a malicious actor.

For example, with RPKI allowing operators to sign ROAs specifying a minimum and maximum prefix length (contrary to ROUTE/ROUTE6 objects), researchers noted that this allows deaggregation attacks

([KIRIN-22]). By configuring a ROA that cover an, e.g., /32, one can effectively authorize an AS to announce 65536 unique prefixes. Leveraging the by now large availability of free and/or cheap opportunities to obtain IPv6 upstream, a malicious party could leverage this to cause significant Internet wide route churn and GRT growth. By constantly advertising and withdrawing prefixes, churn exceeding the size of the IPv6 fulltable at the time of writing (around 200k prefixes) could be created by constantly announcing and withdrawing prefixes to upstream ASes at various PoPs.

It is therefore RECOMMENDED that operators implement continuous monitoring of all prefix limits configured on BGP sessions. That monitoring SHOULD include verifying configured prefix limits against published information on suggested prefix limits by neighbors, if available. Furthermore, the monitoring SHOULD notify operators of sudden changes in the number of received prefixes, as well as of limits being gradually approached over time.

7.3. AS_PATH Handling

This section discusses filtering AS_PATHs, as well as recommendations for AS_PATH manipulation, and which practices to avoid there.

7.3.1. AS_PATH Filtering

This section lists the RECOMMENDED practices when processing BGP AS_PATHs in addition the considerations from Section 6.

* Operators SHOULD should follow Section 6.1 to only accept 2-byte or 4-byte AS_PATHs from customers containing ASNs belonging to (or authorized to transit through) the customer. If operators cannot build and generate filtering expressions to implement this, they SHOULD consider accepting only path lengths relevant to the type of customer they have (as in, if these customers are a leaf or have customers of their own) and SHOULD try to discourage excessive prepending in such paths. This loose policy SHOULD be combined with filters for specific 2-byte or 4-byte AS_PATHs that must not be accepted if advertised by the customer, such as upstream transit providers or peer ASNs.

* Operators SHOULD NOT accept prefixes with private AS numbers in the AS_PATH unless the prefixes are from customers. In any case, operators SHOULD NOT re-export prefixes with AS_PATHs containing private AS numbers. An exception could occur when an upstream is offering some particular service like black-hole origination based on a private AS number: in that case, prefixes SHOULD be accepted. Customers should be informed by their upstream in order to put in place ad hoc policy to use such services.

* Operators SHOULD NOT accept prefixes when the first AS number in the AS_PATH is not the one of the neighbor unless the BGP session is setup towards a BGP route server [RFC7947] (for example, on an IXP) with transparent AS_PATH handling. In that case, this verification needs to be deactivated, as the first AS number will be the one of an IXP member, whereas the neighbor's AS number will be the one of the BGP route server.

* Operators SHOULD NOT advertise prefixes with a nonempty AS_PATH unless they are either announcing prefixes from the GRT to downstreams, or if the whole AS_PATH is within their cone.

* Operators SHOULD NOT advertise prefixes with upstream AS numbers in the AS_PATH to any AS except to those to which they provide upstream transit.

* Private AS numbers are conventionally used in contexts that are "private" and SHOULD NOT be used in advertisements to BGP neighbors that are not party to such private arrangements, and they SHOULD be stripped when received from BGP neighbors that are not party to such private arrangements. Additionally, operators MAY decide to not accept prefixes with private AS numbers in their AS_PATH at all.

* Operators SHOULD NOT accept their own AS number in the AS_PATH by overriding BGP's default behavior. When considering an exception, the impact (which may be severe on routing) should be evaluated carefully.

* Overly long AS_PATH, i.e., longer than 64 entries, may cause issues for some older routing hardware. Hence, operators SHOULD NOT use excessive prepending when advertising prefixes. Excessive prepending is defined as any prepending that leads to an AS_PATH exceeding 64 entries across the GRT. Additionally, it is RECOMMENDED that operators filter any prefix advertised with an AS_PATH of more than 64 entries.

7.3.2. AS_PATH Manipulation

This section lists the RECOMMENDED practices when manipulating BGP AS_PATHs, to limit chances of accidentally producing AS_PATHs that would have to be filtered by neighbors according to Section 7.3.1.

Some BGP implementations offer various advanced AS_PATH manipulation features, such as overriding or rewriting a part of the AS_PATH. For instance, a very commonly used mechanism is the so-called "AS Override" feature, primarily intended for use in MPLS L3 VPNs, where the customer's AS number is overridden with the provider's AS number,

to allow site-to-site communication where both customer sites use the same AS number. Some vendors went even further, offering a possibility to fully rewrite or even delete the AS_PATH Attribute from incoming or outgoing BGP Update messages.

Furthermore, AS_PATH filtering is an option when ASN renumbering is done. Such an operation is common, and mechanisms exist to allow smooth ASN migration [RFC7705]. The usual migration technique, local to a router, consists of modifying the AS_PATH so it is presented to a neighbor with the previous ASN, as if no renumbering was done. This makes it possible to change the ASN of a router without reconfiguring all eBGP neighbors at the same time (as that operation would require synchronization with all neighbors attached to that router). During this renumbering operation, the rules described above may be adjusted.

In principle, use of any AS_PATH modification mechanism except AS_PATH prepend in the public Internet SHOULD be avoided at all. Also, as discussed already, AS_PATH prepends SHOULD NOT be excessive. Operators are RECOMMENDED to not prepend more than five times. The "AS Override" feature MAY still be used in closed environments, such as VPNs not directly exchanging any NLRIs with the Internet. AS_PATH rewriting/deleting SHOULD be avoided. Especially the practice of providing upstream to customers using a private ASN and then using rewriting on either side is strongly NOT RECOMMENDED.

7.4. Next-Hop Filtering

When establishing sessions via a shared network, like an IXP, BGP can advertise prefixes with a third-party next hop, thus directing packets not to the neighbor announcing the prefix but somewhere else.

This is a desirable property for BGP route-server setups [RFC7947], where the route server will relay routing information but has neither the capacity nor the desire to receive the actual data packets. So, the BGP route server will announce prefixes with a next-hop setting pointing to the router that originally announced the prefix to the route server.

In direct sessions between ASes via an IXP LAN, this is undesirable, as one of the neighbors could trick the other one into sending packets into a black hole (unreachable next hop) or to an unsuspecting third party who would then have to carry the traffic. Especially for black-holing, the root cause of the problem is hard to see without inspecting BGP prefixes at the receiving router of the IXP.

Therefore, an inbound route policy SHOULD be applied on direct sessions via an IXP LAN in order to set the next hop for accepted prefixes to the BGP neighbor's IP address (belonging to the IXP LAN) that sent the prefix (which is what "next-hop-self" would enforce on the sending side).

This policy SHOULD NOT be used on sessions with route-servers or on sessions where operators intentionally permit the other side to send third-party next hops.

This policy also SHOULD be adjusted if the best practice of Remote Triggered Black Holing (aka RTBH as described in [RFC6666]) is implemented. In that case, operators would apply a well-known BGP next hop for routes they want to filter (if an Internet threat is observed from/to this route, for example). This well-known next hop will be statically routed to a null interface. In combination with a unicast RPF check, this will discard traffic from and toward this prefix. BGP speakers can exchange information about black holes using, for example, particular BGP communities, see [RFC6666]. Operators could propagate black-hole information to their neighbors using an agreed-upon BGP community: when receiving a route with that community, a configured policy could change the next hop in order to create the black hole.

