Investigating the Nature of Routing Anomalies: Closing in on Subprefix Hijacking Attacks

Johann Schlamp, Ralph Holz, Oliver Gasser, Andreas Korsten, Quentin Jacquemart, Georg Carle, Ernst W. Biersack

Investigating the nature of routing anomalies: closing in on subprefix hijacking attacks

Johann Schlamp[1], Ralph Holz[2], Oliver Gasser[1], Andreas Korsten[1], Quentin Jacquemart[3], Georg Carle[1], Ernst W. Biersack[3]

[1] Technische Universität München
{lastname}@net.in.tum.de
[2] NICTA
{firstname.lastname}@nicta.com.au
[3] Eurecom Sophia Antipolis
{firstname.lastname}@eurecom.fr

Abstract. The detection of BGP hijacking attacks has been at the focus of research for more than a decade. However, state-of-the-art techniques fall short of detecting subprefix hijacking, where smaller parts of a victim's networks are targeted by an attacker. The analysis of corresponding routing anomalies, so-called subMOAS events, is tedious since these anomalies are numerous and mostly have legitimate reasons.
In this paper, we propose, implement and test a new approach to investigate subMOAS events. Our method combines input from several data sources that can reliably disprove malicious intent. First, we make use of the database of a Internet Routing Registry (IRR) to derive business relations between the parties involved in a subMOAS event. Second, we use a topology-based reasoning algorithm to rule out subMOAS events caused by legitimate network setups. Finally, we use Internet-wide network scans to identify SSL-enabled hosts in a large number of subnets. Where we observe that public/private key pairs do not change during an event, we can eliminate the possibility of an attack. We can show that subprefix announcements with multiple origins are harmless for the largest part. This significantly reduces the search space in which we need to look for hijacking attacks.

1 Introduction

Autonomous Systems (ASes) use the Border Gateway Protocol (BGP) to propagate information about paths to certain destinations. Despite being vital to traffic forwarding on the Internet, BGP does not feature any security mechanisms like origin or neighbor authentication. Reports such as [1,2,8,11] have shown that attacks do occur and are real threats. Systems like S-BGP [5] and RPKI [4] have been developed to add integrity protection and origin authentication to BGP. However, due to the considerable resources needed to deploy them, they are not widely used. Consequently, a number of mechanisms to (at least) detect attacks on BGP have been developed [12,7,14,9,15]. Although they are

able to detect certain attacks like the hijacking of entire IP prefixes, they suffer from relatively high rates of false-positive alarms.

In this paper, we investigate a particularly interesting phenomenon in BGP that is elusive to investigations yet can be an indication of a serious threat: subprefix hijacking where rogue ASes announce routes to prefixes that are *fully contained* inside prefixes originated by other, legitimate ASes. We call these subMOAS events. Such an attack leads to a 'black hole' for a victim's network since BGP generally prefers routes to more specific prefixes. However, business relationships between ASes and their customers naturally lead to a very large number of subMOASes as well. It is an unsolved challenge to tell the many benign events apart from the (rarer) malicious ones: on average, we observe nearly 75 subMOASes *per hour* with peaks of several hundred events.

Our contribution in this work is a filter system to identify legitimate subMOAS events such that a much more reasonable number of 'still suspicious' cases remains. These can either be manually inspected or serve as the input for future detection systems. Our approach is to combine data sources that are external to BGP to draw conclusions about the legitimacy of subMOAS events. First, we use information from the RIPE database to infer business and management relationships between the IRR objects stored in the database. Such information can only be altered by entities with valid access credentials. Our assumption is that an attacker does not have these credentials. Second, we use a topology algorithm to reason whether an attacker targets subprefixes of his own upstream provider. This is highly unlikely as the victim would simply be able to filter out the malicious BGP updates. Third, we use data from Internet-wide scans of the SSL/TLS landscape to determine hosts whose public/private key combinations are unique and remain stable over a longer period of time. These hosts serve as beacons. If their public/private key pair remains the same during a subMOAS event, we can rule out malicious interference. The assumption here is that a BGP hijacker cannot compromise hosts in hijacked prefixes and steal their keys. In our evaluation, we will see that our methods are very effective on the input data. Since their coverage can still be increased, this is an encouraging result.

The remainder of this paper is organised as follows. Section 2 presents related work. We describe our methodology in Section 3 and present our results and the lessons learned in Section 4.

