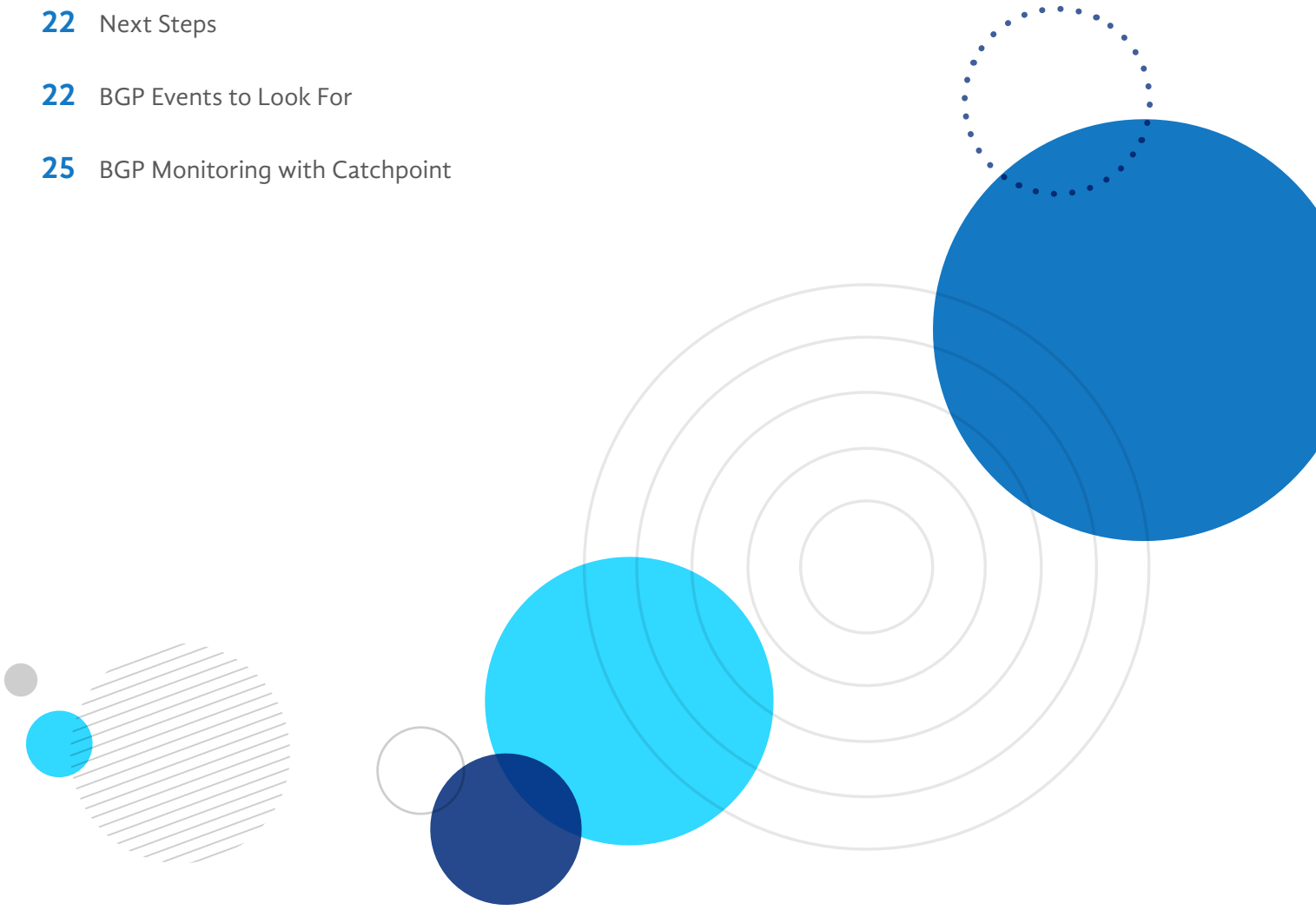
catchpoint

NETWORK EXPERIENCE

The Comprehensive Guide to BGP

Table of Contents



Introduction: Why Care About BGP?

Thank you for taking a few minutes to download and read Catchpoint's Comprehensive Guide to BGP. This useful document will teach you about the history of BGP, how it operates to route traffic, what this means to your bottom line and some of its vulnerabilities. But before diving into the details, let's just take a step back for a minute and consider one simple question: why do we still care about BGP?

BGP IS the Internet

As you'll see, BGP has been around for almost as long as the commercial Internet. In fact, it's fair to say that in many ways BGP actually IS the Internet. This means that if you want to understand how the Internet works, you must first understand BGP since so much of the Internet relies upon it.

The Challenges Posed by BGP Problems

In a modern, cloud-based environment, your organization has outsourced functionality that was once completely under your control or in your own Autonomous System (AS). Now that key business functionality is spread across dozens — or even hundreds! — of other AS's. And you have little to no visibility into or control over any of them. There are a million reasons why this has happened, but the result is the same: your environment is exponentially more complex and relies on BGP routing more than ever.

This means exponentially more trouble for you when there's a BGP problem. While BGP issues were never minor, this new, vastly more complex environment means that they are going to impact more of your systems and be harder than ever to diagnose. However, the other side of the coin is that because BGP is so ubiquitous, it can be an excellent metric for monitoring your environment and measuring the digital experience of your users.

What Visibility Into BGP Can Provide

A BGP status dashboard can make it very simple to identify and diagnose network issues that are caused by BGP — or to quickly eliminate BGP as the culprit. Viewing these ever-more-complicated BGP paths can help determine exactly where routing problems are occurring and whether or not they're due to misconfiguration or malice. In fact, almost any troubleshooting of a complex network of systems will require visibility into BGP in order to gain the insight needed to fix problems.

So as we go into a deeper examination of what BGP is, how it works and why it sometimes doesn't, bear in mind that this isn't just an exploration of ancient history that's still lingering around. BGP is a foundational part of networking that can tell you a lot about the health of your environment while providing detailed, granular data that can speed issue diagnosis and accelerate MTTR.

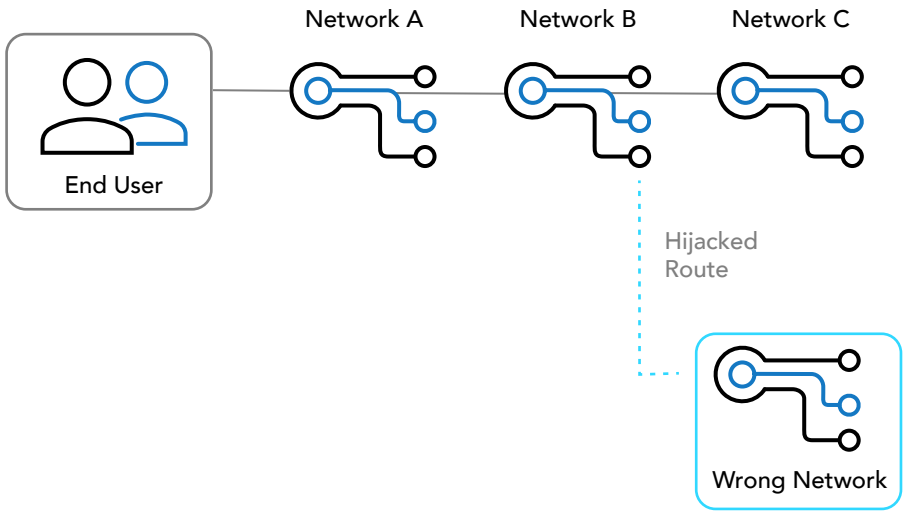
The Border Gateway Protocol (BGP) turned 33 years old this year, making it one of the most long-lasting, widely-used protocols ever deployed on the Internet. BGP was initially conceived in January, 1989 by Yakov Rekhter (IBM) and Kirk Lougheed (Cisco) on **two napkins during the 12th IETF conference** in Austin, Texas.

Curiously enough, **BGP was conceived as an interim solution** to overcome the infeasibility of using the existing Exterior Gateway Protocol (EGP) with the increase in complexity for connectivity between Administrative Domains.

Thirty years passed, and the interim solution became one of the pillars of the Internet architecture. Version 4 (the current version) was released in 1994, and since then it has been sporadically updated with new features and capabilities.

Some Basics

Before we dive into the history of BGP, let's go over some basics of what it is. The primary function of BGP is to manage how packets are routed across the Internet through the exchange of routing and reachability information between edge routers. BGP directs traffic between autonomous systems (AS), which are network routers managed by a single enterprise or service provider. When an AS gets set up, it peers with other autonomous systems to share IP prefixes, which are then shared with other autonomous systems, and so on. In this way, when new prefixes are announced, they get propagated around the Internet.



The biggest problem, however, is that BGP is extremely vulnerable to both malicious attacks and human error. There are roughly 90,000 autonomous systems that make up the global Internet, and little to no oversight for how each AS peering filter must be configured. This means that if a new, bogus route (aka a bogus prefix) is announced (either through intentional hijacking or just a typo) it sends traffic to the wrong network, and can spread like wildfire across the Internet.

A Little Bit of History

Some background is required to better understand the crucial role that BGP has played in the history of the Internet. In 1989, the Internet as we know it today was in its inception. Using it commercially was forbidden, a restriction lifted in 1995 with the decommission of NSFNET, but commercial ISPs were sprouting and offering network access to end users, and the commercial use of the Internet was no longer taboo.

When **BGP was initially standardized in June, 1989**, the long-running ARPANET was just being decommissioned (February 28, 1989), TCP/IP was being used to interconnect different networks from remote countries, and the Internet was about to move from its centric architecture to a more distributed architecture, without a clearly defined backbone. Curiously, the **requiem to ARPANET by Vinton G. Cerf** was performed in the very same IETF meeting where BGP was just being announced to the world.

