A Cooperative Method for Prefix Hijack Detection in the Internet

Xin Liu, Peidong Zhu, Yuxing Peng, Ning Hu
School of Computer, National University of Defense Technology, Changsha 410073, China
xin.liu@nudt.edu.cn, pdzhu@nudt.edu.cn, yxpeng@nudt.edu.cn, ning_hu@163.com

***Abstract*—The problem of detecting prefix hijacks in the Internet remains a challenging problem, when considering no single completely accurate source of truth about which organizations have the authority to advertise which prefixes. This paper proposes a method, called Co-Monitor, for prefix hijack detection based on cooperation. In the Co-Monitor overlay network, participating ASes exchange prefix-to-origin mappings defined by them. Through cooperative monitoring, such an event that the prefix origin of a route is inconsistent with the requested mapping can be detected in a comprehensive monitoring scope. To ensure the accuracy, the method adopts source verifying mechanism to confirm prefix hijacks because an announcer answers for its prefixes. We conduct experiments to evaluate the capabilities of the cooperative monitoring method, and results show that ASes have much incentive to join the Co-Monitor architecture.**

Keywords-BGP; Prefix Hijacking; Cooperative Monitoring; Source Verifying; Monitoring Scope

I. INTRODUCTION

The Internet consists of a large number of interconnected autonomous systems (ASes), which exchange their routes using the Border Gateway Protocol (BGP). The current version, BGP-4, has been used for some years now and has managed the Internet routing for over a decade. Even after such success, several problems remain open to BGP [1]. Among these problems is prefix hijack detection, i.e., how to determine whether a prefix is hijacked by others or not.

Although much attention has been given to the problem of prefix hijack detection in the Internet, no universally acceptable solution has been developed. There are several reasons. On one hand, proactive detecting solutions have been designed to solve the problem in a cleanest way (and hence higher accuracy), but they are impractical due to requiring changes in the routing protocol. On the other hand, anomaly-detection solutions often suffer from high false positives, although they can be deployed incrementally. Moreover, an AS's detecting capabilities are characterized by localization, which blinds it to prefix hijacking.

To address these challenges, we propose a cooperative and deployable method for prefix hijack detection in real-time, called Co-Monitor, in which the complexity and difficulty in determining the ownership of prefixes are completely removed from the receiving AS of a route. The basic principle is cooperative monitoring: ASes look out for one another. Each participating AS defines a mapping of prefixes to ASes (generally contains itself and its prefixes), and then exchanges mappings with other participants. Through cooperative monitoring technique, an inconsistency of a route's prefix origin with the exchanged mappings can be detected. To ensure accuracy, the method adopts source verifying technique to confirm prefix hijacks.

Our contributions are three-fold. First, we propose a novel method for prefix hijack detection that is very promising and allows for many future extensions. Second, a model is proposed to evaluate the monitoring capability of an AS. Third, we evaluate the monitoring scope of an AS, and the result shows that only 3.6% ASes that join the Co-Monitor can monitor the 50% Internet.

This paper is organized as follows: Section 2 is an overview of the state of art. Our proposed method is detailed in Section 3. The monitoring capability that the method provides for ASes is evaluated in Section 4. We discuss remaining issues and future work in Section 5. Finally, Section 6 concludes the paper.

II. RELATED WORK

A lot of work has focused on the problem of prefix hijacking. We present the prior work in two major categories: crypto and crypto-free solutions. The crypto-based schemes [2-4], such as S-BGP, soBGP and psBGP, require BGP routers to sign and verify the origin AS of a route to prevent prefix hijacking, which have significant impact on router performance. The crypto-free schemes, such as [5-8], require changing router software so that inter-AS queries are supported, or additional attributes are added into BGP updates to facilitate detection. All the above schemes are not easily deployable because they require changes to router software, router configuration, or network operations etc.

More recently, people begin to focus on data plane. The Listen approach [9] determines whether a prefix is hijacked by checking whether it has any complete TCP sessions. A distributed scheme [10] detects hijacking only using data

plane information. Since the scheme doesn't rely on any BGP feeds (and hence infers prefix hijacking), it suffers from higher false positives compared with our method.