2 Related work

There is a huge body of relevant and related literature. In the following, we can only focus on a few selected contributions. Evidence that BGP hijacking attacks occur has been provided in several publications, *e.g.,* by Ramachandran and Feamster [10] (short-lived tampering with BGP for spam purposes) and Schlamp *et al.* [11] (a longer-lived occurrence). Possibly the first attempt to detect hijacks was made by Lad *et al.* [7]: a control-plane technique focusing exclusively on reporting multiple-origin AS (MOAS) prefixes. The authors of [9] provided heuristics to assess that the announced MOAS paths comply

with standard economy-based routing policy. Wählisch studied the correlation between routing policies and RPKI-invalid announcements in [13]. The authors of [15] use a hop-count metric to evaluate the number of IP hops between a monitor and a target network—changes in this number indicate a topology change. Argus [12] uses multiple monitors for ping measurements to distinguish between two zones affected and unaffected by the respective BGP updates. Importantly, these techniques focus primarily on MOAS. In contrast, we focus on subMOAS events. Here, active probing to detect an affected and an unaffected part of the Internet topology is not possible, since all of the Internet topology is affected by a corresponding BGP update (due to BGP's preference of routes to more specific prefixes). The above methods would thus not work. The authors of [3] discuss detection techniques for subMOASes. Their approach requires that upstream providers allow IP spoofing, which is not always the case. The mechanism in [14] can detect network cut-offs from inside a victim's network, but works on a local level only.

3 Methodology

Our methodology consists of four steps. First, we determine actual subMOAS events from BGP routing tables and update messages. Subsequent steps focus on eliminating subMOAS events with legitimate causes. To this end, we establish a filter chain. First, we use the RIPE IRR database to infer the ownership for certain so-called IRR resources. If we find that an alleged attacker actually is the legitimate owner of a resource or has been delegated authority over it, we consider such a subMOAS event as legitimate. Our filter is currently limited to the RIPE space, but can be extended to other IRR databases. The next filter is a topology-based reasoning algorithm: the idea is that an attacker is unlikely to hijack his own upstream provider as this provider could simply counter the attack by filtering out malicious BGP udpates. The last filter uses data from active SSL/TLS scans. For a given prefix in a subMOAS event, we verify if Web hosts in this prefix presented the same public key before and during a subMOAS occurence. If so, we may assume that the prefix is not hijacked as the attacker would have to be in possession of the private key, too, to fake a successful connection. This leaves us with a much smaller remainder of subMOAS events.

3.1 Identification of subMOASes

In a subMOAS-based attack, an attacker uses his AS to attract a victim's traffic by advertising a subprefix of a victim's (less specific) prefix. This efffectively blackholes a part of the victim's network. To discover subMOAS events, we analyze RouteViews Oregon's routing table. We store prefix announcements in a binary prefix tree, where nodes hold information about the origin of an announcement. We only consider *effective* subMOAS: we discard cases where affected prefixes are fully announced by multiple origins, *i.e.,* regular MOAS cases. Instead, we look for more specific prefixes that are originated by a different AS

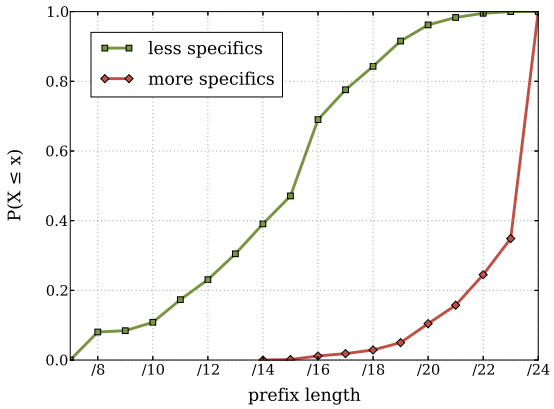


Fig. 1: Distribution of prefix lengths for subMOAS announcements (CDF).

than the enclosing prefix. We thereby compare the most specific parts of a prefix, *i.e.,* those parts that are decisive with respect to longest prefix matching, with its directly enclosing prefix to obtain all IP ranges that are affected by a subMOAS announcement. For instance, if the prefixes `10.0.0.0/22` and `10.0.0.0/24` are originated by the same origin AS, we would still recognize a subMOAS event for the `/22` prefix if `10.0.0.0/23` is originated by a different AS.

As of June 1, 2014, RouteViews Oregon's routing table holds 511,118 announced prefixes (≈62.7% of the IPv4 space). A total of 76,121 prefixes are subMOAS announcements (covering ≈3.44% of the IPv4 space). These figures emphasize that subMOAS are a very common and naturally occuring phenomenon, with attacks hard to detect in the large number of benign events. On average, more specific subMOAS prefixes are longer than corresponding less specifics by a factor of 2^8 (see Figure 1). Hence, it will be essential to identify a great number of SSL/TLS-enabled hosts in advance in order to allow for the comparison of public keys before and during *any* new event.

