Hacking the Internet: How BGP paves the way for attackers

Tim Sauer
tim.sauer@tu-bs.de
TU Braunschweig

Abstract

The Border Gateway Protocol (BGP) is connecting a permanently increasing number of routers all around the globe, building the Internet. Since it has been first drafted in 1989 there have been no attempts on securing it during this time. This leaves space for a high number of flaws, that have been exploited multiple times in history. In this paper I give an overview on different kinds of BGP based attacks, how they can be performed and which flaws they exploit. I also provide a deeper insight into BGP Communities. Those strings are used by Internet Service Providers to implement their routing policies, but can also be misused by attackers. To wrap it up I present two kinds of specific attacks that make use of the previously introduced concepts. The first attack compromises the anonymity of Tor, while the second one attacks DNS Servers to misguide Internet users.

Keywords: Autonomous System, BGP, Communities, Interception, Hijack, Exploits

1 Introduction

This paper presents different flaws of the Border Gateway Protocol (BGP) and how they can be exploited to cause severe damage for Internet Service Providers (ISP) and even everyday users. But these attacks do not only happen with malicious intent. History has shown several incidents where misconfigurations lead to unreachability of Websites or even entire networks. In 2008 Pakistan Telecom tried to restrict access to YouTube for Pakistan based users. This went wrong and they accidentally announced wrong routing information, which made YouTube unreachable worldwide for about two hours. [13, 14] A few years earlier, in 2004, a Turkish ISP claimed that it was the origin of every single IP-address of the Internet. So most Internet traffic was routed towards this ISPs servers for several hours. This lead to a traffic loss for a high amount of (also high ranked) Websites like Amazon, Microsoft, Yahoo and CNN. [4, 13] To get insight into those attacks I will first explain the underlying structure of the Internet Backbone and how traffic is routed. (Section 2) In the next section I present, how routing of Internet traffic can be influenced and how this leaves flaws that can be exploited by attackers. This section also presents multiple different methods that have been used in the past to compromise Internet structure. For instance the two incidents I described earlier, concerning Turkey and Pakistan, can be considered BGP hijacking attacks. Those hijacking attacks lead to a paper that has been published last year. With SICO [3] the authors exploit several techniques that were originally intended for engineering of routing structures. Doing so enables them to intercept Internet traffic and forward it to its original destination, which makes those kinds of attacks really hard to detect. Based on that I take a closer look at two kinds of attacks that have been performed in the past, several times. The first kind of those attacks is the deanonymization of Tor users. Revelations by Edward Snowden say that the NSA has actively collected metadata of Tor users to uncover their identities, based on BGP attacks. [2] The second kind of attacks I present is the manipulation of DNS servers. In the past it has happened multiple times that attackers have rerouted users to manipulated Websites, to fish for login data and other sensitive information. [12]

2 Background

In this section I will explain basic concepts of how the Internet Backbone is built.

2.1 Autonomous System

Since the Internet is a decentralized system it is built by a permanently increasing number of routers that are connected with each other. Several routers that share the same IP-prefix and are under the administration of a common organization (e.g. Internet Service Providers, Scientific Institutions) form an Autonomous System (AS).
In order to identify ASes within the Internet, each one possesses a unique identifier called 'Autonomous System Number' (ASN).

2.1.1 Business Relationships between ASes. In their paper from 2001 Gao et al. present a model of how ASes exchange data based on business relationships [11]. These relationships are caused by contracts between AS owners, which define pricing for traffic exchange. There are three roles ASes can take in relationship to other ASes.

1. Customer:
   A customer is usually a smaller AS that pays a larger AS for routing its traffic from or to other ASes.
2. Peer:
   Two ASes are peers to each other (peering ASes) if they have typically the same size and both benefit from routing traffic over each other and to each others clients.

3. Provider:
   A provider is a larger AS which is paid for routing a smaller ASes (customers) traffic.

2.2 Border Gateway Protocol

To build up the Internet ASes have to communicate with each other. The Border Gateway Protocol (BGP) is being used for this information exchange.
To communicate, an AS announces its IP-prefix and ASN to its direct neighbors. A neighbor receives this so-called 'BGP-Announcement' and decides if it prefers this way to exchange information with the AS it learned the BGP-Announcement from. If it does so, it adds its own IP-prefix and ASN to the BGP-Announcement and passes it on to its own neighbors. By doing this process again and again, a BGP-Announcement grows in its number of ASes and therefore represents a so-called 'AS-Path', which is simply a route between ASes.

2.2.1 Selection of routes. Note that an AS can learn different AS-paths from its neighbors that lead to the same IP-prefix, at the end of the path. In this case the AS has to decide which path to choose. There are several criteria an AS considers when it decides which path it prefers.
The first of these criteria is 'local preference'. To decide which path to prefer, an AS compares which business relationship it has to the neighbors it learned the conflicting paths from. Paths learned from customers are preferred over the ones learned from peers. Paths learned from peers are preferred over the ones learned from providers.
The second criteria is the 'path-length'. If an AS can't make a decision based on local preference (e.g. because all ASes originating the conflicting paths are providers), it decides based on the AS-Path length. So it simply chooses the route which has the lower number of ASes, along the path to the AS with the destined IP-prefix.
If the second criteria still does not break the tie, the AS chooses a path based on AS-internal routing policies, which I will not discuss in this paper as I am focusing on exterior routing between ASes and not on AS internal routing policies.

