IDR Working Group   R. Raszuk, Ed.
Internet-Draft   Mirantis Inc.
Intended status: Standards Track   J. Haas, Ed.
Expires: September 7, 2015   Juniper Networks
                                                      A. Lange, Ed.
                                                      Alcatel-Lucent
                                                          S. Amante
                                                          Apple, Inc.
                                                         B. Decraene
                                                              Orange
                                                           P. Jakma
                                                     Uni. of Glasgow
                                                     R. Steenbergen
                                           nLayer Communications, Inc.
                                                       March 6, 2015

                     Wide BGP Communities Attribute
                   draft-raszuk-wide-bgp-communities-05

Abstract

   Route tagging plays an important role in external BGP [RFC4271]
   relations, in communicating various routing policies between peers.
   It is also a very common best practice among operators to propagate
   various additional information about routes intra-domain. The most
   common tool used today to attach various information about routes is
   through the use of BGP communities [RFC1997].

   Such information is important to allow BGP speakers to perform some
   mutually agreed actions without the need to maintain a separate
   offline database for each tuple of prefix and associated set of
   action entries.

   This document defines a new encoding which will enhance and simplify
   what can be accomplished today with the use of BGP communities. The
   most important addition this specification makes over currently
   defined BGP communities is the ability to specify, carry as well as
   use for execution an operator's defined set of parameters. It also
   provides an extensible platform for any new community encoding needs
   in the future.

Requirements Language

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

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

   RFC 1997 [RFC1997] defines the BGP Community Attribute. This
   attribute is used as a tool to carry additional information in BGP
   routes which may help to automate peering administration. The BGP
   Communities Attribute consists of one or more sets of four octet
   values, where each specifies a different community. Except for two
   reserved ranges, the encoding of community values mandates that the
   first two octets are to contain the Autonomous System number, with
   the next two octets containing some locally defined value.

   With the introduction of 4-octet Autonomous System numbers by RFC
   4893 [RFC4893] it became obvious that BGP Communities as specified in
   RFC 1997 will not be able to accommodate new AS encoding. In fact
   RFC 4893 explicitly recommends use of four octets AS specific
   [RFC5668] extended communities [RFC4360] as a way to encode new 4
   octet AS numbers.

   While the encoding of 4 octet AS numbers is being addressed by
   [I-D.ietf-idr-as4octet-extcomm-generic-subtype], neither the base BGP
   communities (standard or extended) nor as4octet-extcomm-generic
   document define a sufficient level of encoding freedom which could be
   of practical use. The authors believe that defining a new BGP Path

Attribute, with the ability to contain locally defined parameters
will enhance the current level of network policies, as well as
simplify BGP policy management. The proposed simple encoding will
also facilitate the delivery of new network services without a need
to define a new BGP extension each time.

When defining any new type of tool there is always a unique
opportunity to specify a subset of well recognized behaviors. Lists
of the current most commonly used BGP communities, as well as
provision for a new registry for future definitions will be contained
in a separate document.

1.1. Protocol Summary

Each Wide BGP Community consists of three main parts:

1. A Container Header. This header is defined in Section 4.
    The Container Header encodes:

    * Container Type. One Type is defined in this document.
    * Flags to control behavior.
    * Hop Count to control the degree of Community propagation
    * Length

2. The Community itself, defined as a type of Container. The Type 1
    Wide Community is defined in Section 4.
    The Type 1 Wide Community contains:

    * Community Value: This section defines the action that an
       operator wishes a router to take.
    * Source AS: This is the AS originating the community.
    * Context AS: This its the AS context from which community
       should be interpreted.
    * Target(s): This is an optional list that encodes where the
       community's action should be taken.
    * Exclude Target(s): This is an optional list that encodes where
       the community's actions should not be taken.
    * Parameters: This is an optional list that encodes additional
       information that the community's action needs to execute
       properly.

3. Community Atoms. These are values and lists of values that are
    common across community actions. They are defined in Section 2.

