RPKI vs ROVER
Comparing the Risks of BGP Security Solutions
Full version from June 18, 2014.*

Aanchal Malhotra and Sharon Goldberg

Boston University,
aanchal4@bu.edu,goldbe@cs.bu.edu

BGP, the Internet's interdomain routing protocol, is highly vulnerable to routing failures that result from unintentional misconfigurations or deliberate attacks. To defend against these failures, recent years have seen the adoption of the Resource Public Key Infrastructure (RPKI), which currently authorizes 4% of the Internet's routes. The RPKI is a completely new security infrastructure (requiring new servers, caches, and the design of new protocols), a fact that has given rise to some controversy [1]. Thus, an alternative proposal has emerged: Route Origin Verification (ROVER) [4, 7], which leverages the existing reverse DNS (rDNS) and DNSSEC to secure the interdomain routing system.

Both RPKI and ROVER rely on a hierarchy of authorities to provide trusted information about the routing system. Recently, however, [2] argued that the misconfigured, faulty or compromised RPKI authorities introduce new vulnerabilities in the routing system, which can take IP prefixes offline. Meanwhile, the designers of ROVER claim that it operates in a "fail-safe" mode, where "[o]ne could completely unplug a router verification application at any time and Internet routing would continue to work just as it does today". There has been debate in Internet community mailing lists [1] about the pros and cons of both approaches. This poster therefore compares the impact of ROVER failures to those of the RPKI, in a threat model that covers misconfigurations, faults or compromises of their trusted authorities.

1 ROVER and RPKI Primer

The RPKI and ROVER each provide a trusted mapping from an IP prefix[1] to the Autonomous System(s) (ASes) authorized to originate (*i.e.,* claim to be the destination for) the prefixes in BGP. Both therefore protect against *prefix* and *subprefix hijacks*, where the hijacking AS originates routes for a victim IP prefix, that it is not authorized to originate. During the hijack, network traffic destined for the victim IP prefix then flows to the hijacker's AS instead, where it can be dropped, modified, or subject to surveillance.

rDNS. The rDNS is used to resolve IP addresses to hostnames. Zones in rDNS correspond to IP prefixes. Data in the reverse zone can be authenticated by standard DNSSEC signatures. Per Figure 1, the root for IPv4 is in-addr.arpa, maintained by IANA.[2] The root delegates 8.0.0.0/8 to the reverse zone maintained by Level3, which delegates 8.8.8.0/24 to Google and 8.34.114.0/24 to Metro Net. An rDNS zone is *authoritative* for its IP prefix and subprefixes, *excluding* those delegated to a child zone. Level3 is authoritative for all prefixes covered by 8.0.0.0/8 except those covered by 8.8.8.0/24 and 8.34.114.0/24.

* This is the full version of the poster to appear in *SIGCOMM'14*, August 17–22, 2014, Chicago, IL, USA.

[1] IP prefix 8.0.0.0/8 has *length* 8 and *covers* 8.8.8.0/24.

[2] 8.0.0.0/8 is directly allocated by IANA. For non-legacy prefixes, RIRs are also a part of reverse zone hierarchy

ROVER. ROVER augments rDNS zones with records that resolve IP prefixes to AS authorized to originate them. With a goal of requiring minimal changes to the rDNS, ROVER introduces: (a) A new naming convention for IP prefixes of arbitrary length.[3] (b) Two new types of resource records: an *Secure Route Origin (SRO)* to authorize an IP prefix π in the zone to be originated in BGP by an AS a, and a *Route Lock (RLOC)* to opt a zone in or out of ROVER. The presence of an RLOC in a zone means that an SRO is needed for every IP prefix for which the zone is authoritative. In Figure 1, Level3 has an RLOC and is authoritative for 8.0.0.0/9; thus, if AS 3356 wishes to originate 8.0.0.0/9 in BGP, a new SRO must be added to Level3's rDNS zone in Figure 1. Metro Net's zone does not have an RLOC, so the BGP routes for prefixes covered by 8.34.114.0/24 are opted-out of ROVER. (Note the limited scope of Level3's RLOC in Figure 1: even though 8.0.0.0/8 covers 8.34.114.0/24, BGP routes covered by 8.34.114.0/24 are still opted out of ROVER because Level3 delegated this prefix to Metro Net.) SRO and RLOC records are validated in the usual DNSSEC manner. (We use the term *validation chain* to describe the chain of objects from a given SRO or RLO record up to the root of the DNSSEC hierarchy.) An SRO or RLOC is *valid* if it passes DNSSEC validation.

