Route Leak Identification: A Step Toward Making Inter-Domain Routing More Reliable

M. S. Siddiqui[†‡], D. Montero[†], M. Yannuzzi[†] , R. Serral-Gracià[†], X. Masip-Bruin[‡]
[†]Networking and Information Technology Lab (NetIT Lab)
[‡]Advanced Network Architectures Lab (CRAAX)
Technical University of Catalonia
Vilanova i la Geltrú, Spain
Email: {siddiqui, dmontero, yannuzzi, rserral, xmasip}@ac.upc.edu

***Abstract*—Route leaks are one of the anomalies of inter-domain routing that have the capacity to produce large Internet service disruptions. Route leaks are caused because of violation of routing policies among Autonomous Systems. Unfortunately, there are not many studies that formally and thoroughly analyze the route leak problem. There exist few conventional solutions that can be used as a first line of defense, such as route filters. However, these palliatives become unfeasible in terms of scalability, mainly due to the administrative overhead and cost of maintaining the filters updated. As a result, a significant part of the Internet is defenseless against route leak attacks. In this paper, we define, describe, and examine the different types of route leaks that threaten the security and reliability of the routing system. Our main contributions can be summarized as follows. We develop a rather basic theoretical framework, which, under realistic assumptions, enables a domain to autonomously determine if a particular route advertisement received corresponds to a route leak. We reason the possible occurrence of route leaks in different scenarios, with the aim of formulating requirements for their identification, and hence thereof prevention to improve routing reliability.**

I. Introduction

The Border Gateway Protocol (BGP) protocol [1] is at the core of inter-domain routing among Autonomous Systems (ASs) in the Internet. Unfortunately, the implicit trust model among the ASs for exchanging reachability information using BGP makes inter-domain routing vulnerable to a number of security threats. Attacks such as false prefix origination and false route advertisement, have the potential to disrupt Internet globally. Another inter-domain routing anomaly with the potential to produce large scale service disruptions is the "route leak" problem. Route leaks occur due to violations of policies while exporting routes to a neighbor AS. The ASs typically set their policies for exporting or importing routes from a neighbor AS according to the business relationship that they have with that specific neighbor. There are three types of business relationships between any two ASs: 1) customer–provider; 2) peer–peer; and 3) sibling–sibling relation. In the customer–provider relation, the provider AS offers transit to the customer AS. The ASs in the peer-peer relation usually exchange only their customers' traffic between each other, whereas the ASs in the sibling–sibling relation typically offer transit for each other.

A route leak occurs when an AS advertises a route toward a neighbor AS that does not respect the agreed business relationship between them. For instance, if a customer AS starts offering transit between two of its providers, then it is a route leak. Similarly, a route leak will occur if an AS advertises routes learned from one provider toward a peer AS. We will delve into these aspects later on, but in general terms, a route leak entails a violation of the business relationship that rules the interconnection of domains.

The main concern about route leaks is that they are a common occurrence, and regardless if they are due to misconfigurations or deliberate attacks, they can lead to traffic loss, sub-optimal routing, and more importantly, traffic hijacking. For instance, in 2012, a multi-homed ISP leaked routes learned from one of its providers to another provider, causing a national level disruption in Internet service in Australia [2]. Another major route leak incident occurred the same year, when one of Google's peers improperly advertised Google routes to its provider, knocking out Google services for around half an hour [3]—we shall describe these two incidents in more detail later in Section III.

Route leaks are apparently simple but hard to solve. This is because the ASs keep the information regarding their relationships and policies with other ASs confidential, which makes the identification of policy violations a challenging problem. Although there are orthodox countermeasures for the route leak problem, including route filters, Internet Route Registries (IRRs), and several BGP monitoring tools, they become impotent in face of scalability, due to the high cost of maintenance and dependence on third party information.

In this paper, we formally define and analyze the route leak problem. We describe different types of route leaks and explain how, where, and why they occur with the help of example scenarios. More importantly, we show that, under realistic assumptions and routing conditions, a single AS can detect route leaks utilizing only the standard routing information available at hand, and without needing any vantage point deployed in the internetwork. Our approach targets inference and route leak detection requiring neither changes nor extensions to the BGP protocol. As far as our knowledge, our work introduces the first theoretical analysis for autonomously detecting route leaks.

The rest of the paper is organized as follows. The related work is briefly reviewed in Section II. Section III describes

978-1-4799-4009-7/14/$31.00 ©2014 IEEE

two real world examples of route leaks. The definition and description of different types of route leaks, including essential background information to understand the route leak problem are explained in Section IV. Section V introduces the hypotheses and formalizes the identification of route leaks. Finally, Section VI concludes the paper.

II. Related Work

The primary difficulty in solving the route leak problem lies in the secrecy of the AS relationships in the Internet. There are several AS relationship inference schemes proposed in the literature, including contributions such as [4], [5], [6]. The existing solutions typically infer the relationships between any two ASs by analyzing the BGP data collected at different points in the network, called vantage points. One fundamental critique on such inference schemes is that their knowledge base for inferring the AS relationships is partial, i.e., their view of the Internet is restricted to the data collection points. Ager et al. [7] highlight the limited nature of such AS relationship inference schemes, by detecting far higher number of peer-to-peer links within only one large Internet Exchange Point (IXP), as compared to the number of peer-to-peer links in the entire Internet discovered by well-known inference schemes.