The RIPE MyASN [11] and PHAS [12] are similar to our method, but they need a center server for monitoring the events of BGP origin changing. Unlike them, our method distributes the monitoring service on the whole Co-Monitor overlay network, which endows participants with stronger detecting capabilities, such as larger monitoring scope. In practice, BGP Looking Glasses [13] are the most common tools to diagnose the Internet routing, but with this manual "pull-based" mode network operators are labor intensive and react slowly to hijacks about their prefixes. The push mode that our method proposes reduces human effort, leverages growing observation points, and discovers hijacks quickly.

III. CO-MONITOR

In this section, we present a detailed description of our prefix hijack detection method. First, we describe two techniques that are used in the method: source verifying and cooperative monitoring. And then, we naturally present the Co-Monitor.

A. Source Verifying

The main difficulty when detecting prefix hijacks in the real world is indistinguishable hijacking from legitimate routing changes, and the key of detection solutions depends on the trusty ownership of prefixes. However, it would appear to be extremely difficult to obtain the exact ownership of all prefixes given the complexity, if not impossibility, of tracing how existing IP address is allocated, delegated, and tracing any change of their ownership [14].

Instead, our method is based on the technique of source verifying to judge whether a prefix is hijacked or not. It is motivated by two observations in operational environment: 1) a legal announcer (origin AS) of a prefix can accurately distinguish between legitimate changes of routes and events of prefix hijacking; and 2) the legal announcer answers for the prefix and will take necessary actions to solve the hijacking problem if its prefix is hijacked. The source-verification technique for prefix hijack detection completely removed the complexity and difficulty associated with the ownership of prefixes from the receiving AS. Take note of that, although we can't help ISPs to obtain a centralized, trusty ownership of all prefixes, the source-verifying technique emphasizes and introduces that each AS takes charge of its prefixes, which can imaginably form a distributed, trusty ownership of the prefixes concerned. Just as a self-organization system, it distributes the monitoring responsibility of all prefixes among the all participants: no single AS is "in charge" of the overall prefixes, but each contributes to a collective security [15].

Figure 1 gives a graph of AS topology and illustrates the idea behind our source verifying technique. AS A originates route (P, A) to AS E, and AS B originates route (P, B) to AS D. Suppose that AS A is the legal announcer, and AS B is an attacker who hijacked the prefix P. Now the AS C receives the hijacked route r_1 from AS F and the right route r_2 from AS G, but it doesn't know which AS (A or B) have the authority to advertise the prefix P. How can the hijacking event be detected? The answer is, if AS C receives a route with a suspicious origin, it will deliver notification to interested ASes. For example, AS C delivers the event of origin change, that is (P, A→B), to AS A. And then, the legal AS can inspect the validity of origin change of the related prefix.

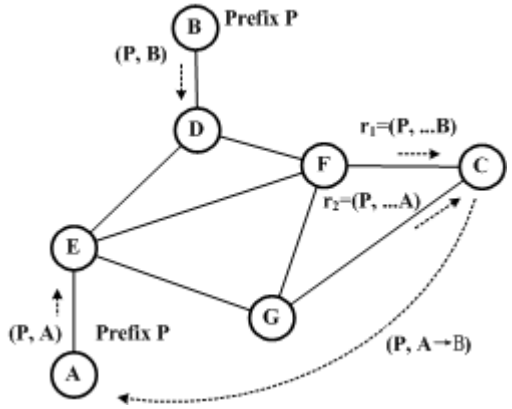

Figure 1. Source Verifying Technique

It is worth noting that AS A needs negotiate with AS B or B's upstream provider to resolve this hijacking problem. Therefore, the source-verifying technique isn't a proactive scheme, but a reactive scheme. Moreover, an entity in a selfish environment unlikely helps others if there is a lack of obvious benefits. That is, AS C may have not the incentive to notify AS A. We need to deal with this case carefully.

B. Cooperative Monitoring

Our method makes use of the source-verifying technique to detect prefix hijacking. But challenges are still in existence. As we have mentioned before, a legal announcer answers for its prefixes in practice. Though most proposed schemes neglect the fact, especially proactive solutions, the idea of source verifying is actually not new, because it is the exact way that ISPs manually do in operation. Why is the way ineffective on detecting prefix hijacks in practice?

