RoLL: Real-time and Accurate Route Leak Location with AS Triplet Features

Jiang Li*, Jiahao Cao*†§, Zili Meng†, Renjie Xie†, and Mingwei Xu*†‡§¶‖
* Department of Computer Science and Technology, Tsinghua University, Beijing, China
† Institute for Network Sciences and Cyberspace, Tsinghua University, Beijing, China
‡ Beijing National Research Center for Information Science and Technology, Beijing, China
§ Zhongguancun Laboratory, Beijing, China
¶ Quan Cheng Laboratory, Shandong, China
‖Peng Cheng Laboratory, Shenzhen, China
{lijiang17@mails, caojh2021, xumw, xrj21@mails}.tsinghua.edu.cn, zilim@ieee.org

***Abstract*—BGP is the only inter-domain routing protocol that plays an important role on the Internet. However, BGP suffers from route leak, which can cause serious security threats. To mitigate the effects of route leak, accurate and timely route leak location is of great importance. Prior studies leverage AS business relationships to locate route leak in real time. However, they fail to achieve high location accuracy. Recent studies apply machine learning to accurately detect route leak from statistical features of massive BGP messages. Nevertheless, they have high detection latency and cannot further locate route leak. In this paper, we propose a real-time and accurate route leak location system named *RoLL*. It leverages distinctive AS triplet features to accurately locate AS triplets with route leak from each BGP update message in real time. Our experimental results on real-world BGP route leak data demonstrate that RoLL can achieve 91% location accuracy with less than 10 ms location latency.**

***Index Terms*—Border Gateway Protocol (BGP), Route Leak, Real-time and Accurate Location**

I. Introduction

Border Gateway Protocol (BGP) plays an important role on the Internet since it is the only inter-domain routing protocol to connect heterogeneous networks, aka Autonomous Systems (ASes), around the world. Unfortunately, BGP lacks security guarantees for route exchange, resulting in many security incidents [1–5]. One of the most BGP security threats is route leak, i.e., routes of ASes propagate to unintended ASes. Route leak can cause large-scale network performance degradation and network unreachablity [6]. Besides, attackers can leverage route leak to intercept or drop AS-level traffic [7]. According to the report [8], there are more and more route leak events in recent years, causing severe global impacts. For example, Google goes down for about 2 hours due to route leak [4]. Another route leak incident [5] disrupts the connectivity for thousands of networks globally.

To mitigate the effects of route leak, accurate and timely route leak location is of great importance. AS operators can clearly understand which ASes leak routes and take countermeasures in time, such as filtering out leaked routes [9]. Moreover, the effects and economic loss due to route leak can be effectively reduced as soon as possible. Researchers have presented several methods on locating route leak in real time according to AS business relationships [10–12]. Specifically, they check whether the AS relationships on a BGP path violate the valley-free principle [13]. Although they can accurately infer the AS relationship for one AS link, they fail to achieve accurate route leak location. As it requires knowing the business relationships of two AS links at the same time [14], the final location accuracy will drop substantially due to multiplication rule of probability. Besides, they cannot locate route leak happening in AS links that are invisible before [15].

Rather than locate route leak based on AS relationships, some studies [16–18] apply machine learning to directly identify route leak from statistical features of massive BGP messages. Although they can achieve accurate route leak detection, they cannot locate which ASes have route leak. Consequently, it still takes many BGP security experts a long time to locate route leak. Besides, these methods require a long time interval to periodically calculate statistical features, which results in a long detection delay. For example, MSLSTM [17, 19] collects and extracts statistical features from BGP messages every 8 minutes for high accuracy. There remain great challenges for them to conduct real-time route leak location while maintaining high accuracy.

In this paper, we propose a real-time and accurate route leak location system named *RoLL*. It extracts AS triplets from the AS_PATH of each BGP update message, and identifies AS triplets with route leak via AS triplet features. Unlike statistical BGP features that must be collected in a long time interval, our AS triplet features are relatively static and stable features that can be collected in advance from multiple sources [20–23]. Hence, it enables RoLL to locate route leak from each BGP update message in real time. To find out AS triplet features that can effectively distinguish AS triplets with route leak from legitimate AS triplets, we conduct a comprehensive measurement study on real route leak data. We identify five distinctive triplet features, i.e., AS distance triplet, AS degree triplet, AS address space triplet, AS geographic location triplet, and AS type triplet. Combined with these triplet features, RoLL applies a machine learning model to accurately locate AS triplets with route leak from BGP messages in real time.

We collect 1130 real-world route leak events, extracting 578 route leak AS triplets and 2333 legitimate AS triplets.

We conduct extensive experiments to evaluate the route leak location performance of RoLL. It achieves high route leak location performance, e.g., 91% location accuracy and 92% recall rate. Compared to the prior location methods based on business relationships [14, 24, 25], RoLL has more than 11% improvement both on accuracy and recall rate. Compared to the prior methods based on machine learning that can only detect route leak [16, 17, 19], RoLL also has more than 1% accuracy improvement. Furthermore, RoLL can locate route leak from a BGP message within 10 ms. In contrast, it takes at least 1 min for the prior machine learning based methods to detect route leak. Our experimental results demonstrate that RoLL can accurately locate route leak in real time.