7.5. BGP Community Scrubbing

For BGP, BGP communities [RFC1997], extended BGP communities [RFC4360], and BGP large-communities [RFC8092] have been defined for additional inband signaling. In the remainder of this section, we use the term 'BGP communities' to mean [RFC1997] and [RFC8092] BGP communities alike, while we explicitly refer to [RFC4360] as 'extended BGP communities'.

Communities are useful in iBGP and eBGP alike. For example, BGP communities are often used by operators to allow neighbors to signal additional traffic engineering requirements, e.g., asking an upstream not to announce a specific NLRI to one of its neighbors. Similarly, BGP communities are essential for proper filtering of downstreams' prefixes in the absence of ASPA/OTC. While usually more focused on L2VPN and L3VPN scenarios, extended BGP communities may also find specific use when interacting with external neighbors, see, e.g., [RFC4364], Inter-AS VPN Option B. Hence, while they should generally should not act transitively, operators SHOULD nevertheless ensure that these communities do not accidentally leak.

However, as they may carry instructive information, external unauthorized neighbors should not be allowed to send NLRI with AS specific BGP communities. Similarly, internally used BGP communities

may reveal non-public information or cause disturbance in misconfigured networks. The in- and outbound filtering rules for all forms of BGP communities in Section 7.5.1 and Section 7.5.2 are RECOMMENDED

Additionally, please note the following general recommendations for community scrubbing:

* Networks administrators SHOULD NOT remove other BGP communities applied on received routes (BGP communities not removed after application of the previous statement). In particular, they SHOULD keep original BGP communities when they apply a community.

* Operators SHOULD NOT remove the no-export community, as it is usually announced by their neighbor for a certain purpose.

* Network operators SHOULD NOT remove RTBH related BGP communities if sent by the customer for a prefix of routable size.

* In case where BGP communities / extended BGP communities / BGP large-communities specific to the own AS are not scrubbed, it is strongly RECOMMENDED to maintain a strict allow-list of permissible BGP communities, and still scrub those BGP communities not contained in that list, even if these BGP communities are not in use.

7.5.1. Inbound BGP Community Scrubbing

* Operators SHOULD scrub inbound BGP communities with their ASN in the high-order bits, unless they have been documented and communicated to neighbors to be used as a signaling mechanism. If a received NLRI contains an excessive amount of BGP communities, i.e., more than 100, operators MAY truncate the list of BGP communities. When truncating BGP communities, operators SHOULD prioritize retaining BGP communities of their neighbors.

* Extended BGP communities (see [RFC4360]) received from external neighbors SHOULD be scrubbed. However, there are operational circumstances where it MAY be reasonable to accept extended BGP communities from neighbors, see, e.g., [RFC4364], Inter-AS VPN Option B.

* When known, BGP communities used to signal RPKI ROV state (see Section 6.2.1) received from eBGP neighbors MUST be scrubbed. This is done to prevent BGP update storms in case a neighbor looses the connection to its validator, changing the validation state of previously valid NLRI, and thereby the applied community for those NLRI, triggering an update message. For further details on why BGP communities MUST NOT be used to signal RPKI ROV state, please see [I-D.ietf-sidrops-avoid-rpki-state-in-bgp].

7.5.2. Outbound BGP Community Scrubbing

* Operators SHOULD scrub outbound BGP communities with their ASN in the high-order bits, unless their semantics have been documented and communicated to neighbors. Even if the semantics of BGP communities has been documented, operators SHOULD be mindful of the number of BGP communities they add to NLRI. When sending NLRI to neighbors, operators SHOULD limit the number of BGP communities they communicate to the outside, i.e., not overload NLRI with an excessive amount of BGP communities by outbound filtering all non-externally useful BGP communities they added for use within their own network. Furthermore, when defining BGP communities, operators MUST be careful not to define redundant BGP communities or using multiple BGP communities to express properties that could sensibly be represented with a single community.

* Extended BGP communities (see [RFC4360]) SHOULD NOT be sent to eBGP neighbors. However, there are operational circumstances where it MAY be reasonable to send extended BGP communities to neighbors, see, e.g., [RFC4364], Inter-AS VPN Option B.

* Operators MUST NOT use BGP communities to signal RPKI ROV state, see [I-D.ietf-sidrops-avoid-rpki-state-in-bgp]. If an operator is still in a migration phase discontinuing this practice, they MUST scrub any such community they use for signaling RPKI-ROV state before sending NLRI to their external neighbors. This is done to prevent the propagation of BGP update storms in case one of their BGP speakers looses the connection to its validator, changing the validation state of previously valid NLRI, and thereby the applied community for those NLRI, triggering an update message. For further details on why BGP communities MUST NOT be used to signal RPKI ROV state, please see [I-D.ietf-sidrops-avoid-rpki-state-in-bgp].

7.6. Handling BGP Attributes

While there is a list of well-known and defined transitive BGP attributes, operators sometimes accidentally or intentionally use undocumented BGP attributes. Similarly, newly introduced attributes may not yet be known to a specific implementation.

In general, unknown transitive BGP attributes SHOULD NOT be filtered. However, sometimes bugs may occur in implementations that require filtering or correction of attributes on the border to protect BGP speakers before a patch for the implementation is available.

This section documents practices for scrubbing and normalizing BGP attribute related data in received NLRI.

7.6.1. BGP Attribute Scrubbing

Over the past years several instances of network disruptions due to routers being unable to process specific BGP attributes were encountered. As such, operators MAY opt to temporarily scrub specific BGP attributes known to cause service disruptions on their infrastructure. Operators SHOULD NOT scrub unknown transitive attributes in general.

However, while being a very useful tool, BGP attribute scrubbing features may cause undesired effects and sometimes even large-scale outages as well. Therefore, they MUST NOT be used as a permanent solution, but only as a last-resort temporary workaround. Furthermore, removing mandatory BGP attributes and optional attributes commonly used in the Internet, such as AS_PATH, Communities, MED etc. may have a significant negative impact beyond an operator’s own AS. Hence, it is RECOMMENDED that such attributes are never removed when importing NLRI.

When sending NLRI to external neighbors, operators SHOULD avoid sending not yet standardized or only internally used attributes, i.e., scrub attributes they added which are not in public use before exporting NLRI.

7.6.2. BGP Attribute Header Correction

BGP attributes are stored within BGP UPDATE messages as a vector of Type-Length-Value (TLV) fields. The Attribute Type field contains a set of control bits, such as the Optional Bit (set to 1 for Optional Attributes and 0 for Well-Known), the Transitive Bit (specifying whether the attribute is Transitive, i.e., should be propagated outside the local AS or, Non-Transitive, i.e., should not be propagated outside the local AS) etc. Initially, [RFC4271] mandated

that a BGP speaker tears down a BGP session when receiving even a single UPDATE message containing a malformed combination of Attribute TLV headers. However, [RFC7606] allows BGP implementers to optionally add features providing self-correction of malformed attributes in a limited number of cases.

