Detection of Invalid Routing Announcement in the Internet [*]

Xiaoliang Zhao, Dan Pei, Lan Wang, Dan Massey, Allison Mankin, S. Felix Wu,Lixia Zhang[†]

Abstract

Network measurement has shown that a specific IP address prefix may be announced by more than one autonomous system (AS), a phenomenon commonly referred to as Multiple Origin AS, or MOAS. MOAS can be due to either operational need to support multi-homing, or false route announcements due to configuration or implementation errors, or even by intentional attacks. Packets following such bogus routes will be either dropped or, in the case of an intentional attack, delivered to a machine of the attacker's choosing.

This paper presents a protocol enhancement to BGP which enables BGP to detect bogus route announcements from false origins. Rather than imposing cryptography-based authentication and encryption to secure routing message exchanges, our solution makes use of the rich connectivity among ASes that exists in the Internet. Simulation results show that this simple solution can effectively detect false routing announcements even in the presence of multiple compromised routers, become more robust in larger topologies, and can substantially reduce the impact of false routing announcements even with a partial deployment.

1 Introduction

The Internet is made of thousands of *Autonomous Systems*, which are loosely defined as a connected set of IP prefixes under a single administration. An AS uses BGP to announce its IP address prefixes to BGP routers in neighboring ASes which in turn propagate the reachability to those prefixes to other ASes. An AS may announce a false route due to either faults or intentional attacks. False route announcements have been observed a number of times over the last few years. For example, in April 2001, an operational fault caused one AS to announce routes for 9177 IP prefixes that were not reachable from that AS. These false route announcements were propagated to other ASes and packets following these bogus routes were dropped. Large scale routing faults, such as the April 2001 event, are often detected quickly because they result in abnormally large volume of route changes and unreachability to numerous destinations, but a number of smaller scale routing faults also occur. While these faults are still damaging to the affected prefixes, the problem is much harder to detect and correct quickly.

With the rise of intentional attacks over recent years, it becomes more and more important to add to the network an ability to verify any route announcement before accepting it. However, the problem of detecting invalid routing announcements is a difficult one. The Internet is a large scale, loosely coupled system, without any centralized control or centralized database that provides the latest connectivity information; the only coordination among all the ASes is through running the same standard routing protocol. Although secure communication techniques, such as authentication and encryption, can be used to secure the routing information exchanges as suggested by some recent work[14], not only their wide deployment will take time but also such techniques alone will not assure the ultimate routing reliability due to potential possibilities of routers being compromised. Other approaches such as DNS based origin lookup have been proposed[3].

In this paper we present a simple protocol enhancement that enables BGP to distinguish false route announcements from valid ones. Our design utilizes the fact that today's Internet topology is a richly interconnected mesh, thus it is difficult, if not impossible, to completely block correct routing information from propagating out. We achieve the goal of false routing detection by detecting a conflict whenever a router receives both correct and false routing information. We verified our design through simulation. The results show that our simple solution can effectively detect false routing announcements even in cases of multiple routers being compromised; this detection ability becomes more robust in larger topologies.

For the 46-AS topology used in our simulation, without

[*] This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No DABT63-00-C-1027. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the DARPA.

[†] xzhao@unity.ncsu.edu, peidan@cs.ucla.edu, lanw@cs.ucla.edu, masseyd@isi.edu, mankin@isi.edu, wu@cs.ucdavis.edu, lixia@cs.ucla.edu

using our solution, when up to 4% of the AS's are injecting false routing data, more than 36% of the remaining AS's will adopt false routes. With our solution, on average only 0.15% of the AS's adopt false routes in the same simulation setting. Even when the number of attackers increases to 30% of the network, only about 9.8% of the remaining AS's adopt false routes, compared to 51% when without validation. Furthermore, our solution is incrementally deployable, and, when partially deployed, it can still substantially reduce the impact of false routing data.

The remainder of the paper is organized as follows. In the rest of Section 1 we define terminology through examples and illustrate how traffic hijacking can happen. Section 2 reviews the related work. Section 3 presents MOAS data from the Internet and discusses the potential causes for MOAS cases. Section 4 describes our approach to detecting invalid MOAS cases and Section 5 presents simulation results, followed by a summary in Section 6.

1.1 Definitions and Terminology

A BGP route includes a list of ASes, called an AS path, followed by a set of IP address prefixes reachable through that AS path.[1] The last AS in the list is commonly referred as the **origin AS**. For example, an AS path of $(10, 20, 30)$ associated with the IP prefix d indicates that AS 10 learned the path from AS 20, AS 20 learned the path from AS 30 and AS 30 originated the route to d.

Figure 1 shows an example of AS 4 originating a route to IP prefix 128.9/16. Prefix 128.9/16 is directly reachable by routers in AS 4 and AS 4 advertises a BGP route to its neighboring ASes. AS X learns two possible routes to prefix 128.9/16, path $(Y, 4)$ and path $(Z, 4)$. In general, an AS may see many different paths leading to prefix 128.9/16, with all of them originating from AS 4. A packet destined for 128.9.176.20 (www.isi.edu) follows the BGP route for prefix 128.9/16 until it reaches AS 4 and then AS 4's interior routing protocol delivers the packet to host 128.9.176.20.