2.2.2 BGP Communities. BGP Communities are optional strings that an AS can add to a BGP-Announcement. There are two kinds of communities that are used by ASes.
The first kind are 'information communities'. Those are informational strings that are used by ASes to exchange routing information with each other.
The second kind are 'action communities'. Those strings are used by ASes to trigger actions at other ASes further down the AS-Path. Actions can be almost everything, but the addressed AS has to support and accept the community, in order to run this action. There is only a small subset of BGP Communities that is supported by a wide range of ASes. Most communities used around the Internet are provider specific. For example ISPs use individual action communities to manage routing of their ASes. Figure 1 presents three communities that are supported by most of the top 10 Internet exchanges.

LowerPref	**Lower local-preference below peer:** Allows a customer to lower the local preference of its routes below default local preference of peer routes. For instance, if this community is applied on a customer C and a provider has to select a route like described in 2.2.1, it would prefer the route learned from a peer over the one learned from customer C.
NoExportSelect(X)	**No export to selected peer:** Causes a provider to not export a route to a specified AS X. (specified by ASN)
NoExportAll	**No export to all peers:** Causes a provider to not export a route to any of its peers, but only to its customers.

Figure 1. Three BGP Communities most of the top 10 Internet exchanges support.

3 Related Work

This section gives an overview on different kind of flaws in the Internet Backbone. I divide it into two subsections. First I will look at insecurities opened up by BGP Communities, based on Streibelt et al.s paper 'BGP Communities: Even more Worms in the Routing Can'. [18] The second part of this section discusses flaws which exist in BGP itself, based on Nordström et al.s paper 'Beware of BGP attacks'. [16]

3.1 Flaws of BGP Communities

The three communities presented in Section 1 belong to the small set of communities which are well-known and standardized around the Internet. Besides to the variability of purposes that communities are used for, the lack of a clear documentation states a problem while trying to understand the usage of communities. Furthermore, there are no strict policies on how ASes should handle incoming communities in terms of forwarding or dropping them.
Streibelt et al. summarize the lacks of communities in two shortcomings:

1. Missing Semantics: Only very few communities have standardized semantics. The semantics of the rest of them varies from AS to AS, as well as the order in which they are executed, if multiple of them come along the same BGP-Announcement. Also, the visibility of communities e.g. if they are visible to only peers or customers and peers varies.

2. No authentification of tagger/community: Any AS of a BGP-Announcement can remove or update any community that already comes with the announcement and can also add additional ones. The receiver of the announcement is not able to determine, which AS along the path made adjustments to which community.

3.1.1 Propagation of BGP Communities.

To study propagation of communities the authors rely on multiple widely-used public datasets from April 2018, that have been collected by so-called 'route collectors'. These route collectors each contain multiple routers that collect all BGP announcements, they get passed by their neighbors.
Looking at this data, one key insight they made was, that almost 50% of the communities travel more than 4 ASes far. Compared to the average path length between every possible tuple of ASes around the Internet, which is 4.06 [22], this seems to be a large distance.
They also observe that 14% of Transit Providers[1] forward received communities. This number seems low but given the amount of links between ASes, they consider this a sign that communities propagate globally.

3.1.2 Running Community Experiments in the Wild.

To validate their insights from Section 3.1.1 they ran several tests on the real Internet Backbone. To not harm users, they used the PEERING Testbed [21] which allows researchers to run BGP experiments under real circumstances. They also partnered with multiple AS providers, who explicitly gave them permission to use their infrastructure as part of their experiments.
Specifically they ran four experiments to find out how communities are forwarded and executed by ASes. Of those four experiments two also look at the impact of using communities as a malicious tool.

Propagation Checking
The authors looked if ASes rather forward or drop communities they do not know and therefore can not execute. To do so, they announced communities they did not see in the data from route collectors they studied in section 3.1.1 over both, the PEERING testbed and one of the partnering ASes. The partnering AS announced the community over two providers from which they discovered only one to forward the unknown community to its neighbors. When reaching Transit Providers seven of them also forwarded the communities further.
In contrast to that, the AS in the PEERING testbed had a far higher number of pairing peers. So it gave better insight in how communities spread over a high number of paths. Observing communities they announced over the PEERING testbed, they found out that within 30 minutes after doing the announcement, more than 50 Transit Providers forwarded the announcement. Within the time of a day more than 112 Transit Providers (out of 434 that could be observed) saw the announced community.

Remotely Triggered Blackholing
Remotely triggered Blackholing (RTBH) is a technique that enables the option to drop undesirable traffic, before it reaches a specific AS. [20] One example of a benefit of RBTH is in case of a DDoS attack, performed towards an AS A. To free itself from the high level of traffic A can drop all traffic destining it. To do so it adds a RBTH community to the BGP-Announcement it sends to its neighbors. This community causes its neighbors to drop all incoming traffic, that is routed towards A.
This method can also be used maliciously to attack a victims incoming traffic. An adversary could announce not its own, but the victim's IP-prefix, along with the RBTH community, causing other ASes to drop traffic routed towards the victim. This is possible because ASes do not validate, if the announced prefix also belongs to the AS it was announced by.
After performing this kind of attack in the wild, Streibelt et al. conclude that 'RBTH is the easiest scenario to realize in the wild, independent of hijacking.'.