Up until then, the so-called Internet gateways were exchanging net-reachability information via the **Exterior Gateway Protocol (EGP)**. EGP was conceived for an Internet composed of a core AS and multiple other smaller autonomous systems directly connected to that core, and it totally relied on having a tree-structured topology of autonomous systems, without cycles.

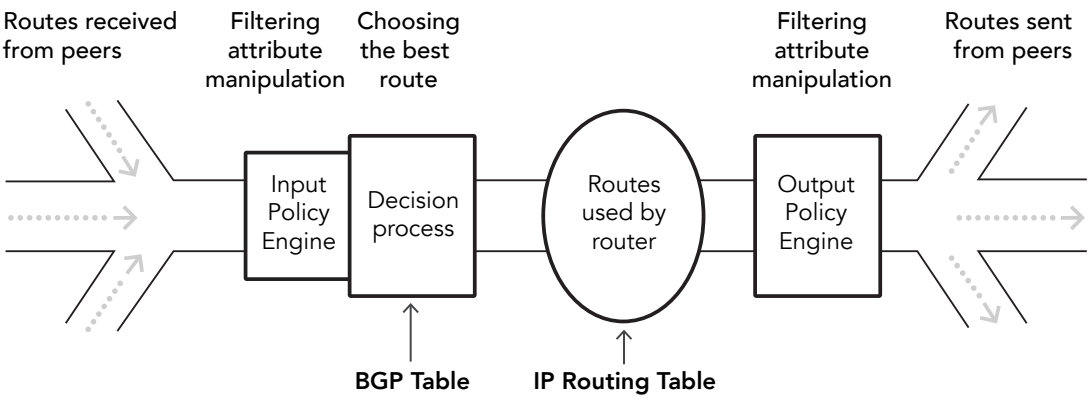
These limitations were bearable in an early stage Internet where stub gateways were talking to each other via the ARPANET backbone. With the advent of commercial entities and multiple backbones (such as NSFNET), its inadequacies became more and more pronounced — not to mention the impossibility of creating policy-based routing, which is key to the success of BGP.

BGP: The Two-Napkin Protocol

BGP is still a path vector protocol like its predecessor EGP, but it was conceived foreseeing a peer-to-peer environment where autonomous systems could exchange routing information without relying either on a priori topology knowledge or on a core AS. With the introduction of BGP, the concept of AS was also changed and re-defined. In the **last BGP version**, an AS "is considered to be a set of routers under a single technical administration, using an interior gateway protocol (IGP) and common metrics to determine how to route packets within the AS, and using an inter-AS routing protocol to determine how to route packets to other autonomous systems."

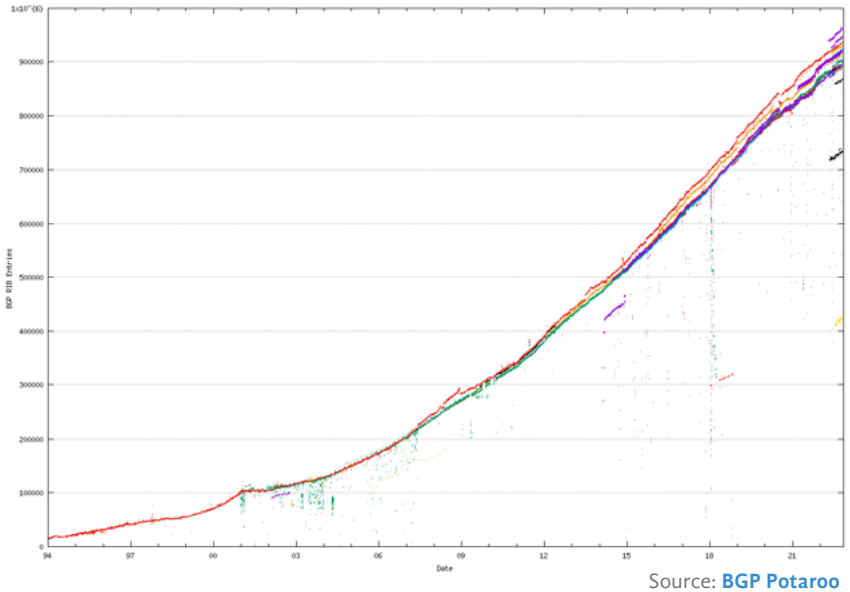
The basic piece of routing information that autonomous systems exchange with each other is called a route. A route is composed of a set of destination IP networks paired with set path attributes, which describe the path toward the destinations: "**This information is sufficient to construct a graph of AS connectivity from which routing loops may be pruned and policy decisions at an AS level may be enforced**."

To guarantee the reliability of transmission, BGP is encapsulated into a TCP connection, meaning that two routers that want to establish a BGP session must have prior IP reachability. After establishing a TCP connection, the two routers — hereafter called BGP peers — agree on the parameters to use in the BGP session via BGP open messages, and then start exchanging routes. These routes can be generated by the peer itself or they can be learned by the peer via other BGP sessions, and each of them is announced via BGP update messages.



The figure above summarizes the BGP process each AS applies when receiving a route from another peer.

Whenever a route is received from an AS, the route is subject to a filtering process where it can be discarded or accepted and, if required, its path attributes are manipulated. Then a BGP decision process is applied to select the best route for each IP destination network, since an AS may receive multiple routes toward the same IP network from different peers.



Source: **BGP Potaroo**

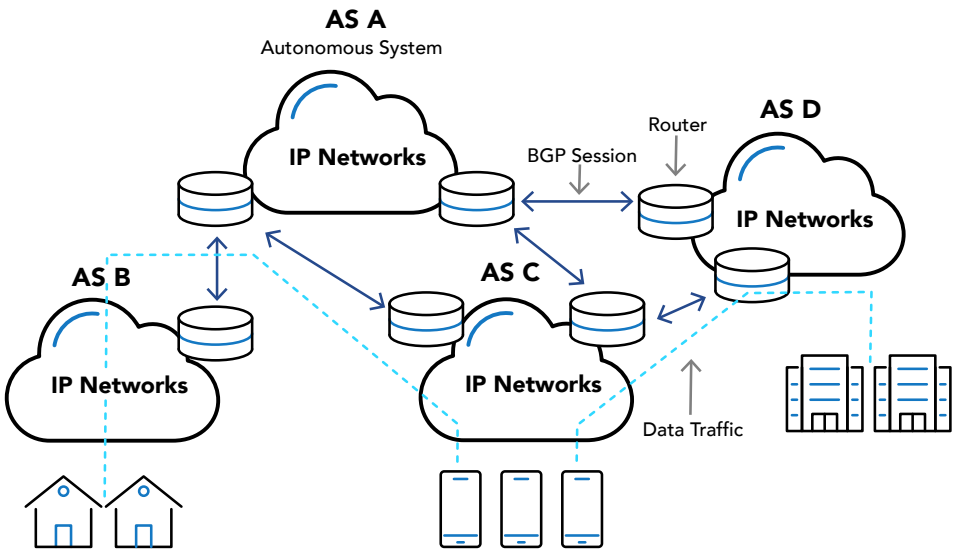
The BGP decision process is composed of a sequence of steps that allow the AS to choose the best route by analyzing the path attributes of each of the candidates, in order to apply criteria that range from purely commercial (e.g. preferring a cheaper provider over the other) to technical reasons (e.g. transit traffic wanting to reach a destination via the smallest number of autonomous systems). Each best route is then installed in the routing table of the router and used to forward traffic. Eventually, after a proper attribute manipulation, each best route is propagated to the all other BGP peers, or a subset of them depending upon the output filtering process applied.

Since the early days of deployment of BGP, the **Internet grew widely in size** and shape. Nowadays, the Internet is composed of about 90,000 autonomous systems that exchange routing information related to about **900,000 IPv4 networks and about 150,000 IPv6 networks**. However, due to the distributed architecture of the Internet, it's impossible to determine the number of BGP sessions established among autonomous systems in the wild.

X-Raying BGP

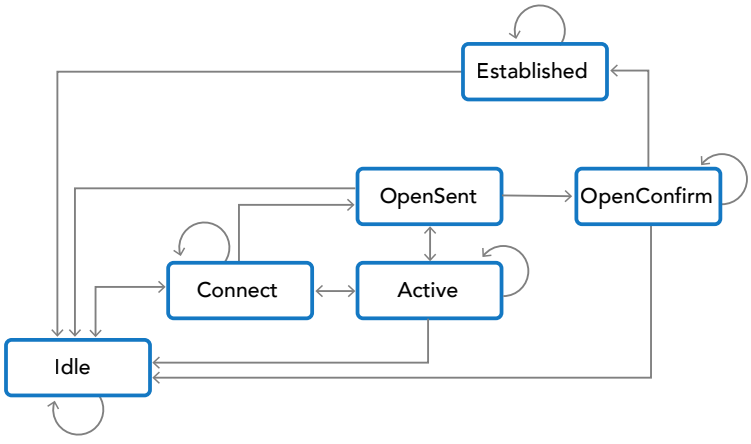
Two autonomous systems exchange routing information by establishing BGP session(s) between pairs of routers running a BGP daemon, namely BGP speakers. After establishing a BGP session, the BGP speakers start exchanging the set of network prefixes that they either received from other autonomous systems or that they already possessed. Each network destination exchanged is paired with a set of attributes that describes the characteristics of the path to reach that destination, forming what is called a route. Eventually, a BGP speaker will receive the routes related to all the Internet destinations. With these routes, the BGP speaker can forward traffic towards any intended destination.

The set of attributes associated with each route enables a BGP speaker to implement routing policies which may reflect either commercial agreements it has with its neighbors or technical considerations. This flexibility is one of the key factors that allowed BGP to become the standard de-facto routing protocol of the Internet.



