Analysing BGP Origin Hijacks

S. J. M. van Veen

TU Delft

Analysing BGP Origin Hijacks

by

S. J. M. van Veen

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Wednesday December 18, 2019 at 10:00 AM.

Student number:	4605993	
Project duration:	December 1, 2018 – December 18, 2019	
Thesis committee:	Dr. C. Doerr,	TU Delft, supervisor
	Dr. S. Picek,	TU Delft
	Dr. P. K. Murukannaiah,	TU Delft

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

TU Delft

Abstract

In the past years, society has become increasingly more reliant on the Internet. Consequently, the security of the Internet became of critical importance. This thesis focusses on the security of one of the Internet's main protocols. This protocol, called the Border Gateway Protocol (BGP), is used to exchange information that allows Internet traffic to reach its intended destination. BGP is vulnerable to misconfigurations and attacks that can cause a range of problems. This thesis focusses on one of them: BGP origin hijacks. In this thesis, a year of possible origin hijacks is analysed. These possible origin hijacks were detected by BGPStream [3] between 20 May 2018 and 31 May 2019. Analysing these hijacks gives insight into the causes and characteristics of origin hijacks. This can help to find the most pressing issues and may provide guidance in securing BGP. Various data sources are used to collect and compute features that give more information on each hijack. These features are used to find relations between hijacks and to label them using labels that indicate a cause or a certain aspect of the hijack. These relations and labels are used to analyse groups of similar hijacks. This approach is very effective. Using the context of a group of hijacks gives much more insight than looking at hijacks individually. It shows that many of the possible hijacks are likely not a hijack at all and that hijacks that look like origin hijacks are often the result of another type of attack called a path hijack. In addition, this thesis provides a way to detect several types of misconfigurations and points out weaknesses in the detection system used by BGPStream. It also gives an overview of the characteristics of hijacks and how often specific behaviour occurs.

Preface

I would like to thank my supervisor Dr. C. Doerr for his guidance and advice. He made me strive for quality and his directions helped me to improve my work. I also want to thank the other members of my thesis committee, Dr. S. Picek and Dr. P. K. Murukannaiah, for taking the time to read through my thesis.

I want to thank Christian Veenman and Anant Semwal for their motivation and support. The weekly meetings always helped me to find direction and were very enjoyable. I also want to thank Cas, Tim, and Mark for their continuous encouragement through the process of researching and writing this thesis.

S. J. M. van Veen
Delft, November 2019

Contents

List of Figures

List of Tables

1 Introduction

In the past years, society has become increasingly more reliant on the Internet. Not only does it provide a wide range of entertainment, it is also more and more used for necessities such as health care, banking, and communication. Consequently, the security of the Internet became of critical importance. This thesis focusses on the security of one of the Internet's main protocols. This protocol, called the Border Gateway Protocol (BGP), is used to exchange information that allows Internet traffic to reach its intended destination. When BGP was designed, this information was expected to be reliable and security was not an issue. For this reason, the protocol is based on trust and does not contain measures that prevent mistakenly or intentionally sending incorrect information. This has become a problem because BGP is vulnerable to misconfigurations and attacks that can cause a range of problems. A recent example of such a problem comes from June 2019. This event prevented people in the Netherlands to pay with a debit card. The problem was caused when large amounts of Internet traffic were redirected through China Telecom's network. [44] [50]

BGP is vulnerable to several types of attacks. This thesis focusses on one of them: BGP origin hijacks. An origin hijack can be used to have Internet traffic sent to the wrong destination. This can have several consequences. Someone with malicious intent can perform an origin hijack so that traffic will be sent to them instead of to the intended destination. In this way, traffic can be monitored or intercepted. Another possibility is to misuse resources to, for example, send spam. When traffic never reaches its destination it can have serious consequences. In the example given above, where people were prevented to pay, the problem was solved fairly quickly, but it still caused a lot of inconvenience. Securing BGP against origin hijacks and other attacks has been the focus of much research in the past years. In general, this research concerns either the security of the protocol itself or the detection of attacks and other anomalies. Research on the events that are happening is less common. In this thesis, a year of possible origin hijacks is analysed. These possible origin hijacks were detected by BGPStream [3] between 20 May 2018 and 31 May 2019. Analysing these 2665 hijacks may give some insight into the causes and characteristics of origin hijacks. The goal is to find out if it is possible to detect the cause of origin hijacks and to find what is needed to prevent them. Additionally, the analysis can give an overview of the most common causes and provide insight in any patterns that occur over the year. Lastly, it helps to differentiate between intentional hijacks and misconfigurations, and to determine if all detected hijacks are indeed hijacks. This can all help to find the most pressing issues and may provide guidance in securing BGP.

The possible origin hijacks that were detected by BGPStream between 20 May 2018 and 31 May 2019 form the basis of our data set. Various data sources are used to collect and compute features that give more information on each hijack. These features are used to find relations between hijacks and to label them using labels that indicate a cause or a certain aspect of the hijack. These relations and labels are used to analyse groups of similar hijacks. The results are divided into three sections. The first section gives an overview of the hijacks and discusses basic characteristics of the data set. This section is based on the computed features. The second section discusses groups of related hijacks and the third section discusses groups of hijacks with the same label. Analysing groups of hijacks is very effective.

Using the context of a group gives much more insight than looking at hijacks individually. It shows that many of the possible hijacks are likely not a hijack at all and that hijacks that look like origin hijacks are often the result of another type of attack called a path hijack. In addition, this thesis provides a way to detect several types of misconfigurations and points out weaknesses in the detection system used by BGPStream. It also gives an overview of the characteristics of hijacks and how often specific behaviour occurs.

The remaining chapters of this thesis are organised as follows. Chapter 2 provides the background knowledge on the structure of the Internet, BGP, and BGP anomalies that is necessary to understand the rest of the thesis. Chapter 3 gives an overview of the research that has been done on the security of BGP, a summary of the research on misconfigurations, and a summary of research on intentional hijacks. This chapter points out the research gap that this thesis helps to fill. Following is chapter 4, which gives an overview of the data sources used to find all necessary information on the possible origin hijacks detected by BGPStream. Chapter 5 then explains the process of feature engineering, finding relationships between hijacks, labelling hijacks, and analysing hijacks. Chapter 6 provides an overview of the characteristics of the hijacks in the data set, discusses the hijacks based on the relations to other hijacks, and analyses how the hijacks were labelled. Finally, chapter 7 gives a conclusion.

2

Background

This thesis describes the process of analysing a set of possible BGP origin hijacks. In order to understand the methods and interpret the results, it is important to have sufficient knowledge of BGP and the structure of the Internet. This chapter will provide the background knowledge that is necessary to understand the remaining chapters of this thesis.

This chapter starts with section 2.1 that explains the fundamentals of the Internet. This explanation includes IP addresses, autonomous systems, and Internet resource management. Section 2.2 explains the relevant parts of BGP. This section includes BGP messages, the process of exchanging routing information, and routing policies. Section 2.3 is the last section of this chapter. It explains BGP anomalies and includes origin hijacks, path hijacks, route leaks with re-origination, and the Resource Public Key Infrastructure.

2.1. Internet fundamentals

The Internet is a global decentralised network that consists of many smaller interconnected computer networks. The Internet Protocol Suite, which is also known as TCP/IP, is used for communication between and within these networks. The Transmission Control Protocol (TCP) and the Internet Protocol (IP) are the two foundational protocols of this suite, but not the only ones it uses. TCP provides a reliable connection between two host applications. A full specification of this protocol can be found in RFC 793 [19].

The Internet Protocol (IP) is responsible for addressing and encapsulating packets that have to be delivered from a source host to a destination host. This protocol provides the addresses known as IP addresses and is essentially the backbone of the Internet. Internet Protocol version 4 (IPv4) is the version that was used for the first generation of the Internet and was introduced in 1981 in RFC 791 [18]. Together with its successor, IPv6 [34], it is still in use today.

2.1.1. IP addresses

IP defines the format of the packets that have to be delivered and provides an addressing system. These addresses, known as IP addresses, are used to label the packets with information about the source and destination hosts. All publicly accessible network hardware, such as home routers and servers, can be reached by a globally unique address. With IPv4, the addresses started as 32-bit numbers. Over the years this proved to be insufficient and thus IPv6 introduced 128-bit number addresses.

An IPv4 address is often provided as four octets expressed as decimal numbers that are separated by dots. The address 192.0.2.0 is an example of an IPv4 address. Blocks of consecutive addresses are allocated to organisations such as Internet Service Providers (ISPs). To refer to a block of IP addresses instead of a list of individual addresses, CIDR notation is used. CIDR is an abbreviation of Classless Inter-Domain Routing [36]. It is a method that improves IP address allocation by allowing addresses to be grouped into blocks of arbitrary length. This is accomplished by taking an IP address

as prefix and adding a suffix to indicate the length of the block. For example, the IP range 192.0.2.0 to 192.0.2.255 would be written as 192.0.2.0/24. The /24 indicates that the first 24 bits are equal and the remaining 8 bits are variable. This notation is often referred to as an IP prefix.

The prefix 192.0.2.0/24 can be further divided. The prefixes 192.0.2.0/25 (192.0.2.0-192.0.2.127) and 192.0.2.128/25 (192.0.2.128-192.0.2.255) together cover the same range. They are called subnets. The /24 prefix is the supernet of the /25 subnets. Using CIDR notation, any range of prefixes can be defined. This includes single IP addresses. A single address has /32 as the suffix. The action of expressing a range of IP addresses as one prefix is called aggregation.

IPv6 is very similar to IPv4. The addresses are represented by eight groups of 16 bits that are written as four hexadecimal digits and are separated by a colon. An example IPv6 address is 2001:0db8:0001:0000:0000:0000:ff02:00a3. In this notation, leading zeros are often omitted and consecutive groups of zeros are replaced by ::. This would result in 2001:db8:1::ff02:a3. This means that the address represented by eight groups of zeros can also be written as ::. CIDR notation is used for IPv6 in the same way as for IPv4. The whole range of IPv6 addresses can be written as ::/0, while a single address is an address that has the suffix /128.

2.1.2. Autonomous systems

The Internet is a network of interconnected Autonomous Systems (ASes). The official definition of an Autonomous System (AS) is given in RFC 1930 [39] as “a connected group of one or more IP prefixes run by one or more network operators which has a SINGLE and CLEARLY DEFINED routing policy”. A routing policy is a set of rules that is used when ASes exchange routing information. This is necessary because an AS is not directly connected to all other ASes. Instead, they form a network where each AS can be reached by following a route through other ASes. Figure 2.1 contains an example of a network of ASes.

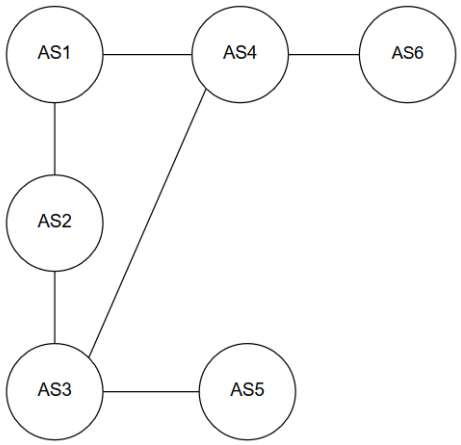


Figure 2.1: Example of AS network

If an AS wants one of its prefixes to be reachable, it sends an announcement to its neighbours. It is not mandatory for an AS to announce its prefixes. An AS is free to decide to only announce part of its prefixes or none at all. There are however some disadvantages to not announcing a prefix. Unannounced prefixes can be used by others for malicious purposes. This is more thoroughly discussed in section 3.3.

Prefixes can be announced by any number of ASes. When a prefix is announced by only one AS this is called a Single Origin AS (SOAS). It is also possible that a prefix is announced by multiple ASes, this is called a Multiple Origin AS (MOAS) or MOAS conflict. There are legitimate reasons for a prefix to be announced by multiple ASes, but it can also be the result of a misconfiguration or malicious behaviour.

When an AS connects to the Internet it has to be uniquely identifiable. The AS is therefore assigned a unique number called the Autonomous System Number (ASN). This is a decimal number and an

AS is often referred by its number as, for example, AS65536. AS numbers started as 16-bit numbers, allowing for a maximum of 65536 assignments. As this was not enough, 32-bit ASNs were introduced to increase the range to 4294967295.

ASes exchange routing information that allows them to know which route traffic needs to take to reach the intended destination prefix. Consider the example network in figure 2.1. AS1 can reach AS5 in two different ways; either via AS2 and AS3, or via AS4 and AS3. ASes use a routing protocol to communicate information about these routes to each other. A routing policy is used to help choosing which route will be used. The exchange of routing information between the ASes that form the Internet is called inter-domain routing. The Border Gateway Protocol (BGP) is the default inter-domain routing protocol and is explained in the next section.

2.1.3. Internet resource management

To avoid overlapping use of IP prefixes and AS numbers there exist organisations that manage these resources. The Internet Assigned Numbers Authority (IANA) [7] is in charge of the global coordination of Internet resources. They allocate IP prefixes and AS numbers to Regional Internet Registries.

A Regional Internet Registry (RIR) is an organisation that manages the Internet resources for a part of the world. The world is divided into five regions, each having its own RIR:

- Africa - African Network Information Centre (AFRINIC) [21]
- Asia and the Pacific - Asia-Pacific Network Information Centre (APNIC) [22]
- North America - American Registry for Internet Numbers (ARIN) [23]
- Latin America and Caribbean - Latin America and Caribbean Network Information Centre (LACNIC) [10]
- Europe, Central Asia, Russia and West Asia - Réseaux IP Européens Network Coordination Centre (RIPE NCC) [11]

A RIR further allocates resources to National Internet Registries (NIRs) and Local Internet Registries (LIRs) which in turn can allocate them to their customers. NIRs function on a national level and are mostly located in Asia and the Pacific. Other regions work more directly with LIRs, which often are Internet service providers or academic institutions.

When RIRs are allocated a block of IP addresses, this is generally a large block that can be further divided by the RIR. For example, a RIR receives a /8 prefix from IANA. The RIR can divide this prefix into subnets of size /16 and allocate these to a LIR. In turn, the LIR can allocate subnets of the /16 to its customers. The result of this is that an AS may own a /16 prefix, but a customer will own a /24 subnet of that prefix. Traffic with a destination IP address in the /24 range therefore has to be routed to the customer and not to the AS that owns the /16. For this reason, traffic is always routed using the route that matches the most specific prefix.

Not all IP prefixes and AS numbers will be allocated by IANA. Some are reserved for special purposes. In the case of IP addresses these are, for example, the ranges that are reserved for private networks. These addresses can be used within a network for computers not connected to the Internet. This way, not every existing device needs a unique IP address, which saves address space. As a consequence, these reserved addresses may not be routed on the Internet. Similarly, there are also ranges of ASNs that are reserved for special purposes and should not be announced to the Internet. These special purposes include private use and the use for documentation and sample code. Aside from IANA, RIRs may also decide to reserve resources. Both IANA and RIRs keep publicly available lists of the status of IP prefixes and AS numbers. In addition to the status of Internet resources, RIRs also keep WHOIS servers that can be used to find information about the owner of a resource.

Some prefixes should not be announced at all. When a reserved or unallocated prefix is announced to the Internet this is referred to as a bogon route. Bogon routes can be caused by misconfigurations as

well as malicious intentions. To avoid such routes there are lists of bogons that are used to implement bogon filters. Unfortunately, filtering is not always applied properly and these routes still appear on the Internet.

2.2. Border Gateway Protocol

The Border Gateway Protocol (BGP) is the dominant inter-domain routing protocol. It is used by ASes to exchange information on how to reach destination prefixes. Each AS is responsible for the routing to and from their prefixes. They set up a dedicated connection with other ASes and communicate network reachability information using BGP. This section will provide the knowledge on BGP that is necessary to understand the rest of this thesis. A full specification of the protocol is available in RFC 4271 [48].

2.2.1. BGP messages

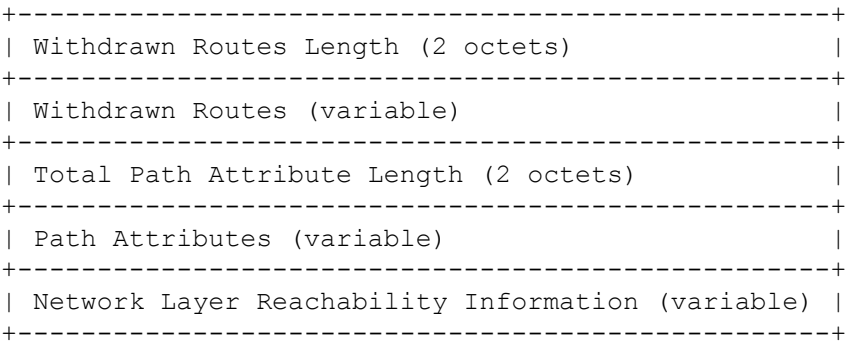
BGP is used so that ASes can exchange routing information that allows traffic to be forwarded to the desired destination. Because an AS is not directly connected to all other ASes, it needs to know which route traffic has to take to reach its destination. When two ASes want to exchange routing information they both set up a router that speaks BGP. After they established a TCP connection between them, they are called BGP peers.

BGP peers communicate using BGP messages. In total there are four types of messages: OPEN, KEEPALIVE, NOTIFICATION, and UPDATE. OPEN messages are the first messages that are sent after a TCP connection is established. These messages allow ASes to identify each other and to agree on several parameters.

One of these parameters is the Hold Time. This is the maximum length of time in seconds that an AS will wait to hear something from their peer before assuming that the connection is down. KEEPALIVE messages are sent between ASes at a rate that keeps the connection from closing. BGP requires that either a KEEPALIVE or an UPDATE message is sent between peers before the hold time expires. The rate at which these messages are sent is often based on the hold time.

If an error occurs during a BGP session a NOTIFICATION message is sent to the other AS. This message contains an error code to indicate the type of error that occurred, and an error subcode to further specify the error. After a NOTIFICATION message is sent, the connection will be closed immediately.

The fourth message type, UPDATE, is the most interesting one. This type of message is used by the sender to announce and withdraw routes, and by the receiver to learn new routes. It comes in the following format:



The first two fields, Withdrawn Routes Length and Withdrawn Routes, are used when an AS no longer wants to forward packets to a specific IP prefix or when it does not want one of its own prefixes to be reachable any more. It can then send an UPDATE message to withdraw the route. Multiple routes can be withdrawn with one message. The Withdrawn Routes field holds a list of IP prefixes for which the sender wishes to withdraw the route. Withdrawing and announcing routes can be done in the same message, but it is also possible to send messages only to announce or withdraw routes.

The remaining fields are used to announce and learn routes. This process is explained in the next subsection. The Total Path Attribute Length indicates the length of the Path Attributes field. The Path Attributes contain information about the routed prefixes and are used to select the best paths. The Network Layer Reachability Information contains a list of prefixes to which the Path Attributes apply. It is thus possible to announce multiple prefixes in the same message, but only if they share all attributes. Using one message for multiple prefixes is encouraged because it saves the resources necessary for processing a single message for each prefix.

2.2.2. Exchanging routing information

To exchange routing information ASes send each other BGP UPDATE messages. These messages contain path attributes that hold information on how to reach one or more prefixes. It is not necessary to discuss each path attribute. A description of all attributes can be found in RFC 4271. One attribute that is important for this thesis is AS_PATH.

The AS_PATH attribute contains the path of ASes through which a prefix announcement has passed. Each time an AS sends an UPDATE message it will prepend its ASN to the path. It is best explained using an example. Consider the network of ASes in figure 2.2. AS1 wants to announce a prefix and sends an UPDATE message to its peers, AS2 and AS3. AS1 is the owner of the prefix and is called the origin. AS2 and AS3 receive the message with '1' as the AS path. After learning this route, both AS2 and AS3 will prepend their ASN to the path and send a message to their peers. This process is repeated and ends with AS5 receiving messages from AS3 and AS4.

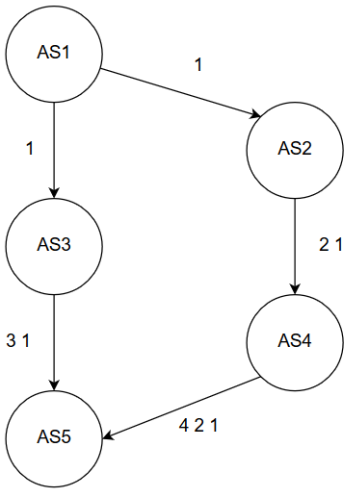


Figure 2.2: Example of AS path

The picture shows that AS5 will receive two paths towards the prefix announced by AS1. From AS3 it receives the path '3 1' and from AS4 it receives the path '4 2 1'. AS5 will not use both paths but will choose one of them. This decision is guided by the route selection process.

If an AS receives multiple routes to a prefix, it has to select the one it wants to use. The route selection process uses several rules to make a decision. The first rule looks at preference values. An AS can influence the decision process by setting these values. Preferences are, for example, a preference for one of its neighbours, or for paths that contain a certain ASN. Taking all preference values into account, the path with the highest preference is chosen. If multiple paths share the same preference, the selection process moves to the next rule.

When a decision cannot be made based on preferences, the shortest AS path is chosen. In the case of the given example, AS5 would choose the path from AS3 because it has a length of 2, while the path from AS4 has a length of 3. If one path can be selected in this way, the decision process ends. Otherwise it will move to other rules. These other rules are not important for this thesis. It is, however,

important to remember that shorter paths are preferred over longer paths.

When an AS sends a path to a neighbour and wants to influence whether or not the path is chosen, the AS may use path prepending. Path prepending means that instead of prepending its ASN once to the AS_PATH attribute, the AS will prepend its ASN multiple times. This way it can influence the decision process of the next AS. Assume that AS3 in the example network would prepend the path thrice. The resulting AS path that is sent to AS5 would be 3 3 3 1. As this path is longer than the path received from AS4, AS5 will choose the path from AS4 instead of the path from AS3.

2.2.3. Routing policies

Decisions on preference values and whether or not to prepend a path are made based on routing policies. These routing policies reflect the business relations between ASes. This allows ASes to select and reject routes based on their own preferences. It also allows an AS to select which routes it wants to send to its peers. A routing policy is based on the relationships an AS has with its peers. In general there are three types of relationships: customer-to-provider (c2p), peer-to-peer (p2p), and sibling-to-sibling (s2s). An AS can thus take on four different roles. It is either a customer, a provider, a peer, or a sibling to another AS.

A customer will pay a provider for transit, meaning the customer pays the provider to have its traffic routed to and from the parts of the Internet it otherwise could not reach. The relation between a customer and a provider is called a customer-to-provider (c2p) link, or a provider-to-customer (p2c) link when looking from the other direction. The provider will send all routes it has learned to the customer and sends routes it learns from the customer to its neighbours. In this way, the customer has access to all the ASes the provider can reach.

In a peer-to-peer (p2p) relationship the ASes do not have to pay. Peers provide transit for each other and each other's customers. This means that an AS only sends customer routes and routes for its own prefixes to a peer. In addition, it sends the routes learned from its peer to its customers. A p2p relation saves money because part of the traffic that an AS would normally have to pay a provider for can now be routed by a peer.

The third type of relation is sibling-to-sibling (s2s). When an organisation owns more than one AS it may set up a s2s relation between them. ASes in a s2s relation provide transit for each other freely. They exchange all routes they learned with each other or decide to set up a more complex policy. It should be noted that the data sources used for this project often omit this type of relation. The reason for this is that AS relationships are not public and have to be inferred, and s2s relations are difficult to distinguish. However, they do not often occur so it is not a large issue. [43]

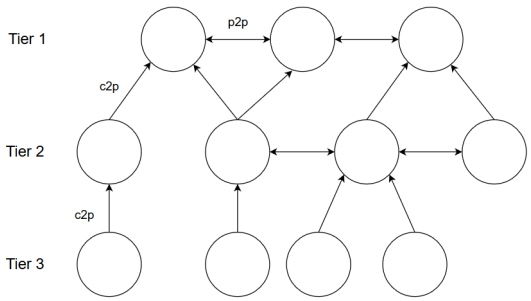


Figure 2.3: AS hierarchy

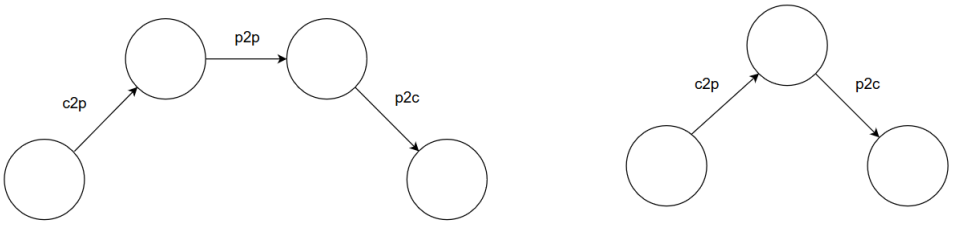
Knowing the AS relationships, the hierarchy of ASes can be explained. The network of ASes that forms the Internet can be seen as a hierarchy with three tiers. Tier 1 ASes are the ASes on top of the hierarchy. They can reach any network connected to the Internet without having to pay transit. They are often owned by large telecom companies and tend to peer with other Tier 1 ASes. Tier 2 consists of the ASes that peer for free with some ASes but also still purchase transit from others. Lastly, Tier 3 consists of ASes that pay others to have access to the Internet. Figure 2.3 contains an example of

what this hierarchy may look like.

2.2.4. Valley-free paths

Another concept that is connected to AS relationships is the valley-free path. A valley-free path is a path that respects the fundamental idea of p2p, c2p, and p2c links by following a specific pattern so that no AS is providing transit against its policy. This property is called the valley-free property. In theory, if every AS respects the routing policies of itself and its neighbours, AS paths should be valley-free.

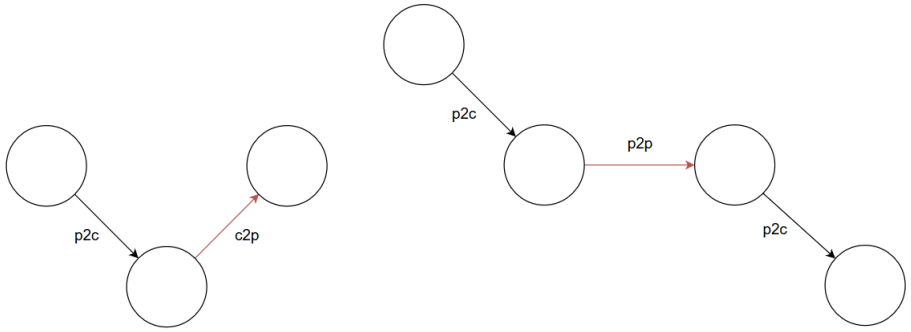
A valley-free path adheres to one of the following formulas: `n*c2p + p2p + m*p2c` or `n*c2p + m*p2c` where `n` and `m` are integers larger than or equal to 0 that indicate the amount of ASes. Figure 2.4 provides two examples of valley-free paths. Both can be extended by multiple c2p relations on the left side and multiple p2c relations on the right side without changing the valley-free property.



(a) c2p + p2p + p2c (b) c2p + p2c

Figure 2.4: Valley-free paths

Two examples of paths that are not valley-free can be found in figure 2.5. Paths that are not valley-free cause ASes to provide transit they are not paid for. This principle violates the concept of the AS relationships. Figure 2.5a shows a path where the customer provides transit for the provider by announcing the routes to its other provider. The customer should only have announced these routes to its customers. In figure 2.5b a customer sends routes from its provider to a peer. Because peers only provide transit for each other's customers this violates the relationship.



(a) p2c + c2p (b) p2c + p2p + p2c

Figure 2.5: Not valley-free

In 2012 Giotsas and Zhou [37] presented their research on valley-free violations. They show that the valley-free principle is not always adhered to. This means that some ASes use more complex routing policies that cannot easily be inferred.

2.2.5. The transition AS

Since 2007 all AS numbers are represented in 32 bits. This includes the 16-bit AS numbers. They are padded with 16 zeros to turn them into 32-bit numbers. Because not all BGP routers can correctly handle AS paths containing 32-bit ASNs, a transition AS was introduced. This transition AS, AS23456 or AS_TRANS, is used by incompatible BGP routers to handle 32-bit AS numbers.

When a BGP router sends a BGP message to a router that cannot handle 32-bit numbers, the numbers are replaced by 16-bit numbers. All numbers that can be represented by 16 bits are written in 16 bits. The 32-bit numbers are replaced by 23456. The AS4-PATH attribute holds the original path. It is an optional transit attribute so it can be ignored by old routers, but it is transited to neighbours so that the information is not lost. A router that can handle the 32-bit numbers will convert everything back to 32 bits and replace the 23456 by the original numbers.

2.3. BGP anomalies

When BGP was developed the data provided by an AS was expected to be correct. For this reason, the protocol is based on trust and there are no security mechanisms implemented to prevent mistakes or protect against intentionally sending incorrect information. Currently, BGP is available widely and the lack of security has become a problem. Anomalies are common and have a variety of causes.

Any BGP update that undermines the routing policy of an AS can be considered anomalous. Anomalies can be caused by intentional behaviour or misconfigurations. This can have a variety of undesired consequences such as outages and traffic interception. There are several reasons to intentionally cause instabilities. Reasons include economic incentives and misusing resources to, for example, send spam. [46]

2.3.1. Origin hijacks

Because BGP has no security guarantees it is vulnerable to attacks. This thesis focusses on one type of attack: BGP origin hijacks. The last section explained that traffic is routed using AS paths that are shared between ASes. In theory, only the AS that owns a prefix should be announcing the prefix, but in practice it is perfectly possible for an AS to announce a prefix it does not own. As a consequence, an AS with malicious intents can cause traffic that is destined for another AS to be routed to itself instead.

When traffic is forwarded to its destination, a router will identify the path corresponding to the longest prefix matching the destination IP address. For example, if the destination address is 192.0.2.150 and there is a path for 192.0.2.0/24 and a path for 192.0.2.128/25, the router will choose the latter. This can be exploited when hijacking. The hijacker can announce the prefix that is announced by its victim, but it can also announce a more specific prefix.

Consider the example network in figure 2.6. Assume that AS1 is the original owner of a prefix. AS4 will now announce the exact same prefix. AS5 and AS6 learn the route from AS4 and will use this route. AS2 learns the route from AS1 and will use that route. AS3 however, will learn routes to both AS1 and AS4. Because it receives two routes for the same prefix, it will use the path selection process and ends up choosing the route to AS1. This is called a local hijack. The hijack is local because it only affects ASes that choose the route to the hijacker based on the path selection process.

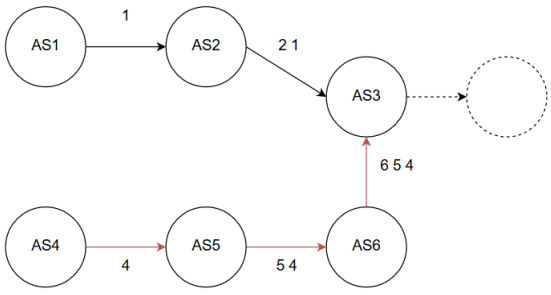


Figure 2.6: Example network: global and local hijacks

Now assume that AS1 still announces the same prefix, but AS4 will announce a more specific prefix (or multiple prefixes that together cover the same range as the original prefix). AS3 will now receive a route to AS1 for the original prefix, and a route to AS4 for a more specific prefix. All traffic destined to an IP address that is part of the range announced by AS4 will be forwarded to AS4. In addition,

because AS3 uses this path it will also announce it to its peers. This is called a global hijack because it affects BGP globally.

2.3.2. Path hijacks

Origin hijacks are not the only type of hijacks that affect BGP. It is also possible to hijack the AS path. When an AS modifies the AS_PATH attribute in a BGP UPDATE message in such a way that it appears to have a (shorter) path to a prefix, this is called a path hijack. An AS can either send a fake path or modify an existing path. In this way, the origin of the path does not have to change, but the hijacker will still have the traffic routed to itself. An example of a path hijack is given in figure 2.7. AS4 announces a false path for 192.0.2.0/24 from AS1 to AS5. Since AS4 and AS1 have no connection, AS4 cannot forward traffic to AS1. However, because it is the shortest path that AS5 receives, AS5 will choose this path.

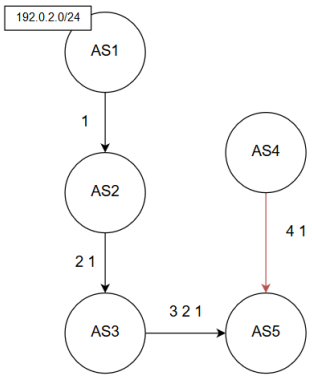


Figure 2.7: Example of path hijack

It is also possible for a hijacking AS to send a fake path with a different origin and its ASN somewhere in the middle. This would appear to be an origin hijack, but the fake origin AS is not the cause of the hijack. An example is given in figure 2.8. AS4 announces the path ‘4 6’ for 192.0.2.0/24 to AS5. Since AS6 is not the owner of this prefix, this seems to be an origin hijack, but AS6 has nothing to do with this announcement as it is not connected to AS4. This type of path hijack may appear in the data set used for this thesis.

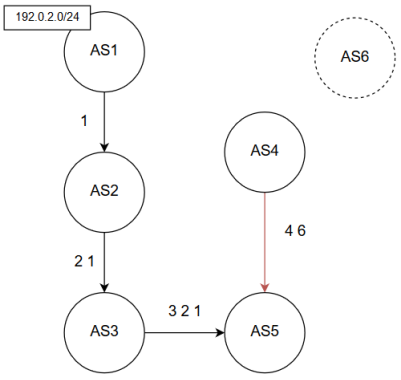


Figure 2.8: Example of path hijack

2.3.3. Route leaks with re-origination

Another anomaly that may look like an origin hijack is a route leak with re-origination. When an AS exports routes to a neighbour in such a way that it violates policies, it is called a route leak. An example can be found in figure 2.9. A provider announces routes to its customer, but the customer propagates these routes to a second provider. The second provider learns these routes from the customer, so it will send traffic to these destinations via the customer. This means the customer is now providing

transit for its provider, which is against its policies.