Another methodology to resolve route leaks was proposed in [8]. This method suggests to color each AS-hop in the AS-Path according to the corresponding link type, e.g., an AS-hop is "Green" if toward a provider, and is "Yellow" if toward a peer or customer. This scheme requires new features on the control plane, such as changes or extensions to the routing protocols (e.g., to secure the coloring information in order to avoid manipulations at every AS-hop). The details of this method are out of the scope of this paper, so interested readers are referred to [8]. It is worth mentioning that the security solutions proposed by the IETF's Secure Inter-Domain Routing (SIDR) Working Group (WG) [9], namely, the Resource Public Key Infrastructure (RPKI) [10], Route Origin Authorization (ROA) [11], and Secure BGP (BGPSEC) [12] do not address the route leak problem, since route leaks were out of the scope of the SIDR WG. Indeed, the latter has recently requested the Global Routing Operations WG [13] to define the route leak problem before even attempting to address it.

Overall, the conventional methods to mitigate route leaks include route filters, Internet Route Registries (IRRs), and BGP monitoring tools. The utilization of route filters on the BGP routers between two ASs aims at filtering out routes that are in violation—or are out of the scope—of the agreed policies. The timely and accurate maintenance of route filters becomes challenging as the number of allowed prefixes increase up to thousands, due to the administrative burden. As a result, the ASs prefer to rely on trust and do not maintain up-to-date prefix filters—hence saving their high maintenance cost. The YouTube incident in 2008 [14], and the Google incident in 2012 [3], could have been avoided if the route filters at the provider PCCW were effective. The IRRs provide an online structured database of route objects that can be used to automate the maintenance of the route filters. However, IRRs also suffer from high maintenance cost because the route objects in the IRRs have to be defined first and then kept up-to-date, so the route filters can be automatically maintained. Besides, IRR records are not maintained by all ASs, and existence of duplicate, false, and incomplete records have raised questions on the sanity of the information contained in IRRs. Finally, the BGP monitoring tools analyze BGP data collected at different vantage points to detect irregularities. These monitoring tools have to be trained on up-to-date policies to detect any irregularity, thus causing similar administrative burden as route filters and IRRs. Such monitoring tools are good as long as the irregularities are observed at the vantage points, so strategic attacks avoiding the vantage points can still succeed without detection. Both, BGP monitoring tools and AS relationship inference schemes depend on BGP data collected at different vantage points. However, the former utilize the data to detect irregularities against pre-defined policies, whereas the latter use the data to infer the business relationships and type of peering among ASs.

III. Route Leaks in Real World

Internet service outages by virtue of the BGP shortcomings are several in number, but only a few succeed to get mass attention—it depends on the scale of the service disruption. Two major Internet disruption incidents, that we refer to as Telstra-Dodo [2] and Google-Moratel [3], have been reported, and their apparent causes point out to route leaks. These problems were thoroughly analyzed, and the collected evidence boils down to the violation of routing policies between ASs. What could not be clarified is if they were due to intentional (e.g., a traffic hijack attack) or unintentional misconfiguration (e.g., a fat-finger problem) over the export policies of an AS. Despite the traces and evidence left, we found that some companies involved in these cases claimed that the issues were due to hardware failures, thereby avoiding to mention the possible case of route leaks [15]. Let us describe these two incidents, which we consider clear examples of what route leaks are and their repercussions.

A country-level Internet service disruption occurred in Australia on February 23, 2012 [2], which was attributed to a misconfiguration problem over an AS's export policies. Apparently, one of Dodo's network (AS38285) edge routers was misconfigured to export and offer all its internal routes to one of its providers, namely Telstra (AS1221). The internal routes that Dodo advertised or leaked to Telstra included all routes learned from its providers. These provider-learned routes enclosed all the exported routes of Optus (AS7474), PIPE Internet Exchange (AS23745, AS18398) and the Equinex Exchange (AS24115). Besides, Optus had a peer link with Telstra and, as the latter learned the route to Optus through Dodo, it preferred the customer path as "the best path" (i.e., all traffic coming from Telstra toward Optus was routed via Dodo). This route leak incident turned into a snowball effect when Telstra advertised the new set of Dodo-learned routes to its provider, Telstra International (AS4637), which further advertised them to its peers and customers. Eventually, the

disruption on the Internet service became visible once Telstra started forwarding large amounts of traffic toward Dodo, which was not equipped to handle the traffic volume. Therefore, the peers and customers of Telstra International also started to experience the Internet service disruption. This entire event, illustrated in Fig. 6, occurred in less than an hour, causing large scale connectivity problems across Australia.

Another major outage that directly affected Google's services over some portions of the Internet took place on November 5, 2012 and lasted for about 27 minutes [3]. In this case, Google Networks (AS15169) experienced routing issues with its peer Moratel (AS23947). In short, Moratel exported the routes learned from its peer, Google, toward its providers, and Moratel's providers accepted the leaked routes and selected them as the preferred ones. As a result, a large fraction of Internet traffic destined to Google was transited through Moratel, which was not prepared to handle the traffic load. Whilst this problem was figured out and solved, Google's outage was seen from different segments of the Internet.

These incidents clearly expose the inefficacy of the techniques and tools outlined in Section II. In summary, route leaks represent a high risk and challenging problem that requires new approaches and research efforts. This is precisely the motivation for this paper.

IV. Route Leaks

Let us first define the terminology and the set of policies that rule the routing among ASs.