ROVER & route validity. We say a *BGP route* is an IP prefix π and the AS a originating it in BGP. A router issues a ROVER query for *every* new route it learns in BGP. The response is (DNSSEC) validated and stored in a local cache, with standard DNS caching used to store responses. For example, if a router learns a BGP route for prefix 8.8.8.0/24 originated by AS 15169, it should query for *any* SRO corresponding to prefix 8.8.8.0/24; if no SRO is returned for the prefix, the router queries for the RLOC for the authoritative zone (Google). ROVER's SROs and RLOCs determine the validity of an advertised BGP route (π, a) as:

- *Valid.* (π, a) is *valid* if it has a matching valid SRO.
- *Unknown.* (π, a) is *unknown* if both the RLOC for the zone that is authoritative for π, and the SRO for (π, a) are absent or if either fails DNSSEC validation.
- *Invalid.* (π, a) is *invalid* if (1) there are valid SROs for π that do not match a, OR (2) there is no valid SRO for π and there is a valid RLOC for π's authoritative zone.

In Figure 1, a route for 8.0.0.0/8 originated by AS 3356 is *valid* (because of the matching valid SRO), and a route for 8.0.0.0/8 originated by AS 666 is *invalid* (because of the mismatched SRO). Any route for 8.34.114.0/24 is *unknown* (because Metro Net does not have an RLOC). Any route for 8.0.0.0/9 is *invalid* (because Level3 has an RLOC).

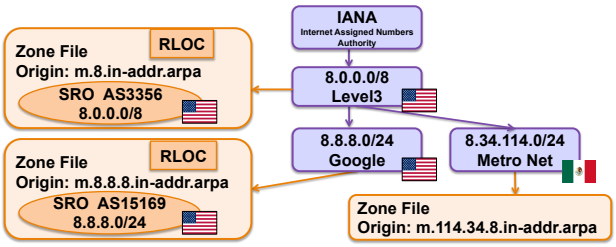

Fig. 1. Sample ROVER hierarchy

[3] This is needed because existing rDNS records deal with individual IP addresses, not IP prefixes. Moreover, the rDNS naming convention creates ambiguities when prefixes do not fall in traditional octet boundaries *i.e.,* /8, /16, /24.

RPKI. RPKI has a similar hierarchy of authorities. Each authority has a *Resource Certificate (RC)*, signed by its parent, containing its allocated IP address space and cryptographic public key. An RC can sign (a) other RCs to suballocate address space, or (b) *Route Origin Authorizations (ROAs)* (the equivalent of SROs in ROVER) to authorize an IP prefix π covered by the RC to be originated by an AS a. The RPKI has no equivalent to an RLOC.

In Figure 2, a Regional Internet Registry (ARIN) issues an RC that delegates 8.0.0.0/8 to Level3. Level3's RC issues an RC subdelegating 8.8.8.0/24 to Google, an RC subdelegating 8.34.114.0/24 to Metro Net, and a ROA authorizing AS 3356 to originate 8.0.0.0/8. Google's RC issues a ROA authorizing AS 15169 to originate 8.8.8.0/24. In contrast to ROVER, The scope of Level3's RC is *all* prefixes covered by 8.0.0.0/8, including those that it has subdelegated to Google and Metro Net; Level3 could issue a ROA mapping 8.8.8.0/24 to AS 3356, even though it has already subdelegated this prefix to Google.

RPKI route validity. In contrast to ROVER, where a new ROVER query is made each time a router learns a new route in BGP, an ISP that uses the RPKI downloads *all* objects in the public RPKI repositories to its local cache once per day. Objects are cryptographically verified, and the ISP pushes a list of valid ROAs to its routers. This list determines the validity states for a BGP route (π, a), in a manner that is slightly different from that of ROVER:

- *Valid.* (π, a) is *valid* if it has a matching valid ROA.
- *Unknown.* (π, a) is *unknown* if there is no valid *covering ROA*; a covering ROA is any ROA for a prefix covering π.
- *Invalid.* (π, a) is *invalid* if it is not *unknown* or *valid*.

In Figure 2, all routes for prefix 8.34.114.0/24 will be *invalid* since (1) they do not have a matching ROA, but (2) there is a covering ROA for prefix 8.0.0.0/8. This is in contrast to the ROVER hierarchy in Figure 1, where routes for prefix 8.34.114.0/24 are *unknown*; this is one manifestation of the "fail-safe" approach of ROVER.

Routing policies. A BGP router uses its own *local policies* to decide whether to discard, or assign lower preference to, *invalid* or *unknown* BGP routes. As discussed in [2], a router that discards *invalid*/*unknown* routes has the best possible protection against attacks on BGP, but can lose connectivity to routes that become *invalid*/*unknown* as a result of threats to the RPKI/ROVER. Here we assume the most plausible set of policies [2]: that a router discards *invalid* routes, and prefers *valid* routes over *unknown* routes. These local policies imply that a route will go offline if a threat to the RPKI/ROVER causes it to become *invalid*, but a route that becomes *unknown* will still be reachable.