When an AS leaks routes in such a way that it seems to be the origin of the routes, this is called a route leak with re-origination. Re-origination happens when a provider AS strips the AS path to the customer AS in such a way that it seems as if the provider is the destination. The traffic is then routed to the provider, and the provider will forward it to its customer. Because the prefix then has a route with an invalid origin, it may show up in the data set used for this thesis.

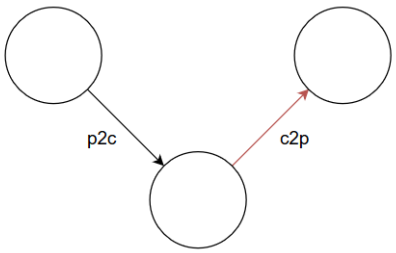


Figure 2.9: Route leak

2.3.4. Resource Public Key Infrastructure

There have been many attempts to secure BGP. One of the architectures that improves the security is the Resource Public Key Infrastructure (RPKI). The full specification can be found in RFC 6480 [42]. RPKI provides digitally signed objects that can be used to identify the legitimate owner of a prefix. If every AS would verify the origin of a route it receives, BGP paths with a false origin would not be propagated any more. Unfortunately, it is not widely used yet.

The objects provided by RPKI, called Route Origin Authorisations (ROAs), state which AS is authorised to originate a certain prefix. Because a prefix can be owned by multiple ASes, it is possible to have multiple ROAs for one prefix. A ROA is connected to a trust anchor that is managed by one of the RIRs. This is a self-signed root certificate for all resources that IANA allocated to that RIR. It is used to validate ROAs.

Each ROA contains a prefix, the AS that legitimately owns the prefix, and the maximum prefix length. The maximum prefix length can be equal to or different from the prefix length. If the prefix would be 192.0.2.0/24 and the maximum prefix length is 24, only the prefix 192.0.2.0/24 may be announced by the AS. If the maximum prefix length would be 25, the AS is also allowed to announce 192.0.2.0/25 and 192.0.2.128/25.

When an AS receives a route, it can use ROAs to validate the origin of the route. In theory, this would solve the problem of origin hijacks. However, the implementation of RPKI comes with some concerns. These concerns are discussed in [46] and include the power given to centralised authorities and the fact that human errors can cause invalid ROAs. RPKI is thus not a perfect solution to origin hijacks. It is not widely deployed yet because it is difficult to convince each AS to spend resources on implementing a solution that is not yet fully functional, and there will always be malicious ASes that are opposed to any security measures.

3

Related Work

BGP version 4 (BGP-4) was first published in 1994 and was specified in RFC 1654 [47]. Although there have been many updates since, BGP still does not provide any security guarantees. The most recent specification of the protocol can be found in RFC 4271 [48].

Because BGP lacks proper security measures it is vulnerable to attacks and misconfigurations that can lead to various problems causing instabilities and outages. Much research has been done on the security of BGP. This chapter will first give a brief overview of the research done in the past years and of the current state of research on the security of BGP. It will then provide a summary of research on misconfigurations and a summary of research on intentional hijacks. Finally, this chapter points out a research gap and explains in which way this thesis aids in filling this gap.

3.1. A overview of research on BGP security

In 2010 Butler et al. published a survey of BGP security issues and solutions [32]. At that time the focus of research on BGP could be roughly divided into two categories. The first category focussed on operational concerns such as scalability, stability, and performance, while the second category was formed by research on the validity, confidentiality, and integrity of BGP messages. [32]

Most solutions that were implemented to protect BGP focussed on having local implementation and a limited need for interaction with external parties. The majority of these solutions were a form of protecting the underlying TCP connection, filtering BGP announcements, or cryptographic protection between routers. Protection against complex and sophisticated attacks on the protocol itself was still very limited. [32]

To improve this protection, several suggestions were given for future research. These suggestions make up four categories. The first category suggests that routing frameworks and policies could be improved to achieve better deployment and scalability, and to build resistance into BGP routing. Concentrating on protecting the most connected nodes could significantly enhance security. The second category refers to the then already active field of research on attack detection. The third category recommends research on protecting BGP's data plane to ensure that packets are actually forwarded along the announced paths. Without this protection, ASes may announce one path and forward traffic over another path. Lastly, in the fourth category research on partial or incremental deployment of security solutions is suggested. As it is difficult to get each AS to implement the same solution, it is beneficial to have a solution that only has to be implemented by a small group of ASes in order to enhance security. [32]

Eight years later a similar survey is published by Mitseva et al. [46]. This survey of attacks and defenses provides an overview of countermeasures against attacks on BGP in addition to a survey of methods to detect, locate, and mitigate routing instabilities. Furthermore, they evaluate the different properties of existing proposals to secure BGP.

By comparing these two surveys one can get some insight into the persistent issues of securing BGP. The survey from 2018 comes with a list of desired properties for BGP security solutions. This list is based on previous research and considers security, privacy, performance, and deployability. [46] Recalling the four categories mentioned in the survey from 2010, one can gain some insight into which issues are solved and which issues still require further research.

The development of security solutions touches three of the four categories; improving routing frameworks and policies, securing the data plane, and partial or incremental deployability. Mitseva et al. compare a large number of state-of-the-art security solutions and discuss their features and flaws. They conclude that the proposed solutions only partly solve the problems, and often at the cost of high overhead. Most solutions still focus on securing the control plane. The control plane is the part of BGP that focusses on sharing and updating routes between ASes. There is also the data plane, which focusses on forwarding traffic based on the information learned in the control plane. Security of the data plane has received little attention in the past years and is again given as a suggestion for future research. It is also noted that full deployment cannot be expected while there is no new protocol that solves at least a large part of the security issues. Expecting an AS to invest money in deploying a new solution that only solves a small part of the issues may not be reasonable. [46] Considering these points, one can conclude that improving routing frameworks and policies is difficult and requires a lot more work.

The fourth category that was mentioned by Butler et al. is attack detection. As long as there are no security guarantees in BGP, it is necessary to be able to detect anomalous events so that any problems can be solved as quickly as possible. An extensive overview of BGP anomaly detection techniques was provided by Al-Musawi et al. in 2017 [30]. These techniques were assessed based on their ability to identify anomalies in BGP and their ability to locate the AS that caused an anomaly. Al-Musawi et al. conclude that there remains much work to be done. None of the proposed solutions offers the combination of real-time detection, differentiation between anomalies, and identification of the source.

Mitseva et al. came to a similar conclusion after reviewing proposals for detection, localization, and mitigation of BGP anomalies. None of the solutions provide a complete detection-recovery system and they are either impractical or inaccurate and easily compromised. [46]

Summarizing the above, most of the research done in the past years has focussed on detecting anomalies and developing a secure version of BGP. However, there is hardly any research done that provides insight into the specifics of the BGP events that are happening. This type of research may help to determine the most common problems of BGP security. This can help to understand which areas need the most attention. The next sections of this chapter concentrate more on this type of research.

3.2. Router misconfigurations

As shown in the previous section, most research on BGP security concentrated on the detection or prevention of events such as hijacks. The goal of this thesis is to provide more insight into the behaviour of BGP hijacks.

Mahajan et al. pursued a similar goal in 2002. They analysed BGP traffic over a three week period to detect cases of misconfiguration. To verify these cases they asked the involved ISP operators what the cause of the event was and whether it was indeed a misconfiguration. Additionally, Mahajan et al. also determine the extent of disruption due to these misconfigurations. [45]

Their system can detect two types of faults: origin misconfigurations and export misconfigurations. [45] Export misconfigurations cause the accidental export of routes in violation of the ISP policy. These misconfigurations are also known as route leaks. Origin misconfigurations cause an AS to accidentally inject prefixes into the global BGP tables. This is also known as hijacking and is the focus of this thesis. During the time of their research however, intentional malicious hijacks were not very common. [54]

Origin misconfigurations were identified as short-lived new routes. On average there were around 600 short-lived routes advertised each day. They received an email response for roughly 30% of the incidents. Most incidents were a case of self-deaggregation. This happens when an AS announces its prefix as a set of smaller prefixes. Other incidents were labeled either as being caused by a related origin (an AS related to the old origin) or as being caused by a foreign origin (unrelated to the old origin). The remaining 70% of incidents most likely followed the same distribution according to the tests of connectivity disruption. [45]

After compiling a list of causes of misconfiguration they conclude that not all misconfigurations are the result of human error and most could be prevented by better router design. [45] However, as this research took place 17 years ago, it is not unlikely that repeating their experiments may lead to different results today.

More research on misconfigurations is done in later years. In 2005 Feamster and Balakrishnan present the routing configuration checker: a tool that finds faults in router configurations using static analysis. They analysed configurations from 17 different ASes finding over 1000 faults that had gone unnoticed by the operators. [35]

Four years later Le at al. publish a paper on their approach to detect misconfigurations using data mining by applying association rules mining to configuration files. In this way they want to address the problem of other proposed solutions which are based on rules that need to be known beforehand. Their system, Minerals, was able to detect various errors that would otherwise be difficult to detect. These errors were confirmed and corrected by network operators. [41]

Although it is network operators' responsibility to properly configure their routers, it is often difficult to do so and systems like the two above can greatly help in preventing misconfigurations and the resulting BGP incidents.

3.3. Intentional hijacks

In this thesis, the behaviour and characteristics of BGP hijacks over a longer period are studied. As pointed out earlier, there is not much research done on this topic. Aside from research concentrating on misconfigurations, as discussed in the previous section, there are also some papers that focus specifically on intentional hijacks.

In 2014 the maliciousness of BGP hijacks was studied by Vervier et al. They developed a system that detects BGP hijacks and investigated whether these hijacks coincide with spam and web scam traffic. Through a case study, they show that this type of coinciding traffic is not enough to prove that a hijack is malicious. They conclude that in order to identify whether a hijack is malicious or not, feedback from network operators is necessary. [53] It should be noted that they detect hijacks by looking for prefixes that are originated by multiple ASes and thus cause a MOAS conflict. By doing so, they cannot detect hijacks of unannounced prefixes and therefore do not provide a complete picture of malicious BGP hijacks.

However, a year later a paper is published by Vervier, Thonnard, and Dacier. This paper concentrates on hijacks of unannounced prefixes. This type of hijacks can, for example, be routinely used to send spam. They analyse 18 months of data in order to investigate how much these hijacks occur. Over the course of this period they found more than 2000 malicious hijacks, although there is a lack of ground truth that is needed to validate the maliciousness of these hijacks. However, they did get confirmation on some hijacks from an unwittingly involved ISP. [54]

After validating the detected hijacks using several data sources they come with the following conclusions. Spammers often hijack IP prefixes that are never announced or have not been announced for a long time. By using these prefixes to send spam they can bypass known protection systems such as IP blacklists. They seem to prefer properly registered prefixes because they are not filtered out by bogon filters. [54]

It is not always the case that this type of hijacks is used for spam. It can also be used for other malicious activities. Because unannounced prefixes are vulnerable, Vervier et al. recommend to always announce prefixes, even if they are not used. [54]

3.4. Our contribution

In section 3.1 and 3.2 it became clear that it is very difficult to properly protect against intentional and unintentional hijacks. The previous section on intentional hijacks discussed two papers that concentrated on hijacks related to spam. But as explained in chapter 2, there are more reasons to hijack a prefix. There is however not much research done on this topic. By studying the characteristics of recent hijacks over a period of 12 months we hope to provide insight into what kind of issues are most prevalent. The goal is to find out if it is possible to detect the cause of origin hijacks and to find what is needed to prevent them. Additionally, we want to give an overview of the most common causes and provide insight in any patterns that occur over the year. Lastly, we want to find out if it is possible to differentiate between intentional hijacks and misconfigurations, and determine if all detected hijacks are indeed hijacks. This can help to find focus in securing BGP by either improving the protocol, securing routers, or providing tools for network operators that allow for easier configuration.

4

Data Sources

To develop an algorithm to analyse the hijacks detected by BGPStream a data set is created using various data sources. The accuracy of the algorithm is highly dependent on the data quality. It is therefore important to know the validity, strengths, and weaknesses of the data. This chapter contains a description and discussion for each used data source.

Section 4.1 will cover the hijacks detected by BGPStream. Section 4.2 explains the relevant parts of the RIPEstat data API. In section 4.3 the CAIDA AS Relationships data set is covered. The CAIDA AS Organizations data set is explained in section 4.4. Section 4.5 contains an explanation of the Regional Internet Registry statistics. Following are section 4.6 and section 4.7 that respectively describe the IP2Location Lite and GeoLite2 databases that are used to find the country in which an AS or prefix is based. Section 4.8 contains an explanation of the data used from the Internet Assigned Numbers Authority. Finally, section 4.9 completes this chapter by providing a description of the RPKI Validator.

4.1. BGPStream

BGPStream [3] is a free resource that monitors hijacks, route leaks and outages in BGP. These events are posted to their website and Twitter feed. BGPStream is part of BGPmon [2] and uses their monitors to extract the largest and most important events. [20] For this thesis only the events labelled as possible hijack (from now on referred to as hijacks) are considered. The hijacks used are those detected between 20 May 2018 and 31 May 2019. Each hijack comes with the following features:

- Event number
- Start time
- Expected prefix - the prefix that is normally announced
- Expected ASN - the AS that normally announces the expected prefix
- Expected organisation - the organisation of the expected AS
- Expected AS country - the country where the expected AS is based
- Detected advertisement - the hijacked prefix
- Detected origin ASN - the AS that originates the detected advertisement
- Detected organisation - the organisation of the detected origin AS
- Detected AS country - the country where the detected origin AS is based
- Detected AS Path - route for detected advertisement received by BGPMon peer
- Detected by number of BGPMon peers - number of BGPMon peers that receive a route to the detected advertisement with the detected origin ASN as the origin.

Along with these features comes the option to watch a replay of the event. The features listed form the basis of the data set that is created to analyse the hijacks.

It should be noted that the detected hijacks are possible hijacks. They may be hijacks but can also be normal behaviour. Unfortunately, there is no public information about the detection system used by BGPStream. However, earlier this year Veenman compared his proposed detection system for origin hijacks with BGPStream [52] and although the systems did not detect the exact same set of hijacks, BGPStream appears to be reliable enough for our research.

4.2. RIPEstat Data API

The RIPEstat Data API [14] is the public interface for RIPEstat. RIPEstat provides all kinds of data related to BGP. The sources used to collect this data are the RIPE database [13], Routing Information Service (RIS) [15], and RIPE Atlas [12]. RIS is used to store raw BGP data collected by RIPE's route collectors and can be accessed using RIPEstat. Because the information gathered from RIPEstat comes from raw BGP data, it is considered to be a reliable source.

For each hijack four data calls are made: the announced prefixes of the detected AS, the announced prefixes of the expected AS, blacklist data of the detected advertisement, and the routing history of the detected advertisement. The announced prefixes of an AS are all announced prefixes during a specified period. It comes in JSON format as a list of announced prefixes and their timelines. A timeline is a list of periods, specified by start time and end time, in which a prefix was announced by an AS. An example of the data is given in Figure 4.1. With this data, one can for example find out if the detected AS has announced the detected advertisement before. It is also used for other purposes like finding normally announced prefixes that relate to the detected advertisement and to find the duration of the hijack.

```
"data": {
    "prefixes": [
    {
        "prefix": "193.0.0.0/21",
        "timelines": [
         {
             "endtime": "2011-12-13T00:00:00",
             "starttime": "2011-12-12T12:00:00"
         },
         ...
         {
             "endtime": "2012-01-17T00:00:00",
             "starttime": "2012-01-16T08:00:00"
         }
         ]
    },
    {
         "prefix": "2001:67c:2e8::/48",
         "timelines": [
             ...
         ]
    }
    ],
    "query_endtime": "2012-01-18T13:21:51.305237",
    "query_starttime": "2011-12-12T12:00:00",
    "resource": "3333"
},
```

Figure 4.1: Example of announced prefixes in JSON format

The routing history of a prefix is similar. It lists the ASes that announced the prefix and the corresponding timelines. The routing history of the detected advertisement is used to check if there was a MOAS conflict at the time of the hijack, if the prefix was in use before the hijack, and to find the duration of the hijack.

RIPEstat also keeps blacklist data for IP prefixes. The sources for this information are UCEPROTECT [28] and SPAMHAUS [26]. The format of this data is similar to the AS history and routing history. For each source it contains the timelines in which the given prefix was blacklisted. If the prefix was never blacklisted, this list is empty. The blacklist data is used to find out if the detected advertisement is blacklisted after the hijack.

4.3. The CAIDA AS Relationships data set

The CAIDA AS Relationships data set [5] contains the inferred relationships between ASes. It consists of three files for each month: the 'serial-1' as-rel file, the 'serial-2' as-rel file, and the 'serial-1' ppdc-ases file. The as-rel files contain the peer to peer (p2p) and provider to customer (p2c) relationships for each AS. The difference between 'serial-1' and 'serial-2' is the method used to infer the relationships [38] [43]. The files use the following format:

```
<provider-as>|<customer-as>|-1
<peer-as>|<peer-as>|0
```

The 'serial-2' file also adds the source of the inference:

```
<provider-as>|<customer-as>|-1|<source>
<peer-as>|<peer-as>|0|<source>
```

The ppdc-ases files contain the provider-peer customer cones. The customer cone of an AS can be inferred using the relationships between ASes and contains all ASes that can be reached by following a customer-link. The format of the ppdc-ases files is as follows:

```
<cone-as> <customer-1-as> <customer-2-as> .. <customer-N-as>
```

Updated files are published each month. The files used to compute the attributes for a hijack are the files for the month in which the hijack occurs and the files for the next month. The reason for this is that relationships between ASes can change during the month. The actual relationship may thus not yet be present in the file at the beginning of the month. This is confirmed by comparing the two files for each hijack. Approximately a quarter is in some way affected by a change in relationships. More detailed information on this can be found in chapter 5.

The methods of inferring relationships that are used to create the CAIDA Relationships data set are explained in [38] and [43]. The serial-1 data set is created using the method explained by Luckie et al. [43]. They validated 34.6% of their data and obtained an accuracy of 98.7% for p2p links and 99.6% for p2c links. The serial-2 data combines this method with the method explained by Giotsas et al. in [38].

Due to various reasons it is difficult to obtain perfect accuracy when inferring relationships. [52] There may be some mistakes in the data but it is accurate enough to serve our purpose. This data set is used to find relations between ASes and to get information about the customer cones of the detected origin AS and the expected AS.

4.4. The CAIDA AS Organizations data set

Another data set from CAIDA is the CAIDA AS Organizations data set [4]. This data set maps autonomous systems to their organisations. A new file is published every three months: in January, April, July, and October. This file contains entries for ASes and organisations. For this project only the AS entries are used. They come in the following format:

```
aut|changed|aut_name|org_id|opaque_id|source
```

Here ‘aut’ is the AS number. The field ‘changed’ is the changed date provided by its WHOIS entry. The name for the AS number is given by ‘aut_name’, and ‘org_id’ gives the organisation entry. The final two elements are ‘opaque_id’, which is an identifier used by the RIR statistics, and ‘source’, which gives the RIR or NIR database that contained this entry.

This data set is used to map AS numbers to organisation IDs. BGPStream hijacks do have the organisation of the detected ASN and expected ASN as features, but because these are the organisation names they are less accurate. Organisation names tend to change spelling over time and between ASes. This is why the organisation ID is also used.

4.5. Regional Internet Registry statistics

All the Regional Internet Registries (RIRs) keep a list of currently allocated and assigned IPv4 prefixes, IPv6 prefixes, and AS numbers. Each RIR publishes an updated list every day. These lists are formatted according to the RIR statistics exchange format [17]. They contain a file header, file summary lines, and a line for each record. The file header includes information like the format version, registry, and number of records. For each type of record, there is a summary line that holds the type and the number of records of that type. The records follow the following format:

```
registry|cc|type|start|value|date|status[|extensions...]
```

Here, ‘registry’ is the RIR that generated the file. The ‘cc’ is the country code according to ISO 3166 [9]. The ‘type’ field represents the record type and is one of the following: ‘asn’, ‘ipv4’ or ‘ipv6’. In case of IPv4 or IPv6, the ‘start’ field gives the first address of the range. For ASN it is the first AS number. The ‘value’ gives the size of the block. In case of IPv4 it is the number of hosts in the range, for IPv6 it is the CIDR prefix length, and for ASN the count of AS numbers from ‘start’. The ‘date’ field gives the date of the allocation or assignment, and ‘status’ indicates whether it is an assignment or allocation, or if the resource is reserved or available. Lastly, the line can hold extra data in the ‘extension’ field.

When Internet resources are assigned or allocated to a RIR, the RIR will further distribute them to Local Internet Registries (LIRs) and National Internet Registries (NIRs). LIRs and NIRs can then distribute the resources to their customers. Note that the RIR statistics only contain the assignments and allocations made by the RIR, not the subsequent distribution by LIRs and NIRs. After a prefix or AS is assigned or allocated by a RIR it can change country, but the record in the RIR statistics never changes. Because of this reason the latest available file is used. It contains all assignments and allocations done at that point in time along with their date.

In our research, the RIR statistics are used to get the country code and status of an AS or prefix. This country code is combined with data from the IP2Location LITE [8] and GeoLite2 [6] databases (see section 4.6 and section 4.7). The BGPStream events also come with a country for each AS but it is not clear where this data comes from. As the registered country is not necessarily the country in which an AS operates, other datasets are used instead.

4.6. IP2Location LITE

IP2Location LITE [8] is a freely available collection of databases containing IP geolocation data. There are databases for IPv4 and IPv6, databases for anonymous proxies, and a database containing ASN information. DB1.LITE is the database containing IP to country information. Together with the IP to ASN database, ASN.LITE, it is used to find the country for an AS or prefix.

DB1.LITE can be downloaded as a CSV or binary file for IPv4 and IPv6 and comes in the following format:

```
ip_from|ip_to|country_code|country_name
```

The fields ‘ip_from’ and ‘ip_to’ give respectively the first and last IP address of a range as integers. The ‘country_code’ field holds the country code for that range and follows the ISO 3166 standard. The last field, ‘country_name’, gives the corresponding country name and also follows ISO 3166.

The ASN.LITE database comes with a CSV file for IPv4 and a separate CSV file for IPv6. These files come in the following format:

```
ip_from|ip_to|cidr|asn|as
```

Similar to the DB1.LITE database, 'ip_from' and 'ip_to' represent the first and last IP address of a range. The 'cidr' field gives this range in CIDR format. The 'asn' field gives the AS number of the AS holding the range, and the 'as' field gives the name of this AS.

The files are updated each month but unfortunately only the most recent files are available to download. IP2Location LITE database has an accuracy of at least 98% on country level. Because there are no historical files available, the files from May 2019 are used. This may lower the accuracy for hijacks that happened earlier. However, when comparing RIR statistics and IP2Location LITE files it seems that the country does not often change.

4.7. GeoLite2 databases

MaxMind's GeoLite2 databases [6] are free downloadable databases containing IP geolocation data. These databases map an IP address to a geographical location using longitude and latitude and an accuracy radius. GeoLite2 consists of 3 databases: GeoLite2 City, GeoLite2 Country, and GeoLite2 ASN. Each of these databases is available in binary and CSV format. The databases used are GeoLite2 Country and GeoLite2 ASN.

The country database comes with two types of files: Blocks files and Locations files. The Blocks files are a file for IPv4 and a file for IPv6. These files use the following format:

```
network|geoname_id|registered_country_geoname_id|
represented_country_geoname_id|is_anonymous_proxy|is_satellite_provider
```

In this file, the 'network' field represents an IP address range and is given in CIDR format. Following are three geoname IDs. These geoname IDs can be used to look up the location in the Locations files. The field 'geoname_id' gives the network's location. The 'registered_country_geoname_id' is the location in which the ISP has registered the network. The 'represented_country_geoname_id' gives the country that is represented by the users of the network. The 'is_anonymous_proxy' and 'is_satellite_provider' fields are deprecated and not used.

The Locations files can be used to look up the geoname IDs. They are available in several languages and use the following format:

```
geoname_id|locale_code|continent_code|continent_name|country_iso_code|
country_name|is_in_european_union
```

The 'geoname_id' field is a geoname ID that can be found in the Blocks files. The locale code is the language of the file. The 'continent_code' and 'continent_name' fields give the continent belonging to the location represented by the geoname ID. The 'country_iso_code' and 'country_name' give the country code and country name following the ISO 3166 standard. Lastly, the field 'is_in_european_union' holds 1 if the country is a member of the European Union, and 0 if this is not the case.

The ASN database maps networks to AS numbers. The format used is as follows:

```
network|autonomous_system_number|autonomous_system_organization
```

Here, the 'network' field is an IP range given in CIDR format. The 'autonomous_system_number' gives the AS number holding the prefix. The organization associated with this AS can be found in the field 'autonomous_system_organization'.

The GeoLite2 databases have similar issues as the IP2Location LITE databases. They are updated regularly, but historical files are not available. There is no accuracy given for the free GeoLite2 databases. By combining GeoLite2 data with data from IP2Location LITE and RIR statistics one can see if the country of an AS or prefix has changed over time and if there is a difference between IP2Location LITE and GeoLite2. More information on this is given in chapter 5.

4.8. The Internet Assigned Numbers Authority

The Internet Assigned Numbers Authority (IANA) [7] performs the global coordination of Internet protocol resources. They manage the IP addressing system as well as the AS numbers. IANA allocates AS numbers to the Regional Internet Registries. While doing so they keep a list [1] that shows which AS numbers are assigned, reserved, and unallocated. This list is updated regularly and contains the date at which a block of AS numbers is assigned. In our case, this list is used to detect hijacks that have an unallocated or reserved AS as source or victim.

IANA also keeps track of IP prefixes that are reserved for special purposes. Some examples are 127.0.0.0/8, which is reserved for loopback, and 10.0.0.0/8, which is reserved for private-use networks. This list of special prefixes is used to detect hijacks that have such a prefix, or subnet of such a prefix, as detected advertisement or expected prefix.

4.9. The RPKI Validator

The RPKI Validator [16] is provided by RIPE NCC. It contains all validated ROAs. As explained in chapter 2, each ROA contains a prefix, the AS that legitimately holds the prefix, the maximum prefix length, and the trust anchor. The RPKI Validator is used to check if there is a ROA for the detected advertisement. If that is the case, the ROA would show the legitimate originator of the prefix.

Currently RPKI is not widely deployed. Since it is a good way to protect against origin hijacks, this data is included to investigate whether there is an interesting pattern in the hijacks with respect to RPKI. Since only the legitimate holder of a prefix can create a ROA, it is fair to assume that the data is reliable.

5

Analysing Hijacks

The aim of this thesis is to study the behaviour and characteristics of hijacks over a longer period. As explained in chapter 4, the basis of our data set is formed by 2665 possible hijacks detected by BGPStream between 20 May 2018 and 31 May 2019. These hijacks come with the features listed in section 4.1. The process of going from this basic data set to the results presented in chapter 6 consists of the following steps:

1. Feature engineering
2. Finding relations between hijacks
3. Labelling hijacks
4. Generating results

Each of these steps is discussed in this chapter. Section 5.1 discusses each of the newly generated features. In section 5.2 is explained how relations between hijacks are detected. Following is section 5.3 that contains an explanation on how the hijacks are labelled. Lastly, section 5.4 describes which results are generated, and how this is done. The code for this project is available at [51].

5.1. Feature engineering

Before labelling and analysing the hijacks more features are needed to provide the necessary information. This section explains how each new feature is generated. The new features can be divided into three categories: AS features, prefix features, and event features. AS features are directly related to the detected origin ASN or expected ASN of a hijack. Prefix features are derived from the detected advertisement or expected prefix. The features that are generated using other information form the last category.

5.1.1. AS features

This subsection discusses the features derived from the detected origin ASN or expected ASN. Each paragraph explains one feature. Table 5.1 provides an overview of the features discussed.

Feature	Description	Type
Relation between detected origin AS and expected AS	Relation taken from CAIDA AS Relationship data set. Either ‘p2p’, ‘c2p’, ‘p2c’, or ‘no relation’.	String
AS customer cone size	Number of customers in the customer cone of an AS. Number is taken from CAIDA AS Relationship data set.	Integer
AS in customer cone of other AS	This feature is used to check if the detected AS is in the customer cone of the expected AS or vice versa.	Boolean

AS number status	Is the ASN unallocated, reserved, available, or assigned. Feature includes the registry and, if ASN is assigned, also the date of the assignment.	Tuple
AS country	AS country found in GeoLite2, RIR statistics, and IP2Location LITE	Tuple
AS organisation ID	Organisation ID taken from CAIDA AS Organizations data set	String
Closest prefix announced by AS	Prefix(es) normally announced by AS that have the smallest string distance to the detected advertisement. Tuple contains string distance and list of prefixes	Tuple
Difference in prefix length	Smallest difference in prefix length between the detected advertisement and the prefixes in the list given by the closest prefix feature	Integer

Table 5.1: List of AS features

Relation between detected origin AS and expected AS This feature is used to detect if there is a relation between the hijacker and the victim. This is informative because not all possible hijacks are necessarily malicious hijacks. For example, a customer may be assigned a new prefix. This will be a subnet of one of the prefixes of its provider. While this is a legitimate announcement, it may be detected as a possible hijack by BGPStream.

The relation between the detected origin AS and the expected AS is taken from the CAIDA AS Relationship data set. The data files used are the ones from the month in which the hijack occurs, and those from the month after the hijack. As the relationship may change before or after the hijack, the two outcomes are compared to ensure better accuracy. In 138 of the 2665 cases (5.18%), there is a difference between the two months. This difference is important when labelling a hijack and is further discussed in section 5.3.

The relation is given as a string and can take the following values: ‘p2p’ (peer to peer), ‘p2c’ (provider to customer), ‘c2p’ (customer to provider), or ‘no relation’. The first letter represents the detected origin AS and the last letter represents the expected AS. In other words, if the detected origin AS is a customer of the expected AS, the relation will be ‘c2p’. When there is no relation between the two ASes, the string takes the value ‘no relation’.

Size of customer cone The customer cone of an AS is the set of ASes that can be reached by following customer-links. The size of the customer cone is equal to the cardinality of this set. An AS without any customers will thus have an empty customer cone, while a Tier 1-provider will have a very large customer cone. This feature gives some insight into the hierarchical position of the AS in the global BGP topology. It is computed for both the detected origin AS and expected AS. It is not used to label hijacks but gives insight into the size of ASes that are hijacking or being hijacked.

The CAIDA AS relationship data set contains a file with inferred customer cones. The size of an AS customer cone can be acquired by looking up the AS number in this file and counting the number of ASes that are listed as part of the customer cone. Because the AS itself is also listed as part of its customer cone, 1 is subtracted from the total. This is done so that ASes without customers can be more easily identified. Considering that the size doesn’t change drastically during a month, and the fact that this feature is only used for basic statistics, the file of the CAIDA data set that is used is the one from the month in which the hijack occurs.

AS in customer cone of other AS The CAIDA customer cone data is not only used to get the size of the customer cone but also to find whether one AS is in the customer cone of the other. While there may be no direct relation between the detected origin AS and the expected AS, they may still be related via customer-links. As with the relation between the detected origin AS and the expected AS this may

change around the time of the detected hijack so the data files for both the month in which the hijack occurs and the month after are used.

The feature is represented by 4 boolean values. The first two state if the detected origin AS was in the customer cone of the expected AS during the month of the hijack and in the month after the hijack. The last two values depict whether the expected AS was in the customer cone of the detected AS, again, during the month of the hijack and the month after. In 132 (4.95%) cases there is a change between the two months with respect to whether the expected AS is in the customer cone of the detected AS. In 31 (1.16%) cases it changes whether the detected AS is in the customer cone of the expected AS.

This feature gives more information on if and how the two ASes are related and can also be used to find evidence that suggests the possibility of a route leak with re-origination.

AS number status As explained in chapter 2 every AS has a unique 16-bit or 32-bit number. IANA assigns these numbers to the RIRs which further assign them to LIRs and NIRs. Both IANA and RIRs keep track of the status of these numbers.

Every hijack has a detected origin ASN and an expected ASN. For both numbers, the status at the time of the hijack is first looked up in IANA data. A number can either be assigned, reserved, or unallocated. Assigned means that the number is assigned to a RIR for further use, reserved means the number will be used for a special purpose, and an unallocated number is not yet in use but may be assigned in the future.

If the number falls out of the range of valid AS numbers, it is labelled invalid. If the number is assigned, the next step is to look into the RIR statistics. There are four possible outcomes: allocated, assigned, available, or reserved. Available means that the AS number can still be assigned by the RIR. The terms allocated and assigned seem to be used interchangeably here. RIPE, APNIC, LACNIC, and AFRINIC only use allocated, reserved, and available as status for an AS. However, ARIN uses assigned, reserved, and available.

The feature is represented by a tuple with three elements. The first element is either 'IANA' or a specific RIR. The second element gives the status, and the third element, if applicable, a date. If the ASN is invalid, reserved by IANA, or unallocated, the tuple will be: `(IANA, <status>, -)`. Otherwise, the tuple will have the registry to which the ASN is assigned as the first element, the status found in that RIR's statistics as the second element, and the corresponding date as the third element: `(<registry>, <status>, <date>)`.

This feature is used to detect hijacks that should not happen because, for example, the detected origin ASN is unallocated. It is also used to find the distribution of detected origin ASNs and expected ASNs over RIRs.

AS country When an AS number is allocated by a RIR it will be registered with a country code. This country code stands for the country in which an AS is first registered and is taken from the RIR statistics. After an AS is allocated it is not obliged to operate in the country in which it is registered. Each of its prefixes may be located in a different country.

The GeoLite2 databases and IP2Location LITE databases that are used to get AS country information both use a format where ASNs are linked to IP prefixes, and IP prefixes are linked to country codes. As was explained in section 4.7, in the case of the GeoLite2 databases there are three country codes per IP prefix, but IP2Location LITE and the RIR statistics have one country code per prefix. When collecting all IP prefixes and corresponding country codes of an AS, the result is most likely a list containing different country codes in various quantities.