2. **Wide Community Atoms**

   Wide BGP communities will act on and hence need to encode some
   distinct atoms of data. These are encoded as TLVs, where each TLV
   has the following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+
|     Type      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Value (variable)                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
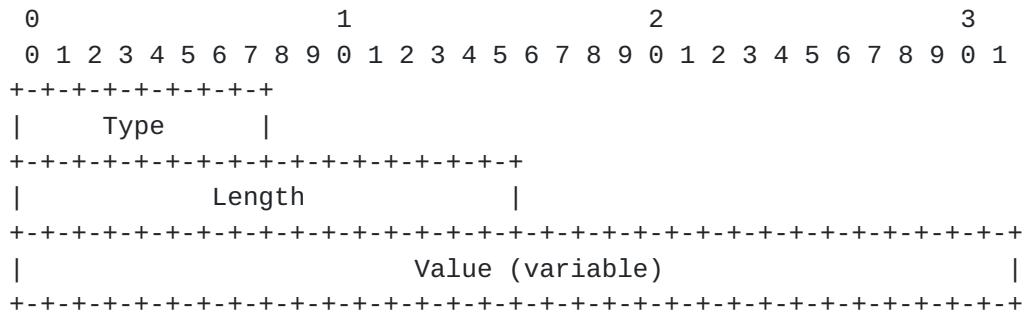
   The Type field contains a value of 1-254. The values 0 and 255 are
   reserved for future use. The TLV types are to be assigned and
   maintained by IANA registry.

   The Length represents the total length of a given TLV's value field
   in octets.

   The Value field contains the TLV value.

   Supported format of the TLVs can be:

```
  Type  1: Autonomous System number list
  Type  2: IPv4 prefix (1 octet prefix length + prefix) list
  Type  3: IPv6 prefix (1 octet prefix length + prefix) list
  Type  4: Integer list
  Type  5: IEEE Floating Point Number list
  Type  6: Neighbor Class list
  Type  7: User-defined Class list
  Type  8: UTF-8 String
```

   The semantics of a given atom will depend upon the context in which
   it is used, as defined by the containing wide community.

   In the following sections defining the different atoms, validation
   rules for the Length of the atom will be presented. If the Length of
   the atom does not match the rules for that atom, it SHALL be
   considered malformed. (See Section 8.)

2.1. The Autonomous System number list atom

This atom represents a list of Autonomous System numbers, each 4
octets in size. The minimum Length of this atom is 4 octets. The
Length MUST be a multiple of 4.

For consistent treatment, all AS numbers MUST be encoded as 4 octet
values. When encoding two octet ASes, the first two octets of this
four octet value MUST be filled with zeros.

Two special values are reserved for the Autonomous System atoms:

    0x00000000 - to indicate "No Autonomous Systems".
    0xFFFFFFFF - to indicate "All Autonomous Systems".

2.2. The IPv4 and IPv6 prefix list atoms

This atom represents a list of IPv4 or IPv6 prefix. IPv4 and IPv6
prefix atom values are encoded in the same format used by BGP NLRI in
Section 4.3 of [RFC4271].

```
+---------------------------+
|  Prefix Length (1 octet)  |
+---------------------------+
|  Prefix (variable)        |
+---------------------------+
```

The Prefix Length for IPv4 prefixes must be in the range of 0..32.

The Prefix Length for IPv6 prefixes must be in the range of 0..128.

The Length field must be able to accommodate the list of prefixes
according to the encoding rules. If the Length cannot fully
accommodate the required number of octets to encode the Prefix Length
and the Prefix, the atom is considered malformed.

2.3. The Integer list atom

This atom represents a list of Integers. Integers are a fixed Length
of 4 octets and are stored in network byte order.

The minimum Length of the Integer list atom is 4 octets. The Length
MUST be a multiple of 4.

2.4. The IEEE Floating Point Number list atom

   This atom represents a list of floating point numbers. Floating
   point numbers are a fixed Length of 4 octets and are stored in
   [IEEE.754.1985] format.

   This atom represents a list of floating point numbers.