Operators MAY use such, self-correcting mechanisms for BGP Attribute TLV headers. However, they SHOULD consider the operational impact such features have, SHOULD monitor for cases where such self-correction is necessary, and SHOULD follow up on such cases to ensure that root-causes are identified and addressed.

7.7. Preventing MED Oscilation

As documented in [RFC3345], the use of BGP Route Reflection [RFC4456] and BGP Confederation [RFC5065] can lead to route oscilation, especially in conjunction with the MULTI_EXIT_DISC (MED) attribute (see [RFC4271]). If BGP route oscilation occurs, routes may be blackholed if dampening is implemented by neighbors, or individual BGP speakers may become overloaded, further aggravating the oscilation issue.

Hence, operators SHOULD familiarize themselves with [RFC7964], which describes methods and approaches to counteract MED related route-oscilation. Operators SHOULD carefully evaluate their network’s requirements and implement the practices documented in [RFC7964] as appropriate.

7.8. Behavior when Connecting via an IXP

IXPs are an essential aspect of the modern Internet, and contribute to keeping local traffic local. As such, IXP fabrics often handle a significant amount of traffic, providing challenges for traffic engineering. Hence, this section documents best practices when connecting to an IXP that inflict on the reliability of the global routing ecosystem.

7.8.1. Not Setting a Higher LOCAL_PREF for NLRI received via an IXP

Given that traffic forwarded via an IXP can be more cost-efficient than sending that same traffic via an upstream, many operators set a higher LOCAL_PREF for NLRI received via an IXP. This means that all traffic from the AS and all members of its cone routing via this AS will preferentially be routed via these paths (see [RFC4271]), effectively overriding the effect of AS_PATH prepending, see also [I-D.ietf-grow-as-path-prepending].

As noted in [I-D.ietf-grow-as-path-prepending], setting a higher LOCAL_PREF on IXP links means that neighbors on the IXP can no longer use AS_PATH prepending for, e.g., traffic engineering. More crucially, it prevents operators from draining traffic flowing via an IXP when necessary, e.g., prior to a scheduled maintenance. Especially when NLRI are exchanged via an RS, simply terminating the session is usually not possible without also impacting other neighbors.

Hence, operators SHOULD NOT set a higher local preference for NLRI received via an IXP RS. Instead, other non-transitive methods, e.g., setting a corresponding MED on imported routes, should be preferred.

If, when trying to drain traffic on an IXP link via AS_PATH prepending of NLRI sent to the RS, an operator encounters an IXP member ignoring these prepends, they may be able to selectively widthdraw routes from being announced to that member by using communities documented by the IXP to prevent the RS from exporting their NLRI to that specific IXP member.

7.8.2. Honoring GSHUT on an IXP

Graceful BGP Session Shutdown (GSHUT) as defined in [RFC8326] is a formalized method for draining traffic from sessions gracefully before, e.g., maintenance. However, while AS_PATH prepending does not have to be supported by two neighbors, GSHUT requires all neighbors to implement it by implementing a policy that assigns a lower LOCAL_PREF to NLRI matching the GRACEFUL_SHUTDOWN BGP community.

GSHUT is a more effective method of traffic draining than, e.g., AS_PATH prepending. Hence, in general, GSHUT SHOULD be supported on all eBGP sessions. However, as an IXP member, when ignoring the previous recommendation and setting a higher LOCAL_PREF for sessions via an IXP LAN, GSHUT MUST be supported.

8. Prefix Filtering Recommendations

8.1. Prefix Filter Implementation Considerations

Besides the overall generation of prefix filters and to which relationships these should be applied, the way how these can be implemented needs to be considered.

8.1.1. Implicit Policies and Default Behavior

Almost all BGP implementations have specific default behavior, including behavior when reaching the end of a policy, behavior when no policy is defined (even though [RFC8212] now requires a default-deny in the absence of policy), etc. However, default behavior and matching characteristics may differ between vendors and implementations. Implicitly relying on vendor-specific default behavior can pose issues if a network operator migrates from one vendor to the other, or when operating a mixed-vendor environment. Furthermore, implicit defaults may change, requiring intervention by operators. Therefore, it is RECOMMENDED that operators create explicit policy statements, even for behavior covered by defaults. Such a practice helps simplifying automation of router configurations, and prevents incidents due to changing or differing implicit defaults, especially when migrating between vendors and in interoperability scenarios.

8.1.2. Order of Prefixfilters

BGP is a policy-based routing protocol with import/export policies controlling advertisements/acceptance of NLRI (see [RFC4272] Sec. 9.1), and BGP sessions without policy being applied should default to a deny-all stance (see [RFC8212]). The specific implementation of import/export policies varies between vendors in terms of complexity and naming, from basic prefix-based / AS_PATH-based filters, to complex IF-THEN-like policy structures (typical names are: "route maps", "route policies", "policy statements").

Independent of the implementation, all BGP policies consist of one or more rule sets, that are executed in a sequence, one after another. The first rule set will scan the complete content of Adj-RIBs-in or Adj-RIBs-out; NLRIs permitted by a rule set will be passed to subsequent rule sets, while denied prefixes are discarded.

Policies SHOULD avoid computationally expensive setups, or setups of rules that apply computation to NLRI that will subsequently be discarded. Hence, the more prefixes a rule is likely to discard, the earlier it SHOULD be evaluated.

To further illustrate this, you can find an (incomplete) example for a simple inbound filter for a session with a neighbor in a peer relationship who has 60 prefixes in its cone creating additional load. We assume that, accidentally, an IPv4 fulltable of 1,000,000 entries is being sent. 2,500 NLRI contain unregistered/private AS numbers, 500 NLRI relate to bogon prefixes, and 5,000 NLRI are RPKI invalid, with none of the routes in the neigbor’s cone falling in any of these categories. The number of operations per line are given in parentheses.

* 1. Add community imported from peer (1,000,000)

* 2. Add community imported in LOCATION (1,000,000)

* 3. Reject NLRI with private AS numbers in the AS path (1,000,000)

* 4. Reject bogon prefixes (997,500)

* 5. Reject RPKI invalid NLRI (997,000)

* 6. Accept only prefixes in the neighbor’s cone (992,000)

In this list, rules 1-5 will be executed for most NLRI seen from the neighbor. In total, 5,986,500 operations are executed on the received NLRI, even though ultimately only 60 prefixes should be imported.

While most hardware implementations of BGP speakers should be sufficiently equipped with resources to handle such individual spikes, practice shows that operators can not always use BGP speakers with an abundance of resources. Furthermore, even more well equipped platforms may suffer if multiple neighbors coordinate and utilize this mechanic to induce load. Ultimately, it is also desirable to reduce unnecessary computation independent of security considerations.

Hence, it is RECOMMENDED that operators structure rulesets in a way that prioritized early decisions on the majority of routes. For the example above, this would mean, again noting the number of operations per rule:

* 1. Reject prefixes not in the neighbor’s cone (1,000,000)

* 2. Reject bogon prefixes (60)

* 3. Reject NLRI with private AS numbers in the AS path (60)

* 4. Reject RPKI invalid NLRI (60)

* 5. Add community imported from peer (60)

* 6. Add community imported in LOCATION (60)

Overall, this reduces the number of operations for our hypothetical full-table from 5,986,500 operations to 1,000,300. Similar effects can occur when not filtering on, e.g., the OTC attribute first when sending prefixes to customers.