The AS country feature is used to get insight into the global picture of hijacks. For example, which countries contain most hijacking or hijacked ASes. For this reason, it is important to know the country from which the AS is operated. It is reasonable to assume that this is also the country where most of

its prefixes are registered. Although it will not be true for all ASes, it will hold for most.

The country code for an AS is taken from IP2Location LITE by collecting the country codes for each IP prefix and finding the most occurring code. This code represents the country where most of the prefixes for that AS are located. When there are two or more codes that share the maximum number of occurrences, they are all listed. The resulting country information from IP2Location LITE is thus a list with one or more country codes. This list may be 'None', indicating there is no information available for the AS itself, or contain only '-' if none of the prefixes have a country code.

A similar process is followed to get the country codes from the GeoLite2 data. The country codes for each prefix are collected. In this case, each prefix has three country codes. The actual country, the country where it is registered, and the country that is represented by the users. In most cases the first two codes are equal and the last is 'nan'. It is therefore decided to put the first two country codes of all prefixes in one list and find the most common one. Again, if two or more codes share the maximum number of occurrences, they will all be listed. The result is a list of one or more country codes. If there is no country information available for the AS, the result will be 'None'. If none of the prefixes have a country code, the result will be `['nan']`.

This feature combines the results from the three data sources into a tuple. The tuple has the format `(<GeoLite2 result>, <RIR statistics result>, <IP2Location LITE result>)`, where the GeoLite2 and IP2Location LITE results are both lists, and the RIR statistics result is a country code. This feature is computed for both the detected origin AS and the expected AS. The results of the three data sets are combined to ensure accuracy.

To verify that the country of an AS does not often change over time, the results are compared. For each AS it is checked if there is a difference between the results from the GeoLite2 data, the RIR statistics, and the IP2Location LITE data. Each hijack has two associated AS numbers. In total, the 2665 hijacks together yield 2582 unique AS numbers. For each of these 2582 ASes, the country information is compared between the three data sources. A match between two data sources, in this case, means that they share a common country. As the RIR statistics always result in one country code, it should be present in the list resulting from another source to be counted as a match.

This approach gives the following results. Between GeoLite2 and IP2Location LITE, there are 93.65% of ASes that have the same country in the list of most occurring countries. There is a match between the RIR statistics and GeoLite2 for 84.0% of ASes and a mismatch due to unavailable data in GeoLite2 for 6.97% of ASes. The mismatch caused by unavailable data in GeoLite2 is however not a real mismatch. It can be caused when, for example, the ASN is allocated by a RIR but only used to announce a prefix that has no geographical information. The results for IP2Location LITE are similar to those of GeoLite2. There is a match for 83.42% of the ASes and a mismatch due to unavailable data in IP2Location LITE in 6.89% of the ASes.

In summary, there is a match between each data set for approximately 91% of the cases. If the data sets are reliable, the accuracy of this feature would thus be close to 91%. Due to the lack of historical data and the fact that many ASes operate in multiple countries, it is difficult to gain higher accuracy. However, as this data is only used to get a global picture of hijacks, it is not a huge issue.

AS organisation ID The BGPStream hijacks come with the organisations of the expected and detected AS. However, this is the organisation name. Because the name sometimes changes over time or is spelled differently between ASes, the organisation ID is taken from the CAIDA AS Organizations data set. This ID is used to find hijacks that happen to or are caused by the same organisation. This data set has a new file uploaded every three months. The file used for this feature is the one from the date closest to the hijack start time.

The data set uses WHOIS information as its source. This is not fully reliable and often there is no organisation ID found for an AS. An option is to also use the next file, but this still results in a lot of unavailable IDs, and as these files are 3 months apart the data may have changed after the hijack. For

this reason, this feature is not relied on to label hijacks, it is only used for extra information.

Closest prefix announced by AS When an origin hijack is detected, it is because the detected origin AS is announcing a prefix that it should not be announcing. There are several reasons why this may happen, one of which is a typographical error in the prefix. For example, an AS that normally announces the prefix 192.0.2.0/24, may now announce 192.0.3.0/24 by mistake. While this can be an intentional hijack, it may also be the result of mistakenly typing a 3 instead of a 2.

By looking at what an AS normally announces one can detect if the hijack is possibly caused by a typographical error. The first step is to define when a prefix is 'normally announced'. The AS history from RIPEstat is used to find all prefixes announced by the AS in the year before the event to a week after the event. This period is chosen because it gives a recent overview of the announced prefixes.

Some prefix announcements are very short-lived, but others go on for a long time. To determine what can be considered a normal period of announcement, some statistics are calculated using the AS histories. For each AS history, the number of announced prefixes is counted. In addition, a list is kept that holds the total duration of the announcements for each prefix. Another list holds the length of each period a prefix is announced. In other words, for each prefix there is the total announcement duration and a list holding the duration of each period the prefix was announced. The length of this last list is the number of periods a prefix was announced. By keeping this data for each prefix announced by each AS the following statistics can be calculated. The average total time a prefix is announced is 16.97 weeks. The median total time a prefix is announced is 7.1 weeks. On average there are 1.27 periods in which a prefix is announced.

Using these statistics a 'normally announced' prefix is defined as a prefix that is announced for at least six weeks in total. This is one week below the median and is chosen to weed out most short-lived announcements. For clarification, a prefix that is announced for one period of at least six weeks is considered normal, and a prefix that is announced over any number of separate periods that in total make six weeks is also considered normal. As the average number of periods in which a prefix is announced is close to 1, this approach is reasonable. It is difficult to find a perfect solution because there are many edge cases one can think of. It is not feasible to look more closely and there is no ground truth that defines a normal announcement.

Using this definition of a normally announced prefix, one can find the closest prefix. The closest prefix announced by an AS is the prefix with the lowest string distance to the detected advertisement. When finding the closest prefix, the prefix length is ignored. This is handled in another feature. The string distance is calculated using the restricted Damerau-Levenshtein [33] distance implemented in the StringDist package [31].

Choosing an appropriate string distance measure is important as there exist many different measures. One requirement, in this case, is the ability to compare two strings of different lengths. This is because IPv4 prefixes range between 7 and 15 characters (0.0.0.0 - 255.255.255.255) and IPv6 prefixes between 2 and 39 characters (:: - ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff). It should be possible to compare two prefixes of the same version but different lengths. As discussing every possible string distance measure is outside the scope of this thesis, an explanation of why the Damerau-Levenshtein distance is suitable should suffice.

Mistakes made when typing a prefix can be divided into the following categories:

- Forgetting a character
- Adding a character
- Switching two adjacent characters
- Mistyping a character

Each of these mistakes can be made multiple times and each time such a mistake is made, the distance should preferably increase by 1. Fortunately, this is exactly what the chosen implementation of the Damerau-Levenshtein distance does. The weight for each of the listed mistakes is equal. If there is no difference between two prefixes, the distance will be 0. The distance between two different prefixes can be defined as the number of operations that have to be performed in order to turn one prefix into the other. Some examples are given in table 5.2.

Mistake made	Prefix 1	Prefix 2	Distance
Forgetting a character	192.0.2.0	12.0.2.0	1
Adding a character	192.0.2.0	192.0.23.0	1
Switching two adjacent characters	192.0.2.0	19.20.2.0	1
Mistyping a character	192.0.2.0	192.0.3.0	1
Combination	192.0.2.0	19.20.23.0	2
Different prefix	192.0.2.0	198.51.100.0	6

Table 5.2: String distance between prefixes

Finding the closest prefix is done as follows. First, a list is created that contains all prefixes that are announced for at least six weeks in the year before the event. For each prefix in this list, the string distance between the prefix and the detected advertisement is calculated. The prefixes that have the lowest distance are saved in a list. This list is returned together with the lowest distance. This feature is represented by the tuple `(<lowest distance>, <prefix list>)`. This feature is computed for the detected origin AS. If the lowest distance is 0, the prefix has been announced before. Possibly with a different prefix length.

Difference in prefix length The previous feature contains a subset of the list of prefixes that are normally announced by the detected origin AS. The prefixes in this subset have the lowest string distance between the detected advertisement and any of the prefixes in the complete list. The previous feature is used to detect typographical errors in the prefix and to detect if the detected origin AS has announced the detected advertisement before.

Typographical errors can also be made in the prefix length. This feature takes the detected advertisement and the list from the previous feature. It gives the smallest difference in prefix length between the detected advertisement and the prefixes in the list. This value can be zero, positive, negative, or infinity. If this difference is infinite, this means that the list of prefixes contains only prefixes of a different version or no prefixes at all. A positive value means that the detected advertisement is more specific, a negative value means the prefix from the list is more specific. If the value is 0, there is no difference. If both this value and the lowest distance from the previous feature are 0, this means that the detected AS has announced the detected advertisement before.

5.1.2. Prefix features

This subsection discusses the features that are directly related to the detected advertisement or expected prefix. Each paragraph explains one feature. Table 5.3 gives an overview of the features.

Feature	Description	Type
IP version	IP version of the detected advertisement. Given by integer representing version 4 or 6.	Integer
Detected advertisement subnet of expected prefix	Boolean indicating if the detected advertisement is a subnet of the expected prefix.	Boolean
Special prefix	Boolean indicating if the detected advertisement is part of a range of IP addresses that is reserved for special purposes.	Boolean
RPKI	List of ROAs for detected advertisement and expected prefix.	List

Prefix in use	Boolean indicating if the detected advertisement was in use during the six months before the hijack.	Boolean
Prefix status	Is the prefix assigned, allocated, available, or reserved. Includes RIR and, if applicable, a date.	String
Prefix country	Prefix country taken from GeoLite2, RIR statistics, and IP2Location LITE	Tuple
Announcing subnet	List of prefixes that are normally announced by the detected AS, and of which the detected advertisement is a subnet.	List
Announcing supernet	List of prefixes that are normally announced by the detected AS, and of which the detected advertisement is a supernet.	List
Blacklisted prefix	Dictionary including timelines of when the detected advertisement, expected prefix, and other supernets and subnets were blacklisted.	Dictionary

Table 5.3: List of prefix features

IP version The IP version feature shows whether the hijack affects an IPv4 or IPv6 prefix. The feature is used to find out whether IPv4 or IPv6 prefixes are more affected by hijacks, and if this is proportional to the global use of IPv4 and IPv6. The feature takes an integer value that is either 4 or 6.

Detected advertisement subnet of expected prefix This feature checks if the detected advertisement is a subnet of the expected prefix. This is represented by a boolean value that is set to true of the detected advertisement is a subnet, and false if it is not. This information is used to make a distinction between global and local hijacks, the difference between the two was explained in chapter 2.

Special prefix Some parts of the IPv4 and IPv6 address space are used for special purposes like private networks, loopback, or documentation. IANA keeps a list of the prefixes that are reserved for such purposes. This feature checks if the detected advertisement is a special prefix or a subnet of a special prefix. This is represented by a boolean value that is set to true if the detected advertisement is special, and false otherwise. This feature is used to detect cases where a special prefix is showing up in BGPStream as the victim of a hijack.

RPKI For each detected advertisement and expected prefix, the RPKI validator is used to extract Route Origin Authorisations (ROAs). This feature is represented by a list holding the ROAs. If none is found for a prefix the list is empty, otherwise the list contains one or multiple ROAs. This feature is used to get statistics on RPKI with respect to origin hijacks. It is also used to detect cases where the detected origin AS has a ROA for the detected advertisement. In this case, the detected hijack is most likely not a hijack, but a legitimate announcement. This cannot be said with full certainty because a ROA does not contain a creation date. It may thus have been created after the hijack. However, this still makes it likely that the announcement was legitimate.

Prefix in use Section 3.3 discussed a paper that Vervier et al. [54] published in 2015. One of the conclusions in that paper was that spammers often hijack IP prefixes that were never announced or had not been announced for a long time. This feature shows whether the detected advertisement was used in the six months before the hijack occurred for a period of at least six weeks. This information is computed using the prefix history and is represented by a boolean value. If the detected advertisement or the expected prefix is announced by any AS for six weeks or longer during the six months before the hijack, the value is set to true. Otherwise, it is set to false.

Prefix status Similar to the AS number status discussed in 5.1.1, a prefix also has a status. The feature 'special prefix' shows if a prefix is reserved for special purposes. This feature takes the prefix status from the RIR statistics. It uses the same format as the AS status. The feature gives a tuple of three elements: `(<registry>, <status>, <date>)`. The tuple will have the registry to which the prefix is assigned as the first element, the status found in that RIR's statistics as the second element, and the corresponding date as the third element. The status takes one of the following values: 'assigned', 'allocated', 'available', or 'reserved'. This feature is used to find the status of the detected advertisement and the expected prefix at the time of the hijack.

Prefix country The prefix country is computed for the detected advertisement. It is represented as a tuple of three elements containing the results from GeoLite2 data, IP2Location LITE data, and RIR statistics. It uses the same format as the AS country feature:
`(<GeoLite2 result>, <RIR statistics result>, <IP2Location LITE result>)`

Computing the prefix country is easier than computing the AS country because each data source has IP prefixes linked to country codes. The results are thus acquired by looking up the prefix in the data sets. Again, the results are compared between data sources. GeoLite2 data matches with RIR statistics for 90.48% of prefixes and has missing GeoLite2 data for 1.42% of prefixes. IP2Location LITE data matches with RIR statistics for 80.73% of prefixes and has missing IP2Location LITE data for 0.08% of prefixes. IP2Location LITE and GeoLite2 match for 90.8% of prefixes. In this case, GeoLite2 seems to be the most accurate source. The information gained from this feature is used to get a global overview of hijacked prefixes.

Announcing subnet Sometimes a hijack is caused because an AS fails to aggregate its prefixes. It then announces a set of subnets instead of the prefix itself. This feature uses the AS history to check if the detected origin AS normally announces a supernet of the detected advertisement. The feature is represented by a list of prefixes that are announced for at least six weeks in the year before the hijack, and of which the detected advertisement is a subnet. This feature is used to detect a failure to aggregate.

Announcing supernet Instead of failing to aggregate prefixes, a failure to summarize is also possible. In that case, the AS announces a supernet of its prefix. This feature uses the AS history to check if the detected origin AS normally announces a subnet of the detected advertisement. The feature is represented by a list of prefixes that are announced for at least six weeks in the year before the hijack, and of which the detected advertisement is a supernet. This feature is used to detect a failure to summarize.

Blacklisted prefix RIPEstat keeps blacklist data for prefixes. A prefix is blacklisted if it is used to send spam. To find out if the prefix is hijacked to send spam, the blacklist data of the detected advertisement is looked up. If the prefix appears on a blacklist after the hijack, this strongly suggests the hijack was malicious.

This feature collects blacklist data for the detected advertisement. This includes the data for its supernets and subnets. The feature is represented by a dictionary. It includes the timelines of when the prefix was blacklisted, the total number of times the prefix was blacklisted, and the total duration of the blacklist timelines. The same data is also given for each of its blacklisted supernets and subnets.

5.1.3. Event features

The remaining features are discussed in this section. They are generated using information other than the involved ASes and prefixes. Each paragraph discusses a feature. Table 5.4 gives an overview.

Feature	Description	Type
AS23456	Boolean indicating if the transition AS appears in the AS path.	Boolean
MOAS conflict	List of ASes that announce the detected advertisement at the time of the hijack.	List

Path prepending	Number of times the detected origin ASN appears at the beginning of the AS path.	Integer
Path relations	Relations between neighbouring ASes in the AS path.	String
Continuous path	Boolean indicating if all neighbouring ASes in the AS path are directly related to each other.	Boolean
Valley-free path	Boolean indicating if the AS path is valley-free	Boolean
Path length	Number of ASNs in the AS path	Integer
Hijack duration	Hijack duration taken from prefix history and AS history	Tuple

Table 5.4: List of event features

AS23456 AS23456 or AS_TRANS is the transition AS that handles 32-bit numbers for incompatible routers. If this is handled correctly, then this number should be replaced by the original ASN. When this AS shows up in the AS path, it may indicate a misconfiguration. The feature is represented by a boolean value that takes true if AS23456 is part of the AS path, and false otherwise.

MOAS conflict When an AS announces a prefix that is owned by another AS, this may cause a MOAS (Multiple Origin AS) conflict. A MOAS conflict happens when two ASes are both announcing the same prefix. The prefix will then seemingly originate from two different ASes. This feature helps to detect hijacks of prefixes that are in use at the time of the hijack. The prefix history of the detected advertisement is used to create a list of all ASes that announce the prefix at the time of the hijack. Each element of the list is a tuple containing the number of the AS announcing the prefix, and the start and end time of the announcement.

Path prepending Each hijack comes with an AS path. This path is the route from the detected origin AS to a BGPMon peer. When a BGP speaker decides on a path using the path selection algorithm, the length of the path can be taken into consideration. A shorter path is preferred over a longer path. A more thorough explanation of this process was given in chapter 2.

Sometimes an AS prepends the path with its own AS number to make the path longer. When intentionally hijacking a prefix, one desires that the announced path is chosen. In general, it is thus preferable that the path length is as short as possible. Path prepending goes directly against this principle. This feature gives the number of times the detected origin AS appears at the front of the AS path as an integer value. If it appears more than once, this may suggest the hijack is not a malicious intentional hijack.

Path relations This feature shows the relationships between the ASes in the AS path. The relations are taken from the CAIDA AS Relationships data set. The files used are those from the month in which the hijack occurs and the month after. There is a difference between the two months in 726 (27.24%) cases. The feature is represented by a string that consists of a combination of the following substrings: ‘p2p’ (peer to peer), ‘p2c’ (provider to customer), ‘c2p’ (customer to provider), ‘no relation’, and ‘self’.

Consider the following example. The given AS path is ‘5 4 3 2 2 1’. Assume that AS1 is a customer of AS2. AS2 is a provider of AS3. There is no relation between AS3 and AS4, and AS4 is a provider of AS5. The resulting path relations are then `c2p, no relation, c2p, self, p2c`. This feature is used as a basis for the two features that are discussed next.

Continuous path The feature ‘path relations’ gives the relationships between the ASes in the AS path that comes with the hijack. One possibility is that two consecutive ASes in the path have no relation. When there is a gap between two ASes the path is not continuous and thus invalid. This can happen for two reasons. Either the relation data is incorrect, or the path announcement is false. A false path suggests the possibility of a path hijack.

This feature can take three possible values: true, false or '-'. The path relations from the month of the hijack and the month after are compared. If both have 'no relation' in the same position, the path is considered to not be continuous. The value is then set to false. If the path is continuous in one or both months the value is set to true. There are six cases where the position of the 'no relation' changes between the two months. This means that there is a different gap in the path. Because it is unclear if this happened before or after the hijack, it cannot be said with certainty if the path is continuous. For these cases the value is set to '-'.

Valley-free path The concept of valley-free paths was explained in chapter 2. This feature checks if the AS path is valley-free. This is done by checking the path relations for the month of the hijack and the month after. If the path is not continuous, the path can also not be valley free. For continuous paths it is checked if they adhere to one of the following formulas: `n*c2p + p2p + m*p2c` or `n*c2p + m*p2c`. If that is the case in at least one month, the AS path is considered to be valley-free and this feature is set to true. Otherwise, it is set to false.

As valley-free paths are a theoretical concept and not a strict requirement for AS paths, this feature is not used to label hijacks. It is only used to see how many of the paths that reach a BGPMon peer are valley-free.

Path length The length of the AS path can be considered during the path selection process. As shorter paths are preferred over longer paths, a short path is preferred when hijacking a prefix. The origin AS does not have much influence over the path length. It can only make the path longer by prepending its own AS number. This feature is therefore not used when labelling hijacks, but only to find out the distribution of path lengths. The path length is computed by counting the number of ASes in the AS path. This includes doubles. The length of the path '5 4 3 2 2 1' would thus be 6.

Hijack duration This feature is used to distinguish between long-lived and short-lived hijacks. The hijack duration is calculated using both the prefix history of the detected advertisement and the AS history of the detected origin ASN. In the prefix history is looked for an announcement by the detected origin AS at the time of the hijack. The AS history is used to find the announcement of the detected advertisement at the time of the hijack. The duration of the announcement is calculated for both cases, and both are returned in a tuple. This results in the following tuple:
`(<prefix history duration>, <AS history duration>)`
It is chosen to return both because there tends to be a difference between the two. The reason for this difference is not clear.

5.2. Finding relations between hijacks

Each possible hijack detected by BGPStream is presented as a solitary event. However, many hijacks are related to others in some way. For example, hijacks that are caused by the same AS at the same time are related; they are said to be part of the same event. Knowing relations between hijacks can help labelling them because it gives more insight in the circumstances in which a hijack happened. This section focusses on detecting relations between hijacks. Every paragraph in this section discusses a way in which hijacks can be related and how this relation is detected.

AS The first way in which hijacks can be related is by AS. This can either be the detected origin AS or the expected AS. Hijacks with the same detected origin AS are caused by the same AS, while hijacks with the same expected AS have the same target AS. To group hijacks by AS each hijack gets two group IDs. The first group ID is given based on the detected origin AS of the hijack. The hijacks that share the same detected origin AS get the same group ID. Similarly, the second group ID is given based on the expected AS. Hijacks that have the same expected AS will get the same group ID.

Prefix Similar to hijacks that are related by detected or expected AS, hijacks can also be related by prefix. They can share the same detected advertisement or the same expected prefix. If hijacks share the same detected advertisement they are targeting the same prefix. Sharing the expected prefix

means they are targeting either the same prefix, or prefixes that are related because they are part of the same supernet. Grouping hijacks by prefix is done in the same way as grouping hijacks by AS. Each hijack gets two extra group IDs. One for the detected advertisement, and one for the expected prefix. Hijacks that share the same prefix will get the same group ID.

Event Occasionally an AS hijacks multiple prefixes at the same time or within a short period. Multiple hijacks caused by the same AS within a short period are likely part of one event. A misconfiguration, for example, can cause multiple hijacks at the same time. Finding these groups of hijacks is done by looking for hijacks that are caused by the same AS and happen within two hours of each other. These hijacks will share the same group ID.

The two hour period is chosen because it groups hijacks that happen closely together. It is possible that an event causes multiple hijacks over a longer period, but increasing the period also increases the chance of grouping hijacks caused by different events. However, as the hijacks are already grouped by detected AS, events that cause hijacks over a period longer than two hours can still be detected.

Prefixes hijacked by same set of detected ASes Another way to group hijacks is to look at prefixes that are hijacked by the same set of ASes. This is best explained by looking at an example. The example in table 5.5 is taken from the actual data. The prefixes 103.69.168.0/24 and 103.69.169.0/24 are both hijacked by AS1, AS2, AS1007, and AS1008. All these hijacks happen within 20 minutes. It is unlikely that this is a coincidence, but the relation would not be visible if the hijacks were not grouped properly. The relation is detected by looking for prefixes that are hijacked by multiple ASes. If prefixes share the same set of detected origin ASes, they will get the same group ID.

Start time	Detected advertisement	Detected origin AS	Expected AS
2019-05-25 13:02:17 UTC	103.69.168.0/24	1008	8
2019-05-25 13:02:17 UTC	103.69.169.0/24	1008	8
2019-05-25 13:05:47 UTC	103.69.168.0/24	1007	8
2019-05-25 13:05:47 UTC	103.69.169.0/24	1007	8
2019-05-25 13:11:37 UTC	103.69.168.0/24	2	8
2019-05-25 13:11:37 UTC	103.69.169.0/24	2	8
2019-05-25 13:22:59 UTC	103.69.168.0/24	1	8
2019-05-25 13:22:59 UTC	103.69.169.0/24	1	8

Table 5.5: Prefixes hijacked by the same set of detected ASes

Detected AS becomes expected AS or vice versa There are many cases in which a prefix is hijacked multiple times. Sometimes the hijacking AS is later detected as the owner of the prefix. In other words, the detected AS becomes the expected AS. Of course, the other way around is also possible; if an AS hijacks a prefix it previously owned, the expected AS is now the detected AS. Following is an example to clarify the relation.

Start time	Detected advertisement	Detected origin AS	Expected AS
2019-02-31 00:00:00 UTC	192.0.2.0/24	1	2
2019-05-12 00:00:00 UTC	192.0.2.0/24	3	1

Table 5.6: Detected origin AS becomes expected AS

These relations are detected by looking for prefixes that are hijacked multiple times. By combining these hijacks, one can find the intersection between the detected origin ASes and the expected ASes. If the intersection is not empty, then there is at least one AS that changed from detected origin AS to expected AS or vice versa. In the example, the prefix is hijacked twice. The set of detected ASes is $(1, 3)$, the set of expected ASes is $(1, 2)$. The intersection of the two sets is (1) and thus not the empty set. AS1 hijacked the prefix on the 31st of February and was the owner on the 12th of May. Hijacks that are part of a group in which there has been such a switch get the same group ID.

It is possible that the first hijack was actually a legitimate announcement. If the prefix had just changed owners, the announcement by AS1 could by detected by BGPStream as an origin hijack because they were using outdated information.

5.3. Labelling hijacks

Using the new features and the relations between hijacks, the hijacks can be labelled. This section explains the process of labelling hijacks. The first step is to filter out invalid hijacks. This is explained in subsection 5.3.1. The second step is to find the hijacks that are likely legitimate announcements. Subsection 5.3.2 describes how this is accomplished. After looking for invalid hijacks and legitimate announcements the hijacks are further labelled. Subsection 5.3.3 discusses the labels that are given based on the hijack features and subsection 5.3.4 explains labels that are given based on the relations between hijacks. Not all labels are mutually exclusive. They can be combined to further classify hijacks. This is done by analysing groups of hijacks with the same labels and is explained in subsection 5.4.3.

5.3.1. Invalid hijacks

Not all hijacks that appear on BGPStream are valid. Some contain invalid data or inconsistencies in the BGPStream features. The first step is to filter out these hijacks and label them as invalid. By analysing the data, two categories of invalid hijacks are found. One category contains hijacks with an invalid AS number. This is a number that falls outside of the range of legitimate AS numbers. The other category contains hijacks of which the detected origin AS does not match the origin AS in the AS path. These hijacks are given the label ‘invalid hijack’ because the features resulting from the BGPStream data are unreliable. In total there are 13 invalid hijacks.

5.3.2. Legitimate announcements

BGPStream detects possible origin hijacks. This means that hijacks detected by BGPStream can be legitimate announcements. After filtering out the invalid hijacks, the second step focusses on hijacks that are likely legitimate. An AS should only be announcing a prefix if it is the owner of the prefix. One way to find the owner of a prefix is to look if there exists a ROA that has the detected origin AS listed as the owner of the detected advertisement. If this is the case, the hijack is most likely legitimate. It cannot be said with certainty because the ROAs do not have a creation date. However, it is more unlikely that an AS hijacks a prefix that later becomes its own.

Another way to detect announcements that are likely legitimate is by using the group IDs for hijacks where the detected AS became the expected AS for that prefix. If the detected origin AS later becomes the owner of the prefix, it is likely that the possible hijack was a legitimate announcement. Hijacks that meet one of these two conditions are labelled as ‘likely legitimate’. Using this approach there are 89 hijacks found that are most likely legitimate announcements.

5.3.3. Labels based on hijack features

This subsection contains an explanation of the labels that are used to categorise hijacks. These labels are assigned to hijacks using the features that were discussed in section 5.1. Each paragraph in this subsection discusses one label. It is explained why the label is used and how it is assigned to hijacks.

Hijack of 1 IP address A hijack does not always affect a range of IP addresses. In some cases, an AS hijacks only one IP address. These hijacks are labelled because it is not common practice to announce just one IP address. Most announced prefixes have a maximum length of 25 for IPv4 and 48 for IPv6. [49] This label is given by looking at the range of the detected advertisement. If the detected advertisement only contains one IP address, the hijack is given the label ‘1 IP address’. Based on this label the hijacks of one IP address can be analysed together to see if an explanation can be found using the other labels.

Failure to summarize A failure to summarize means that an AS announces a supernet of the prefixes it owns. Assume for example that an AS owns the prefixes 192.0.2.0/24 and 192.0.3.0/25. These prefixes cover the range of IP addresses from 192.0.2.0 to 192.0.3.127. The AS should announce

these two prefixes. If instead, it decides to announce the supernet 192.0.3.0/23, which covers the prefixes 192.0.2.0 to 192.0.3.255, it will also be announcing a route for the addresses 192.0.3.128 to 192.0.3.255 that it does not own. This is called a failure to summarize. This label is given to any hijack of a supernet of prefixes that are owned by the detected AS and cover at least 75% of the range of the supernet. This number is chosen because if the AS only owns a small part of the prefix it announces, it is not necessarily a failure to summarize. The AS is then just announcing a supernet of its prefixes. Furthermore, if the AS does own the whole range covered by the announced supernet, it is a legitimate announcement because the AS is only aggregating its prefixes. This label is a definite label because it identifies a clear cause of the hijack.

Failure to aggregate A failure to aggregate is the opposite of a failure to summarize. When an AS fails to aggregate its prefixes, it announces a set of subnets instead of the supernet covering the same range. Because an AS is allowed to announce only part of its prefixes, the label 'failure to aggregate' is only given to a hijack if the AS also announces the other subnets at the same time. For example, if an AS owns the prefix 192.0.3.0/23 and announces the prefixes 192.0.2.0/24 and 192.0.3.0/24, it fails to aggregate. Instead of the two /24 prefixes, it should have announced the /23. If the AS decides to announce only one of the two /24 prefixes, it is not a failure to aggregate because an AS may decide not to use part of its prefixes.

A failure to aggregate may show up in the hijacks when an AS has recently become the owner of a prefix. Assume that the detected AS had been announcing the prefix 192.0.2.0/23 and one of its customers announced the prefix 192.0.3.0/24. This is common practice because traffic will always be routed to the most specific prefix. Now assume that the 192.0.3.0/24 changes ownership again and is given back to the detected AS. When the detected AS starts announcing this prefix, the announcement may be detected as a possible hijack because the owner changed. The feature 'announcing subnet' will show that 192.0.3.0/24 is a subnet of the prefix 192.0.2.0/23 that is normally announced by the detected AS. If instead of this /23 prefix, the prefix will now announce the /24 prefixes, the hijack is labelled as 'failure to aggregate'. Just as 'failure to summarize' this label identifies a clear cause of the hijack and is thus a definite label.

Possible typographical error An origin hijack may be the result of an AS announcing a prefix that is very similar to one of its own due to mistyping its prefix. This label is given using the feature 'closest prefix announced by AS'. This feature comes with the string distance between the detected advertisement and the closest prefix that is normally announced by the detected AS. If this string distance is 1 and there is no difference in the prefix length, the hijack is labelled as possibly caused by a typographical error. Looking into the detected advertisement and the closest prefix of hijacks with this label can show how many hijacks may be caused by a simple misconfiguration.

Wrong prefix length Aside from a typographical error in the address part of a prefix, a mistake can also be made in the prefix length. The features 'closest prefix announced by AS' and 'difference in prefix length' are used to detect these cases. If the distance between the detected advertisement and the closest prefix is 0, and there is a difference in prefix length, the hijack receives the label 'wrong prefix length'. It is not necessarily always unintentional, but this label does provide a clear cause of the hijack and is thus a definite label.

Hijacking customer route This label is given to a hijack when the detected AS is the provider of the expected AS. The expected AS owns the prefix, but during the hijack, its provider is (also) announcing the prefix. The provider is thus hijacking a route of its customer. If the customer has only one provider it does not make sense to do this with malicious intent, because most traffic destined for the expected AS already passes the detected AS. If the customer has multiple providers the hijack could force traffic to be forwarded via the detected AS, but this also depends on the topology and whether the hijack is global or local. This label is definite because although it is not known if these hijacks are caused intentionally, the cause of these hijacks is clear.

Hijacks of prefix that appears on blacklist The feature 'blacklisted prefix' is used to label hijacks that are either causing a prefix to be blacklisted or are hijacking a prefix that is blacklisted at the time

of the hijack. When an IP prefix is blacklisted it is used to send spam. It is not always the detected advertisement that is blacklisted. Sometimes it is a supernet or subnet. All these cases are considered when labelling. This results in eight possible labels. Four of these labels refer to hijacks of a prefix that was already blacklisted and four to hijacks of a prefix that was blacklisted during the hijack.

The four labels in each group refer to the prefix that was blacklisted: one for the detected advertisement, one for the expected prefix, one for other supernets of the detected advertisement, and one for subnets of the detected advertisement. For example, in case the detected advertisement is blacklisted during the hijack, the hijack gets the label 'detected blacklisted after', meaning the detected advertisement is blacklisted after the start of the hijack but before the end of the hijack, according to the hijack duration. When an AS hijacks a prefix of which a subnet was blacklisted when the hijack started, the hijack gets the label 'subnet blacklisted before'. Each hijack can receive multiple labels. The labels are used to find hijacks that cause spam, but also to find hijacks of prefixes that were being misused at the time of the hijack.

Hijacks of unused prefix An AS that owns a prefix is not obliged to announce this prefix. When a prefix is not announced by any AS for longer than 6 weeks in the 6 months before the hijack, the prefix is called an unused prefix. Hijacking an unused prefix is in some cases done with the intention of misusing the prefix. When this label is combined with the labels based on the IP blacklists that were discussed in the previous paragraph, it shows if unused prefixes are hijacked and used to send spam. This label is assigned based on the feature 'prefix in use'.

Announcing subnet/supernet ASes that are announcing a subnet or a supernet of a prefix they own are not always failing to summarize or aggregate. The labels 'announcing subnet' and 'announcing supernet' are given to hijacks based on the corresponding features. This label is used to find hijacks of prefixes that are related to a prefix that is owned by detected AS. These hijacks can be done intentionally, but can also be the result of a misconfiguration or mistyping the prefix length.