Traffic Steering
For Internet Service Providers (ISP) it is an important task to control, how traffic between their ASes is routed. To do so they use several communities. Consider an ISP which uses two intercontinental links between Europe and Asia, one being a peering link and the other leading to a provider. In this situation the ISP prefers traffic being routed over the peering link. This results in far lower costs than routing traffic via the provider link. Using the provider link it would have to pay for every routed package. The ISP can now use communities to make its ASes route traffic over the cheaper link.
Again, since ASes do not validate the origin of BGP Communities, an attacker could now reverse the communities the ISP just used to adjust its routing. By doing so, the ISPs traffic would now take the much more expensive link. This would cause severe financial damage for the ISP.
To test traffic steering in the wild they used one AS as their target, for which they were aiming to apply communities. Those communities were supposed to change the local preference of the target. Therefore, neighbored ASes of the target would prefer/neglect routes being announced by the target AS over/under the ones being announced from other ASes. They found a path from their own AS via another AS to their target in which the middle AS was a customer of the target AS. This made up a desirable situation for testing traffic steering, because BGP Communities are only applied along links from customers up to providers.

[1] 'An Internet Transit Provider is an ISP that provides transit to customers as a paid transport service.' [1]

Attaching communities to BGP announcements they advertised from their AS, towards the target AS, they were able to observe these communities arriving at the target AS. They compared the different routes, Internet traffic took, before and after applying communities. Doing so they observed that these routes differed. They considered this a success, performing traffic steering based on BGP Communities. Despite the success running this experiment, they conclude that Traffic Steering is really hard to launch. They reason, that this is caused by business relationships ASes have to each other, which often make it hard to apply communities, allowing Traffic Steering.

Route Manipulation
Besides Traffic Steering another feature of communities can be exploited maliciously. Consider the following situation: An AS gets two conflicting communities, for example one that prevents it from exporting an AS-Path to neighboring systems and one that forces it to export the same AS-Path to neighboring systems. Then the AS parses through a list that documents the execution order of communities. Now there is an ISP using a specific community to perform Traffic Steering between its ASes. If an adversary is able to add a conflicting (bogus) community to BGP announcements that traverse the ISPs ASes, these ASes decide how to route traffic, based on their execution order for communities. If the bogus community has a higher priority than the valid one, the adversary is able to destroy the ISPs routing structure.
To perform route manipulation in the wild, the authors used a well known AS which provides information about its execution order for communities, as their target. This AS also gives insight into its routes, as well as in its supported communities.
They first sent out an announcement towards this target AS, along with a community that instructed the target to forward its announcement to another specific AS. Observing announcements at the target AS they could discover the announcement they just made, along with the community they had attached. In a next step they added a second community, which conflicted with the first one added. Discovering the announcement at the target again, they found out that the first community was not executed anymore. Therefore, they were able to exploit the evaluation order of the target AS.
They conclude that an adversary could perform this kind of attacks as well.

3.2 BGP Based Attacks

Next to BGP Communities, also the Border Gateway Protocol itself leaves room for a huge spectrum of BGP based attacks. Since it was first drafted in 1989 there have been no attempts towards security during this time.

3.2.1 AS-Path Poisoning. In AS-Path Poisoning an adversary poisons a victim's AS so that other ASes will not route traffic over the victim's AS. To do so, AS-Path Poisoning exploits BGPs loop-prevention mechanism.
For instance if an adversary (AS A) wants to poison AS B, it not only adds itself with its ASN and IP-prefix to BGP announcements, but also AS B with its information. By doing so it seems that B has already been visited. If the announcement now actually reaches B, it will be dropped because there seems to be a loop in the AS-Path. This happens to every announcement that contains the poisoned AS B. Thus, no traffic will be routed over B anymore.
It is not only possible to poison single ASes but also entire AS paths. This simply means that the adversary poisons each AS along this path.

3.2.2 BGP hijacking attacks. In their paper from 2004 [16] Nordström et al. present several kinds of BGP Attacks. They consider 'prefix hijacking' one of the most straightforward types of BGP attacks. To exploit hijacking attacks, an adversary has to fully control at least one AS. The adversary's goal is to hijack another ASes incoming traffic. There are two ways how an adversary can reach this goal.

1. The adversary announces to originate the IP prefix of a system, whose traffic it wants to hijack (victim). This announcement spreads via the adversary's neighbors. By doing so, it attracts traffic that is originally routed towards the victim.
2. The adversary announces, to have a direct connection to the victim's AS. This announcement spreads via the adversary's neighbors. Other ASes are likely to prefer the adversary's AS, to route their traffic towards the victim's AS. They do so, because the adversary seems to have a short route to the victim.

Either way the adversary announces an AS path that does not exist. Such a path is called 'bogus path'.
If the adversary is able to attract traffic that is routed towards the victim in any way, it will be in full control of it. It can now use the data for whatever it likes. Afterwards the adversary either drops the traffic, or forwards it to the victim.