To summarize, our paper makes the following contributions:

- We design a system named RoLL that can accurately locate route leak from BGP messages in real time.
- We conduct a comprehensive measurement study on real route leak data to identify distinctive AS triplet features.
- We conduct extensive experiments to demonstrate the location performance of RoLL.

We have released our source code of RoLL and the evaluation dataset on Github: `https://github.com/yangtzeriverli/RoLL.git`.

II. Background

Border Gateway Protocol (BGP). BGP [26] is a policy-based routing protocol to exchange routing and reachability information among Autonomous Systems (ASes). As BGP update messages contain the IP prefix that AS originates and the AS path along which receiver ASes can reach the prefix, ASes can choose their best routes mainly according to their business relationships with neighbor ASes [27]. There are mainly two kinds of business relationships between ASes, i.e., provider-to-customer (or customer-to-provider) and peer-to-peer. In the provider-to-customer relationship, a provider AS provides transit service to a customer AS and the latter pays the former for the transit service. In the peer-to-peer relationship, the two ASes exchange their traffic and their customers' traffic without charge. Based on the AS business relationship, Gao et al. [13] propose the Gao-Rexford routing principle. It specifies that BGP update messages should always propagate along a *valley-free* path. For an AS, routes from its customer ASes can be advertised to its neighbor ASes. However, routes from its peer ASes can only be advertised to its customers, and routes from its provider ASes can only be advertised to its customers.

Route Leak. Route leak occurs when BGP update messages propagate to unintended ASes, i.e., violating the Gao-Rexford model [13]. Fig. 1 shows nine scenarios on the propagation of BGP update messages. Here, all the BGP update messages in the figure propagate from left to right. The four AS triplets marked red violate the Gao-Rexford principle, and thus result in route leak. The others are legitimate AS triplets.

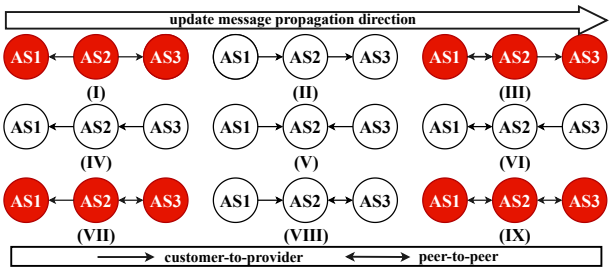


Fig. 1. Scenarios on BGP update message propagation. Here, the red AS triplets have route leak due to the violation of the valley-free principle.

III. AS Triplet Features for Route Leak

In this section, we study the AS triplet features that can be used to distinguish whether there are route leak incidents in AS triplets. For simplicity, we use *leak AS triplets* to denote AS triplets with route leak, and *legitimate AS triplets* to denote AS triplets without route leak. Based on our analysis, we have identified five discriminative AS triplet features between leak and legitimate AS triplets, which are summarized in Table I. These features can be divided into two categories, i.e., AS topology features and AS node features. AS topology features come from the AS topology, which describe the topological attributes of ASes. AS node features come from the features of ASes themselves. We detail them in the following subsections.

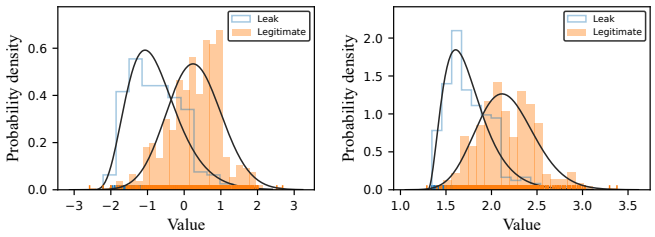
Dataset Collection. We build a dataset containing legitimate AS triplets and AS triplets with route leak from `BGPstream` [28]. We collect 1130 route leak events, extracting 578 leak AS triplets and 2333 legitimate AS triplets from AS_PATHes that `BGPstream` [28] provides. For example, considering the AS_PATH of [205148, 9002, 3356, 3257, 3320, 8373], the route is leaked by AS_{3257} to AS_{3356}. We extract an AS triplet with route leak, i.e., $\langle 3356, 3257, 3320\rangle$. The remaining AS triplets such as $\langle 3257, 3320, 8373\rangle$ are legitimate AS triplets.