Labels based on RPKI When there is a ROA created for an IP prefix, this ROA lists the legitimate owner of the prefix. This owner should be the one announcing the prefix. It is possible that a prefix has multiple ROAs and thus multiple owners. However, an AS that is not listed as the owner should not be announcing the prefix. Based on the ROAs found for the detected advertisement each hijack can receive four possible labels. If there is no ROA found, the hijack gets the label 'no ROA'. If one or multiple ROAs are found, the hijack receives labels based on the ASes listed as the owner. In case the detected AS is listed the hijack receives the label 'ROA for detected ASN'. Similarly, there is the label 'ROA for expected ASN' when the expected AS is listed, and 'ROA for other ASN' when an AS is listed that is not the detected or expected AS. Aside from finding legitimate announcements these labels also help to find if prefixes that have a ROA are still hijacked.

AS23456 appears in AS path The transition AS, AS23456, is used by routers that cannot handle 32-bit AS numbers. It temporarily replaces the 32-bit numbers and is replaced by the original numbers again by an AS that can handle those. If AS23456 shows up in the AS path this may indicate a misconfiguration. This label indicates whether or not AS23456 is part of the detected AS path.

Hijacks with AS path prepended by the origin When an AS prepends the path with its ASN multiple times it does so to decrease the chance the path is used. When hijacking a prefix so that the traffic is sent to the detected AS instead of the origin AS, it is thus illogical to prepend the path. This label is given to hijacks with a path that contains the origin ASN more than once. The 'path prepending' feature is used for this purpose.

Global hijack Global hijacks are hijacks of a subnet of a prefix that is normally announced by the expected AS. These hijacks often have more impact because they can affect a larger part of the BGP topology. This label is used to find out whether there are some clear differences in the characteristics of global and local hijacks. The label is given using the feature 'detected advertisement subnet of

expected prefix'. If the detected advertisement is a subnet of the expected prefix the hijack is global. Otherwise, it is a local hijack.

MOAS conflicts When a prefix is announced by multiple ASes it is called a MOAS conflict. This is not necessarily a bad thing, sometimes it is intended. The feature 'MOAS conflict' shows which ASes were announcing the detected advertisement at the time of the hijack. Based on this feature a hijack can receive several labels. These labels indicate if the AS causing the conflict is related to the detected or expected AS of the hijack. There are five possible labels. Hijacks can receive multiple labels.

If there is an AS that announces the detected advertisement and is not related to the detected or expected AS, the hijack will receive the label 'MOAS unrelated AS'. If the expected AS is announcing the prefix, the hijack receives the label 'MOAS expected AS'. The remaining three labels are for MOAS conflicts caused by ASes that are related. These labels are 'MOAS related to expected', 'MOAS related to detected', and 'MOAS related to detected and expected'. These labels give insight into how many ASes are announcing the prefix. In general only the expected AS should announce the prefix. If there are multiple ASes announcing the prefix, it may be more likely that the detected AS is also legitimately announcing this prefix.

Labels based on the AS number status This paragraph explains several labels that are given based on the AS number status for the detected and expected AS. In total, there are 7 labels that can be given based on these statuses. An ASN that is not assigned to a RIR is either 'unallocated by IANA' or 'reserved by IANA'. When an ASN is assigned to a RIR it does not mean the ASN is use. It can still be 'available by RIR' or 'reserved by RIR'. When a RIR does allocate an ASN, it lists the date of the allocation. Using this date and the start time of the hijack the last three possible labels are given. These labels are 'allocated by RIR before hijack', 'allocated by RIR on day of hijack', and 'allocated by RIR after hijack'.

In theory, only ASes that are allocated should be announcing prefixes. Unallocated ASes should thus not show up as the expected AS in a hijack, because this means they are announcing something. They should also not show up as the detected AS because, aside from the fact that they should not cause hijacks at all, if they are not owned by anyone, they should not be in use. When a hijack thus involves an AS that is reserved or not yet allocated it is suspicious. These hijacks need to be examined to see if there is an explanation.

Hijacks involving reserved or special prefixes The previous paragraph discussed how labels are given based on the AS number status of the detected and expected AS. Similar labels are given based on the status of the detected advertisement and the feature 'special prefix'. The status of the detected advertisement allows for five possible labels. These labels are 'prefix reserved', 'prefix available or without status', 'allocated before hijack', 'allocated on day of hijack', and 'allocated after hijack'. If the detected advertisement is a special prefix or part of a special prefix it will be labelled as such. These labels are used to find hijacks of prefixes that should not be in use.

Detected AS announced hijacked prefix before When an AS shows up as the detected AS in a hijack, it is not always the first time the AS announces that prefix. Sometimes the AS hijacked the prefix before, in other cases it was the owner of the prefix. This label is given to hijacks where the detected AS has announced the detected advertisement before. The features 'closest prefix' and 'difference in prefix length' are used for this purpose. When the string distance between the detected advertisement is 0 and the difference in prefix length is 0, the detected AS has announced the prefix before. Because the 'closest prefix' feature only contains prefixes that have been announced for at least six weeks, this label is given only to hijacks in which the detected advertisement is considered to be a normal announcement for the detected AS. In this way, it can help to find legitimate announcements and cases where the detected AS hijacks a previously owned prefix.

AS path is not continuous AS paths that are not continuous are either the result from incorrect data or from path hijacks. This label is given to hijacks with a detected AS path that is not continuous. It

allows for further inspection of these hijacks to see if the AS path and the path relations can be used to determine whether a hijack is a path hijack or just a hijack with an incorrect AS path.

Hijack duration The hijack duration feature gives the duration of the announcements found in the AS history of the detected AS and the prefix history of the detected advertisement. To get a better overview of the hijack duration with respect to the other labels, the maximum duration is taken and converted from seconds to days. There are four categories created based on this duration: hijacks that last less than a day, up to a week, up to a month, and more than a month. Each hijack receives a label based on the category it belongs to. In general, it is expected that hijacks that last less than a day are more often the result of misconfigurations and hijacks that last longer than a month are more likely to be legitimate announcements. The labels are used to find if this is true for the hijacks in the data set.

Hijack of a prefix that follows a pattern While engineering hijack features it came to light that there is an unexpectedly high number of hijacked prefixes that follow a certain pattern. These prefix all appear to be IPv4 prefixes of the form x.x.x.0/y, where x is a number between 1 and 255 and y is a valid IPv4 prefix length. By itself, this could be a coincidence, but because they appear quite often it is decided to label these hijacks to allow further investigation. To avoid leaving out other prefixes with a similar pattern, hijacks of prefixes with three or more equal octets are marked with this label.

5.3.4. Labels based on relations between hijacks

The previous subsection explained all labels that are given to hijacks based on the hijack features. These labels, basic hijack features such as the detected AS and AS path, and the relations between hijacks are used to analyse groups of related hijacks. It is found that using the relation between hijacks makes it easier to determine if the hijacks are legitimate announcements, intentional, or caused by misconfigurations. Based on the relations some of the hijacks are labelled. The process is discussed more in subsection 5.4.2 and section 6.2.

5.4. Generating results

The goal of this thesis is to give an overview of the behaviour and characteristics of possible hijacks that happened between 20 May 2018 and 31 May 2019 and were detected by BGPStream. To accomplish this goal much data has been collected and computed to help grouping and labelling the hijacks. This process followed three steps: generating features, finding relations between hijacks and labelling hijacks. The results are divided into the same categories: results based on features to give an overview of the hijacks data set, results based on the relations between hijacks, and results based on the labels. The three subsections explain why these results are important and how they are generated.

5.4.1. Overview of the hijacks data set

The results discussed in this subsection are results that are based on the hijack features only. These results help to give an overview of the ASes and prefixes that were involved in the hijacks and also to get an idea of some basic characteristics of the hijacks. The results are divided into five parts: prefix statistics, hijack statistics, AS statistics, AS path statistics, and the detection of hijacks by BGPStream. This subsection discusses how and why these results were generated. The results can be found in the next chapter.

Prefix statistics The first results in this section are based on the prefixes that were involved in the hijacks. They give an overview of the distribution of IP versions and prefix lengths and show how this relates to the global distribution. First, the distribution of IP versions is compared to the global distribution. This shows whether or not there is any preference for an IP version when hijacking a prefix. The IP version of the detected advertisement and expected prefix of a hijack are always equal because the detected advertisement is either equal to or a subnet of the expected prefix. The feature ‘IP version’ is used to show the distribution. The results are plotted in a bar chart.

When an AS announces one of its prefixes or when it hijacks a prefix, the prefix is announced with a specific length. Comparing the distribution of prefix lengths in the hijacks data set and in BGP globally can show if there is any significant difference between the two that should be further investigated.

Strowes [49] investigated the IPv4 and IPv6 prefix lengths that were visible in BGP in 2011 and 2019. The results of 2019 are compared with the prefix lengths that are visible in the hijack data set. The prefix length for each hijack is taken from the detected advertisement. This is because the detected advertisement is the prefix that is announced by the hijacker. The expected prefix is the prefix that is normally announced by the owner. Taking the prefix length of the expected prefix would thus not give an overview of the prefix lengths that are used when hijacking. The prefix lengths are plotted in a bar chart to show the distribution. There is one chart per IP version because IPv4 and IPv6 use different prefix lengths.

Hijack statistics The second category of results are some statistics about the hijacks itself. This includes the number of global and local hijacks, the hijack duration, and the relation between the detected AS and expected AS for each hijack. The number of global and local hijacks can be found by looking if the detected advertisement of a hijack is a subnet of the expected prefix. If this is the case, the hijack is a global hijack. In the other case, where the detected advertisement is equal to the expected prefix, the hijack is local.

Second is the duration of hijacks. Hijacks that are very short-lived are often caused by mistakes and are quickly withdrawn. A hijack going on for a long time can have different reasons, but it may indicate that the hijack is actually a legitimate announcement. The duration of hijacks was calculated using the prefix history of the detected advertisement and the AS history of the detected AS. In many cases, the hijack duration was less than a day and did not show up in the data thus giving a duration of 0 seconds. In the other cases, the result was two values that were often different. There is no clear reason for this difference but it may be caused by delay or failure when RIPE collected the data. [27] The maximum of these two values is taken as the hijack duration. It is chosen to take the maximum value after analysing the hijack durations. In some cases, one of the two durations is 0, while the other is a large number. In other cases, there is just a difference between the two. Taking the maximum value gives the fairest results. The distribution is plotted in a histogram.

The last category of results in this subsection considers the relation between the detected AS and the expected AS of each hijack. There can either be a direct link between the two, a link because one is in the customer cone of the other, or no relation at all. A hijack that has a direct relation between the detected AS and the expected AS is often caused by different reasons than a hijack that happens between two completely unrelated ASes. This information should give more insight into how many hijacks are caused by a related AS. The cause of hijacks is discussed in more depth in the label-based results.

AS statistics The AS statistics should give an overview of the ASes that were involved in the hijacks. This category of results includes the size of the customer cones of the involved ASes and how this relates to the global distribution, the distribution of ASes over the RIRs, and the countries in which the ASes are located.

The size of the customer cone of an AS gives some information about the position in the global AS topology. How well an AS is connected may impact how many ASes will receive the announcement and how often a hijack is detected. ASes without customers have no customer cone and are called Tier 3 ASes. These ASes are dependent on their providers for connectivity. On the contrary, Tier 1 ASes have very large customer cones. They provide transit for other ASes and are very well connected. The distribution of the sizes of the customer cones of the ASes involved in the hijacks can give insight into whether small or large ASes are more prone to be hijacking or being hijacked. This distribution is plotted in a histogram that contains the distribution for detected ASes and expected ASes separately.

A second histogram shows the global distribution of the AS customer cone sizes. This data is taken from the customer cone files from May 2018 and May 2019 of the CAIDA AS Relationships data set. The files are chosen because they are the first and last month of the year of data. They only have to give a general overview of the distribution. Using both files also shows if the sizes change much over the year.

After looking at the AS customer cone sizes, the distribution of the ASes over RIRs is plotted. The

detected ASes and expected ASes are plotted separately. This gives an idea of which RIR most detected or expected ASes belong to. This is compared to the total number of ASes per RIR globally. The total number of ASes per RIR is taken from the RIR statistics file from 30 November 2018. November 2018 is chosen because it is the month in the middle of the year of data. These numbers do not change much over the year, so taking the data from November 2018 is sufficient. Comparing the distribution of ASes per RIR in the hijacks versus the global distribution shows if there is any RIR that has a disproportionally small or large number of detected or expected ASes.

The ASes in the hijacks are not all assigned to a RIR, some are unallocated or reserved by IANA. These ASes are shown in the first plot under the category IANA. The comparison to the number of ASes per RIR globally does not contain IANA because the number of unallocated and reserved ASes is very large as these contain all the ASes that are still available for use in the future. It is also not proportional in any way to the ASes that are involved in the hijacks, because in theory they should all be assigned to a RIR before they can be used. Hijacks that have an AS that is unallocated or reserved by IANA form a group that is discussed in section 6.3.

In addition to looking at the distribution of the ASes over RIRs, a table is made that shows how many hijacks are caused by an AS in one RIR and target an AS in another. This table helps to see if ASes from a certain RIR are mainly targetting ASes in the same or another RIR, or if there is an even distribution without a clear preference.

The last type of plot made in this category shows how many detected ASes and expected ASes are in each country. The number of ASes per country is plotted on a world map. There is a separate map for detected and expected ASes. These maps show if a country contains ASes that cause many hijacks or are the target of many hijacks. The numbers are discussed and compared with the total number of ASes per country. Two additional plots are made that show the number of detected or expected ASes per country divided by the total number of ASes in that country. This helps to see if there is any country with a disproportionally large number of detected or expected ASes.

The AS country feature, which is discussed in subsection 5.1.1, is a tuple that contains the countries found in three different data sets. For these maps, the most common country is used. If there is not one most common country, the first country in the list is chosen because the GeoLite2 result is more recent than the result from the RIR statistics. It also does not often differ from the IP2Location LITE result. ASes that have no country are not included in the maps.

AS path statistics The fourth category of results is based on the AS path. It is discussed whether or not the paths are continuous and valley-free, how many of the paths are prepended, and what is the distribution of the path lengths.

It is important to know if the AS path is continuous because when it is not there is a gap in the path that stops traffic from being forwarded to its destination. This can be caused by incorrect data, but it can also indicate a path hijack. The percentage of paths that is continuous is calculated using the ‘continuous path’ feature that was discussed in subsection 5.1.3.

After looking at how many paths are continuous, the number of valley-free paths can be calculated. Continuous paths are per definition not valley-free because a valley-free path does not have a gap. Valley-free paths follow the routing policies that are inherent to the basic AS relationships. In 2012 Giotsas and Zhou [37] investigated the violation of the valley-free property. They found that a path that is not valley-free can be caused by a misconfiguration, but is in up to 50% of the cases a result of ASes that use complex routing policies. The percentage of valley-free paths in the hijacks is calculated separately for IPv4 and IPv6, and compared with the results from the paper by Giotsas to see if there is a significant difference.

The next property of AS paths that is investigated is the path length. The path length is defined by the number of ASNs in the path. In the route selection process, shorter routes are preferred over longer routes. When hijacking a prefix with the intent to have other ASes use your route, it is thus

preferable to keep the AS path as short as possible. The AS path length is plotted in a histogram with a separate distribution for IPv4 and IPv6 hijacks. This is done so the path lengths can be compared to the average path length in BGP in 2018.

Each AS in the path can influence the path length by prepending its ASN multiple times. The origin AS would only do this to decrease the chance of the route being chosen. When an AS hijacks a prefix and prepends the AS path, it thus goes directly against the principle of keeping the path as short as possible so it is more likely to be used by other ASes. To see how many of the AS paths are prepended and if this is the cause of longer AS paths, the distribution of path prepending by the origin AS is plotted. The plot has a separate distribution for IPv4 and IPv6. To see the direct relation between the path length and path prepending by the origin AS a scatter plot is made. This plot shows the points for IPv4 and IPv6 separately.

Because every AS in the path can use path prepending to increase the path length, another plot is made shows the distribution of the number of unique ASes in the path. If this distribution is similar to the distribution of the AS path length, the longer paths are not necessarily caused by path prepending. It also gives insight into whether paths are more often prepended by the origin AS or by other ASes in the path.

Detection of hijacks by BGPStream To detect possible origin hijacks BGPStream uses monitors that BGPMon has placed around the world. These monitors are connected to BGP peers who send them BGP updates for prefixes they are originating and for routes that they have received from their own peers. The monitors then determine if a route is possibly anomalous. For each hijack, the number of BGPMon peers that received the route is given. Based on this number three graphs are made. These graphs may give some insight into how well the monitors cover BGP globally. This can help to give an indication of whether or not most of the hijacks that are happening are also detected.

The first graph that is made is a histogram of the number of BGPMon peers that received the route for a hijack. This histogram contains the data for global hijacks and local hijacks separately. Local hijacks are often not propagated through a large part of the BGP topology and it is interesting to see if many monitors can detect local hijacks. At the same time, the graph also shows if global hijacks are always detected by a large number of monitors or not.

To give further insight into the coverage of BGPMon, a second graph is created that shows the relation between the customer cone size of the detected AS and the number of BGPMon peers that receive the route of the hijack. How well an AS with a small or non-existent customer cone is connected is dependent on its provider. It could be more difficult to detect hijacks from ASes that are not as well connected because their routes may not reach BGPMon peers. This graph should show if there is such a relation. It should be noted that this figure only shows data for hijacks that have reached BGPMon. There may exist many hijacks that have not reached the monitors, and will thus not be detected.

The third graph shows the relation between the AS path length and the number of BGPMon peers that received the route. The AS path length is the number of ASNs in the path. If the AS path length has an influence on the number of BGPMon peers that receive the route, it should be related to the number of ASes the route passes from the AS that originated the route to the BGPMon peer that received the route. ASes can prepend the path with their ASN to increase the path length. This affects the path length, but it cannot affect the number of BGPMon peers that receive that route because it does not increase the number of ASes that receive it. For this reason, to show the real influence of the number of ASes in the path, the number of unique ASNs in the path is used instead of the AS path length.

5.4.2. Results based on relations between hijacks

In section 5.2 was explained how hijacks can be related and how these relations are detected. The second category of results are results based on these relations. In total there are five types of relations detected between hijacks. The first type is the relation based on the detected AS or expected AS. Hijacks that have one of these in common are related. Similarly, the second type of relation is based

on the detected advertisement or expected prefix. The remaining three types are a bit more complex. Hijacks can also be related because they are part of the same event. This means they are caused by the same AS within a period of two hours. Another way in which hijacks can be related is when prefixes are hijacked by the same set of ASes. Lastly, hijacks are said to be related when a prefix is hijacked by an AS that earlier was or later becomes the expected AS for that prefix. In other words, the detected AS becomes the expected AS or vice versa.

Each of these relations are used to group hijacks. Hijacks are given several group IDs based on these relations. When multiple hijacks share a group ID for a type of relation, the hijacks are related. For each type of relation, the distribution of the group sizes is plotted. Hijacks that belong to a group of related hijacks are manually inspected to see if they can be labelled using this relation and the labels that were given to each hijack. A part of the hijacks will have a definite label after this step.

5.4.3. Results based on the hijack labels

The previous subsection discussed how part of the hijacks will get a definite label based on their relation to other hijacks and the labels that were given to them in the labelling process. Some hijacks were already given a definite label based on their features only. This includes, for example, the invalid hijacks, the hijacks that are likely legitimate, and the hijacks that belong to a category such as 'hijacking a customer route'. For other hijacks, the cause is not clear.

The last section of the results gives an overview of the distribution of the labels that were given to each hijack. Manually analysing groups of hijacks with the same label should give more insight into whether these labels are useful to detect the cause of hijacks. Each label is discussed separately and the section ends with a summary and a discussion on which labels are most informative and which labels did not provide any insight into the cause of origin hijacks.

6 Results

This chapter gives an overview of the behaviour and characteristics of possible hijacks that happened between 20 May 2018 and 31 May 2019 and were detected by BGPStream. The previous chapter explained how the necessary data was collected and computed, how relations were found between hijacks, and how the hijacks were labelled. This chapter presents the results of these steps. The structure followed is similar to the structure in the previous chapter. Section 6.1 gives an overview of the hijacks data set. Section 6.2 discusses the relations between hijacks and how the relations are used in labelling hijacks. Lastly, section 6.3 contains the results based on how the hijacks were labelled. Each section in this chapter ends with a summary that lists the most important outcomes.

6.1. Overview of the hijacks data set

This section gives an overview of the ASes and prefixes involved in the hijacks and discusses some basic characteristics of the data set. It contains results that are generated using features that were explained in section 5.1. How and why these results are generated can be found in subsection 5.4.1. This section is divided into six subsections. The first subsection discusses the prefixes that were involved in the hijacks. The second subsection contains basic statistics about the hijacks such as the duration and whether hijacks are global or local. The third discusses the ASes that were involved. This is followed by the subsection that contains characteristics of the AS paths. The fifth subsection discusses the detection of hijacks by BGPStream. This section is ended by a summary that lists the most important outcomes.

6.1.1. Prefix statistics

The prefix statistics are based on the IP prefixes that were involved in the hijacks. The set of origin hijacks analysed in this thesis contains 2665 possible origin hijacks in total. Each hijack has a detected advertisement and an expected prefix. The detected advertisement is the prefix that is announced in the hijack and the expected prefix is the hijacked prefix. The detected advertisement can be equal to the expected prefix (local hijack) or a subnet of the expected prefix (global hijack). The IP version of the detected advertisement and the expected prefix is always equal.

The 2665 hijacks affect 2246 unique expected prefixes of which 2134 (95%) are IPv4 and 112 are IPv6. In total there are 2450 hijacks of IPv4 prefixes and 215 hijacks of IPv6 prefixes. In other words, 91.9% of the hijacks are hijacks of an IPv4 prefix. This may seem unbalanced, but this distribution is comparable to the distribution of IPv4 and IPv6 prefixes that are currently in use globally. The global distribution has approximately 90% IPv4 prefixes and 10% IPv6 prefixes and has been fairly consistent over the past years [24]. This means that in terms of hijacks there does not seem to be a clear preference for any of the two IP versions.

Aside from an IP version, every detected advertisement has a prefix length. Strowes [49] investigated the IPv4 and IPv6 prefix lengths that were visible in BGP in 2011 and 2019. A similar approach is taken for the prefix lengths of all detected advertisements. Using this data one can find if there is a difference

between the length of prefixes that are usually announced globally, and the length of prefixes that are announced in hijacks. Figure 6.1 contains an overview of the IPv4 prefix lengths in the hijacks and figure 6.2 shows the global distribution from January 2019. Note that the y-axis in figure 6.1 has a logarithmic scale. This distribution is mostly similar to the distribution of all IPv4 prefix lengths in 2019, except for one big difference. The hijacks contain a large group of 302 detected advertisements with a prefix length of 32. These are hijacks of one specific IP address. This is interesting because this group forms 11.33% of the data set, whereas announcements of one IP address is globally very uncommon. This group of hijacks is therefore further analysed in section 6.3.

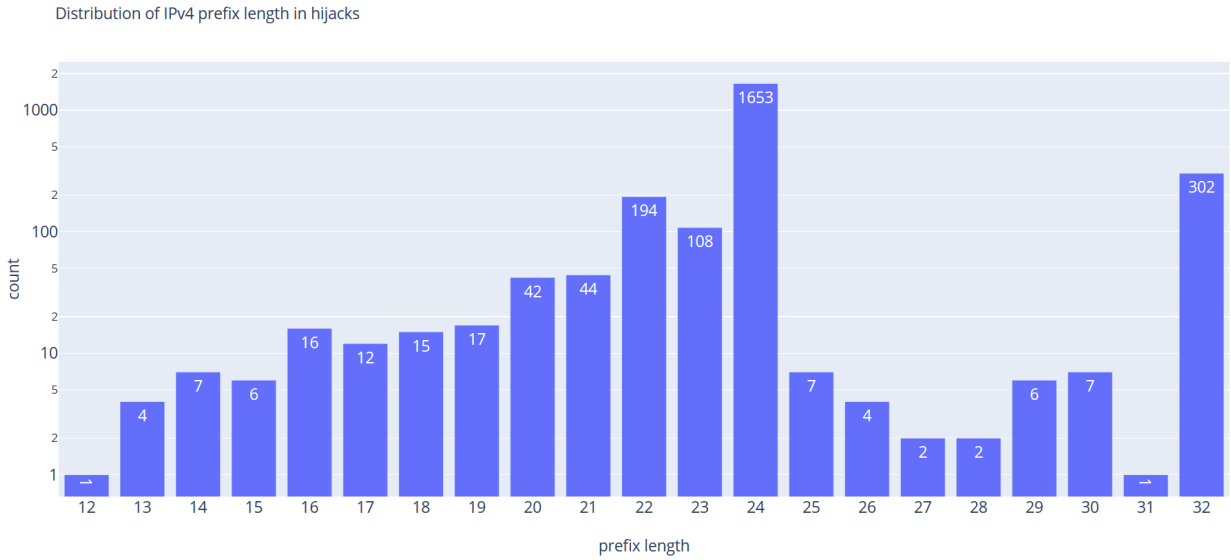


Figure 6.1: Distribution of IPv4 prefix length in hijacks

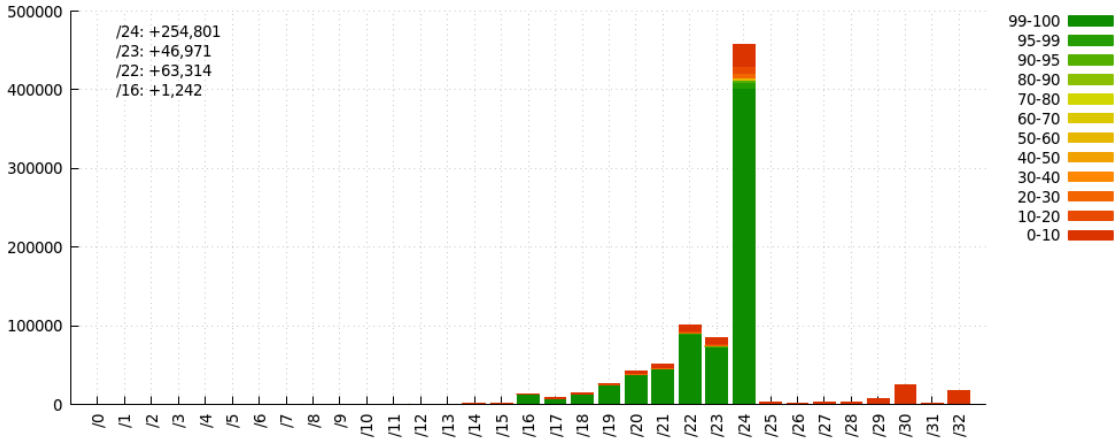


Figure 6.2: Distribution of IPv4 prefix length globally in January 2019 [49]

Figure 6.3 contains the distribution of the IPv6 prefix lengths in the hijacks and figure 6.4 shows the global distribution from January 2019. Although IPv6 prefixes represent only 8% of the data set, the distribution of IPv6 prefix lengths in the hijacks is very similar to the global distribution. It is also interesting to see that there are no hijacks of one IP address in IPv6, contrary to IPv4 where they are very common.

To summarise, these prefix statistics have shown that there is no clear preference for IPv4 or IPv6 when hijacking prefixes. In addition, the distribution of prefix lengths of the detected advertisements is similar to the global distribution of prefix lengths for both IPv4 and IPv6, except for the large group of hijacks that target a single IPv4 address. This group is discussed in section 6.3.

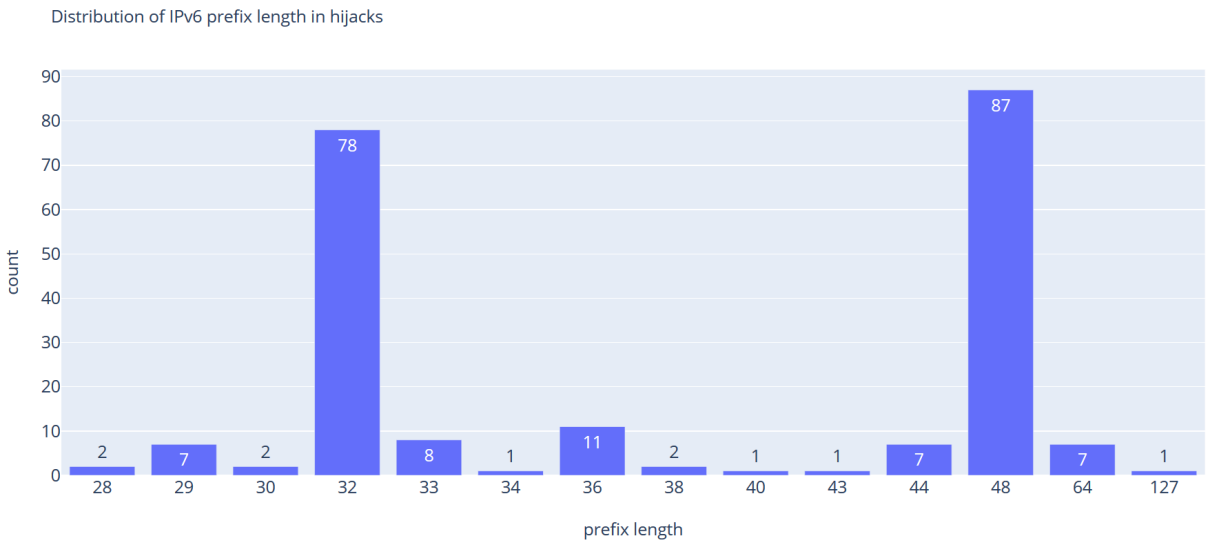


Figure 6.3: Distribution of IPv6 prefix length in hijacks

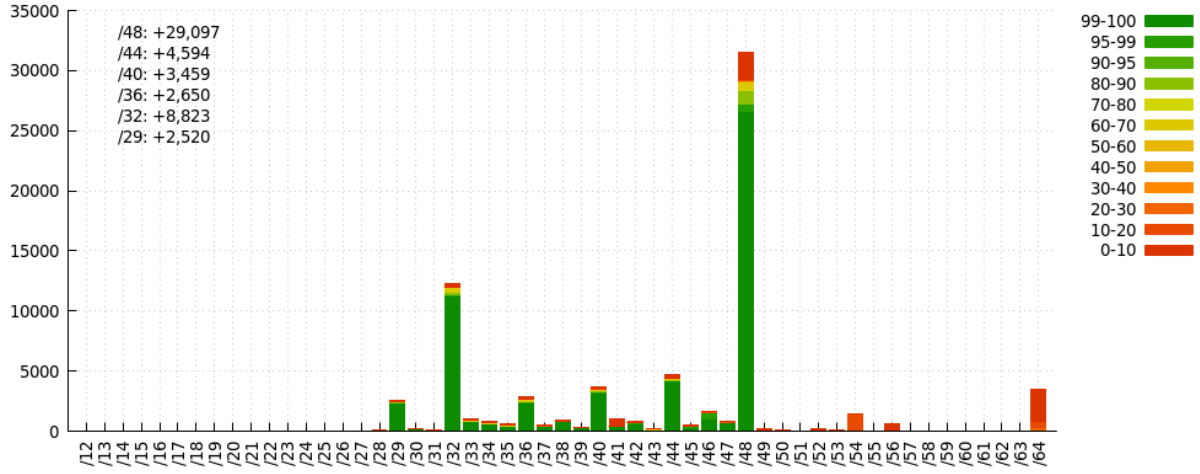


Figure 6.4: Distribution of IPv6 prefix length globally in January 2019 [49]

6.1.2. Hijack statistics

This subsection contains some statistics about the hijacks themselves. This includes the hijack duration, whether they are global or local, and if they have an origin that is related to the legitimate owner of the prefix. In total, the data set contains 2665 possible origin hijacks. There are 1386 (52%) local hijacks and 1279 (48%) global hijacks. They thus seem to happen equally likely.

The duration of the hijacks varies. There are 1529 (57.37%) hijacks that are very short-lived and don't show up in the AS history or prefix history. They received a duration of 0 seconds, but since the minimum duration in the data is 86400 seconds (1 day) they may last up to a day. The quick withdrawal of these announcements could indicate that they were the result of a mistake, but this cannot be said with certainty. This is discussed more thoroughly in section 6.3. The other 1136 hijacks last longer. The duration ranges from 86400 seconds (1 day) to 33004800 seconds (382 days). Figure 6.5 contains the distribution of the hijack duration in days. A large number of hijacks last under 10 days, but there is also quite a number that goes on for a longer period. Whether these long-lasting hijacks are actual hijacks or legitimate announcements is discussed in section 6.3.

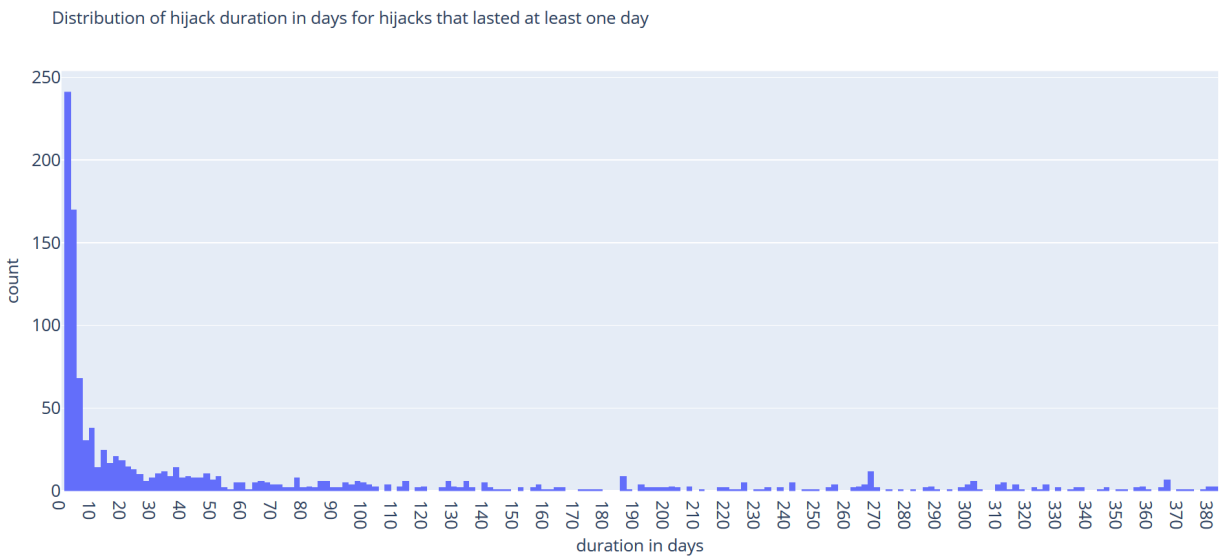


Figure 6.5: Hijack duration in days for hijacks that lasted at least one day

The last category of results in this subsection considers whether hijacks are caused by an AS that is related to the expected AS. There can either be a direct link between the two, they can be in each other's customer cone, or they are not related in any way. Out of the 2665 hijacks, 2456 (92.16%) have no direct relation between the detected AS and the expected AS. From these hijacks 242 (9.08% of the complete set) do have a relation between the two ASes because one is in the customer cone of the other. For the other 2214 (83%) hijacks there is no relation at all. These hijacks cannot be caused by any mistake that is related to the relation between ASes. One example of such a mistake is an AS unintentionally announcing a route of its customer.