8.1.3. Ensuring Consistency when Changing Prefixfilters

As discussed in Section 8.1.2, many BGP implementations use a sequential order for applying different prefix filters to ingested routes. However, at the same time, several implementations do not perform atomic operations when applying rules. This means that, especially on resource constraint BGP speakers or BGP speakers under load consistency of a ruleset may be lost during a rule-set update.

For example, consider the following simplified export rule-set towards a peer:

* 1. Reject prefixes learned from upstreams

* 2. Reject bogon prefixes

* 3. Reject RPKI invalid NLRI

* 4. Accept all remaining prefixes

If one now wants to swap the order of Rule 2 and 3, an implementation applying rule updates not atomically would proceed as follows:

* 1. Delete Rule 2, Reject bogon prefixes

* 2. Add Rule 2, Reject RPKI invalid NLRI

* 3. Delete Rule 3, Reject RPKI invalid NLRI

* 4. Add Rule 3, Reject bogon prefixes

During the timeframe between the execution of Step 1 and 2, an NLRI for a bogon prefix would be passed by the filter. While, technically, this timeframe should be negligibly small, a loaded control plane my create unexpected overhead allowing prefixes that should be filtered to pass. Similarly, an error during the application of a ruleset, making the application stop after the execution of Step 1 may have a similar effect if rule-set changes are not atomic.

Hence, it is RECOMMENDED that operators assess whether the application of changes to rule-sets on their BGP speakers is atomic. If it is not atomic, operators SHOULD take special care in drafting rule-set updates concerning inconsistent state that could be created by a delayed or incomplete update. If no atomicity is provided by the BGP speaker, and the load-conditions are uncertain, operators SHOULD consider creating a new complete rule-set with the desired changes, and then changing the referenced rule-set for a given neighbor instead of updating an existing rule-set in-place. Naturally, after the new rule-set has been activated, the old rule-set should be deleted.

8.1.4. Ensuring Idempotency for Prefixfilter Changes

As prefix filters are changed regularly, idempotency is essential when issuing automated updates of prefix filters. Specifically, prefix filters SHOULD NOT be generated on routers itself.

Instead, filter lists SHOULD be generated on dedicated systems. These systems SHOULD ensure the idempotency of changes to filters applied to routers, i.e., they should only deploy a policy, if the policy changed. This ensures no unnecessary regular load is placed on the control plane of BGP speakers.

8.1.5. Ruleset Size Considerations

Some BGP speaker implementations, and especially older BGP speakers, are restrained in terms of the number of prefixes and rules they can apply. A common reaction of operators in such cases is reducing the number of filters applied on sessions. Even though it is NOT RECOMMENDED to aggregate prefix lists for filtering, operators SHOULD consider aggressive aggregation of prefix filter lists to restrict the perfixes accepted by neighbors if the alternative is not using filters at all.

Another approach that MAY be a suitable rule-set creation approach for downstreams and peers is offline validation. In that case, a dedicated system regularly, e.g., every two hours, obtains the list of prefixes advertised by a given peer or downstream. That list is then validated according to the applicable section below. Subsequently, instead of using a full representation of the neighbors cone, a condensed prefix list matching the aggregate of the exact prefixes announced is generated and deployed to the BGP speaker. While this increases the timeframe for newly added prefixes to be accepted, and may be unsuitable for, e.g., DDoS defense services, it can also reduce the size of prefix lists significantly.

8.1.6. Ruleset Generation Failure

As noted in Section 6.1.3, the creation and application of filter rules should be automated to reduce the margin for error and misconfigurations. Nevertheless, the regeneration of filter rules may fail.

Before applying a generated ruleset, an operator should check it for obvious errors and potentially require manual intervention to remediate the issue. Examples include a ruleset for a neighbor suddenly significantly increasing or decreasing in size, or being empty.

In case of such a failure, each administrator MAY decide which actions they will take. Options include re-using the previously active rule set, or either accepting or rejecting all routes depending on routing policy. Generally accepting all routes during that time frame could break BGP routing security. However, rejecting them might re-route too much traffic towards upstreams, and could cause more harm than accepting invalid prefixes. Similarly, reusing the previously active rule set may lead to prefixes being wrongfully accepted or rejected, despite on a smaller scale than for a general accept or reject decision.

Hence, to still provide sufficient protection for an individual AS experiencing issues with rule generation, and therefore deciding to deviate to more permissive inbound filters, it is strongly RECOMMENDED that all BGP speakers in general employ inbound and outbound filtering as described in this document.

8.2. Prefix Filtering Recommendations in Full Routing Networks

For networks that have the full Internet BGP table, policies should be applied on each BGP neighbor for received and advertised routes. It is RECOMMENDED that each Autonomous System configures rules for advertised and received routes at all its borders, as this will protect the network and its neighbors even in case of misconfiguration. The most commonly used filtering policy is proposed in this section and uses prefix filters defined in Section 5, Section 6, and Section 7.

8.2.1. Filters with Internet Peers

8.2.1.1. Inbound Filtering

Inbound filtering on sessions with peers does not only ensure that an operator does not ingest maliciously or wrongfully advertised routes, but also serves as an additional safety net in case of unintentional misconfigurations. For inbound filters with peers, the following rules SHOULD be applied in the given order to limit resource use on filter application (see Section 8.1).

The RECOMMENDED filters ensure advertisements strictly conform to what is declared in routing registries (Section 6.1). Warning is given as registries are not always accurate (prefixes missing, wrong information, etc.). This varies across the registries and regions of the Internet. Hence, before applying this policy, the reader SHOULD check the impact on the filter and make sure no prefixes are filtered that should actually be accepted.

* 1. All prefixes not in the neighbor’s cone based on IRR data (Section 6.1)
* 2. Prefixes not allocated by IANA (IPv6 only) (Section 5.2.1)
* 3. RPKI invalid prefixes (Section 6.2.1)
* 4. Prefixes with an invalid AS path (ASPA) (Section 6.2.2)
* 5. Prefixes with an invalid AS path (longer than 64 entries) (Section 7.3.1)
* 6. Prefixes with an invalid AS path (containing private or reserved AS numbers) (Section 7.3.1)
* 7. Prefixes belonging to the local AS (Section 6.3)
* 7.1. Optionally: Reprioritizing prefixes belonging to the local AS’ cone instead of filtering them (Section 6.4)
* 8. Prefixes that are not globally routable (Section 5.1)
* 9. IXP LAN prefixes of IXPs the local AS is connected to (Section 6.6)
* 10. The default route (Section 5.4)
* 11. Routes not matching the neighbor’s next-hop (Section 7.4)
* 12. Routes that are too specific or too unspecific (Section 5.3)

Note that Rule 12 MAY be formulated as an acceptance rule, i.e., accepting all prefixes that are between a /8 and a /24 for IPv4 and between a /16 and a /48 for IPv6.