BGP Operations

BGP protocol is quite unique in the family of the routing protocols. Its most relevant peculiarity is that it relies on TCP (port 179) to guarantee the ordered and reliable exchange of protocol messages. This is because — unlike other routing protocols — there is no peer discovery process, and each peer is statically configured by the network administrator. Indeed, BGP is conceived to be an inter-AS protocol where peers should have quite a large degree of stability, thus making the discovery process useless. Therefore, it is a sine qua non condition for two BGP speakers to have IP reachability.



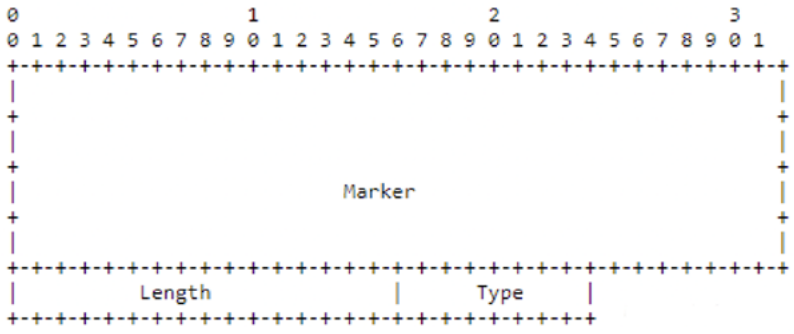
The process of establishing a BGP session between peers is performed via a simple finite state machine (FSM), which is described in **the original RFC** and can be summarized in the figure above.

The process can be simplified as follows:

- Each BGP speaker starts in *Idle* state. This is a transient state where the BGP speaker initializes the required resources for the connection, and where it starts to listen for TCP attempt connections on port 179 and, at the same time, attempts to connect to the other BGP speaker via TCP on port 179.
- Once these steps are performed, the BGP speaker moves to the *Connect* state, where it sets a timer and waits for the TCP connection to be completed. If the *ConnectRetryTimer* expires, the TCP connection is dropped, and the timer resets while still listening for incoming TCP connection attempts. If the TCP connection fails, the BGP speaker moves to the *Active* state.
- The *Active* state is another transient state where the BGP speaker basically stops to actively attempt to connect to the other party and just listen for incoming TCP connection attempts. Once the *ConnectRetryTimer* expires, the BGP speaker goes back to *Connect* state.
- If the TCP connection is successful either in *Connect* or *Active* states, the BGP speaker must send an *OPEN* message containing a list of its capabilities, moving to the *OpenSent* state.
- Once in *OpenSent* state, a BGP speaker waits for an *OPEN* message from the other party, and if no error occurs, sends a *KEEPALIVE* message, moving then to *OpenConfirm* state. Otherwise, it sends a *NOTIFICATION* message and goes back to *Idle* state.
- Finally, in the *OpenConfirm* state the BGP speaker waits for a *KEEPALIVE* message or a *NOTIFICATION* message from the other party. If it receives a *KEEPALIVE* message, then it moves to the *Established* state, otherwise it moves back to *Idle*. In general, any error in any state cause the BGP speaker to move to *Idle* state.
- Once in *Established* state, each BGP speaker announces *routes* towards via *UPDATE* messages to allow the other party to reach those destinations. The number of destinations announced depends on the agreement between network operators.

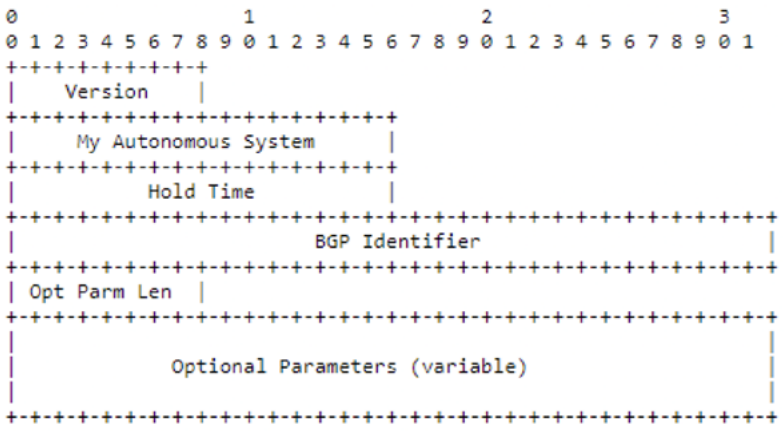
Messages

The evolution of the FSM described above is regulated via the exchange of BGP messages. Each BGP message starts with a common BGP header composed of 19 bytes and encoded as follows:



BGP header format courtesy of **IETF**

The Marker is a 16-byte field set all to one. This field is included for compatibility with older BGP versions and has no specific semantic in the current BGP version. The Length field contains the length of the BGP message, header included. **The original RFC** specifies that the maximum length of a BGP message is 4096 bytes despite the field size. This value was considered to be **more than enough for the protocol requirements**. Finally, the Type field contains the type of the message. The most common BGP message types are four: *OPEN (1), NOTIFICATION (2), KEEPALIVE (3), and UPDATE (4)*.



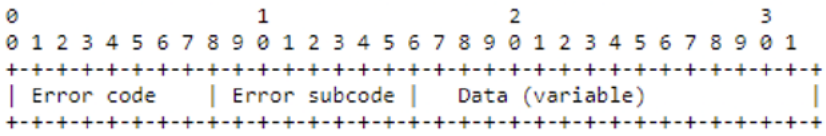
BGP header format courtesy of **IETF**

Open messages

The most important type of messages in the initial setup phase of a BGP session are the *Open messages*. As detailed in the FSM, these messages are used by the two BGP speakers to inform each other about the parameters they propose to use for the BGP session and inform the other party about their *capabilities*. Capabilities are **Optional Parameters** describing which BGP extension each speaker supports, like the support for **four-octet AS numbers**, the support of **multiple protocols in BGP (e.g. IPv6)**, and the **support for multiple paths**. If two BGP speakers share a common capability, they will be automatically enabled to exploit the capability new features in the BGP session, such as announcing each other's IPv6 routes.

In addition to capabilities, Open messages carry the current Version of BGP (set to 4 since 1994) and some information about each of the peer, like the self-explanatory *My Autonomous System* field and the BGP *identifier* field, where one of the IPv4 addresses belonging to the announcing BGP speaker is encoded.

Another important mandatory parameter found in these messages is the *Hold Time*, which regulates how long the BGP session can stay up without the exchange of any protocol message. This parameter is crucial to avoid the reset of the session upon a temporary network failure; however, its value cannot be too high otherwise it could take too long for a speaker to realize that the session is no longer available. The **BGP specification** suggests using 90 seconds for that value.



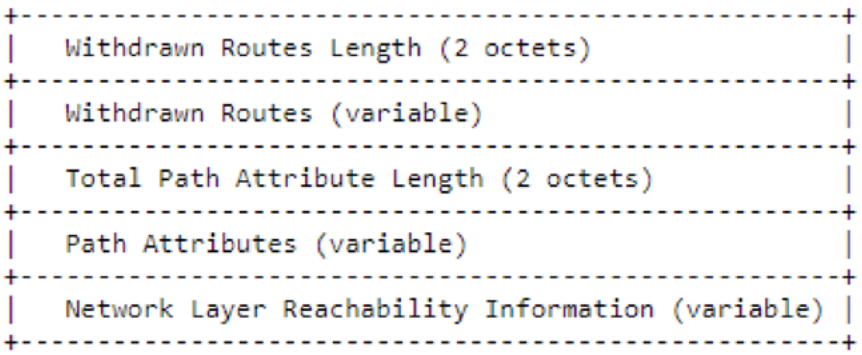
NOTIFICATION message courtesy of **IETF**

Notification messages

Notification messages are quite the opposite of *Open messages*. They are triggered whenever one of the BGP speaker incurs an error (for any reason). In these cases, the BGP speaker notifies the other party about the type of error it has experienced, just before tearing down the BGP session. The possibility of adding **text to this message** was recently introduced to show readable information in the log of the other BGP speaker.

Keepalive messages

Keepalive messages are empty BGP messages, composed by the header and without any payload. These simple messages are used to acknowledge decisions (like in the setup phase) and to keep the session up in the absence of routing information exchanged by the two BGP speakers.



UPDATE message courtesy of **IETF**

Update messages

Finally, there are the *Update messages*. These are the messages that carry the routing information that autonomous systems exchange with each other. An *Update* message can be thought to be composed of three main parts: *Withdrawn Routes*, *Path attributes* and *Network Layer Reachability Information (NLRI)*.

***Withdrawn routes* and *NLRI* fields** are quite straightforward. They contain the subnets that are the subject of the route carried by the UPDATE message. If the subnet is among the *Withdrawn routes*, it means that the BGP speaker has no more routes involving that specific subnet. If the subnet is among the *NLRI*, then it means that the BGP speaker found a new route to for that specific subnet, whose characteristics are described in the *Path attributes* field. This can either mean that a new subnet has been announced in the Internet, or that an already existing subnet could be reached via a different route.