The group of hijacks that have a direct link between the detected AS and the expected AS consists of 209 (7.84% of the complete set) hijacks. In this group 61 (2.29%) hijacks have a p2p link. The other 149 (5.59%) hijacks have a p2c (122 hijacks) or c2p (27 hijacks) link from the detected to the expected AS. There is one hijack that has a p2c link in the month of the hijack and a p2p link in the month after. All ASes that cause a hijack in this group do so by announcing a prefix that belongs to an AS it has a direct relation with. If these hijacks are caused by misconfigurations or something else is examined later.

These hijack statistics show that global and local hijacks happen equally likely. Most hijacks last less than 10 days, but there is still a rather large number that lasts a lot longer. The characteristics of short-lived and long-lived hijacks are discussed later in this chapter. Furthermore, 83% of hijacks are caused by an AS that is not related to the expected AS. It was discussed that this eliminates several causes for these hijacks. Hijacks that are caused by a related AS, are most often caused by a provider or an AS that is not directly related to the expected AS but related via customer cone. Whether these hijacks all have a similar cause is examined in section 6.3.

6.1.3. AS statistics

This subsection focusses on the ASes that were involved in the hijacks. It discusses the size of the customer cones of the ASes, the distribution over the RIRs, and the countries where these ASes are located. Note that the plots in this section include all hijacks and some hijacks are caused by the same AS. This means that when a plot shows the number of detected ASes, it means the number of hijacks caused and not the number of unique ASes. The consequence of this will be explained for each plot separately.

The distribution of the sizes of the AS customer cones is plotted in figure 6.6. Figure 6.7 contains the global distribution of customer cone sizes in May 2018 and May 2019. Note that the y-axis has

a logarithmic scale in both figures. Comparing the two figures shows that most ASes involved in the hijacks and most ASes globally have a small customer cone. In the global distribution, 8 ASes have a customer cone with around 15000 or more ASes. They appear a lot more often in the hijack distribution. This can be explained because some ASes hijack or are hijacked multiple times. The AS with 25k ASes in its customer cone that only appears once in the global distribution but multiple times in the hijack distribution must, therefore, be causing multiple hijacks.

In general, ASes with a large customer cone seem to be relatively more often involved in hijacks than ASes with a small customer cone. ASes with a large customer cone also cause more hijacks than that they are targeted in hijacks. This is interesting because ASes with a very large customer cone are often Tier 1 ASes that are operated by large telecom companies. Although it can be difficult to properly configure a large AS, these companies generally should have the resources and experience available to avoid misconfigurations. However, these ASes often do more than simply announcing prefixes. Because of the complexity of their operations it is more likely they make mistakes, and because they are well connected this can easily affect one or more other ASes. It is thus not necessarily strange that they show up so often.

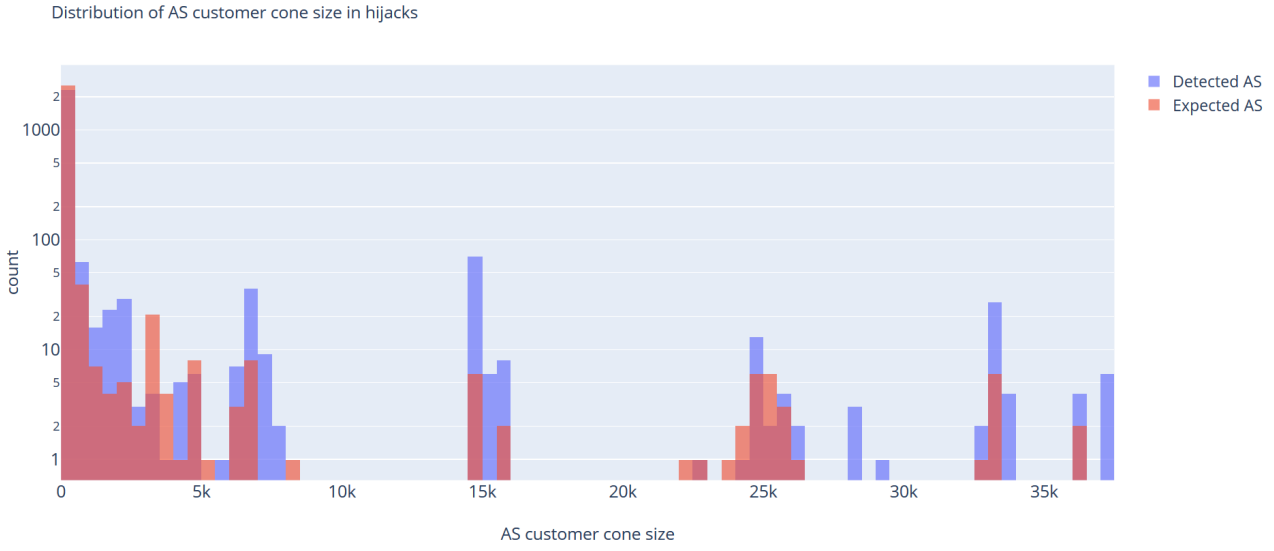


Figure 6.6: Distribution of AS customer cone size in hijacks

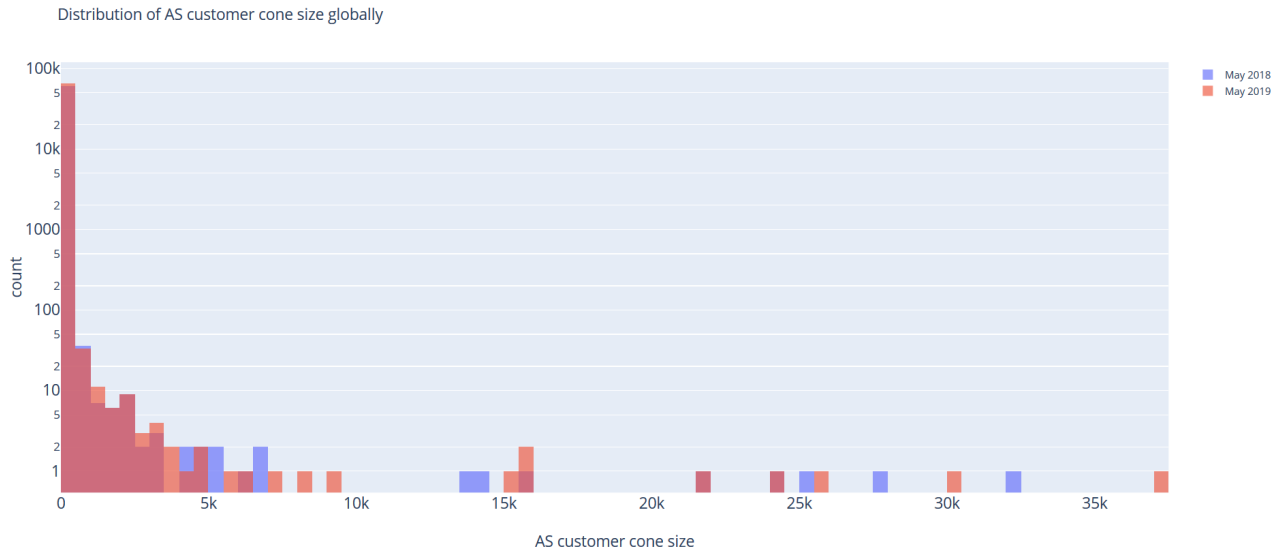


Figure 6.7: Distribution of AS customer cone size globally

The next property of ASes that is considered in this section is the RIR they are assigned to. The distribution of ASes over RIRs in hijacks is compared to the distribution of ASes over RIRs globally. Figure 6.8 contains the distribution in hijacks. Table 6.1 contains the number of detected and expected ASes per RIR, the number of ASes per RIR globally, and the ratio between these numbers. IANA is not included in the table because it is a separate case. This is discussed later.

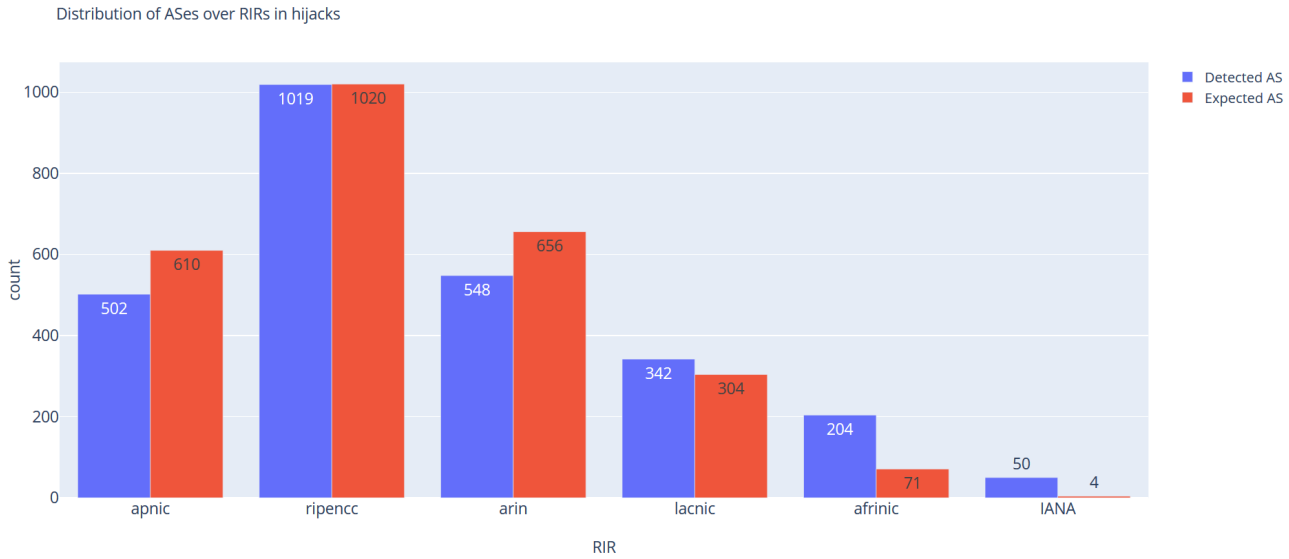


Figure 6.8: Number of hijacks caused by ASes in each RIR

RIR	Detected	Expected	Global	Detected/Expected	Detected/Global	Expected/Global
apnic	502	610	9323	0.823	0.054	0.065
ripencc	1019	1020	39447	0.999	0.026	0.026
arin	548	656	27222	0.835	0.020	0.024
lacnic	342	304	9190	1.125	0.037	0.033
afrinic	204	71	2302	2.873	0.089	0.031

Table 6.1: Number of ASes per RIR and ratio hijacks to global

One AS can cause multiple hijacks and may thus be listed as detected AS multiple times, the plot shows the number of hijacks caused by an AS in a specific RIR. The distributions of the hijacks and the global distribution are relatively similar, but APNIC and AFRINIC have a relatively large number of hijacks being caused by their ASes and APNIC is relatively often targeted.

The distribution of detected and expected ASes is similar except for AFRINIC, which has a rather high number of hijacks being caused by one or more of its ASes. The reason for this is not clear based on this data only, but it will become clear in the remaining sections of this chapter. IANA has more detected ASes than expected ASes. This is expected because these ASes are unallocated or reserved. Because they are in theory not supposed to be announcing prefixes, they should not have prefixes that can be hijacked. The fact that they are showing up as the cause of hijacks can be the result of other ASes using their ASN as the origin in a path hijack.

Table 6.2 is created to show if ASes target prefixes from ASes in their own RIR more often than from ASes outside of their RIR. The table shows that this really differs between ASes. For example, ASes from AFRINIC often target ASes from ARIN and RIPE but less often ASes from other RIRs, whereas ASes from RIPE in more than half the cases target other ASes from RIPE. A possible reason for this could be that hijacks that happen between ASes from RIPE are caused more often by related ASes than other hijacks, but without further analysis this is only guessing. It is thus looked into after labelling all hijacks, but it is found that hijacks between related ASes do not happen more often for hijacks with both ASes in RIPE. In general, it is difficult to say if ASes have a certain preference or if it is just coinci-

dence. It is likely that ASes from RIPE, ARIN and APNIC are more often targeted because they have more ASes, but also because large ASes are more often part of these RIRs.

Detected | Expected	IANA	afrinic	apnic	arin	lacnic	ripencc
IANA	0.00%	2.00%	46.00%	32.00%	8.00%	12.00%
afrinic	0.00%	13.73%	7.35%	48.53%	3.43%	26.96%
apnic	0.40%	1.79%	45.42%	30.48%	2.79%	19.12%
arin	0.36%	2.18%	30.11%	19.89%	12.04%	35.40%
lacnic	0.00%	0.29%	9.36%	27.78%	50.29%	12.28%
ripencc	0.00%	1.96%	14.42%	18.06%	4.24%	61.53%

Table 6.2: Number of detected ASes hijacking expected ASes per RIR

The last figures in this subsection show the number of hijacks being caused by or targeting an AS in each country. Figure 6.9 contains the number of hijacks being caused per country and figure 6.10 contains the number of detected ASes per country divided by the total number of ASes in that country. What is immediately clear from these pictures is that ASes from the USA are causing a lot more hijacks than ASes from other countries. This is however not necessarily strange when looking at the global distributions of ASNs. The USA owns the largest number of ASNs in the world, more than three times as much as Brazil, the country that comes second [29]. This is supported by the second map. The USA does not stand out there. Other countries that stand out in the first map are Poland, Brazil, and Angola. Both Poland and Brazil have a large number of ASes, which partly explains it, but Angola does not. Angola is also the only country that stands out in the second map The large number of hijacks caused in Angola also explains the large number of hijacks caused by ASes from AFRINIC. Whether this is the result of one or multiple ASes will be clear in the next section.

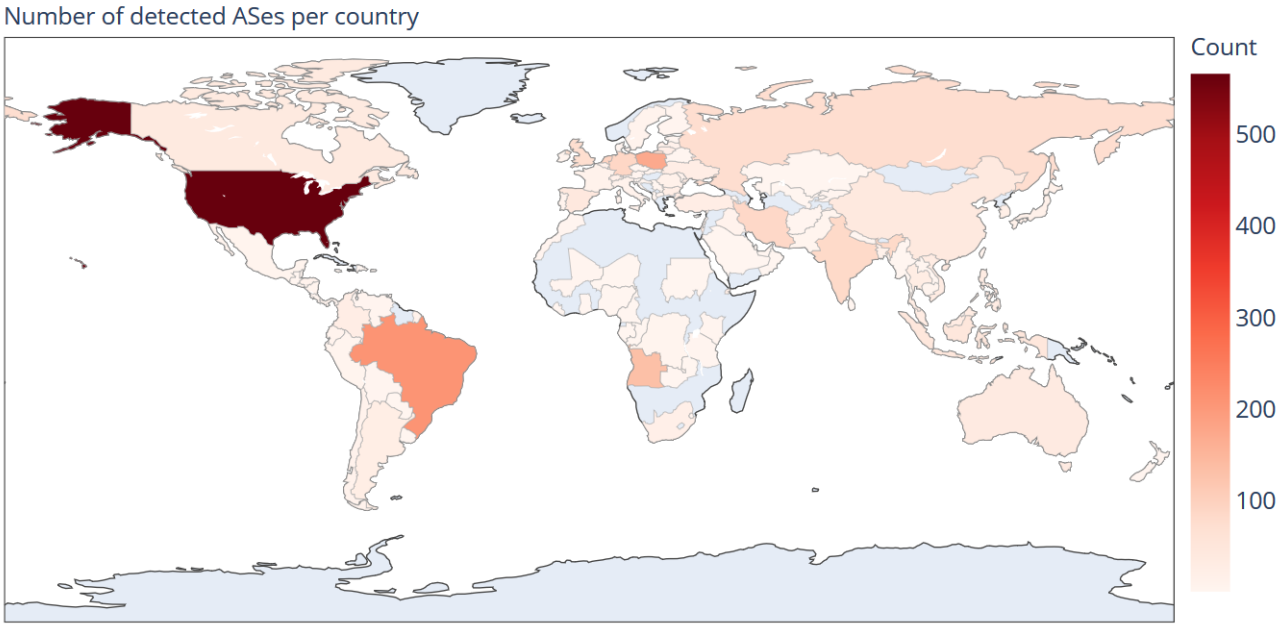


Figure 6.9: Number of detected ASes per country

Figure 6.10: Number of detected ASes per country normalised

Figure 6.11 and figure 6.12 are the last figures in this subsection. They show how many hijacks are targeting an AS in a specific country. Figure 6.11 looks similar to the figure 6.9, but Angola is never targeted. The number of ASes in this country is comparable to the countries surrounding it, so this is not unexpected. In addition, India and China are a lot more often the target of a hijack than that they are causing one. If there is, for example, one specific AS that is often targeted is looked into later. Figure 6.12 has several countries that contain a relatively high number of expected ASes. However, these countries have so few ASes compared to others that just one hijack may give this result. This picture thus does not show any significant results.

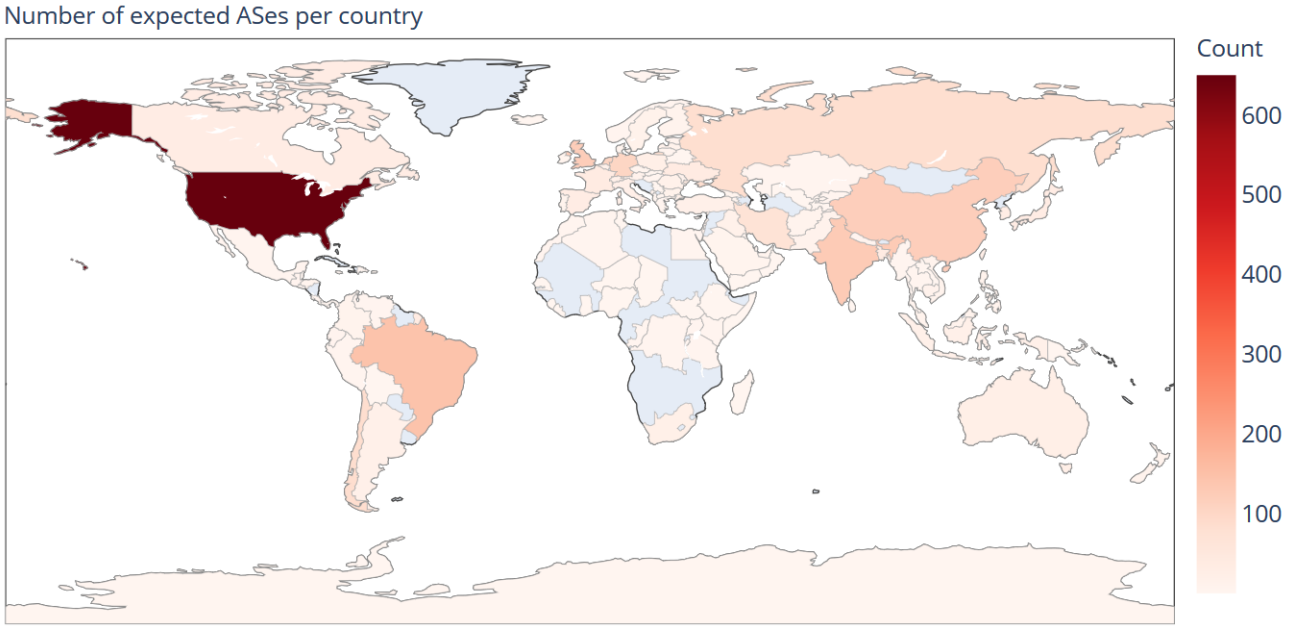


Figure 6.11: Number of expected ASes per country

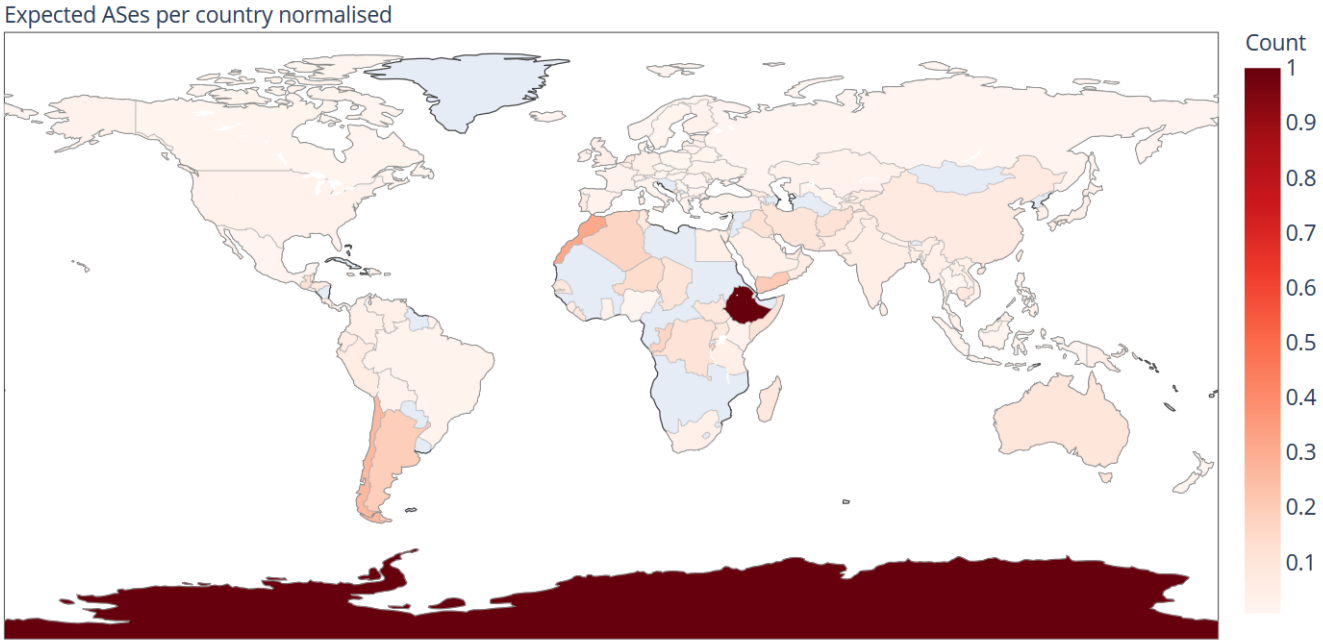


Figure 6.12: Number of expected ASes per country normalised

This subsection has not provided many unexpected results. Large ASes are relatively more often involved in hijacks, but because of their complicated operations, this is not unexpected. Most ASes in the data set are small, which is in agreement with the global distribution of AS customer cone sizes. When looking at the RIR and country of each AS it became clear that there are many hijacks caused in Angola. If this is the result of one AS becomes clear later in this chapter. Other countries or RIRs do not contain a disproportionally large number of detected or expected ASes.

6.1.4. AS path statistics

This subsection contains the results that are based on the AS path. The AS path comes with each hijack and is the path between the detected AS and one of BGPMon's peers. If, according to the CAIDA AS Relationship data set, the ASes in this path all have a direct relation to their neighbours in the path, the path is continuous. If there is one pair of neighbours that have no direct link between them, the path has a gap and is not continuous. This means that traffic can only be transferred to the AS before the gap, but not further. Traffic will thus not reach the origin. Since the AS path is normally created by ASes prepending their number to the path after they received an update message from their peers, a gap should in theory not be possible. It can be the result of incorrect data, but may also indicate a path hijack. In total there are 2060 (77.3%) continuous paths meaning that 605 (22.7%) AS paths have a gap. The percentage of continuous paths in hijacks of an IPv4 prefix and hijacks of an IPv6 prefix differs a lot. In total, 79.06% of AS paths from an IPv4 hijack are continuous. In IPv6 hijacks only 57.21% of the AS paths are continuous. Whether this is the result of path hijacks or something else is further investigated in section 6.3.

Continuous paths are checked to be valley-free. The valley-free property of paths is a theoretical concept. When a path is valley-free it adheres to routing policies that come with the basic relations between ASes. An AS path that is not valley free is a sign of a misconfiguration or more complex policies being used. In total 1919 paths are valley-free. This is 72.01% of all paths and 93.16% of continuous paths. The percentage of paths that are valley-free in IPv4 hijacks is 94.79% of continuous paths. For IPv6 hijacks only 67.48% of the continuous paths are valley-free. In 2012, Giotsas and Zhou [37] investigated the violation of the valley-free property in Internet routing. They found that up to 50% of valley-free violations are not the result of misconfiguration but of intended complex routing policies. Another finding was that IPv6 paths are disproportionally often not valley-free. The reason they give is low traffic volumes, complex configurations, and the central role of non-profit/governmental

organisations in IPv6 Internet. The large number of AS paths that are not valley-free in IPv6 hijacks is thus not unheard of.

When an AS learns multiple routes for a prefix it can choose a route based on preference values. If no routes are preferred, or when multiple routes have the same preference, the AS will choose the shortest path. The length of a path is defined by the number of ASNs in the path. Every AS that is part of the path can influence the path length by prepending its ASN multiple times. In January 2019, Huston [40] published an article on IPv4 and IPv6 metrics in BGP in 2018. This includes the average AS path length for IPv4 and IPv6. For IPv4 the average path length was 5.7, for IPv6 this was 4.7. Figure 6.13 shows the distribution of the length of AS paths. The figure shows that most of the paths for both IPv4 and IPv6 have indeed a length that is close to the average. However, some paths are much longer. To find out if this is the result of path prepending, the distribution of the number of times paths are prepended is plotted.

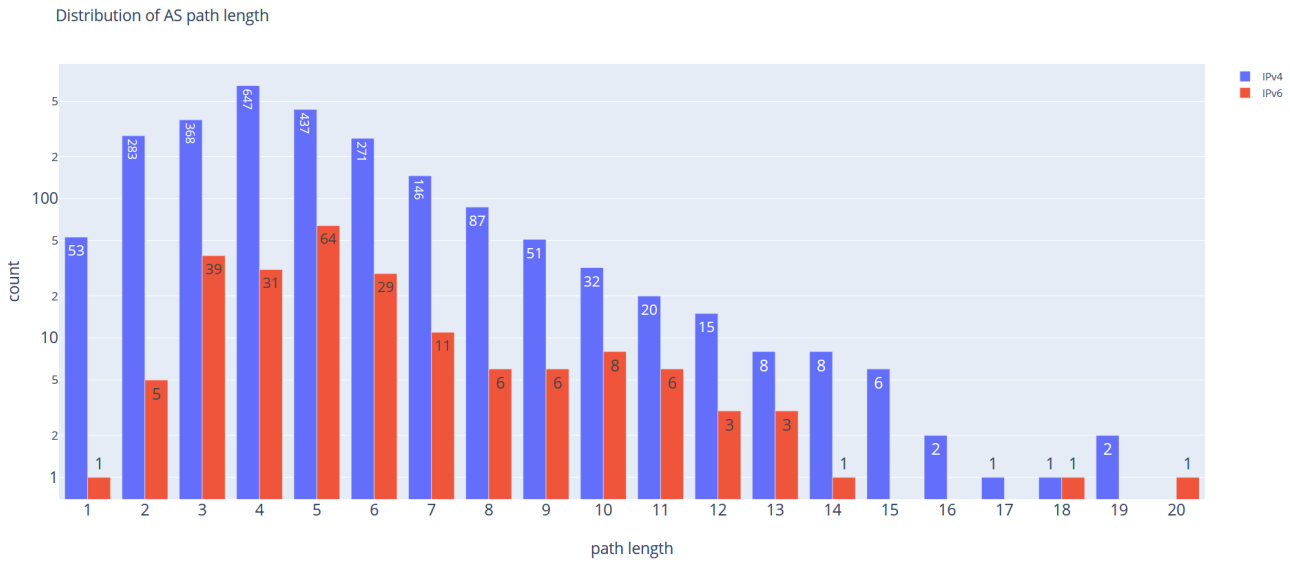


Figure 6.13: Distribution of AS path length

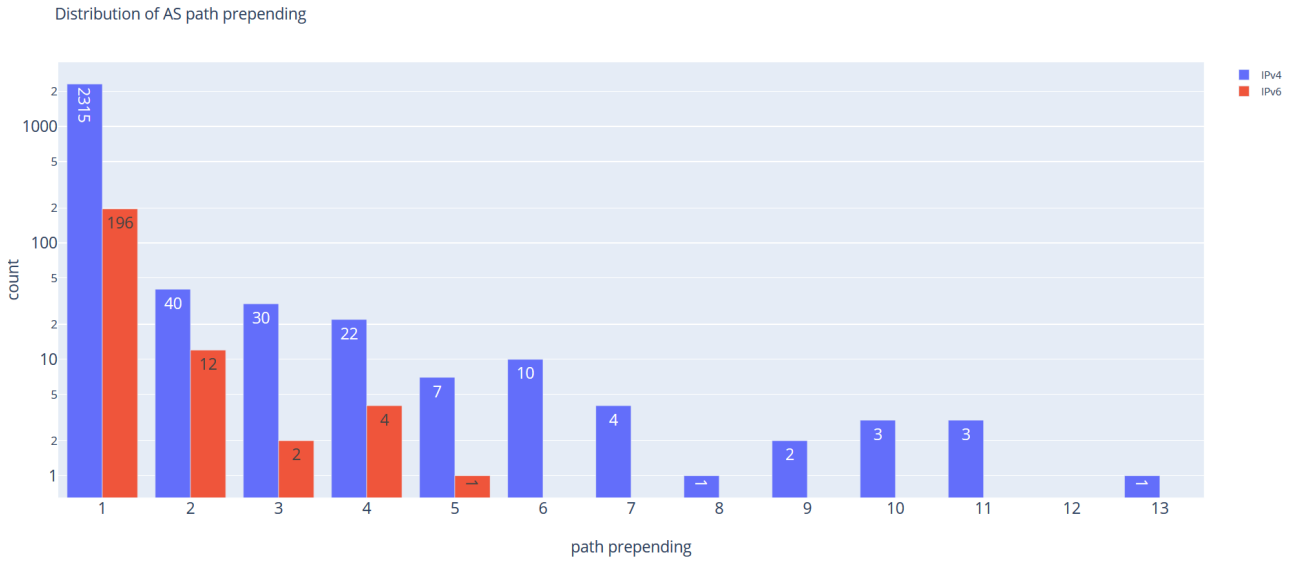


Figure 6.14: Path prepending by origin AS

Figure 6.14 shows how many paths are prepended by the origin AS and how many times the AS prepended its ASN to the path. From this graph, it is clear that most hijacks do not have an AS path

that is prepended multiple times by the origin AS. The ASN of the origin AS only appears once in these paths. Some paths are prepended many times. This reduces the likelihood that the route would be used. Either the AS originating the path has done this when announcing a hijack, or another AS intentionally changed the path. If the origin AS prepended its ASN multiple times, this was likely done to reduce the chance of the path being used. To determine if path prepending by the origin AS is the cause of the long AS paths figure 6.15 is created. It shows the relation between path prepending and path length. Long paths may be the result from path prepending, but this figure shows it is not necessarily the case. The paths that are not prepended range from lengths between 1 and 20. At the same time, paths that are prepended many times by the origin AS are not necessarily much longer than the number of times the origin ASN is prepended to the path.

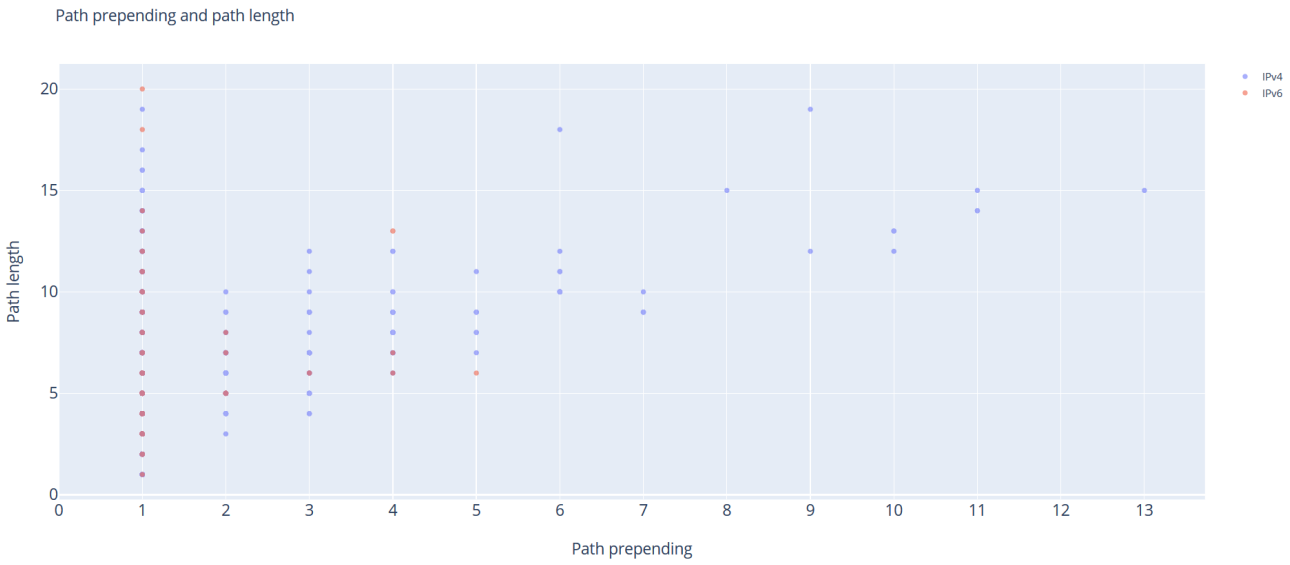


Figure 6.15: Path prepending and AS path length

It is also possible that another AS in the path prepended its ASN multiple times. Figure 6.16 shows the distribution of AS path lengths without any duplicates. The maximum path length is now 15 instead of 20, but there are still many paths that are a lot longer than average. Long paths are thus not always caused by path prepending. If these longer paths are used, it is either because of preference values set by ASes, or because of a lack of alternative paths.