TABLE I
Discriminative AS triplet features

Category	Triplet Feature	Description
AS Topology Feature	AS Distance Triplet $\langle d_1, d_2, d_3\rangle$	d_i denotes the average distance from AS_i to clique ASes
	AS Degree Triplet $\langle \eta_1, \eta_2, \eta_3\rangle$	η_i denotes the degree of AS_i
AS Node Feature	AS Address Space Triplet $\langle a_1, a_2, a_3\rangle$	a_i denotes the IPv4 address space of AS_i
	AS Geographic Location Triplet $\langle g_1, g_2, g_3\rangle$	g_i denotes RIR, country or IXP in which AS_i is located
	AS Type Triplet $\langle t_1, t_2, t_3\rangle$	t_i denotes the type of AS_i, i.e., content, enterprise and transit/access

A. AS Distance Triplet

As the Internet is hierarchical, each AS has an implicit position in the Internet hierarchy. Clique ASes or Tier-1 ASes [29] are at the top of the hierarchy and connect with each other to form a full-mesh structure for settlement-free peering. Typically, high-layer ASes provide transit service to low-layer ASes. However, if there are route leak incidents in AS triplets, low-layer ASes may provide transit service to high-layer ASes. Hence, the hierarchy relationship among the three ASes in an AS triplet can reflect potential route leak. We can use the distance from an AS to clique ASes to denote the hierarchy of the AS. As there are multiple clique ASes, we use the average



(a) $d_1 - d_2 + d_3 - d_2$ (b) $(d_1 + d_3)/d_2$

Fig. 2. The differences of the distance to clique ASes between leak and legitimate AS triplets.

hops from the AS to all the clique ASes as the distance. Here, we do not use the minimum hops because most ASes have the same minimum hops but different average hops to the clique ASes. For an AS triplet $\langle AS_1, AS_2, AS_3\rangle$, we can construct an AS distance triplet $\langle d_1, d_2, d_3\rangle$ from AS topology [30].

Fig. 2 shows the differences of the distance to clique ASes between leak and legitimate AS triplets. As we cannot draw figures to directly denote the differences among distance triplets, we show the processed result after simple calculations. Fig. 2(a) shows the probability density of $d_1 - d_2 + d_3 - d_2$, which is the sum of distance difference in an AS triplet $\langle AS_1, AS_2, AS_3\rangle$. Fig. 2(b) shows the probability density of $(d_1 + d_3)/d_2$, which means the relative distance difference in an AS triplet. As we can see, there is a clear distinction between leak and legitimate AS triplets. This is because for a leak AS triplet $\langle AS_1, AS_2, AS_3\rangle$, $d_1, d_3 < d_2$ holds at most cases. However, for a legitimate AS triplet $\langle AS'_1, AS'_2, AS'_3\rangle$, $d'_1, d'_3 > d'_2$ holds at most cases.

B. AS Degree Triplet

AS degree is the number of neighbors that an AS directly connects to. It reflects the connectivity of an AS on the Internet. Provider ASes such as Tier-1 AS typically provide transit service to a number of customer ASes, which results in rich connectivity and high degree. On the contrary, customer ASes like stub ASes, purchase transit service from a few provider ASes, which results in low degree. Thus, high-degree ASes probably provide transit service to low-degree ASes. However, when route leak incidents occur, low-degree ASes probably provide transit service to high-degree ASes. Hence, the degrees of the three ASes in an AS triplet may reflect potential route leak. For an AS triplet $\langle AS_1, AS_2, AS_3\rangle$, we construct an AS degree triplet $\langle \eta_1, \eta_2, \eta_3\rangle$ from AS topology [30].

Fig. 3 shows the proportion of different AS degree triplets between leak and legitimate AS triplets. We sort the AS degree of each AS triplet in descending order, resulting in 6 categories of AS degree triplets. We can see that the proportion of legitimate AS triplets is larger than that of leak AS triplets for the first, third and fourth and sixth categories. This is because high-degree ASes provide transit service to low-degree ASes at most cases. However, the proportion of leak AS triplets is larger than that of legitimate AS triplets for the second and fifth categories. These two categories imply that low-degree ASes provide transit service to high-degree ASes at most cases

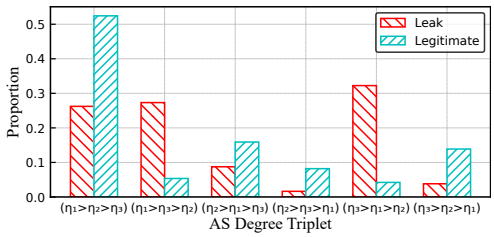


Fig. 3. The proportion of different AS degree triplets between leak and legitimate AS triplets. We sort the AS degree in each AS triplet in descending order, resulting in 6 categories of AS degree triplets.

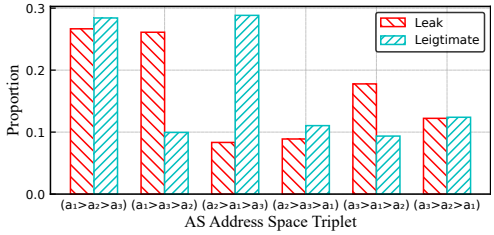


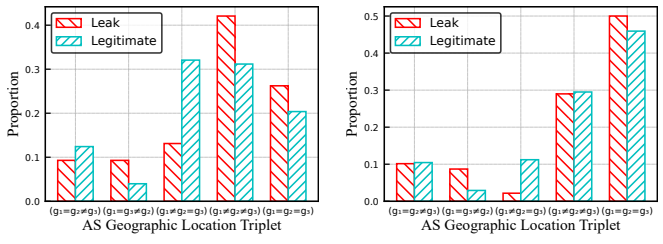
Fig. 4. The proportion of different AS address space triplets between leak and legitimate AS triplets. We sort the number of AS addresses in each AS triplet in descending order, resulting in 6 categories of AS address space triplets.

when route leak incidents occur. We also notice that there are a few leak AS triplets in the first, third and fourth and sixth categories, as well as a few legitimate AS triplets in the second and fifth categories. It is because there exist some cases where high-degree ASes do not provide transit service to low-degree ASes.