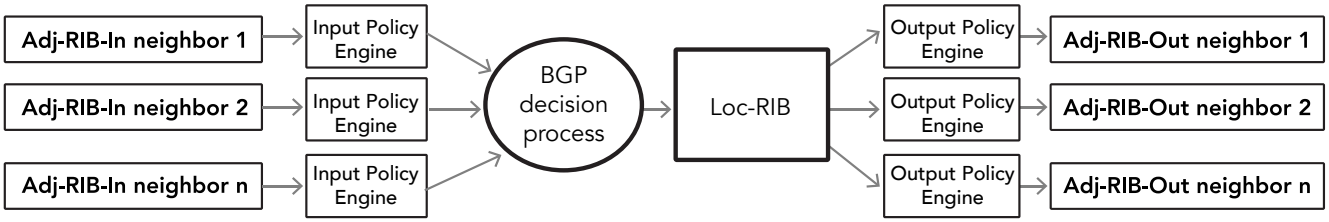
The *Path Attributes* field contains a set of attributes that describe the path toward the destinations contained in the *NLRI* field. There are many different attributes, each with its own role and format. **The original RFC** mandates that the *Path Attributes* field must be present if the *NLRI* field contains at least one destination, but only a few *Path Attributes* are required to be present:

- AS_PATH: the list of autonomous systems that must be traversed to reach the destination via the given route
- ORIGIN: the origin of path information
- NEXT_HOP: the IP address of the router that should be used as next hop towards the given destination

Route Elaboration

Once the session is established, the two BGP speakers announce *Update messages* to each other to advertise their reachability to the other party. Since BGP speakers could be connected to multiple BGP speakers — possibly belonging to different autonomous systems — and thus could receive multiple routes toward the very same destination, each BGP speaker must run a decision process to select the best route for each subnet received. This is called the *BGP decision process*, and that's exactly where BGP conveys its flexibility.

Going into further detail, when a BGP speaker receives a BGP update message from a neighbor, it stores each route advertised in a table dedicated to that neighbor, called *Adj-RIB-In*.



Once a route has been installed in the Adj-RIB-In, it is checked against a filter to decide whether it can be accepted or not. The ingress filter is completely customizable by the network administrator, who can decide to discard a route for a plethora of reasons. A route could be discarded, for example, if the destination network was not expecting it to be received from that specific neighbor, or if it contains a specific path attribute value.

If a route is accepted, then it participates in the BGP decision process, together with all the accepted routes toward the same destination learned from other neighbors. This process is conceptually comprised of three phases, with some slight differences from router vendor to router vendor.

Phase one

The first phase is triggered whenever an *Update* message has been accepted and entails/involves calculating a degree of preference for each route advertised in the message.

Phase two

Once this phase is completed, the BGP speaker chooses the *best route* among all the routes available for each distinct destination in the message and installs each best route in the *Local Routing Information Base (Loc-RIB)*. The *Loc-RIB* is the table containing all the routes the BGP speaker is using to route the traffic received from its neighbors.

Phase three

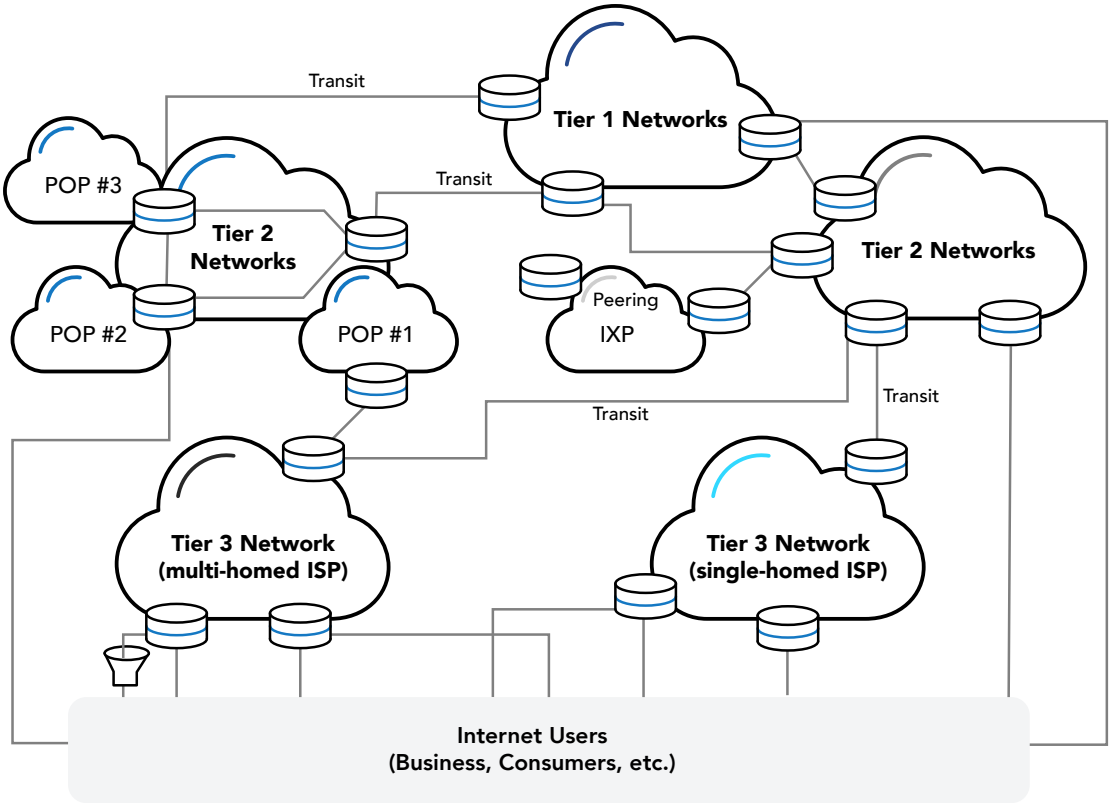
The third phase is triggered once the *Loc-RIB* has been modified. In this phase, each route that contributed to change the Loc-RIB is checked against neighbor-specific output filters. From there, it's installed into the neighbor Adj-RIB-out and becomes ready to be advertised. Like the ingress filters, the output filters are completely customizable by the network administrator who could decide, for example, to exclude a neighbor from receiving a route towards a specific destination.

BGP and Your Brand's Bottom Line

The Internet still works on the very same foundations that it started with back in the early 90s. A limited number of companies (e.g. CenturyLink, AT&T, Verizon) offer transit via their worldwide backbone to a much larger number of companies. Among them, we have eyeball networks such as most national telcos (e.g. Telecom Italia, British Telecom), regional providers (e.g. Apuacom), and CDNs (e.g. Comcast, Sky), all aiming at providing the best performance to end users — even if from different perspectives — via their interconnections, often established on Internet Exchange Points (IXPs).

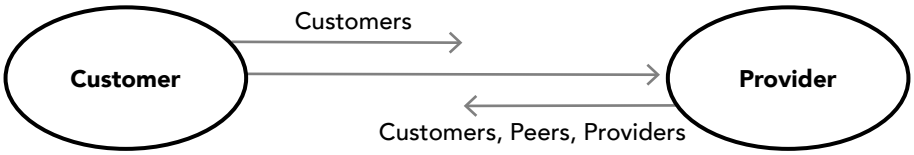
In the last few decades, big players (such as Google, Facebook, and Netflix) have joined this interconnection game with enough resources to mine their own foundation. They decided to play on their own by creating a closed worldwide infrastructure, interconnecting directly to as many autonomous systems as possible and offering direct access to their content, thus completely bypassing third-party backbones.

Today, about **90k autonomous systems are interconnected to each other, mostly regionally**, via BGP. Each enforces its role via a specific set of import/export filter policies. Among these 90k, only about twenty of them can reach all Internet destinations without purchasing transit from any other AS, **forming the so-called Tier-1 club**.



Economic Relationships and BGP

The Internet is a complex system made of interconnected autonomous systems, each with its own role, market, and resources. Nevertheless, it's possible to roughly categorize the type of relationships existing between autonomous systems in two broad categories, each identified by a BGP import/export filter policy.



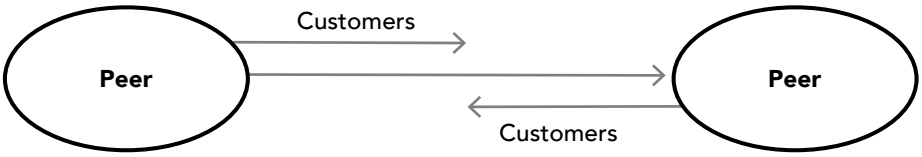
Provider-to-customer

The first relationship is **provider-to-customer (p2c)**, or **customer-to-provider (c2p)**, where one of the two autonomous systems provides transit to the whole Internet for the other AS. This type of relationship is often on a contract basis involving a fee paid by the customer to the provider, and it is established via private facilities.

To fulfill its role, the provider announces to the customer every route required to reach the whole Internet. Depending on the agreement, this could consist of a single default route (0.0.0.0/0 in IPv4 and ::/0 in IPv6) or of a full routing table, which as of today consists of about 900k subnets in IPv4 and about 150k subnets in IPv6. On the other side, the customer will announce to the provider only its own routes and the routes received from its customers to allow the provider to use their interconnection to reach those destinations.

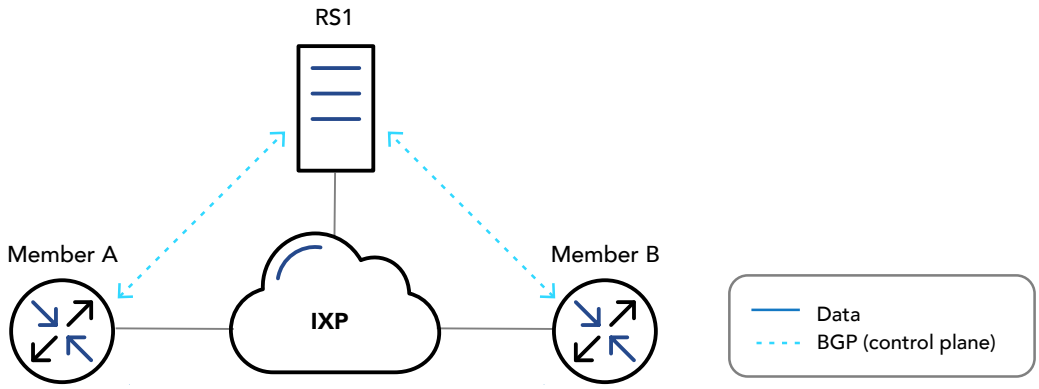
Peer-to-peer

The second relationship is **peer-to-peer (p2p)**, where the two autonomous systems decide to announce to each other the networks which each AS can reach without using any transit connection or any other p2p relationship. One of the main reasons behind these relationships is to **keep traffic as local as possible via public or private facilities**, thus avoiding extra delays introduced by the transit connection which potentially can route the traffic via another country.