In nature, the problem is not in the source-verifying technique itself, but in the autonomous Internet. First of all, the global routing infrastructure is not a fully automated system. It depends on the constant efforts of thousands of network operators and engineers around the world every day. So, it is slow and labor intensive to resolve a prefix hijacking, requiring highly experienced engineers. Next, existing monitoring solutions are limited to routers of an organization. They can't resolve the problems that originate beyond the network's administrative boundary, and the situation gets worse if a problem originates further beyond the next-hop peer or provider networks. As we can see, the problem of prefix hijacking is just so. More unfortunately, a hijacked AS usually can't obtain sufficient help because of

"selfishness" of entities in the self-organization Internet. As a result of these limits, an AS is devoid of the capabilities to monitor their prefixes by themselves.

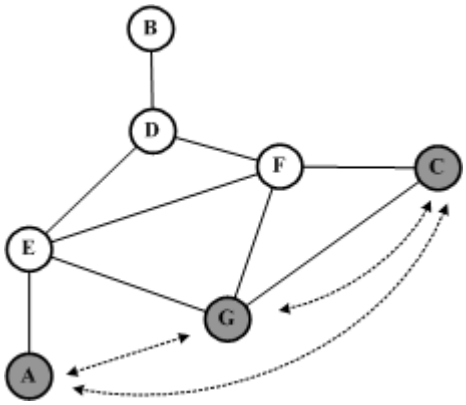

Figure 2. Cooperative Monitoring Technique

Obviously, the problem of prefix hijacking detection has to be tackled in a novel way. Unlike any existing solutions, our method is based on the idea of cooperative monitoring, namely, ASes look out for one another. Figure 2 illustrates the simple, but powerful idea. The three ASes, A, C and G, are coordinated to monitoring respective prefixes with the help of the other ASes. If AS D (or AS C) detects a disagreement according to the requesting of AS A, it will notify AS A of this event and vice versa. Like the philosophy of the Peer-to-Peer (P2P) computing, we regard the monitoring capability of an AS as a kind of resource and call for coordinating so that ASes can quickly and accurately detect prefix hijacking.

C. Co-Monitor Method

With above techniques explained, we can naturally present our method. Co-Monitor consists of four key steps. First, each participating AS constructs a *prefix-to-origin* table that is a mapping of prefixes to ASes. Generally, a participant's table contains the mapping of its prefixes to itself at least. Second, these ASes exchange respective mappings each other. Third, these ASes locally keep an eye on routes announced by neighboring ASes. Four, if a participating AS detects an event that the prefix origin of a route is inconsistent with the requested mappings (by request of others), a notification is delivered to the requesting AS(es) (Generally, it is the legal announcer of the prefix, and our method is not restricted to this). By this means, ASes can extend the capabilities of monitoring their prefixes from different vantages in the Internet, and can determine whether their prefixes are hijacked by others or not in real-time.

For joining the Co-Monitor architecture, each AS is required to set up a dedicated AS-level server, called *AS-Monitor*, to detect prefix hijacking cooperatively. The method is entirely embodied by *AS-Monitors* and the interaction between them, which form an overlay network at application level. Therefore, the method is not required to modify routing protocol and can be deployed incrementally. However, the method needs to modify the configurations of BGP border routers of each participating AS, because its *AS-Monitor* need watch BGP routes that neighboring AS advertises in real-time, which can achieve this purpose by peering with these routers silently.

The *prefix-to-origin* table that a participating AS defined, such as AS A, can be denoted as *A.prefix-to-origin*. According to the table of AS A, others, such as AS B, can generate notifications and send them to AS A. The procedure of generating notifications of AS B is shown in Algorithm 1. Notice that, a notification only includes its generator, the prefix of the route examined, the prefix's suspect origin and the route's time (without other information, such as the AS path of the route). Therefore, the Co-Monitor doesn't leak any private route information of a participating AS.