If an IP address prefix appears to originate from more than one AS, we call this a **Multiple Origin Autonomous System (MOAS)** case, or MOAS. More precisely, if prefix d is associated with AS paths $asp_1 = (p_1, p_2, \dots p_n)$ and $asp_2 = (q_1, q_2, \dots q_m)$, then we say a MOAS occurs if $p_n \neq q_m$.

A MOAS can be either **valid** or **invalid**. A MOAS is valid if each originating AS can directly reach the prefix. For example, Figure 2 shows a valid MOAS. If any of the origin ASes cannot reach the prefix, then we say it is an invalid MOAS. Figure 3 shows an example of an invalid MOAS involving AS 4 and AS 52. Our objective is to detect invalid MOAS cases.

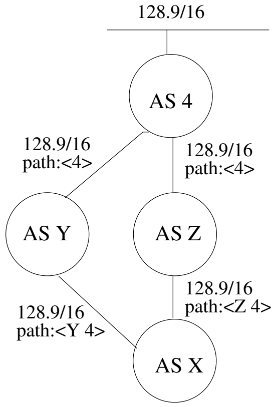


Figure 1. Originating a BGP Route

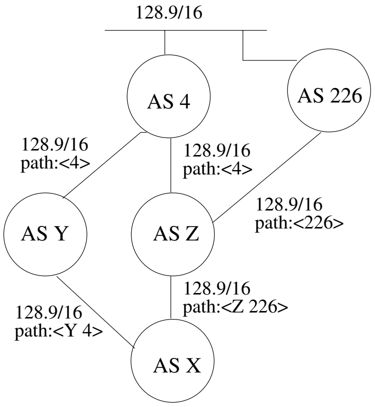


Figure 2. Prefix With Two Valid Origin ASes

The problem of detecting invalid origins is a complex one due to BGP operational practices. RFC 1930[11] recommends that each prefix originate from a single AS, but this is not a requirement. Legitimate operational needs, as discussed in Section 3, may result in a single prefix being announced by multiple Autonomous Systems. Figure 2 shows such an example, where the prefix 128.9/16 originates from both AS 4 and AS 226. In this case, AS X observes that 128.9/16 originates from both AS 4 and AS 226, though it has no way to tell whether this is the result of a legitimate operational need, or a routing fault of the type in the following example.

Figure 3 shows the effect of a fault or intentional attack at AS 52. AS 52 originates a route to prefix 128.9/16 even though AS 52 cannot directly reach this prefix. With the topology in Figure 3, AS 52 appears to AS X to offer the shortest route to prefix 128.9/16. With today's BGP implementation, AS X would accept and propagate this false route to its neighbors. Any packets destined for 128.9.176.20 that follow this faulty route would be for-

[1] In the case of route aggregation, an element in the AS path may include a set of ASes.

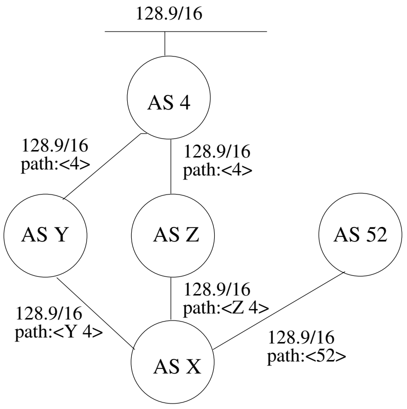


Figure 3. An Incorrect Origin AS

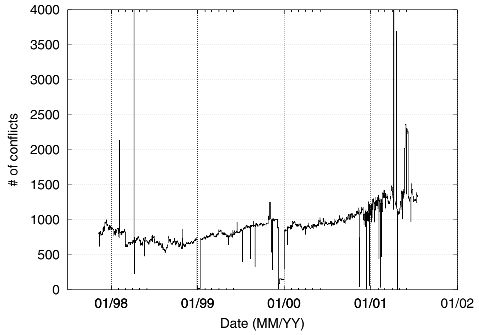


Figure 4. The number of MOAS cases from 11/1997 to 07/200

warded to AS 52 instead of reaching the intended destination.

2 Related work

BGP specification [11] recommends that a prefix be announced by a single AS. However, this recommendation is often not followed due to operational needs. Our earlier work[22] provided a detailed analysis of MOAS cases observed in the Internet, as well as explanations of both valid and invalid MOAS cases. Geoff Huston's BGP Table Statistics website [12] also provides a basic count on the number of MOAS cases observed in the Internet but without addressing the issue of whether they are valid or invalid, let alone the issue of how to distinguish the two cases.

As early as 1998 Bates et al [3] brought up the need for identifying the correct origin AS; the authors proposed to use DNS to store (prefix, origin AS) pairs in the originator's DNS. Each incoming route update could be checked against the DNS record to determine the correct origin AS. But given that DNS operations rely on the routing to function correctly, requiring BGP to interact with the DNS for correctness checking introduces a circular dependency. Furthermore, the DNS database can be incorrect or easily forged[1]. [21] proposes a filtering model that uses the Internet Route Registry (IRR) records to check the validity of route announcements. However, because keeping the IRR record updated is not a mandatory requirement for ISPs, some IRR records are outdated or inaccurate, reducing the effectiveness of this approach. [14] proposes to use some form of public key infrastructure (PKI) to verify the origin of the route advertisement. However, this approach calls for substantial modification to the current routing protocol implementations, such as changes needed to query the PKI.