This type of relationship is typically settlement-free and established on IXPs and known as public peering. However, this is not always the case. Depending on the agreement made by the two autonomous systems, p2p relationships can also involve a fee (paid peering) and/or can be established in private facilities (private peering). According to the **PCH survey**, only very few p2p relationships are paid or private peering, and most of them are not formalized in any written document. Moreover, during the last year most autonomous systems exploit route-server on IXPs to establish p2p relationships among them (**multi-lateral peering**).

Route servers are basically the eBGP alter-egos of iBGP route reflectors. Usually they are **software solutions** that IXPs offer to their participants to easily interconnect to each other within a single BGP session. They run on the peering LAN of the IXP and accept BGP sessions only from BGP speakers located in the very same peering LAN of the IXP.



Once a network is announced to the route server, that route is propagated to every AS connected to the route server without modifying the NEXT_HOP attribute as in every other regular BGP session. This way, the receiver will see the original NEXT_HOP attribute announced and will be able to route its packets directly via the NEXT_HOP indicated and present on the peering LAN of the IXP. Each AS connected to the route server also has the possibility to control how announced networks are re-advertised towards other peers using **operational BGP communities**.

In any case, autonomous systems involved in any p2p relationship will act as each other's customers. In other words, each AS will announce to the other peer only its own routes and the routes received from its customers to avoid becoming a transit for the other peer.

These economic relationships were firstly introduced by Lixin Gao in 2001 [8], and still stand today. Whenever one AS violates the above relationships with malformed or even missing filters, **route leaks occur**, which are defined as "the propagation of routing announcements beyond their intended scope." One of the last leaks, which caused a ripple effect across the Internet, **was extensively described in one of our blogs**.

How BGP Routing Really Works

The Internet is always in constant evolution. Nowadays there are more than **4 billion users connected to the Internet**, browsing around **2 billion websites**, playing games, watching videos, and doing business with each other no matter where in the world they are. Users can reach their desired content via Internet routes provided by the interconnections of around 90k Autonomous Systems exchanging reachability information via BGP on about 900k different IPv4 networks and about 150k IPv6 networks. And these numbers are growing every passing minute.

The main role of BGP is "**[...] to exchange network reachability information with other BGP systems.**" Routes are announced and withdrawn constantly from various parts of the world. Whenever a new AS joins the routing game, the first thing it will do is get routes from its provider(s) to reach every Internet destination and announce to its neighbors that there is a new route towards the network(s) it owns. Each neighbor will then inform its own neighbors about these new routes, and so on so forth.

On the other hand, any AS shutting down causes the withdrawal of its routes to spread all over the Internet. But there are only a few causes of route announcement/withdrawal. A few other examples of route change include whenever a fiber cable is accidentally cut, when two autonomous systems sign a new economic agreement (or whenever that expires), and if there's any kind of network failure.

Route Announcements and Replacement

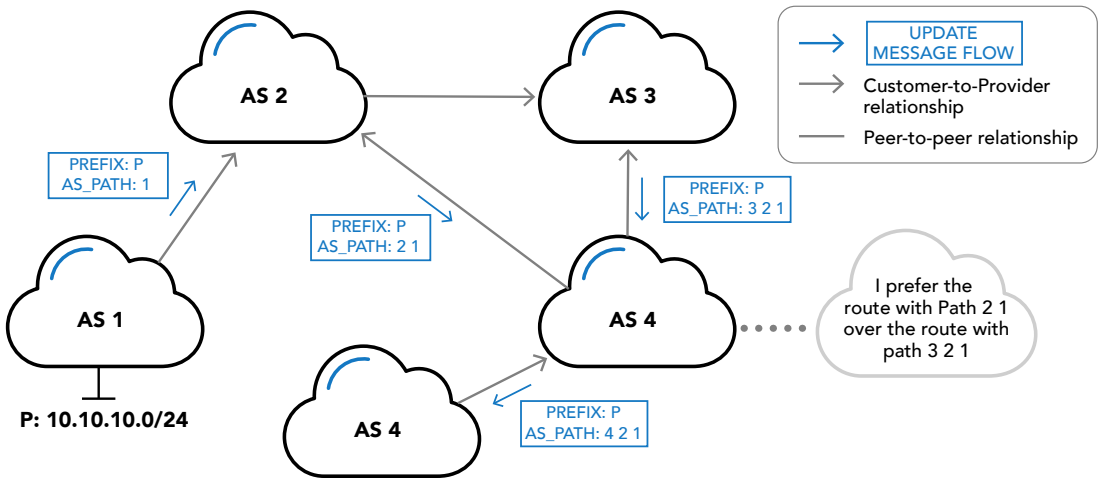
Once an organization gets an AS number and IPv4/IPv6 subnet from one of the five Regional Internet Registries (see the map below for the geographic distribution of RIRs) or one of the Local Internet Registries (LIRs), that organization is ready to announce its network reachability towards the global Internet.

First, that organization needs to settle an agreement with one or more providers to be connected to the Internet. Then it must start advertising to the Internet the subnets obtained from the RIR/LIR so that every other player in the Internet is aware of the presence of this new network resource and forward traffic accordingly.

To better understand how the Internet learns about the presence of the newly advertised subnets, see the figure below. For ease of understanding this and the following examples, we will assume that a customer will announce only its subnets to its provider, a provider will announce everything to its customers, and a peer will announce to other peers its subnets and all the subnets it received from its customers. We will also assume that the BGP decision process will always choose the route with the shortest AS path as the best.

Source: **RIPE NCC**

AS 1 is a brand new organization which managed to get the subnet 10.10.10.0/24 from one of the available LIRs. It contacted AS 2 and signed an agreement so that AS 2 can transit traffic for AS 1 and allow AS 1 to be connected to the Internet. As soon as the network administrator of AS 1 installs the subnet 10.10.10.0/24 in its router, it will generate an UPDATE message for AS 2 carrying the information that a new subnet has been announced.



More importantly, this message will carry the information that this new subnet will be reached crossing only AS 1, meaning that AS1 is the origin AS of the subnet. This information is carried in the AS path, which is one of the mandatory attributes in Update messages. This attribute is manipulated by each border router crossed to keep track of the path followed by the Update message and **avoid routing loops**. This is part of the BGP best route selection process.

Once AS 2 receives the Update message carrying 10.10.10.0/24, it will install this new route in its Adj-RIB-In, select it as the best route for 10.10.10.0/24, and then install the proper Adj-RIB-Out according to AS 2 outbound-filter policy. Thus, it will announce an Update message towards its peer, AS 3, and its other customer, AS 4, prepending its own AS number in the AS path.

At this point, AS 2 will be aware of the presence of this new subnet and will start to route traffic towards it whenever required. The same procedure will be followed by AS 3, which will propagate the Update message towards its customer AS 4, prepending its own AS number.

AS 4 will then receive two different Update messages to reach 10.10.10.0/24 at two different times. If the Update message coming from AS 3 will be received before the Update message of AS 2, then AS 5 will first receive an Update message with AS path 4 3 2 1, then another Update message with AS path 4 2 1. Otherwise, AS 5 will only receive one single Update message with AS path 4 2 1, since the piece of routing information carried by the packet announced by AS 3 will not be considered by AS 4 as the best route.

Let's now assume that the BGP session between AS 2 and AS 4 is torn down. In this case, both autonomous systems can no longer reach each other and the content of the related Adj-RIB-In tables will be invalidated. Consequently, the two autonomous systems will re-run the decision process for all the best routes which involved the other AS. In this example, this means that AS 4 will analyze every other Adj-RIB-In to find a route feasible to reach 10.10.10.0/24. This process is called Path exploration and can potentially involve many routes, **affecting the performance of the router**. Once a feasible replacement is found, AS 4 will inform its customer that the path has changed sending an Update message with AS path 4 3 2 1.

Route Withdrawals

Let's now assume that the organization decides to shut down its operations. In this case, the router will be shut down and most probably sold to the highest bidder. Once the router is shut down, the BGP session established will be torn down and will trigger a domino effect across the Internet to let everybody know that the subnets owned by the organization are no longer available to receive traffic.

Consider once again the example shown in the figure above, with all the BGP sessions up and running. Once the BGP session between AS 1 and AS 2 is shut down, then AS 2 will start its Path exploration phase, finding no feasible routes to reach 10.10.10.0/24. Then it will generate a special Update message announcing that it cannot reach subnet 10.10.10.0/24, thus telling its neighbor to stop propagating traffic towards AS 2 to reach AS 1. AS 3 will receive this piece of information and will behave like AS 2, announcing to AS 4 that 10.10.10.0/24 is no longer reachable via AS 3.

As before, AS 4 will receive two different Update messages in time. If the Update message coming from AS 3 is received before the Update message coming from AS 2, then the first message will remove the route from the Adj-RIB-In related to AS 3. While the second message will trigger a Path exploration phase on AS 4, which will find no feasible routes to reach 10.10.10.0/24 and will propagate the Update message to AS 5, which will be the last in line to know that the subnet has been withdrawn from the Internet.

On the other hand, if the Update message from AS 2 is received first, then AS 4 will run a Path exploration phase which will let AS 4 believe there is still an available route towards 10.10.10.0/24, and will advertise this new reachability to AS 5 via an Update message with AS path 4 3 2 1. In this case, only the reception of the message from AS 3 carrying the withdrawal of 10.10.10.0/24 will trigger another Path exploration phase on AS 4 and let AS 4 (and AS 5, consequently) finally understand that the route is no longer there.

