Network Working Group
Request for Comments: 1105

K. Lougheed
cisco Systems
Y. Rekhter
T.J. Watson Research Center, IBM Corp.
June 1989

A Border Gateway Protocol (BGP)

Status of this Memo

This RFC outlines a specific approach for the exchange of network reachability information between Autonomous Systems.

At the time of this writing, the Border Gateway Protocol implementations exist for cisco routers as well as for the NSFNET Nodal Switching Systems. A public domain version for "gated" is currently being implemented.

Distribution of this memo is unlimited.

1. Introduction

The Border Gateway Protocol (BGP) is an inter-autonomous system routing protocol. It is built on experience gained with EGP as defined in RFC 904 [1] and EGP usage in the NSFNET Backbone as described in RFC 1092 [2] and RFC 1093 [3].

The primary function of a BGP speaking system is to exchange network reachability information with other BGP systems. This network reachability information includes information on the autonomous systems (AS's) that traffic must transit to reach these networks. This information is sufficient to construct a graph of AS connectivity from which routing loops may be pruned and policy decisions at an AS level may be enforced.

BGP runs over a reliable transport level protocol. This eliminates the need to implement explicit update fragmentation, retransmission, acknowledgement, and sequencing. Any authentication scheme used by the transport protocol may be used in addition to BGP's own authentication mechanisms.

The initial BGP implementation is based on TCP [4], however any reliable transport may be used. A message passing protocol such as VMTP [5] might be more natural for BGP. TCP will be used, however, since it is present in virtually all commercial routers and hosts. In the following descriptions the phrase "transport protocol connection" can be understood to refer to a TCP connection. BGP uses TCP port 179 for establishing its connections.

2. Summary of Operation

Two hosts form a transport protocol connection between one another. They exchange messages to open and confirm the connection parameters. The initial data flow is the entire BGP routing table. Incremental updates are sent as the routing tables change. Keepalive messages are sent periodically to ensure the liveness of the connection. Notification messages are sent in response to errors or special conditions. If a connection encounters an error condition, a notification message is sent and the connection is optionally closed.

The hosts executing the Border Gateway Protocol need not be routers. A non-routing host could exchange routing information with routers via EGP or even an interior routing protocol. That non-routing host could then use BGP to exchange routing information with a border gateway in another autonomous system. The implications and applications of this architecture are for further study.

If a particular AS has more than one BGP gateway, then all these gateways should have a consistent view of routing. A consistent view of the interior routes of the autonomous system is provided by the intra-AS routing protocol. A consistent view of the routes exterior to the AS may be provided in a variety of ways. One way is to use the BGP protocol to exchange routing information between the BGP gateways within a single AS. In this case, in order to maintain consist routing information, these gateways MUST have direct BGP sessions with each other (the BGP sessions should form a complete graph). Note that this requirement does not imply that all BGP gateways within a single AS must have direct links to each other; other methods may be used to ensure consistent routing information.

3. Message Formats

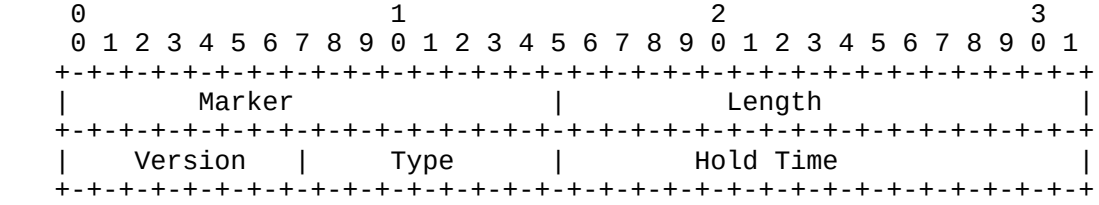
This section describes message formats and actions to be taken when errors are detected while processing these messages.