Algorithm 1: Notification Generating

```
1:  for each eBGP route r received by AS B do
2:    pick-up the origin of r,
3:      that is (r.prefix, r.origin, r.time)
4:    for each item λ of A.prefix-to-origin do
5:      if r.prefix = λ.prefix
6:        and r.origin ∉ λ.origins then
7:          send (B, r.prefix, r.origin, r.time)
8:            to the requesting AS, that is A
9:      end if
10:   end for
11: end for
```

The Co-Monitor overlay network that is provided by our method is a comprehensive monitoring network for its all participants. As soon as an *AS-Monitor*, such as AS A, receives the first notification of a prefix, it may continuously receives notifications delivered by others AS-Monitors. This feature of the Co-Monitor makes itself excel any proposed solutions. Because the Co-Monitor can help its participants to answer the more, important questions in real-time: not only whether their prefixes are hijacked or not, but also when and where hijackings occurs.

IV. EVALUATION

The objective of this section is to evaluate monitoring capabilities that our method provides to participants. We model monitoring capabilities of an AS at first, and then evaluate our method using public BGP data.

A. Model and Formulation

The monitoring capability of an AS is important to determine whether its prefix is hijacked or not. For simplicity, in this paper we only focus on the principal aspect of monitoring capabilities of ASes, that is, *monitoring scope*. The monitoring scope of an AS is the bound that it can monitor in the Internet. Since the Internet is autonomous, an AS can only monitor itself with ease. It is highly dis-advantageous for an AS to detect hijacking. However, our method can intuitively help its participants to detect hijacking accurately, mainly due to the comprehensive monitoring scope.

We model the monitoring scope of an AS as follows. Let *INT* be the set of all ASes in the Internet. For any AS in *INT*, e.g. *A*, let its neighbors be S_A. Recall that an AS in operation can easily monitor the routes advertised by its neighbors. Therefore, we evaluate the monitoring scope of AS A, denoted V_A, through the rate of the number of neighbors of AS A to all ASes.

$$V_A = \frac{|S_A|}{\sum_{\alpha \in INT} |S_\alpha|}. \quad (1)$$

Based on the formula above, V_A can take values in [0,1] where 1 means that AS A can monitor the whole Internet, and 0 means that AS A can't monitor the Internet any more (disconnected from the Internet). And then, we assume that a set of ASes, denoted *CMM*, joins the Co-Monitor. Accordingly, the monitoring scope of AS A is given by:

$$V_A = \begin{cases} \dfrac{\sum_{\beta \in CMM} |S_\beta|}{\sum_{\alpha \in INT} |S_\alpha|}, & A \in CMM \\ \dfrac{|S_A|}{\sum_{\alpha \in INT} |S_\alpha|}, & A \notin CMM \end{cases}. \quad (2)$$

Obviously, the monitoring scope of AS A is extended to all partners when it participates in the Co-Monitor architecture, and the more partners the larger monitoring scope participant can obtain. In contrast, an AS that is not in *CMM* will not benefit from it, and the monitoring scope of non-members is identical to Equation 1.

B. Experimental Results

To demonstrate the benefit of the Co-Monitor in the context of the Internet, we select a BGP snapshot from RouteViews on June 20, 2007 [16]. The Internet topology consists of 25699 ASes. We sort these ASes according to their node degrees, and assign an ID (from 1 to 25699) to every AS. Figure 3 show the monitoring scope of every AS without Co-Monitor, mostly less than 10^{-4} (close to zero).

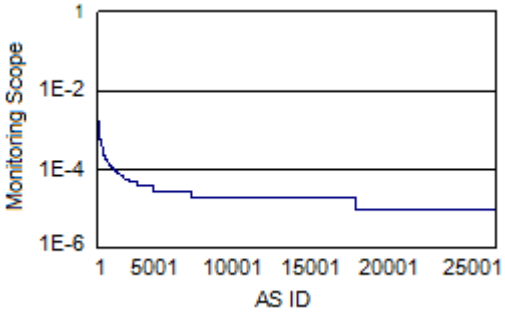

Figure 3. Monitoring scope without Co-Monitor

We assume that ASes join the Co-Monitor architecture in turn according to their IDs. The experimental results are depicted in Figure 4. The results demonstrate clearly that the Co-Monitor makes the monitoring scope of an AS increase rapidly. For example, if the top 10 ASes joined, their monitoring scope is the 12.9% Internet. More importantly, because of the power-law property of the Internet, the top 916 ASes (less than 3.6% of 25699) can monitor the 50% range of the Internet.