3.2.3 BGP Interception Attacks. BGP Interception Attacks are a special kind of hijacking attacks. In these cases the adversary doesn't drop the hijacked traffic, but routes it towards its initial destination and therefore intercepts it.
In the first step of an interception attack the adversary attracts traffic, that is directed towards the victim, like described in 3.2.2. In the next step it routes the intercepted traffic towards the victim. So it is likely to happen that the adversary again attracts the traffic, that it just sent towards the victim. In this case the traffic is trapped in a loop. It never reaches the victim and therefore the interception attack fails. So, the key challenge for the adversary is, to always maintain a path to the victim, on which no AS prefers a bogus route over a valid one. See Figure 2 a) for an example:
The adversary uses AS B to route intercepted traffic towards

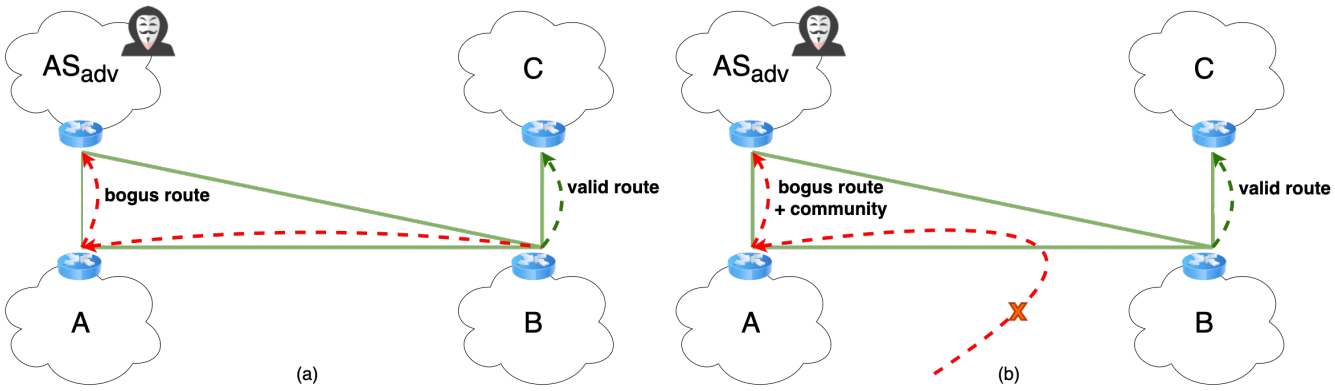

Figure 2. a) Bogus announcement without BGP Communities b) Bogus announcement with BGP community which is hindering AS *A* from exporting bogus path to AS *B*

the victim. If *B* prefers the bogus path (red) over the valid one (green), because of local preference or path length the intercepted traffic is routed back to the adversary.
To avoid this behavior shown in Figure 2 a) the adversary has to make sure that its bogus announcement does not reach AS *B*. To do so it could use AS-Path Poisoning in its bogus announcement and poison AS *B* (like described in 3.2.1). Thus, *B* would not use the bogus path anymore, because of BGPs loop-prevention mechanism. The big limitation of using Path Poisoning for enabling Interception Attacks is the spread loss of the bogus announcement. By using Path Poisoning, the bogus announcement reaches far less ASes and therefore the adversary only attracts a fraction of the traffic it would attract without using Path Poisoning. Studying the spread of bogus announcements shows, that by applying Path Poisoning on only one AS, an adversary loses up to 70% of its attracted traffic. [3]

4 Community Based Interception Attacks

A more sophisticated, but also much more effective method to overcome the challenges of BGP Interception Attacks, like described in 3.2.3 is introduced by Birge-Lee et al., in their paper 'SICO: Surgical Interception Attacks by Manipulating BGP Communities' [3] They present a method of community based Interception Attacks, which they call 'SICO attacks'. See Figure 3 for notation used for SICO attacks. Remember that the key challenge that comes with BGP Interception Attacks is, to always be able to route the intercepted traffic forward towards the victim.
To maintain a valid path from the adversary to the victim and hinder ASes along this path to prefer a bogus path over the valid one, they use BGP Communities. See Figure 2 b) for an example:
The adversary uses AS *B* to route intercepted traffic towards the victim. It adds a community to hinder AS *A* from exporting the bogus path to *B*. *B* only learns the valid path and

AS_{adv}	AS controlled by the adversary
AS_{vic}	AS for the victim's IP prefix
AS_{tar}	AS for the target IPs in Targeted Interception Attacks
A, B	providers of AS_{adv}
$R(X)$	Route from AS X to AS_{vic} (learned by X)
$R^*(X)$	Route from AS X to AS_{adv} (learned by X)
$\mathbb{R}^*(X)$	Set of all routes from AS X to AS_{adv} (learned by X)

Figure 3. Notation used for community based Interception Attacks

therefore has no other option than routing the intercepted traffic via the valid path.
To enable SICO attacks, three BGP Communities most of the top 10 Internet exchanges support are used (see Figure 1). To see how communities are exploited there are three different situations of routing in Figure 4, where the red arrow represents the bogus route R^* and the green arrow stands for the valid route R. Recap that an AS prefers paths received from customers over the ones learned from peers, over the ones learned from providers. In a) AS *A* and AS *B* are the adversary's providers, AS *D* is the victim's provider and AS *C* is a provider for *A*, *B* and *D*. Since *B* gets the announcement from its provider *C* and from its peer *A* it prefers the route it gets from its peer *A* which is R^*. An adversary should now hinder *A* from exporting R^* to *B*, by applying NoExportSelect(*B*) on *A*. Consequently *B* would now only learn *R* from *C* and use it to route traffic towards the victim.
In b) the adversary's providers *A* and *B* have a shared provider *C*. *E* is a provider to *B* and *D* is the victim's provider. Since both systems *C* and *E* are providers to *B*, it uses the criteria of the path length to decide which route to prefer. *R* has four hops and R^* has only three hops. So *B* prefers R^* over *R*. An adversary should now apply LowerPref on *C* which would