   The minimum Length of the Floating Point Number list atom is 4
   octets. The Length MUST be a multiple of 4.

2.5. The Neighbor Class list atom

   The Neighbor Class list atom represents a classification of a BGP
   peering session, each 4 octets in size. This class currently can
   contain three values:

     1 - Peer: This class is typically applied to sessions where a
                    transit-free relationship exists between two
                    providers.
     2 - Customer: This class is typically applied to sessions where
                    the remote end of the session is operated by a
                    customer.
     3 - Upstream: This class is typically applied to sessions where the
                    remote end of the session is operated by a network
                    from which you receive transit routes.

   The Neighbor Class list atom represents a classification of a BGP
   peering session.

   The minimum Length of the Neighbor Class list atom is 4 octets. The
   Length MUST be a multiple of 4.

2.6. The User-defined Class list atom

   Similar to the Neighbor Class atom, the User-defined Class list atom
   represents a classification of a network property. The exact
   property definition is up to the semantics of the defining Autonomous
   System. The semantics governing a given User-defined Class list are
   defined by the Context AS Number.

   Examples of User-defined Class properties include geography (East,
   West), continent (North America, Asia, Europe), etc. Similar to the
   [RFC1997] BGP Communities, it is necessary that the Context AS
   provide a registry of the value and the semantics of a given
   community.

The minimum Length of the User-defined Class list atom is 4 octets.
The Length of this atom MUST be a multiple of 4.

2.7. The UTF-8 String atom

The UTF-8 String atom represents an arbitrary Unicode string in UTF-8
[RFC3629] format. The Length is required to be of sufficient size to
carry the UTF-8 string in the Value field.

Implementations MUST be prepared for truncated/improperly formed
UTF-8 strings. When detecting such a string, the implementation
should remove trailing octets of a multi-octet sequence in order to
have a well-formed string.

Implementations MUST be prepared to receive empty (zero-Length) UTF-8
String atoms as they may be used as Parameters.

3. Wide BGP Community Attribute

This document defines a new BGP Path Attribute, the Wide BGP
Community. The attribute type code is (TBA by IANA).

The Wide BGP Community Attribute is an optional, transitive BGP
attribute, and may be present only once in the BGP UPDATE message.

The attribute contains a number of typed containers. Any given
container type may appear multiple times, unless that container
type's definition specifies otherwise.

3.1. Wide BGP Community Attribute Container Header

Containers always start with the following header:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Type             |     Flags     |   Hop Count   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Length             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

This document defines one Type code - Type 1. See the Section 11 for
information on additional type registration policies.

```
+------+-------+----------------------------------------------------+
| Bit  | Value | Meaning                                            |
+------+-------+----------------------------------------------------+
|  0   |   0   | Local community value.                             |
|      |   1   | Registered community value.                        |
|  1   |   0   | Do not decrement Hop Count field across            |
|      |       | confederation boundaries.                          |
|      |   1   | Decrement Hop Count field across confederation     |
|      |       | boundaries.                                        |
| 2..7 |   -   | MUST be zero when sent and ignored upon receipt.   |
+------+-------+----------------------------------------------------+
```

Flags are defined globally, to apply to all wide community container
types.

Table 1: Flags

Bit 0 (aka R bit) set (value 1) indicates that the given container
carries a Wide BGP Community which is registered with IANA. When not
set (value 0) it indicates that community value which follows is
locally assigned with a local meaning.

Bit 1 (aka C bit) is used to manage the propagation scope of a given
Wide BGP Community across confederation boundaries. When not set
(value of 0), the Hop Count field is not considered at the sub-AS
boundaries. When set (value of 1), sub-AS border router follows the
same procedure regarding the handling of the Hop Count field as
applicable to ASBR at the domain boundary.

The Hop Count field represents the forwarding radius, in units of AS
hops, for the given Wide BGP Community. A Hop Count value of zero
indicates that this wide community must not cross any further AS
boundaries. At each AS boundary, when propagating a given wide
community over an EBGP session, the Hop Count field MUST be
decremented by the sending EBGP speaker.