[19] proposed to add a signed "predecessor" to protect the full AS path from being falsely modified. The predecessor identifies the AS directly following the origin. Path finding techniques such as those found in [8] can be used to authenticate the path. However, this approach cannot prevent an AS from falsely originating a route to a prefix that it cannot reach, such as the case we study in this work.

3 MOAS Cases in the Internet

Routing data collected from the Internet operations shows that MOAS is a common occurrence in today's Internet. This section presents a brief summary of the MOAS measurement, collected from the Oregon Route Views Server[18]; a more detailed analysis of Internet MOAS cases can be found in [22].

3.1 MOAS Measurement

Figure 4 shows the daily number of observed MOAS cases from 11/08/1997 to 07/18/2001. Over the 1279-day period 38225 MOAS cases were observed; the daily observed MOAS cases increased from a median value of 683 in 1998 to 1294 in 2001. The maximum reached 11842 on 04/07/1998 and 10226 on 04/06/2001.

Figure 5 shows the histogram of the duration time for all the observed MOAS cases. The duration of an individual MOAS case counts the total number of days when the routes to an address prefix were announced by more than one origin, regardless of whether the days were continuous and regardless of whether the same set of origins was involved. Figure 5 shows that, although some small number of MOAS cases are long lasting, most MOAS cases are short-lived. Those extremely short-lived MOAS cases, with a duration of one or two days, suggest an unintended behavior. In fact, 13730 (35.9%) out of the total 38225 MOAS cases lasted one day[2], and 82.7% of these short-lived MOAS cases can

[2] Because the Oregon Route Views Server, from which we collected

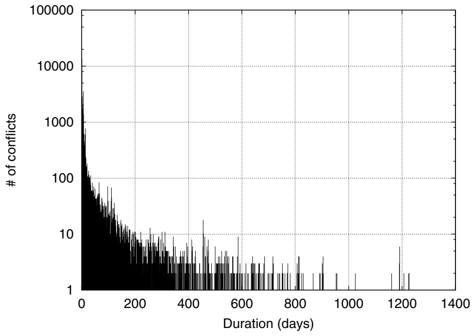


Figure 5. Duration of MOAS

be attributed to a configuration fault that occurred on April 7th, 1998.

Overall, these results indicate that MOAS does occur in the Internet today. Our further investigation shows that some of the MOAS cases, in particular the long lasting ones, are due to certain operational practice as explained below. However a large number of MOAS cases in a single day are most likely caused by faults or attacks; the few large spikes in Figure 4 match to the well known BGP route faults due to operational errors.

3.2 Valid MOAS Cases

Various forms of supporting multi-homing lead to MOAS. In multi-homing an organization is connected to more than one ISP and the organization's IP prefixes are advertised by these ISPs. In the simplest case, the multi-homed organization has its own AS number AS_N and connects to each of the ISPs via BGP peering. In this simple case, AS_N will appear to be the origin AS for the organization's IP prefix(es) and MOAS does not occur. But in other cases, the organization (denoted ORG) may connect to multiple ISPs via a number of different techniques and MOAS may occur.

- Due to convenience or other reasons, ORG may connect to ISP-1 via BGP peering and connect to ISP-2 via static configuration. This results in ISP-2 advertising ORG's address prefixes as if they were local to ISP-2. From a BGP perspective, it appears as if two origin ASes, ORG and ISP-2, can both directly reach the same set of prefixes and the result is MOAS. Figure 2 could be an example of such scenario, where the network 128.9/16 connects to AS 4 via BGP peering and connects to AS 226 via static configuration.
- AS numbers are currently a 2–byte identifier. To prevent AS number exhaustion, many organizations have not obtained a globally unique AS number, instead they peer with ISPs using a private AS number [9]. This technique is called AS number Substitution on Egress (ASE). When such an organization makes routing announcements to its multiple ISP connections, the ISPs strip off the private AS number before propagating the announcements further downstream, resulting in a MOAS where all the ISPs the organization connects to appear to announce the same set of address prefixes.

Because the links using non-BGP routing mechanisms or private AS numbers are ”hidden” to BGP, our measurement data often can not identify the cause of MOAS. However, by contacting individual ASes we did confirm valid MOAS exist and valid MOAS due to multi-homing tend to be long lasting in duration. Note that in these valid MOAS cases, packets may follow any of the routes announced by seemingly different origin ASes and will still reach the destination.

In addition, some IP prefixes are naturally associated with more than one AS. In particular, the prefixes associated with a BGP exchange point may be advertised by each AS connected to the exchange point. According to recommended operational practice, exchange point addresses should *not* be advertised into the global topology, although they might be announced to stub ASes for diagnostic uses. Our measurement data shows that MOAS exists for a number of exchange point prefixes, but the number of such MOAS cases make up only a very small percentage of the total MOAS cases we observed.

3.3 Invalid MOAS Cases

It has been observed multiple times that, due to operational errors, an AS incorrectly announced IP address prefixes that belong to other organizations. In most cases the faulty AS did not have a route to the incorrectly announced prefixes, thus IP packets that followed the incorrectly originated route arrived at the faulty AS and got dropped.

