Detecting Large Route Leaks

Qing Ju Varun Khare Beichuan Zhang
{qingju, vkhare, bzhang}@cs.arizona.edu
The University of Arizona

Abstract

Prefix hijacking, in which an unauthorized network announces IP prefixes of other networks, is a major threat to the Internet routing security. Existing detection systems either generate many false positives, requiring frequent human intervention, or are designed to protect a small number of specific prefixes. Therefore they are not suitable to protect data traffic at networks other than the prefix owner during on-going hijacks. We design and implement a system that detects a specific type of prefix hijacking, large route leaks, at real time and without requiring authoritative prefix ownership information. In a large route leak, an unauthorized network hijacks prefixes owned by multiple different networks. By correlating suspicious routing announcements along the time dimension and comparing with a network's past behavior, we are able to identify a network's abnormal behavior of offending multiple other networks at the same time. Applying the detection algorithm to routing data from 2003 through 2009, we identify five to twenty large route leaks every year. They typically hijack prefixes owned by a few tens of other networks, last from a few minutes to a few hours, and pollute routes at most vantage points of the data collector. In 2009 there are ten events detected, none of which was mentioned on operator mailing lists, but most are confirmed through our communication with individual operators of affected networks. The system can take real-time routing data feed and conduct the detection quickly, enabling automated response to these attacks without requiring authoritative prefix ownership information or human intervention.

1 Introduction

The Internet is an interconnection of tens of thousands independently administered networks called Autonomous Systems (ASes). An AS announces its IP prefixes onto the Internet via the Border Gateway Protocol (BGP). Due to the lack of any authentication mechanism in BGP, an AS can make false routing announcements, including announcing prefixes owned by other networks, *i.e.*, hijacking the prefix. Once a prefix is hijacked, some or all traffic destined to the prefix will be diverted to the perpetrator network. Malicious attackers can use prefix hijacking to hide their network identity in sending spams, inflict denial-of-service attacks by dropping victim's traffic, or even manipulate victim's traffic before forwarding it to the legitimate destination [13].

A number of detection systems have been developed in recent years, including Cyclops [14], PHAS [18], MyASN [8], IAR [3], iSPY [28], Neighborhood Watch [22], origin list [30], Lightweight Probing [31] and LOCK [21]. These systems detect prefix hijacks by examining routing updates, probing data paths, cross-checking with registry databases, or a combination of these techniques. Once a prefix hijack is detected, the owner of the prefix will be notified, and it is expected that the owner will take actions to resolve the problem, which, in today's Internet, usually involves contacting the offending network or its upstream provider to stop the false announcements. This process of detection, notification and resolution takes time, during which the damage to data traffic has already been made and malicious attackers may have already achieved their goals. For instance, in the 2008 incident [12] when one of YouTube's prefixes was hijacked by AS 17557, it took

80 minutes for YouTube to launch the first countermeasure, and 2 hours and 14 minutes before the false announcement was withdrawn. Meanwhile, YouTube service suffered worldwide outage.

There is an urgent need to protect data traffic during on-going prefix hijacks. This calls for an accurate, real-time detection system that does not require authoritative information from the prefix owner. Such a detection system would let networks other than the prefix owner quickly detect prefix hijacking and respond to it, *e.g.*, by dropping the false routing announcements. However, this is an extremely challenging task that none of the existing systems is up to. Those that use BGP routing data and registry databases usually report too many false positives, requiring human intervention or authoritative information to filter the results. For example, one such system [20] generates around 20 alarms daily. Those that use traceroute to probe data paths are designed to protect specific prefixes; they cannot be used to probe all prefixes in a routing table. Thus none of them is suitable to protect traffic at networks other than the prefix owner.

As the first step towards protecting data traffic during on-going prefix hijacks, we design and implement a system that detects a specific type of prefix hijack, *large route leaks (LRL)*, at real-time without any authoritative prefix ownership information. In a large route leak event, an unauthorized network hijacks prefixes of multiple different networks. For instance, in September 2008, AS 8997 announced more than 117K prefixes, affecting data delivery at more than 15K ASes [7]. By restricting to large route leaks, we are able to exploit its unique characteristics in minimizing false positives. The detection algorithm goes through BGP routing updates to identify individual suspicious announcements based on the past history of the prefix-origin announcements observed. It then correlates the suspicious announcements along the time dimension to see how many other networks an AS is offending at the same time. If the number of offended networks is above a threshold, which is 10 in our current implementation, this event is reported as a large route leak. Since we correlate suspicious announcements along time dimension and look for statistically abnormal behavior, the accuracy of detecting *individual* prefix hijacking becomes less important. The goal is to detect a non-trivial set of large route leaks without false positives, so that networks can respond to the attacks quickly, maybe even automate the response to drop false routing announcements at ingress routers. Inevitably false negatives exist. They may be dealt with other methods that take longer time or need more information, but are not the focus of our current system.

We applied the detection algorithm to BGP routing data from 2003 through 2009 collected by Route-Views [10] Oregon collector. We identify 5 to 20 large route leaks each year. They typically hijack prefixes owned by a few tens of other networks, last from a few minutes to a few hours, and pollute routes at most vantage points of the data collector, implying that they inflict significant damage to data traffic. In year 2009, there are totally 10 events detected, 9 of them are confirmed via emails from operators of the affected networks, and the remaining one is likely to be correct too based on the attacker network's past behavior. Thus the 2009 result does not have false positives.[1] Surprisingly, none of the 10 events was mentioned in operator mail list such as NANOG list [10], which means that our detection results are non-trivial and useful. We have also implemented an online version of the algorithm to take real-time BGP data feed and report detection results.

The rest of the paper is organized as follows. We present background and motivation of the LRL detection problem in Section 2. Section 3 presents detection algorithm, both the offline form and the online form. Section 4 reports the detection results. We discuss related work in Section 5 and conclude the paper in Section 6.

2 Background and Motivation

Figure 1 illustrates prefix hijacking via a simple example. AS O is the owner of prefix p and it announces the prefix to the Internet. Without hijacks, all traffic destined to p should go to AS O. When the attacker

[1] We did not attempt to confirm results of earlier years via emails as it may not be convenient for operators.

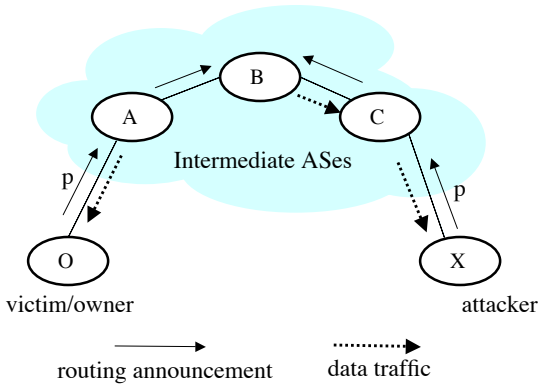

Figure 1: Prefix Hijack Example

AS X also announces the same prefix p, some networks may prefer the new path to this prefix and forward data towards AS X. Thus the damage is made. We call the networks that are neither the prefix owner nor the attacker *intermediate ASes*, and if an intermediate AS sends data towards the attacker, we say this intermediate AS has been *polluted*, *e.g.*, B and C in Figure 1. Prefix hijacking can be caused by inadvertent misconfigurations or intentional attacks, but in this paper we do not differentiate them because the effect of diverting traffic to unauthorized network is the same. Existing detection systems have focused on letting the prefix owner know about the prefix hijack, so that they can take actions to stop it. Our focus is to protect data traffic at intermediate ASes when prefix hijacks are going on.