3.2 Utilizing IRR databases

All five Internet Routing Registries (IRR) maintain databases that contain information pertaining to the management of Internet resource holders. A recent study [6] matched prefixes and ASes observed in BGP and IRR by looking for appropriate database objects. We provide a generalized set of inference rules for benign subMOAS events, which take into account multiple origins observed in BGP as well as complex relationships between the affected prefixes and a suspicious origin AS.

Our filter is designed for the RIPE database as RIPE provides daily snapshots with a precise data model and a certain amount of consistency enforced. Still, IRR databases are updated by individual resource holders and can thus be outdated or even hold conflicting information. Our filter accounts for this. Note that filters for other IRR databases are easy to design; this is ongoing work.

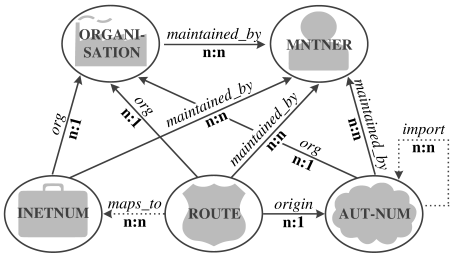


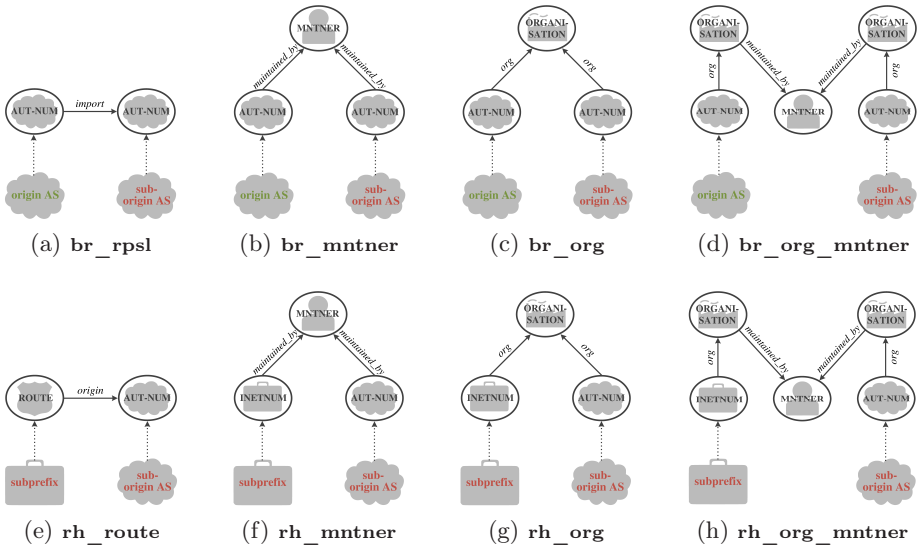
Fig. 2: Entities and relations in the RIPE database relevant for our filter.

Instance	Nodes	Relations
MNTNER	**48,465**	
←*maintained_by*– [*]		5,307,883
ORGANISATION	**81,260**	
←*org*– [*]		199,644
AUT-NUM	**27,616**	
←*import*– AUT-NUM		221,690
←*origin*– ROUTE		245,831
INETNUM	**3,871,827**	
ROUTE	**236,604**	

Table 1: Information stored in our graph database, June 2014.

Data model Since February 2012, we download and evaluate daily snapshots of the RIPE database. Figure 2 shows entities and relations in the RIPE database that are of significance for our work. We use a graph database to store the extracted data using the same schema as in the figure. We also track all changes over time. The RIPE database models access rights with MNTNER objects. Only maintainers with valid credentials can modify or delete objects. For any object, this is expressed by adding a *maintained_by* reference pointing to the respective MNTNER object. ORGANISATION objects are optional and mainly used to provide administrative contact details. The RIPE snapshots remove details for privacy reasons but preserve the references to the objects themselves. INETNUM objects represent allocated or assigned IPv4 prefixes managed by RIPE. ROUTE objects are created by resource holders and are used to document or confirm intended prefix announcements by specific ASes. To create a ROUTE object, a resource holder needs to provide valid maintainer credentials for both the INETNUM and the AUT-NUM object. The corresponding *maps_to* relation is computed by our parsing algorithm. AUT-NUM objects represent AS numbers and may be referenced as the *origin* of ROUTE objects. Our parsing algorithm also deduces *import* relations from free-text description fields, which are often used to model routing policies in the so-called Routing Policy Specification Language (RPSL). When resources are deleted from the RIPE database, RPSL definitions may still reference (now) non-existing ASes. We account for this by tracking such orphaned *import* relations.