The exact same decrement procedures described above apply also to
sub-confederation boundaries when the bit 1's flag is set to 1.

The special value of 0xFF indicates that the enclosed community may
always be propagated over an EBGP boundary. A Hop Count value of
0xFF MUST NOT be decremented during propagation.

The Length field represents the total length of a given container in
octets.

4. Container Type 1: Wide Community

The Wide BGP Community Type 1 container is of variable size (but
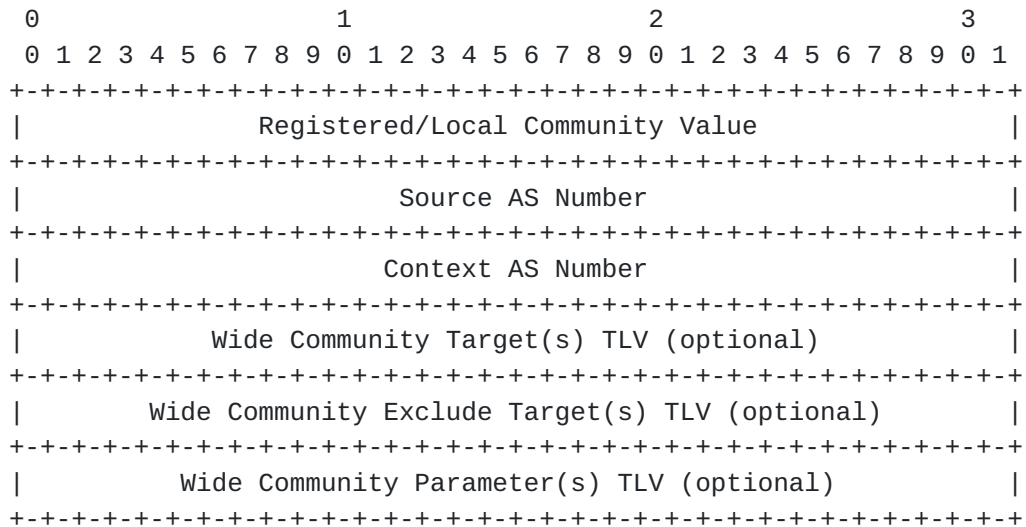minimum length 12) and is encoded as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Registered/Local Community Value                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Source AS Number                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Context AS Number                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Wide Community Target(s) TLV (optional)            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Wide Community Exclude Target(s) TLV (optional)       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Wide Community Parameter(s) TLV (optional)          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: Wide BGP Community Type 1

4.1. Community Value

Community Value: 4 octets

The Wide BGP Community value indicates what set of actions a
router is requested to take upon reception of a route containing
this community. The semantics of this value depend on whether
this is a private/local community or registered.

4.2. Source AS Number

Source Autonomous System Number: 4 octets

The Autonomous System number which indicates the originator of
this Wide BGP Community.

When the Autonomous System is a two octet number the first two
octets of this 4 octet value MUST be filled with zeros.

4.3. Context AS Number

Context Autonomous System Number: 4 octets

The Autonomous System number that indicates the context of the
Registered/Local Value. For Wide Community Containers that are
Locally defined, the Context Autonomous System number provides the
context for the Community Value and thus the meaning of the community
as well as any User-defined Classes for that Context AS.

For Registered Community Containers, the Context Autonomous System
number MAY be utilized to provide specific meaning for that
Registered community. When no such context is implied, this field
MUST be 0.

When the wide community is locally registered, the Context Autonomous
System Number indicates the AS that defines the format of this wide
community for the given Local Value. (In other words, value 1 will
likely refer to different formats for AS 1 vs. AS 2.)

4.4. Wide Community Target(s) TLV

The Wide Community Target(s) TLV (Type 1) contains a list of a Wide
Community atoms.

Wide Community Targets define the matching criteria for the
community. A given wide community may have a number of targets that
it applies to. The semantics of these targets will vary on a per
community basis. Depending on the definition of the community,
targets may be optional.