Fast and accurate detection of prefix hijacks at intermediate ASes is extremely challenging due to two reasons. First, there is no authoritative database about prefix ownership available. The closest that one can get is the various Internet registries, which are maintained mostly on a voluntary basis and known to be incomplete and out-of-date. Thus it can be used as a source of information, but not authoritative. Second, there are many operational practices that look exactly like a prefix hijack but are legitimate. For examples, an anycast prefix may be announced by multiple unrelated ASes at the same time, a provider network may announce the prefix of its customer in case of network problems, an airplane may announce its prefix via different ASes as it flies over different continents, and so on. All of these cases make it difficult to differentiate real prefix hijacks from legitimate network operations without authoritative information.

Existing detection systems do not suite well for intermediate ASes. Traceroute-based solutions (*e.g.*, [28, 31]) periodically probe data paths to a specific prefix, thus they are best to be used by prefix owners to protect their small set of known prefixes, not by intermediate ASes, who have an entire routing table to protect. BGP-based solutions (*e.g.*, [3, 20]) can monitor the entire routing table passively, but they usually end up with a large number of alarms, many of which may be false positives. Some BGP-based solutions (*e.g.*, [14, 18, 8] use information provided by prefix owners to filter out false positives, but then their effectiveness is limited by the number of participating prefix owners. Besides, the effectiveness of all the existing detection systems depend on how well their vantage points cover the Internet. If an intermediate AS is not covered by these vantage points, it will not benefit.

Intermediate ASes are in need of a fast and accurate detection system in order to protect their data traffic. Given the accuracy is very difficult to achieve, a sensible tradeoff would be to tolerate false negatives but minimize false positives. A false negative is the case that a real prefix hijack is not reported, a false positive is the case that a reported hijack is actually legitimate. Minimizing false positives allows networks to respond to attacks quickly, maybe even automate the response at the network operation center. However, the danger of going too far down this direction is to detect only the very large scale events that everyone will notice without any detection system. Thus the goal set off for this work is to develop a system that can

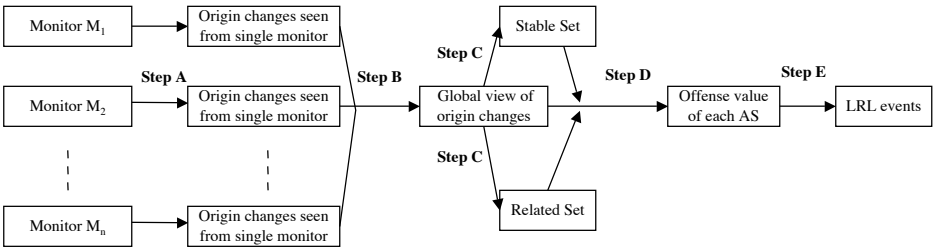

Figure 2: An overview of LRL detection Scheme

detect a non-trivial set of certain prefix hijacks at intermediate ASes with minimal false positives.

We call the type of prefix hijacks that we detect "large route leaks," in which a network hijacks prefixes of multiple other networks at the same time. An extreme case would be that one network leaks its full routing table, effectively hijacking the entire Internet, which happened quite a few times in the history of the Internet. The earliest one was reported in 1997 when AS 7007 accidentally leaked its routing table [1]. Nowadays, leaking the full table is less common, partly because of better awareness of the problem and partly because of the adoption of prefix limit, which caps the number of prefixes allowed from a given BGP peer. However, as our results will show, route leaks that hijack tens or a couple of hundreds prefixes are much more frequent than one would expect, and the operation community is generally unaware of them. Thus our system, even only detects a subset of all prefix hijacks, can improve current Internet routing security significantly.

3 LRL: Large Route Leak Detection

The detection of LRL events exploits the fact that the attacker AS offends multiple ASes at the same time. Though it is possible that a network legitimately announces prefixes of another network, it is unlikely that a network does this to many different networks at the same time. The detection algorithm obtains individual prefix origin conflicts, correlate them in time, and identify LRL events by looking for outliers in the number of networks being offended. The rest of this section describes the algorithm in detail.

3.1 Overview

Since the LRL detection is designed to protect data traffic at intermediate networks, it uses mainly the BGP routing data, which is readily available in a network and covers all prefixes and all routing changes seen by the intermediate network. The detection algorithm also uses WHOIS data and some information about Internet Exchange Points (IXPs) to reduce noises, which will be explained later.

We envision that LRL detection would be running in a network operation center, which receives real-time BGP routing feeds from the network's operational routers, and/or public data collectors such as Route-Views [10] and RIPE [9]. Once an LRL event is detected, it will trigger an alarm sent to the operator, or an automatic response mechanism such as instructing the routers dropping the false routing announcements of the attacker AS.

Figure 2 shows the overall work flow of the detection. Basically the routing update streams from each router are first processed individually to generate each router's single view of the origin changes (Step A), then the single views are merged to get the global view of all origin changes (Step B). At Step C, legitimate prefix announcements are identified into so-called "stable set" and "related set" based on history behavior, most of the remaining origin changes cause conflicts and are counted to get the offense value for each AS at Step D. Finally Step E applies a threshold of the offense value to identify outliers as LRL events.

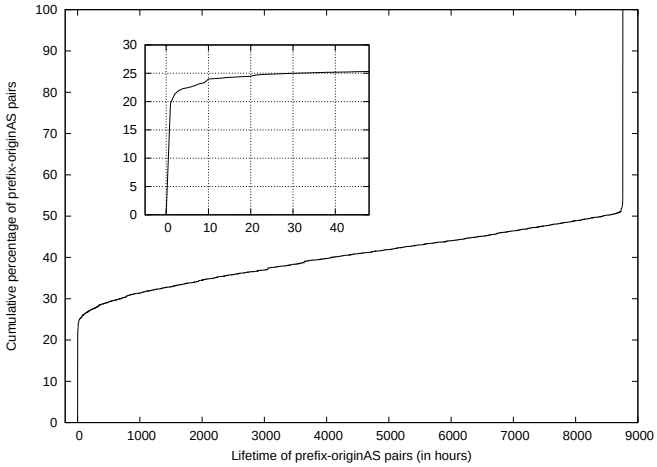

Figure 3: CDF of the lifetime of prefix-originAS pairs in 2009. Short-lived prefix-originAS pairs last less than one day.

3.2 The Detection Algorithm

3.2.1 Step A: Obtain the Single View of Origin Changes

The first step is simply filter out BGP updates that do not make any changes to prefix origins. It keeps track of the origin AS for every prefix, and record the change if there is any announcement of a new prefix-origin or a withdrawal of existing origin. It ignores all other BGP updates. For instance, if monitor M_1 sees that AS X is announcing prefix p at time t_1 and thereafter withdrawing the prefix p at time t_2 where $t_1 \leq t_2$ which simply means prefix p is live for the duration of t_1 to t_2 by origin AS X.

3.2.2 Step B: Obtain the Global View of Origin Changes

The second step merges all the individual view of origin changes into a global view. The result is the set of origin ASes for each prefix at any time. For instance, if at time t, monitor M_1 sees AS X as the origin of prefix p, but M_2 sees AS Y as p's origin, then the global view will have the set of $\{$X, Y$\}$ as p's origin at time t. It is a union of all the individual views of prefix origins.