Figure 4 shows several notable spikes of MOAS cases caused by faults. A large MOAS spike occurred on April 7th, 1998. Discussions on the North American Network Operators mailing list [15] indicated that AS 8584 erroneously announced 11357 prefixes on that day that belonged to other organizations. Consequently, some routers selected the bogus routes in packet forwarding, causing noticeable disturbance to the Internet operation. Another example of MOAS caused by faults occurred on April 10th,

our data, takes only daily routing table dumps, it is impossible for us to distinguish a MOAS case that lasts for a short time period around the time when the routing table is dumped, from one that lasts longer than one day but not long enough to appear in two consecutive routing table dumps; both cases will be considered as a one-day MOAS in our measurement.

2001 and the sequence (AS 3561, AS 15412) was involved in 5532 out of 6627 MOAS cases that occurred during that day. Based on the archived data from RIPE RIS [17], AS 15412 normally originates only 5 address prefixes. However, on April 6th, 2001, AS 15412 suddenly originated thousands of prefixes due to a configuration error[7]. Yet another example, predating our measurement, occurred on April 25th, 1997 [2], when one ISP falsely de-aggregated its internal routing table and advertised the IP address prefixes it learned externally as its own prefixes [4], resulting in another MOAS spike.

These examples show that MOAS due to faults do occur and often have serious impacts on Internet data delivery. The false routing announcements that lead to MOAS can also be caused by intentional attack. We are yet to gain a full understanding of the causes of all the observed MOAS cases.

4 Detecting Invalid MOAS

Blind acceptance of MOAS that occurs in BGP announcements is dangerous because invalid MOAS cases could adversely affect packet forwarding. In this section we describe a simple mechanism that allows BGP routers to distinguish invalid MOAS cases from valid ones.

4.1 MOAS List

Our solution is to first create a list of the multiple ASes who are entitled to originate a particular IP address prefix p, and then attach this list to the route announcements by all those originating ASes. BGP routers that receive the route announcements from multiple origins can verify that the MOAS is intentional and valid. If another AS makes a faulty route announcement to prefix p, BGP routers which have received the right route to p can easily detected the fault since this faulty route's origin AS will not be in p's MOAS list.

For example, suppose multi-homing allows prefix p to be originated by both AS 1 and AS 2. A MOAS list will be attached to the routing announcements indicating that both AS 1 and AS 2 can serve as the origin AS for this prefix. A faulty AS, say AS 3, may also originate a route to prefix p, but AS 3 does not appear in the MOAS list advertised by AS 1 and AS 2. Although AS 3 could attach its own MOAS list that includes AS 1, AS 2, and AS 3, this list would not be in agreement with the MOAS list advertised by AS 1 and AS 2. Any router that sees both the faulty route and at least one of the valid routes can compare the MOAS lists and detect that there is potentially a problem.

However, if the origin AS(es) for p has only one path to reach the rest of the Internet, a fault or attack can defeat the MOAS detection mechanism by altering the origin AS or the origin AS list on this single path. But in this case, the attacker has compromised the only path to reach p and can cause other arbitrary damage to p as well. In more general cases, multiple origin ASes make route announcements for p and/or the origin AS(es) announces its route to multiple AS peers. As we demonstrate in our simulation, it is difficult for attackers to block or modify the origin AS list on all of these route advertisements, especially considering the increasing inter-connectivity in today's Internet topology [13].

The origin MOAS list does not use cryptographic authentication and may be removed or altered, either intentionally or unintentionally, as the route propagates through chains of ASes. Our technique relies on the distributed nature of the Internet topology for fault detection. While it may be possible to tamper with the routes to the prefix p along some of its propagation paths, trying to tamper with the routes to p along all the paths that route p announcement propagates would seem very difficult, if not impossible, in a large, well connected network topology. As long as the correct route p announcement can propagate out to a number of other ASes, it is likely that the conflict due to the tampering will be detected, thus protecting the routing system from blindly accepting bogus routes injected by potential attacks or fault.

4.2 Implementing the MOAS List in BGP

The BGP community attribute [5] provides a simple way of attaching the MOAS list to a route announcement. The community attribute is an optional transitive BGP attribute of variable length. It can be used to convey additional information to the global routing system for a group of prefixes that share some common properties. Each community attribute consists of four octets. By convention, the first two octets are used to encode an AS number and the semantics of the final two octets may be defined by the AS listed in the first two octets. We propose to reserve one of the 2^{16} values available in the last two octets to indicate a MOAS list. This value is denoted by $MLVal$, *MOAS List Value*, in the remainder of this paper. The community attribute $(X : MLVal)$ indicates that AS X may originate a route to this prefix. The MOAS community value is formally specified in [23].

For example, if a prefix p is originated from all of AS_1, AS_2, ... AS_n, the route updates from $AS_i, (1 \leq i \leq n)$ will include the MOAS List $(AS_1, MLVal), ..., (AS_n, MLVal)$. In Figure 6, AS 1 and AS 2 agree that both of them may originate routes to the same prefix p. When AS 1 originates p, AS 1 will attach the MOAS List, as shown in Figure 7. Similarly, AS 2 will attach the same MOAS list to its route announcement for p.