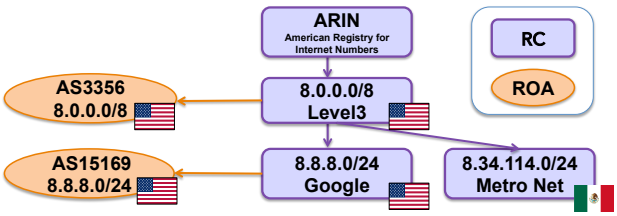

Fig. 2. Sample RPKI hierarchy

2 Attacking Paths to ROVER/RPKI

We consider an attacker that disrupts the communication path to an RPKI/ROVER server (dropping and corrupting valid records, or injecting ill-formed records) but *cannot* forge signatures of RPKI/ROVER authorities. How does this impact the validity of BGP routes?

Blackholing BGP routes. The assumed routing policies imply that a BGP route can be blackholed if the attacker can change its validity state from *valid* to *invalid*. [2] argues that an on-path RPKI attacker can do just that, by disrupting the delivery of ROAs from an RPKI repository during an ISP's daily update of its local cache. If an on-path attacker corrupts a single bit in the ROA for 8.8.8.0/24 in Figure 2 during a bulk download of ROAs from the RPKI, then the corresponding route for 8.8.8.0/24 becomes *invalid* if there is a covering ROA authorizing AS 3356 to originate 8.0.0.0/8. Thus, the 8.8.8.0/24 route would no longer be reachable. (On the other hand, the RPKI uses *manifests* to indicate which objects it stores. Disrupting delivery of objects or manifests raises alarms at routers, making this attack more transparent; see discussion in [5].) With ROVER, however, the route would still be reachable; if an attacker disrupts the response to a ROVER query for 8.8.8.0/24, causing it to fail DNSSEC validation, then routes for 8.8.8.0/24 become *unknown*. This is a crucial manifestation of ROVER's "fail-safe" approach.

Circumventing ROVER/RPKI protection of BGP. Another concern is an attacker that disrupts communications with the RPKI (or ROVER) so that it may circumvent its protections against routing attacks. Suppose a subprefix hijacker at AS 666 originates 8.0.0.0/9, thus *sub*prefix hijacking 8.0.0.0/8 away from victim AS 3356. The RPKI (ROVER), respectively, mark the hijacked route as *invalid*, as it has a covering ROA but lacks a matching ROA in Figure 2 (lacks a matching SRO but is opted into to ROVER with an RLOC in Figure 1). If an attacker can disrupt communications with RPKI/ROVER so that the hijacked route is marked as *unknown* instead of *invalid*, then routers will select the hijacked route, and fall victim to the hijack. (This follows because there is no *valid* BGP route for 8.0.0.0/9 that is preferred over the *unknown* hijacked route).

Can an on-path attacker alter the status of an *invalid* route to *unknown*? With ROVER, this is certainly possible. Suppose a router issues a ROVER query for the subprefix 8.0.0.0/9 that was hijacked in BGP; while there is no valid matching SRO in Figure 1, the on-path attacker will just inject a bogus, invalidly-signed SRO of its own for 8.0.0.0/9 and any AS. This SRO will fail DNSSEC validation, and the hijacked route therefore becomes *unknown*. Because ROVER issues a separate query for each route learned in BGP, this highly-targeted attack will affect only the hijacked prefix. Worse yet, advanced DNSSEC attacks [3, 6] mean that there is a risk that *off-path* attackers could launch such attacks; we will investigate these in future work.

Meanwhile, [2] argues that attacks of this form are less targeted with the RPKI. To transition a route (π, a) from *invalid* to *unknown*, an on-path RPKI attacker needs to disrupt the delivery of *all* the ROAs that cover prefix π. However, this would cause all routes authorized by these ROAs to go from *valid* to *unknown*, making the attack easier to detect. Moreover, disrupting delivery of RPKI objects (or manifests) would also raise alarms at routers.

A tradeoff. This analysis suggests that ROVER's nice fail-safe characteristics also allow for targeted circumvention of ROVER's protections of BGP via disruptions to the communication path. Meanwhile, the RPKI is more brittle, since a disruption on the communication path to the RPKI can lead to routes being blackholed in BGP. One the other hand, unlike ROVER, the RPKI also contains mechanisms (*i.e.,* manifests) that can help routers detect and respond to disruptions on the communication path.

3 Misconfigurations

We consider how misconfigurations of ROVER or RPKI authorities impact BGP route validity.