3.2.3 Step C: Characterize Legitimate Announcements

This step is important to reduce noises in the final detection results. The goal is to identify origin changes that can be regarded as legitimate. The underlying assumption is that if an origin AS can announce a prefix for a substantial period of time, it is likely to be legitimate, since otherwise it would have been stopped by the owner of the prefix. Given a prefix, we define two sets of origin ASes, *stable set* and *related set*, that can legitimately announce the prefix.

stable set The stable set is meant to capture the owners of a prefix. A network's possession of IP addresses and AS numbers is long term in nature. To make a good use of the prefixes, any network would like to maintain uninterrupted connectivity to their prefixes by keeping announcing the prefixes via BGP. Therefore the expectation is that the real owner should show up in BGP routing updates as a persistent origin AS of a prefix.

The lifetime of prefix-originAS pair is analyzed to estimate the announcement duration threshold required for any AS to be safely considered in the stable set of any prefix. The lifetime is defined as the *cumulative* time that an AS announces a prefix over an entire year. Figure 3 shows the CDF of lifetime for all prefix-originAS pairs in year 2009. More than 40% of the prefix-originAS pairs are live for the entire duration of the year. Thereafter nearly 40% of prefix-originAS pairs are live for a duration somewhere between one day and one year. Upon further analysis most of these prefixes are found to be newly allocated prefixes which in previous years were unallocated by RIRs (not announced by any other AS), or prefixes that were ceased to be announced sometime in the middle of the year. Finally about 20% of the prefix-originAS pairs are extremely short-lived, lasting less than a day. False routing announcements are likely to be part of these short-lived prefix-originAS pairs. Therefore, we use the threshold of one day to define stable set. In other words, if an origin AS has announced a prefix cumulatively more than one day during a year, then we regard this AS as a member of the prefix's stable origin set. In 2009, 22.06% prefixes have no stable set, 74.45% prefixes have stable sets of only a single AS and the remaining 3.49% prefixes have stable sets of multiple ASes. We have also tested threshold of longer than one day and obtained similar final detection results.

Related Set The related set is meant to capture the ASes that are not the owner of a prefix but can legitimately announce the prefix to the Internet from time-to-time. It is impossible to enumerate all operational practices that can lead to such legitimate announcements and try to find all of them in BGP data. We identify four main types and use them to classify ASes into a prefix's related set.

First, if AS X is in the stable set of prefix p, X is automatically in the related set of any sub-prefix of p. Therefore if AS X belongs to stable set (p) and AS Y belongs to stable set (p') such that p' is a sub-prefix of p, then AS X is in the Related Set (p'). This captures the cases that an ISP allocates some of its address space to its customers, but sometime may need to announce the sub-space on behalf of the customer.

Second, if AS X has a stable network connectivity with AS Y, then X is in the related set of Y's prefixes. This captures the cases when a neighbor AS, likely a provider, needs to announce a prefix on behalf of its neighbor, likely a customer. AS belongs to Related Set if it is expected to be the provider of an AS already in the stable set of one prefix. Provider and customer relationship can be inferred based on the observation that provider and customer relationship is more likely to remain unchanged over time. This is mainly due to the fact that the contract between provider and customer is usually on a long term basis. Previous work such as [15] has also confirmed this observation. If AS a_0 originate prefix p through AS path$\{a_k, ..., a_1, a_0\}$, it can be inferred that a_1 is the upstream of a_0. In addition, if this AS path does not change during the period $[t-T, t)$, the lifetime of this upstream and downstream AS pair (a_1, a_0) is T. In LRL detection scheme, AS $(a_1$ and AS $a_0)$ are provider and customer of prefix p if the lifetime of upstream-downstream relationship exists for more than a threshold in one year. For example, if AS a_1 is in stable set (p), AS a_0 is in stable set (p'), and the lifetime of upstream-downstream relationship between AS a_1 and a_0 for prefix p' exists for more than a threshold in a year, then AS a_1 is in RelatedSet (p').

Provider-customer relationships are expected to be stable due to the underlying business contracts which form their basis. We attempt to set a threshold to remove short-lived AS pairs which are results of path spoofing attacks. Figure 4 presents the CDF of lifetime of AS pair (X, Y) as seen in the routing announcement data in 2009. As shown in the figure, 28.46% of the AS pairs are extremely short-lived lasting less than a day within an entire year. One day is a conservative threshold for provider-customer relationship since it captures most of the short-lived AS pairs which are candidate for path spoofing attacks. On the other hand, if the threshold is set to a value longer than one day, the difference in percentages of AS pairs which are classified as legitimate is marginal. Therefore, 1 day is a conservative threshold for provider and customer relation.

Third, any AS participating in an Internet Exchange Point (IXP) indirectly owns the prefix associated with the exchange points. AS belongs to Related Set if AS is an Internet Exchange Point (IXP). An In-

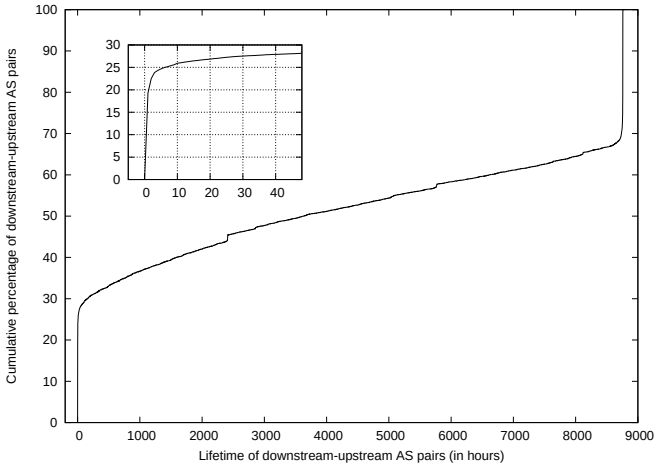

Figure 4: CDF of the lifetime of downstream-upstream AS pairs in 2009

ternet exchange point (IXP) is a physical infrastructure through which Internet service providers(ISPs) exchange Internet traffic between their networks. From the data, some ASes may offend IXP prefixes within a short period of time. For example, Cogent (AS 174) offended 10 IXP prefixes including London Internet Exchange(LINX) and Milan Internet Exchange Point on October, 7th, 2009. In addition, IXP ASes also offended other ASes' prefixes. For example, on August 08, 2009, Starhub Internet Exchange (AS 38861) was offending more than 410 prefixes. In the LRL detection scheme, it is legitimate for ASes to announce IXP prefixes and for IXPs to announce other ASes' prefixes. A list of most IXPs and their participants is obtained from UCLA IRL [4].

Fourth, ASes belonging to the same organization are related and therefore indirectly own each others assigned prefix blocks. AS belongs to related set if offender AS and victim AS are in the same organization. It is legitimate for an AS to originate other ASes' prefixes as long as they are in the same organization. This could be inferred from ASes' contact email domains. For example, AS 36625 offended AS 36617, AS36618 and other 8 ASes' prefixes on June 26th, 2009. All 11 ASes involved belong to VeriSign and share the same contact email domain "verisign.com". Hence, this is not an LRL offense. ASes' contact emails can be accessed from WHOIS [11].

In 2009, 22.02% prefixes have no related set, around 22.94% prefixes have related sets of only a single AS, and the remaining 55.04% of related sets have multiple ASes. We have also tested 2008 data and obtained similar related set results.