When a BGP router receives route announcements for

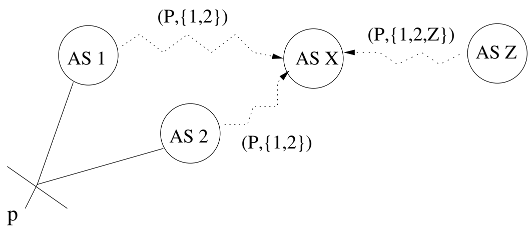


Figure 6. Example scenario

AS 1
$MLVal$
AS 2
$MLVal$

Figure 7. Example MOAS List

the same prefix p from multiple origins, it checks to see whether the MOAS Lists for p from all the announcements are consistent[3]. Here, the consistency is defined as the same set of ASes listed in all the MOAS Lists. The order in the list may differ, but the set of ASes included in each route announcement must be identical. Whenever a BGP router notices any inconsistency in the MOAS Lists received, it should generate an alarm signal; further investigation should be conducted to identify the cause of the inconsistency.

In Figure 6, both AS 1 and AS 2 attached a MOAS List, $(AS1 : MLVal, AS2 : MLVal)$, to their route announcements for p. If AS Z falsely originates a route to p with a MOAS List $(AS1 : MLVal, AS2 : MLVal, ASZ : MLVal)$, another AS, say AS X, will observe an inconsistent MOAS List and will generate an alarm.

Attaching MOAS List to route announcement requires only BGP configuration changes. Checking MOAS List consistency, on the other hand, requires BGP implementation be modified accordingly. However one could deploy the MOAS List checking quickly in the operational Internet via an off-line monitoring process, which periodically downloads the BGP routing messages and checks the MOAS List consistency from multiple peers. If the router is equipped to support the new BGP MIB [10], one could also run a management application to get all MOAS List through the MIB interface and check the MOAS List consistency.

Note also that checking the MOAS list is single set comparison. When an route update for p is received, the MOAS list is simply compared with the existing MOAS list for p (or is simply accepted if this is the first and only announcement for p).

4.3 Limitations of the Proposed Solution

There are a few issues regarding a wide deployment of our solution in the Internet today. In particular, given that BGP community attribute is an optional transitive value, some routers may drop community attribute values associated with a route announcement, an allowed behavior under the current specification. When a router receives multiple route announcements to the same prefix p, some with MOAS list and some do not, it would raise a false alarm. However we note that dropping the MOAS community value from some route announcements should not cause an invalid case to be considered valid, as long as such dropping is limited to a fraction of all the route announcements.

The attachment of a MOAS list also adds to the overall size of the routing table and route announcements. Routes that originate from a single AS need not attach a MOAS list. A route with no MOAS list attached implies that the route may only originate from the AS listed as the last one in the AS path. Our earlier measurement results [22] showed that in today's Internet less than 3,000 routes originate from multiple ASes (including the routes that incorrectly originated from multiple ASes). Furthermore, about 99% of all MOAS cases involve 3 or fewer origin ASes. Thus the MOAS list itself should be relatively short.

Our simple MOAS solution, as described in this paper, helps enhance BGP reliability by distinguishing valid MOAS cases from invalid ones. In its current form, however, it may not be effective in detecting more complex forms of invalid routing announcements. For example, an AS could make a false route announcement with a correct origin AS but a manipulated AS path, or it could falsely announce a route to a prefix longer than p where p is an IP address prefix belonging to another AS. However, our simple MOAS solution shows a first example of how one may utilize the existing network topology itself in detecting faults. We are continuing our work in this direction by enhancing the current solution to detect more complex routing faults.

4.4 Identifying the Correct Origin AS

With our simple MOAS solution, a route announced by a false origin will conflict with the route carrying the correct MOAS list, causing an alarm to be raised. Once an alarm is raised, the router (or network administrator) needs to distinguish the route with correct origin AS(es) from the one with the false origin.

There exist a variety of potential approaches to determine the correct origin AS(es). One possibility is to enhance the DNS database to carry the information of valid origin AS(es) for each address prefix, as proposed in [3]. In this approach, whenever a MOAS conflict for prefix

[3] If a route does not contain a MOAS list, it will be treated as if it carries a MOAS list containing the origin AS.

p, the router performs a DNS lookup to verify the origin AS of p by specifying the DNS Resource Record type as $MOASRR$. If the origin AS in a route announcement does not match any AS number in the AS list of DNS $MOASRR$ record, the route announcement should be considered as bogus. DNS security [16, 6] can be used in this approach to assure the correctness of the DNS database. Combining our solution with this DNS-based checking minimizes the frequency of DNS queries from BGP routers; only in cases of invalid MOAS or dropped MOAS lists will DNS queries be triggered.

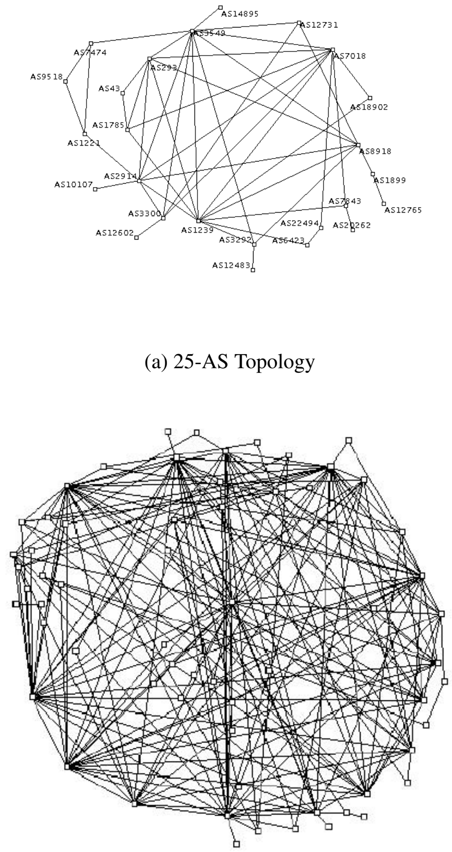
5 Simulation

We used simulation to evaluate the effectiveness of our approach. More specifically, we assume a model where attackers inject false routing announcements at randomly selected locations. We compare the damage the attackers may cause with and without our MOAS solution. We also examine the effectiveness of our solution with different topology sizes and with partial deployment. In the rest of this section we first describe the simulator and the topology used in our simulation and then present the results from three experiments.