An authority that misconfigures its ROAs can cause all routes covered by the misconfigured ROA to become *invalid*; this also includes routes for prefixes that were subdelegated to descendant authorities. In Figure 2, if Level3 issues a ROA for 8.0.0.0/8 and forgets to issue a ROA authorizing Metro Net's AS to originate 8.34.114.0/24, then routes for Metro Net's prefix become *invalid*.

ROVER, however, avoids this. In ROVER, any ill-formed (badly signed), missing, or mismatching SRO/RLOC record in a zone can only impact routes within the zone; the limited scope of the RLOC ensures that a misconfigured RLOC/SRO has no impact on child zones. In Figure 1, Metro Net is opted out of ROVER, since it has no RLOC. Thus, all routes covered by 8.34.114.0/24 are marked as *unknown*, regardless of any misconfigurations in Level3's zone. On the other hand, a DNSSEC misconfiguration at an ancestor zone (Level3) that breaks the DNSSEC validation chain to a descendant's (Google's) SRO/RLOC records, can impact routes in the descendant zone; again, however, the worst that can happen is that they all become *unknown*. This is another manifestation of ROVER's "fail-safe" approach.

4 Takedowns by an Ancestor

Thus far, we have considered misconfigurations, and attacks on the communication path. We now consider threats created by malicious or compromised ancestor authorities. For example, if Level3 goes rogue, what harm can it do to Google and Metro Net?

ROVER allows an ancestor authority to execute very targeted blackhole attacks on BGP routes. A ROVER resolver cannot track the zones delegated by an authority, unless the delegation chain is ready stored in its (DNS) cache. In what follows we assume that the delegation chain is not cached. (Indeed, an attacking ancestor that wishes to enforce this assumption can always wait for a DNS record's TTL to expire.) Now suppose Level3 wants the *valid* route for 8.8.8.0/24 originated by Google's AS 15169 per Figure 2 to become *invalid*, and thus unreachable in BGP. To do this, Level3 can simply respond to a ROVER query for 8.8.8.0/24 with a mismatched but valid SRO, *e.g.,* an SRO for prefix 8.8.8.0/24 and AS 666. Again, this is a highly targeted attack, affecting only 8.8.8.0/24 and no other prefix.

Meanwhile, [2, Appendix A] shows that it is more complicated for a malicious RPKI authority to cause a BGP route to become *invalid* when it matches a ROA issued by a descendant RC. Attacking a specific BGP route (without harming other BGP routes as collateral damage) sometimes requires the RPKI ancestor authority to issue suspicious new RPKI objects, making its actions easier to detect. We refer to reader to [2], because these attacks can be quite complex.

5 Summary & Open Questions

ROVER provides several nice "fail-safe" characteristics. However, these characteristics can be exploited by (1) attackers that disrupt ROVER queries in order to hijack BGP routes, and (2) ROVER authorities that want to blackhole BGP routes authorized by their descendants. ROVER's use of point queries also means that targeted attacks are easier, as compared to the RPKI. Our future work seeks to experimentally validate these threats to ROVER, and to quantify their impact on ROVER's ability to prevent routing attacks.

There are many other controversial issues related to ROVER that remain open. For instance, [2] shows that circular dependencies are present between RPKI and BGP; are they present between ROVER and BGP? Does RPKI's approach of downloading repositories wholesale provide better performance than the point-queries used in ROVER? How do ROVER and the RPKI compare in partial deployment? We believe that experimentation with both solutions is necessary to resolve these issues.

References

1. R. Austein. "Re: rpki vs. secure dns?", msg18. seclists NANOG Archive, June 2012. `http://seclists.org/nanog/2012/Jun/18`.
2. D. Cooper, E. Heilman, K. Brogle, L. Reyzin, and S. Goldberg. On the risk of misbehaving rpki authorities. In *HotNets*, page 16, 2013.
3. H. Duan, N. Weaver, Z. Zhao, M. Hu, J. Liang, J. Jiang, K. Li, and V. Paxson. Hold-on: Protecting against on-path dns poisoning. *SATIN*, 2012.
4. J. Gersch and D. Massey. Rover: Route origin verification using dns. In *ICCCN*, pages 1–9, 2013.
5. E Heilman, D Cooper, L Rezyin, and S Goldberg. From the consent of the routed: Improving the transparency of the rpki. In *SIGCOMM'14*, 2014.
6. A. Herzberg and H. Shulman. Fragmentation considered poisonous. In *IEEE CNS*, pages 224–232, 2013.
7. C. Olschanowsky J. Gersch, D. Massey and L. Zhang, editors. *DNS Resource Records for Authorized Routing Information*. IETF Internet-Draft, February 2013.