A. Preliminaries

A "provider link" of an AS is a link that connects it to its provider AS. Similarly, the terms "customer link", "peer link" or "sibling link" refer to a link that connects an AS with a customer AS, a peer AS or a sibling AS, respectively. In this paper, we focus on the two dominant AS relationships in the Internet, which are the customer–provider and peer–peer relationships, since the percentage of sibling relations in the Internet is comparatively negligible.

Whilst the relationship between two ASs is business oriented, pragmatically it is implemented through the BGP protocol. BGP provides complete flexibility for implementing route export or import policies according to the defined relationship, by means of several attributes associated with each advertised route. For example, a provider AS will export all its routes toward its customer ASs in order to attract traffic through its customer links. We are more interested in the export policies, as route leaks occur due to violation of business policies through these exports. The guidelines used for exporting routes (i.e., how to advertise routes depending on the type of relationship with the neighbor AS) are referred to as valley-free rules [4], and they can be summarized as follows:

> **Rule** $\mathcal{R}$.1. *"Routes learned from Customers can be further advertised to other Customers, Peers and Providers."*

> **Rule** $\mathcal{R}$.2. *"Routes learned from Peers can be further advertised to Customers only."*

> **Rule** $\mathcal{R}$.3. *"Routes learned from Providers can be further advertised to Customers only."*

Therefore, in a customer–provider relationship, the customer AS only advertises its own routes and the routes of its customers cone (*i.e., Customer's Customer routes*) toward its provider AS. A customer cone of an AS is the collection of all ASs that are reachable from an AS following only the provider–customer links. On the other hand, the provider AS advertises all routes toward its customer, hence providing it transit to rest of the Internet. In a peer-peer relation, both ASs only advertise their own or their customer's routes to each other. From the business perspective, the provider AS charges its customer AS for forwarding its traffic to and from it. Whereas in the peer–peer relation, the ASs do not charge each other for exchanging each other's customer traffic up to an agreed threshold.

Consequently, ASs prefer a route received from a customer over a route received from a peer or provider to maximize their revenues. Similarly, ASs prefer a route received from a peer over a route received from a provider for any prefix.

B. Defining Route Leaks

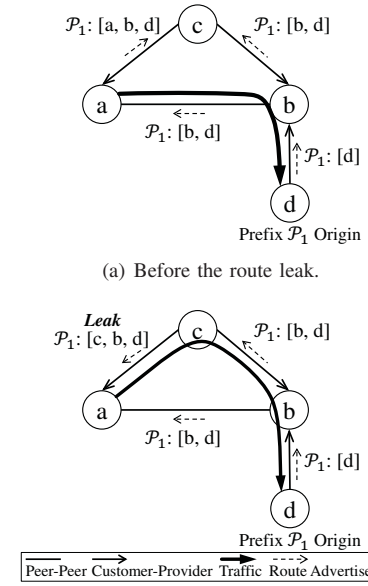
At present, there is no standard definition of the route leak problem in the Internet community. The working group in charge of securing inter-domain routing, namely, the SIDR WG [9], has delegated the task of defining the route leak problem to the GROW WG [13]. The reason for this is that SIDR not only considers route leaks out of their scope but also because their proposals, including RPKI [10], ROA [11] and BGPSEC [12], fail to counter route leaks. There exist some attempts in the literature from where we can extract the initial understanding of the route leak problem. In [8], the author defines route leaks as *the advertisement of a non-customer route over a peer or a provider link*.

It is worth mentioning that a route leak requires neither a false route origin claim nor a false AS-path advertisement to succeed. For example, when Dodo network leaked Optus routes toward Telstra, it neither needed to claim ownership of Optus routes nor to advertise an inexistent path toward Optus. The only violation was that Dodo advertised Optus routes toward Telstra, against the business policy set on the link between Dodo and Telstra. Therefore, a route leak can only occur when exporting routes to a neighbor AS, and the root cause is the violation of the business policy according to the link classification between the two ASs. The valley-free rules summarize the best practice guidelines for exporting routes. In this regard, the valley-free rules can be used as basis for providing an initial definition of the route leak problem.

> **Definition 1.** *"If a route is advertised by an AS toward a neighbor AS, such that it is in violation of the valley-free rules $\mathcal{R}$.2 or $\mathcal{R}$.3, then the route advertisement is a route leak."*

That is, any route advertisement by an AS which infringes the valley-free rules $\mathcal{R}.2$ or $\mathcal{R}.3$ is a route leak. Note that rule $\mathcal{R}.1$ cannot be infringed, since an AS can always export customer routes independently of the business relationship with the neighbor to which it is exporting the route to. Also note that the valley-free rules are not necessarily upheld while exchanging routes under complex AS relationships, e.g., under hybrid relationships—these will be discussed later in Section V. However, such complex relationships are quite uncommon in practice, so the above definition provides a realistic and quite general basis for our initial modeling of route leaks. Using the above definition, we identify two possible types of route leaks from the perspective of an AS which wants to detect route leaks corresponding to the type of the link they occur on, namely, *Customer Route Leaks* and *Peer Route Leaks*. We proceed to describe them through examples.