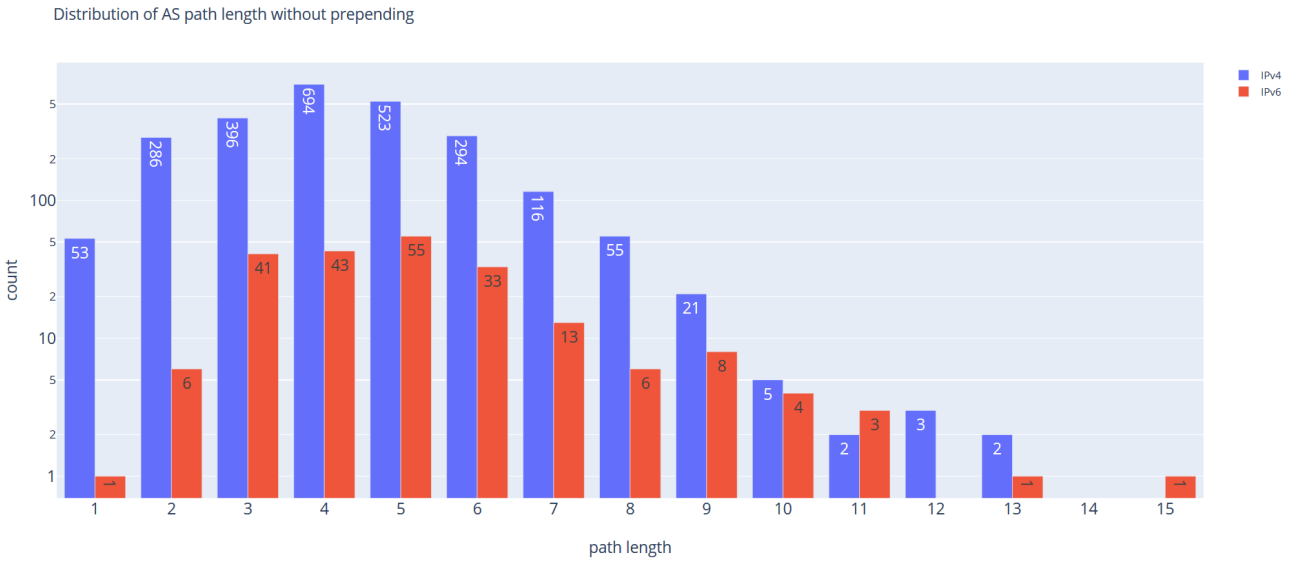


Figure 6.16: AS path length without prepending

This subsection focussed on AS path statistics. It was found that a large number of paths are not continuous, this may indicate that many hijacks in this data set are not origin hijacks but path hijacks. Continuous paths are in most cases also valley-free. IPv6 has a larger percentage of paths that are not valley-free but Giotsas and Zhou [37] found that this is due to low traffic volumes, complex configurations, and the role of non-profit/governmental organisations in the IPv6 Internet. Other properties discussed were the path length and path prepending. Most AS paths had a length close to the global average. Path prepending was not used often by the origin AS and it was also not always the cause of long AS paths. When an origin AS prepends the path multiple times, it reduces the likelihood the path is used, but it is also possible that another AS intentionally changed the AS path, making it look like the origin AS used path prepending. In general, there were no unexpected patterns and the path length and path prepending are thus not very informative with respect to origin hijacks.

6.1.5. Detection of hijacks by BGPStream

BGPStream detects hijacks by using monitors from BGPMon that are placed around the world. These monitors receive routes from their BGP peers and inspect them to determine if the routes are anomalous. This subsection contains three graphs that should give some insight into the global coverage of BGPMon's monitors.

Figure 6.17 contains the first graph. It shows the number of BGPMon peers that received the route from a global or local hijack. One can see in this graph that a large number of local hijacks (15%) reached the smallest possible number of peers. This may indicate that some hijacks will not be detected at all, but since there is no public information about BGPMon's monitor placement, this cannot be said with any certainty. The rest of the distribution is mostly similar for global and local hijacks. Both types of hijacks can reach many or few peers. The two other graphs may give more insight.

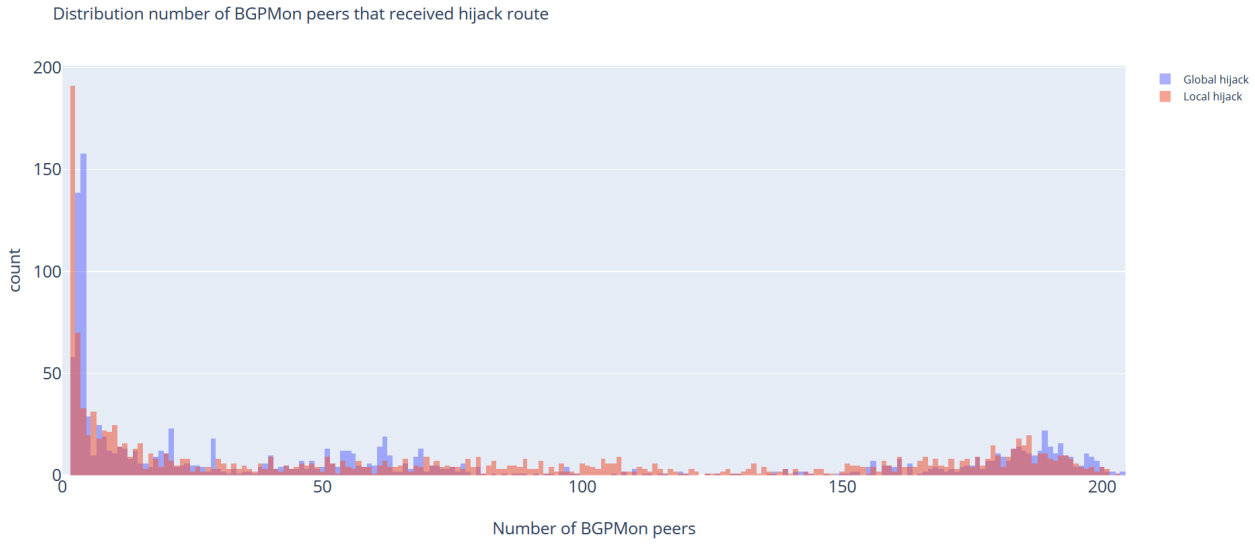


Figure 6.17: Number of BGPMon peers that received the hijacked route

The second graph, figure 6.18, shows the relation between the size of the AS customer cone of the detected AS and the number of BGPMon peers that received the announcement of the hijack. Note that subsection 6.1.3 showed how most ASes have a small customer cone. What can be seen in this graph is that there is no relation between the size of an AS and the number of peers that received the hijacked route. Both small and large ASes cause hijacks that are received by any number of BGPMon peers. This means that small ASes are not necessarily more difficult to monitor, but it cannot be said with certainty that this holds for all ASes, because again we only see data from hijacks that reached BGPMon.

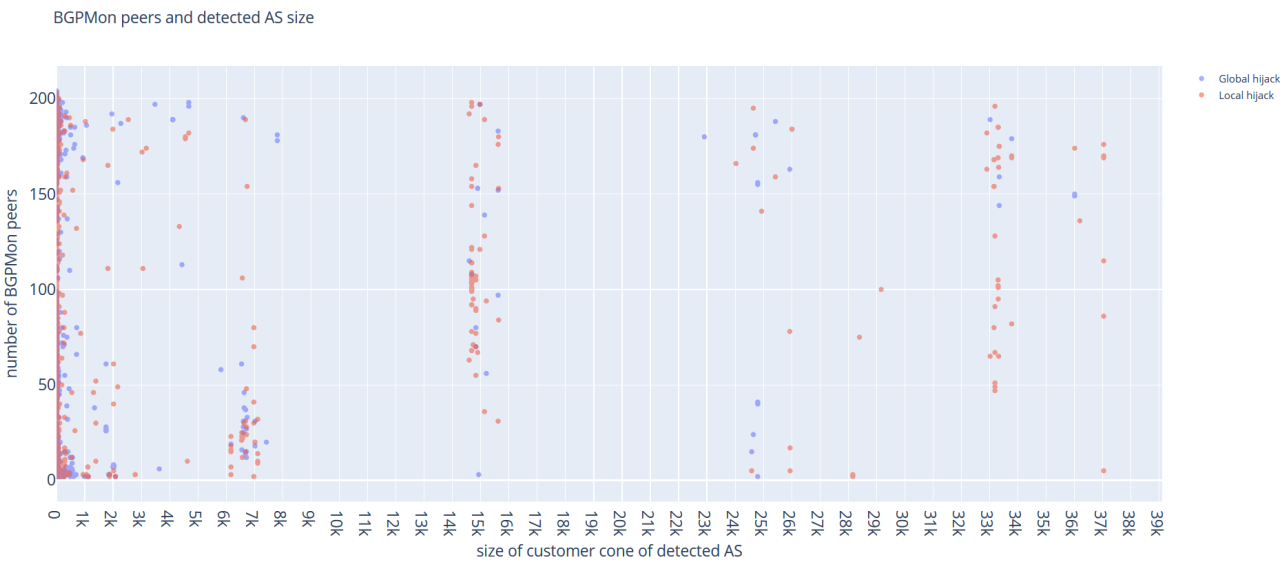


Figure 6.18: Size of AS customer cone versus number of BGPMon peers that received the hijacked route

The last graph that involves BGPMon can be found in figure 6.19. It shows the relation between the number of unique ASNs in the AS path and the number of BGPMon peers that received the route. Again there does not seem to be any clear relation. This means that the number of peers that receive the route does not seem to be based on AS size or AS path length nor on whether the hijack is global or local. Unfortunately, nothing can be said about the hijacks that are not detected by BGPMon as it would require a different approach to detecting origin hijacks, but the monitor coverage may be an interesting topic for further research.

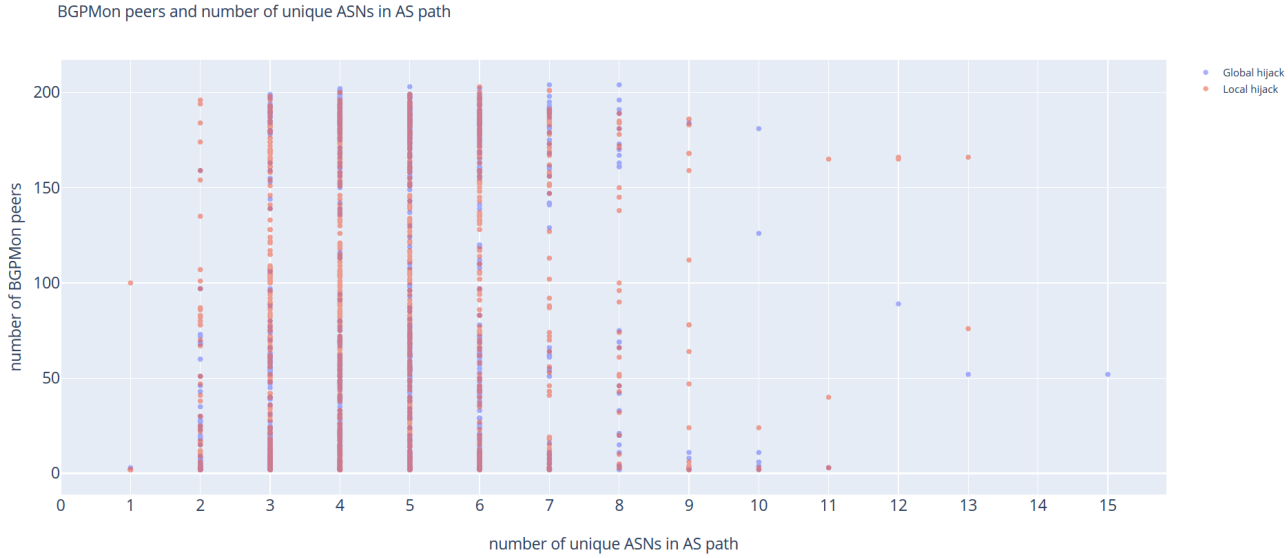


Figure 6.19: AS path length versus number of BGPMon peers that received the hijacked route

6.1.6. Summary

Following is a summary of the most important findings discussed in this section.

- There is no preference for IPv4 or IPv6 when hijacking prefixes.
- There is a large number of hijacks that target one IPv4 prefix. This is peculiar because it is unusual to announce just one IP address.
- Global and local hijacks happen equally often.
- More than half of the hijacks are withdrawn the same day and thus mitigated quickly.
- Announcements that last less than a day do not show up in the prefix histories and AS histories that are maintained by RIPEstat.
- In 92% of the hijacks there is no direct relation between the detected AS and the expected AS. In 83% there is no relation at all. This means these hijacks cannot be caused by mistakenly announcing a prefix that is owned by a peer, provider, or AS in the customer cone of the detected AS.
- Hijacks that are caused by an AS that is directly related are most often caused by a provider.
- Large ASes are relatively often involved in hijacks and they are more often the cause of hijacks than the target of hijacks. Because of their complex operations this is not necessarily unexpected.
- APNIC and AFRINIC have a relatively large number of hijacks caused by one or more of their ASes. It seems that the hijacks from AFRINIC come from one or more ASes located in Angola.
- ASes from RIPE often target other ASes from RIPE. The reason for this is unclear.
- The number of hijacks with a detected AS in Angola is high compared to the number of ASes that are located in this country.
- 22% of hijacks have a path that is not continuous. This could mean that a large number of possible origin hijacks are actually path hijacks.
- Most AS paths are not prepended
- Path prepending is not always the cause of long AS paths.
- The number of BGPMon peers that receive the hijack announcement does not seem to be related to anything. This can either mean that the BGPMon monitors are placed well and cover most of the BGP topology, or they cover one part very well and hijacks that happen elsewhere are not detected.

6.2. Results based on the relations between hijacks

This section discusses the relations between hijacks. Section 5.2 explained how hijacks can be related and how these relations are detected. This section is divided into five subsections that discuss one type of relation. Every subsection contains a plot of the distribution of the group sizes and a discussion of the most interesting groups. Because some relations partly overlap each other, the most complex ones are discussed first. This avoids having to discuss some hijacks twice. For example, hijacks that are part of the same event will also be related by the detected AS.

First the hijacks are discussed that have a prefix that is hijacked by an AS that later becomes its expected AS. The second subsection discusses the hijacks that are related because a group of ASes is hijacking the same prefixes. The third subsection focusses on hijacks that are part of the same event. Following are the groups that are related because of their detected advertisement or expected prefix. The last subsection discusses groups of hijacks that are related because they have the same detected or expected AS.

6.2.1. Detected AS and expected AS switch

The first type of relation concerns hijacks of the same prefix where the detected AS in one hijack is the expected AS in another. This type of relation can help to find announcements that are detected as possible hijacks but are likely legitimate announcements. Figure 6.20 shows the distribution of the group sizes. Note that the y-axis has a logarithmic scale.

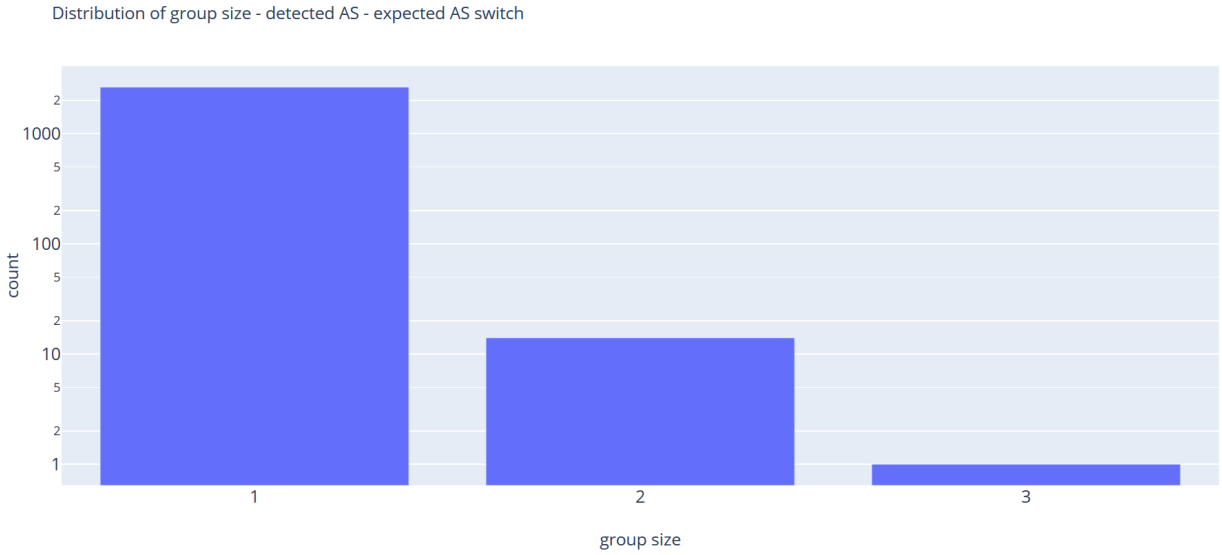


Figure 6.20: Group size - Detected AS and expected AS switch

There are only a few hijacks that are connected by this relation, 15 groups in total. Because the groups are small, each of them can easily be analysed. The most important features in this analysis are the start time, the detected ASN and expected ASN, the hijack duration, MOAS conflicts, and RPKI. These features show how much time there was between the hijacks in a group, who performed the hijacks, and if it is likely that the announcement was legitimate. To make the features more readable the hijack duration is given in days and the MOAS conflicts and RPKI entries only show the AS numbers. Following are some examples of groups that have this relation. Discussing every group is unnecessary and infeasible so only the most interesting ones are shown. The groups that are not discussed are similar.

Group ID: 90 First is the largest group. Table 6.3 shows the most important features. There are three hijacks of the prefix 80.81.192.0/21 that happen within 5 days and are caused by three different ASes. The first hijack is caused by AS3549 and lasts four days. A day later the same prefix is hijacked by AS702 and after four days the prefix is hijacked again. The detected AS is now AS13124 and AS3549 has become the expected AS. What is most interesting in this case is the number of ASes that is

announcing this prefix, which can be found in the column ‘MOAS’, in combination with the RPKI entry. At any given time there is a large number of ASes that announce the hijacked prefix. However, the AS that is the legitimate owner according to RPKI does not announce it.

Group ID	Start time	Detected ASN	Expected ASN	Hijack duration	MOAS	RPKI
90	2018-05-31 17:25:30 UTC	3549	8218	4	3549 8218 12552 29467 29686	AS6695
90	2018-06-01 19:07:33 UTC	702	8218	224	3549 8218 12552 29467 29686	AS6695
90	2018-06-04 07:17:03 UTC	13124	3549	0	702 3549 12552 29467 29686	AS6695

Table 6.3: Group 90 - Detected AS becomes Expected AS

When looking up the ASes that are listed as announcing the prefix, it seems that they all belong to different organisations in different countries. Except for AS3549 and AS702 they are, however, peering with each other. Because AS3549 becomes the expected AS for this prefix the first hijack is most likely legitimate, but the fact that the new expected AS is unrelated to all others, in combination with the large number of ASes that are announcing this prefix, is rather unusual. Looking into the prefix itself only reveals that AS6695, the one listed as the legitimate owner by RPKI, has been announcing the supernet 80.81.192.0/20 consistently from the end of 2001 until March 2013. It then switches to announcing the subnet 80.81.196.0/22 until May 2015. To know with certainty what is happening one should contact the operators of the ASes involved. Because the AS3549 becomes the detected AS, the first hijack in this group is labelled as likely legitimate. The other two hijacks cannot be labelled using this relation.

Group ID: 426 Another example is group 426. Table 6.4 shows the relevant features. This group is taken as an example because the second hijack is caused by the previous owner of the prefix. The detected AS is thus announcing a prefix it has previously owned. Because this hijack lasts less than a day, it is most likely a misconfiguration.

Group ID	Start time	Detected ASN	Expected ASN	Hijack duration	MOAS	RPKI
426	2018-07-04 09:13:05 UTC	26484	137915	4.0	26484 137915	
426	2018-08-11 14:01:32 UTC	137915	26484	0.0	137915	

Table 6.4: Group 426 - Detected AS becomes Expected AS

Group ID: 1127 The last example in this subsection is group 1127. For this group it becomes very clear that the first hijack is most likely a legitimate announcement. The detected AS, AS210250, does not only become the expected AS for the prefix but there also exists a ROA that lists it as the legitimate owner. This strengthens the belief that some hijacks are only detected because BGPStream had no access to or did not use the most recent information. This problem would be solved if there was an up-to-date and publicly available data set of prefixes and their owners.

Group ID	Start time	Detected ASN	Expected ASN	Hijack duration	MOAS	RPKI
1127	2018-09-10 11:41:02 UTC	210250	20473	269.0	20473 132335 210250	AS210250
1127	2018-10-23 13:07:19 UTC	266400	210250	0.0	210250	AS210250

Table 6.5: Group 1127 - Detected AS becomes Expected AS

Other groups The other groups of hijacks with this relation are similar to those discussed. They all contain two hijacks of which the first is labelled as likely legitimate. The second hijack cannot be labelled based on this relation. In total, this relation gave 15 hijacks a definite label. It is useful in these cases only because the same prefix is hijacked again at a later stage. If this would not happen, other criteria have to be used to determine the announcement is legitimate.

6.2.2. Detected AS set

The hijacks discussed in this subsection are those that are related because they have a detected advertisement that is announced by the same set of ASes. An example was given in table 5.5 in section 5.2. Figure 6.21 shows the distribution of the group sizes. There are not many groups with this relation, and only one group contains a large number of hijacks.

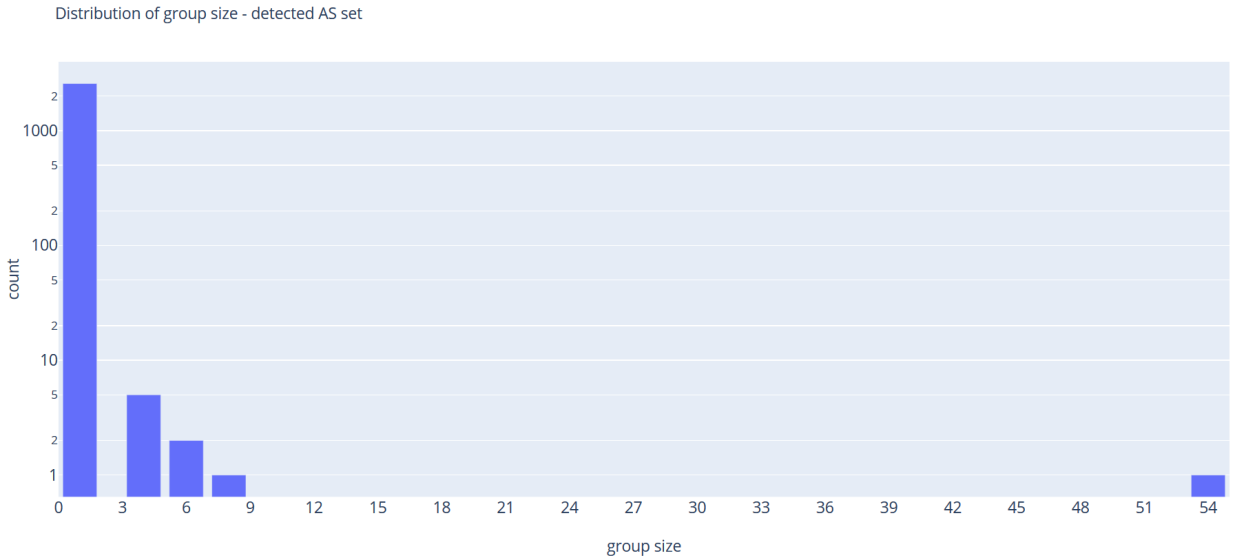


Figure 6.21: Group size - Hijacked by the same set of ASes

Group ID: 2447 Figure 6.21 shows that there is one group that is very large compared to the others. This group contains 54 different hijacks. These hijacks are caused by AS3549 and AS3356 that both belong to the same organisation. These two ASes are consistently hijacking the same prefixes from 12 June 2018 to 30 May 2019. Since the first hijack in our data set was from 20 May 2018, these ASes have been causing these hijacks during almost the whole year that is analysed. This in itself makes it already unlikely that the hijacks were caused by a misconfiguration.

The expected ASes are ASes located in various countries. In four cases they are directly related to the detected ASes as their customer. In 23 other cases, the expected AS is in the customer cone of the detected AS. The remaining 27 hijacks have no relation between the detected and the expected AS.

The duration of these hijacks varies from less than a day to more than a month. These varying durations, in addition to the fact that the hijacks have been caused so consistently during the year by two ASes of the same organisation, and because ASes with and without a relation to the detected ASes are affected, these hijacks are labelled as intentional hijacks. It is very unlikely that they are the result of a misconfiguration. It should be noted that an intentional hijack is not necessarily malicious and causing harm, it only means that the AS causing the hijack has done so on purpose.

Group ID: 2505 The group with ID 2505 is the second largest group in this category and was given as an example when explaining the relation in section 5.2. This group contains 8 hijacks that happened within 20 minutes and were caused by four different ASes. Table 6.6 shows some of the features of these hijacks. The two prefixes that are hijacked belong to an AS that has a university as the registered owner. The four detected ASes belong to three different organisations, one of which is also a university.

Looking into the hijack labels, it seems that the cause of the hijacks might be visible in the detected AS paths. Table 6.7 shows the detected AS paths and the path relations. None of the detected AS paths are continuous. All of them have a gap between AS134006 and the next AS. The paths that start at AS1007 and AS1008 do not seem logical at all because they have a number of unrelated ASes at the beginning of the path. These hijacks are a good example of path hijacks can do. Which AS has

caused the hijacks is not certain, but it is likely that it was AS134006 or AS9498 as they both appear in all paths. All hijacks in this group are labelled as path hijacks.

Start time	Detected advertisement	Detected origin AS	Expected AS
2019-05-25 13:02:17 UTC	103.69.168.0/24	1008	8
2019-05-25 13:02:17 UTC	103.69.169.0/24	1008	8
2019-05-25 13:05:47 UTC	103.69.168.0/24	1007	8
2019-05-25 13:05:47 UTC	103.69.169.0/24	1007	8
2019-05-25 13:11:37 UTC	103.69.168.0/24	2	8
2019-05-25 13:11:37 UTC	103.69.169.0/24	2	8
2019-05-25 13:22:59 UTC	103.69.168.0/24	1	8
2019-05-25 13:22:59 UTC	103.69.169.0/24	1	8

Table 6.6: Group 2505 - Prefixes hijacked by the same set of detected ASes

Detected AS path	Path relations
29834 4637 9498 134006 1001 1002 1003 1004 1005 1006 1007 1008	p2p p2c p2c no relation no relation no relation no relation no relation no relation no relation
133812 137363 58552 9498 134006 1001 1002 1003 1004 1005 1006 1007 1008	c2p c2p p2p p2c no relation no relation no relation no relation no relation no relation no relation
4826 1221 4637 9498 134006 1001 1002 1003 1004 1005 1006 1007	p2c c2p p2c p2c no relation no relation no relation no relation no relation no relation
7018 1299 9498 134006 1001 1002 1003 1004 1005 1006 1007	p2p p2c p2c no relation no relation no relation no relation no relation no relation no relation
8319 21385 3356 1299 9498 134006 1 2	c2p c2p p2p p2c p2c no relation no relation
20205 6939 9498 134006 1 2	c2p p2p p2c no relation no relation
63774 59103 2907 4637 9498 134006 1	c2p c2p p2p p2c p2c no relation
63774 59103 2907 4637 9498 134006 1	c2p c2p p2p p2c p2c no relation

Table 6.7: Detected AS paths and path relations for group 2505

Group ID: 2455 The last example of groups with this type of relation is group 2455. Table 6.8 shows the most important features of this group. This group is an example of hijacks that were falsely labelled as related. AS42440 and AS41689 both hijack the same two prefixes, but they do this at completely different times. Therefore it is more likely a coincidence so the hijacks are not labelled based on this relation.

group ID: detected advertisement set	start time	detected ASN	expected ASN	detected advertisement
2455	2018-07-18 10:40:56 UTC	41689	62375	79.143.84.0/24
2455	2019-01-13 06:40:33 UTC	41689	12697	46.249.96.0/24
2455	2019-04-01 05:45:40 UTC	42440	12697	46.249.96.0/24
2455	2019-04-14 21:17:28 UTC	42440	209488	79.143.84.0/24

Table 6.8: Group 2455 - Prefixes hijacked by the same set of detected ASes

Other groups The other groups of hijacks are similar to the groups that were discussed in the previous paragraphs. There were 8 other path hijacks, 2 other hijacks that were intentionally caused by a group of ASes, 6 hijacks that were already labelled as legitimate announcements, and 12 hijacks that had no relation and could not be labelled. In total, 72 hijacks received a label based on this relation.

Without this relation, it would have been difficult to label these hijacks. So although there are not many groups, the relation is very useful.

6.2.3. Event

The third type of relation discussed in this subsection is the one hijacks that are part of the same event. This means they are caused by the same AS within a period of 2 hours. Figure 6.22 shows the distribution of the group sizes. The figure shows one very large group, many groups of with less than 10 hijacks, and some larger ones. The largest groups are discussed in this subsection.

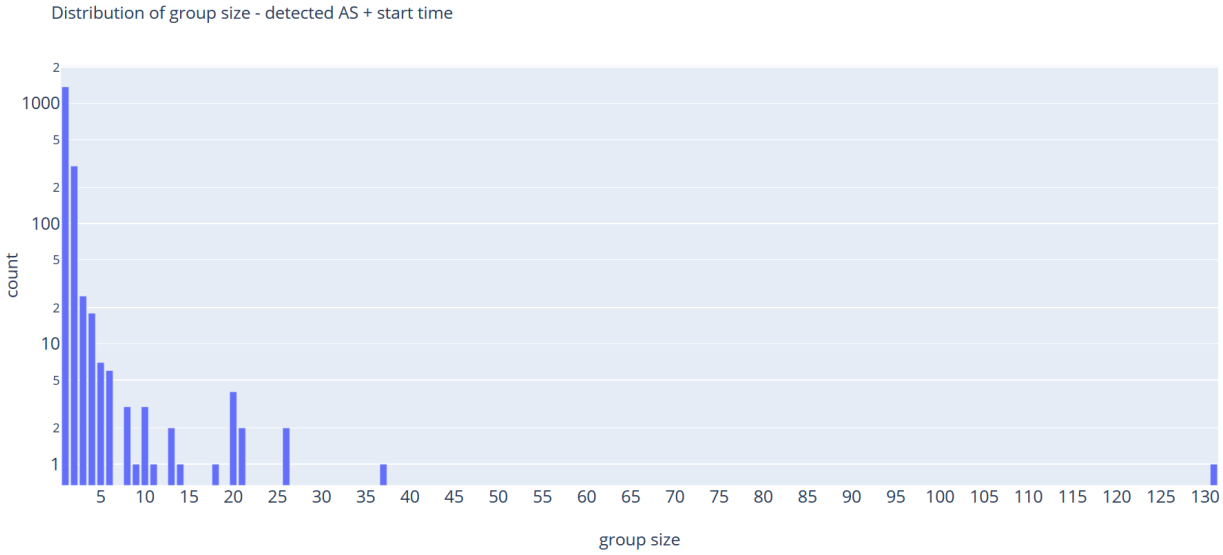


Figure 6.22: Group size - Event

AS37468 The first group discussed is the one that consists of 131 hijacks. The hijacks in this group are caused by AS37468 that is located in Angola. It thus seems that the large number of hijacks from Angola and thereby AFRINIC that were found in the previous section were mostly caused by one AS. The hijacks were started on 19 July 2018 within 17 minutes. All announcements were withdrawn the same day. The expected ASes are from various countries, but 93 (71%) of the hijacks have an expected AS located in the US. The expected ASes are not related to AS37468 except for three: AS36086, AS12353, and AS12390. These three ASes are peers of the detected AS.

The prefixes targeted in these hijacks are all prefixes that are in use by one other AS. The hijacked prefixes are equal to the expected prefixes, meaning that the hijacks are local. Because the hijacks are all caused around the same time, are withdrawn quickly, and only have expected ASes that are not in the customer cone of AS37468, this group may very well be caused by a route leak with re-origination. Looking at the AS paths shows that the detected AS has announced the routes to at least two of its peers. Because there are no routes of its customers, these routes come from another peer or a provider, or are originated by the detected AS itself. If these hijacks are not caused by a route-leak they have to be done intentionally. There is no other type of misconfiguration that could cause this. Because a route leak is a bit more likely based on the characteristics of this group, all hijacks are labelled as a route leak with re-origination.

AS8859 The second largest group contains 37 hijacks that are caused by AS8859 within 6 minutes. This group also explains a part of the large group of hijacks of one IP address. All hijacks in this group have a /32 IPv4 prefix as the detected advertisement except for one which is a /30. These prefixes are not part of the same expected prefix and belong to 34 different ASes that are not related to the detected AS. The expected ASes are mostly located in different countries, but 5 are in China and 7 in Russia. All hijacks were withdrawn within a day. Because they target different IP addresses from

different ASes that are not related to the detected AS, these hijacks are labelled as intentional hijacks. No misconfiguration could cause this and it would also be very difficult to do this accidentality.