Messages are sent over a reliable transport protocol connection. A message is processed after it is entirely received. The maximum message size is 1024 bytes. All implementations are required to support this maximum message size. The smallest message that may be sent consists of a BGP header without a data portion, or 8 bytes.

The phrase "the BGP connection is closed" means that the transport protocol connection has been closed and that all resources for that BGP connection have been deallocated. Routing table entries associated with the remote peer are marked as invalid. This information is passed to other BGP peers before being deleted from the system.

3.1 Message Header Format

Each message has a fixed size header. There may or may not be a data portion following the header, depending on the message type. The layout of these fields is shown below.



Marker: 16 bits

The Marker field is 16 bits of all ones. This field is used to mark the start of a message. If the first two bytes of a message are not all ones then we have a synchronization error and the BGP connection should be closed after sending a notification message with opcode 5 (connection not synchronized). No notification data is sent.

Length: 16 bits

The Length field is 16 bits. It is the total length of the message, incluluding header, in bytes. If an illegal length is encountered (more than 1024 bytes or less than 8 bytes), a notification message with opcode 6 (bad message length) and two data bytes of the bad length should be sent and the BGP connection closed.

Version: 8 bits

The Version field is 8 bits of protocol version number. The current BGP version number is 1. If a bad version number is found, a notification message with opcode 8 (bad version number) should be sent and the BGP connection closed. The bad version number should be included in one byte of notification data.

Type: 8 bits

The Type field is 8 bits of message type code. The following type codes are defined:

1 - OPEN
2 - UPDATE
3 - NOTIFICATION
4 - KEEPALIVE
5 - OPEN CONFIRM

If an unrecognized type value is found, a notification message with opcode 7 (bad type code) and data consisting of the byte of type field in question should be sent and the BGP connection closed.

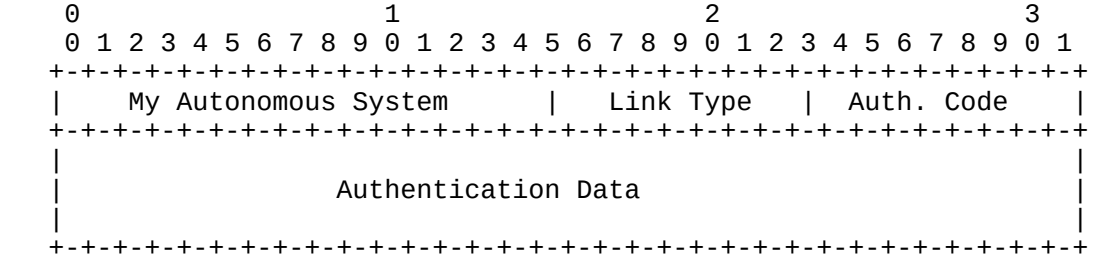
Hold Timer: 16 bits.

This field contains the number of seconds that may elapse since receiving a BGP KEEPALIVE or BGP UPDATE message from our BGP peer before we declare an error and close the BGP connection.

3.2 OPEN Message Format

After a transport protocol connection is established, the first message sent by either side is an OPEN message. If the OPEN message is acceptable, an OPEN CONFIRM message confirming the OPEN is sent back. Once the OPEN is confirmed, UPDATE, KEEPALIVE, and NOTIFICATION messages may be exchanged.

In addition to the fixed size BGP header, the OPEN message contains the following fields.



My Autonomous System: 16 bits

This field is our 16 bit autonomous system number. If there is a problem with this field, a notification message with opcode 9 (invalid AS field) should be sent and the BGP connection closed. No notification data is sent.

Link Type: 8 bits

The Link Type field is a single octet containing one of the

following codes defining our position in the AS graph relative to our peer.