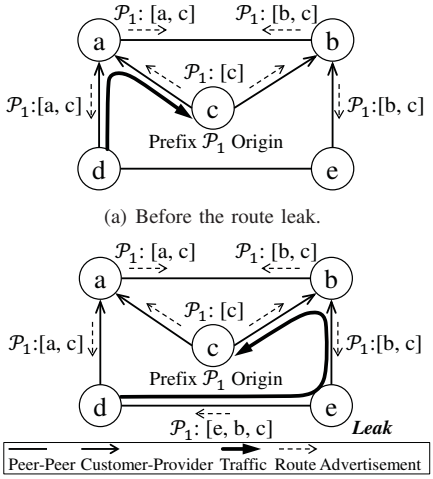
Customer Route Leak: Consider the scenario shown in Fig. 1 (a). The AS b has a peer relation with AS a, and a provider relation with ASs c and d, i.e., c and d are customers of b. The AS c is multihomed with ASs a and b, i.e., c has two providers, a and b. Let us consider now the propagation of a route for prefix $\mathcal{P}_1$ owned by d, i.e., d advertises $\mathcal{P}_1 : [d]$ to its provider b. Following $\mathcal{R}.1$, b forwards $\mathcal{P}_1 : [b, d]$ toward its other customer c and its peer a. In line with $\mathcal{R}.2$, a advertises $\mathcal{P}_1 : [a, b, d]$ to its customer c. The traffic for a source in a and a destination in d would follow the path $[a, b, d]$, as shown in Fig. 1 (a). In the case that c advertises a route learned from one provider to another provider, i.e., advertises the route for prefix $\mathcal{P}_1$ to its provider a, then a would receive two routes for prefix $\mathcal{P}_1$, i.e., $\mathcal{P}_1 : [b, d]$ via b and $\mathcal{P}_1 : [c, b, d]$ via c, as shown in Fig. 1 (b). As mentioned earlier, ASs usually prefer routes learned from customers over routes learned from peers. Consequently, the traffic between a and d will now follow the path $[c, b, d]$. It is worth mentioning that, although the AS-path length via b is shorter than the AS-path length via c, AS a would select the customer route, since the latter is prioritized by setting a higher value of the *local-pref* attribute, which is evaluated before the *AS-Path Length* attribute during the BGP route selection algorithm [1]. According to Definition 1, the advertisement of prefix $\mathcal{P}_1$ by c toward its provider a is a route leak, since it violates the valley-free rule $\mathcal{R}.3$.



(a) Before the route leak.

(b) After the route leak (AS c leaks a route toward its provider AS a).

Fig. 1. Customer route leak scenario.



(a) Before the route leak.

(b) After the route leak (AS e leaks a route toward its peer AS d).

Fig. 2. Peer route leak scenario.

Peer Route Leak: Let us consider now the scenario shown in Fig. 2(a). The AS c is multi-homed with provider ASs a and b. AS d has a peer relation with AS e, and AS d and AS e have a customer-provider relationship with ASs a and b, respectively. AS a and AS b also have a peer link between them. Let us consider the propagation of a route for prefix $\mathcal{P}_1$ owned by AS c, i.e., c advertises the route $\mathcal{P}_1 : [c]$ to its providers. Following $\mathcal{R}.1$, a forwards $\mathcal{P}_1 : [a, c]$ to its customer d and b forwards $\mathcal{P}_1 : [b, c]$ to its customer e. By $\mathcal{R}.3$, d does not advertise the route to its peer e, and reciprocally. The traffic aimed for $\mathcal{P}_1$ originated in AS d would follow the path $[a, c]$ as shown in Fig. 2(a). Now, if as shown in Fig. 2(b), AS e advertises the route for prefix $\mathcal{P}_1$ to its peer d, the latter would receive two different routes for prefix $\mathcal{P}_1$, i.e., $\mathcal{P}_1 : [a, c]$ via a, and $\mathcal{P}_1 : [e, b, c]$ via e. Since d will prefer routes learned from peers over routes learned from providers, the traffic between d and c will now follow the path $[e, b, c]$. Note that similarly to the case shown in Fig. 1, the path length via a is shorter than the path length via e, but still d will select the peer route, since d will prioritize it by setting higher the *local-pref* value. In this example, the route $\mathcal{P}_1 : [e, b, c]$ exported by AS e toward AS d results in a route leak, given that it violates the valley-free rule $\mathcal{R}.3$. Observe that, the route leak examples shown in Figs. 1 and 2 infringe rule $\mathcal{R}.3$, but other examples can be easily elaborated infringing rule $\mathcal{R}.2$.

V. Route Leak Identification

The identification of route leaks is the first step toward solving the route leak problem. Thus, we systematically analyze the various environments where route leaks are possible, and then propose a mechanism for their identification using the definition of valley-free rules stated in the previous section.

A. Identification

In our study, we assume that the route leak identification analysis only uses readily available data, e.g., information obtained directly from the routing tables—that is, from the Route Information Base (RIB) of the routers. We particularly exclude from our analysis data obtained from external sources, such as route information imported from vantage points. In this sense, our identification analysis focuses on what can actually be inferred in a domain under realistic routing conditions, by solely examining the routes received from its neighbors.

We start by defining two facts that we shall use later on while formalizing the identification of route leaks.

> **Fact** $\mathcal{F}.1$.*"A route leak can only be produced by an AS on its peer or provider links".*

Given the definitions detailed in the previous section, we know that an AS acting as provider cannot leak a route toward its customers, since it inherently has the role of providing transit to its customers, so it can advertise "all" its routes toward them. Directly derived from $\mathcal{F}.1$ and Definition 1, we obtain the cases where a route leak is possible.

