SIDR Operations
Internet-Draft
Intended status: Standards Track
Expires: 8 January 2026

Y. Li
H. Zou
G. Xie
CNIC, CAS

7 July 2025

Matching-first Route Origin Validation
draft-li-sidrops-matching-rov-00

Abstract

Route Origin Validation (ROV) using the Resource Public Key Infrastructure (RPKI) enables BGP routers to identify illegitimate routes that violate Route Origin Authorizations (ROAs). However, widespread deployment of RPKI requires validation systems to process high volumes of route announcements against increasingly large ROA datasets, where ROV adoption faces significant barriers due to concerns about its processing efficiency. This document identifies a performance issue inherent in current ROV procedure, which could be exacerbated as the expanding coverage of ROAs.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 8 January 2026.

Table of Contents

1. Introduction

The rapid deployment of RPKI has significantly expanded Route Origin Authorization (ROA) coverage. However, validating large volumes of route announcements against increasingly massive ROA datasets poses a significant challenge for routers. These efficiency concerns regarding Route Origin Validation (ROV) create a barrier to its wider adoption.

Current ROV implementations, defined in RFCs 6483 and 6811, typically perform an initial coarse-grained match of a BGP route against all ROAs before verifying an exact match. This approach optimizes for routes without ROA coverage, allowing their validity state (e.g., NotFound) to be determined quickly. However, as ROA coverage expands, this method suffers performance degradation: validation latency increases proportionally with the expanding ROA datasets, especially the increasing number of BGP routes that can pass the validation. Evaluations of open-source ROV implementations confirm this scaling issue.

To mitigate performance degradation caused by shifting validity distributions, this document defines a modified validation model through step reordering. The optimized workflow first performs precise matching against all ROAs to immediately identify Valid routes, followed by coverage verification to determine NotFound status. Performance evaluations of reference implementations confirm the model's superiority.

1.1. Requirements Language

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in RFC 2119 [RFC2119] when they appear in all upper case.

1.2. Terminology

Besides, the terms "VRP", "VRP Prefix", "VRP ASN", "Route", "Route Prefix", "Route Origin ASN", "Covered", "Matched", "NotFound", "Valid", "Invalid" are to be interpreted as described in RFC 6811 [RFC6811] when they appear in all upper case.

If a Route is Covered by a VRP, it is termed a covering VRP of that Route. Similarly, if a Route is Matched by a VRP, it is termed a matching VRP of that Route. For a given Route, a matching VRP is a special covering VRP such that: 1) its VRP ASN is the same as the Route Origin ASN and 2) its maxLength value is greater than or equal to the length of Route Prefix.

For a given Route, the process of finding out a covering VRP or figuring out whether there exists a covering VRP is termed *FC*, while the process of finding out a matching VRP or figuring out whether there exists a matching VRP is termed *FM*. The process to examine whether a covering VRP is a matching VRP is termed *EM*.

2. Overview of Current ROV Procedure

As a informational, RFC 6483 [RFC6483] describes a general procedure to determine the validity state of a given Route against a set of VRPs, which can be summarized as follows using the terms FC and EM.

* FC step: iteratively perform FC to find out all covering VRPs, output NotFound if there is no covering VRP; otherwise go to the EM step.

* EM step: iteratively perform EM to examine every covering VRP, output Invalid if there is no matching VRP; otherwise output Valid once a matching VRP is found.

RFC 6811 [RFC6811] has standardized the validation procedure and provides the corresponding pseudo-code, which illustrates a similar procedure as the above one but could be more efficient regarding system implementation. This validation procedure can be summarized as follows using the terms FC and EM.

* FC step: perform FC to find out the first covering VRP, output NotFound if this step fails; otherwise go to the EM step.

* EM step: iteratively perform EM to examine the current covering VRP and perform FC to find out the next covering VRP, output Invalid if there is no matching VRP; otherwise output Valid once a matching VRP is found.

Actually, the ROV implementations with the RTRLib [RTRLib], BIRD [BIRD] and BGP-SRX [BGP-SRX] all follow the above validation procedure. They all adopt Trie-like structures for FC and can output NotFound fast at the earliest stage. We call this validation procedure the *non-covering* first validation.

3. Performance Issue Analysis and Evaluation

Non-covering first validation optimizes for BGP routes in the NotFound state. However, this approach delays determining the validity state of Valid routes, which was a tolerable tradeoff during early RPKI deployment when sparse ROA coverage resulted in predominantly NotFound states. However, today, with more than half of global BGP routes are in the Valid state (and growing), non-covering first validation incurs escalating latency as both ROA volume and Valid routes increase simultaneously.