cause B to prefer the route it gets from E, R.
In c) A and B again are the adversaries providers and D is the victim's provider. B is peering with both A and D. Since both paths R and R^* are two hops long B would now choose a route based on some interior metrics. Similar to a), an adversary could now apply NoExportSelect(B) on A to prevent A from announcing R^* to B. Now B would only learn R and therefore choose it to route traffic towards the vitim.
Birge-Lee et al. differentiate between two kinds of SICO-attacks: Untargeted Attacks and Targeted Attacks. In Untargeted Attacks an adversary wants to intercept as much of a victim's incoming traffic as possible. In Targeted Attacks an adversary only wants to intercept the victim's incoming traffic, that comes from specific target IPs. One motivation for a Targeted Attack could be, that the adversary is only interested in traffic, coming from the target. Another reason could be that the adversary can not provide the infrastructure for processing the high amount of data, it would attract with an Untargeted Attack. This would result in a decreased performance for the victim and it could discover the attack. In general, Targeted Attacks make Interception Attacks much more efficient and also lower its cost.
To enable community based attacks the adversary needs at least two providers (AS A and AS B) it can route traffic over. The adversary attracts the victim's traffic by announcing a bogus path ($R^*(X)$) over A. After interception the adversary routes traffic towards the victim via B. The key to success is to maintain a valid path $R(B)$ leading from B to the victim.

4.1 Untargeted SICO attacks

Birge-Lee et al. exploit untargeted SICO attacks based on a 4-steps-method:

1. **Make Sample Announcement:** To analyze the spread of your announcements do a sample announcement and let it spread over the Internet which means: Do *not yet* announce a bogus path but your *own* IP-prefix.
2. **Collect Info:** For each AS X along path $R(B)$ check if X prefers any member r of $\mathbb{R}^*(X)$ over $R(B)$. If so add r to a set $s \subseteq \mathbb{R}^*(X)$. See Figure 5 for Pseudo-Code.
3. **AddCommunities:** The members of s from step (2) should now be suppressed with fitting communities: If a route $r \in s$ contains a peer to peer link from AS X to AS Y, apply NoExportSelect at X along with AS Ys ASN. This is preventing X from announcing the bogus path to Y and therefore suppresses r.
   If r does not contain a peer to peer link, apply Lower-Pref at the highest provider in the route.
4. **LaunchAttack:** Announce the victim's prefix via AS A, attracting the victim's traffic. Attach the communities learned from step (3) to the announcement.

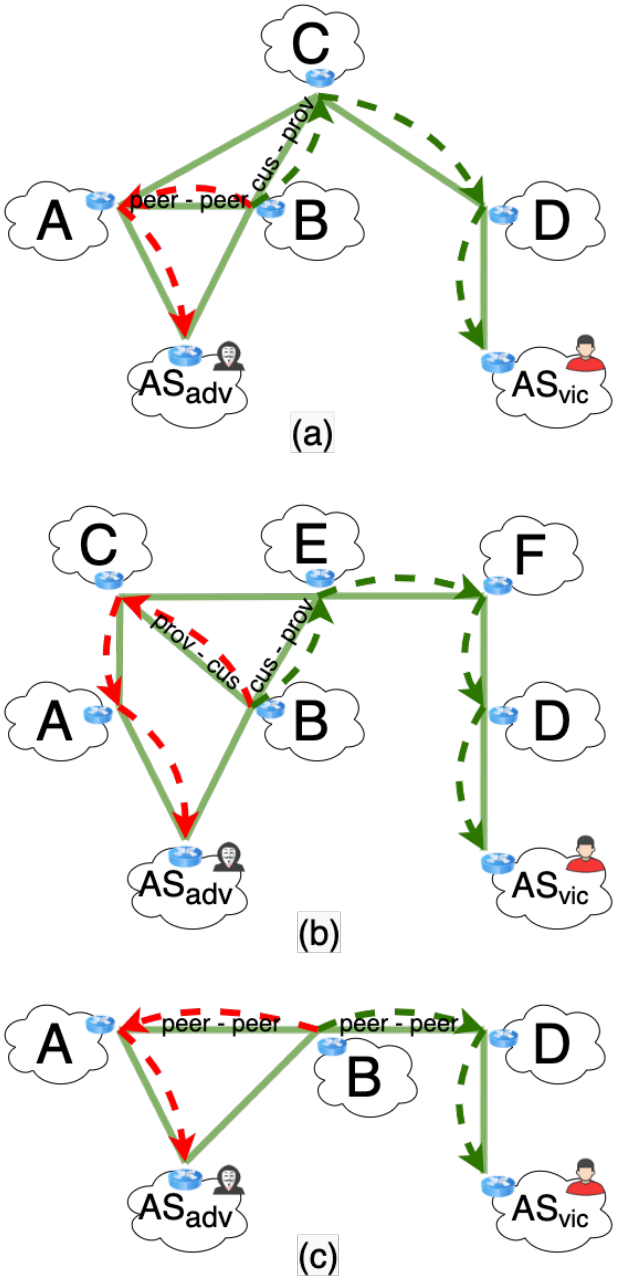

Figure 4. Three situations showing how BGP Communities are exploited. (as presented in [3]) Important links (the ones influencing which path is preferred) are labeled with their relationships (prov $\widehat{=}$ provider, cus $\widehat{=}$ customer)