3.2.4 Step D: Detect Origin Conflicts

The stable and related sets together capture all the possible ASes which can legitimately announce a given prefix. Any other AS originating the prefix can be deemed as an *attacker AS*. The victims of the attack are only ASes in the stable set and not the ASes in the related set of the involved prefix. The ASes in the related set are expected to announce the prefix only in special situations and not for significant duration of time. Therefore offense against AS(es) in related set is ignored to avoid unnecessary origin conflict noise. Therefore if AS X originates prefix p and AS X $\notin$ stable set(p) and AS X $\notin$ related set (p), then AS X attacks ASes in stable set(p). The above provides a way to identify an attacker AS if there is one for any given BGP routing announcement.

However in order to detect route-leak events there needs to be a way to quantify the impact of the false routing announcement made by the attacking AS. We introduce the notion of offense value of AS which

	Method A(simple-prefix)	Method B (simple-AS)	Method (simple-set)
Total number of offense	122531	49360	15396
Offense Value=1	45446(37.1%)	43403(87.9%)	13500(87.6%)
Offense Value$\leq$2	65244(53.1%)	47887(97.0%)	15005(97.5%)
Offense Value$\leq$9	104261(85.1%)	49286(99.8%)	15376(99.9%)
Offense Value$\geq$10	18270(14.9%)	74(0.15%)	20(0.13%)
Number of statistical anomalies	74(0.06%)	30(0.06%)	9(0.06%)

Table 1: Comparison of Three Detection Methods

captures the overall impact of a false routing announcement. There are three possible methods to estimate the offense value of an AS for any false routing announcement: (A) count number of falsely originated prefixes (B) count number of attacked ASes in the stable set of involved prefix and (C) count the number of unique stable sets attacked. For example if AS X falsely originates routes for prefix p_1, p_2 and p_3 each with the same stable set of $\{$AS Y_1, $Y_2\}$ then method A counts offense value of 3, method B counts offense value of 2 and method C counts offense value of 1. Counting offending prefixes introduces noise since an AS offend many prefixes but impact only few ASes as seen in aforementioned example where AS X offends p_1, p_2 and p_3 prefixes but only impacts AS Y_1 and Y_2. Counting number of attacker ASes poses problems when multiple ASes can legitimately announce a prefix block [29]. In such cases offenses for each legitimate owner AS is noted even though the same prefix is involved thereby causing unnecessary increase in offense values. However counting number of uniquely attacked stable sets is a reasonable trade-off between counting affected prefixes and counting affected ASes.

Table 1 presents the number of offenses generated as the metrics to quantify the offense in individual false routing announcement is changed in 2008. The total number of detected offenses with method A, method B and method C are 122531, 49360 and 15396, respectively. Method A represents the counting of the number of offended prefixes. Method B represents the counting of the number of offended ASes. Finally method C represents the counting of the number of stable sets offended. For those offense events whose offense values are equal or larger than 10, 18270 offense cases are detected using method A while using method B the offense cases goes down to 74 and finally using method C produces only 20 offense cases. In favor of reducing offense noise the method C i.e. counting the number of offended stable sets is chosen as the preferred metric for counting offense value of an attacking AS generating a false routing announcement.

3.2.5 Step E: Identify Large Route Leak Events

After the offense value has been calculated in Step D, it will be compared with a threshold to determine whether it is a LRL event. The step E as shown in Figure 2, compares the calculated offense values of suspicious ASes with a threshold, and then reports LRL events. The goal of picking an appropriate threshold is to minimize false positives without detecting only the very large scale events that everyone will notice without any detection system.

Figure 5 shows the distribution of offense values of all the offenses in 2009. In the figure, the majority of offense events have very small offense value, e.g. 1, 2 and 3. Very small number of offense events have offense values larger than 10. The similar distribution is observed in 2009 data. Thus, in our implementation, we set the threshold to be 10. If the threshold is set to larger than 10, the difference in number of offense events detected is marginal. Hence, it is conservative to set the threshold of offense value to be 10.

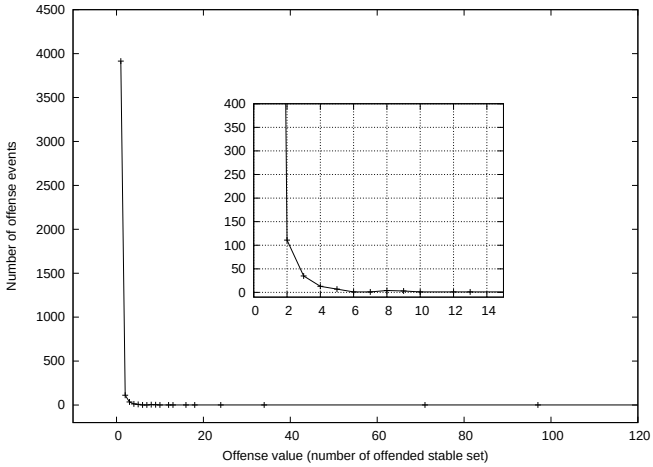

Figure 5: Distribution of the Offense Value of All Offenses in 2009

3.3 Offline versus Online Detection

In the offline detection scheme, one year of archival BGP routing message data is used to detect route-leak events. The stable set for any prefix computed over the archival BGP data is static due to the fixed prefix-announcement duration. Similarly the related set for any prefix computed over the archival data is also static since the provider-customer relationships are fixed, IXP prefixes are known beforehand and so are the contact address for ASes. Therefore the stable and related set for each prefix are pre-computed from the archival BGP data. Thereafter each BGP routing message is analyzed using the Algorithm 1 to detect LRL events.

Each BGP routing announcement composed of origin AS X and prefix p is checked for possible origin conflicts by comparing AS X against the stable and related set of prefix p. In case the BGP routing announcement is legitimate, which can only be if originating AS X either belongs to the stable or related set of the involved prefix p, then the prefix is recorded as live. The reason for recording liveness of the prefix is to catch origin conflicts only during its lifetime. Now until every AS in the stable set of prefix p withdraws it, the prefix p remains live. In case the BGP routing announcement by origin X for prefix p is false, the attacking AS X offends stable set for the prefix p if it is live i.e. only when there exists a legitimate origin announcement in the system corresponding to the prefix p. For each such origin conflict the offense value of attacking AS is updated respectively. In the event the offense value of an AS exceeds the offense threshold of 10 the AS is declared to be engaged in a LRL event. Upon withdrawal of such a false routing announcement the origin conflict disappears and the offense value of attacking AS is reduced to reflect it.

Online detection as presented in Algorithm 2 is needed to detect on-going route leak events on the Internet. The online detection is performed on a moving observation window $[t - T, t)$ of BGP routing message data. Therefore the stable and related set for any prefix are not static and need to be dynamically updated. Initially the stable and related sets for every prefix are empty. One year worth of training data is used to construct the initial stable and related set for every prefix. Archival BGP routing message data as mentioned earlier can be used as the training data. But with movement of observation window the stable and related sets need to be updated. The stable set for any prefix depends upon the prefix announcement duration meeting a day threshold. The IXP prefixes and contact information of ASes needed for related set construction is still constant and known beforehand. However the provider-customer relationships again depend upon downstream-upstream AS pair durations meeting day thresholds. Therefore prefix announcement duration is updated by tracking the announcement and withdrawal time for each prefix-origin AS pair. At the end