To quantify this scaling limitation, we measured ROV efficiency using real-world ROA datasets and BGP routing tables. We evaluated four implementations: RTRLib, BGP-SRX, BIRD’s basic ROV (BIRD-basic), and BIRD’s Trie-optimized ROV (BIRD-opt). The mean validation latency per route during full-table processing served as the efficiency metric. Related source code, datasets and results can be accessed via: https://github.com/FIRLab-CNIC/h-2ROV

3.1. Comparative Validation Latency Impact by Validity Distribution

Validation of the full RIB (collected from RIPE NCC RIS RRC00 collector on 1st July, 2025) against ROAs (collected from RIPE NCC on 1st July, 2025) classifies routes into three subsets: Valid, Invalid, and NotFound. Using these subsets, we synthesized 55 test RIBs of fixed size (1,050,986 routes each) with controlled validity distributions. Each test RIB was constructed by randomly sampling specified quantities of routes from the Valid, Invalid, and NotFound

subsets. The four ROV mechanisms preloaded with the collected ROAs were evaluated against all 55 synthetic RIBs, with results detailed below.

Experimental results confirm an inverse correlation between the proportion of Valid routes and validation efficiency in RTRLib, BGP-SRX, and BIRD-opt when the sum of routes in Valid or NotFound states is held constant. In tests with no Invalid routes, increasing the Valid ratio from 0 to 100% in 10% increments induced latency multipliers of 1.6, 1.9, and 3.5 for the respective implementations. With 10% of routes in Invalid state, the latency multipliers become 1.5, 1.8, and 3.1 under identical conditions. The root inefficiency stems from how non-covering first validation defers validation of Valid routes, where the latency penalty compounds as the proportion of Valid routes increases.

Contrasting with other implementations, BIRD-basic exhibits an inverse performance relationship: mean validation latency decreases with higher Valid route proportions. This anomaly stems from its structural divergence from true non-covering first validation. Without Trie optimization, BIRD-basic employs iterative hash probes to identify covering VRPs and cannot output NotFound until all probes complete. This design causes severe inefficiency in NotFound-dominant environments. This is the reason why BIRD-opt was developed. However, as Valid routes become prevalent, the way using iterative hash probes show its advantage since many Valid routes could be confirmed within few probes. Experimental results show BIRD-basic outperforming BIRD-opt in pure-Valid scenarios. Thus, BIRD-basic accidentally approximates (though inefficiently) the validation model proposed in Section 4.

The Validation Latency with RTRLib (nanosecond):

% of Valid	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
% of NotFound											
0%	1190	1214	1241	1266	1282	1306	1351	1359	1376	1406	1464
10%	1173	1192	1215	1237	1361	1280	1321	1346	1476	1399	
20%	1147	1170	1194	1221	1232	1254	1281	1315	1321		
30%	1123	1148	1170	1195	1206	1245	1268	1351			
40%	1092	1119	1141	1237	1189	1218	1238				
50%	1072	1095	1119	1144	1165	1184					
60%	1047	1069	1092	1117	1137						
70%	1018	1048	1062	1092							
80%	996	1020	1038								
90%	962	984									
100%	929										

Table 1

The Validation Latency with BGP-SRX (nanosecond):

% of Valid	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
% of NotFound											
0%	580	650	703	761	817	872	946	994	1022	1115	1138
10%	597	653	701	765	804	859	947	1007	1054	1115	
20%	605	650	706	760	813	865	951	1002	1019		
30%	601	653	706	765	809	892	945	974			
40%	606	655	725	770	840	877	941				
50%	609	664	719	767	835	872					
60%	614	659	715	775	823						
70%	616	668	720	771							
80%	617	667	724								
90%	621	669									
100%	616										

Table 2

The Validation Latency with BIRD-opt (nanosecond):

% of Valid	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
% of NotFound											
0%	442	458	474	494	508	522	534	564	564	616	591
10%	420	439	449	466	475	496	524	557	569	589	
20%	392	416	421	439	450	466	511	519	502		
30%	366	381	411	410	419	449	513	464			
40%	337	353	372	398	405	411	435				
50%	309	323	343	353	385	382					
60%	279	296	310	327	362						
70%	253	270	299	296							
80%	225	240	254								
90%	198	220									
100%	168										

Table 3

The Validation Latency with BIRD-basic (nanosecond):

% of Valid	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
% of NotFound											
0%	1457	1358	1259	1169	1076	995	882	800	699	602	508
10%	1501	1418	1314	1215	1123	1031	947	859	757	685	
20%	1542	1456	1354	1273	1171	1080	980	916	793		
30%	1594	1502	1437	1329	1219	1150	1068	931			
40%	1646	1559	1494	1423	1295	1224	1101				
50%	1681	1592	1513	1410	1351	1226					
60%	1757	1637	1553	1458	1399						
70%	1777	1735	1634	1513							
80%	1842	1742	1647								
90%	1931	1853									
100%	1969										

Table 4

3.2. Temporal Dynamics in ROV Efficiency

To evaluate ROV efficiency over time, quarterly snapshots of RIBs and ROAs were collected from January 2015 to July 2025 (January 1, April 1, July 1, and October 1 annually). These form 43 paired datasets, each containing a RIB and ROAs collected on the same date. Every dataset was evaluated by validating all routes in its RIB against corresponding ROAs using the four ROV mechanisms respectively, with results detailed below.