The value field of the Wide Community Target(s) TLV is a series of
Wide Community Atom TLVs. The semantics of any given atom TLV MUST
be part of the definition of a given Wide Community.

Typically, Wide Community Targets consist of a series of atoms that
have "match any" semantics. Thus, if any given target matches per
the semantics of that atom for the community, the community is
considered to match and the action defined by the community should be
executed.

When no Target(s) TLV is specified, it is considered "match all".

If the semantics of a given atom is undefined for the community in
question, it MUST be ignored.

When no targets are required by the definition of a given Wide
Community, the Wide Community Target(s) TLV SHOULD NOT be encoded in
the community. Implementations MUST be prepared to accept a Wide
Community Target(s) TLV with an empty value field.

4.5. Wide Community Exclude Target(s) TLV

   The Wide Community Exclude Target(s) TLV (Type 2) contains a list of
   a Wide Community atoms.

   Wide Community Exclude Targets define criteria by which the community
   is considered to NOT match. Depending on the semantics of the Wide
   Community, Exclude Target(s) may be optional.

   The semantic of the Wide Community Exclude Target(s) is to match all
   specified Target(s) with the exception of those listed in this TLV.

   The value field of the Wide Community Exclude Target(s) TLV is a
   series of Wide Community Atom TLVs. The semantics of any given atom
   TLV MUST be part of the definition of a given Wide Community.

   If the semantics of a given atom is undefined for the community in
   question, it MUST be ignored.

   If the Wide Community Target(s) TLV and the Wide Community Exclude
   Target(s) TLV have conflicting semantics, priority MUST be given to
   the Wide Community Exclude Target(s) TLV.

   When no exclude targets are required by the definition of a given
   Wide Community, the Wide Community Exclude Target(s) TLV SHOULD NOT
   be encoded in the community. Implementations MUST be prepared to
   accept a Wide Community Exclude Target(s) TLV with an empty value
   field.

4.6. Wide Community Parameter(s) TLV

   The Wide Community Parameter(s) TLV (Type 3) contains a list of a
   Wide Community atoms.

   A given wide community may have parameters which are used as inputs
   for executing actions defined for that community. These parameters,
   and any constraints implied by the parameters, MUST be defined by the
   wide community definition. Parameters consist of an ordered set of
   atom sub-TLVs. The semantics of any specific positional instance of
   an atom MUST be defined by the wide community.

   If it is the case that a parameter for a given community is of an
   unexpected type or length, the community MUST be ignored.

   If it is the case that there are too many or two few parameters for a
   given community, the community MUST be ignored.

When no parameters are required by the definition of a given Wide
Community, the Wide Community Paramters TLV SHOULD NOT be encoded in
the community. Implementations MUST be prepared to accept a Wide
Community Parameter TLV with an empty value field.

4.7. Usage

The detailed interpretation of the targets or parameters SHALL be
provided when describing given community type in a separate document
or when locally defined by an operator.

5. The AS-4 List Generic Wide BGP Community

In standard [RFC1997] BGP Communities, a commonly deployed format is
AS:0x0000-0xFFFF. The left-hand-side AS is the context AS while the
right-hand-side represents an action intended to be taken upon that
AS. While the [I-D.ietf-idr-as4octet-extcomm-generic-subtype]
Extended Community format is intended to address the use cases where
the right-hand-side represents a semi-opaque integer value, operators
that desire to utilize the right-hand-side as an Autonomous System
Number have been unable to do so.

In the AS:0x0000-0xFFFF format, there are no explicit semantics on
the action for a given right-hand-side. The semantics are implicit
to the Autonomous System number itself. While such a format doesn't
take advantage of the more flexible semantics of Wide BGP
Communities, the format is capable of addressing this operational
need.

This document defines a Registered Wide BGP Community to address that
need.