C. AS Address Space Triplet

Each AS originates a number of IP addresses that constitute the address space of the AS. ASes with big address space provide transit service to ASes with small address space at most cases since the former have more powerful network capacity. If ASes with small address space provide transit service to ASes with big address space, there are likely to be route leak. Hence, the address space relationship among the three ASes in an AS triplet can reflect potential route leak. We can obtain AS address space from `CAIDA prefix2as` dataset [31] and construct an AS address space triplet $\langle a_1, a_2, a_3\rangle$ for an AS triplet $\langle AS_1, AS_2, AS_3\rangle$.

Fig. 4 shows the proportion of different AS address space triplets between leak and legitimate AS triplets. We sort the number of AS addresses of each AS triplet in descending order, resulting in 6 categories of AS address space triplets. We find that the proportion of leak AS triplets is larger than that of legitimate AS triplets in the second and fifth categories. For a leak AS triplet $\langle AS_1, AS_2, AS_3\rangle$, $a_2 < a_1$ *and* $a_2 < a_3$ typically holds. The proportion of legitimate AS triplets is larger than that of leak AS triplets in the other four categories. These four categories imply that ASes with big address space usually provide transit service to ASes with small address space. We also notice that the proportion of legitimate AS triplets and leak AS triplets have little difference in the first and sixth categories. There exist a few cases where ASes with



(a) Country Dimension (b) IXP Dimension

Fig. 5. The proportion of different AS geographic location triplets between leak and legitimate AS triplets. We classify AS geographic location triplets based on if the ASes in an AS triplet are in the same geographic location.

big address space do not provide transit service to AS with small address space.

D. AS Geographic Location Triplet

ASes have geographic locations in the real world. Here, we consider AS geographic locations in two dimensions, i.e., country location and Internet Exchange Point (IXP) location. Typically, two ASes in the same location will not use transit service of ASes in another locations to reach each other. Thus, for an AS triplet $\langle AS_1, AS_2, AS_3\rangle$, if AS_1 and AS_3 are in the same location while AS_2 is in another location, it is likely that route leak occurs. Besides, ASes in the same IXP typically have peer-to-peer relationship with each other. Therefore, if all three ASes of a triplet are in the same IXP, there is probably route leak in the AS triplet. Hence, AS geographic location relationship among the three ASes in an AS triplet can reflect potential route leak. We can get AS geographic location triplet $\langle g_1, g_2, g_3\rangle$ for an AS triplet $\langle AS_1, AS_2, AS_3\rangle$ from `CAIDA AS organization`, `CAIDA IXP` [32, 33] and `PeeringDB` datasets [34].

Fig. 5 shows the proportion of different AS geographic location triplets between leak and legitimate AS triplets. We classify AS geographic location triplets based on if the ASes in an AS triplet are in the same geographic location. We find the proportion of leak AS triplets is larger than that of legitimate AS triplets in the second category for both location dimensions. For the third category, the proportion of legitimate AS triplets is larger than that of leak AS triplets. It is because ASes typically first purchase access service from ASes in the same location for convenience. Then, the access service ASes purchase transit service from ASes in another locations for global Internet connectivity. We can see that the first category presents similar results as the third category for both location dimensions. In the fifth category, leak AS triplets own larger proportion than legitimate AS triplets in the IXP dimension. It is consistent with our insight, i.e., if all the three ASes of a triplet are in the same IXP, there is likely to be route leak. Here, we focus on the relationship among geographical locations of ASes in triplets rather than the exact AS geographical location.

E. AS Type Triplet

There are different types of ASes playing different roles on the Internet, which can be typically divided into content

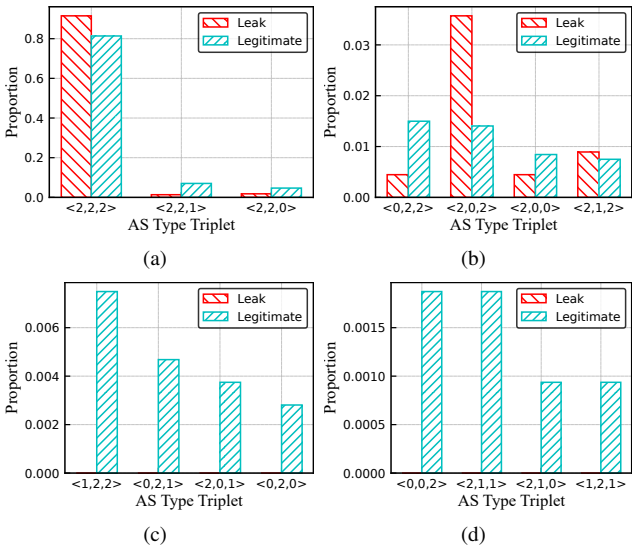


Fig. 6. The proportion of different AS type triplets between leak and legitimate AS triplets. We use 0, 1, and 2 to denote an AS belonging to content, enterprise, and transit/access, respectively.