Additionally, Rule 11 MUST NOT be set for sessions with IXP routeservers, while it SHOULD be set on direct sessions via IXP LANs (see Section 6.6).

If BGP roles are used, the OTC attribute should be set according to [RFC9234].

8.2.1.2. Outbound Filtering

The configuration should ensure that only appropriate prefixes are sent, i.e., prefixes a neighbour would not need to filter based on Section 8.2.1.1. These can be, for example, prefixes belonging to both the network in question and its downstreams. This can be achieved by using BGP communities, AS paths, or both.

Also, it may be desirable to add the following filters before any further policy to avoid unwanted route announcements due to bad configuration:

* 1. All prefixes not in the local AS' cone based on IRR/Internal data (Section 6.1)
* 2. All prefixes with the OTC attribute set evaluated according to [RFC9234] (Section 6.5.1)
* 3. Prefixes not allocated by IANA (IPv6 only) (Section 5.2.1)
* 4. RPKI invalid prefixes (Section 6.2.1)
* 5. Prefixes with an invalid AS path (ASPA) (Section 6.2.2)
* 6. Prefixes with an invalid AS path (longer than 64 entries) (Section 7.3.1)
* 7. Prefixes with an invalid AS path (containing private or reserved AS numbers) (Section 7.3.1)
* 8. Prefixes that are not globally routable (Section 5.1)
* 9. The default route (Section 5.4)
* 11. Routes not intended for re-export (Section 6.5)

* 12. Routes not listing the BGP speaker as the next-hop (Section 7.4)

* 13. Routes that are too specific or too unspecific (Section 5.3)

If it is possible to list the prefixes to be advertised, then just configuring the list of allowed prefixes and denying the rest is technically sufficient. Nevertheless, to ensure robustness in case of failure, especially for manually operated BGP speakers, it is RECOMMENDED that operators apply the full rule-set.

Note that Rule 12 is technically not necessary for eBGP. However, in some rare cases misconfigurations or implementation errors may occur, especially for sessions with a neighbor via an IXP LAN (directly or indirectly), where the implementation on the BGP speaker might export routes with a non-local next-hop. While Rule 12 could prevent disturbance in such cases, the likelihood of such events is sufficiently low that operators MAY opt to not use Rule 12.

8.2.2. Filters with Customers

8.2.2.1. Inbound Filtering

From customers, only customer prefixes SHOULD be accepted, all others SHOULD be discarded. However, additionally, an operator should ensure that prefixes announced by customers also conform to best practices in terms of other BGP aspects (AS path, IRR compliance, RPKI etc.) Not doing so might lead to intransparent failures when the customer is able to export routes to the upstream, but these are then not ingested by the upstream’s neighbors. Applying filtering close to the source ensures better debugability for such issues.

* 1. All prefixes not in the neighbor’s cone based on IRR data (Section 6.1)

* 2. Prefixes not allocated by IANA (IPv6 only) (Section 5.2.1)

* 3. RPKI invalid prefixes (Section 6.2.1)

* 4. Prefixes with an invalid AS path (ASPA) (Section 6.2.2)

* 5. Prefixes with an invalid AS path (longer than 64 entries) (Section 7.3.1)

* 6. Prefixes with an invalid AS path (containing private or reserved AS numbers) (Section 7.3.1)

* 7. Prefixes belonging to the local AS (Section 6.3)

* 8. Prefixes that are not globally routable (Section 5.1)

* 9. IXP LAN prefixes of IXPs the local AS is connected to (Section 6.6)

* 10. The default route (Section 5.4)

* 11. Routes not matching the neighbor’s next-hop (Section 7.4)

* 12. Routes that are too specific or too unspecific (Section 5.3)

Technically, the inbound policy with end customers is pretty straightforward: only customer prefixes SHOULD be accepted, all others SHOULD be discarded. For smaller downstreams, the list of accepted prefixes can be manually specified, after having verified that they are valid. This validation can be done with the appropriate IP address management authorities. For larger downstreams, an approach as documented in Section 8.1.5 MAY also be suitable.

Additionally, Rule 11 MUST NOT be set in the rare case of an IXP routeserver providing upstream (see [CommunityIX]), while it SHOULD be set when providing upstream to a customer with a direct session via IXP LANs (see Section 7.4).

8.2.2.2. Outbound Filtering

The outbound policy with customers may vary according to the routes the customer wants to receive. In the simplest possible scenario, the customer may want to receive only the default route; this can be done easily by applying a filter with the default route only.

In case the customer wants to receive the full routing table (if it is multihomed or if it wants to have a view of the Internet table), the following filters SHOULD be applied on the BGP session:

* 1. Prefixes not allocated by IANA (IPv6 only) (Section 5.2.1)

* 2. RPKI invalid prefixes (Section 6.2.1)

* 3. Prefixes with an invalid AS path (ASPA) (Section 6.2.2)

* 4. Prefixes with an invalid AS path (longer than 64 entries) (Section 7.3.1)

* 5. Prefixes with an invalid AS path (containing private or reserved AS numbers) (Section 7.3.1)

* 6. Prefixes that are not globally routable (Section 5.1)

* 7. The default route (Section 5.4)

* 8. Routes not intended for re-export (Section 7.5)

* 9. Routes not listing the BGP speaker as the next-hop (Section 7.4)

* 10. Routes that are too specific or too unspecific (Section 5.3)

In some cases, the customer may desire to receive the default route in addition to the full BGP table. This can be done by the provider removing the filter for the default route in Rule 7. As the default route may not be present in the routing table, operators SHOULD only originate it for neighbors that requested it.

Note that Rule 9 is technically not necessary for eBGP. However, in some rare cases misconfigurations or implementation errors may occur, especially on sessions with a neighbor via an IXP LAN (directly or indirectly), where the implementation on the BGP speaker might export routes with a non-local next-hop. While Rule 9 could prevent disturbance in such cases, the likelihood of such events is sufficiently low that operators MAY opt to not use Rule 9.

8.2.3. Filters with Upstream Providers

8.2.3.1. Inbound Filtering

If the upstream provider is supposed to announce only the default route, a simple filter will be applied to accept only the default prefix and nothing else.

If the full routing table is desired from the upstream, the prefix filtering below should be applied:

* 1. Prefixes not allocated by IANA (IPv6 only) (Section 5.2.1)

* 2. RPKI invalid prefixes (Section 6.2.1)

* 3. Prefixes with an invalid AS path (ASPA) (Section 6.2.2)

* 4. Prefixes with an invalid AS path (longer than 64 entries) (Section 7.3.1)

* 5. Prefixes with an invalid AS path (containing private or reserved AS numbers) (Section 7.3.1)

* 6. Prefixes belonging to the local AS (Section 6.3)

* 6.1. Optionally: Reprioritizing prefixes belonging to the local AS' cone instead of filtering them (Section 6.4)

* 7. Prefixes that are not globally routable (Section 5.1)

* 8. IXP LAN prefixes of IXPs the local AS is connected to (Section 6.6)

* 9. The default route (Section 5.4)

* 10. Routes not matching the neighbor's next-hop (Section 7.4)

* 11. Routes that are too specific or too unspecific (Section 5.3)

Sometimes, the default route (in addition to the full BGP table) can be desired from an upstream provider. In that case, Rule 9 MAY be removed.