5.1. Definition

```
Community Value: 1 - AS-4 List Generic Wide BGP Community
Source AS Number: <Your AS number>
Context AS Number: <Targeted AS number>
Wide Community Target(s): <MUST BE ABSENT>
Wide Community Exclude Target(s): <MUST BE ABSENT>
Wide Community Parameter(s): <Wide Community Atom Type 4>
```

6. Well Known Standard BGP Communities

According to RFC 1997, as well as IANA's Well-Known BGP Communities
registry, the following BGP communities are defined to have global
significance:

          0xFFFF0000   planned-shut   [draft-francois-bgp-gshut]
          0xFFFFFF01   NO_EXPORT   [RFC1997]
          0xFFFFFF02   NO_ADVERTISE   [RFC1997]
          0xFFFFFF03   NO_EXPORT_SUBCONFED   [RFC1997]
          0xFFFFFF04   NOPEER   [RFC3765]

   This document recommends for simplicity as well as for avoidance of
   backward compatibility issues the continued use of BGP Standard
   Community Attribute type 8 as defined in RFC 1997 to distribute non
   Autonomous System specific Well-Known BGP Communities.

   For the same reason, this document does not intended to obsolete the
   currently defined and deployed BGP Extended Communities.

7. Operational Considerations

   Having two different ways to propagate locally assigned BGP
   communities, one via the use of Standard BGP Communities and the
   other one via the use of Wide BGP Communities, may seem to
   potentially cause problems when considering propagation of
   conflicting actions. However, even at present, an operator may
   append Standard BGP Communities with conflicting information. It is
   therefore recommended that any implementation, in supporting both
   standard and Wide BGP communities, allow for their easy inbound and
   outbound processing. The actual execution of all communities should
   be treated as a union and, if supported by an implementation, their
   execution permissions are to be a local configuration matter.

8. Error Handling

   If any atom in a Wide Community container's Exclude Targets TLV is
   unrecognized, no actions should be taken for that Wide Community.
   While the Targets TLV is meant to be inclusive, the Exclude Targets
   TLV is meant to be proscriptive of applying the action.

   If any Wide Community container or atoms contained therein are
   determined to be malformed, the Wide Community Path attribute must be
   considered malformed. BGP implementations should use "treat-as-
   withdraw" semantics as defined in [I-D.ietf-idr-error-handling].

9. Example

9.1. Example Wide Community Definition

   An operator of an AS 64496, wishes to locally define a Wide Community
   with the semantics of permitting AS_PATH prepending with targets that
   include AS numbers of peer ASes and peers who have been marked with a
   set of enumerated city locations.

AS 64496 has established a registered set of values to use for its
User-defined Class:

     100 - Amsterdam
     101 - New York
     102 - San Francisco
     103 - Tokyo
     104 - Moscow

   Target semantics:

     The Autonomous System Number list atom refers to the target peer
     AS Numbers.

     The User-defined Class for AS 64496 has been defined elsewhere and
     the values 100..104 may be used for this locally defined Wide
     Community.

     The Targets TLV MUST contain at least one entry.

     The Exclude Targets TLV MAY contain entries of the above supported
     atoms.

     The semantics of all other atoms are undefined for this community.

   Parameter semantics:

     The parameter TLV shall consist of exactly one Integer atom value
     that is constrained to have a value of 2..8.

9.2. **Example Wide Community Encoding**

   AS_PATH prepend 4 TIMES TO AS 2424, AS 8888, to peers marked as
   Amsterdam (100) or to peers marked Moscow (104), but not to peers
   in New York (101).

   Use Hop Count 0 to request the receiving router to not propagate
   this wide community.

   Locally community value (flag bit 0 = 0).
   Do not decrement Hop Count field across confederation boundaries
   (0)

   Local community 1 for sample AS 64496.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Container Type 1 (1)      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|0 0 0 0 0 0 0 0|
+-+-+-+-+-+-+-+-+
| Hop Count: 0  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Length:               57 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Community: LOCAL PREPEND ACTION CATEGORY                    1 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Own ASN                                                64496 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Context ASN#                                           64496 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Target TLV (1)|   Length:                    22 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| ASN List (2)  |   Length:                     8 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Target ASN#                                             2424 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Target ASN#                                             8888 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  User List(7) |   Length:                     8 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Amsterdam                                               100 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Moscow                                                  104 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|ExcTargetTLV(2)|   Length:                     7 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  User List(7) |   Length:                     4 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| New York                                               101 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Param TLV (3) |   Length:                     7 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Integer  (4) |   Length:                     4 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Prepend #                                                4 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

10. Security considerations

   All the security considerations for BGP Communities as well as for
   BGP RFCs apply here.