0 - INTERNAL
1 - UP
2 - DOWN
3 - H-LINK

UP indicates the peer is higher in the AS hierarchy, DOWN indicates lower, and H-LINK indicates at the same level. INTERNAL indicates that the peer is another BGP speaking host in our autonomous system. INTERNAL links are used to keep AS routing information consistent with an AS with multiple border gateways. If the Link Type field is unacceptable, a notification message with opcode 1 (link type error in open) and data consisting of the expected link type should be sent and the BGP connection closed. The acceptable values for the Link Type fields of two BGP peers are discussed below.

Authentication Code: 8 bits

The Authentication Code field is an octet whose value describes the authentication mechanism being used. A value of zero indicates no BGP authentication. Note that a separate authentication mechanism may be used in establishing the transport level connection. If the authentication code is not recognized, a notification message with opcode 2 (unknown authentication code) and no data is sent and the BGP connection is closed.

Authentication Data: variable length

The Authentication Data field is a variable length field containing authentication data. If the value of Authentication Code field is zero, the Authentication Data field has zero length. If authentication fails, a notification message with opcode 3 (authentication failure) and no data is sent and the BGP connection is closed.

3.3 OPEN CONFIRM Message Format

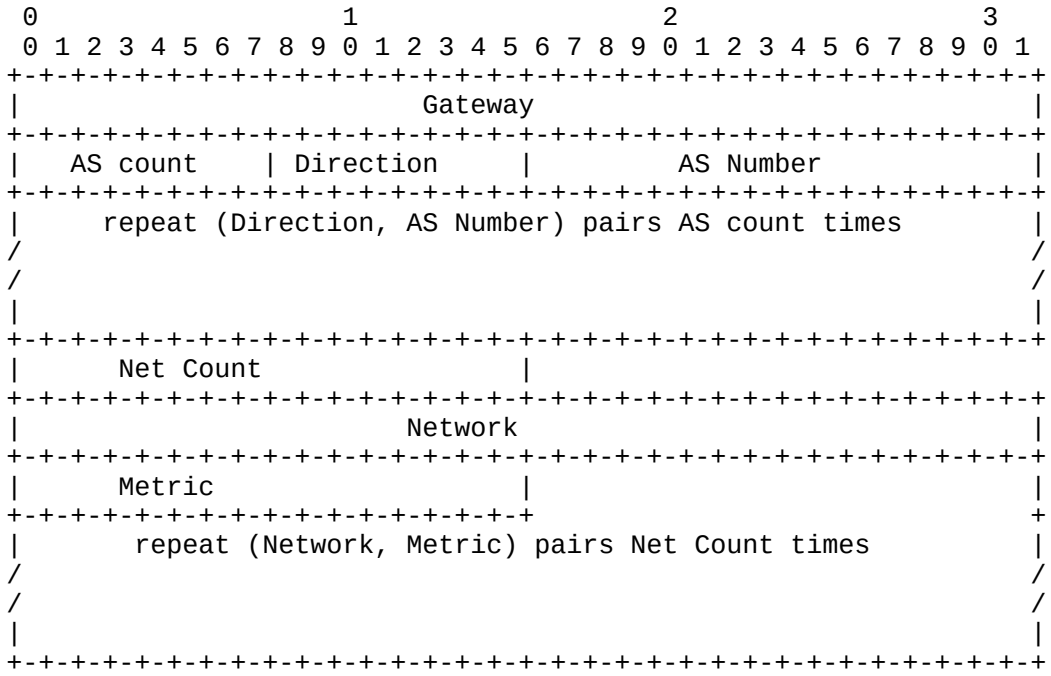
An OPEN CONFIRM message is sent after receiving an OPEN message. This completes the BGP connection setup. UPDATE, NOTIFICATION, and KEEPALIVE messages may now be exchanged.

An OPEN CONFIRM message consists of a BGP header with an OPEN CONFIRM type code. There is no data in an OPEN CONFIRM message.

3.4 UPDATE Message Format

UPDATE messages are used to transfer routing information between BGP peers. The information in the UPDATE packet can be used to construct a graph describing the relationships of the various autonomous systems. By applying rules to be discussed, routing information loops and some other anomalies may be detected and removed from the inter-AS routing.