5.1 Simulation setup

We use a modified version of the BGP simulator in SSFnet [20] in our simulation. Our simulations use three topologies: a 25-node topology, 46-node topology, and 63-node topoology. In the simulation topologies, each node represents an Autonomous System (AS), and a link between two nodes represents a BGP peering connection (i.e. the two ASes exchange routing information). Figure 8 shows the 25 node and 63 node topologies. The 46-node topology is similar but is omitted for brevity.

In order to generate simulation topologies close to the actual Internet topology, we first get the full BGP routing table from the Oregon RouteViews server [18]. Then we infer BGP peering relations based on the AS Path attribute in the collected BGP routes. For example, if a route to a prefix p has the AS Path *1239 6453 4621*, we consider AS 6453 to have two BGP peers, AS 1239 and AS 4621. We also mark AS 6453 as a *transit AS* since packets to and from AS 4621 may traverse through it (note that AS 1239 is also a transit AS). If an AS does not appear to be a transit AS in any of the routes, we consider it a *stub AS*. Transit ASes represent ISPs (e.g. AS 1239 is Sprint), while stub ASes are networks at the edges of the Internet such as commercial companies and universities. Next, we randomly select $x\%$ of the stub ASes and construct a topology containing these stub ASes and their ISP peers, with the peering relations among all the selected ASes completely preserved. If a transit AS has only one peer left after the initial section, we prune it from the topology. Since the removal of an AS may make it necessary to prune its peer if that peer will have only one or no neighbors left, the pruning needs to be done iteratively. Finally we inspect the topology to make sure that it is a connected graph.



(a) 25-AS Topology

(b) 63-AS Topology

Figure 8. Simulation Topologies

To generate MOAS, we randomly select origin ASes from the stub ASes. In our simulation, each prefix is originated by either one or two valid origin ASes. We do not simulate prefixes that are correctly originated by more than 2 origin ASes since according to our measurement, 96.14% of MOAS cases involve two ASes and 2.7% involve three ASes. We allow any number of attacker ASes to originate invalid routes to the prefix and we choose the attacker ASes randomly from all the ASes. Note that the attackers may have a higher probability to block more valid routes if they are located in transit ASes. Stub (non-tranist) ASes may have a lower level of security, but compromise of a stub AS is less valuable to an attacker since the attacker has less ability to block valid routes.

5.2 Experiment 1: Effectiveness of MOAS List

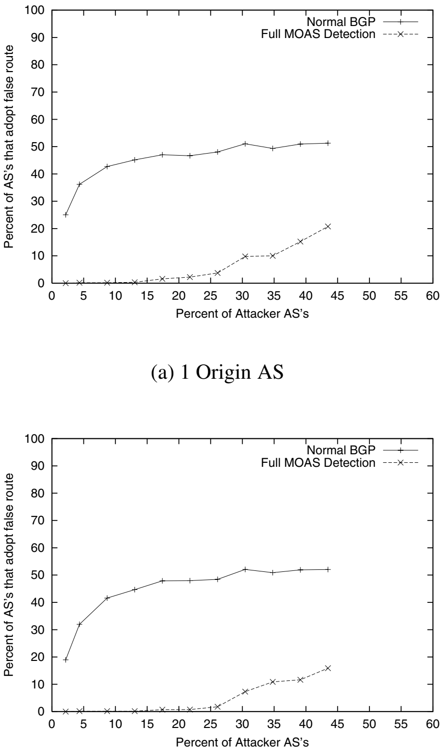
In this experiment, we evaluate how effectively our scheme prevents the propagation of false routing information, by comparing the number of routers adopting false routes with and without using the MOAS List. We assume that all the nodes check the MOAS attributes received from their peers and, once they detect a MOAS case, they stop the further propagation of a false route (e.g. by checking with DNS as proposed in the paper or using some other mechanism).

We randomly select either one or two origin ASes for a prefix and then randomly select M attacker ASes. An attacker AS will incorrectly advertise a route to the prefix. It is easy to see that the number of different selections can be rather large for large topologies. Therefore, rather than simulating all the possible selections, we perform 15 runs for a given number of origin ASes and attackers[4]. In other words, each data point is the average of 15 simulation runs.

Figure 9 shows the results for 46-AS topology, X axis is the percentage of attackers over the total number of the ASes, and Y axis is the percentage of the remaining ASes (excluding attackers) adopting to the false routes announced by the attackers. As one can see, when the number of attackers increases, more (non-attacker) ASes are affected by the false routing information. However, deployment of our simple MOAS solution reduces the percentage of (non-attacker) ASes adopting the false routes greatly. When up to 4% of the AS's are injecting false routing data, more than 36% of the remaining AS's will adopt false routes when without validation of the routes. With our solution, on average only 0.15% of the AS's adopt false routes in the same simulation setting. Even when the number of attackers increases to 30% of the network, only about 9.8% of the remaining AS's adopt false routes, compared to the number of 51% when the MOAS list is not employed. The results are similar for the 25-AS and the 63-AS topologies.