Algorithm 1 offline LRL detection algorithm

StableSet(p): stable set of prefix p;
RelatedSet(p): related set of prefix p;
Live(p): Prefix p is alive;
for all BGP routing messages **do**
  if BGP announces (prefix p, AS X, time t) **then**
    if (AS $X \notin$ StableSet(p) or RelatedSet(p)) AND Live(p) **then**
      AS X offends StableSet(p);
    else if $X \in$ StableSet(p) **then**
      Live(p);
    end if
  else if BGP withdraws (prefix p, AS X, time t) **then**
    if (AS $X \notin$ StableSet(p) or RelatedSet(p)) AND Live(p) **then**
      AS X stops offending StableSet(p);
    else if $X \in$ StableSet(p) **then**
      ! Live(p);
    end if
  end if
end for
Calculate AS offense value when it starts or stops offending a stable set.
Detect and report outliers if AS offense value > 10;

Algorithm 2 online LRL detection algorithm

Window(t_1, t_2) := BGP announcements and withdraw data from time t_1 to t_2.
t_0 is the current time; T = 365 days; X : detecting Xth days data
Initialize Sets : window$(t_0 - T, t_0)$
for all prefix p **do**
  Initialize stable set(p) and related set(p);
  Track announcement and withdraw time;
end for
Online Detection (Real-time BGP feed) : window$(t_0 - T + Xdays, t_0 + Xdays)$
for all BGP announcement or withdraw **do**
  Update offense value as in offline algorithm.
  Report LRL events for AS offense value ≥ 10.
end for
for all prefix origin AS pair AND downstream-upstream ASes pair **do**
  Update the lifetime, stable set and related set.
end for
Day End: move observation window to $[t_0 - T + X + 1days, t_0 + X + 1days)$

DATE	ASN	OFFVAL	AS-NAME	DURATION	POLLUTION	LOCATION
04/28/08	44237	13	JointStock Central Telecom	7.86 mins	88.89%	Russia
06/17/08	8953	108	Orange Romania AS	2.12mins	88.89%	Romania
08/26/08	24739	20	Severen-Telecom AS	18.02 mins	94.44%	Russia
09/22/08	8997	17728	OJSC NorthWest Telecom	21.66 hours	63.89%	Russia
12/14/08	29651	16	CenterTelecom Service	6.33 hours	61.11%	Russia
12/31/08	1967	17	MiddleEast Tech University	5.72 mins	27.78%	Turkey
12/31/08	6849	48	JSC UKRTELECOM	2.22 hours	94.44%	Ukraine

Table 2: Large route-leak events detected by offline LRL scheme in 2008

of the day when the window is moved, the immediate day's prefix announcement and withdrawal history is processed and first day's prefix announcement and withdrawal history is discarded. The prefix announcement duration and downstream-upstream durations are updated and so are the stable and related sets for each prefix. The BGP routing announcement and withdrawal are processed as before in the offline scheme to detect and report route leave events. The online system has been running since January 5th, 2010.

4 Evaluation

In order to evaluate the LRL offline detection scheme, archival BGP routing tables (RIB) and update messages (announcements and withdrawals) are used from Route Views [10] monitors. And in order to implement the LRL online detection scheme, real-time BGP routing tables and update messages are used from BGPMon [2]. Both offline and online LRL detection schemes need a single merged view of origin changes for each prefix as seen by all the monitors. So for each monitor, changes in origin AS for each prefix are recorded from the routing table which is initialized by the RIB table and modified by the BGP update messages. The above provides an individual view of origin changes for each prefix as seen by each monitor. Thereafter these individual views are merged together by piecing together the time-stamps of prefix and origin AS pairs as seen by the individual monitors. The Internet Exchange Point (IXP) prefixes needed as part of the related set are downloaded from IRL [4]. The AS contact information again needed as part of the related set is collected from the whois database [11]. We present the detected LRL events along with confirmations received from network operators regarding the validity of these leaks. We characterize LRL events into different types and present detailed analysis for each type through case studies. We analyze the general characteristics of LRL events and report necessary requirements from any detection and mitigation scheme used to safeguard against such leaks.

4.1 LRL Events detected in 2008 and 2009

To detect the large route-leak events, BGP RIB and update data from all the Oregon Route View monitors is used by the offline LRL scheme. Table 2 and Table 3 present the large route-leak events detected by the offline LRL scheme in the year 2008 and 2009 respectively. More comprehensive list of detected LRL events by the offline LRL scheme from 2003 to 2009 are available [5]. In 2009, 10 LRL events have been detected and in 2008, 7 LRL events have been detected. For each LRL event the table reports the exact date when the event occurred, the duration of the event as recorded by the offline LRL scheme, the AS number of the attacker as well as the name of the organization responsible for maintaining the AS in order to identify it exactly, maximum offense value achieved by the attacking AS during the attack, the percentage of monitors polluted by the attack and the geographical location where the attack originated. The offline LRL scheme

DATE	ASN	OFFVAL	AS-NAME	DURATION	POLLUTION	LOCATION
02/14/09	8895	34	KACST/ISU Riyadh	1.96 hours	95.35%	Saudi Arabia
04/07/09	36873	13	VNL1-AS	9.98 mins	90.70%	Nigeria
05/05/09	10834	97	Telefonica	3.06 hours	93.02%	Argentina
05/11/09	4795	10	INDOSATM2	7.43 mins	93.02%	Indonesia
07/12/09	29568	16	COMTEL Supernet	23.45 mins	48.84%	Romania
07/22/09	8997	170	OJSC NorthWest Telecom	59 secs	4.85%	Russia
08/12/09	4800	12	LINTASARTA-AS-AP	32 secs	93.02%	Indonesia
08/13/09	4800	71	LINTASARTA-AS-AP	7.82 hours	93.02%	Indonesia
12/04/09	31501	18	SPB-TELEPORT	68 secs	20.93%	Russia
12/15/09	39386	24	Saudi Telecom	62 secs	86.05%	Saudi Arabia

Table 3: Large route-leak events detected by offline LRL scheme in 2009

begins recording the route-leak activity as soon as the offense value of any AS hits the threshold 10. The offline LRL scheme records the AS number of the attacker to identify it accurately and to cross-reference it to a managing organization with the help of whois database. The offline scheme also records the AS number of victim ASes along with their compromised prefixes through the false routing announcements generated by the attacking AS. Again, cross-referencing the AS number of victim ASes and the attacked prefix to managing organization became crucial towards the validation efforts as explained in Section 4.2. The offline LRL scheme time-stamps the route-leak activity in order to provide accurate duration of the event. The offline LRL scheme stops recording route-leak activity only when the offense value of attacking AS falls below the threshold 10.

LRL events as reported in Table 2 and Table 3 typically last from a few minutes to a few hours. LRL events are seen to be short-lived with none of them lasting for more than a day. Therefore fast mitigation response is required from any AS trying to safeguard itself against such events. For the detected LRL events the reported maximum offense value of attacking AS shows the scale of the leak. Even if conservatively single AS is in the stable set of attacked prefix, for certain cases significant number of ASes have been impacted by the LRL events. Such route-leak events have more devastating effect on the network services due to the large number ASes and prefixes involved in the attack. For instance, detected LRL event caused by AS 8997 on September 22, 2008 offends 17728 stable sets for nearly 22 hours. The aforementioned detected LRL event has also been reported by network operators to the Nanog [7] mailing list.