Additionally, Rule 10 MUST NOT be set in the rare case of an IXP routeserver providing upstream (see [CommunityIX]), while it SHOULD be set when receiving upstream on a direct session via IXP LANs (see Section 7.4).

8.2.3.2. Outbound Filtering

In general, at least the same outbound filters as applied for Internet peers (Section 8.2.1.2) SHOULD be applied for upstreams. However, different policies could be applied if a particular upstream should not provide transit to all prefixes.

When deciding to selectively announce prefixes to an upstream, it is important to be mindful of potential issues with uRPF in case of asymmetric traffic flows. In certain strict uRPF cases traffic for a prefix may be blackholed if the outbound route to a destination traverses one upstream, while the prefix is only announced to another upstream. It is RECOMMENDED that operators do not implement strict uRPF solely based on visible or selected routes received from a peer. Instead, either an approach similar to the cone determination (see Section 6.1), or loose uRPF should be used (see [RFC8704]).

8.3. Prefix Filtering Recommendations for Leaf Networks

8.3.1. Inbound Filtering

The leaf network will deploy the filters corresponding to the routes it is requesting from its upstream. If a default route is requested, a simple inbound filter can be applied to accept only the default route (Section 5.4). If the leaf network is not capable of listing the prefixes because there are too many (for example, if it requires the full Internet routing table), then it SHOULD follow the filter recommendations in Section 8.2.3.1.

8.3.2. Outbound Filtering

A leaf network will most likely have a very straightforward policy: it SHOULD only announce its local routes. For additional scrutiny, it is also RECOMMENDED that leaf ASes follow the recommendations in Section 8.2.3.2 to avoid announcing invalid routes to its upstream provider, and for additional resillience if the network later becomes multihomed.

8.4. Prefix Filtering Recommendations for Mutual Transit

If a mutual-transit relationship as defined in [I-D.ietf-sidrops-aspa-verification] exists between two neighbors, each neighbor SHOULD follow the recommendations in [I-D.ietf-sidrops-aspa-verification]. Furthermore, it is RECOMMENDED that both parties in a mutual-transit relationship take additional precautions to ensure that they do not export routes the other neighbor learned from their own upstreams to peers and upstreams of their own. This can be accomplished, e.g., via annotations of imported routes (see Section 6.5.2) differing based on a filter representing the neighbor’s cone (see Section 6.1).

8.5. Prefix Filtering Recommendations for iBGP

While iBGP sessions should generally be trusted, it is good practice to implement basic filters on iBGP sessions carrying external NLRI as well. It is RECOMMENDED that other internal routing signalling is handled by a dedicated IGP or via a dedicated VRF/Routing Domain (see [NSRC-17]) to reduce the likelihood of internal routes leaking due to misconfigurations, with routes appropriately annotated to not be exported (see Section 6.5). Doing so ensures that, e.g., localized misconfigurations, e.g., leaked (internal) routes, remain localized to a region or PoP, instead of spreading throughout the whole AS and to external neighbors, ideally limiting their impact.

If the iBGP mesh/sessions via a route reflector (see [RFC4456]) of BGP speakers connected to external neighbors only carries external NLRI, the following filters are RECOMMENDED, the following rules should be applied when importing or exporting routes.

* 1. Prefixes not announced by the local AS not carrying an annotation (either via the OTC attribute evaluated according to [RFC9234], or a large community as outlined in Section 6.5)

* 2. Prefixes not allocated by IANA (IPv6 only) (Section 5.2.1)

* 3. RPKI invalid prefixes (Section 6.2.1)

* 4. Prefixes with an invalid AS path (ASPA) (Section 6.2.2)

* 5. Prefixes with an invalid AS path (longer than 64 entries) (Section 7.3.1)

* 6. Prefixes with an invalid AS path (containing private or reserved AS numbers) (Section 7.3.1)

* 7. Prefixes that are not globally routable (Section 5.1)

* 9. The default route (Section 5.4)

* 11. Routes that are too specific or too unspecific (Section 5.3)

Depending on the local circumstances an operator MAY deviate from this suggestion and refrain from using individual rules. For example, if the AS ingests a default route from at least one neighbor, Rule 9 should be omitted. Similarly, when allowing downstreams to announce hyperspecifics (see Section 6.5.2), Rule 10 SHOULD be omitted.

While an operator MAY opt to not use any of the suggested rules, it is RECOMMENDED that at least Rule 1 is applied to iBGP sessions to ensure absent annotations do not propagate and cause route leaks.

9. IANA Considerations

This document does not require any IANA actions.

10. Security Considerations

This document is entirely about BGP operational security. The document understands security not only as resilience against attacks, but also in the context of safety, i.e., ensuring that systems remain operational and behave as expected even if individual components fail or are mishandled. It depicts best practices that one should adopt to secure BGP infrastructure: protecting BGP routers and BGP sessions, adopting consistent BGP prefix and AS path filters, and configuring other options to secure the BGP network.

This document does not aim to describe specific BGP implementations, their potential vulnerabilities, or ways they handle errors. It does not detail how protection could be enforced against attack techniques using crafted packets.

11. References

11.1. 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, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC2439] Villamizar, C., Chandra, R., and R. Govindan, "BGP Route Flap Damping", RFC 2439, DOI 10.17487/RFC2439, November 1998, <https://www.rfc-editor.org/info/rfc2439>.

[RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <https://www.rfc-editor.org/info/rfc4271>.

[RFC4456] Bates, T., Chen, E., and R. Chandra, "BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)", RFC 4456, DOI 10.17487/RFC4456, April 2006, <https://www.rfc-editor.org/info/rfc4456>.

[RFC5065] Traina, P., McPherson, D., and J. Scudder, "Autonomous System Confederations for BGP", RFC 5065, DOI 10.17487/RFC5065, August 2007, <https://www.rfc-editor.org/info/rfc5065>.

[RFC5082] Gill, V., Heasley, J., Meyer, D., Savola, P., Ed., and C. Pignataro, "The Generalized TTL Security Mechanism (GTSM)", RFC 5082, DOI 10.17487/RFC5082, October 2007, <https://www.rfc-editor.org/info/rfc5082>.

[RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, DOI 10.17487/RFC5925, June 2010, <https://www.rfc-editor.org/info/rfc5925>.

[RFC6811] Mohapatra, P., Scudder, J., Ward, D., Bush, R., and R. Austein, "BGP Prefix Origin Validation", RFC 6811, DOI 10.17487/RFC6811, January 2013, <https://www.rfc-editor.org/info/rfc6811>.

[RFC7606] Chen, E., Ed., Scudder, J., Ed., Mohapatra, P., and K. Patel, "Revised Error Handling for BGP UPDATE Messages", RFC 7606, DOI 10.17487/RFC7606, August 2015, <https://www.rfc-editor.org/info/rfc7606>.

[RFC7115] Bush, R., "Origin Validation Operation Based on the Resource Public Key Infrastructure (RPKI)", BCP 185, RFC 7115, DOI 10.17487/RFC7115, January 2014, <https://www.rfc-editor.org/info/rfc7115>.