> **Fact** $\mathcal{F}.2$.*"A route leak can only occur when an AS receives routes from a peer or a customer AS, which were imported by them from their respective peers or providers".*

To illustrate this fact let us consider Fig. 3(a). Let us assume a reference AS a in charge of identifying route leaks. Then for domain a, route leaks can only occur as a result of routes exported by its customer AS c or peer AS p. In the case that c exports routes owned by itself, then such route advertisements can never produce a route leak, since c, being customer of a, can export its own routes to its provider. Similarly, p is allowed to export its own routes to its peer a. Hence, it should be clear that the advertisement of routes owned by a customer or a peer ASs can never cause a route leak on AS a. In other words, a route leak could only occur when a customer or a peer AS exports routes that they imported from their respective neighbors. Observe that, according to $\mathcal{R}.1$, an AS can export the routes it imported from its customers toward its providers or peers, i.e., both c and p are allowed to export the routes that they imported from their customer cones toward AS a.

Then, by using the facts $\mathcal{F}.1$ and $\mathcal{F}.2$ together, it is obvious that the possible network topologies for the occurrence of a route leak for AS a are the ones shown in Fig. 3(b). For the customer route leak case, c could leak either its peer or its provider routes to a. Similarly, for peer route leak scenario, p could leak either its peer or provider routes to a. In any route leak scenario, there are at least three ASs involved; the *victim* AS which receives the leaked routes, the *route leaker* AS which leaks the route, and the *owner* AS which owns the routes that are leaked. For example, in Fig. 3(b) (i), a is the *victim*, c is the *route leaker*, and d is the *owner* of the routes that can be leaked.

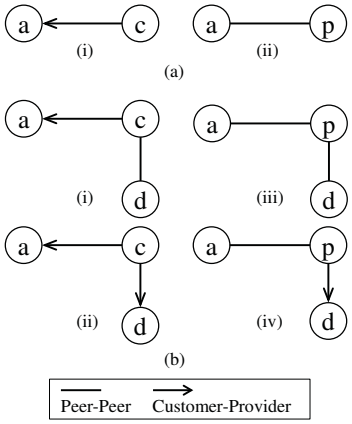
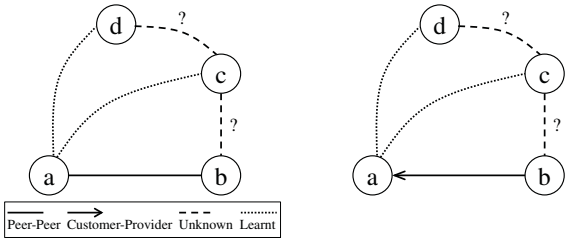


Fig. 3. (a) Possible cases for the occurrence of a route leak on AS a; (b) Possible neighbor links of AS a's customers and peers that can produce a route leak on AS a.

It is worth mentioning that a, the victim, is only aware of AS relationships with its direct neighbors, but has no information about the relationships that its neighbors have with their respective neighbors. AS a can learn the identity of the neighbors' neighbors from the AS path information included in the route advertisements, but remains unaware of their relation. This is because an AS has limited knowledge of the network, since the relationships and policies among ASs are kept confidential. The challenge for AS a is thus to independently detect route leaks despite the lack of information of its neighbors' neighbors relationships.

Let us then consider a network topology scenario for generalizing the local identification of a route leak. Figure 4 depicts the case where our reference AS a is the victim receiving new route advertisements from its neighbors. The goal is to examine under which conditions AS a can locally validate these advertisements prior to inserting them in the RIB and FIB tables of its routers. Domain b represents a neighbor that is directly connected to AS a by a peer-peer or customer-provider link, and it is the one that the victim a suspects that is responsible for leaking the routes (the leaker). Furthermore, c is a direct neighbor of b, which advertises valid routes to AS b of the form $[c, \dots]$ (where "$\dots$" refers to zero or more ASs in the AS-path). These routes can be potentially announced by b to a, e.g, through routes of the form $[b, c, \dots]$. These announcements can be identified as leaks by the victim if they are against the valley-free rules. However, from a's perspective, the announcements cannot be validated due to the lack of information about the type of relationship between the suspect b and its direct neighbor c.

We already stated that the minimum scenario required for a route leak occurrence contemplates three actors: the victim,



(a) Generalized topology for the peer route leak. (b) Generalized topology for the customer route leak.

Fig. 4. Generalized topologies for route leak detection.

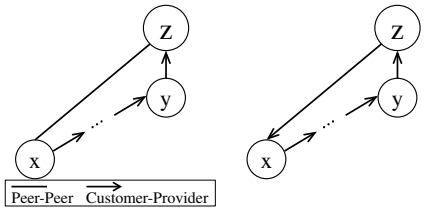
the leaker, and the owner of a route. However, for the sake of generality, we consider the case when the suspect b leaks a route imported from c, but that was originated by another AS, e.g., d. Thus, the potential route that AS b would leak to the victim a would be one learned from c toward d, of the form $[c, \ldots, d]$. Considering that the Internet is a connected graph, it is sound to assume that before the leak occurrence, the victim has a valid route to d, of the form $[\ldots, d]$. When the suspect AS b leaks the route to AS d to attract its traffic (i.e., AS b advertises to AS a a route of the form $[b, c, \ldots, d]$), the victim will be in a position to observe a new route advertisement for the same destination AS. This reference topology and the general assumptions that we will make next shall be used in the remainder of this Section, while formalizing the identification of route leaks in Theorems 1 and 2.