Whenever an error in a UPDATE message is detected, a notification message is sent with opcode 4 (bad update), a two byte subcode describing the nature of the problem, and a data field consisting of as much of the UPDATE message data portion as possible. UPDATE messages have the following format:



Gateway: 32 bits.

The Gateway field is the address of a gateway that has routes to the Internet networks listed in the rest of the UPDATE message. This gateway MUST belong to the same AS as the BGP peer who advertises it. If there is a problem with the gateway field, a notification message with subcode 6 (invalid gateway field) is sent.

AS count: 8 bits.

This field is the count of Direction and AS Number pairs in this UPDATE message. If an incorrect AS count field is detected, subcode 1 (invalid AS count) is specified in the notification message.

Direction: 8 bits

The Direction field is an octet containing the direction taken by the routing information when exiting the AS defined by the succeeding AS Number field. The following values are defined.

```
1  - UP            (went up a link in the graph)
2  - DOWN          (went down a link in the graph)
3  - H_LINK        (horizontal link in the graph)
4  - EGP_LINK      (EGP derived information)
5  - INCOMPLETE    (incomplete information)
```

There is a special provision to pass exterior learned (non-BGP) routes over BGP. If an EGP learned route is passed over BGP, then the Direction field is set to EGP-LINK and the AS Number field is set to the AS number of the EGP peer that advertised this route. All other exterior-learned routes (non-BGP and non-EGP) may be passed by setting AS Number field to zero and Direction field to INCOMPLETE. If the direction code is not recognized, a notification message with subcode 2 (invalid direction code) is sent.

AS Number: 16 bits

This field is the AS number that transmitted the routing information. If there is a problem with this AS number, a notification message with subcode 3 (invalid autonomous system) is sent.

Net Count: 16 bits.

The Net Count field is the number of Metric and Network field pairs which follow this field. If there is a problem with this field, a notification with subcode 7 (invalid net count field) is sent.

Network: 32 bits

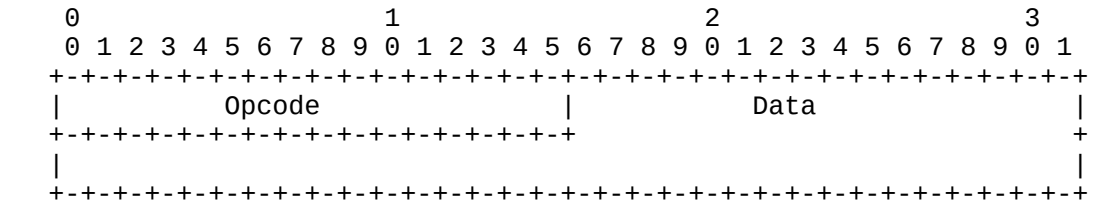
The Network field is four bytes of Internet network number. If there is a problem with the network field, a notification message with subcode 8 (invalid network field) is sent.

Metric: 16 bits

The Metric field is 16 bits of an unspecified metric. BGP metrics are comparable ONLY if routes have exactly the same AS path. A metric of all ones indicates the network is unreachable. In all other cases the metric field is MEANINGLESS and MUST BE IGNORED. There are no illegal metric values.

3.5 NOTIFICATION Message Format

NOTIFICATION messages are sent when an error condition is detected. The BGP connection is closed shortly after sending the notification message.



Opcode: 16 bits

The Opcode field describes the type of NOTIFICATION. The following opcodes have been defined.

1 (*) - link type error in open. Data is one byte of proper link type.
2 (*) - unknown authentication code. No data.
3 (*) - authentication failure. No data.
4 - update error. See below for data description.
5 (*) - connection out of sync. No data.
6 (*) - invalid message length. Data is two bytes of bad length.
7 (*) - invalid message type. Data is one byte of bad message type.
8 (*) - invalid version number. Data is one byte of bad version.
9 (*) - invalid AS field in OPEN. No data.
10 (*) - BGP Cease. No data.