Please note, however, that an Update message advertising the withdrawal of a subnet does not necessarily mean that the destination is no longer reachable from any AS composing the Internet; for example, such a message could be generated in a specific geographic area due to a temporary local network failure and/or due to BGP session misconfigurations, while the subnet is still reachable from other autonomous systems.

Path exploration is a natural consequence of path vector protocols like BGP. In this family of protocols, the path information is always updated dynamically so that updates looping through the network can be discarded easily. On the other hand, the path dependencies created tend to prolong BGP protocol convergence, which can be reduced by **applying special timers on the border routers**.

Regional Internet Registries:

- **RIPE NCC**: Europe, Middle East, Russia, and parts of central Asia
- **ARIN**: United States, Canada, and some Caribbean countries
- **APNIC**: Remaining parts of Asia and Oceania
- **LACNIC**: South America, Mexico, and the remaining Caribbean countries
- **AFRINIC**: Africa

Vulnerabilities of BGP

BGP protocol has allowed network operators to apply and enforce the most varied inter-AS routing policies over the past 30 years. It is amazing how this protocol efficiently sustained the ever-increasing number of subnets and autonomous systems, as well as the evolution of the Internet from a mostly hierarchical structure made of customers and providers to a structure where peering and IXPs become more important every day.

Despite all its good qualities, BGP shows several vulnerabilities which, if exploited, can cause far reaching ripple effects. The root of the problem is that BGP was conceived in an early development stage of the Internet when there were only a few players. Consequently, its **design didn't consider protection against deliberate or accidental errors**, so malicious or misconfigured sources can potentially propagate fake routing information, exploiting this lack of protection. Even worse, the source of fake or malicious routing information could be either a real BGP peer or a fake peer, since BGP runs on TCP/IP and is consequently subject to every classic TCP/IP attack such as IP spoofing.

Part of the problem can be solved by applying cryptographic authentication on each BGP peer, but this won't help stop bogus information spreading on the Internet from legitimate misconfigured sources (route leaks), from legitimate sources which either didn't apply cryptographic authentication at all, or from sources that deliberately announced bogus routing information (prefix hijacks).

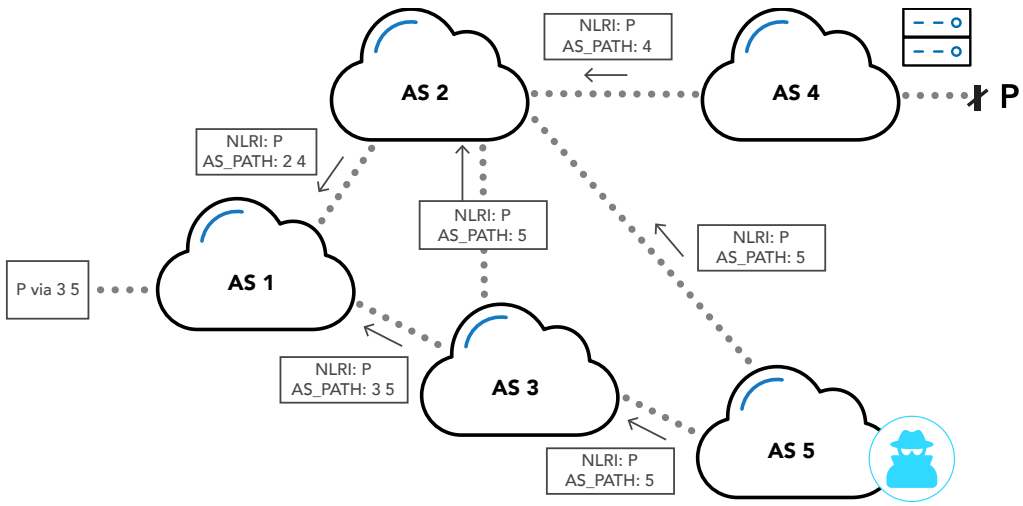
Solutions like **Resource Public Key Infrastructure (RPKI)** and **BGPsec path validation** have been recently standardized by IETF, but they still require the collaboration of many autonomous systems and thus are **difficult to deploy**.

Prefix Hijack Attacks

Prefix hijacks are deliberate intentional generators of bogus routing information; the reasons behind them are of a multitude that is difficult to fathom.

The attacker could announce routes to disrupt the services running on top of the IP space covered by the routes, or hijack the traffic to analyze confidential information flowing towards that service. The attacker could also simply announce routes with a crafted AS path to show fake neighboring connections in famous websites, like the **BGP toolkit of Hurricane Electric**. Or even worse, the attacker could hijack the traffic to manipulate the flowing packets at his/her will, or simply want to exploit unused routes to generate spam.

Scenario one



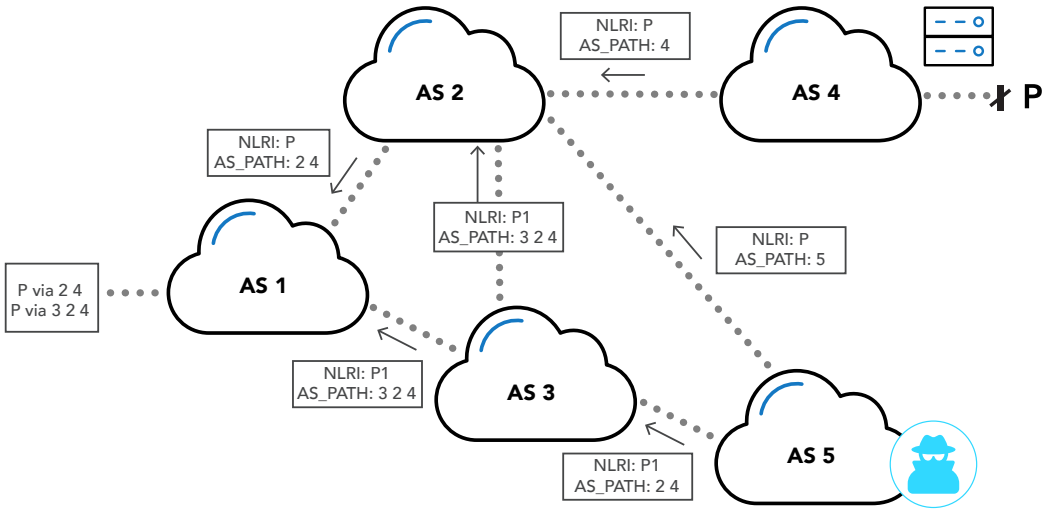
Let's consider the above scenario to better understand how prefix hijacks can be performed; we will consider the following topology in this and in the following examples. AS 5 is a malicious attacker and is connected to the Internet via two providers: AS 2 and AS 3. AS 1 is customer of AS 2 and provider of AS 3, while AS 4 is a peer of AS 2 and AS 2 is a provider of AS 3. Finally, we will assume that AS 2 has properly set its incoming BGP filters, while AS 1 and AS 3 have a loose filter configuration (if any).

In this first scenario, AS 5 will announce network P, which is owned and already announced by AS 4. Due to the filter configurations described above, the Update message announced by AS 5 will be dropped by AS 2, while it will be accepted by AS 3. AS 3 will then announce that to its providers (AS 1 and AS 2). AS 2 will again drop the packet due to the filters, while AS 1 will accept it. If the BGP decision process of AS 1 selects the path from AS 5 as the best route, then traffic from AS 1 to AS 5 will be sent to the attacker instead of towards the proper owner.

Now imagine this example to be comprised of around 90k autonomous systems, each with its own filter policy, if any. The impact would be that part of the Internet would redirect its traffic towards the attacker, while the rest would redirect its traffic towards the proper origin. The amount of autonomous systems redirecting their traffic towards the attacker will depend on two factors: the **quality of the filters applied by the providers**, and the BGP decision process output of each AS.

Note that in this scenario it is possible to identify the attacker by checking BGP packets involving P either from route collectors (with a proper post-mortem analysis), via dedicated real-time BGP monitoring systems, or via customer complaints since traffic is not re-directed to the original owner.

Scenario two



Let's now consider another scenario on the very same topology. AS 5 will now announce the network P1 subnet of network P, still owned by AS 4 but never advertised by AS 4. For example, consider P to be 10.0.0.0/23, then P1 could either be 10.0.0.0/24 or 10.0.1.0/24. AS 5 will only announce it to AS 3, knowing that AS 3 filters are loose. In addition, AS 5 will know that AS 2's filters are tight and will exploit that to ensure a safe route towards the destination.

In this scenario, P1 will propagate the same way as P in the previous scenario. The only slight difference is that now every affected AS has two different routes for the IP space covered by P: P and P1. Let's focus first on AS 1.