4.2 Targeted SICO attacks

The goal of targeted SICO attacks is to attract traffic only from targeted IPs. To do so the adversary suppresses any bogus routes that are connecting the adversary with any other AS than the one with the targeted IP prefix. In detail, the adversary adds extra communities in step 3 of the 4-steps-method:

Input: $\mathbb{R}^*(X)$, $R(B)$
Output: Set $s \subseteq \mathbb{R}^*(X)$ of bogus paths that have to be suppressed
$s = \emptyset$;
for *AS X along* $R(B)$ **do**
  for *bogus_path in* $\mathbb{R}^*(X)$ **do**
    if *X prefers bogus_path over* $R(B)$ **then**
      $s = s \cup bogus\_path$;
    end
  end
end
return s

Figure 5. Pseudo-Code representing **CollectInfo** (step 2) of the 4-steps-method

for *AS links X - Y along* $R^*(AS_{tar})$ **do**
  for *neighbor in neighbors of X* **do**
    if *neighbor != Y* **then**
      Prevent X from exporting $R^*(X)$ to *neighbor*;
    end
  end
end

Figure 6. Pseudo-Code for adding communities for Targeted Interception Attacks

Let $p_1, ..., p_i$ be the links between two neighbored ASes X and Y (where X is closer to AS_{adv} than Y) along $R^*(AS_{tar})$. Prevent X from exporting $R^*(X)$ to as many neighbors as possible, while still allowing it to export $R^*(X)$ to Y. See Figure 6 for Pseudo-Code.

5 Application

Knowing about how BGP Interception Attacks work, the question comes up for what to use these attacks. Although there are many use-cases of Interception Attacks, I want to focus on two of them in this paper. One field where Interception Attacks are used is the deanonymization of Tor. A second field where Interception Attacks are used is the manipulation of DNS servers, to misguide Internet users and steal sensitive data.

5.1 Attacking Tors Privacy by using Interception Attacks

Tor is a free and open-source anonymity system used by whistle-blowers, activists, journalists and other people who care about their privacy while using the Internet. To communicate, a user's traffic is routed through the three different layers of the Tor network, before it reaches its destination. Each layer contains several relay-servers which tunnel traffic to a randomly picked relay-server in the next layer. The first layer is the guard, which stays the same for every website a user visits. The guard connects to the middle layer which passes the traffic on to the third, the exit layer. While surfing the Internet the servers in the middle and exit layer change, every time a user visits a new website. Communication between the layers is encrypted to make sure that a relay only knows the identity of the previous and the next hop. By doing so an adversary can not reconstruct the route a user's traffic is taking and it is kept anonymous which users communicate with each other.
In their paper 'Raptor: Routing Attacks on Privacy in Tor' [19] the authors discuss how to deanonymize Tor users based on BGP Interception Attacks.
To do so an adversary has two options: It either controls enough Tor relays or it compromises the underlying structure of Anonymous Systems to gain visibility into user traffic. Around the Internet traffic is often routed asymmetric, which means that the path from a user to a destination does not equal the path back from destination to user. For instance, if a user uploads a file to a server, the actual upload takes another AS route than the TCP acknowledgement the server sends back to the user to confirm that the user's traffic has been received.
It has been shown in the past that an adversary can uncover a Tor user's identity when intercepting a user's traffic from user to guard and from exit relay to destination. [15] This is done by matching packages, intercepted before entering and after leaving the Tor network against each other. This method only makes use of symmetric traffic analysis, which means that an adversary can only run analysis based on traffic that is going in the same direction (from user to entry and from exit to destination).
What stands out in Sun et al.'s research is, that they also make use of asymmetric traffic to find out client-server connections in the Tor network. So when using Raptor-Attacks, an adversary only needs to observe one out of the four options, whereas in previously introduced methods an adversary could only run analysis based on the first two options:

1. Traffic going from user to entry relay and traffic going from exit relay to destination
2. Traffic going from entry relay to user and traffic going from destination to exit relay
3. Traffic going from user to entry relay and traffic going from destination to exit relay
4. Traffic going from entry relay to user and traffic going from exit relay to destination

Having a bigger set of options to choose from also increases the probability that an adversary can observe one out of this four sets of traffic.
Looking at Figure 7 an adversary needs to control at least