> **Hypothesis** $\mathcal{H}.1$. *"The state of the routing databases of the victim AS is valley-free valid before the route leak occurs."*

Remark: The purpose of our theoretical study is to capture what the victim AS can infer upon a route leak. Therefore, our analysis is focused on the transition from a valley-free valid routing state to the routing state right after the leak. Later, in Section V-C, we will discuss the engineering aspects about how the victim can actually start the identification analysis from a valley-free valid state. In summary, $\mathcal{H}.1$ indicates that any route contained in the initial state of the RIBs at AS a is compliant with $\mathcal{R}.1$, $\mathcal{R}.2$ and $\mathcal{R}.3$.

> **Hypothesis** $\mathcal{H}.2$. *"An AS does not have a peer relationship with the providers of its provider."*

Remark: This hypothesis is based on the assumption that a provider AS is much larger than the customer AS in terms of infrastructure. As shown in Fig. 5(a), it is very unlikely that AS



(a) Hypothesis 2. (b) Hypothesis 3.

Fig. 5. Unlikely AS relationships among ASs.

x has a peer relationship with a provider of its providers, since a very large provider z will have no economical incentives for peering with a domain x at lower tiers of the AS hierarchy. On the contrary, the incentive will be to charge AS x for the transit traffic (cf. Fig. 5(a)).

> **Hypothesis** $\mathcal{H}.3$. *"A cyclic chain of provider relationships among ASs is non-existent."*

Remark: This hypothesis means that we assume an Internet that is loop-free in terms of provider-customer relationships. As shown in Fig. 5(b), it is implausible that AS x is the provider of the provider of its providers. It is a common assumption in the literature that Internet topologies can be modeled as Directed Acyclic Graphs (DAGs) [16].

Now, given the valley-free rules (i.e., $\mathcal{R}.1$–$\mathcal{R}.3$), and the hypotheses defined above, we proceed to formalize the conditions for detecting peer route leaks (cf. Fig. 4(a)).

Theorem 1. *Let the initial state of the routing databases of an AS a contain the following:*

- *A direct route to a peer AS b, i.e., $[b]$.*
- *An alternative route to the peer AS b via AS b's direct neighbor AS c, i.e., a route of the form $[\ldots, c, b]$.*

Under the hypotheses $\mathcal{H}.1$, $\mathcal{H}.2$, and $\mathcal{H}.3$, if AS a receives a route from its peer AS b with AS-path $[b, c, \ldots]$, then AS a can identify it is a route leak.

Proof: According to $\mathcal{R}.1$–$\mathcal{R}.3$, AS b could only advertise a route with AS-path $[b, c, \ldots]$ to AS a, iff, AS c is a customer of AS b. This is because if AS c is a peer or provider of AS b, then AS b is not allowed to advertise routes learned from AS c to its peer AS a. Let us suppose then that AS c is a customer of AS b. We know that the initial state of the routing databases at AS a contain a route to b with AS-path $[\ldots, c, b]$. Now, a could only receive the route to b with AS-path $[\ldots, c, b]$, iff, AS a belongs to the customer cone of AS c. This is because according to $\mathcal{R}.3$, c would advertise its provider routes through b only to its customers. But if a belongs to the customer cone of c, then this contradicts the hypothesis $\mathcal{H}.2$, that is, a has a peer relation with the provider of its provider. Therefore, we conclude that AS c cannot be a customer of AS b. This implies that c is either a peer or a provider of b, and therefore, the route advertised by AS b toward AS a with AS-path $[b, c, \ldots]$ is a route leak. ■

To illustrate the reach and potential application of Theorem 1, let us consider again the peer route leak example given in Fig. 2(b). In practice, the route database of AS d would have a route with AS-path $[a, b, e]$ to e via b, plus the direct route $[e]$ to e in its initial state. The former is because a and b would exchange customer routes with each other. Assuming that the initial state at AS d is valley-free valid, the set up in Fig. 2(b) is under the hypotheses of Theorem 1, so AS d can autonomously conclude that the route $\mathcal{P}_1 : [e, b, c]$ received from AS e is a route leak.

We proceed now to formalize the detection of customer route leaks (cf. Fig. 4(b)).

Theorem 2. *Let the initial state of the routing databases of an AS a contain the following:*

- *A direct route to a customer AS b, i.e., $[b]$.*
- *An alternative route to the customer AS b via AS b's direct neighbor AS c, i.e., a route of the form $[\dots, c, b]$.*

Under the hypotheses $\mathcal{H}.1$, $\mathcal{H}.2$, and $\mathcal{H}.3$, if AS a receives a route from its customer AS b with AS-path $[b, c, \dots]$, then AS a can identify it is a route leak.

Proof: Just as in the proof of Theorem 1, AS b could only advertise a route with AS-path $[b, c, \dots]$ to a, iff, c is a customer of b. This is because if c is a peer or provider of b, then b is not allowed to advertise routes learned from c to its provider AS a. Let us suppose then that c is a customer of b. We know that the initial state of the routing databases at AS a contain a route to b with AS-path $[\dots, c, b]$. Now, a could only receive the route to b with AS-path $[\dots, c, b]$, iff, a belongs to the customer cone of c. This is because according to $\mathcal{R}.3$, c would advertise its provider routes only to its customers. But if a belongs to the customer cone of c, then this contradicts $\mathcal{H}.3$, since there is a cyclic chain of provider relationships among a, b, and c, that is, a is a provider of b, which is a provider of c, which in turn is provider of a. We conclude that AS c cannot be a customer of AS b. This implies that c is either a peer or a provider of b. Hence, the route advertised by AS b toward AS a with AS-path $[b, c, \dots]$ is a route leak. ■

It can be shown that if the initial conditions are met, then Theorem 2 applies to the example illustrated in Fig. 1(b).