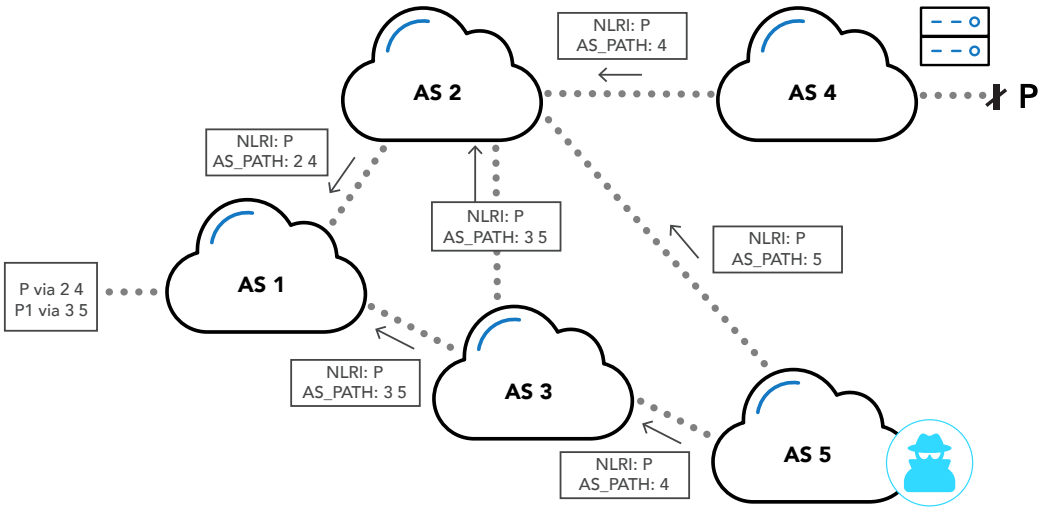
Even if a proper route to P is installed in AS 1's router, only a portion of traffic of the original P will be directed to the proper owner due to the longest prefix match. Please note that since AS 5 kept one of its providers explicitly out of the hijack, AS 5 can now route traffic received from AS 1 directed to P1 to the proper owner, after analyzing and/or manipulating each packet.

Now consider again this real-world example and imagine that AS 4 is hosting some bank servers on P1. Consider now that the attacker is interested in collecting data from the bank, and that he/she studied the problem deeply enough to know that P1 is the ideal target for its purposes and starts announcing it. In contrast to the previous scenario, the bogus routing information spread will now depend only on the quality of the filters applied by autonomous systems, since the subnet P1 and P will not interfere with each other in BGP decision processes. As soon as everything is set up, then AS 5 will be able to receive data from the affected portion of the world, while keeping a safe routing leg to forward traffic and (hopefully for him/her) go unnoticed.

Again, note that in this scenario, it is still possible to identify the attacker by checking BGP packets involving P and any subnet of P either from route collectors or via dedicated real-time BGP monitoring systems. However, the network operator can't identify the attack from the complaints received by his/her customers if the delay introduced by the attacker is short enough to go unnoticed.

An example of a route leak which falls perfectly in this scenario is the **infamous hijack of YouTube prefixes by Pakistan Telecom back in late February 2008**. In that case, Pakistan Telecom attempted to blackhole traffic towards 208.65.153.0/24 by announcing routes where Pakistan Telecom was appearing as the origin AS to fulfill a censorship request from the Pakistan government. The problem is that they also announced this route to its provider PCCW, which didn't apply proper filters. This led to a domino effect, causing about three hours of service disruption to YouTube.

Scenario three

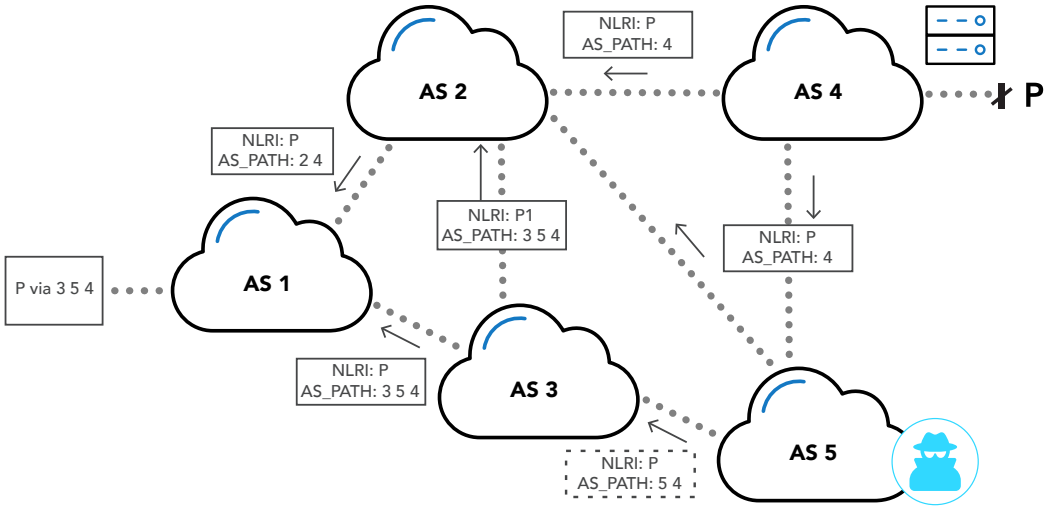


Consider now the scenario above. AS 5 is now smart enough to forge a fake AS path in the Update message by keeping the AS of the real owner at the end of the AS path as well as the original provider of the real owner (AS 2).

The propagation of the attack is the same as in the previous examples, but now the detection of the attack is much harder. It is still possible to check BGP packets involving P and any subnet of P either from route collectors or via dedicated real-time BGP monitoring systems, but now the detection of the attack must also rely on additional pieces of information, such as the knowledge of each relationship between each pair of autonomous systems in the AS path. Indeed, in this example it would have been possible to detect that since AS 3 is customer of AS 2, and the AS path 3 2 4 detected at AS 1 would have shown the involvement of AS 3 as transit of AS 2 for P1, which is against the valley-free property.

Route Leaks and Fat Finger Syndrome

Route leaks are the result of unintentional generating of bogus routing information caused by router misconfigurations, such as typos in the filter configuration or mis-origination of someone's else network (fat finger). Even if unintentional, the consequences of a route leak can be the same as a prefix hijack.



Consider the very same topology we used in the prefix hijack examples, with the difference that AS 5 is now a normal network operator which simply applied wrong BGP filters, such as "accept everything from my provider, announce everything to my provider." This is sadly not uncommon, and it is an error that several autonomous systems can make when switching from a single provider (where this rule works) to multiple providers (where this rule would make the AS a transit of each provider).

Due to that mistake, now AS 5 will propagate everything it received from its provider towards another provider, clearly against the valley-free property. This piece of routing information will then spread all over the Internet and autonomous systems will start routing traffic depending on the result of the BGP decision process of each AS.

Now think again about the estimated 90,000 autonomous systems in the Internet and imagine that AS 4 is a rural service provider with few resources, both technical and economic. This would probably mean that the upstream connection he/she bought from a provider is very limited, making the two links a bottleneck in this route leak scenario. In this case, it is possible that AS 5 will not be able to handle the amount of traffic directed to P, causing not only an additional delay, but also packet losses.

This was the case of **the route leak we discussed on our blog**, which affected several banks in addition to Facebook and CloudFlare. Returning to our theme of the impact of BGP problems, we can see that the consequences when things go wrong (whether intentionally or unintentionally) can be significant. The Facebook outage of 2021 may not been caused by BGP, but as we examine in **our blog on the outage**, when a routine maintenance job went wrong, it led to a safety mechanism being triggered in which the BGP routes towards their DNS servers were withdrawn from the network. As a result, over $100M was lost in revenue in just one day. Having visibility into BGP is crucial to troubleshoot quickly when things do go awry.

Next Steps

Hopefully this White Paper has helped you improve your understanding of what BGP is, how it ticks and what can go wrong. But what's the best way to apply this knowledge? While it's never a bad thing to better understand network architecture, what can BGP do for you? The answer all depends on how much you plan to monitor it.

BGP monitoring has become a required part of ensuring a network is available, reachable and performant. Because of its ubiquity, being able to see the current global status of BGP routing in your environment can provide an incredibly powerful way to identify and diagnose network issues. While not every network issue is caused by BGP, a comprehensive and real-time BGP dashboard can make it simple to either eliminate BGP as the culprit or pinpoint the problem.

Another important reason to monitor BGP is the fact that BGP issues can represent a serious security threat. Remember that BGP was built based on trust between AS's in the early 90s and security was not even a consideration when the two-napkins protocol was created. That quickly changed and security in BGP has been a hot topic in the research world for the first two decades of the 2000s. However, little effort has been made to push improvements out to the industry. Only recently has the industry showed interest in "fixing" security with some on-going efforts (e.g., RPKI) and new standards for the future (e.g., BGPSEC).

Therefore, it is crucially important that anyone relying on the Internet for their business has the means to understand how they can be affected by BGP (in)security issues — and react immediately. This makes a comprehensive BGP monitoring solution like Catchpoint absolutely essential for modern networks.

BGP Events to Look For

But what should a monitoring solution be looking for? Not every BGP event represents a problem, but you should be on the lookout for all of the following in order to understand exactly what is going on with your BGP network at all times.

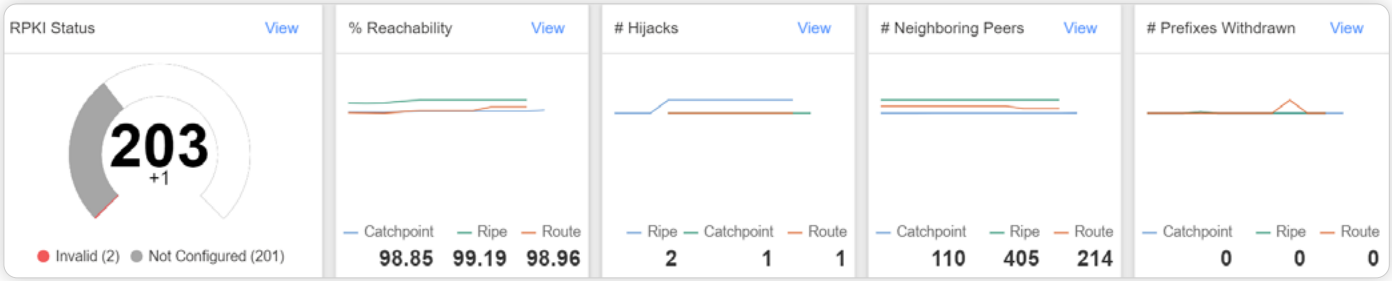
Prefix Hijacks

Prefix hijacks are the most (in)famous attacks performed on the Internet. They consist of the attacker announcing networks belonging to third parties as their own, aiming at either redirecting traffic to the attacker to disclose important data flowing between source and destination, or simply to cause a disservice to the original destination.