[RFC7196] Pelsser, C., Bush, R., Patel, K., Mohapatra, P., and O. Maennel, "Making Route Flap Damping Usable", RFC 7196, DOI 10.17487/RFC7196, May 2014, <https://www.rfc-editor.org/info/rfc7196>.

[RFC7964] Walton, D., Retana, A., Chen, E., and J. Scudder, "Solutions for BGP Persistent Route Oscillation", RFC 7964, DOI 10.17487/RFC7964, September 2016, <https://www.rfc-editor.org/info/rfc7964>.

[RFC8092] Heitz, J., Ed., Snijders, J., Ed., Patel, K., Bagdonas, I., and N. Hilliard, "BGP Large Communities Attribute", RFC 8092, DOI 10.17487/RFC8092, February 2017, <https://www.rfc-editor.org/info/rfc8092>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

[RFC8201] McCann, J., Deering, S., Mogul, J., and R. Hinden, Ed., "Path MTU Discovery for IP version 6", STD 87, RFC 8201, DOI 10.17487/RFC8201, July 2017, <https://www.rfc-editor.org/info/rfc8201>.

[RFC8212] Mauch, J., Snijders, J., and G. Hankins, "Default External BGP (EBGP) Route Propagation Behavior without Policies", RFC 8212, DOI 10.17487/RFC8212, July 2017, <https://www.rfc-editor.org/info/rfc8212>.

[RFC8326] Francois, P., Ed., Decraene, B., Ed., Pelsser, C., Patel, K., and C. Filsfils, "Graceful BGP Session Shutdown", RFC 8326, DOI 10.17487/RFC8326, March 2018, <https://www.rfc-editor.org/info/rfc8326>.

[RFC8704] Sriram, K., Montgomery, D., and J. Haas, "Enhanced Feasible-Path Unicast Reverse Path Forwarding", BCP 84, RFC 8704, DOI 10.17487/RFC8704, February 2020, <https://www.rfc-editor.org/info/rfc8704>.

[RFC9234] Azimov, A., Bogomazov, E., Bush, R., Patel, K., and K. Sriram, "Route Leak Prevention and Detection Using Roles in UPDATE and OPEN Messages", RFC 9234, DOI 10.17487/RFC9234, May 2022, <https://www.rfc-editor.org/info/rfc9234>.

[RFC9319] Gilad, Y., Goldberg, S., Sriram, K., Snijders, J., and B. Maddison, "The Use of maxLength in the Resource Public Key Infrastructure (RPKI)", BCP 185, RFC 9319, DOI 10.17487/RFC9319, October 2022, <https://www.rfc-editor.org/info/rfc9319>.

[I-D.ietf-sidrops-aspa-verification] Azimov, A., Bogomazov, E., Bush, R., Patel, K., Snijders, J., and K. Sriram, "BGP AS_PATH Verification Based on Autonomous System Provider Authorization (ASPA) Objects", Work in Progress, Internet-Draft, draft-ietf-sidrops-aspa-verification-17, 23 March 2025, <https://datatracker.ietf.org/doc/draft-ietf-sidrops-aspa-verification/>.

[I-D.ietf-sidrops-avoid-rpki-state-in-bgp] Snijders, J., Fiebig, T., and M. A. Stucchi, "Guidance to Avoid Carrying RPKI Validation States in Transitive BGP Path Attributes", Work in Progress, Internet-Draft, draft-ietf-sidrops-avoid-rpki-state-in-bgp-01, 3 October 2024, <https://datatracker.ietf.org/doc/draft-ietf-sidrops-avoid-rpki-state-in-bgp/>.

11.2. Informative References

[RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191, DOI 10.17487/RFC1191, November 1990, <https://www.rfc-editor.org/info/rfc1191>.

[RFC1918] Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G. J., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, DOI 10.17487/RFC1918, February 1996, <https://www.rfc-editor.org/info/rfc1918>.

[RFC1997] Chandra, R., Traina, P., and T. Li, "BGP Communities Attribute", RFC 1997, DOI 10.17487/RFC1997, August 1996, <https://www.rfc-editor.org/info/rfc1997>.

[RFC2827] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, DOI 10.17487/RFC2827, May 2000, <https://www.rfc-editor.org/info/rfc2827>.

[RFC3345] McPherson, D., Gill, V., Walton, D., and A. Retana, "Border Gateway Protocol (BGP) Persistent Route Oscillation Condition", RFC 3345, DOI 10.17487/RFC3345, August 2002, <https://www.rfc-editor.org/info/rfc3345>.

[RFC3704] Baker, F. and P. Savola, "Ingress Filtering for Multihomed Networks", BCP 84, RFC 3704, DOI 10.17487/RFC3704, March 2004, <https://www.rfc-editor.org/info/rfc3704>.

[RFC4012] Blunk, L., Damas, J., Parent, F., and A. Robachevsky, "Routing Policy Specification Language next generation (RPSLng)", RFC 4012, DOI 10.17487/RFC4012, March 2005, <https://www.rfc-editor.org/info/rfc4012>.

[RFC4272] Murphy, S., "BGP Security Vulnerabilities Analysis", RFC 4272, DOI 10.17487/RFC4272, January 2006, <https://www.rfc-editor.org/info/rfc4272>.

[RFC4360] Sangli, S., Tappan, D., and Y. Rekhter, "BGP Extended Communities Attribute", RFC 4360, DOI 10.17487/RFC4360, February 2006, <https://www.rfc-editor.org/info/rfc4360>.

[RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, DOI 10.17487/RFC4364, February 2006, <https://www.rfc-editor.org/info/rfc4364>.

[RFC6192] Dugal, D., Pignataro, C., and R. Dunn, "Protecting the Router Control Plane", RFC 6192, DOI 10.17487/RFC6192, March 2011, <https://www.rfc-editor.org/info/rfc6192>.

[RFC6480] Lepinski, M. and S. Kent, "An Infrastructure to Support Secure Internet Routing", RFC 6480, DOI 10.17487/RFC6480, February 2012, <https://www.rfc-editor.org/info/rfc6480>.

[RFC6518] Lebovitz, G. and M. Bhatia, "Keying and Authentication for Routing Protocols (KARP) Design Guidelines", RFC 6518, DOI 10.17487/RFC6518, February 2012, <https://www.rfc-editor.org/info/rfc6518>.

[RFC6666] Hilliard, N. and D. Freedman, "A Discard Prefix for IPv6", RFC 6666, DOI 10.17487/RFC6666, August 2012, <https://www.rfc-editor.org/info/rfc6666>.

[RFC6952] Jethanandani, M., Patel, K., and L. Zheng, "Analysis of BGP, LDP, PCEP, and MSDP Issues According to the Keying and Authentication for Routing Protocols (KARP) Design Guide", RFC 6952, DOI 10.17487/RFC6952, May 2013, <https://www.rfc-editor.org/info/rfc6952>.

[RFC7132] Kent, S. and A. Chi, "Threat Model for BGP Path Security", RFC 7132, DOI 10.17487/RFC7132, February 2014, <https://www.rfc-editor.org/info/rfc7132>.