5.3 Experiment 2: Larger Topology

The goal of this experiment is to examine the effectiveness of the MOAS solution in larger topologies. We also compare the results from this experiment with those from the previous experiment to understand the impact of topology size on the robustness of our solution. The topology we use here is the network with 25, 46 and 63 nodes repectively. We have run the experiment with both one origin AS and two origin ASes, the results are similar as shown in Figure 10.



(a) 1 Origin AS

(b) 2 Origin ASes

Figure 9. Spoof-Resilience of Our Scheme in the 46-AS Topology

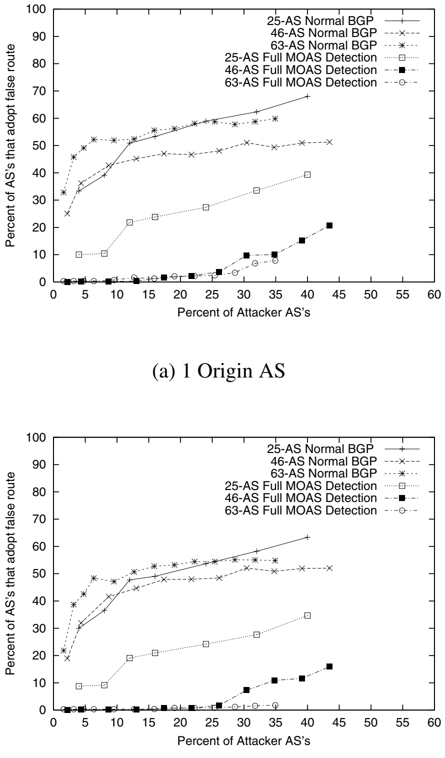
One can make the following observations from Figure 10(a):

1. Without our MOAS solution, the effects of the attackers on the two topologies are quite similar (the gap between the top three curves is much smaller than the gap between the other three curves).

2. With our scheme, the larger 63-AS topology is more robust against random attackers than the smaller 25-AS topology. When the attackers are less than 20% of the total number of ASes, only 2.1% of the remaining ASes are affected by the false routing information. Even when about 35% of the ASes are compromised and announce false routes, only 7.8% of the remaining ASes adopted false route in the 63-AS topology, compared to about 31.2% of (non-attacker) ASes in the 25-AS topology.

The above results suggest that our scheme becomes more effective in larger topologies. We believe that the improved

[4] To get the 15 combinations of origin ASes and attackers, we first select 3 sets of origin ASes from the stub ASes. Then we select 5 sets of attackers for each set of origin ASes.



(a) 1 Origin AS

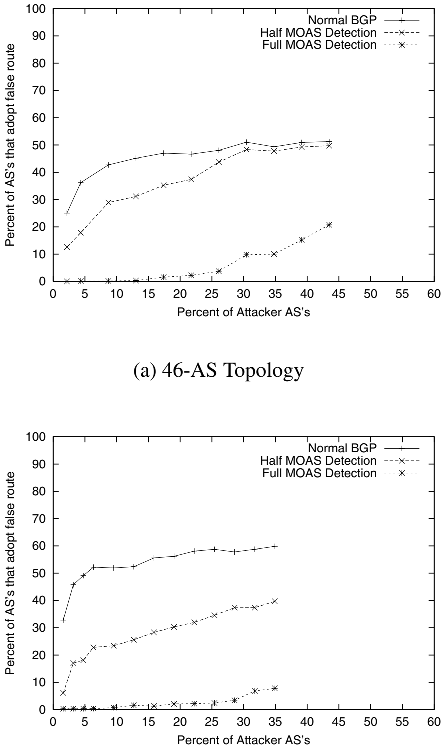
(b) 2 Origin ASes

Figure 10. Comparison between 25-AS, 46-AS and 63-AS Topology

robustness of our solution comes from the fact that ASes are more richly connected in the larger topology, which enables route announcements with the correct AS or correct MOAS lists to reach more ASes. As a result, more ASes detected the inconsistency between correct routes and false routes by the attackers. As part of our continuing research effort we are currently seeking a formal validation proof of this phenomenon.

5.4 Experiment 3: Partial Deployment of MOAS Checking Capability

This experiment evaluates the effectiveness of our MOAS solution when it is only partially deployed in the network. To simulate partial deployment, we randomly select 50% of the nodes to have the capability of processing MOAS List, i.e., they can distinguish between valid and invalid MOAS cases and eliminate false routing announcements. The other nodes ignore the MOAS List, which means they may accept and install a false route in their routing table and advertise the false route to their peers if this route is considered the best path to reach a prefix.



(a) 46-AS Topology

(b) 63-AS Topology

Figure 11. Comparison between Partial and Complete Deployment of MOAS Detection

In Figure 11, we compare the effect of partial and full deployments of MOAS detection. As one can see, even when only half of the nodes can detect MOAS cases, they can still provide protection benefit for the other nodes, because these MOAS-capable nodes stop the false routes from further propagation through them. For example, in the 63-AS topology, partial deployment reduces the percentage of ASes adopting the attackers' routes by more than 63% in the presence of 3% attackers. One can also observe that the larger topology performs better than the smaller topology when MOAS detection is partially deployed.