The Validation Latency with Five ROV Implementations (nanosecond):

	h-ROV	RTRLib	BIRD-basic	BIRD-opt	BGP-SRX
20150101	165.098	523.560	757.002	118.413	175.254
20150401	158.655	491.642	791.139	115.407	178.476
20150701	155.087	497.265	827.130	113.161	180.930
20151001	154.083	497.265	834.028	113.122	182.548
20160101	163.399	503.778	863.558	113.804	185.322
20160401	149.410	504.032	689.180	113.623	184.570
20160701	165.920	514.139	738.007	114.837	188.005
20161001	146.649	517.331	744.602	115.580	189.179
20170101	147.189	519.751	755.858	116.618	192.976
20170401	151.378	528.262	791.766	117.495	195.810
20170701	155.063	531.632	819.672	119.775	201.450
20171001	158.403	542.005	884.956	121.907	204.583
20180101	161.316	541.712	924.214	123.305	212.811
20180401	171.438	553.403	732.601	125.109	216.967
20180701	154.154	634.921	823.723	143.740	264.971
20181001	159.668	648.508	838.926	145.879	273.973
20190101	154.967	668.003	923.361	154.727	298.063
20190401	164.123	684.463	759.301	160.436	314.367
20190701	174.307	699.790	759.301	164.177	328.192
20191001	158.957	722.543	872.600	174.917	353.982
20200101	178.987	742.390	985.222	185.632	371.609
20200401	173.370	771.010	1025.641	195.313	405.186
20200701	179.340	800.000	827.815	210.793	448.029

20201001	188.076	835.422	912.409	228.102	490.677
20210101	205.719	865.801	989.120	244.141	529.661
20210401	194.970	882.613	1109.878	254.582	559.284
20210701	205.592	906.618	1273.885	265.816	575.705
20211001	198.570	883.392	1140.251	242.307	544.959
20220101	206.313	930.233	1406.470	271.370	592.417
20220401	210.881	962.464	1034.126	285.225	651.466
20220701	211.685	982.318	1046.025	298.954	678.887
20221001	212.811	1005.025	1101.322	314.169	707.714
20230101	216.263	1000.000	1187.648	303.767	679.348
20230401	223.115	1034.126	1254.705	328.947	695.894
20230701	225.785	1039.501	1245.330	329.056	767.460
20231001	255.754	1062.699	1333.333	355.999	775.795
20240101	257.533	1079.914	1362.398	370.096	795.545
20240401	239.292	1062.699	1443.001	350.140	747.384
20240701	244.618	1104.972	1631.321	383.877	823.045
20241001	246.488	1097.695	1510.574	376.506	794.913
20250101	246.488	1106.195	1555.210	385.654	859.845
20250401	255.493	1156.069	1049.318	397.298	884.173
20250701	254.001	1173.709	1089.325	409.668	892.857

Table 5

Over time, Valid route prevalence increases significantly, with particularly rapid growth after 2019. This correlates with a proportional decline in NotFound states. Meanwhile, Invalid routes remain stable below 3.0%, exhibiting minor fluctuations.

Consequently, ROV efficiency becomes predominantly governed by the latency performance for either Valid or NotFound route processing. Concurrently, expanding ROA datasets can also compound ROV computational demands.

Expanding ROA datasets and rising Valid route prevalence collectively drive significant latency growth in RTRLib, BGP-SRX, and BIRD-opt, which have increased by factors of 1.6, 2.4, and 2.2, respectively since 2020. However, BIRD-basic, while consistently outperformed by its Trie-optimized variant (BIRD-opt) throughout the observation period, exhibits marginal throughput gains since April 2025. This confirms that validity distribution has become the dominant factor to ROV efficiency of BIRD-basic.

3.3. A Brief Summary

* While non-covering first validation provided efficiency advantages during RPKI’s initial deployment phase, it has become unsuitable for present-day validity distributions and will exhibit progressively worse scalability as RPKI adoption continues to grow.