ASes, enterprise ASes, and transit/access ASes. Content ASes provide content hosting and distribution. Enterprise ASes are owned by various organizations, universities, and companies at the network edge. Transit/access ASes are transit or access providers. Typically, transit/access ASes provide transit/access service to enterprise and content ASes. However, if enterprise and content ASes provide transit service to transit/access ASes, it may be due to route leak. Hence, AS types of the three ASes in an AS triplet can reflect potential route leak. We can obtain AS types from `CAIDA AS classification` dataset [35] and construct an AS type triplet $\langle t_1, t_2, t_3\rangle$ for an AS triplet $\langle AS_1, AS_2, AS_3\rangle$.

Fig. 6 shows the proportion of different AS type triplets between leak and legitimate AS triplets. Theoretically, there should be 27 kinds of AS type triplets. However, our collected data show that there are only 15 kinds of AS type triplets in practice. In Fig. 6(a) and Fig. 6(b), we find that most AS type triplets are $\langle 2, 2, 2\rangle$ since about 65% ASes are transit/access ASes [35]. $\langle 2, 2, 1\rangle$, $\langle 2, 2, 0\rangle$ and $\langle 0, 2, 2\rangle$ are the common legitimate AS type triplets. It means that access ASes provide access service for content and enterprise ASes, and transit ASes provide transit service for access ASes. For $\langle 2, 0, 0\rangle$, the proportion of legitimate AS triplets is larger than that of leak AS triplets. This is possibly due to content redirection between two content ASes [36]. However, we find that there are more leak AS triplets than legitimate AS triplets for $\langle 2, 0, 2\rangle$ and $\langle 2, 1, 2\rangle$. It is likely to cause route leak when content and enterprise ASes provide transit service to transit/access ASes. In Fig. 6(c) and Fig. 6(d), there are almost no leak AS triplets. $\langle 1, 2, 2\rangle$, $\langle 0, 2, 1\rangle$, $\langle 0, 2, 0\rangle$ and $\langle 1, 2, 1\rangle$ indicate that transit ASes provide transit service for content and enterprise ASes. These are all legitimate AS type triplets. For $\langle 0, 0, 2\rangle$, this is like the case for $\langle 2, 0, 0\rangle$ as before. There are more triplets in $\langle 2, 0, 0\rangle$ than $\langle 0, 0, 2\rangle$ in our dataset because route collectors

are often closer to transit/access AS than content AS. For $\langle 2,0,1\rangle$, $\langle 2,1,1\rangle$ and $\langle 2,1,0\rangle$, these triplets are legitimate, which may be due to dedicated connection between enterprise and content ASes.

IV. RoLL Architecture

The architecture of RoLL is shown in Fig. 7, which consists of three modules, i.e., preprocessor, AS triplet feature extractor, and leak locator. The preprocessor module receives live BGP update messages from route collectors, including `RIS Live` [37] and `Route Views Stream` [22]. It extracts AS triplets from AS_PATH in update messages and delivers them to AS triplet feature extractor. The AS triplet feature extractor collects feature data from multiple sources periodically and stores the data in the database. Then, it delivers AS triplets and AS triplet features to the leak locator. The leak locator filters out legitimate AS triplets and locates AS triplets with route leak. For each leak AS triplet, it sends a real-time notification to subscribers.

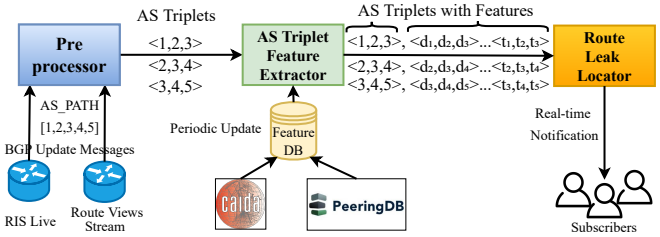


Fig. 7. RoLL Architecture.

Preprocessor. The module extracts AS triplets from received BGP update messages. It scans the AS_PATH in update messages and extracts AS triplets one by one. For example, given AS_PATH [1,2,3,4,5], the module extracts $\langle 1,2,3\rangle$, $\langle 2,3,4\rangle$ and $\langle 3,4,5\rangle$. As AS_SET is used in the route aggregation [26], the module also generates the corresponding triplets for each AS in AS_SET. For example, given AS_PATH [1,2,{3,4,5}], the module extracts $\langle 1,2,3\rangle$, $\langle 1,2,4\rangle$ and $\langle 1,2,5\rangle$. Moreover, the module filters out three kinds of anomaly BGP messages in advance. First, it drops the update message whose AS_PATH contains a loop since BGP router always drops it for loop prevention. Second, it removes the continuous repeated ASes. It is typically a result of path padding for traffic engineering [38]. For example, given AS_PATH [1,3,3,3,5], it only extracts $\langle 1,3,5\rangle$. Third, the module removes the reserved AS number in the AS_PATH introduced by configuration errors. These AS numbers should not appear in inter-domain routing [39]. For example, given AS_PATH [1,2,65000,3], the module only extracts $\langle 1,2,3\rangle$.