4.2 Validation of detected LRL Events

We sent emails to the contact information of organizations managing the attacking ASes to figure out any legitimate operational reason for the particular route leak event. We also sent out emails to victim ASes seeking confirmation of the individual false routing announcements involved in the route-leak event. For LRL events detected in 2009, a total of 9 out of 10 events have been individually confirmed by either single victim AS or in most cases multiple victim ASes. The only remaining event not validated is caused by AS4795 on May 11, 2009 for which email replies from attacker and victim ASes are still awaited. However as shown in Section 4.3, AS4795 has been involved in repeated LRL events over the last 7 years, which suggests that it is extremely likely that the aforementioned event is another LRL event. In addition to the confirmation of LRL events detected in 2009, some replies provided detailed explanation of the leaks. For instance, AS 34397 attributes the LRL event caused by AS 8895 on February 14, 2009 to a misconfiguration error which caused disruption of service for about 2 hours. Due to the misconfiguration

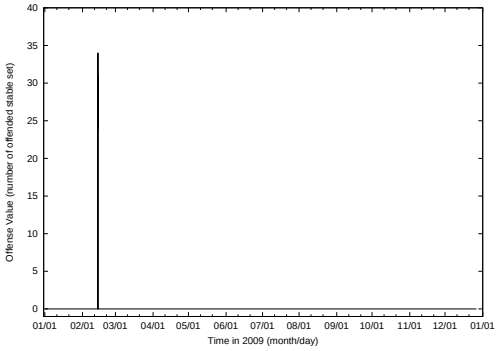

Figure 6: A Typical LRL Offense Event by AS 8895

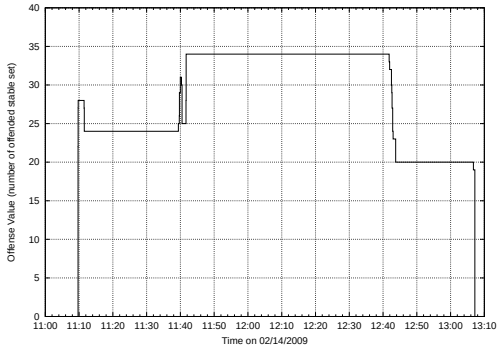

Figure 7: A Typical LRL Offense Event on one day by AS 8895

error, many local Saudi ISP prefixed were announced by KACST/ISU(AS 8895) to the Internet making it the preferred download path rather than the Saudi Telecom-IGW(AS 34397). Most of the detected LRL events have been verified by network operators providing the ground truth reality. Furthermore, having almost all events verified implies the LRL detection scheme produces near zero false positives. Surprisingly, none of the 10 events was mentioned in operator mail list such as NANOG list [6], which means that our detection results are non-trivial and useful.

4.3 LRL Event Case Studies

We have identified three different types of LRL events in the reported results. The first type is the typical LRL event occurring for a short duration within a day and posing a high offense value which corresponds to the significant disruption caused by the event. Figure 4.3 presents the offense value of AS 8895 for 2009 which remains near constant zero for whole year except on February 14, 2009. Figure 4.3 presents the change in offense value of AS 8895 for the duration of the LRL event on February 14, 2009. The LRL event starts around 11:10 AM with AS 8895 gradually attacking increasing number of ASes in the stable sets of multiple prefixes. In less than a minute, the offense value of AS 8895 jumps to 27 and remains around 27 for half an hour. Thereafter the offense value does fluctuate a couple of times but remains consistent at 34 for a duration of more than an hour. Finally the offense value of AS 8895 begins dropping and reaches near zero at around 1:10 PM. The LRL event which lasted for nearly 2 hours has been verified by multiple victim ASes.

The second type of LRL events is characterized by an attacking AS exhibiting low offense values. Figure 8 shows the change of offense value of AS 36873 which remains near constant zero for all the year except on April 7, 2009. On April 7, 2009, the offense value of AS 36873 jumps to 13 and the event lasted for about 10 minutes. Although the offense value on April 7 is 13 and just satisfies the threshold, it is still an abnormal case for AS 36873 based on Figure 8 and has been confirmed by replies from victims ASes. The third type of case is identified where an individual AS has been involved in multiple LRL events over the years. After running the offline LRL detection scheme over several years of BGP archival data, AS 4795 is found to be responsible for multiple LRL events. Figure 9 presents the change in the offense value of AS 4795 over the past 7 years. As is clearly evident from the result, the offense value of AS 4795 has exceeded the threshold 10 in the years 2004, 2005, 2007 and 2009. Even though AS 4795 has not been confirmed by emails, it is a big chance that those cases were abnormal events because of the offense history of AS4795 in the past 7 years.

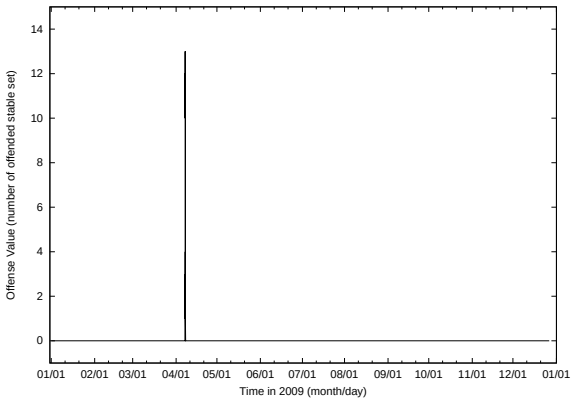

Figure 8: LRL Offense Event With Low Offense Value

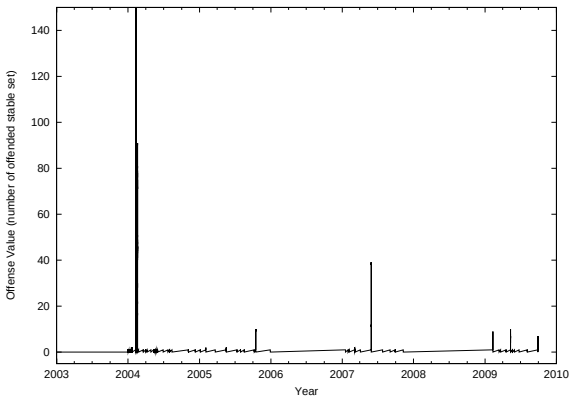

Figure 9: Repetition of LRL Offense Events over years

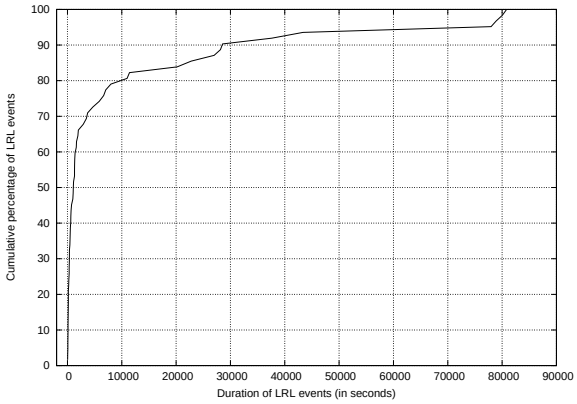

Figure 10: Duration of LRL events from 2003 to 2009

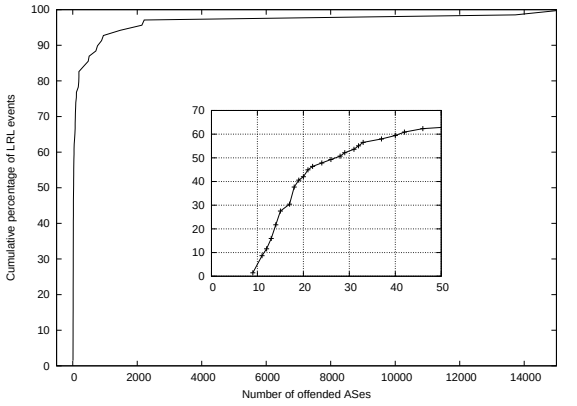

Figure 11: number of offended ASes by LRL events from 2003 to 2009

4.4 LRL Events Characteristics

We now investigate specific characteristics of the verified LRL events. LRL events are seen to be short-lived in nature with most of them lasting less than 3 hours. Furthermore LRL events are also seen to impact a significant number of monitors which is representative of the wide range of ASes impacted during the attack. In this section, we evaluate these two unique characteristics of LRL: short-liveness and significant disruption.