* As RPKI adoption increases, two distinct factors, expanding ROA datasets and rising Valid route prevalence, independently yet collectively impact ROV efficiency. These separable influences warrant distinct optimization strategies.

4. Matching First Route Origin Validation

Non-covering first validation, though pivotal during initial RPKI deployment, fails to scale in modern environments where Valid routes have exceed 50% and continue growing, rendering it fundamentally mismatched to current validity distributions and those in future. Revisiting the validation model is now imperative to guide ROV implementations toward capitalizing on rising Valid route prevalence while mitigating scaling challenges from expanding ROA datasets. The critical optimization involves prioritizing matching VRP discovery over covering VRP searches. Such a matching first validation procedure can be defined as follows.

* FM step: find out a matching VRP or figuring out whether there exists a matching VRP, output Valid if there is one; otherwise go to the FC step.

* FC step: find out a covering VRP or figuring out whether there exists a covering VRP, output Invalid if there is one; otherwise output NotFound.

4.1. Another Reference Implementation

BIRD-basic constitutes an implementation of this model, thus benefiting significantly from rising Valid route proportions. However, its efficiency remains suboptimal due to requiring multiple hash probes, each involving sequential FM examinations, to confirm Valid state. In contrast, h-ROV [h-ROV] provides a more efficient reference implementation through bitmap-encoded VRPs. This approach guarantees most Valid routes are identified via a single hash probe followed with a bitwise operation.

Experimental results (detailed below) for h-ROV, evaluated per the methodology described in Section 3.1, demonstrate 66.3%~85.7% lower validation latency than BIRD-basic under identical validity distributions. When Valid routes exceed 30%, h-ROV achieves the lowest latency among all evaluated ROV mechanisms.

The Validation Latency with h-ROV (nanosecond):

% of Valid	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
% of NotFound											
0%	209	207	204	199	195	192	187	186	192	174	171
10%	218	217	213	211	205	202	200	197	192	189	
20%	228	225	222	220	215	211	206	206	199		
30%	236	236	235	229	222	223	218	199			
40%	246	245	242	220	234	232	227				
50%	255	251	251	247	245	239					
60%	265	263	257	257	253						
70%	276	271	266	266							
80%	283	281	276								
90%	294	293									
100%	298										

Table 6

Experimental results for h-ROV (methodology per Section 3.2) show latency escalation consistent with other implementations due to expanding ROA datasets. However, h-ROV exhibits more moderate latency growth, offset by efficiency gains from rising Valid route prevalence. While BIRD-opt demonstrated the lowest latency before July 2019, h-ROV became the most efficient implementation after October 2019, with its performance advantage widening over time.

4.2. Lessons Lenard from Reference Implementations

* While the matching-first validation model enables scalable performance amid expanding ROA coverage, realized ROV efficiency remains contingent on implementation-specific optimizations.

* Prefix-level VRP management enables highly efficient implementations of the matching first validation model.

5. Security Considerations

The security considerations in Section 6 of RFC6811 [RFC6811] are also applied to this document.

6. IANA Considerations

This document has no IANA actions.

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

7.2. Informative References

[BGP-SRX] NIST, "BGP Secure Routing Extension (BGP-SRx) Software Suite", Web https://github.com/usnistgov/NIST-BGP-SRx, 2022.

[BIRD] CZ.NIC, "The BIRD Internet Routing Daemon", Web https://bird.network.cz/, 2023.

[h-ROV] ZD Ni, "From Address Blocks to Authorized Prefixes Redesigning RPKI ROV with a Hierarchical Hashing Scheme for Fast and Memory-Efficient Validation", 2025, <https://www.usenix.org/system/files/nsdi25-ni.pdf>.

[RFC6483] Huston, G. and G. Michaelson, "Validation of Route Origination Using the Resource Certificate Public Key Infrastructure (PKI) and Route Origin Authorizations (ROAs)", RFC 6483, DOI 10.17487/RFC6483, February 2012, <https://www.rfc-editor.org/rfc/rfc6483>.

[RFC6811] Mohapatra, P., Scudder, J., Ward, D., Bush, R., and R. Austein, "BGP Prefix Origin Validation", RFC 6811, DOI 10.17487/RFC6811, January 2013, <https://www.rfc-editor.org/rfc/rfc6811>.

[RTRLib] NIST ANTD, "RTRlib. The RPKI RTR Client C Library.", Web https://rtrlib.realmv6.org/, 2023.

Authors' Addresses

Yanbiao Li
CNIC, CAS
Email: lybmath@cnic.cn

Hui Zou
CNIC, CAS
Email: zouhui@cnic.cn

Gaogang Xie
CNIC, CAS
Email: xie@cnic.cn