The starred opcodes in the list above are considered fatal errors and cause transport connection termination.

The update error (opcode 4) has as data 16 bits of subcode followed by the last UPDATE message in question. After the subcode comes as much of the data portion of the UPDATE in

question as possible. The following subcodes are defined:

1 - invalid AS count
2 - invalid direction code
3 - invalid autonomous system
4 - EGP_LINK or INCOMPLETE_LINK link type at other than the end of the AS path list
5 - routing loop
6 - invalid gateway field
7 - invalid Net Count field
8 - invalid network field

Data: variable

The Data field contains zero or more bytes of data to be used in diagnosing the reason for the NOTIFICATION. The contents of the Data field depend upon the opcode. See the opcode descriptions above for more details.

3.6 KEEPALIVE Message Format

BGP does not use any transport protocol based keepalive mechanism to determine if peers are reachable. Instead KEEPALIVE messages are exchanged between peers often enough as not to cause the hold time (as advertised in the BGP header) to expire. A reasonable minimum frequency of KEEPALIVE exchange would be one third of the Hold Time interval.

As soon as the Hold Time associated with BGP peer has expired, the BGP connection is closed and BGP deallocates all resources associated with this peer.

The KEEPALIVE message is a BGP header without any data.

4. BGP Finite State machine.

This section specifies BGP operation in terms of a Finite State Machine (FSM). Following is a brief summary and overview of BGP operations by state as determined by this FSM. A condensed version of the BGP FSM is found in Appendix 1.

Initially BGP is in the BGP_Idle state.

BGP_Idle state:

In this state BGP refuses all incoming BGP connections. No resources are allocated to the BGP neighbor. In response to the Start event (initiated by either system or operator) the local

system initializes all BGP resources and changes its state to BGP_Active.

BGP_Active state:

In this state BGP is trying to acquire a BGP neighbor by opening a transport protocol connection. If the transport protocol open fails (for example, retransmission timeout), BGP stays in the BGP_Active state.

Otherwise, the local system sends an OPEN message to its peer, and changes its state to BGP_OpenSent. Since the hold time of the peer is still undetermined, the hold time is initialized to some large value.

In response to the Stop event (initiated by either system or operator) the local system releases all BGP resources and changes its state to BGP_Idle.

BGP_OpenSent state:

In this state BGP waits for an OPEN message from its peer. When an OPEN message is received, all fields are checked for correctness. If the initial BGP header checking detects an error, BGP deallocates all resources associated with this peer and returns to the BGP_Active state. Otherwise, the Link Type, Authentication Code, and Authentication Data fields are checked for correctness.

If the link type is incorrect, a NOTIFICATION message with opcode 1 (link type error in open) is sent. The following combination of link type fields are correct; all other combinations are invalid.

Our view	Peer view
UP	DOWN
DOWN	UP
INTERNAL	INTERNAL
H-LINK	H-LINK

If the link between two peers is INTERNAL, then AS number of both peers must be the same. Otherwise, a NOTIFICATION message with opcode 1 (link type error in open) is sent.

If both peers have the same AS number and the link type between these peers is not INTERNAL, then a NOTIFICATION message with opcode 1 (link type error in open) is sent.

If the value of the Authentication Code field is zero, any

information in the Authentication Data field (if present) is ignored. If the Authentication Code field is non-zero it is checked for known authentication codes. If authentication code is unknown, then the BGP NOTIFICATION message with opcode 2 (unknown authentication code) is sent.

If the Authentication Code value is non-zero, then the corresponding authentication procedure is invoked. The default values are a zero Authentication Code and no Authentication Data.

If any of the above tests detect an error, the local system closes the BGP connection and changes its state to BGP_Idle.