[RFC7454] Durand, J., Pepelnjak, I., and G. Doering, "BGP Operations and Security", BCP 194, RFC 7454, DOI 10.17487/RFC7454, February 2015, <https://www.rfc-editor.org/info/rfc7454>.

[RFC7947] Jasinska, E., Hilliard, N., Raszuk, R., and N. Bakker, "Internet Exchange BGP Route Server", RFC 7947, DOI 10.17487/RFC7947, September 2016, <https://www.rfc-editor.org/info/rfc7947>.

[RFC8195] Snijders, J., Heasley, J., and M. Schmidt, "Use of BGP Large Communities", RFC 8195, DOI 10.17487/RFC8195, June 2017, <https://www.rfc-editor.org/info/rfc8195>.

[RFC9229] Chroboczek, J., "IPv4 Routes with an IPv6 Next Hop in the Babel Routing Protocol", RFC 9229, DOI 10.17487/RFC9229, May 2022, <https://www.rfc-editor.org/info/rfc9229>.

[TBD] Holder, P., "Reference still to be added", November 2023, <https://example.com>.

[NSRC-17] Smith, P., "BGP Best Current Practices", February 2017, <https://nsrc.org/workshops/2017/apricot2017/bgp/bgp/preso/05-BGP-BCP.pdf>.

[KIRIN-22] Prehn, L., Foremski, P., and O. Gasser, "Kirin: Hitting the Internet with Millions of Distributed IPv6 Announcements", October 2022, <https://arxiv.org/abs/2210.10676>.

[APNICTRN-17] Roman, N. and A. Bhatia, "BGP Routing and IXP Workshop", March 2017, <https://wiki.apnictraining.net/_media/ixpworkshop-kolisoc-in/2.routing_ixp_workshop_upd.pdf>.

[RIPE-804] RIPE, "IPv4 Address Allocation and Assignment Policies for the RIPE NCC Service Region", RIPE 804, September 2023, <https://www.ripe.net/publications/docs/ripe-804>.

[CCR-22] Sediqi, Z., Prehn, L., and O. Gasser, "Hyper-specific prefixes: gotta enjoy the little things in interdomain routing", June 2022, <https://dl.acm.org/doi/abs/10.1145/3544912.3544916>.

[NLNOG-22] Li, Q., "Voorkom AS-set namespace collisions. AS-A2B vs AS51088:AS-A2B", NLNOG Mailinglist 2022, November 2022, <http://mailman.nlnog.net/pipermail/nlnog/2022-November/003046.html>.

[FENG-22] Feng, X., Li, Q., Sun, K., Xu, K., Liu, B., Zheng, X., Yang, Q., Duan, H., and Z. Qian, "PMTUD is not Panacea: Revisiting IP Fragmentation Attacks against TCP", NDSS 2022, February 2022, <https://csis.gmu.edu/ksun/publications/TCP%20Fragmentation_NDSS22.pdf>.

[RIPE-351] Daniel, D., "De-Bogonising New Address Blocks", RIPE 351, October 2005, <https://www.ripe.net/publications/docs/ripe-351>.

[RIPE-378] Smith, P. and C. Panigl, "RIPE Routing Working Group Recommendations On Route-flap Damping", RIPE 378, May 2006, <https://www.ripe.net/publications/docs/ripe-378>.

[RIPE-399] Smith, P., Evans, R., and M. Hughes, "RIPE Routing Working Group Recommendations on Route Aggregation", RIPE 399, December 2006, <https://www.ripe.net/publications/docs/ripe-399>.

[RIPE-532] Evans, R. and P. Smith, "RIPE Routing Working Group Recommendations on IPv6 Route Aggregation", RIPE 532, November 2011, <https://www.ripe.net/publications/docs/ripe-532>.

[RIPE-580] Bush, R., Pelsser, C., Kuhne, M., Maennel, O., Mohapatra, P., Patel, K., and R. Evans, "RIPE Routing Working Group Recommendations on Route Flap Damping", RIPE 580, January 2013, <https://www.ripe.net/publications/docs/ripe-580>.

[IANAv4Spec] IANA, "IANA IPv4 Special-Purpose Address Registry", <https://www.iana.org/assignments/iana-ipv4-special-registry>.

[IANAv6Spec] IANA, "IANA IPv6 Special-Purpose Address Registry", <https://www.iana.org/assignments/iana-ipv6-special-registry>.

[IANAv4Reg] IANA, "IANA IPv4 Address Space Registry", <https://www.iana.org/assignments/ipv4-address-space>.

[IANAv6Reg] IANA, "Internet Protocol Version 6 Address Space", <https://www.iana.org/assignments/ipv6-address-space>.

[RADb] Merit Network Inc., "RADb - The Internet Routing Registry", <https://www.radb.net>.

[RFC7705] George, W. and S. Amante, "Autonomous System Migration Mechanisms and Their Effects on the BGP AS_PATH Attribute", RFC 7705, DOI 10.17487/RFC7705, November 2015, <https://www.rfc-editor.org/info/rfc7705>.

[RFC7353] Bellovin, S., Bush, R., and D. Ward, "Security Requirements for BGP Path Validation", RFC 7353, DOI 10.17487/RFC7353, August 2014, <https://www.rfc-editor.org/info/rfc7353>.

[PeeringDB] PeeringDB, "PeeringDB: The Interconnection Database", <https://www.peeringdb.com/>.

[CommunityIX] IN-Berlin e.V., "Community-IX by IN-Berlin e.V., the connectivity-platform for non-commercial projects, ideas and organisations in Berlin and Frankfurt", <https://www.community-ix.net/>.

[rpki-client]
OpenBSD Project, "rpki-client 8.6 has been released", <https://marc.info/?l=openbsd-announce&m=169645206105360>.

[OpenBGPd] OpenBSD Project, "OpenBGPD 8.2 released", <https://marc.info/?l=openbsd-announce&m=169624217322602>.

[bgpq4] Snijders, J., "bgpq4 Git Repository", <https://github.com/bgp/bgpq4>.

[hotRPKI] Cooper, D., Heilman, E., Brogle, K., Reyzin, L., and S. Goldberg, "On the risk of misbehaving RPKI authorities", DOI 10.1145/2535771.2535787, November 2013, <https://dl.acm.org/doi/pdf/10.1145/2535771.2535787>.

[I-D.ietf-grow-as-path-prepending]
McBride, M., Madory, D., Tantsura, J., Raszuk, R., Li, H., Heitz, J., and G. Mishra, "AS Path Prepending", Work in Progress, Internet-Draft, draft-ietf-grow-as-path-prepending-13, 16 January 2024, <https://datatracker.ietf.org/doc/draft-ietf-grow-as-path-prepending/>.

Acknowledgements

This document is based on [RFC7454] and we thank the original authors for their work.

We thank the following people for reviewing this draft and suggesting changes:

* Gert Doerring
* Jeff Haas
* Nick Hilliard
* Geng Nan
* Martin Pels
* Job Snijders
* Berislav Todorovic
* Q Misell

Author’s Address

Tobias Fiebig
Max-Planck-Institut fuer Informatik
Campus E14
66123 Saarbruecken
Germany
Phone: +49 681 9325 3527
Email: tfiebig@mpi-inf.mpg.de