As of June, 2014, our database holds more than 4 million nodes and 5 million relations extracted from the RIPE database. Table 1 provides details for selected objects that are relevant for our approach. We can see that less than 50,000 MNTNER objects share more than 5 million incoming *maintained_by* references. Although optional, roughly 80,000 ORGANISATION objects are referenced by almost 200,000 other objects. Less than 30,000 AUT-NUM objects *import* routing policies from more than 220,000 other AUT-NUM objects. Nearly 250,000 ROUTE objects bind prefix announcements to less than 30,000 AUT-NUM objects. We will see that these figures allow our filter to be very effective.



(a) **br_rpsl** (b) **br_mntner** (c) **br_org** (d) **br_org_mntner**

(e) **rh_route** (f) **rh_mntner** (g) **rh_org** (h) **rh_org_mntner**

Fig. 3: IRR inference rules used for the legitimization of subMOAS events.
(a)–(d) Legitimate business relationships. (e)–(h) Legitimate resource holders.

Infering resource ownership Recall that our fundamental assumption is that an attacker does not have the credentials to change the RIPE database in order to cover his attack. Accordingly, we look for legitimate relationships between the parties involved in a subMOAS event to disprove an attack. Given a routing change that results in a subMOAS, we map the affected AS numbers and prefixes to `AUT-NUM` and `INETNUM` objects in our graph database. We then traverse the graph along a path of legitimizing relations. We look for paths between a) the two affected AS or b) the more specific prefix and its origin AS. If we succeed with a), we can infer a valid business relation between the victim and the suspected attacker. If we succeed with b), the suspected attacker holds ownership rights for the more specific prefix and is thus authorized to originate it from his AS.

Legitimizing paths are formed by one or more of the following relations: *import*, *origin*, *maintained_ by* and *org*. Figure 3 shows the complete set of our inference rules. Entities without surrounding circles represent subMOAS information derived from BGP data, encircled items represent nodes in our database. We first look for an *import* relation from the alleged victim to the attacker (Figure 3 (a)). This would imply that the suspected victim deliberately updated the RIPE database to document his willingness to accept the suspected attacker's route updates. This indicates a business relationship rather than an attack, and we consider it proof for a legitimate subMOAS event. Similar arguments apply for the victim's `AUT-NUM` object being maintained by the attacker's `MNTNER` object (Figure 3 (b)) since no victim would grant his attacker such privileges. Relations to a common `ORGANISATION` object (Figure 3 (c)) and even a path from different

affected organisations to a common MNTNER (Figure 3 (d)) can also be considered strong evidence for an underlying business relationship.

If we are not able to find a path with the above rules, we look for evidence that a suspected attacker is in fact the legitimate holder of a subprefix resource in question. We first check if we can map the subprefix to a ROUTE object. If so, we search for an *origin* relation to the suspected attacker's AUT-NUM object (Figure 3 (e)). To create such a ROUTE object, valid maintainer credentials are needed for the AUT-NUM object, but also for the implicitly given INETNUM object represented by the subprefix. If the alleged attacker is able to provide both, we consider him the owner of the subprefix and the subMOAS case to be legitimate. Note that we also check for ROUTE objects that bind less specific prefixes to the suborigin AS. This implies that the attacking AS is the owner of the corresponding larger IP range, of which only a part is advertised in BGP. As network operators are free to announce their networks in any given size, such cases are legitimate, too.

The remaining rules in Figure 3 (f)–(h) are similar to those in (b)–(d): we aim to identify a legitimizing path based on shared MNTNER or ORGANISATION objects—in these cases between the subprefix mapped to an INETNUM object and the AUT-NUM object of the originating AS. Once again, we do not look for exact matches to the INETNUM object but also allow for larger IP ranges since a resource holder is not required to advertise his assigned prefixes as a whole.

Our figures from Table 1 show that these rules have the potential to be highly effective, since we observe a high degree of interconnections: On average, MNTNER objects are referenced by 110 other objects, and ORGANISATION objects have at least eight incoming relations. In addition, we have nearly ten times more ROUTE objects and *import* relations than AUT-NUM objects. It is therefore promising to look for objects with common references to these objects. Note that our approach does not require the RIPE database to be complete, and not even to be conflict-free. Our inference rules are solely based on legitimate objects. In case of absent or conflicting database objects, we are unable to establish a legitimizing path—we cannot wrongly legitimate a subMOAS event this way.