If there are no errors in the BGP OPEN message, BGP sends an OPEN CONFIRM message and goes into the BGP_OpenConfirm state. At this point the hold timer which was originally set to some arbitrary large value (see above) is replaced with the value indicated in the OPEN message.

If disconnect notification is received from the underlying transport protocol or if the hold time expires, the local system closes the BGP connection and changes its state to BGP_Idle.

BGP_OpenConfirm state:

In this state BGP waits for an OPEN CONFIRM message. As soon as this message is received, BGP changes its state to BGP_Established. If the hold timer expires before an OPEN CONFIRM message is received, the local system closes the BGP connection and changes its state to BGP_Idle.

BGP_Established state:

In the BGP_Established state BGP can exchange UPDATE, NOTIFICATION, and KEEPALIVE messages with its peer.

If disconnect notification is received from the underlying transport protocol or if the hold time expires, the local system closes the BGP connection and changes its state to BGP_Idle.

In response to the Stop event initiated by either the system or operator, the local system sends a NOTIFICATION message with opcode 10 (BGP Cease), closes the BGP connection, and changes its state to BGP_Idle.

5. UPDATE Message Handling

A BGP UPDATE message may be received only in the BGP_Established state. When a BGP UPDATE message is received, each field is checked for validity. When a NOTIFICATION message is sent regarding an UPDATE, the opcode is always 4 (update error), the subcode depends on the type of error, and the rest of the data field is as much as possible of the data portion of the UPDATE that caused the error.

If the Gateway field is incorrect, a BGP NOTIFICATION message is sent with subcode 6 (invalid gateway field). All information in this UPDATE message is discarded.

If the AS Count field is less than or equal to zero, a BGP NOTIFICATION is sent with subcode 1 (invalid AS count). Otherwise, the complete AS path is extracted and checked as described below.

If one of the Direction fields in the AS route list is not defined, a BGP NOTIFICATION message is with subcode 2 (invalid direction code).

If one of the AS Number fields in the AS route list is incorrect, a BGP NOTIFICATION message is sent with subcode 3 (invalid autonomous system).

If either a EGP_LINK or a INCOMPLETE_LINK link type occurs at other than the end of the AS path, a BGP NOTIFICATION message is sent with subcode 4 (EGP_LINK or INCOMPLETE_LINK link type at other than the end of the AS path list).

If none of the above tests failed, the full AS route is checked for AS loops.

AS loop detection is done by scanning the full AS route and checking that each AS in this route occurs only once. If an AS loop is detected, a BGP NOTIFICATION message is sent with subcode 5 (routing loop).

If any of the above errors are detected, no further processing is done. Otherwise, the complete AS path is correct and the rest of the UPDATE message is processed.

If the Net Count field is incorrect, a BGP NOTIFICATION message is sent with subcode 7 (invalid Net Count field).

Each network and metric pair listed in the BGP UPDATE message is checked for a valid network number. If the Network field is incorrect, a BGP Notification message is sent with subcode 8 (invalid network field). No checking is done on the metric field. It is up

to a particular implementation to decide whether to continue processing or terminate it upon the first incorrect network.

If the network, its complete AS path, and the gateway are correct, then the route is compared with other routes to the same network. If the new route is better than the current one, then it is flooded to other BGP peers as follows:

- If the BGP UPDATE was received over the INTERNAL link, it is not propagated over any other INTERNAL link. This restriction is due to the fact that all BGP gateways within a single AS form a completely connected graph (see above).

- Before sending a BGP UPDATE message over the non-INTERNAL links, check the AS path to insure that doing so would not cause a routing loop. The BGP UPDATE message is then propagated (subject to the local policy restrictions) over any of the non-INTERNAL link of a routing loop would not result.

- If the BGP UPDATE message is propagated over a non-INTERNAL link, then the current AS number and link type of the link over which it is going to be propagated is prepended to the full AS path and the AS count field is incremented by 1. If the BGP UPDATE message is propagated over an INTERNAL link, then the full AS path passed unmodified and the AS count stays the same. The Gateway field is replaced with the sender’s own address.