B. Example of Real World Route Leak Detection

Let us revisit the Dodo-Telstra incident explained in Section III, under Theorem 2 and the hypotheses defined above. For this case, the AS of reference, i.e., the victim, is Telstra, AS t (cf. Fig. 6); the leaker is Dodo, AS d; and the leaked route owner is Optus, AS o, and all its internal routes exported toward its client Dodo. Assuming that $\mathcal{H}.1$ holds before the leak, the routing databases at Telstra contained: $\{[d], [o], [o, d], [\dots, o], [\dots, x, o], [\dots, o, d]\}$. Observe that, the victim, Telstra (t), has a direct route to its suspect client Dodo (d). Also note that there are alternative routes to the customer d via a direct neighbor of d, namely, Optus (o). Thus, the real scenario depicted in Fig. 6 satisfies the hypothesis of Theorem 2. Hence, when AS t receives a route of the form $[d, o, \dots]$, then it is in the position to detect it as a route leak.

C. Discussion

Even though our proposals can be applied in many practical situations (e.g., the Dodo-Telstra incident could have been avoided), there are still some others that might not satisfy the hypotheses of Theorems 1 and 2, and therefore, they need further analysis. In the remainder of this Section, we discuss the reach and limitations of the contributions in this paper.

Hybrid Relationships: The valley-free rules for exporting routes serve as a reasonable stepping stone toward theoretically modeling the route leak problem. However, the valley-free export rules are not necessarily satisfied under certain complex relationships between ASs, such as hybrid relationships. These latter refer to cases where two large ASs have different relationships between them at geographically different points of presence (PoP). For example, two ASs may have a customer–provider relation in one region and a peer–peer relation in another region. We contend that the analysis presented in this paper may even stay valid in various hybrid scenarios, since the routing information that is relevant for the detection is the one contained in the routers in proximity with the occurrence of the route leak—independently of the divergence on the routing views at geographically separated areas.

Route Leak Propagation: Observe that our analysis can only be used for detecting when a route leak is initiated. Detecting route leak propagation is far more difficult than detecting its initiation. The route leak propagation refers to the scenario where the *victim* AS receives a route leak and forwards it further to its neighbors. The *victim* AS may forward the route leak to its neighbors according to the relationship it has with them, which makes it more difficult for any AS receiving the propagated route to detect it as a route leak. An extended version of the customer route leak example presented in Section IV is shown in Fig. 7. The AS a forwards the leaked route $\mathcal{P}_1[a, c, b, d]$ received from its customer AS c to its peer AS e, which is allowed according to $\mathcal{R}.1$–$\mathcal{R}.3$. The AS e further advertises this leaked route to its customers, including AS f. Note that neither AS e nor AS f can detect this route advertisement as a route leak, since they receive it in accordance with the relationship that they have with their corresponding neighbors. We leave the detection of route leak propagation for future research.

Initial Valley-Free State: From an engineering perspective,

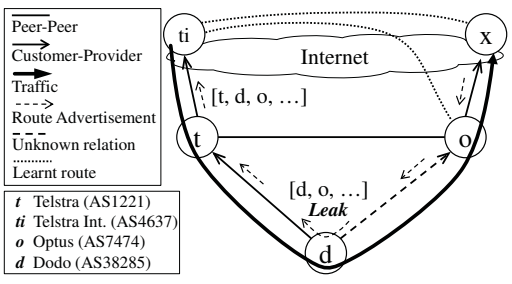


Fig. 6. The Dodo-Telstra incident (a customer route leak case).

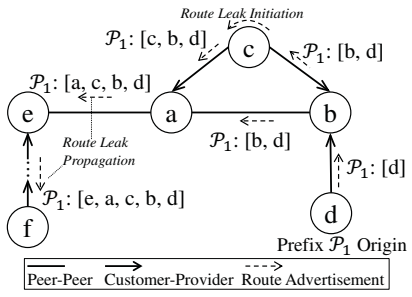


Fig. 7. Route leak initiation and route leak propagation.

the hypothesis $\mathcal{H}.1$ is reasonably achievable by many transit domains, since route filters can be set to that end for a short period. This will ensure that the routes imported up to that stage are valley-free. Once this is guaranteed, the route filters need not be maintained and could be removed. Observe that the reluctance of providers for using filters does not lie on their initial configuration, but rather on keeping them updated. In any case, this method of applying and removing filters is challenging for very large providers, and without SIDR's solutions in place, it can only be achieved through a chain of trust during filter configuration. Further research is needed on how to ensure that the initial state at the potential victims is valley-free.