3.3 Topology reasoning

The next filter in our chain is topology-based. For each subMOAS occurrence, we extract all AS paths that lead to the affected subprefixes and build a directed graph. In essence, this graph represents all possible paths to the subprefixes' origins, regardless of the selected route. We use the graph to check if at least one of the observed AS paths to the more specific origin AS contains the origin AS of the less specific one. If this is the case, we consider the subMOAS event to be legitimate: if it were illegitimate, the owner of the less specific prefix would not forward malicious BGP updates upstream. The legitimate scenario occurs, for example, when a smaller Internet service provider obtains Internet connectivity and a block of IP addresses from a larger carrier; other reasons might be multihoming setups or the use of static routes invisible to BGP.

(a) IPv4-wide ground truth (b) Rescan of active hosts

Fig. 4: Timeline for obtaining our ground truth.

3.4 Cryptographic assurance with SSL/TLS

Our final filter uses data sets obtained from our regular Internet-wide scans of the SSL/TLS protocols on port 443 (HTTPS). The idea is to identify legitimate subMOAS events by checking the public/private key pair used in SSL/TLS handshakes. We assume that an attacker cannot obtain cryptographic keys from a victim. Thus, if a host uses the same key pair before and during a subMOAS event, we may infer the legitimacy of an event. For this to work, we first need to establish a ground truth: a collection of mappings of IP addresses to public keys. Due to the fluctuating nature of the Internet in terms of IP address assignments, routing paths and change-overs of SSL/TLS keys, we carry out two subsequent scans to establish a ground truth.

First, we initiate a SSL/TLS scan of the entire routable IP space. To reduce the intrusiveness and to avoid our probes being dropped by destinations, the scans are carried out much more slowly than it would be technically possible. We also inform a number of CERTs, research institutes and blacklist providers before a scan, and maintain our own blacklist of networks based on feedback from operators.

Figure 4 shows the timeline for obtaining our ground truth. Our first scan lasted from 7-24 April 2014. It yielded 27.2 million IP addresses where we could retrieve certificate chains in the SSL/TLS handshake. For our ground truth, we focus on particularly stable hosts with unchanging IP addresses and stable, unique public keys. We thus scanned the 27.2 million hosts a second time one month later (7-24 May 2014) and filtered out all IP addresses for unresponsive hosts or for which the public key had changed. We arrived at 5.4 million stable hosts. The final step was to discard hosts that had already been affected by subMOAS events. This is necessary since a subMOAS event at the time of the scan would mean we would have connected to a host possibly under the control of an attacker. By checking against *all* BGP messages received in intervals of 15 minutes, roughly 20,000 hosts were discarded in this step. Note that discarded hosts may be eventually reincluded into the ground truth by rescanning on a periodic basis, thus mitigating the effects of short-lived subMOAS events. The resulting set of 5,356,634 hosts can be considered stable: for each host, both its IP address and corresponding public key had remained unchanged, and no subMOAS event occured during our connection to the host.

Note that our ground truth naturally becomes less effective over time due to long-term changes of hosts. The implication for our methodology, however, is

once again unproblematic, since we gradually miss out on legitimizing subMOAS events, but we cannot accidentally overcount. In addition, we update our ground truth on a monthly basis to overcome a decrease in our coverage.

With this ground truth available, we can now reliably detect whether hosts affected by an emerging subMOAS event still present the same public key as before the event. To this end, we are in need of a real-time framework to timely initiate the re-scanning of affected hosts.

3.5 Real-time framework

subMOAS events may be of long duration (in the range of several months), but we also observed events that lasted much shorter (*e.g.,* for several hours or minutes only). To account for this variability in duration, we set up a real-time framework to continously analyze subMOAS events. Note that it is imperative that our SSL/TLS scans are carried out *before and within* the life time of an event, *i.e.,* we need to perform our scans quickly after a subMOAS arises.

Our real-time framework comprises several steps that are executed every two hours. First, we obtain the latest BGP data: a two-hour old RIB dump and all BGP update messages until present time. We extract all subMOAS events that started within this time frame and have not been withdrawn yet. Next, we apply our IRR filters and identify legitimate events. We also apply our topology reasoning algorithm and use our ground truth scan to look up stable SSL/TLS hosts contained in the more specific prefixes to initiate SSL/TLS scans.

At the same time, we obtain all scan results from the previous run and compare cryptographic host keys to those obtained in our ground truth scan. Note that, in general, one must not assume that a scan always reaches the more specific prefix. At the moment we observe a subMOAS event, routing may have already changed along the path of our upstreams, hence our BGP view might be out-dated. Due to such propagation delays inherent to BGP, this issue cannot be resolved by a tight coupling of our SSL/TLS scanner to the subMOAS detection alone. Instead, we sanitize our scan results with the help of a subsequent validation process. After we have collected a new set of cryptographic keys, we further evaluate the following two hours of BGP data, and discard scan results for which the subMOAS event changed or vanished within this time frame. Note that man-in-the-middle attacks where an attacker is able to forward our scans to the legitimate destination are beyond the scope of our work. Besides, our approach does not allow us to analyze events that last shorter than two hours. However, this is no inherent limitation and can be mitigated by selecting a shorter analysis period (*i.e.,* investing more resources).