6. Acknowledgements

We would like to express our thanks to Len Bosack (cisco Systems), Jeff Honig (Cornell University) and all members of the IWG task force for their contributions to this document.

Appendix 1

BGP FSM State Transitions and Actions.

This Appendix discusses the transitions between states in the BGP FSM in response to BGP events. The following is the list of these states and events.

BGP States:

1 - BGP_Idle
2 - BGP_Active
3 - BGP_OpenSent
4 - BGP_OpenConfirm
5 - BGP_Established

BGP Events:

1 - BGP Start
2 - BGP Transport connection open
3 - BGP Transport connection closed
4 - BGP Transport connection open failed
5 - Receive OPEN message
6 - Receive OPEN CONFIRM message
7 - Receive KEEPALIVE message
8 - Receive UPDATE messages
9 - Receive NOTIFICATION message
10 - Holdtime timer expired
11 - KeepAlive timer expired
12 - Receive CEASE message
13 - BGP Stop

The following table describes the state transitions of the BGP FSM and the actions triggered by these transitions.

Event	Actions	Message Sent	Next State
BGP_Idle (1)			
1	Initialize resources	none	2
BGP_Active (2)			
2	Initialize resources	OPEN	3
4	none	none	2
13	Release resources	none	1
BGP_OpenSent(3)			
3	none	none	1
5	Process OPEN is OK	OPEN CONFIRM	4
	Process OPEN Message failed	NOTIFICATION	1
11	Restart KeepAlive timer	KEEPALIVE	3
13	Release resources	none	1
BGP_OpenConfirm (4)			
6	Complete initialization	none	5
3	none	none	1
10	Close transport connection	none	1
11	Restart KeepAlive timer	KEEPALIVE	4
13	Release resources	none	1
BGP_Established (5)			
7	Process KEEPALIVE	none	5
8	Process UPDATE is OK	UPDATE	5
	Process UPDATE failed	NOTIFICATION	5
9	Process NOTIFICATION	none	5
10	Close transport connection	none	1
11	Restart KeepAlive timer	KEEPALIVE	5
12	Close transport connection	NOTIFICATION	1
13	Release resources	none	1

All other state-event combinations are considered fatal errors and cause the termination of the BGP transport connection (if necessary) and a transition to the BGP_Idle state.

The following is a condensed version of the above state transition table.

Events	BGP_Idle (1)	BGP_Active (2)	BGP_OpenSent (3)	BGP_OpenConfirm (4)	BGP_Estab (5)
1	2				
2		3			
3			1	1	
4		2			
5			4 or 1		
6				5	
7					5
8					5
9					5
10				1	1
11			3	4	5
12					1
13		1	1	1	1

References

[1] Mills, D., "Exterior Gateway Protocol Formal Specification", RFC 904, BBN, April 1984.

[2] Rekhter, Y., "EGP and Policy Based Routing in the New NSFNET Backbone", RFC 1092, T. J. Watson Research Center, February 1989.

[3] Braun, H-W., "The NSFNET Routing Architecture", RFC 1093, MERIT/NSFNET Project, February 1989.

[4] Postel, J., "Transmission Control Protocol - DARPA Internet Program Protocol Specification", RFC 793, DARPA, September 1981.

[5] Cheriton, D., "VMTP: Versatile Message Transaction Protocol", RFC 1045, Stanford University, February 1988.

Authors' Addresses

Kirk Lougheed
cisco Systems, Inc.
1360 Willow Road, Suite 201
Menlo Park, CA 94025

Phone: (415) 326-1941

Email: LOUGHEED@MATHOM.CISCO.COM

Jacob Rekhter
T.J. Watson Research Center
IBM Corporation
P.O. Box 218
Yorktown Heights, NY 10598

Phone: (914) 945-3896

Email: YAKOV@IBM.COM