AS Triplet Feature Extractor. The module collects features of ASes mentioned in Section III from multiple data sources. The module stores each AS's features in a database for quick access. Since these AS features are relatively stable, the module just needs to update the database periodically. After receiving AS triplets from the preprocessor module, it generates the corresponding AS triplet features and delivers them to leak locator.

Route Leak Locator. The module takes AS triplets with corresponding triplet features as inputs and identifies AS triplets with route leak. It leverages a machine learning model to classify legitimate and leak AS triplets. In our implementation, we apply Random Forests [40] as our classifier, since it is one of the most effective machine learning models for classification and performs well in our experiments. Once the module finds an AS triplet with route leak, e.g., $\langle AS_1, AS_2, AS_3\rangle$, the module will send a real-time notification to RoLL subscribers. Hence, subscribers can know that AS_2 leaks the routes from AS_3 to AS_1. Therefore, subscribers can take actions to handle route leak events in time. For example, if subscribers are AS operators, they can configure routers to drop the BGP update messages containing the AS triplet with route leak. A real-time and accurate route leak locator can speed up operator's response to route leak incidents and reduce the impact they bring.

V. Evaluation

In this section, we conduct experiments to evaluate the performance of RoLL and compare it with prior methods.

A. Experiment setup

We implement RoLL with `Python` and `scikit-learn`. We implement Random Forests as the route leak locator to identify leak AS triplets. Note that other algorithms, such as RNN and SVM, can achieve similar performance. Following the source code of the prior work *ISP-Self Operated* [16, 41], we use 80% data from our dataset mentioned in Section III to train our model and 20% data to test our model. We balance the training dataset via under-sampling since there are more legitimate AS triplets than the leak ones. During the training phase, we use a grid search to find the best hyperparameters of the random forest model. For detailed hyperparameters, please refer to our public source code [42]. We implement *AS-Rank, Problink, TopoScope, ISP Self-Operated and MSLSTM* using their source codes [41, 43–45]. We apply them to our collected dataset for comparisons with RoLL. For *CAIDA AS Relationship*, we use its public results [30]. All our experiments are conducted on Dell PowerEdge R740 Rack Server with Intel(R) Xeon(R) Gold 6230R CPU @ 2.10 GHz and 128 GB RAM.

B. Experimental Results

Location Performance. Table II shows the route leak location performance of different methods. As the methods based on business relationship are highly affected by AS link invisibility [25], they cannot always give the business relationships of two ASes in AS triplets. Therefore, whether there are route leak incidents for a proportion of AS triplets are unknown. Our results show that at least 6% AS triplets cannot be judged by them. Even though we show their location performance without unknown business relationships in Table II, they only achieve about 80% location accuracy, 65% recall rate, and less than 75% F1-Score.

The two prior methods based on machine learning can achieve acceptable route leak detection performance. In particular, ISP Self-Operated [16] can achieve 90% F1-Score and

TABLE II
ROUTE LEAK LOCATION PERFORMANCE OF DIFFERENT METHODS

Category	Method	FPR	Recall	Precision	Accuracy	F1-score	Unknown
Based on Business Relationship	CAIDA AS Relationship [30] ¶	0.09	0.65	0.87	0.78	0.75	6%
	AS-Rank [24] ¶	0.10	0.67	0.86	0.79	0.75	19%
	Problink [14] ¶	0.09	0.65	0.87	0.79	0.75	20%
	TopoScope [25] ¶	0.10	0.64	0.85	0.77	0.73	10%
Based on Machine Learning	ISP Self-Operated [16] †	**0.04**	0.90	0.89	0.90	0.90	0
	MSLSTM [17, 19] †	0.34	**0.94**	0.73	0.80	0.82	0
	RoLL (Ours)	0.11	0.92	**0.90**	**0.91**	**0.91**	0

¶ The methods based on business relationships cannot always give the business relationship of two ASes in an AS triplet. Hence, whether there are route leak incidents for a proportion of AS triplets is unknown.
† **For the two methods, the table shows their route leak detection performance since they cannot locate route leak.**

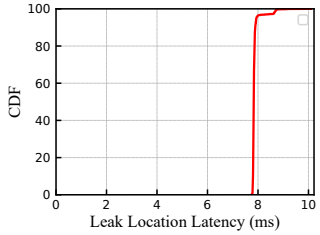


Fig. 8. Route Leak Location Latency of RoLL.

MSLSTM [17, 19] can achieve 94% recall rate. However, they fail to locate route leak since they use coarse-grained statistical features. They just detect whether there is route leak. Compared to the prior methods, RoLL can enforce accurate route leak location based on AS triplet features and machine learning. It achieves 90% precision, 91% accuracy, and 91% F1-score, which are all the best results.

Location Latency. As RoLL analyzes route leak directly from AS triplets of BGP update messages, it thus can achieve real-time route leak location. Specifically, we define the location latency as the interval between the time when RoLL receives a BGP update message and the time when it locates route leak in AS triplets from the message. As Fig. 8 shows, more than 90% AS triplets with route leak can be located in less than 8 ms. Moreover, all the route leak events can be located within 10 ms.