4.4.1 short-liveness

Figure 3 presents the CDF of the duration of detected LRL events from 2003 to 2009. The majority of the LRL events are extremely short-lived not lasting for more than a few hours. Nearly 80% of LRL events last less than 3 hours which implies LRL detection and mitigation needs to be fast. Therefore the online detection scheme is setup to get real-time BGP monitor feeds from BGPMon [2] and is able to detect LRL events in matter of seconds. LRL detection results having negligible false positives can be trusted and therefore intermediate ASes can save reaction time from any on-going attacks.

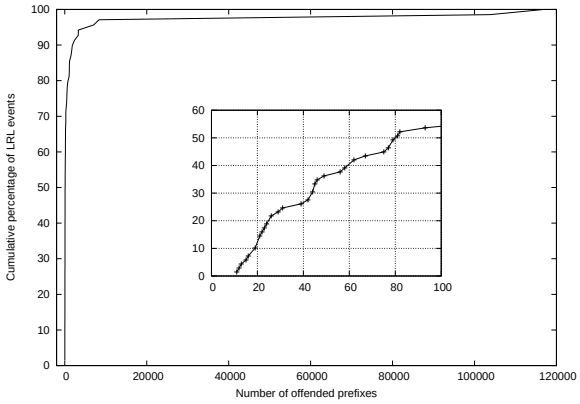

Figure 12: number of offended prefixes by LRL events from 2003 to 2009

4.4.2 significant disruption

The attacking AS generates a large number of false routing announcements during LRL events which affects large number of prefixes and ASes. Figure 12 presents the CDF of the number of prefixes offended by attacking AS during detected LRL events from 2003 to 2009. In about 50% of the LRL events, the attacking AS offended more than 76 prefixes. Therefore the average LRL event is expected to disrupt data traffic for nearly 76 prefixes. Furthermore, two specific LRL events in 2004 and 2008 have offended more than 100,000 prefixes, which shows the potential for huge disruptive behavior by any LRL event. Figure 11 shows the CDF of the number of victimized ASes during detected LRL events from 2003 to 2009. In this figure, for about 50% of LRL events the attacking AS offended more than 24 ASes. The disruptive behavior of LRL event can also be estimated by measuring the percentage of monitors affected by the false routing announcements of the attacking AS during the event. Any monitor which accepts the false routing announcements by the attacking AS during a LRL event is considered to be polluted. These monitors directly peer with border routers of ASes across the Internet. Therefore any corruption in the monitor's routing table implies corruption in the corresponding AS routing table. Counting the number of polluted monitors therefore provides a rough estimation to the degree of disruption caused by a LRL event on the Internet. High degree of pollution is reported in Table 2 and Table 3, which implies significant number of ASes are impacted by LRL events. Figure 13 shows the CDF of the percentage of polluted monitors by LRL events. For most of the LRL event the number of polluted monitors is significantly high showing the vulnerability of the network to such attacks. For instance, nearly 80% of the LRL events pollute more than 60% of the Route Views monitors reflecting the huge disruption caused by majority of the LRL events.

4.5 The Fast Response of LRL Detection Scheme

Intermediate ASes are in need of a fast and accurate detection system in order to protect their data traffic. LRL detection scheme is accurate and has been validated by victim ASes as mentioned in Section 4.2. Minimizing false positives allows networks to respond to attacks quickly, maybe even automate the response at the network operation center. Small number of detected results also help to accelerate the processing time. Figure 14 shows the number of LRL events reported from 2003 to 2009. We identify 5 to 20 large route leaks each year. This is significantly better than previous results. For example, [20] generated around 20 alarms daily. In addition, LRL detection scheme is able to set up the alarm within seconds and networking operator can mitigate the damage accordingly before the damage is made.

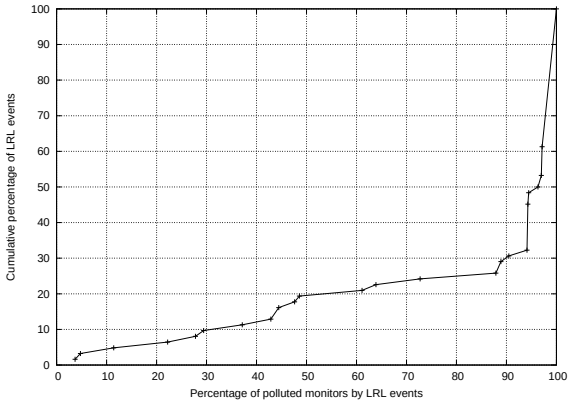

Figure 13: Percentage of polluted monitors LRL events from 2003 to 2009

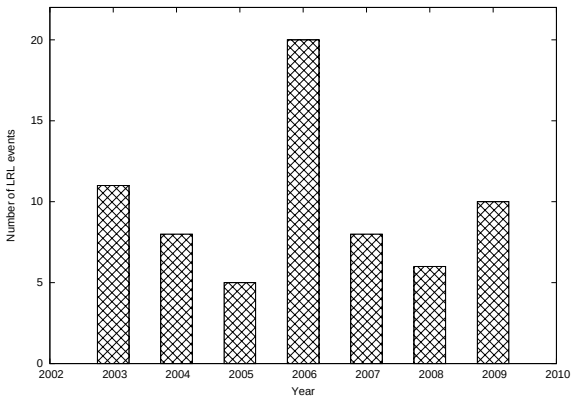

Figure 14: the number of LRL events per year from 2003 to 2009

5 Related Work

There have been three different kinds of solutions proposed to eradicate the problem of false routing announcement: prevention [17, 19, 23, 16, 26], detection [28, 21, 18, 20, 14, 3] and mitigation [24, 25, 27].

The prevention techniques attempt to restrict ASes from making false routing announcements. S-BGP [17] makes use of strict hierarchical public key infrastructures (PKIs) for both AS number authentication and IP prefix ownership verification. Routers are expected to sign and verify origin AS and AS path information which makes routine routing tasks computationally expensive. So-BGP [19] proposes a secure database which maintains authenticated topology and prefix ownership and only allows signed updates to the database to avoid tampering. These proposals require extensive cryptographic key distribution infrastructure and/or a trusted central database. Listen & Whisper [23] monitors the route validity by passively probing the data plane to different destinations and imposing cryptographic chains on the control plane to check for inconsistencies. PG-BGP [16] uses route history to validate BGP update messages and delays the usage and propagation of new routes in favor of known trusted alternatives. QBGP [26] avoids forwarding data traffic on suspicious paths but still propagates these paths in order to facilitate the attack detection.

The detection techniques focus on identifying prefix hijack events through control plane or data plane based monitoring. LOCK [21] and iSPY [28] actively monitor network paths to the owner AS in order to detect any on-going hijack events or to identify the hijacker carrying out the attack. Monitoring the data plane allows accurate and timely detection of prefix hijack events but only for specific prefixes since it require frequent probing of the network paths which is impractical for every prefix. Jian et. all [20] and IAR [3] attempts to find bogus routes by searching for inconsistencies such as suspicious routes and unseen objects in the control plane. At the same time systems such as PHAS [18], Cyclops [14] and MyASN [8] account for the input provided by network operators while searching for inconsistencies in the control plane. The control plane can be monitored to search for any false routing announcements related to any prefix but this generates a large number of alarms.