Hijacks can be roughly categorized into two main families: sub-prefix hijacks and origin hijacks.

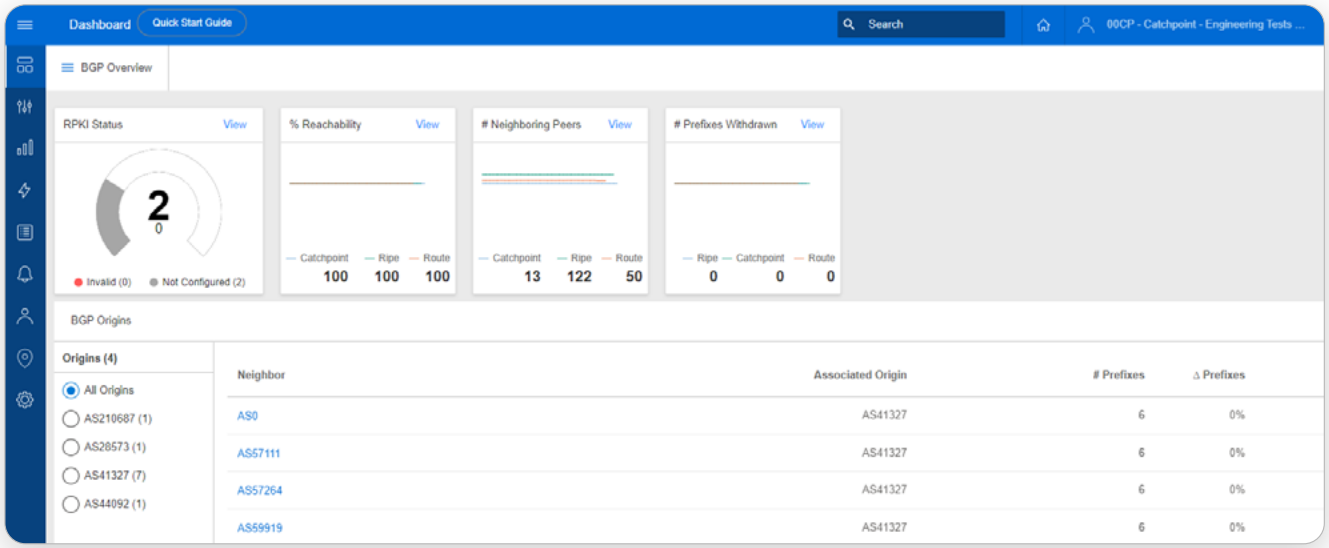
In the sub-prefix hijack case, the attacker exploits the concept of longest match — which is the common rule applied by routers to identify the most fitting route in the routing table to use to redirect traffic to a given IP address. This consists of preferring the network subnet with the longest netmask that includes the target IP address. Practically, the attacker announces a sub-prefix of the target network aiming at re-directing the traffic towards a portion of the target network and — unless proper ingress filtering mechanisms were applied by AS's — it spreads all over the world creating the hijack.

In the origin hijack case, the attacker announces the same network with a different AS number as origin. Again, if AS's have not applied proper filtering mechanisms, the attack will spread all over the world and split the reachability of the network between the attacker and the original AS, depending on the outcome of the BGP decision process of each border router of each AS.



RPKI Invalid

The Internet community during recent years has heavily promoted the Resource Public Key Infrastructure (RPKI), and more and more network operators are complying with it. RPKI consists of a cryptographic signature of entries — called Routing Origin Authorizations (ROAs) — representing networks and the AS's that are allowed to originate these networks. It's an efficient way to stop prefix hijacks described above, allowing routers to discriminate between valid and invalid routes and take action on the invalid routes.

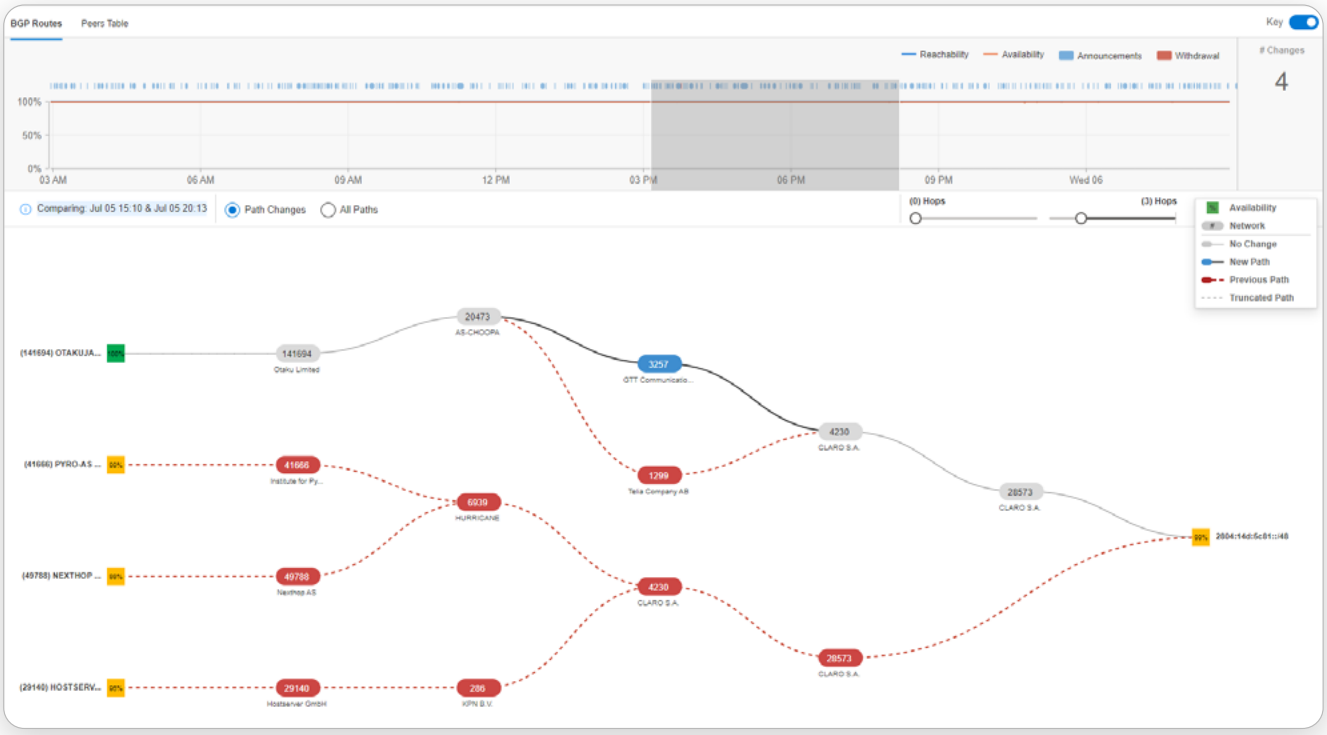


Mass Withdrawals

Being unreachable from customers is the nightmare of every company running their business on the Internet. Depending on the organization's infrastructure, internal monitoring may fail to reveal this. Often, monitoring may indicate that all services appear to be functioning properly from an internal vantage point. Reachability can also be caused by external failures, such as a provider link failure and poor redundancies in inter-domain connectivity. The result is that the world cannot find any way to get to the organization's network.

Mass Route Changes and Route Flaps

Counterintuitively, BGP routing activity is the opposite of the human heartbeat. When everything's fine, you will see a flat line – meaning that your network reachability is stable, and no one is seeing relevant changes in how to reach your network. However, when the line starts fluctuating it often hides a problem somewhere on the Internet that is somehow affecting the reachability of your network — possibly introducing network performance issues. A particular example is route flaps which happen whenever a route is frequently changing reachability over time.



Route Changes Confirmation

It's not unusual for network operators to perform maintenance on portions of their networks or try to apply traffic engineering techniques to improve the reachability of their networks. However, that's the extent of what a network operator can do. The Internet is asymmetrical by definition and even though BGP provides network operators with some ways to push their peers into making particular routing decisions (e.g., via MED or some special communities), the network operator will never understand how their routing changes were perceived by the rest of the Internet.

BGP Monitoring with Catchpoint

While there are several tools on the market that can monitor BGP, Catchpoint has unrivaled coverage and unique BGP functionality, allowing you to detect problems at a glance. With Catchpoint's powerful BGP Smartboards you can take advantage of the world's largest independent observability network to view global BGP status and pinpoint all routing issues. Real-time detecting and alerting ensures that you'll be able to keep your network secure and resolve BGP issues fast. And most importantly, Catchpoint's proactive BGP Monitoring means you'll be able to resolve BGP problems before they impact your users.

Learn More about Catchpoint's BGP Monitoring Solution

Catchpoint is the world leader in Network Experience Observability and has unrivaled coverage and unique BGP functionality, allowing you to detect problems at a glance. **Our BGP solution** lets you slice and dice your route data, drill down deep by filtering on key criteria and quickly determine root cause. But don't take our word for it, look at what **our customers are saying**. Then **give us a try**.

> **"We were never able to monitor reachability in this detail before using the Catchpoint solution. Now we can figure out whether an issue is related to a BGP announcement, routing, or something else instead of blindly trusting our partners that everything is working as it should."**
>
> Andreas Lunz, Head of Technology, Team Internet

Further Information

BGP Monitoring with Catchpoint – Video

Incident Review for the Facebook Outage – Blog

Incident Review: An Account of the Telia Outage – Blog

Incident Review: What was Behind the September '21 Spectrum Outage – Blog

BGP Monitoring – Educational Guide

Catchpoint can help you identify and resolve your BGP environment too — just let us know when you'd like to give us a try.

www.catchpoint.com/trial