4 Evaluation

We begin our evaluation with an analysis of the frequency of subMOAS events during the time frame of our experiment. We then show how much each filter in our chain can contribute to identify legitimate events. Based on our results, we discuss lessons learned at the end of this section.

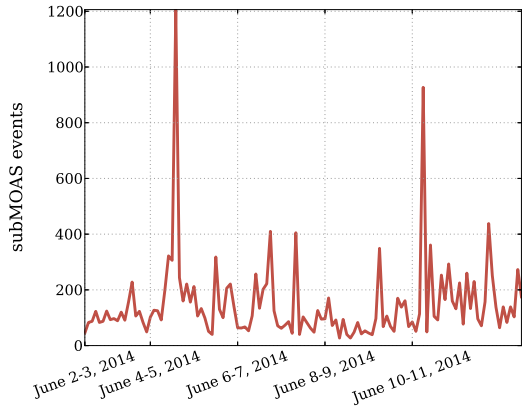


Fig. 5: subMOAS events observed over the duration of our experiment.

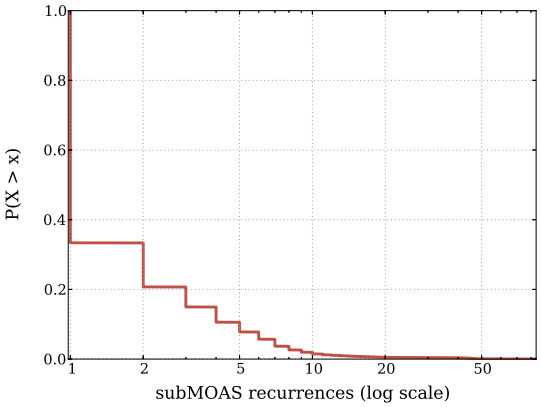


Fig. 6: Distribution of subMOAS recurrences, CCDF.

4.1 subMOAS analysis

Figure 5 shows the frequency of subMOAS events we observed in the period of 2-12 June 2014. On average, we encountered 148.2 events over two hours (the minimum number is 27; the maximum number is 1,206). Figure 6 gives details on subMOAS events that occurred more than once, *i.e.,* concerned the same prefixes and ASes. On average, subMOASes recurred 2.2 times, with a maximum of 84 occurrences.

During the duration of our experiments, we observed a total of 8,071 unique subMOAS events. We were able to legitimize 46.5% of these events by subsequent application of our filter chain. Table 2(a) presents an overview of individual filter results. IRR-based analysis could rule out 10.8% legitimate events; topology reasoning could contribute about 31.7%, and SSL/TLS about 22.9%.

With our combined filter chain, we are able to legitimize nearly half of all subMOAS events present in today's routing tables. We emphasize that this is not

	total	in %
All subMOAS events	**8,071**	**100%**
IRR analysis	870	10.78%
topology reasoning	2,560	31.72%
SSL/TLS scans	1,851	22.93%
Legitimate events (cum.)	**3,755**	**46.53%**

(a) Combined filter results.

	total	in %
Individual SSL/TLS scans*	**37,043**	**100%**
with different SSL/TLS key	773	2.09%
no response (port closed)	3,302	8.91%
with same SSL/TLS key	32,968	89.0%
Covered subMOAS events	**2,116**	**100%**
Legitimate events	**1,851**	**87.48%**

(c) SSL/TLS scan results.
*986 scans were removed due to routing changes

	total	in %
Covered subMOAS events	**1,048**	**100%**
`br_rpsl`	362	34.54%
`br_mntner`	519	49.52%
`br_org`	51	4.87%
`br_org_mntner`	145	13.84%
`rh_route`	692	66.03%
`rh_mntner`	599	57.16%
`rh_org`	159	15.17%
`rh_org_mntner`	160	15.27%
Legitimate events (cum.)	**870**	**83.02%**

(b) IRR analysis results.

Table 2: Overview of our results.

an upper limit that would be inherent to our methodology: it is simply because, at this point, we only use sources that cover about 60% (4,795) of all events. Rather, the results for the individual filters suggest that adding further data source like other IRRs (ARIN, APNIC, etc.) or other cryptographic protocols (SSH, IMAPS, etc.) have the potential to shrink the result space much further.