TABLE III
LATENCY OF DIFFERENT METHODS

Category	Method	Average Latency
Based on Business Relationship	Caida AS Relationship [30]	0.21 ms
	AS-Rank [24]	0.18 ms
	Problink [14]	0.19 ms
	TopoScope [25]	0.24 ms
Based on Machine Learning	ISP Self-Operated [16]	1 min
	MSLSTM [17, 19]	8 min
	RoLL (Ours)	7.86 ms

We also compare the latency of RoLL with prior methods, which is shown in Table III. The methods based on business relationship achieve the fastest route leak location due to the pre-inferred business relationship. However, the two prior methods based on machine learning require at least 1 min. It is because they accurately detect route leak based on statistical features extracted from massive BGP update messages in a long time interval. The time interval of ISP Self-Operated and MSLSTM is 1 min [16] and 8 min [17, 19], respectively. However, RoLL only requires about 8 ms to locate route leak while achieves high location accuracy.

VI. RELATED WORK

Route Leak Location Based on Business Relationship. As the business relationship between ASes is confidential, a number of studies [14, 24, 25] infer AS business relationship according to heuristic rules on the Internet's inter-domain structure and probabilistic models. One of their important applications is to locate route leak based on AS business relationship in real time. Although the state-of-the-art inferring algorithms can achieve approximately 90% accuracy for one AS link, they fail to achieve accurate route leak location. As it requires knowing the business relationships of two AS links at the same time, the final location accuracy will drop substantially due to multiplication rule of probability. Furthermore, since inferring business relationships is highly affected by AS link invisibility [25], it cannot always give the business relationship of two ASes. Therefore, these methods may fail to locate route leak in some cases.

Route Leak Detection Based on Machine Learning. Recent studies [16–18] apply machine learning to detect route leak. They extract numerous statistical features from massive BGP update messages in a long time interval. These statistical features include the number of update messages, the number of new peers, the number of new prefixes announced by ASes per unit time, etc. Although they can achieve high accuracy of route leak detection, they fail to locate AS triplets with route leak. Moreover, collecting and extracting features from massive BGP messages in a long time interval causes high detection latency. Different from them, RoLL leverages fine-grained AS triplet features to accurately locate leak AS triplets from a BGP update message in real time.

Route Leak Prevention. Researchers [46–48] have presented several countermeasures to effectively prevent route leak in advance. However, compared to route leak detection or location methods, they are much more difficult to implement in practice due to the lack of incentive and cooperation of ASes [46].

Peerlock [46] requires neighboring ASes to cooperate with each other, and may cause the disclosure of confidential business relationships. ASPA [47] relies on RPKI that is not yet widely deployed [49]. Besides, recent studies show RPKI faces various security threats [50, 51]. Internet Routing Registries [48] (IRRs) are databases where AS operators record their routing policy information. Hence, AS operators can get the information from IRRs to configure filters to prevent route leak. However, data from IRRs are inaccurate and outdated since AS operators have less incentive to timely update its accurate policy information in IRRs.

VII. Conclusion

In this paper, we analyze and identify AS triplet features that can effectively distinguish legitimate and leak AS triplets. Based on the triplet features, we design a system named RoLL that can accurately locate route leak from a BGP update message in real time. We implement the prototype of RoLL and evaluate it with real BGP data. Comprehensive experiments demonstrate that it can achieve high route leak location accuracy with low latency.

Acknowledgment

The research is supported in part by the National Natural Science Foundation of China (NSFC) under Grant 62221003, 61832013, and 62202260; and in part by the China Postdoctoral Science Foundation under Grant 2022M721824; and in part by the Shuimu Tsinghua Scholar Program. Mingwei Xu and Jiahao Cao are the corresponding authors of the paper.