6 Summary

This paper presented a simple and easily deployable approach for detecting invalid MOAS cases. Instead of adding new cryptographical checks to secure routing information exchanges, our solution adds a simple MOAS list to route announcements. Whenever a prefix is announced by more than one AS, each of the ASes attaches to the prefix an identical MOAS list. If a fault results in an invalid route announcement, the MOAS list attached to this route will be either missing or otherwise inconsistent with the MOAS list on valid announcements for the same address prefix. To prevent a router from detecting the false announcement, an attacker must block all the potential paths through which the valid route can reach the router. As demonstrated by our simulation experiments, blocking all the paths that valid routing information may take to reach a router is difficult, if not impossible, in a richly connected mesh topology such as that found in today's Internet. Our simulation results also show that the solution exhibits more robust behavior against randomly selected attackers in larger networks. One distinguished advantage of our solution is that it can be incrementally deployed in the current network using existing BGP techniques, and can effectively protect the routing system against false routes even when it is partially deployed.

However we believe that the key contribution of this work is our solution's resilience against any single point of failure. In cases where one solely relies on encryption-based techniques to secure routing information exchanges, the compromise of one router can allows the propagation of false route announcements to other routers, and such faults may not be easily detected. On the other hand with our solution, a compromised router can inject false routes into the system, but it cannot easily prevent correct routes from being propagated everywhere, thus other routers can detect the faults by noticing the conflicts between correct and false route announcements. Our solution complements encryption-based security techniques in assuring correct operation of the routing protocol in a large scale network by adding a simple, yet robust fence against traffic hijacking by false route announcements.

References

[1] D. Atkins and R. Austein. Threat Analysis Of The Domain Name System. Technical report, Nov. 2001. Internet Draft draft-ietf-dnsext-dns-threats-00.txt.

[2] R. Barrett et al. Routing Snafu Causes Internet Outage. *ZDNet*, 1997.

[3] T. Bates, R. Bush, T. Li, and Y. Rekhter. DNS-based NLRI origin AS verification in BGP. Internet Draft, Work in Progress, 1998.

[4] V. J. Bono. 7007 Explanation and Apology. NANOG Mailing List msg00444, Apr. 1997.

[5] R. Chandra, P. Traina, and T. Li. BGP Communities Attribute. RFC 1997, Aug. 1996.

[6] D. Eastlake. Domain Name System Security Extensions. RFC 2535, Mar. 1999.

[7] J. Farrar. C&W routing instability. NANOG Mailing List msg00209, Apr. 2001.

[8] J. Garcia-Lunes-Aceves and S. Murthy. A Loop-Free Path-Finding Alogirthm: Specification, Verification and Complexity. In *Proceedings of the IEEE INFOCOM*, Apr. 1995.

[9] J. Haas. Autonomous System Number Substitution on Egress. Internet Draft, Work in Progress, 2001.

[10] J. Haas, S. Hares, S. Willis, J. Burruss, and J. Chu. Definitions of Managed Objects for the Fourth Version of Border Gateway Protocol (BGP-4). Internet Draft, draft-ietf-idr-bgp4-mib-08.txt, 2001.

[11] J. Hawkinson and T. Bates. Guidelines for creation, selection, and registration of an Autonomous System (AS). RFC 1930, 1996.

[12] G. Huston. BGP table statistics. http://www.telstra.net/ops/bgp/as6447/bgp-multi-orgas.html.

[13] G. Huston. Analyzing the Internet's BGP Routing Table. *The Internet Protocol Journal*, 4(1), Mar. 2001.

[14] S. Kent, C. Lynn, and K. Seo. Secure Border Gateway Protocol. In *IEEE Journal of Selected Areas in Communications*, number 4, Apr. 2000.

[15] B. Kroenung. AS8584 taking over the internet. NANOG Mailing List msg00047, Apr. 1998.

[16] D. Massey and S. Rose. DNS Security Introduction and Requirements. Technical report, Sept. 2001. Internet Draft draft-ietf-dnsext-dnssec-intro-01.txt.

[17] RIPE Routing Information Service. http://www.ripe.net/ripencc/pub-services/np/ris-index.html.

[18] The Route Views Project. http://www.antc.uoregon.edu/route-views/.

[19] B. Smith and J. Garcia-Luna-Aceves. Securing the Border Gateway Routing Protocol. In *Proceedings of Global Internet*, Nov. 1996.

[20] The SSFNET Project. http://www.ssfnet.org.

[21] J. Yu. A Route-Filtering Model for Improving Global Internet Routing Robustness. http://www.iops.org/Documents/routing.html.

[22] X. Zhao, D. Pei, L. Wang, D. Massey, A. Mankin, S. Wu, and L. Zhang. An Analysis of BGP Multiple Origin AS (MOAS) Conflicts. Nov. 2001. ACM SIGCOMM Internet Measurement Workshop 2001.

[23] X. Zhao, D. Pei, L. Wang, D. Massey, A. Mankin, S. Wu, and L. Zhang. Validation of Multiple Origin ASes Conflicts through BGP Community Attribute. Internet Draft, draft-zhao-idr-moas-validation-00.txt, 2001.