IRR analysis Table 2(b) shows how effective our IRR-based filters are at eliminating legitimate subMOAS events for prefixes registered by RIPE. Rules that aim at capturing business relationships can eliminate about 65% of these events. Rules that establish legitimate resource holding can eliminate about 72%. In combination, we find that **83.0%** of events that are based in the RIPE service region are legitimate. Our previous analysis with Table 1 indicated that IRR inference rules based on `MNTNER` and `ROUTE` objects could perform best; the results presented above confirm this finding.

SSL/TLS scans Table 2(c) shows the total numbers of observed keys. In terms of legitimized subMOAS events, we are able to rule out **87.5%** of events with at least one SSL/TLS-enabled host in the respective subprefixes. Figure 7 shows the distribution of SSL/TLS hosts per subMOAS prefix. 75% of the prefixes host at least one SSL/TLS-enabled machine, 25% even contain more than 10 hosts.

Note that for more than 75% of all subMOAS events, we have more than one host available to use for cryptographic confirmation. We even have more than ten hosts available in about 25% of all events. The average number of SSL/TLS hosts per subMOAS subprefix is 17; the minimum and maximum numbers are 1 and 2,070, respectively. These figures allow our SSL/TLS filter to be highly

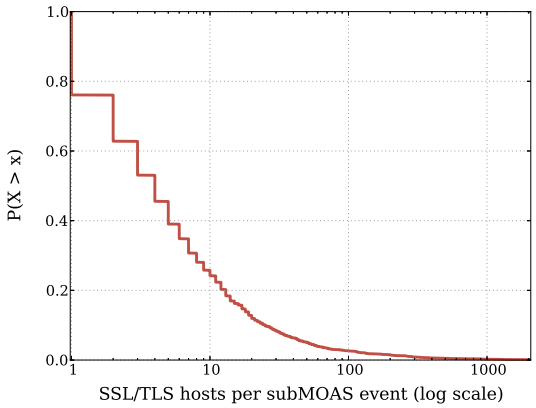


Fig. 7: Distribution of SSL/TLS hosts per subMOAS subprefix (CCDF). Only subprefixes with at least one SSL/TLS host have been considered.

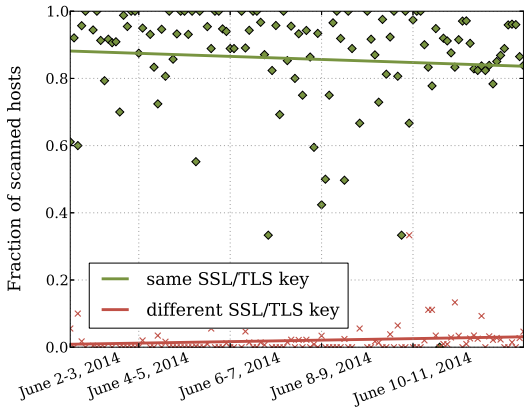


Fig. 8: Percentage of same and different SSL/TLS keys during our experiment.

robust against short outages of single hosts, since it is enough for us to confirm that *at least one* cryptographic key remains unchanged per subMOAS event.

Figure 8 shows that the populations of unchanging and changing keys remain relatively stable for the lifetime of our ground truth. While a certain decline is evident, it remains in the range of 5% or less. Finally, Figure 9 shows the percentages of hosts that became unresponsive during our live scans, which increases very slowly, too. These findings suggest that the interval for obtaining new ground truth hosts can be set to one month or even longer. Note that outliers with a larger fraction of changed certificates or unresponsive hosts are the result of a lower initial number of available ground truth hosts.

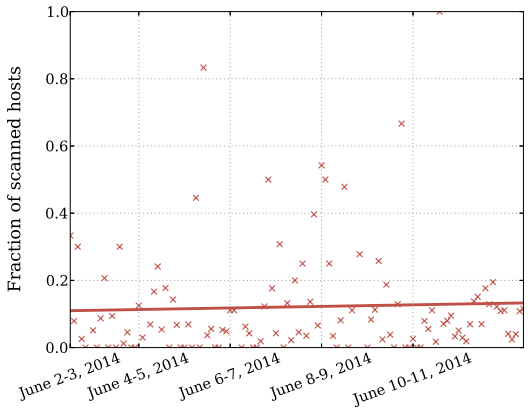


Fig. 9: Number of unresponsive SSL/TLS hosts over the duration of our experiment.

4.2 Lessons learned

The results from our filters are quite encouraging. Given that we achieve high elimination rates for the IP space we can currently cover (already 60%), we offer the following conclusions.

First, data obtained from IRR databases is highly useful to identify legitimate subMOAS events, even if some data may be incomplete or outdated. Our results encourage us to extend our IRR analysis to the remaining databases in other service regions—we expect a significant increase of our coverage. Furthermore, we would encourage IRR operators to publish database snapshots on a daily basis to aid in this effort at demystifying routing anomalies.