   Given the flexibility and power offered by Wide BGP communities, it
   is important to consider the additional possibilities allowed by
   their definition. In particular, for locally defined Wide BGP
   Communities, it may be wise to restrict the range of parameters. For
   registred Wide BGP Communities, the security considerations of the
   document defining them MUST address issues specific to those newly
   defined Wide Communities.

   Security considerations specific to Wide BGP Communities will be
   discussed in a later revision of this draft.

11. IANA Considerations

   This document defines a new BGP Path Attribute called Wide BGP
   Community Attribute. For this new type IANA is to allocate a new
   value in the corresponding registry:

   Registry Name: BGP Path Attributes

   This document makes the following assignments for the optional,
   transitive Wide BGP Communities Attribute:

   Name   Type Value
   ----   ----------
   Wide BGP Community Attribute   TBA

   This document requests IANA to define and maintain a new registry
   named: "Wide BGP Communities Attribute Container Types".

   The pool of: 0x0000-0xFFFF has been defined for its allocations. The
   allocation policy is on a first come, first served basis.

   This document makes the following assignments for the Wide BGP
   Communities Container Types values:

   Name   Type Value
   ----   ----------
   Reserved   0x0000
   Type 1, Wide BGP Community   0x0001
   Types 2-1023 to be allocated using IETF Consensus
   Types 1024-64511 to be allocated first come, first served
   Types 64512-65534 are reserved for experimental use
   Reserved   0xFFFF

This document requests IANA to define and maintain a new registry
named: "Wide BGP Communities Atom Types". The pool of 0x0000-0xFFFF
has been defined for its allocations.

This document makes the following assignments for the Wide BGP
Communities Atom Type values:

```
   Name                              Type Value
   ----                              ----------
   Reserved                          0x00
   Autonomous System Number List     0x01
   IPv4 Prefix list                  0x02
   IPv6 Prefix list                  0x03
   Integer list                      0x04
   IEEE Floating Point Number list   0x05
   Neighbor Class list               0x06
   User-defined Class list           0x07
   UTF-8 string                      0x08
   Reserved                          0xFF
```

This document requests IANA to define and maintain a new registry
named: "Registered Wide BGP Communities". The pool of
0x00000000-0FFFFFFF has ben defined for its allocation.

This document makes the following assignments for the Registered Wide
BGP Communities:

```
   Name                              Type Value
   ----                              ----------
   Reserved                           0x00
   AS-4 List Generic Wide BGP Community  0x01
   Reserved                           0xFFFFFFFF
```

Values 2-1023 are to be allocated using IETF Consensus. Values
64512-65534 are reserved for experimental use. All other values are
available on a first-come, first served basis.

12. Change History

Changes from -03 via -04 to -05:

Update the Introduction.

Substantial re-work of atom types removing proposed Group
container and moving atoms to be lists.

Added the Exclude Targets TLV to the Wide Community container.

Added a section on error handling.

Updated the example.

Changes from -02 to -03:

Removed C and R named bit fields originally from -00.

Rename Target AS field to Context AS.

Make Integer Atom a fixed 4 octets in length.

Add Neighbor Class Atom

Rename TTL to Hop Count

Changes from -01 to -02:

The Type field has been expanded to 2 octets.

The Length field has been moved to the common header.

Changed format to use TLVs.

Added atom TLV to define well defined syntactic items.

Added TLVs to distinguish targets from parameters.

Various editorial changes to language.