References

[1] "Youtube hijacking." https://www.ripe.net/publications/news/industry-developments/youtube-hijacking-a-ripe-ncc-ris-case-study.
[2] "Cloudflare," https://blog.cloudflare.com/the-deep-dive-into-how-verizon-and-a-bgp-optimizer-knocked-large-parts-of-the-internet-offline-monday/.
[3] "Cloudflare2022," https://blog.cloudflare.com/route-leaks-and-confirmation-biases/.
[4] "Google," https://arstechnica.com/information-technology/2018/11/major-bgp-mishap-takes-down-google-as-traffic-improperly-travels-to-china/.
[5] "Vodafone leak," https://www.bleepingcomputer.com/news/security/major-bgp-leak-disrupts-thousands-of-networks-globally/.
[6] Sriram, Kotikalapudi et al., "Problem Definition and Classification of BGP Route Leaks," RFC 7908, Jun. 2016. [Online]. Available: https://www.rfc-editor.org/info/rfc7908
[7] S. Goldberg, "Why is it taking so long to secure internet routing?" *Communications of the ACM*, vol. 57, no. 10, pp. 56–63, 2014.
[8] "Manr2021report," https://www.manrs.org/2022/02/bgp-security-in-2021/.
[9] Li, Song et al., "Route leaks identification by detecting routing loops," in *EAI SecureComm*. Springer, 2015, pp. 313–329.
[10] Su, Shen et al., "Towards real-time route leak events detection," in *IEEE ICC*. IEEE, 2015, pp. 7192–7197.
[11] Siddiqui, Muhammad Shuaib er al., "Route leak detection using real-time analytics on local bgp information," in *IEEE GLOBECOM*, 2014, pp. 1942–1948.
[12] Bagnulo, Marcelo et al., "Practicable route leak detection and protection with asiria," *Computer Networks*, vol. 211, p. 108966, 2022.
[13] L. Gao, "On inferring autonomous system relationships in the internet," *IEEE/ACM ToN*, vol. 9, no. 6, pp. 733–745, 2001.
[14] Jin, Yuchen et al., "Stable and practical as relationship inference with problink," in *Proc. USENIX NSDI*, 2019, pp. 581–598.
[15] Arnold, Todd et al., "Cloud provider connectivity in the flat internet," in *Proc. ACM IMC*, 2020, pp. 230–246.
[16] Dong, Yutao et al., "Isp self-operated bgp anomaly detection based on weakly supervised learning," in *Proc. IEEE ICNP*, 2021, pp. 1–11.
[17] Cheng, Min et al., "Multi-scale lstm model for bgp anomaly classification," *IEEE TSC*, vol. 14, no. 3, pp. 765–778, 2018.
[18] Abd El Monem, Salma et al., "Bgp route leaks detection using supervised machine learning technique," in *2020 2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES)*. IEEE, 2020, pp. 15–20.
[19] Cheng, Min et al., "Ms-lstm: A multi-scale lstm model for bgp anomaly detection," in *Proc. IEEE ICNP*, 2016, pp. 1–6.
[20] "Caida." https://www.caida.org/.
[21] "Peeringdb." https://www.peeringdb.com/.
[22] "The route views project." http://www.routeviews.org/routeviews/.
[23] "Ripe ris (routing information service)," https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris/ris-raw-data.
[24] Luckie, Matthew et al., "As relationships, customer cones, and validation," in *Proc. ACM IMC*, 2013, pp. 243–256.
[25] Jin, Zitong et al., "Toposcope: recover as relationships from fragmentary observations," in *Proc. ACM IMC*, 2020, pp. 266–280.
[26] Rekhter, Yakov et al., "A Border Gateway Protocol 4 (BGP-4)," RFC 4271, Jan. 2006. [Online]. Available: https://www.rfc-editor.org/info/rfc4271
[27] Caesar, Matthew et al., "Bgp routing policies in isp networks," *IEEE network*, vol. 19, no. 6, pp. 5–11, 2005.
[28] "Cisco bgpstream," https://bgpstream.crosswork.cisco.com/.
[29] Wikipedia contributors, "Tier 1 network — Wikipedia," 2022, [Online; accessed 23-October-2022]. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Tier_1_network&oldid=1117011984
[30] "Caida as relationship." https://www.caida.org/catalog/datasets/as-relationships/.
[31] "Caida routeviews-prefix2as." https://www.caida.org/catalog/datasets/routeviews-prefix2as/.
[32] "Caida as organizations." https://www.caida.org/catalog/datasets/as-organizations/.
[33] "Caida ixps." https://www.caida.org/catalog/datasets/ixps/.
[34] "Peeringdb route server." https://www.peeringdb.com/search?q=route\%20server.
[35] "Caida as classification." https://www.caida.org/catalog/datasets/as-classification/.
[36] Systems Storigen et al., "Known Content Network (CN) Request-Routing Mechanisms," RFC 3568, Jul. 2003. [Online]. Available: https://www.rfc-editor.org/info/rfc3568
[37] "Ris live." https://ris-live.ripe.net/.
[38] "As-path-prepending." https://www.noction.com/blog/as-path-and-as-path-prepending.
[39] "As-numbers." https://www.iana.org/assignments/as-numbers/as-numbers.xhtml.
[40] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.
[41] "Isp-self operated." https://github.com/universetao/A-General-Framework-BGP-Anomaly-Detection.git.
[42] "Roll." https://github.com/yangtzeriverli/RoLL.git.
[43] "Problink." https://github.com/YuchenJin/ProbLink.git.
[44] "Toposcope." https://github.com/Zitong-Jin/TopoScope.git.
[45] "Mslstm." https://github.com/jayvischeng/MSLSTM.git.
[46] "Ntt peer locking," http://instituut.net/~job/peerlock_manual.pdf.
[47] Azimov, Alexander et al., "Verification of as path using the resource certificate public key infrastructure and autonomous system provider authorization. ietf, 2018," 2021.
[48] "Internet routing registry (irr)." http://www.irr.net/.
[49] "Rpki-monitor," https://rpki-monitor.antd.nist.gov/.
[50] van Hove, Koen et al., "Rpkiller: Threat analysis from an rpki relying party perspective," *arXiv preprint arXiv:2203.00993*, 2022.
[51] Hlavacek, Tomas et al., "Stalloris:{RPKI} downgrade attack," in *USENIX Security*, 2022, pp. 4455–4471.