AS6453 Another interesting group is a group of 26 hijacks that are caused by AS6453. The hijacks are withdrawn after 4 days. These hijacks are 26 local hijacks from ASes that are all in the customer cone of AS6453 and are located in India. When looking into the expected ASes it is found that they all have AS45194 in common. Either as their provider or as the provider of their provider. It thus seems that routes coming from this AS were re-originated by AS6453. However, AS45194 itself has no direct link with AS6453. There is one AS in between. AS4755 belongs to the same organisation as AS6453 and is a provider of AS45194. AS4755 is a customer of AS6453. Based on this information and the fact that all hijacks happen at the same time, they are labelled as a misconfiguration. It is not labelled as a route leak with re-origination, because although the routes are re-originated, AS6453 is announcing routes it received from its customer. It is thus not a route leak. The hijacks could be intentional, but a misconfiguration is in this case more likely. Because all ASes are in the customer cone of AS6453 and AS4755 that belong to the same organisation, most of the traffic will be provided by these ASes and therefore hijacking these routes will not have any advantage.

AS17639 The fourth group discussed here is again interesting in its own way. This group contains 26 hijacks that are caused by AS17639. These hijacks happen within 10 minutes and are all withdrawn the same day. What is interesting about these hijacks is the detected advertisements and the expected prefixes. All hijacks are global hijacks and they are done in such a way that they always cover the whole expected prefix. Table 6.9 contains some examples. For each expected prefix there are two hijacks that cover half of this prefix. Each hijack in this group covers half of an expected prefix.

detected advertisement	expected prefix
119.28.164.0/24	119.28.164.0/23
119.28.165.0/24	119.28.164.0/23
151.101.52.0/23	151.101.52.0/22
151.101.54.0/23	151.101.52.0/22
172.81.120.0/22	172.81.120.0/21
172.81.124.0/22	172.81.120.0/21
85.240.0.0/14	85.240.0.0/13
85.244.0.0/14	85.240.0.0/13
95.211.0.0/17	95.211.0.0/16
95.211.128.0/17	95.211.0.0/16

Table 6.9: AS17639 - Examples of detected advertisement and expected prefix

The expected prefixes have a range of lengths and are owned by ASes that are either peers of the detected AS or are not related at all. In 10 cases the hijacks are targeting a prefix of a peer, the other 16 cases target a prefix of an AS that is not related. There is no way that these hijacks can be caused by a misconfiguration, all are labelled as intentional hijacks.

AS36937 The last group discussed here is a group of hijacks caused by AS36937. Table 6.10 shows some of the features. This group is discussed because most of the detected advertisements follow a pattern. They start with three octets that are equal, followed by a 0 and /24. This could be done intentionally, there is also a very small chance that it is the result of a strange misconfiguration. However, looking at the AS paths, more than half are not valid paths. All ASes in these paths are located in South-Africa, but they belong to different organisations. Altogether it is difficult to give these hijacks a label. They could be intentional, path hijacks, or the cause of a misconfiguration. For this reason, these hijacks are not labelled.

Expected AS	Detected advertisement	Expected prefix	Detected AS Path	Path relations
4668	150.150.150.0/24	150.150.0.0/16	37439 36937	no relation
21195	192.16.144.0/24	192.16.144.0/24	37439 36937	no relation
37918	192.68.245.0/24	192.68.245.0/24	37105 36937	p2p
3215	2.2.2.0/24	2.2.0.0/16	37439 36937	no relation
206747	25.25.25.0/24	25.25.25.0/24	37439 36937	no relation
4134	27.27.27.0/24	27.16.0.0/12	37105 36937	p2p
12638	5.5.5.0/24	5.4.0.0/14	37105 36937	p2p
16509	52.52.52.0/24	52.52.0.0/15	37105 36937	p2p
721	55.55.55.0/24	55.0.0.0/8	37439 36937	no relation
2647	57.57.57.0/24	57.0.0.0/8	37439 36937	no relation
701	63.63.63.0/24	63.48.0.0/12	37439 36937	no relation
3209	82.82.82.0/24	82.82.0.0/15	37439 36937	no relation
2860	95.95.95.0/24	95.95.64.0/18	37105 36937	p2p

Table 6.10: AS36937 - Example of patterned prefixes

Other groups The other groups are similar to those discussed already. The smaller the groups are, the more difficult it is to label them based on this relation. For this reason, not all hijacks are labelled. In total, there are 418 hijacks labelled using this relation. 183 hijacks received the label 'intentional'. There were 131 hijacks caused by a route leak with re-origination, 54 path hijacks, and 50 hijacks that resulted from a misconfiguration.

6.2.4. AS

The fourth type of relation that is used to group hijacks is the relation based on the involved ASes. This subsection discusses groups that share the same detected AS and groups that share the same expected AS. Groups that are already covered by another relation are left out. First are the groups based on the detected AS. Figure 6.23 shows the distribution of group sizes. Note that the y-axis has a logarithmic scale. Most detected ASes cause only one hijack, but there are many that cause multiple. Groups of more than 25 hijacks are manually analysed. This size is chosen because it is infeasible to analyse all of them. The most interesting groups are discussed in this subsection.

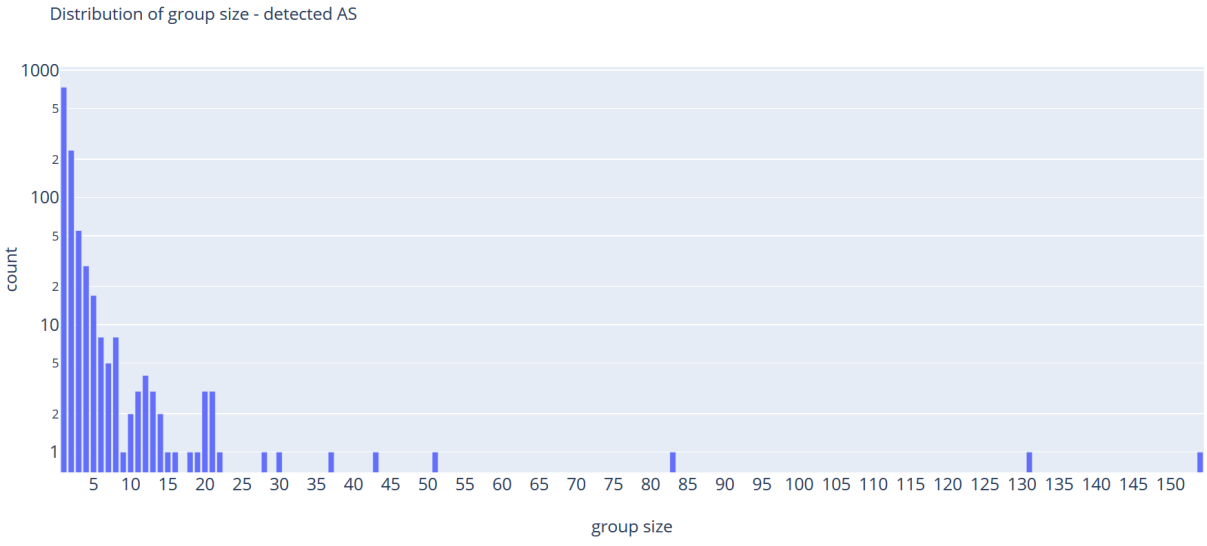


Figure 6.23: Group size - Detected AS

AS50607 The first group of hijacks discussed here is a group of 154 hijacks caused by AS50607. This group explains some of the findings from the previous section. AS50607 is an AS located in Poland, which was one of the countries where a large number of hijacks were caused. In addition, all hijacks

caused by AS50607 are hijacks of IPv4 prefixes with prefix length 32. As seen in subsection 6.1.1, there was an unusually large number of /32 IPv4 prefix announcements found in the hijacks data set. This group accounts for a bit more than half of these announcements.

The hijacks happened more or less continuously between 20 May 2018 and 11 January 2019. Each hijack targets a different prefix from a different ASN. The expected ASes are located in many different countries, but 25% are located in China. The ASes that own the hijacked prefixes are in most cases not related to AS50607, but some are in its customer cone. The hijacks do not go on for a long time, the announcements are mostly withdrawn within a day. Only two cases last two days.

In summary, there are 154 hijacks caused over the course of 9 months that all target a different IP address from a different AS. The targetted ASes are in most cases completely unrelated to the detected AS and each hijack is quickly withdrawn. These points together make it very unlikely that the hijacks are the result of a mistake made by AS50607. For this reason, these hijacks are labelled as 'intentional'.

AS6453 AS6453 has in total caused 83 hijacks. A part of these hijacks was already labelled because they were part of one event. These hijacks received the label 'misconfiguration'. Looking at all hijacks caused by AS6453 one could wonder if this was correct. AS6453 has been causing hijacks during the whole year. AS6453 is located in the US but owned by a company with its headquarters in India. Most expected ASes are from India, but there are also several ASes from Lebanon and Afghanistan and some from other countries. There does not seem to be a clear pattern in the hijacks. There are global and local hijacks, they have a duration from less than a day to up to a month, some of the prefixes are in use, some are not.

In general, it is difficult to label these hijacks. The misconfigurations that were labelled earlier, do look like misconfigurations. For the remaining hijacks it is more likely they were done intentionally. It is also possible that the misconfigurations were caused intentionally. For this reason, based on this relation, all hijacks in this group get the label intentional. This does mean that some hijacks now have two labels.

AS3356 and AS3549 The hijacks that were related because they were hijacked by the same set of ASes contained a large group that was caused by AS3356 and AS3549. Both ASes also cause some other hijacks, but a large part of their hijacks was already labelled as intentional. These ASes form the fourth and fifth largest group of hijacks that are related by detected AS only. After looking at the unlabelled hijacks it is decided to keep them unlabelled. The reason for this is that there does not seem to be a clear pattern in the hijacks. They happen on different days, have different targets, different durations, etc. It is not possible to label the entire groups with confidence, but it is suspicious that these ASes cause so many hijacks.

Other groups There are other larger groups that are not discussed here. This is because these hijacks were part of the same events. They were already discussed in the previous subsection. Smaller groups are not analysed because it is infeasible and it is more difficult to find a label based on the relation.

Hijacks related by the expected AS Figure 6.24 shows the distribution of the size of groups of hijacks that are related because they have the same expected AS. It is more difficult to label groups based on the expected AS, because the hijacks are not necessarily related. They just target a prefix from the same AS. For this reason, only groups with 15 or more hijacks are analysed. After analysing the groups it seems that these hijacks are more related by coincidence than by expected AS. Large ASes are hijacked more often and form a group, but the hijacks in the group are not related. In other groups, the hijacks are related by expected AS and also by the expected prefix. Because they share the expected prefix, they are discussed in the next subsection.

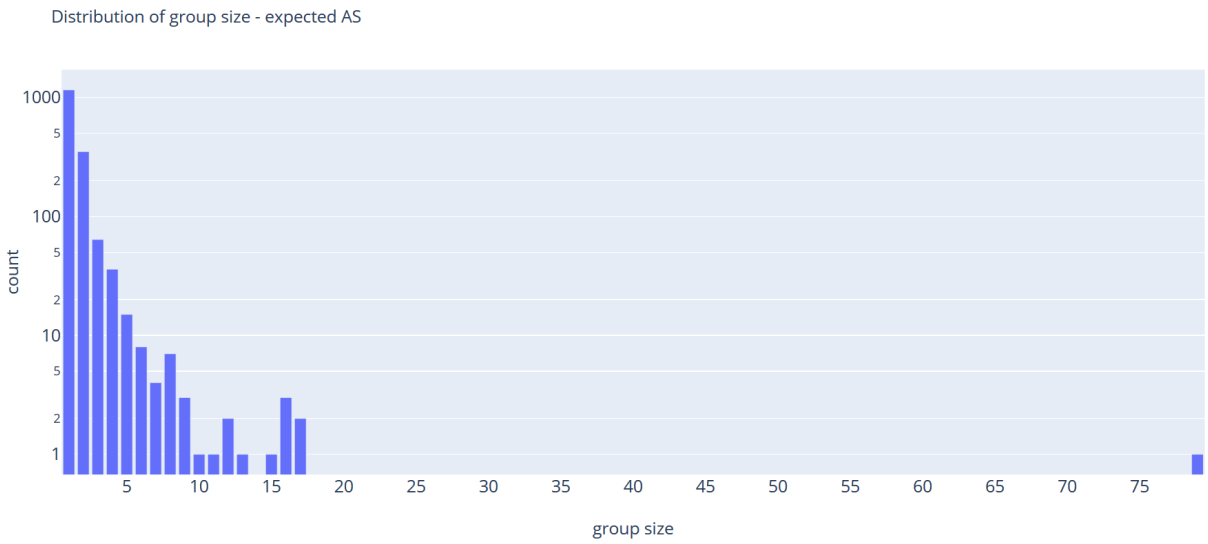


Figure 6.24: Group size - Expected AS

6.2.5. Prefix

The last type of relation is based on the prefixes that are hijacked. This means that related hijacks share the detected advertisement or the expected prefix. As the previous subsection, this subsection is divided into two parts. The first part discusses groups of hijacks that share the same detected advertisement, the second part discusses groups of hijacks that share the same expected prefix.

Figure 6.25 shows the distribution of group sizes based on the detected advertisement. In total, there are 2532 different groups. The smallest groups contain one hijack, the largest group contains five. After looking at the largest groups, it seems that this relation is not useful for labelling hijacks. The hijacks in the largest groups are not related, they are only hijacking the same prefix. This relation can thus not be used to label hijacks in this data set. It would only be useful if a group of unique ASes would simultaneously target the same prefix.

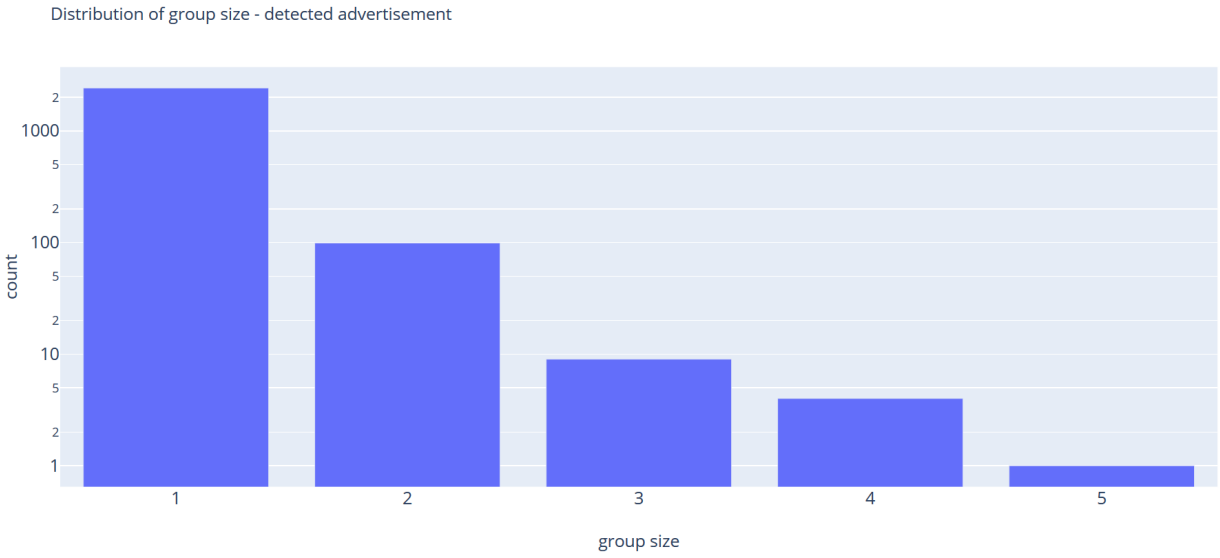


Figure 6.25: Group size - Detected advertisement

Figure 6.26 shows the distribution of group sizes based on the expected prefix. The largest group is discussed here.

Figure 6.26: Group size - Expected prefix

2800::/11 Over a period of two weeks, there are 79 hijacks of a subnet of the prefix 2800::/11. These hijacks have 79 different detected advertisements and are caused by 69 different ASes. The lengths of the detected advertisements range from /32 to /48. These are the prefix lengths that are most announced in IPv6. Because of this, it looks like a large IPv6 prefix was divided into subnets that were assigned to various ASes. These ASes started to announce these prefixes and the announcements were detected as possible hijacks. The hijack durations support this idea because most hijacks last more than a month. For this reason, these hijacks are labelled as 'legitimate announcement'.

Other groups Most of the other groups were similar to the one discussed. They had a large expected prefix that was divided into smaller prefixes that were announced by several ASes. There was one group of hijacks that could not be labelled because there was no clear pattern in the hijacks. Out of the 151 analysed hijacks 145 are labelled as legitimate announcements. The other 6 could not be labelled.

6.2.6. Summary

Following is a summary of this section. Altogether it appears that the relations between hijacks are very informative. Using the context of a group of hijacks helps to label the individual hijacks.

- Labelling hijacks because the detected AS is found as the expected AS in a later hijack is useful, but it is only possible when a prefix is hijacked at a later point.
- It seems that some legitimate announcements are detected as possible hijacks because BGP-Stream did not use or did not have access to up-to-date information.
- Finding groups of ASes that hijacked the same prefix helped to label 76 hijacks that would be more difficult to label otherwise. However, the relation should be used with caution because it is possible the hijacks in a group are related only by coincidence.
- Several hijacks that were detected as possible origin hijacks were in fact path hijacks. Because it can be problematic to find the AS that caused a path hijack, it is important that BGP implements protection against path hijacks. Until that happens it is necessary to be able to detect these hijacks.
- There was one AS in Angola that caused a large number of hijacks. This was most likely the result of a misconfiguration. It explains the large number of hijacks from Angola and AFRINIC.
- Looking for hijacks that are part of the same event is very helpful. Using this approach 418 hijacks could be labelled. It helped to find misconfigurations, path hijacks, and other intentional hijacks.

- More than half of the hijacks that target one IP address were caused by an AS from Poland. These hijacks were caused during 9 months and were mostly withdrawn within a day. The reason for these hijacks is unclear, but they are likely done intentionally.
- There are several ASes that cause many hijacks over a longer period. Looking for hijacks with the same detected AS helped to find these ASes. Without this relation the hijacks would look like independent events.
- Hijacks with the same expected AS are often not related. They just target prefixes of the same AS. It is more useful to look for hijacks with the same expected prefix.
- Hijacks that have the same detected advertisement are not necessarily related. Small groups with this relation are often formed by coincidence. This relation is thus only useful when a prefix is hijacked many times or when the hijacks in one group start at the same time.
- There are several groups of hijacks with the same expected prefix. Almost all of these hijacks were labelled as legitimate announcements. The reason for this is that it looks like some larger prefixes were divided into subnets that were allocated to several ASes. When these ASes announced their prefixes, the announcements were detected and considered to be possible origin hijacks.

6.3. Results based on the hijack labels

The last section of this chapter gives an overview of the labels that were given to hijacks. The labels were explained in section 5.3 of the previous chapter. They are not mutually exclusive so a hijack can receive many labels. Hijacks that have the same label form a group. Looking at the other labels of the hijacks in a group may help to give insight into the cause of hijacks. Each subsection discusses one label and this section is ends with a summary.

6.3.1. Invalid hijacks

In total there are 13 (0.49%) invalid hijacks. One of these hijacks is invalid because it has the negative expected ASN -94966896. Since negative AS numbers do not exist, this hijack is not further labelled. The other 12 hijacks are invalid because their detected AS does not match the origin AS in the AS path. The detected AS for each of these hijacks is AS2147483647 but the origin AS in the AS path varies. It is however always an AS with an ASN larger than 4200000000. What is interesting is that 2147483647 is the largest number a signed 32-bit number can represent. As the origin ASN is always larger than this number, the inconsistency is probably caused by mishandling 32-bit ASNs. To avoid drawing wrong conclusions these hijacks are not further labelled.

6.3.2. Legitimate announcements

In subsection 5.3.2 was explained that not all possible hijacks detected by BGPStream are actual hijacks. Some are legitimate announcements. Hijacks were immediately labelled as ‘likely legitimate’ if there was a ROA that had the detected AS listed as the owner of the detected advertisement, and if the detected AS was later found as the expected AS in a hijack of the same prefix. In total there are 89 hijacks that fall into one or both categories. There are 75 hijacks that have a ROA with the detected AS listed as the owner of the detected advertisement, and 15 hijacks were the detected AS later becomes the expected AS. This last category of hijacks was also discussed in subsection 6.2.1

Some hijacks were labelled as legitimate announcements based on another label. This label is ‘failure to summarize’. It was found that some ASes that were announcing a supernet were doing so legitimately. They announced the whole range as subnets before and were thus only aggregating their prefixes. There are 7 hijacks of this type.

Lastly, hijacks could be labelled as legitimate announcements based on their relation to other hijacks. This was explained in section 6.2. There were 145 hijacks labelled as legitimate announcements because they were part of a group of hijacks that had a large expected prefix divided into smaller prefixes that were announced by different ASes. Based on all the criteria mentioned, there are in total 241 (9.04%) hijacks labelled as legitimate announcements. Some of these hijacks matched multiple criteria.

6.3.3. Hijacks of one IP address

In the previous section it was explained that announcements of one IP address are uncommon in BGP and that because there is a large number of hijacks of one IP address, these hijacks are further analysed. In total, there are 302 (11.33%) hijacks with this label. They are not all caused by the same AS, but more than half were caused by AS50607. This group was analysed in section 6.2 and the hijacks were labelled as intentional hijacks. There are 4 other ASes that caused 20 or more of these hijacks and 28 ASes that caused 4 or less. So it is not something only a few ASes do.

It is not clear why ASes would hijack one IP address. Looking at the blacklist labels, they are not used for spam. Only 11 of the hijacks have the detected advertisement appear on a blacklist, but they were already blacklisted before the hijack. Because 8 of these 11 hijacks were caused by AS8859, it could be that AS8859 caused the hijack to analyse the spam.

Overall, the hijacks of one IP address in this data set thus do not seem to be used for spam. They are also not caused by ASes using the wrong prefix length and other labels neither provide any clarity. The only thing that can be said is that the detected AS is hardly ever related to the expected AS in any way.

6.3.4. Failure to summarize and failure to aggregate

Failures to summarize and failures to aggregate do not appear often in the hijacks data set. In fact, there are no cases of failures to aggregate. As was discussed in the subsection about legitimate announcements, the label ‘failure to summarize’ helped to identify 7 hijacks as legitimate. Only one hijack is an actual failure to summarize. It is a hijack of a prefix that was announced before by the detected AS. The announcement was withdrawn within a day.

6.3.5. Possible typographical error

A hijack may be caused by an announcement of a mistyped prefix. In 66 (2.48%) hijacks the detected AS normally announces a prefix that has a string distance of 1 from the detected advertisement. This is not always necessarily the result of a mistake, because an AS can also announce a prefix that is close to one of its own. For example, 192.0.2.0/24 and 192.0.3.0/24 are both a subnet of 192.0.3.0/23. Deciding to announce 192.0.3.0/24 may well be a conscious decision instead of the result of mistyping 192.0.2.0/24.

In this subsection some examples are shown that make it clear that typos are truly a problem. Table 6.11 contains the closest prefix and detected advertisement of some of the hijacks and the type of mistake that is possibly made. It also contains examples of hijacks that are likely not caused by a typographical error, because the differing numbers are nowhere close to each other on a keyboard. Not all hijacks with this label are caused by mistyping, but there are several cases where this seems to be the problem. It could be easily prevented and is thus worth looking into when securing routers.

Closest prefix	Detected advertisement	Type of error
185.82.215.0/24	185.83.215.0/24	Neighbouring number on keyboard
181.50.140.0/23	182.50.140.0/23	Neighbouring number on keyboard
103.52.190.0/24	103.52.180.0/24	Neighbouring number on keyboard
125.212.133.0/24	125.212.33.0/24	Forgetting a digit
199.227.47.0/24	99.227.47.0/24	Forgetting a digit
114.69.241.0/24	14.69.241.0/24	Forgetting a digit
210.176.130.0/24	210.176.30.0/24	Forgetting a digit
81.161.238.0/24	81.61.238.0/24	Forgetting a digit
125.21.189.0/24	125.214.189.0/24	Adding a digit
23.205.42.0/24	23.209.42.0/24	Not a typo
46.148.113.0/24	46.148.117.0/24	Not a typo
192.107.1.0/24	197.107.1.0/24	Not a typo

Table 6.11: Closest prefix and detected advertisement of hijacks possibly caused by a typo

6.3.6. Wrong prefix length

There are 7 (0.26%) hijacks that received the label ‘wrong prefix length’. Out of these hijacks, 5 already received the label ’likely legitimate’. One of them when labelling hijacks with a failure to summarize, and the other four because they were part of a group of hijacks of a large IPv6 prefix as was explained in subsection 6.2.5. All hijacks are hijacks of a prefix that is related to another prefix announced by the detected AS. Three hijacks fall into the category ‘announcing supernet’ and the other four have the label ‘announcing subnet’. Table 6.12 shows for each hijack the closest prefix of the detected AS and the detected advertisement. Whether the hijacks are truly caused by mistakenly using the wrong prefix length is difficult to say.

Closest prefix	Detected advertisement
2803:9140::/32	2803:9140::/36
2804:332c::/33	2804:332c::/32
2804:4e78::/33	2804:4e78::/32
2804:2038::/32	2804:2038::/33
192.162.212.0/24	192.162.212.0/22
185.145.192.0/22	185.145.192.0/24
93.152.208.0/20	93.152.208.0/22

Table 6.12: Closest prefix and detected advertisement of hijacks with wrong prefix length

6.3.7. Hijacking customer route

There are 122 (4.58%) hijacks caused by the detected AS hijacking a route of its customer. In total, 88 different ASes cause these hijacks. Most of these hijacks are local hijacks but 20 are global. These 20 global hijacks are all withdrawn within a day. In fact, most hijacks (71) are withdrawn within a day. There are 51 that last longer and 31 of these are withdrawn within a week. Looking at the other labels it does not seem that these hijacks happen for a clear reason. However, in 102 of the 122 hijacks, the expected AS is announcing the detected advertisement at the time of the hijack. Most of these hijacks may be the result of a misconfiguration, but this cannot be said with certainty. Another possibility is that they are legitimate announcements. Seven hijacks were labelled as such because of a ROA for the detected AS.

6.3.8. Hijacks of prefix that appears on blacklist

The blacklist labels were divided into two groups. The labels that refer to prefixes that were already blacklisted at the time of the hijack, and those there were blacklisted during the hijack. Each group has a label for the detected advertisement, the expected prefix, other supernets of the detected advertisement, and subnets of the detected advertisement. There are 182 (6.83%) hijacks with a prefix appearing on a blacklist before the start of the hijack. There are 100 (3.75%) hijacks with a prefix that is blacklisted during the hijack.

Detected advertisement blacklisted before hijack There are 19 hijacks of a detected advertisement that was blacklisted at the time of the hijack. More than half, 11 out of 19, were hijacks of one IP address. It was already discussed in subsection 6.3.3 that 8 of these 11 hijacks were caused by one AS. Possibly because it was investigating the spam. This could also be the reason why the other hijacks were caused. These hijacks do not last long, 14 are withdrawn within a day and the other 5 within a week.

Expected prefix blacklisted before hijack A hijack receives the label ‘expected blacklist before’ when the expected prefix is a supernet of the detected advertisement and was on a blacklist at the time of the hijack. All hijacks with this label are therefore global hijacks. In total, there are 18 hijacks with this label. Of these hijacks there are 6 that target only one IP address. It is unknown if this address was causing spam. These hijacks were withdrawn within a day.

What is interesting is that of the other 12 hijacks, 9 had a detected advertisement that was announced before by the detected AS and all these hijacks last more than a month. It could thus be legitimate announcements, but four of these were labelled as a path hijack because they were part of the same event. This shows again that it is really difficult to find a definite cause of hijacks.

Supernets and subnets blacklisted before hijack Only two hijacks have supernet other than the expected prefix on a blacklist at the time of the hijack. As such, not much can be said about them. This in contrast to the number of hijacks that have a subnet blacklisted at the time of the hijack. There are 154 of such cases. It is difficult to say why these prefixes are hijacked. The blacklisting is not caused by the hijack, and because it concerns a subnet, it may not even be caused by the expected AS. The hijack is thus not necessarily related to the fact that the prefix was blacklisted. Other labels and features also provide no insight in this matter.

Detected advertisement blacklisted during hijack There are 30 hijacks of a prefix that is blacklisted during the hijack. It is not sure if the hijack always causes the blacklisting because in 11 cases the expected AS was also announcing the prefix at the same time. These 11 hijacks were all local hijacks. The other 19 hijacks were global. There was no MOAS conflict for these hijacks, so the detected AS is the only one announcing the prefix at the time of the hijack. In 26 out of 30 (86.7%) cases there was no ROA with an owner for the detected advertisement. Considering that out of all hijacks 86.5% has no ROA it does not seem that prefixes with ROAs are more avoided when hijacking a prefix for spam.

Expected prefix blacklisted during hijack There are 4 hijacks that have the expected prefix blacklisted during the hijack. Two of these were labelled as a legitimate announcement based on their relation to other hijacks, and one was labelled as a path hijack. Because all 4 hijacks are global hijacks, it is not possible to tell if the detected AS or another AS caused the spam. The spam may come from another range of the expected prefix.

Supernets and subnets blacklisted during hijack For blacklisted expected prefixes of a global hijack, it is difficult to say if the blacklisting was the result of the hijack. This also holds for blacklisted supernets. It is not clear which range of the supernet is causing the spam and it does not have to be related to the hijack. There are 15 hijacks in the data set that had a supernet of the detected advertisement blacklisted during the hijack. Except for one, all these hijacks lasted longer than a week and 7 lasted longer than a month.

Hijacks that had a subnet of the detected advertisement blacklisted during the hijack form the largest group. It is difficult to directly link the blacklisting to the hijack because the subnet can be announced by another AS. In 22 out of the 72 cases, there was already a subnet blacklisted before the hijack, so part of the prefix was already being misused. If the hijack lasts longer, the chances of a subnet being blacklisted during the hijack increases because there is a longer period in which a range of the prefix can be misused. Since 38 of these hijacks last longer than a month, and 17 of them more than a week, it is impossible to tell if the hijacker caused the spam.

6.3.9. Hijacks of unused prefix

There are 43 (1.61%) hijacks of unused prefixes. It was discussed earlier that in some cases unused prefixes are hijacked with the intent to misuse the prefix for malicious purposes such as spam. Looking at the blacklist labels, this is not the case for these hijacks. Only one hijacked prefix is blacklisted after the hijack. In all other cases, the detected advertisement does not appear on a blacklist and neither do its subnets or supernets. The other labels do not provide any insight into these hijacks either.

6.3.10. Announcing subnet

There are 42 (1.58%) hijacks in which the detected AS announces a subnet of a prefix it is normally announcing. In 10 cases the expected AS is announcing this prefix at the time of the hijack, these are all local hijacks. The other 32 cases are all global hijacks. The detected AS is announcing a subnet of one of its own prefixes and this subnet is owned by the expected AS. It could thus be that the detected AS and the expected AS are related. However, this almost never the case. There are two hijacks in which the ASes are peers, but in other cases they are not directly related and also not in each other's customer cone.

Looking at other labels, it seems that 21 of these hijacks have a detected AS that announced the detected advertisement before, and 18 of these hijacks last more than a month. Because there are no ROAs it is unknown who the owner really is, but it might well be that these hijacks are in fact legitimate announcements. It is the case for 7 hijacks. They were labelled legitimate based on their relation to other hijacks. The other 21 hijacks were almost all withdrawn within a day, so it is unlikely that they are also legitimate.

6.3.11. Announcing supernet

In 15 (0.56%) hijacks the detected AS is announcing a supernet of a prefix it normally announces. Three of these are labelled as a legitimate because there is a ROA listing the detected AS as the owner.

Another 6 hijacks are labelled as legitimate announcement by looking for a failure to summarize. From the 6 hijacks that are now left, 3 are also labelled as legitimate, but these labels were based on relations to other hijacks. It thus seems that hijacks of a supernet in most cases are not hijacks but legitimate announcements. Therefore it does not seem to be a real issue in BGP, but more so of the detection algorithm that BGPStream uses.

6.3.12. Hijacks of prefixes that have a ROA

A ROA lists the legitimate owner of a prefix. If a hijack targets a prefix for which a ROA exists that lists the detected AS as the owner, it means the detected AS is allowed to announce the detected advertisement. There are 75 hijacks where this is the case. The date the ROAs were created is unknown, but it is fair to assume these announcements are legitimate.

There can be multiple ROAs for one prefixes. For 21 hijacks there is a ROA for both the detected and expected AS, meaning both are allowed to announce the prefix. For 5 of these hijacks, there is also a ROA for another AS. There are 5 hijacks for which there is a ROA for the detected AS and another AS, but not for the expected AS. This does not necessarily mean that the expected AS cannot announce the prefix, it only means it did not create a ROA.

ROAs should, in theory, prevent origin hijacks because before an AS propagates a route it can check if the origin of the announcement matches the AS listed in the ROA. There are 241 hijacks where the expected AS has created a ROA for its prefix. This is 9.04% of all hijacks. In total, 13% of hijacks have a detected advertisement for which a ROA exists. Globally, 16.5% of IPv4 prefixes have a ROA so the number is comparable. [25] It does not seem that prefixes with ROAs are avoided when hijacking but it is possible that in some cases a ROA was created after the hijack.

6.3.13. AS23456 appears in the AS path

Only two hijacks have AS23456 appearing in the detected AS path. These hijacks are related. They happen at the same time, target the same prefix, and have a detected AS path that is mostly the same. The number 23456 does not appear at the beginning of the path, but somewhere in the middle. It is thus not related to the detected AS. The number only appears twice, so it seems that 32-bit numbers are handled correctly in most cases, it is thus not a large issue in BGP.