[13]. Outstanding Issues

The following are known issues that have yet to be resolved in this draft:

o The interaction of the Container TTL field with VPN peers.

o The name Wide Communities is overloaded in this document. The scope of this feature has evolved since the initial -00 of the draft. The general feature of a containerized BGP Community extension and the Type 1 container, the Wide community, currently share names. "There are only two hard things in Computer Science: cache invalidation and naming things."

14. Contributors

   The following people contributed significantly to the content of the
   document:

   Shintaro Kojima
   OTEMACHI 1st. SQUARE EAST TOWER, 3F
   1-5-1, Otemachi,
   Chiyoda-ku, Tokyo 100-0004
   Japan
   Email: koji@mfeed.ad.jp

   Juan Alcaide
   Cisco Systems
   Research Triangle Park, NC
   United States
   Email: jalcaide@cisco.com

   Burjiz Pithawala
   Cisco Systems
   170 West Tasman Dr
   San Jose, CA
   United States
   Email: bpithaw@cisco.com

   Saku Ytti
   TDC Oy
   Mechelininkatu 1a
   00094 TDC
   Finland
   Email: ytti@tdc.net

15. Acknowledgments

   This document owes draft-lange-flexible-bgp-communities a debt for
   the inspiration of many features contained herein.

   The authors would like to thank Enke Chen, Pedro Marques and Alton Lo
   for their valuable input.

16. References

16.1. Normative References

[I-D.ietf-idr-error-handling]
          Scudder, J., Chen, E., Mohapatra, P., and K. Patel,
          "Revised Error Handling for BGP UPDATE Messages", draft-
          ietf-idr-error-handling-05 (work in progress), February
          2014.

[IEEE.754.1985]
          Institute of Electrical and Electronics Engineers,
          "Standard for Binary Floating-Point Arithmetic", IEEE
          Standard 754, August 1985.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
          Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO
          10646", STD 63, RFC 3629, November 2003.

[RFC4271] Rekhter, Y., Li, T., and S. Hares, "A Border Gateway
          Protocol 4 (BGP-4)", RFC 4271, January 2006.

16.2. Informative References

[I-D.ietf-idr-as4octet-extcomm-generic-subtype]
          Rao, D., Mohapatra, P., and J. Haas, "Generic Subtype for
          BGP Four-octet AS specific extended community", draft-
          ietf-idr-as4octet-extcomm-generic-subtype-06 (work in
          progress), October 2012.

[RFC1997] Chandrasekeran, R., Traina, P., and T. Li, "BGP
          Communities Attribute", RFC 1997, August 1996.

[RFC4360] Sangli, S., Tappan, D., and Y. Rekhter, "BGP Extended
          Communities Attribute", RFC 4360, February 2006.

[RFC4893] Vohra, Q. and E. Chen, "BGP Support for Four-octet AS
          Number Space", RFC 4893, May 2007.

[RFC5668] Rekhter, Y., Sangli, S., and D. Tappan, "4-Octet AS
          Specific BGP Extended Community", RFC 5668, October 2009.

Authors' Addresses

Robert Raszuk (editor)
Mirantis Inc.
615 National Ave. #100
Mt View, CA   94043
USA

Email: robert@raszuk.net

Jeffrey Haas (editor)
Juniper Networks
1194 N.Mathilda Ave
Sunnyvale, CA   94089
US

Email: jhaas@juniper.net

Andrew Lange (editor)
Alcatel-Lucent
777 E. Middlefield Road
Mountain View, CA   94043
US

Email: andrew.lange@alcatel-lucent.com

Shane Amante
Apple, Inc.
1 Infinite Loop
Cupertino, CA   95014
US

Email: amante@apple.com

Bruno Decraene
Orange
38-40 rue du General Leclerc
Issy Moulineaux cedex 9   92794
France

Email: bruno.decraene@orange.com

Paul Jakma
University of Glasgow
School of Computing Science
Sir Alwyn Williams Building
University of Glasgow
Glasgow G12 8QQ
UK

Email: paulj@dcs.gla.ac.uk

Richard A Steenbergen
nLayer Communications, Inc.
209 W Jackson Blvd
Chicago, IL 60606
US

Email: ras@nlayer.net