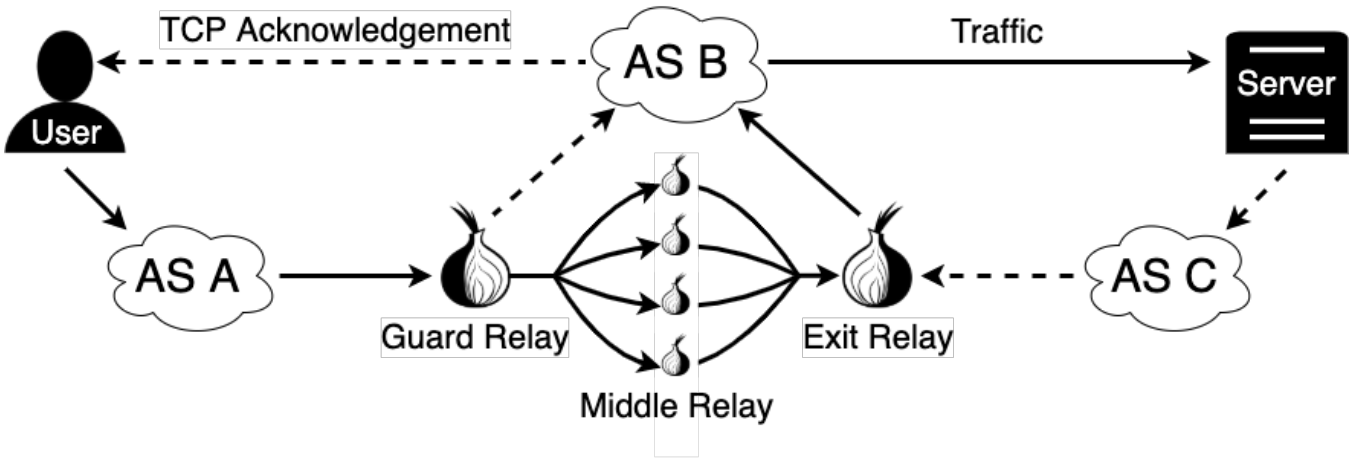

Figure 7. With Raptor Attacks an adversary can uncover a user's identity by controlling either AS *A* and AS *B* or controlling only AS *B*.

AS *A* and AS *B*, or AS *C* and AS *B* when using regular attacks. When using Raptor Attacks an adversary only needs to control AS *B* to uncover a user's identity.
This is where an adversary makes use of Interception Attacks. Using SICO attacks like described earlier in this paper, an adversary would for instance consider AS *B* as its victim and perform an attack on it like described in 4.1. Because AS *B* routes both, the TCP acknowledgement from guard relay to user and also traffic from exit relay to destination server, the adversary could now correlate traffic with TCP acknowledgement. Doing so it would deanonymize owners whose traffic was routed through AS *B*.
To prove the viability of their method Sun et al. performed it on the real Tor network. To do so they set up a guard relay which they made accessible to the Internet via a virtual AS, which had full BGP functionality. To prevent real Tor users from using their guard as a relay and being attacked, they set up a firewall which dropped every user's traffic except of their own. In a next step they set up 50 Tor clients connecting to the Tor network via their guard and 50 web servers connected to the Tor network. By successfully performing an interception attack on their guard relay while the clients simultaneously downloaded a file from the 50 servers, they were able to capture the clients TCP acknowledgement traffic, that was routed towards the Tor network over this guard. Correlating this captured traffic with traffic they collected at their web servers during the same time, they were able to deanonymize Tor sources with an accuracy of 90%. As I concentrate on BGP based attacks in this paper I do not discuss how traffic can be correlated. To get insight into traffic correlation, have a closer look into RAPTOR [19].

5.2 Manipulation of DNS Servers

Another field where BGP hijack attacks have caused severe damage is the rerouting of DNS servers. In April 2018 attackers hijacked some of Amazon's IP space. [12] The attackers where able to exploit a breach in the AS of an American Internet provider (eNET). Being able to propagate some of Amazon's IP addresses over this AS, allowed them to hijack these IP prefixes, which were belonging to authoriative DNS nameservers. DNS nameservers act like a 'phone book' of the Internet, containing links between a website name / URL and the IP address of a server. For instance if a user enters 'www.google.com' in the address field of its browser, the browser requests the URL from a DNS server and gets the IP '216.58.210.14'. Now the user is redirected to the web-server behind this IP address.
In the case of the attack from 2018 the hackers were able to hijack some IP prefixes of Amazon's DNS servers. They redirected users to DNS servers containing malicious DNS tables. These tables connected the URL of the cryptocurrencies website 'myetherwallet.com' to an IP address in eastern Ukraine. Hosted on this IPs server was a fake duplicate of myetherwallet.com. Users who logged in on this fake website were robbed of the entire content of their wallet.

6 Conclusion

Having in mind the different kinds of attacks shows that they have one thing in common: They are all based on the vulnerable structure of the Border Gateway Protocol. The two attacks presented in the last chapter of this paper are only a fraction of possible use cases of BGP interception or hijacking attacks. Doing a web search and scrolling the results shows the huge impact BGP based attacks have. Addressing these flaws it have been made several attempts on securing BGP in the past [5, 6, 8–10, 17], but none of these concepts

has been widely deployed. Although, it might be interesting to keep an eye on the deployment of RPKI (Resource Public Key Infrastructure) which seems to state a promising attempt on securing BGP. It has even been adopted by major ISPs, recently. [7] One point that I did not focus on in this work, but which should definitely be focused on in future work, is discussing how already existing or even new solutions can counter the attacks I outlined.

In this paper I gave an overview on different ways the Border Gateway Protocol can be exploited. I explained how BGP Communities, which are actually intended for engineering purposes, can be misused by attackers. Finally, I underlined the huge impact those attacks can have by bringing up two terrifying real world examples. Considering the high amount of flaws and the impact they have, I conclude that one can indeed say that BGP paves the way for attackers.