6.3.14. Hijacks with the AS path prepended by the origin

In total there are 142 (5.33%) hijacks that have an AS path that is prepended by the origin AS. These hijacks are caused by 107 different ASes. Path prepending by the origin AS is thus not very common, but it is neither done by just a small group of ASes. It happens equally often in both local and global hijacks and also for all hijack durations. When an AS prepends its path, it does so intentionally but this label by itself gives not enough certainty to assume all hijacks with a prepended path are intentional hijacks, and other labels do not provide more insight.

6.3.15. Global hijacks and local hijacks

At the beginning of this chapter, it was discussed that the number of global and local hijacks is almost equal. This subsection discusses if there are any obvious differences between them and if there are typical features that are only seen in one group.

The first point concerns the local hijacks. During 1300 of the 1380 (94.2%) local hijacks, the expected AS was announcing the detected advertisement at the time of the hijack. Since it is not obligatory to announce a prefix, and because it can be advantageous for an attacker to hijack an unused prefix, it is surprising to see such a high percentage. However, when a prefix is hijacked with the intention to route the traffic towards the detected AS the prefix has to be in use, otherwise there would be no traffic. Global hijacks announce a subnet of the expected prefix, so for these hijacks the number of MOAS conflicts is, as expected, low.

Another interesting point is the number of hijacks that have no ROA for the detected advertisement. In global hijacks this happens more than twice as often as in local hijacks. Local hijacks announce

the expected prefix and are thus more likely to be the result of a misconfiguration than global hijacks. It was discussed in subsection 6.3.12 that the percentage of prefixes with a ROA in hijacks does not differ much from the global distribution, and that it therefore does not seem that hijacks avoid prefixes with ROAs. But if global hijacks are more often intended hijacks and they have a significantly lower number of detected advertisements with ROAs, it may be that they do avoid these prefixes after all.

The last aspect that differs is that local hijacks are a bit more often labelled based on the relations between hijacks than global hijacks. However, global hijacks are three times as often labelled as a legitimate announcement. They are almost equally often labelled as an intended or path hijack.

6.3.16. MOAS conflicts

When multiple ASes announce the same prefix it is called a MOAS conflict. When an AS hijacks a prefix that is in use by another AS it causes such conflict. This is not always a problem. Sometimes it is intended that multiple ASes announce a prefix. There are 1879 (70.5%) hijacks of prefixes that were already announced at the time of the hijack. In 1321 (49.6%) cases the expected AS was announcing the prefix. This means that in 20.9% of the hijacks there was another AS announcing the detected advertisement. If these announcements are legitimate or not is unknown, but one may wonder why these ASes are not the expected AS. In 29 cases there is a MOAS conflict with an AS that is related to the detected AS, but neither the expected AS or an AS related to it is announcing the prefix. Most of these hijacks were labelled based on relations between hijacks and 14 were labelled as a legitimate announcement.

There does not seem to be a limit on the number of ASes that can announce a prefix. The most announced prefix is at the time of the hijack announced by 18 different ASes. There are 7 other hijacks with a MOAS conflict of 10 or more ASes but these hijack all have either AS25577 or AS26415 as the expected AS. The reason why so many ASes are announcing one prefix is unclear, but if all announcements are legitimate, it may well be that the detected AS is also allowed to announce that prefix.

6.3.17. Hijacks with an unallocated or reserved detected AS

This subsection discusses all hijacks that have a detected AS that was not or only just allocated by a RIR when the hijack started. In total, there are 76 (2.85%) of such hijacks. They are divided into different categories. The AS can be unallocated by IANA, reserved by IANA, reserved by a RIR, available by a RIR, allocated by a RIR after the hijack, and allocated by a RIR on the day of the hijack. Each paragraph in this subsection discusses one category.

Unallocated by IANA There are 11 hijacks that have an unallocated AS showing up as the origin. All these hijacks are withdrawn within a day and none of them have a continuous path. Seven of these hijacks happen at the same time and they all target prefixes owned by AS16509. It seems these hijacks are path hijacks. However, it is difficult to see which AS caused them, because the AS paths all contain different ASes.

Table 6.13 shows the start time, detected AS, AS path, path relations, and the number of BGPMon peers that received the route. If there is one AS that is causing these hijacks, it does not appear in all paths. One possible explanation is that an AS is sending false information to the monitors. There is a very low number of peers who receive these hijacks, but the highest number is still 10. Because there is no information about the BGPMon monitors, it is impossible to say if this is what happened. Because of the timing, it is unlikely that they are caused by multiple ASes, but that is another possible explanation.

Reserved by IANA The hijacks that have a detected AS that is reserved by IANA are similar to those with an AS that is unallocated by IANA. There are 27 of these hijacks. Almost all have a detected AS that is reserved for documentation and sample code and should thus not be announcing anything. As expected, none of the hijacks have a continuous path so most likely they are the result of a path hijack. The duration of the hijacks varies. Two last more than a month, but 11 are withdrawn the same day.

Start time	Detected AS	AS path	Path relations	BGPMon peers
2018-12-03 17:57:14 UTC	1000075508	46044 56246 1000075508	no relation no relation	3.0
2018-12-03 17:57:14 UTC	1000075508	46044 56246 1000075508	no relation no relation	3.0
2018-12-03 17:57:10 UTC	1161216244	28300 262589 9304 1161216244	c2p p2p no relation	10.0
2018-12-03 17:57:14 UTC	914325748	203507 33891 914325748	c2p no relation	6.0
2018-12-03 17:57:14 UTC	914325748	553 33891 914325748	c2p no relation	6.0
2018-12-03 17:56:47 UTC	995197172	31463 6762 995197172	c2p no relation	9.0
2018-12-03 17:57:06 UTC	1207234804	33891 1207234804	no relation	4.0

Table 6.13: Path hijacks with an unallocated AS as origin

Several hijacks that last longer than a day have a MOAS conflict with the expected AS. Because it is difficult to find the cause of these hijacks, they may have a lot of impact on the legitimate owner.

Reserved by RIR ASes that are reserved by a RIR seem to be used differently than ASes reserved by IANA. There are 16 hijacks that have a detected AS that was reserved by a RIR at the time of the hijack. Seven of these hijacks have a continuous path. This means that the detected AS has direct links to other ASes. One even has a ROA for the detected advertisement. What these ASes are reserved for is unknown, but it appears that an AS that is reserved by a RIR does not necessarily operate differently than allocated ASes.

Available by RIR Seven hijacks have a detected AS that was never allocated by a RIR according to the RIR statistics files. Similar to ASes that are reserved by RIRs, it looks like they do have direct relations to other ASes because some AS paths are continuous. There is also one AS that has a ROA for the detected advertisement and several who have announced the hijacked prefix at least once before the hijack. It is possible that the AS numbers are misused, but it could also be that allocation was never entered in the statistics file.

Allocated by RIR after or on the day of the hijack Because ASes that were available or reserved at the time of the hijack apparently could have been in use, it is not unexpected to see the same holds for ASes that were allocated after the hijack. There are 11 hijacks where this is the case. Most of these hijacks have a continuous path. The allocation date listed is in almost all cases months away from the hijack start time. It could be that the ASes were allocated before and reallocated at a later time. If the statistics file then only shows the later date, it may look like the AS was unallocated before.

Another four hijacks were caused by an AS that was allocated on the day of the hijack. It is possible that these hijacks are legitimate but that the data used by BGPStream was outdated. In one case, the detected AS has a ROA for the detected advertisement, but this AS had announced the prefix before the hijack and thus also before the day it was theoretically allocated.

It appears the RIR statistics files are inconsistent with the actual use of AS numbers. In the case of ASNs that are allocated after the hijack, it is possible that this was a reallocation and not the first allocation. However, it does not explain the use of available ASNs. It may be that someone forgot to record the allocations, or that the ASNs are misused by other ASes.

6.3.18. Hijacks with an unallocated or reserved expected AS

The previous subsection discussed the unallocated and reserved detected ASes. This subsection discusses the expected ASes. There are 44 (1.65%) hijacks that have an unallocated or reserved

expected AS. There are no expected ASes that were unallocated by IANA or were allocated on the day of the hijack, but expected ASes in the other four categories do appear.

Allocated by RIR after hijack There are 4 hijacks that have an expected AS that was allocated after the hijack. In all cases, this allocation is listed for months after the hijack. If this data is correct the ASes thus should not have been in use at the time of the hijack, unless they were already assigned and re-assigned later. Because this is not clear, not much can be said about these hijacks but three of the four hijacks had the expected AS listed as announcing the prefix at the time of the hijack. So it does seem they were in use long before their allocation was listed.

Available by RIR Five hijacks have an expected AS that is listed as available by a RIR. In three of these cases, the expected AS was announcing the prefix at the time of the hijack. Because the AS was not allocated, it should not have been in use. It is thus unclear why these ASes were announcing a prefix. It could be that the AS numbers were used in a path hijack and thus showed up as originating these prefixes, but it is not clear how BGPStream finds the expected AS of a prefix.

Reserved by RIR Hijacks with an expected AS reserved by a RIR form the largest group. There are 32 hijacks with this label. They have 17 different expected ASes that are allocated to either ARIN or RIPE NCC. Looking at the hijacks and all labels it seems that it is not abnormal for these ASes to be announcing prefixes. Many of these hijacks also have the label ‘MOAS expected AS’, meaning that the expected AS was announcing the prefix at the time of the hijack. Just as in the previous subsection, it seems that an AS reserved by a RIR is used differently than an AS reserved by IANA.

Reserved by IANA There are 3 hijacks with the expected AS reserved by IANA. Two hijacks have AS65551 as the expected AS. AS65551 is an AS that is reserved for use in documentation and sample code. It seems that this AS is (mis)used more often, because for one of the hijacks AS65551 is announcing the detected advertisement during the hijack. However, there exists a ROA for the detected advertisement listing the detected AS as the owner, and the detected AS has announced the prefix before. It is thus more likely that AS65551 is ‘hijacking’ the detected advertisement. It may be used in a path hijack of that prefix as the origin AS. The other hijack with AS65551 as the expected AS is similar. There is no ROA, but the detected AS has announced the detected prefix before. Both hijacks last more than a month and are likely both legitimate announcements.

The third hijack with an expected AS reserved by IANA is a hijack with AS11111 as the detected AS and AS123456 as the expected AS. These numbers in itself are interesting and what adds to the curiosity of this hijack is that the path is not continuous. There is a gap right before the detected AS. Most likely this hijack is not an origin hijack but a path hijack. It remains unclear why the reserved AS shows up as the expected AS.

6.3.19. Hijacks involving unallocated or reserved prefixes

The previous two subsections discussed unallocated and reserved ASes. This subsection will focus on unallocated and reserved prefixes. In the case of prefixes, there are four different categories. A prefix can be allocated after the hijack, reserved by a RIR, available or without status, or reserved for special purposes. Each paragraph in this subsection will discuss one category. There are 40 (1.50%) hijacks with a detected advertisement that falls in one of the four categories. These prefixes should not have an owner, so the fact that they show up in hijacks and therefore with an expected AS is unexpected.

Prefix allocated after the hijack There are 8 hijacks with a detected advertisement that was allocated after the hijack. However, in 4 cases the expected AS was also announcing the prefix, and in 3 other cases the prefix was announced before by the detected AS. Just as with the ASes that were allocated after the hijack, it is unclear if the allocation date in the RIR statistics files is a reallocation or if the prefix was misused. Seven of these hijacks last longer than a day and four of them last longer than a month.

Reserved and available prefixes There are 19 hijacks of a prefix that was unallocated or without status at the time of the hijack, and 13 hijacks of a prefix that is reserved by a RIR. These prefixes

were already in use or used before the hijack. It is unclear why these prefixes are used and as such not much can be said about these hijacks based on the fact that the prefix was not allocated.

Special prefixes In four hijacks the detected advertisement is a prefix that is reserved for special purposes. These four hijacks are caused by two ASes. The first AS hijacks the prefixes 2001:db8:8000::/36 and 2001:db8:9000::/36. These prefixes are a subnet of the prefix 2001:db8::/32 that is reserved for documentation. Both hijacks are withdrawn within a day.

The other two hijacks have 100.66.228.0/22 and 100.66.232.0/22 as the detected advertisement. These prefixes are subnets of the prefix 100.64.0.0/10 that is reserved for Shared Address Space. This prefix is intended for use on Service Provider networks. It is possible that these two hijacks are the result of a misconfiguration. Both are withdrawn within a day.

6.3.20. Detected AS announced hijacked prefix before

There are 455 (17.07%) hijacks of a prefix that was already announced for at least six weeks by the detected advertisement. These prefixes are thus considered normal announcements for the detected AS. It could be that the owner changed and that the detected AS is announcing a prefix it does not own anymore. It is also possible the AS has already hijacked the prefix for six weeks or longer, or that it is legitimately announcing the prefix.

Looking at other labels shows that a large number of these hijacks are already labelled as a legitimate announcement. While looking for failures to summarize, 5 of these hijacks received the label ‘legitimate announcement’. Another 37 hijacks have a ROA listing the detected AS as owner of the detected advertisement. Based on relations between hijacks, 55 hijacks were also labelled as legitimate. In total there are thus 97 of these hijacks already labelled as legitimate announcements.

The duration of these hijacks strengthens the idea that most of these hijacks are likely legitimate announcements. Only 41 are withdrawn within a day while 377 last more than a month. In the complete data set, there are 440 hijacks lasting longer than a month and 1516 that are withdrawn within a day. In terms of distribution, this is thus the complete opposite.

6.3.21. AS path is not continuous

There is a very large number of hijacks that have a path that is not continuous. These 585 (21.95%) hijacks are detected as possible origin hijacks and list the first AS in the AS path as the cause of the hijack. After analysing the related hijacks and some of the labels discussed above, it became clear these hijacks may very well be path hijacks. This is an issue because it is difficult to detect who causes these hijacks.

In most cases the hijacks don’t last long. For 388 hijacks the announcements were withdrawn within a day. In 97 cases the hijack lasts longer than a week. If this is causing problems for the owner of the prefix it would be difficult to solve because it is unclear who is causing the hijack. Especially for global hijacks (252 in this data set) this may have a lot of impact. It is thus important to implement protection against path hijacks in BGP. If RPKI would be used globally it would prevent path hijacks that resemble an origin hijack, but currently it does not seem that ROAs discourage these hijacks. The percentage of hijacks of prefixes with a ROA is 21% for the hijack without a continuous path, and 25% of all hijacks. These percentages are comparable to the global use of RPKI [25].

6.3.22. Hijack duration

Mahajan et al. [45] showed that misconfigurations and hijack attempts mostly last less than 24 hours. The duration of the possible hijacks in this data set is divided into four categories: less than a day, up to a week, up to a month, and more than a month. These categories are mutually exclusive. The hijack duration was already mentioned in subsection 6.3.20. Most of the hijacks that had a detected advertisement that was considered a normal announcement for the detected AS lasted more than a month.

In this data set there are 1516 (56.89%) hijacks that last less than a day. There are 16 hijacks with a ROA for the detected advertisement and the detected AS. For hijacks that last longer than a month, there are 37 out of 440 hijacks with such a ROA, so it does seem that legitimate announcements indeed last a lot longer in most cases. However, looking only at announcements that last less than a day is not a good way to detect hijacks. There are 479 hijacks that last between a day and a week, and 217 that last more than a week but less than a month. These are definitely not all legitimate announcements.

The duration does not tell much about the type of hijack. Unless it was specifically mentioned, for all labels discussed above the hijack duration varied from less than a day to more than a month. However, it does appear that legitimate announcements often last longer.

6.3.23. Hijack of a prefix that follows a pattern

This last subsection discusses hijacks of prefixes that have 3 or more equal octets. There are 49 (1.84%) of these hijacks caused by 29 ASes. There is one AS, AS36937, that causes 11 of the hijacks. This group was already discussed in subsection 6.2.3. Except for two hijacks, all are withdrawn within a day and otherwise within a week. The hijacks happen over the course of a year and it does not seem they are all related. It could be that there is a reason the detected ASes all chose to hijack this type of prefix but it may also be a coincidence. After all, there is nothing wrong with these prefixes.

6.3.24. Summary

This summary starts with a table that lists each label and the number and percentage of hijacks that received that label. It follows with a discussion on which labels were most useful, and which labels did not provide much information. The summary ends with a list of the most important findings of this section. Table 6.14 shows the number and percentage of hijacks that received a certain label.

Label	Number of hijacks	Percentage of hijacks
Invalid hijacks	13	0.49%
Legitimate announcements	241	9.04%
Hijack of one IP address	302	11.33%
Failure to summarize	1	0.04%
Possible typo	66	2.48%
Wrong prefix length	7	0.26%
Hijacking customer route	122	4.58%
Blacklisted before hijack	182	6.83%
Blacklisted during hijack	100	3.75%
Hijack of unused prefix	43	1.61%
Announcing subnet	42	1.58%
Announcing supernet	15	0.56%
ROA for detected advertisement	360	13.51%
AS23456	2	0.08%
Path prepended by origin AS	142	5.33%
MOAS conflict	1879	70.51%
Hijack with unallocated or reserved detected AS	76	2.85%
Hijack with unallocated or reserved expected AS	44	1.65%
Hijacks with unallocated or reserved prefix	40	1.50%
Detected AS announced prefix before	455	17.07%
AS path is not continuous	585	21.95%
Detected advertisement has pattern	49	1.84%

Table 6.14: Number and percentage of hijacks that received a certain label

Labelling hijacks started with filtering out invalid hijacks and finding legitimate hijacks. Legitimate hijacks were found based on ROAs and their relation to other hijacks. This approach allowed 9.04% of the hijacks to be labelled as legitimate. Because most were labelled based on their relation to other hijacks it would be difficult for a detection system to immediately recognise them as legitimate. This requires up-to-date information about which AS owns which prefix. Unfortunately, this is only partly

available in the form of ROAs.

The other labels can be roughly divided into two categories. One category consists of labels that specify the cause of a hijack, the other category contains labels that indicate a certain characteristic in hijacks. The labels that specify the cause of a hijack are ‘failure to summarize’, ‘possible typo’, ‘wrong prefix length’, ‘hijacking customer route’, ‘announcing subnet’, and ‘announcing supernet’. These labels are useful because they indicate a direct cause. All of these labels can be given to a hijack when it is detected because they use information that is already available at that point. Using these labels when detecting hijacks would make it easier to solve them because the cause is immediately clear.

The labels that indicated a characteristic of hijacks were not always helpful. For example, labels based on IP blacklists did not provide much information because they didn’t specify the exact source of the spam. It is also still unclear why ASes announce only one IP address. On the other hands, looking at ROAs for the detected advertisement of a hijack helped to find legitimate hijacks. In addition, looking at unallocated and reserved resources showed that there might be some issue with the way BGPStream detects the expected AS, and that the status listed in a RIR statistics file is not always in agreement with the use of resources.

Another useful label is the one that indicates if the hijacked prefix is a normal announcement for the detected AS. This label can help to find legitimate announcements and is based on information that is already available at the start of a hijack. Lastly, finding hijacks with AS paths that are not continuous helped to identify many potential path hijacks. These path hijacks are detected as origin hijacks because the first AS in the path is not the expected AS. However, it is also possible to perform a path hijack with the expected AS as the origin of the path. It is thus possible that there are many more path hijacks than just those found in this data set.

Following is a list of most important findings of this section.

- Almost 10% of the hijacks could be labelled as a legitimate announcement and there many more are suspected to be legitimate. This raises some questions about the detection system of BGPStream. Although they specify that the detected hijacks are possible origin hijacks, it is still a large amount.
- There are 33 ASes that cause hijacks of one IP address. In total these hijacks form 11.33% of the data set, but the reason for these hijacks is unclear.
- Several hijacks are caused by mistyping a prefix. Forgetting a digit seems to be especially common. Because these hijacks could be easily prevented, it is worth looking into developing a system that helps to configure routers.
- It is difficult to relate the blacklist data to the hijacks. Even if the detected advertisement is blacklisted during the hijack, the detected AS may not be the cause of the spam. Especially when a hijack lasts longer, the chance increases that another AS will misuse the prefix. Because prefixes can be announced by multiple ASes it is never certain who is causing the blacklisting.
- In section 3.3 was discussed that unused prefixes may be misused for malicious purposes. In this data set, there are 43 hijacks of unused prefixes. Only one has the detected advertisement blacklisted during the hijack. This does not mean that these prefixes are not misused, because there are other ways to misuse a prefix besides sending spam.
- Almost all hijacks that had the detected AS announcing a supernet of one of its own ASes were legitimate announcements. This is not necessarily always the case, but it does seem that hijacking supernets is not a real issue.
- It does not seem that ROAs keep prefixes from being hijacked, but because ROAs do not contain a creation date this cannot be said with certainty.
- In global hijacks it happens twice as often that there is no ROA for the detected AS. If global hijacks are more often intended hijacks, it could be that prefixes with a ROA are avoided after all.

- It is unclear how BGPStream determines the expected AS of a prefix. It seems that in some cases the expected AS is not supposed to be announcing the expected prefix. If the expected AS is not reliably determined, this has an impact on the reliability of the features and labels that use this AS.
- Another concern with the reliability of the data set is that BGPMon peers may send false information to the monitors. If a hijack is only received by a few peers, it is possible that the hijack never happened.
- RIRs keep statistics on which ASNs are allocated and when they were allocated. It seems that ASNs that are reserved or available are still being used. It is not clear why this is the case. It could be that the allocations were never entered in the statistics files, but it is also possible that ASNs are misused by other ASes.
- Some hijacks have a detected advertisement that is reserved or unallocated. These prefixes should not be used and there should thus not be an expected AS. It appears these prefixes are in use, meaning there is nothing to prevent an AS from announcing these prefixes.
- Many hijacks in the data set are likely path hijacks. It is difficult to find the cause of these hijacks, because the detected ASes are not announcing them. The hijacks in this data set look like origin hijacks, but it is also possible to perform a path hijack without changing the origin. It is therefore likely that there are much more path hijacks happening.
- Looking at the duration of the hijacks and the labels the hijacks received, it appears that legitimate announcements indeed last longer than hijacks.

7

Conclusion

The goal of this thesis was to analyse a year of possible BGP origin hijacks detected by BGPStream. In the introduction of this thesis it was mentioned that the intention was to find out if it is possible to detect the cause of origin hijacks, to find out what is needed to prevent them, to give an overview of the most common causes and to provide insight in any patterns that occur over the year. This can also help to differentiate between intentional hijacks and misconfigurations, and to determine if all detected hijacks are indeed hijacks. The reason to do this was that research on BGP security often focusses on securing the protocol and on detecting anomalies. Research analysing BGP anomalies is less common. Analysing BGP origin hijacks can give insight in the causes and characteristics of these hijacks, which may help to find the most pressing issues and provide guidance in securing BGP.

The possible origin hijacks that were detected by BGPStream between 20 May 2018 and 31 May 2019 formed the basis of the data set. The process of going from this basic data set to the results presented in the previous chapter consisted of the following steps:

- Feature engineering
- Finding relations between hijacks
- Labelling hijacks
- Generating results

The features, relations between hijacks, and labels were used to find hijacks that shared certain characteristics. Analysing groups of similar hijacks proved to be very effective. Using the context of a group gives a lot more insight than looking at hijacks individually. This approach helped to find ways to detect several causes of hijacks such as hijacks of customer routes. Using these labels when detecting hijacks can help to solve the problem quickly. Combining these labels and the labels given to hijacks based on relations to other hijacks may also help to create a labelled dataset that would allow for further analysis using machine learning. It should be noted, however, that the labels provide a likely cause but it cannot be said with complete certainty what truly happened and it is even more unsure if a hijack is caused intentionally or by a misconfiguration. It is also possible that there are other causes that were not mentioned in this thesis.

Another disadvantage of this approach is that it relies on the availability of other hijacks. When a hijack is the first in a group of related hijacks, or when there are no related hijacks, it is impossible to use the context of a group to determine the cause. Additionally, for some labels information is required that may not yet be present at the time of the hijack. Without this information it is not possible to use this approach to build a real-time detection system. However, ASes may have the required information for their own AS so that it might be possible to detect misuse of their own resources.

Aside from finding the cause of hijacks, this approach also helped to find legitimate announcements

and path hijacks in the set of possible origin hijacks. It is important to be able to make this distinction because legitimate announcements should not be detected as an anomaly, and detecting and solving path hijacks requires a different approach. This shows that it is important to have a detection system that is accurate and able to correctly identify anomalies. As this data set already has a large number of possible path hijacks it is important to have a system that is able to detect these hijacks. This data set only includes path hijacks that have an unexpected origin, and only those with non-continuous AS path are labelled as such. However, path hijacks do not necessarily have a gap in their AS path and it is perfectly possible that a path hijack has the expected AS as the origin of the AS path. It is thus likely that there are many more path hijacks than those detected using our approach.

During the analysis, some questions arose about the detection system used by BGPStream. It is unclear how the expected AS of a hijack is determined. It seems that in some cases the expected AS should not have been announcing the prefix, and should thus not be listed as the expected AS. Since some of the labels and relationships are dependent on the expected AS the resulting conclusions may be incorrect. Having a reliable source that lists the ASes that can legitimately announce a prefix would be very helpful for research and for detecting hijacks.

Aside from BGPStream's detection algorithm, the placement of the BGPMon's monitors is also a cause for concern. The hijacks in our data set are those detected by BGPStream, but it is unknown how many hijacks are not detected. This thesis may thus not provide a full overview of all hijacks that happened during the year. In addition, it is possible for BGPMon peers to send incorrect information to the monitors. Depending on how often this happens, this could have some influence on the results. However, this only influences the number of hijacks receiving a certain label. It does not affect the approach of detecting relations between hijacks or labelling hijacks. This also holds for the accuracy of the data sets used to compute the features for each hijack. The accuracy of detection systems and algorithms to label hijacks both depend on the quality of the available data sets but the approach of detecting and labelling hijacks does not.

Securing BGP in such a way that origin hijacks are not possible would solve many problems. Until then, the focus should be on avoiding misconfigurations and quickly detecting and solving problems. Avoiding misconfigurations requires a system that allows for easier configuration of ASes so easily avoidable misconfigurations, such as typographical errors, are no longer a problem. Unfortunately, this does not prevent hijacks that are caused intentionally. It is thus important to have a system that can accurately detect hijacks and the AS causing the hijack. The approach taken in this thesis can help to find a way to distinguish between origin hijacks, path hijacks, and legitimate announcements and to find a possible cause of a hijack. However, any detection system is completely depended on the quality of the data used. It may thus be useful to focus on having a publicly available accurate database that combines all necessary BGP data.

Bibliography

[1] Autonomous system (AS) numbers. URL https://www.iana.org/assignments/as-numbers/as-numbers.xhtml.

[2] BGPmon, . URL https://bgpmon.net/.

[3] BGPStream, . URL https://bgpstream.com/.

[4] The CAIDA AS organizations dataset, 1 April 2018 – 1 April 2019, . URL http://www.caida.org/data/as-organizations.

[5] The CAIDA AS relationships dataset, 1 May 2018 – 1 June 2019, . URL http://www.caida.org/data/active/as-relationships/.

[6] GeoLite2 free downloadable databases. URL https://dev.maxmind.com/geoip/geoip2/geolite2/.

[7] Internet Assigned Numbers Authority. URL https://www.iana.org/.

[8] Free IP geolocation database | IP2Location LITE. URL https://lite.ip2location.com/.

[9] Country codes - ISO 3166.

[10] Lacnic. URL https://www.lacnic.net/.

[11] RIPE network coordination centre, . URL https://www.ripe.net/.

[12] RIPE Atlas, . URL https://atlas.ripe.net/.

[13] RIPE database, . URL https://www.ripe.net/manage-ips-and-asns/db.

[14] RIPEstat data API, . URL https://stat.ripe.net/docs/data_api.

[15] Routing Information Service (RIS). URL https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris/routing-information-service-ris.

[16] RPKI Validator. URL https://rpki-validator.ripe.net/trust-anchors.

[17] RIR statistics exchange format. URL https://www.apnic.net/about-apnic/corporate-documents/documents/resource-guidelines/rir-statistics-exchange-format/.

[18] Internet protocol. RFC 791, Information Sciences Institute, September 1981. URL https://tools.ietf.org/html/rfc791.

[19] Transmission control protocol. RFC 793, Information Sciences Institute, September 1981. URL https://tools.ietf.org/html/rfc793.

[20] OpenDNS announces security alert tools BGP Stream and DNS Stream, 2015. URL https://umbrella.cisco.com/blog/2015/08/06/opendns-announces-alert-tools-bgp-stream-and-dns-stream-at-blackhat/.

[21] AFRINIC the region internet registry (RIR) for africa, 2018. URL https://afrinic.net/.

[22] APNIC, 2019. URL https://www.apnic.net/.

[23] American registry for internet numbers, 2019. URL https://www.arin.net/.

[24] BGP prefix report, July 2019. URL https://bgp.he.net/report/prefixes#_prefixes.

[25] RPKI deployment monitor, August 2019. URL https://rpki-monitor.antd.nist.gov/.

[26] Spamhaus, 2019. URL https://www.spamhaus.org/.

[27] What is the timeliness of the data?, 2019. URL https://stat.ripe.net/faq#data-timeliness.

[28] UCEPROTECT, 2019. URL http://www.uceprotect.net/.

[29] World report, August 2019. URL https://bgp.he.net/report/world.

[30] Bahaa Al-Musawi, Philip Branch, and Grenville Armitage. BGP anomaly detection techniques: A survey. *IEEE Communications Surveys & Tutorials*, 19(1):377–396, 2017.

[31] Oleg Bulkin. StringDist PyPI, May 2017. URL https://pypi.org/project/StringDist/.

[32] Kevin Butler, Toni R. Farley, Patrick McDaniel, and Jennifer Rexford. A survey of BGP security issues and solutions. *Proceedings of the IEEE*, 98:100–122, 2010.

[33] Fred J. Damerau. A technique for computer detection and correction of spelling errors. *Communications of the ACM*, 7:171–176, 1964.

[34] S. Deering and R. Hinden. Internet protocol, version 6 (IPv6) specification. RFC 8200, Internet Engineering Task Force (IETF), July 2017. URL https://tools.ietf.org/html/rfc8200.

[35] Nick Feamster and Hari Balakrishnan. Detecting BGP configuration faults with static analysis. In *NSDI'05*, volume 2, pages 43–56, May 2005.

[36] V. Fuller and T. Li. Classless inter-domain routing (CIDR): The internet address assignment and aggregation plan. RFC 4632, The Internet Society, August 2006. URL https://tools.ietf.org/html/rfc4632.

[37] Vasileios Giotsas and Shi Zhou. Valley-free violation in internet routing - analysis based on bgp community data. *IEEE ICC 2012 - Communication QoS, Reliability and Modeling Symposium*, 2012.

[38] Vasileios Giotsas, Shi Zhou, Matthew Luckie, and kc claffy. Inferring multilateral peering. *ACM SIGCOMM Conference on emerging Networking EXperiments and Technologies (CoNEXT)*, pages 247–258, December 2013.

[39] J. Hawkinson and T. Bates. Guidelines for creation, selection, and registration of an autonomous system (AS). RFC 1930, March 1996. URL https://tools.ietf.org/html/rfc1930.

[40] Geoff Huston. BGP in 2018 - part 1 - the BGP table, January 2019. URL https://www.potaroo.net/ispcol/2019-01/bgp2018-part1.html.

[41] Franck Le, Sihyung Lee, Tina Wong, Hyong S. Kim, and Darrell Newcomb. Detecting network-wide and router-specific misconfigurations through data mining. *IEEE/ACM Transactions on Networking*, 17(1):66–79, February 2009.

[42] M. Lepinski and S. Kent. An infrastructure to support secure internet routing. RFC 6480, February 2012. URL https://tools.ietf.org/html/rfc6480.

[43] Matthew Luckie, Bradley Huffaker, Amogh Dhamdhere, Vasileios Giotsas, and kc claffy. AS relationships, customer cones, and validation. *Internet Measurement Conference (IMC)*, pages 243–256, October 2013.

[44] Doug Madory. Large european routing leak sends traffic through china telecom, June 2019. URL https://blog.apnic.net/2019/06/07/large-european-routing-leak-sends-traffic-through-china-telecom/.

[45] Ratul Mahajan, David Wetherall, and Tom Anderson. Understanding BGP misconfiguration. In *SIGCOMM '02*, pages 3–16, 2002.

[46] Asya Mitseva, Andriy Panchenko, and Thomas Engel. The state of affairs in BGP security: A survey of attacks and defenses. *Computer Communications*, 124:45–60, 2018.

[47] Y. Rekhter. A border gateway protocol 4 (BGP-4). RFC 1654, July 1994. URL `https://tools.ietf.org/html/rfc1654`.

[48] Y. Rekhter, T. Li, and S. Hares. A border gateway protocol 4 (BGP-4). RFC 4271, January 2006. URL `https://tools.ietf.org/html/rfc4271`.

[49] Stephen Strowes. Visibility of ipv4 and ipv6 prefix lengths in 2019, April 2019. URL `https://labs.ripe.net/Members/stephen_strowes/visibility-of-prefix-lengths-in-ipv4-and-ipv6`.

[50] Olaf van Miltenburg. Storing bij provider leidt tot landelijke pinproblemen in nederland - update 2, June 2019. URL `https://tweakers.net/nieuws/153678/storing-bij-provider-leidt-tot-landelijke-pinproblemen-in-nederland.html`.

[51] Simone van Veen. Master thesis: Analysing BGP Origin Hijacks, 2019. URL `https://gitlab.com/svveen/msc_thesis_code`.

[52] Christian Veenman. Detecting BGP origin hijacks. Master's thesis, TU Delft, January 2019.

[53] Pierre-Antoine Vervier, Quentin Jacquemart, Johann Schlamp, Olivier Thonnard, Georg Carle, Guillaume Urvoy-Keller, Ernst Biersack, and Marc Dacier. Malicious BGP hijacks: Appearances can be deceiving. *IEEE International Conference on Communications (ICC)*, June 2014.

[54] Pierre-Antoine Vervier, Olivier Thonnard, and Marc Dacier. Mind your blocks: On the stealthiness of malicious BGP hijacks. In *Network and Distributed System Security Symposium*, January 2015.