Cross Paths: This refers to the phenomenon when an AS receives two routes, one with AS-path $[\ldots, x, y, \ldots]$, and another with $[\ldots, y, x, \ldots]$, i.e., it has cross paths between AS x and AS y. The observance of cross paths at the *victim* AS involving the *route leaker* AS and the *route owner* AS is the crux of Theorems 1 and 2 for detecting route leaks. Due to the existence of different and complex relationships among ASs, it may not always be possible for the *victim* AS to observe cross paths between the *route leaker* and the *route owner*. Figure 8 shows a variant of the customer route leak example, where AS a cannot observe cross paths between AS b and AS c upon a route leak at c. In the initial state, AS a will have routes $[c], [b]$, and $[b, d]$ only. Clearly, this setting does not satisfy the hypothesis of Theorem 2. Observe, however, that upon receiving the route $[c, b, d]$ from AS c, AS a must assume that AS b belongs to customer cone of AS c. This is because this is the only valid possibility according to $\mathcal{R}.1$–$\mathcal{R}.3$. Given that AS a has a direct peer link with AS b, AS a concludes that AS b has a peer relation with the provider of its provider, which violates $\mathcal{H}.2$. This implies that AS b cannot belong to the customer cone of AS c. Hence, AS a can conclude that the route $\mathcal{P}_1$ advertised by AS c is a route leak. The reason for which AS a is able to detect the route leak from AS c without the cross paths is because in this particular case, it has a direct link with the *route owner* AS, i.e., AS b. Note that this might not be the case in practice. For instance, consider the case where c is leaking routes from a peer b that AS a is not connected to, and a does not have an alternative route to reach c through b. We plan to address the problem of absence of cross paths in our future research.

VI. Conclusion

In this paper, we studied a set of anomalies that threaten the security and reliability of the inter-domain routing system, which are referred to as route leaks. We introduced a basic theoretical framework including realistic hypotheses and theorems, under which an AS is able to detect route leak initiation autonomously. The main advantages of our approach include: a) no reliance on third party information (e.g., vantage points); b) no changes required to control-plane protocols (e.g., to BGP); and c) from an engineering perspective, route filters may be needed for an initial training period to ascertain the defined hypotheses, but their continuous maintenance is not required. We concede that the theoretical analysis presented in this paper is valid for detecting—under certain conditions—route leak initiations only. The detection of a propagated route leak requires further investigations. Further research is also needed to find ways of detecting route leaks under conditions relaxing the hypotheses of Theorems 1 and 2 (e.g., in the absence of cross paths in the RIBs for the route leaker and the route owner).

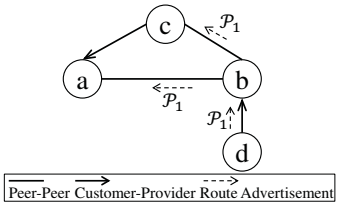


Fig. 8. Topology scenario where AS a cannot observe cross paths between AS b and AS c.

Acknowledgment

This work was supported in part by an RFP granted by Cisco Systems, Inc., by the Spanish Ministry of Science and Innovation under contract TEC2012-34682, and by the Catalan Government under contract 2009 SGR1508.

References

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

[2] G. Huston, "Leaking Routes," http://www.potaroo.net/ispcol/2012-03/leaks.html, March 2012.

[3] T. Paseka, "Why Google Went Offline Today and a Bit about How the Internet Works," http://blog.cloudflare.com/why-google-went-offline-today-and-a-bit-about, Nov. 2012.

[4] L. Gao, "On Inferring Autonomous System Relationships in the Internet," *IEEE/ACM Trans. Netw.*, vol. 9, no. 6, pp. 733–745, Dec. 2001.

[5] L. Subramanian, S. Agarwal, J. Rexford, and R. Katz, "Characterizing the Internet Hierarchy from Multiple Vantage Points," Berkeley, CA, USA, Tech. Rep., 2001.

[6] X. Dimitropoulos, D. Krioukov, M. Fomenkov, B. Huffaker, Y. Hyun, k. claffy, and G. Riley, "AS Relationships: Inference and Validation," *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 1, pp. 29–40, Jan. 2007.

[7] B. Ager, N. Chatzis, A. Feldmann, N. Sarrar, S. Uhlig, and W. Willinger, "Anatomy of a Large European IXP," in *Proceedings of the ACM SIGCOMM 2012 conference on Applications, technologies, architectures, and protocols for computer communication*, ser. SIGCOMM '12. New York, NY, USA: ACM, 2012, pp. 163–174.

[8] B. Dickson, "Route Leaks – Requirements for Detection and Prevention thereof," draft-dickson-sidr-route-leak-reqts, March 2012.

[9] Secure InterDomain Routing (SIDR) Working Group IETF, http://datatracker.ietf.org/wg/sidr/, 2013.

[10] M. Lepinski and S. Kent, "An Infrastructure to Support Secure Internet Routing," IETF, RFC 6480, 2012.

[11] M. Lepinski, S. Kent, and D. Kong, "A Profile for Route Origin Authorizations (ROAs)," IETF, RFC 6482, 2012.

[12] M. Lepinski, "BGPSEC Protocol Specification," draft-ietf-sidr-bgpsec-protocol, February 2013.

[13] Global Routing Operations (GROW) Working Group IETF, http://datatracker.ietf.org/wg/grow/, 2013.

[14] RIPE NCC, "YouTube Hijacking: A RIPE NCC RIS case study," http://www.ripe.net/internet-coordination/news/industry-developments/youtube-hijacking-a-ripe-ncc-ris-case-study, November 2010.

[15] R. Crozier and J. Hutchinson, "Dodo cops blame for national internet outages," http://www.itnews.com.au/News/291364,dodo-cops-blame-for-national-internet-outages.aspx, Feb. 2012.

[16] B. Hummel and S. Kosub, "Acyclic Type-of-relationship Problems on the Internet: An Experimental Analysis," in *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*, ser. IMC '07. New York, NY, USA: ACM, 2007, pp. 221–226.