References

[1] [n.d.]. https://www.thousandeyes.com/learning/techtorials/transit-provider

[2] James Ball. 2017. NSA stores metadata of millions of web users for up to a year, secret files show. *The Guardian* (2017). https://www.theguardian.com/world/2013/sep/30/nsa-americans-metadata-year-documents

[3] Henry Birge-Lee, Liang Wang, Jennifer Rexford, and Prateek Mittal. 2019. SICO: Surgical Interception Attacks by Manipulating BGP Communities. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security* (London, United Kingdom) *(CCS '19)*. Association for Computing Machinery, New York, NY, USA, 431–448. https://doi.org/10.1145/3319535.3363197

[4] Dyn Guest Blogs. 2005. Internet-Wide Catastrophe - Last Year. *blogs.oracle.com* (2005). https://blogs.oracle.com/internetintelligence/internet-wide-catastrophe%e2%80%94last-year

[5] Alexandra Boldyreva and Robert Lychev. 2012. Provable Security of S-BGP and Other Path Vector Protocols: Model, Analysis and Extensions. In *Proceedings of the 2012 ACM Conference on Computer and Communications Security* (Raleigh, North Carolina, USA) *(CCS '12)*. Association for Computing Machinery, New York, NY, USA, 541–552. https://doi.org/10.1145/2382196.2382254

[6] Haowen Chan, Debabrata Dash, Adrian Perrig, and Hui Zhang. 2006. Modeling Adoptability of Secure BGP Protocol. *SIGCOMM Comput. Commun. Rev.* 36, 4 (Aug. 2006), 279–290. https://doi.org/10.1145/1151659.1159946

[7] Inc. Cloudfare. [n.d.]. Is BGP safe yet? https://isbgpsafeyet.com/

[8] Phillipa Gill, Michael Schapira, and Sharon Goldberg. 2011. Let the market drive deployment: a strategy for transitioning to BGP security. In *SIGCOMM 2011*.

[9] Yih-Chun Hu, Adrian Perrig, and Marvin Sirbu. 2004. SPV: secure path vector routing for securing BGP. *Computer Communication Review - CCR* 34, 179–192.

[10] Stephen Kent, Charles Lynn, Joanne Mikkelson, and Karen Seo. 2000. Secure Border Gateway Protocol (S-BGP) - Real World Performance and Deployment Issues. (03 2000).

[11] Lixin Gao and J. Rexford. 2001. Stable Internet routing without global coordination. *IEEE/ACM Transactions on Networking* 9, 6 (2001), 681–692.

[12] Doug Madory. September 2018. Recent Routing Incidents: Using BGP to Hijack DNS and more. LACNIC 30, Rosario, Argentina. https://www.lacnic.net/innovaportal/file/3207/1/dougmadory_lacnic_30_rosario.pdf

[13] Carolyn Duffy Marsan. 2009. Six worst Internet routing attacks. *networkworld.com* (2009). https://www.networkworld.com/article/2272520/six-worst-internet-routing-attacks.html

[14] Declan McCullagh. 2008. How Pakistan knocked YouTube offline (and how to make sure it never happens again). *cnet.com* (2008). https://www.cnet.com/news/how-pakistan-knocked-youtube-offline-and-how-to-make-sure-it-never-happens-again/

[15] S.J. Murdoch and George Danezis. 2005. Low-cost traffic analysis of Tor. *Proceedings - IEEE Symposium on Security and Privacy*, 183– 195. https://doi.org/10.1109/SP.2005.12

[16] Ola Nordström and Constantinos Dovrolis. 2004. Beware of BGP Attacks. *SIGCOMM Comput. Commun. Rev.* 34, 2 (April 2004), 1–8. https://doi.org/10.1145/997150.997152

[17] P.C. van Oorschot, Tao Wan, and Evangelos Kranakis. 2007. On Interdomain Routing Security and Pretty Secure BGP (PsBGP). *ACM Trans. Inf. Syst. Secur.* 10, 3 (July 2007), 11–es. https://doi.org/10.1145/1266977.1266980

[18] Florian Streibelt, Franziska Lichtblau, Robert Beverly, Anja Feldmann, Cristel Pelsser, Georgios Smaragdakis, and Randy Bush. 2018. BGP Communities: Even More Worms in the Routing Can. In *Proceedings of the Internet Measurement Conference 2018* (Boston, MA, USA) *(IMC '18)*. Association for Computing Machinery, New York, NY, USA, 279–292. https://doi.org/10.1145/3278532.3278557

[19] Yixin Sun, Anne Edmundson, Laurent Vanbever, Oscar Li, Jennifer Rexford, Mung Chiang, and Prateek Mittal. 2015. RAPTOR: Routing Attacks on Privacy in Tor. In *24th USENIX Security Symposium (USENIX Security 15)*. USENIX Association, Washington, D.C., 271–286. https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/sun

[20] Cisco Systems. 2005. REMOTELY TRIGGERED BLACK HOLE FILTERING - DESTINATION BASED AND SOURCE BASED. https://www.cisco.com/c/dam/en_us/about/security/intelligence/blackhole.pdf

[21] PEERING Testbed. [n.d.]. ABOUT PEERING. https://peering.ee.columbia.edu/

[22] C. Wang, Z. Li, X. Huang, and P. Zhang. 2016. Inferring the average as path length of the Internet. In *2016 IEEE International Conference on Network Infrastructure and Digital Content (IC-NIDC)*. 391–395.