Workgroup: Network Working Group
Internet-Draft:
draft-abraitis-bgp-version-capability-12
Published: 25 January 2023
Intended Status: Informational
Expires: 29 July 2023
Authors: D. Abraitis
Hostinger

Software Version Capability for BGP

Abstract

In this document, we introduce a new BGP capability that allows the advertisement of a BGP speaker's routing daemon version.

This BGP capability is an optional advertisement. Implementations are not required to advertise the version nor to process received advertisements.

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 29 July 2023.

Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction

In modern data center designs, we tend to have conventional routers participating in the routing process. And the fleet of routers has different versions of routing daemon. This means that knowing which versions of the routing daemons are running the various routers in the network can be a crucial factor in quickly identifying the root cause of any protocol or network problems.

This BGP capability is an optional advertisement. Implementations are not required to advertise the version nor to process received advertisements.

Information about the version of the routing daemon could also be exchanged in protocols such as LLDP and CDP. However, in containerized environments, it is very hard and not recommended to exchange this information between background processes. Therefore, and to help minimize operational costs, it is helpful to exchange the routing daemon information between BGP peers directly.

2. Specification of Requirements

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.

3. Software Version Capability

Although this document is not an IETF Standards Track document, it makes use of the terminology from BCP 14 in order to clearly state the implementation behaviors.

Capabilities advertisements with BGP are defined in [RFC5492]. They utilize the BGP Capabilities Optional Parameter that contains one or more triples <Capability Code, Capability Length, Capability Value>. This document defines a new BGP capability, the Software Version Capability, with Capability Code TBD and Capability Length and Capability Value as described below.

The inclusion of the Software Version Capability is OPTIONAL. If an implementation supports the inclusion of the capability, the implementation MUST include a configuration switch to enable or disable its use, and that switch MUST be off by default.

The Software Version Capability is intended for environments where more visibility is needed for troubleshooting purposes. It is NOT RECOMMENDED for use outside a single Autonomous System, or a set of Autonomous Systems under a common administration.

An implementation that does not recognize or support the Software Version Capability but receives one must ignore it, as described in [RFC5492].

The triple for the Software Version Capability is as follows:

Capability Code

75

Capability Length

The Capability Length for the Software Version Capability MUST be greater than zero. A value of zero SHALL be treated as an encoding error and the Capability MUST be ignored.

The Capability Length SHOULD be no greater than 64. This is the limit to allow other capabilities as much space as they require.

Capability Length is a one-octet unsigned binary integer that contains the length of the Capability Value field in octets.

Capability Value

The Capability Value field is encoded in UTF-8 [RFC3629]. It is unstructured data and can be formatted in any way that the implementor decides.

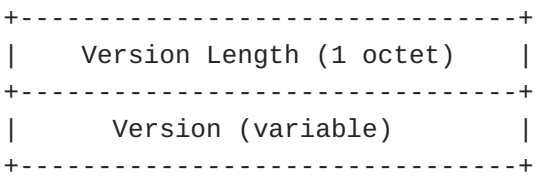

Figure 1

Version Length:

The number of characters in the Version

Version:

The Version field MUST be encoded using UTF-8. A receiving BGP speaker MUST NOT interpret invalid UTF-8 sequences.

The value of consists of one identifier, which identifies the software and its significant version. Software version identifier consists of a product and optional product version.

identifier = product ["/" product-version]

A sender SHOULD limit generated product identifiers to what is necessary to identify the product; a sender MUST NOT generate advertising or other nonessential information within the product identifier. A sender SHOULD NOT generate information in product-version that is not a version identifier (i.e., successive versions of the same product name ought to differ only in the product-version portion of the product identifier).

Example:

frrouting/8.4.2, ios/12.5.1, junos/12.1

3.1. Capabilities Length Overflow

As defined in [RFC5492] the total length of capabilities that can be carried by the BGP Capabilities Optional Parameter is 255 bytes. If an implementation is constructing a BGP Capabilities Optional Parameter and its length exceeds 255 bytes, there is not enough space for other more important capabilities. An implementation is REQUIRED Extended Optional Parameters Length for BGP OPEN Message support as defined in [RFC9072].