Second, active scans are equally powerful. The coverage of our methodology corresponds exactly to the number of Web hosts that use unique keys, a set of hosts that remained pleasingly stable throughout our experiments. The coverage can be even increased in the future by focusing on additional cryptographic protocols, e.g. like IMAPS and SSH. We intend to perform regular ground truth scans and to deploy our filter techniques continously.

Our work aims at the detection and analysis of subMOAS events. It is thus not applicable to other types of routing anomalies that do not exhibit subMOAS conflicts, e.g. interception attacks. However, our ultimate goal is to be able to reduce the huge search space for subprefix hijacking attacks to a manageable size for manual inspection, and to allow automated reasoning about subMOAS routing anomalies. Our analysis chain lends itself well to integration of future detection systems: a) to narrow down the number of suspicious routing anomalies and b) to cross-check the resulting alarms.

5 Conclusions and outlook

We introduced a methodology that allows us to reliably identify subMOAS events with legitimate causes. Our method combines data from several sources and

proves promising: although coverage for the entire Internet can be improved, our individual filter techniques are highly effective. Our findings show that both IRR databases and active scans are useful tools to reason about routing anomalies in-depth. Moreover, we outlined straightforward steps to increase coverage, which puts manual inspection of the remaining subMOAS events within reach. Finally, we intend to grow our framework into a service that makes its data publicly available on a continuous and permanent basis. This framework promises to be greatly beneficial for future systems to detect subprefix hijacking. We invite the research community to participate in this effort. We would be delighted to have our results used as input for further detection systems or by seeing further filters developed by fellow researchers.

Acknowledgements This work has been supported by the German Federal Ministry of Education and Research (BMBF) under support code 01BY1203C, project *Peeroskop*, and by the European Commission under the FP7 project *EINS*, grant number 288021.

References

1. H. Ballani, P. Francis, and X. Zhang. A study of prefix hijacking and interception in the Internet. In *Proc. ACM SIGCOMM 2007*, pages 265–276, 2007.
2. Hepner, Clint and Earl Zmijewski. Defending against BGP man-in-the-middle attacks. Talk at BlackHat 2009, 2009.
3. X. Hu and Z. M. Mao. Accurate real-time identification of IP prefix hijacking. In *Proc. IEEE Symposium on Security and Privacy*, pages 3–17, 2007.
4. G. Huston and R. Bush. Securing BGP and SIDR. *IETF Journal*, 7(1), 2011.
5. S. Kent, C. Lynn, and K. Seo. Secure Border Gateway Protocol (SBGP). *IEEE Journal on Selected Areas in Communications*, 18(4), April 2000.
6. A. Khan, H.-c. Kim, T. Kwon, and Y. Choi. A comparative study on ip prefixes and their origin ases in bgp and the irr. *SIGCOMM Comput. Commun. Rev.*, 43(3):16–24, July 2013.
7. M. Lad, D. Massey, D. Pei, Y. Wu, B. Zhang, and L. Zhang. PHAS: A prefix hijack alert system. In *Proc. 15th USENIX Security Symposium*, volume 15, 2006.
8. A. Pilosov and T. Kapela. Stealing the Internet: An Internet-scale man in the middle attack. In *Talk at DEFCON 16*, 2008.
9. J. Qiu and L. Gao. Detecting bogus BGP route information: going beyond prefix hijacking. In *In Proc. 3rd Int. Conf. on Security and Privacy in Communication Networks (SecureComm)*, 2007.
10. A. Ramachandran and N. Feamster. Understanding the network-level behavior of spammers. In *Proc. ACM SIGCOMM 2006*, 2006.
11. J. Schlamp, G. Carle, and E. W. Biersack. A forensic case study on as hijacking: the attacker's perspective. *ACM SIGCOMM CCR*, 43(2):5–12, 2013.
12. X. Shi, Y. Xiang, Z. Wang, X. Yin, and J. Wu. Detecting prefix hijackings in the Internet with argus. In *Proc. ACM SIGCOMM IMC*, 2012.
13. M. Wählisch, O. Maennel, and T. C. Schmidt. Towards Detecting BGP Route Hijacking Using the RPKI. *ACM SIGCOMM CCR*, 42(4):103–104, August 2012.
14. Z. Zhang, Y. Zhang, Y. C. Hu, Z. M. Mao, and R. Bush. iSPY: Detecting IP prefix hijacking on my own. *IEEE/ACM Trans. on Networking*, 18(6):1815–1828, 2010.
15. C. Zheng, L. Ji, D. Pei, J. Wang, and P. Francis. A light-weight distributed scheme for detecting IP prefix hijacks in real-time. In *Proc. ACM SIGCOMM 2007*, 2007.