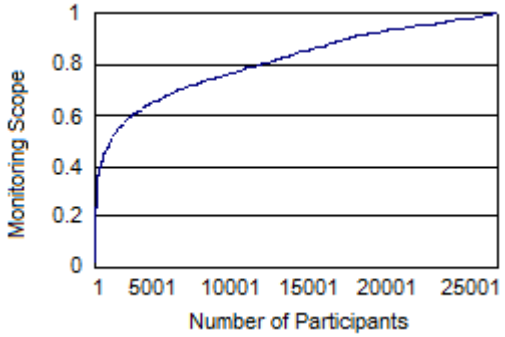

Figure 4. Monitoring scope with Co-Monitor

Evidently, the benefit to participants is larger than to non-participants. Therefore, an AS should have much incentive to join the Co-Monitor architecture.

V. Discuss and Future Work

A. Notification Delivery

As long as the prefixes that *AS-Monitors* of participants reside in, called *key prefixes*, are not hijacked, our method can be used to easily detect prefix hijacking. Nevertheless, the method will become quite complex if only we take the case into account. The major challenge in the Co-Monitor is how to deliver notifications successfully when it is in face of the hijacking of *key prefixes*. For simplicity, we don't consider this problem in this paper, and it is a flaw.

In fact, it depends on the design goal of the Co-Monitor. On one hand, due to the large scale of Internet routing, a hijacking of a *key prefix* is unlikely to affect all paths of other *AS-Monitors* to the *AS-Monitor* (its IP resides in the hijacked *key prefix*). Thus, a light-weight method requires that an *AS-Monitor* analyzes notifications that it has received to detect prefix hijacking. On the other hand, we must consider the problem of delivery seriously if the design requires that the method should try its best to deliver notifications successfully. Undoubtedly, an *AS-Monitor* is easy to confirm prefix hijacking, but the method is involved in the complex overlay routing among *AS-Monitors*. As part of our future work, we will continue to research this problem.

B. Security Consideration

Any system is confronted with security problems and so is our method. The bad news is that like BGP our method has no mechanism for authenticating notifications.

Therefore, some old problems of BGP still exist in our method. For example, how does an *AS-Monitor* trust other *AS-Monitors*? How can a malicious *AS-Monitor* be prevented from collaborating with a hijacker or flooding the overlay network with false notifications? Maybe we should control the admission of an AS to the Co-Monitor, and introduce some reputation mechanism to punish misbehaved participants.

The good news is two-fold. First of all, the security objective of the Co-Monitor is that an *AS-Monitor* can receive notifications when its prefix is hijacked, which is different from BGP. As we have discussed above, it is impossible for an attacker to intercept all notifications of a *key prefix hijacking*. And then, an attacker can only spread false notifications, which still can't prevent our method from its security objective. Secondly, our method is an application-level scheme. Many crypto mechanisms, such PGP, can be adopted without modifying BGP.

C. Other hijacks

There are two representative problems in BGP. One is the problem of unauthorized advertisement of prefixes; the other is the problem of illegal AS path. Though they are all considered to be hijacking in some papers, we distinguish them, and only regard the first problem as prefix hijacks. For example, an attacker can advertise a route with legitimate origin but invalid path to the origin (false last hop), and this hijacking (and any bogus AS path) can evade any form of inter-domain routing authentication if not examine the AS path of the route. Our method is also difficult to solve the problem of illegal AS path.

Furthermore, prefix hijackings include [12]: *exact-prefix* hijacking (the hijacked prefix is an exact-prefix of some valid prefix), *sub-prefix hijacking* (the hijacked prefix is a sub-prefix of some valid prefix) and *super-prefix hijacking* (the hijacked prefix is a super-prefix of some valid prefix). Although we have focused on detecting *exact-prefix hijacking* of a prefix so far, it is worth noting that our method can be easily extended to detect the others. That is, participants exchange mappings of *IP address blocks to ASes*. In *IP address blocks* of the mapping, the participant defines all possible prefixes concerned, which include not only announced but also unannounced prefixes. We will explore the feasibility of using our method to resolve the second problem as part of our future work.