A rogue node can prevent the proper operation of a BGP session, or the advertisement of other Capabilities, by not excluding the Software Version Capability as required in Section 3.1. This risk is equivalent to a rogue node simply not advertising a specific Capability and is not new to BGP.

4. Operation

The Software Version Capability MUST only be used for displaying the version of a BGP speaker's router daemon to make troubleshooting easier.

Consider a group of routers each with a number of upstream nodes, and suppose that each router has a different operating system and different routing daemon at a different version installed. Assuming that a specific feature is not working or that there is a bug which has not been fixed in a particular version of the code, knowledge of the routing daemon versions would allow an operator to quickly identify the pattern of which versions are affected.

Enabling (i.e., turning on) this capability requires bouncing all existing BGP sessions and the feature MUST be explicitly configured before an implementation advertizes the Software Version Capability.

4.1. Example Usage

Below is an example from the [FRRouting] implementation showing both the received and advertised Software Version Capability:

```
:~# vtysh -c 'show ip bgp summary failed'
...
Neighbor EstdCnt DropCnt ResetTime Reason
ens192         3       3  00:00:35 Waiting for peer OPEN (n/a)
ens224         3       3  00:01:12 Waiting for NHT (frrouting/7.2)
eth0           3       3  00:00:14 Neighbor deleted (frrouting/7.3)
...
```

Figure 2

```
:~# vtysh -c 'show ip bgp neighbors 198.51.100.1 json' \
> | jq '."198.51.100.1".neighborCapabilities.versions'
{
  "advertisedVersion": "frrouting/7.2",
  "receivedVersion": "frrouting/7.3"
}
```

Figure 3

5. IANA Considerations

IANA has assigned capability number 75 for the Software Version Capability described in this document. This registration is in the BGP Capability Codes registry.

Value	Description
75	Software Version Capability

Table 1: Software Version Capability

6. Security Considerations

The Software Version Capability should be treated as sensitive information: it could be easier for an attacker to exploit the system if they know the specific software version and manufacturer of a BGP speaker. This information could be gathered by inspecting BGP OPEN messages that carry the Software Version Capability defined in this document. Furthermore, this knowledge may facilitate a number of social-engineering attacks.

Modifying the information advertised by a router might lead to attacks including bogus software upgrades and also might mask the causes of faults in the network.

Users of this mechanism should be aware that unless a transport that provides integrity is used for the BGP session in question, the Software Version Capability can be forged. Unless a transport that provides confidentiality is used, the Version Capability could be snooped by an attacker. These issues are common to any BGP message but may be of greater interest in the context of this extension as explained above. Refer to the related considerations in [RFC4271] and [RFC4272].

Users of this mechanism should consider applying data minimization practices as outlined in Section 6.1 of [RFC6973], as appropriate within the deployment context.

Sensitive information leaks can be minimized by using the [RFC5082] mechanism or firewalls to filter out TCP 179 port from untrusted networks. This capability can be disabled per neighbor, thus the sensitive information can't be disclosed to untrusted neighbors.

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/

RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, DOI 10.17487/RFC3629, November 2003, <https://www.rfc-editor.org/info/rfc3629>.

[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>.

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

[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>.

[RFC6973] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., and R. Smith, "Privacy Considerations for Internet Protocols", RFC 6973, DOI 10.17487/RFC6973, July 2013, <https://www.rfc-editor.org/info/rfc6973>.

[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>.

[RFC9072] Chen, E. and J. Scudder, "Extended Optional Parameters Length for BGP OPEN Message", RFC 9072, DOI 10.17487/RFC9072, July 2021, <https://www.rfc-editor.org/info/rfc9072>.

7.2. Informative References

[FRRouting] Abraitis, D.A., "FRRouting - BGP Software Version Capability", 2019, <https://github.com/opensourcerouting/frr/commit/4e91cdb63a9a2008e9ebe11cf0bc1ea5b8020e75>.

Author's Address

Donatas Abraitis
Hostinger

Email: donatas.abraitis@hostinger.com