Once the false routing announcement is identified the next task is to mitigate such an attack. To mitigate and purge the false routing announcement the owner of the prefix can contact the offending network or its upstream provider to filter the false routing announcement. Prefix limiting is another practical solution adopted by network operators which caps the number of prefixes allowed to be advertised over eBGP sessions thereby limiting the possibility of full table leaks. There exist several other techniques for mitigating prefix hijack attacks such as installation of general filters by provider networks, route purge-propagation [27], ACR [24] and MIRO [25]. Route purge [27] attempts to suppress BGP routes which are deemed to be suspicious and in effect promotes propagation of trustworthy routes to mitigate the impact of attacks. ACR [24] and MIRO [25] focus on providing multiple routes any of which can be used for data delivery in the eventuality of primary route being compromised. However we have not seen any specific technique directed towards the detection and resolution of route leak events which keeps reemerging on the Internet.

6 Conclusions

By identifying suspicious routing announcements based on past prefix-origin announcement history and correlating them along time dimension, our algorithm can effectively detect large route leak events. In the past seven years, there were 5 to 20 events detected each year. These events typically lasted from a few minutes to a few hours and affected most monitors which implies these events can inflict significant damage to data traffic. In 2009, none of the detected results have been reported in Nanog but most have been individually confirmed by network operators. With the online version of the algorithm and no false positives in detected results, it is possible to enable real-time response to these large route leak events by intermediate networks. Our detection method only needs BGP updates as input; it does not require knowledge from the

real prefix owner. Our detection results using the past seven year's data also provide a collection of events that can be used for evaluating other prefix hijacking events.

References

[1] AS 7007 incident. `http://en.wikipedia.org/wiki/AS_7007_incident`.

[2] BGPmon. http://bgpmon.netsec.colostate.edu/.

[3] Internet Alert Registry. http://iar.cs.unm.edu/.

[4] Internet Topology Collection. http://irl.cs.ucla.edu/topology.

[5] LSRL Detection Results from 2003 to 2009. http://dyadis.cs.arizona.edu/projects/lsrl-events-from-2003-to-2009.

[6] North American Network Operators' Group. http://www.nanog.org.

[7] Prefix hijack by ASN 8997. http://www.merit.edu/mail.archives/nanog/2008-09/msg00704.html.

[8] RIPE myASn System. http://www.ris.ripe.net/myasn.

[9] RIPE RIS Raw Data. http://www.ripe.net/projects/ris/rawdata.html.

[10] University of Oregon Route Views Archive Project. http://www.routeview.org.

[11] Whois Database. http://www.whois.net/.

[12] YouTube Hijacking: A RIPE NCC RIS case study. http://www.ripe.net/news/study-youtube-hijacking.html.

[13] H. Ballani, P. Francis, and X. Zhang. A Study of Prefix Hijacking and Interception in the Internet. In *SIGCOMM '07: Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 265–276, New York, NY, USA, 2007. ACM.

[14] Y.-J. Chi, R. Oliveira, and L. Zhang. Cyclops: the as-level connectivity observatory. *SIGCOMM Comput. Commun. Rev.*, 38(5):5–16, 2008.

[15] L. Gao. On Inferring Autonomous System Relationships in the Internet. In *IEEE ACM Transactions on Networking*, volume 9, pages 733–745, 2000.

[16] J. Karlin, S. Forrest, and J. Rexford. Pretty Good BGP: Improving BGP by Cautiously Adopting Routes. In *ICNP '06: Proceedings of the Proceedings of the 2006 IEEE International Conference on Network Protocols*, pages 290–299, Washington, DC, USA, 2006. IEEE Computer Society.

[17] S. Kent, C. Lynn, J. Mikkelson, and K. Seo. Secure Border Gateway Protocol (S-BGP). *IEEE Journal on Selected Areas in Communications*, 18:103–116, 2000.

[18] M. Lad, D. Massey, D. Pei, Y. Wu, B. Zhang, and L. Zhang. PHAS: A Prefix Hijack Alert System. In *USENIX-SS'06: Proceedings of the 15th conference on USENIX Security Symposium*, Berkeley, CA, USA, 2006. USENIX Association.

[19] J. Ng. Extensions to BGP to Support Secure Origin BGP, April 2004. ftp://ftp-eng.cisco.com/sobgp/drafts/draft-ng-sobgp-bgp-extensions-02.txt.

[20] J. Qiu, L. Gao, S. Ranjan, and A. Nucci. Detecting Bogus BGP Route Information: Going Beyond Prefix Hijacking. In *Security and Privacy in Communications Networks and the Workshops, 2007. SecureComm 2007. Third International Conference on*, pages 381–390, 2007.

[21] T. Qiu, L. Ji, D. Pei, J. Wang, J. Xu, and H. Ballani. Locating Prefix Hijackers using LOCK . In *Proceedings of 18th USENIX Security Symposium*, 2009.

[22] G. Siganos and M. Faloutsos. Neighborhood Watch for Internet Routing: Can We Improve the Robustness of Internet Routing Today? In *INFOCOM*, pages 1271–1279. IEEE, 2007.

[23] L. Subramanian, V. Roth, I. Stoica, S. Shenker, and R. H. Katz. Listen and Whisper: Security Mechanisms for BGP. In *NSDI'04: Proceedings of the 1st conference on Symposium on Networked Systems Design and Implementation*, pages 10–10, Berkeley, CA, USA, 2004. USENIX Association.

[24] D. Wendlandt and I. Avramopoulos. Dont secure routing protocols, secure data delivery. In *In Proc. 5th ACM Workshop on Hot Topics in Networks (Hotnets-V*, 2006.

[25] W. Xu and J. Rexford. MIRO: Multi-path Interdomain ROuting, 2006.

[26] M. Zhang, B. Liu, and B. Zhang. Safeguarding data delivery by decoupling path propagation and adoption. In *INFOCOM*, 2010.

[27] Z. Zhang, Y. Zhang, Y. C. Hu, and Z. M. Mao. Practical defenses against bgp prefix hijacking. In *CoNEXT '07: Proceedings of the 2007 ACM CoNEXT conference*, pages 1–12, New York, NY, USA, 2007. ACM.

[28] Z. Zhang, Y. Zhang, Y. C. Hu, Z. M. Mao, and R. Bush. iSPY: Detecting IP Prefix Hijacking on My Own. In *SIGCOMM*, pages 327–338, 2008.

[29] X. Zhao, D. Pei, L. Wang, D. Massey, A. Mankin, S. F. Wu, and L. Zhang. An Analysis of BGP Multiple Origin AS (MOAS) Conflicts. In *IMW '01: Proceedings of the 1st ACM SIGCOMM Workshop on Internet Measurement*, pages 31–35, New York, NY, USA, 2001. ACM.

[30] X. Zhao, D. Pei, L. Wang, D. Massey, A. Mankin, S. F. Wu, and L. Zhang. Detection of Invalid Routing Announcement in the Internet. In *DSN '02: Proceedings of the 2002 International Conference on Dependable Systems and Networks*, pages 59–68, Washington, DC, USA, 2002. IEEE Computer Society.

[31] C. Zheng, L. Ji, D. Pei, J. Wang, and P. Francis. A Light-Weight Distributed Scheme for Detecting IP Prefix Hijacks in Real-Time. In *SIGCOMM '07: Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 277–288, New York, NY, USA, 2007. ACM.