VI. CONCLUSION

In this paper, we propose a cooperative method for detecting prefix hijacks. Different from most, if not all, other previous work on this topic, the method introduces each AS take charge of its prefixes, which can imaginably form a distributed, trusty ownership of the prefixes concerned.

The Co-Monitor has several advantages that the previous any solution doesn't hold collectively: 1) it is cooperative, detecting with a comprehensive monitoring scope; 2) it is practically accurate in hijack detection by source verifying; 3) it can detect prefix hijacking in real-time; and 4) it doesn't require any changes to existing routing protocols, and hence can be deployed incrementally.

ACKNOWLEDGMENT

The authors would like to thank the generous comments from the anonymous reviewers. Their comments have greatly helped improve this research and prepare the camera-ready version of this paper. This research is being funded by the National High-Tech Research and Development Plan of China under Grant No. 2006AA01Z213, the National Natural Science Foundation of China under Grant No. 60673169 and No. 60433040, and the Research Foundation for Ph.D. Candidates of National University of Defense Technology of China.

REFERENCES

[1] M. Yannuzzi, X. Masip-Bruin, and O. Bonaventure, "Open Issues in Interdomain Routing: A Survey," IEEE NETWORK, vol. 19, pp. 49-56, 2005.

[2] S. Kent, C. Lynn, and K. Seo, "Secure Border Gateway Protocol (S-BGP)," IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS Special Issue on Network Security, vol. 18, pp. 582-592, 2000.

[3] R. White, "Securing BGP Through Secure Origin BGP," Internet Protocol Journal, vol. 6, pp. 15-22, 2003.

[4] T. Wan, E. Kranakis, and P. v. Oorschot, "Pretty Secure BGP (psBGP)," in ISOC. San Diego, CA, USA, 2005.

[5] 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 Proceedings of the International Conference on Dependable Systems and Networks (DSN), 2002.

[6] G. Goodell, W. Aiello, T. Griffin, J. Ioannidis, and P. McDaniel,, "Working Around BGP: An Incremental Approach to Improving Security and Accuracy of Interdomain Routing," in Symposium on Network and Distributed Systems Security, 2003, pp. 75-85.

[7] J. Karlin, S. Forrest, and J. Rexford, "Pretty good bgp: Protecting bgp by cautiously selecting routes," University of New Mexico, 2006.

[8] S. Y. Qiu, F. Monrose, A. Terzis, and P. D. McDaniel, "Efficient Techniques for Detecting False Origin Advertisements in Inter-domain Routing," in IEEE NPsec, 2006.

[9] L. Subramanian, "Listen and whisper: Security mechanisms for BGP," in First Symposium on Networked Systems Design and Implementation (NSDI'04), 2004.

[10] C. Zheng, L. Ji, D. Pei, J. Wang, and P. Francis, "A Light-Wight Distributed Scheme for Detecting IP Prefix Hijacks in Real-Time," in SIGCOMM'07. Kyoto, Japan, 2007.

[11] "Ripe myasn system," http://www.ris.ripe.net/myasn.html.

[12] M. Lad, D. Massey, and D. Pei, "PHAS: A Prefix Hijack Alert System," in Proceedings of 15th USENIX Security Symposium, 2006, pp. 153-166.

[13] "BGP looking glasses," http://www.bgp4.as/looking-glasses.

[14] W. Aiello, J. Ioannidis, and P. McDaniel, "Origin Authentication in Interdomain Routing," in Proceedings of 10th ACM Conference on Computer and Communications Security, Washington, DC., 2003.

[15] C. Prehofer and C. Bettstetter, "Self-Organization in Communication Networks: Principles and Design Paradigms," IEEE Communications Magazine, pp. 78-85, 2005.

[16] "Route Views Project," http://www.routeviews.org/.