

Poster: Scalable and Interpretable Multilayer Overlay Network Checking via Ensemble Verification

Xinzhe Liu^{1,3}, Yahui Li¹, Han Zhang^{1,3}, Xia Yin^{1,3}, Xingang Shi^{1,3},
Zhiliang Wang^{1,3}, Gang Ren^{1,3}, Jiangyuan Yao²

¹Tsinghua University, ²Hainan University, ³Zhongguancun Laboratory

ABSTRACT

The multilayer network configurations are more complex and error-prone. Modern control plane verifiers optimized for underlay networks exhibit limitations in scalability and interpretability when extended to multilayer overlay networks. This paper proposes MEV, the first control plane verifier for multilayer overlay networks. MEV introduces a new verification method named ensemble verification, which decouples the complex interactions and properties across multilayer into each single layer, performing modeling and verification within each layer independently.

CCS CONCEPTS

• **Networks** → **Network reliability**;

KEYWORDS

Network Verification, Multilayer Network, Ensemble Verification

ACM Reference Format:

Xinzhe Liu^{1,3}, Yahui Li¹, Han Zhang^{1,3}, Xia Yin^{1,3}, Xingang Shi^{1,3}, Zhiliang Wang^{1,3}, Gang Ren^{1,3}, Jiangyuan Yao². 2024. Poster: Scalable and Interpretable Multilayer Overlay Network Checking via Ensemble Verification. In *ACM SIGCOMM 2024 Conference (ACM SIGCOMM Posters and Demos '24)*, August 4–8, 2024, Sydney, NSW, Australia. ACM, New York, NY, USA, 3 pages. <https://doi.org/10.1145/3672202.3673722>

1 INTRODUCTION

Today, the overlay/underlay pattern has been successfully used in the design of network architecture. To meet the diverse needs from tenants, networks [2, 7] are configured with many encapsulation protocols such as IP-in-IP [11], VXLAN [10], virtualizing a physical underlay network into multiple and multilayer logical overlay networks. However, this also renders the network configuration more complex and error-prone, which may cause widespread service interruptions and substantial economic losses [3, 13].

Hence, numerous verification tools have been developed. One category is data plane verifiers such as [4, 9], which analyze the data plane snapshots (i.e., FIB) of networks and enable efficient verification of overlay/underlay networks. However, they can only find errors after undesirable forwarding has occurred. Another category is control plane verifiers like [1, 5, 6, 14, 15], which take

network configurations (i.e., control plane) as input and analyze the resulting data plane, proactively preventing errors before deploying buggy configurations. However, all of them focus on the underlay network and cannot be easily extended to the overlay network due to its unique characteristics:

(1) The sheer size of virtual networks, combined with the frequent updates in overlay configurations in practice, implies a high complexity for modern verifiers. For example, in our nationwide network FITI [12], there are 4096 virtual networks, hosting over 100K users concurrently, with more than ten overlay configuration updates per day to meet service requirements. However, all modern control plane verifiers work by utilizing a **unified model** to abstract all protocol layers within the network and jointly exploring their behaviors, which becomes increasingly complex with the growth of network layers and size. For example, Minesweeper [5] models the entire network as a SMT expression with thousands of variables and solve it collectively. Furthermore, changes in the configuration of a single protocol layer require modern verifiers re-modeling and re-inferring all protocol layers, leading to practical scalability issues.

(2) The complex encapsulation and propagation operations of packets across multiple protocol layers may reduce the scalability and interpretability of modern verifiers. For example, in Figure 1(a), when verifying the reachability from VM1 to VM3, packets will be encapsulated twice by VXLAN and SRv6 in modern verifiers. Additionally, the same encapsulation and propagation of encapsulated packets will be redundantly carried out when it comes to VM2 and VM4. Furthermore, consider the error in Figure 1(a) where the misconfiguration of the SRv6 tunnel at R3 leads to the failed establishment of the BGP EVPN neighbor between R1 and R2, thus resulting in the unsuccessful propagation of EVPN routes and unreachable between VM1 and VM3. However, due to the joint exploration in **unified model**, modern verifiers would only return the result "VRF1 has no route to VM3", providing little help for operators to locate exact errors.

Hence, we propose MEV, which introduces a new **ensemble model** that decomposes the entire network into multiple independent layers. As shown in Figure 1, MEV first collects network configurations, and then constructs specific and independent models for each protocol layer, conducting route computation within each single layer scalably. Finally, MEV performs verification in each encapsulation protocol layer independently, ensuring high scalability and interpretability.

2 OUR APPROACH

During the modeling and verification, we primarily encounter two challenges: (1) How to decouple the complex interactions between protocols into each single layer, i.e., the successful establishment

Yahui Li is the corresponding author.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

ACM SIGCOMM Posters and Demos '24, August 4–8, 2024, Sydney, NSW, Australia

© 2024 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-0717-9/24/08.

<https://doi.org/10.1145/3672202.3673722>

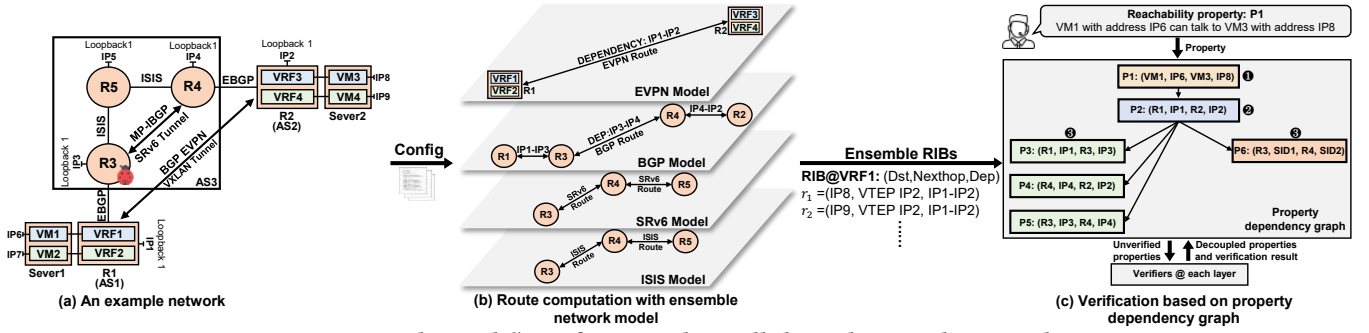


Figure 1: The workflow of MEV with a walkthrough example network

of the iBGP neighbor between R3 and R4 relies on the reachability of IP3 and IP4 provided by ISIS; (2) How to decouple the property requiring verification across multiple layers into dependent sub-properties within each single encapsulation protocol layer, i.e., the packets for verifying reachability between VM1 and VM3 will be encapsulated into VXLAN and SRv6 tunnels and propagated within different layers. We address these challenges by the following two designs: *dependency topology* and *property dependency graph* (PDG).

2.1 Layer Model with Dependency Topology

The complex interactions between different layers in the control plane mainly fall into two categories: route redistribution and route propagation dependency. (1) Route redistribution refers that routes in certain protocol layer may be imported from or exported to another layer. To this end, we reserve redistribution interfaces for each protocol layer model to maintain redistributed routes from other layers. (2) Route propagation dependency implies that routing at a certain protocol layer may require communication support from other layers as the example in challenge 1. We address this problem based on a fundamental observation: routing protocols are generally designed to operate on their own graphs, the successful construction of which relies on other layers. For example, the goal of R4 in BGP layer is to directly propagate BGP routes to its BGP neighbors R2 and R3. Hence, we construct a *dependency topology* for each layer based on their configurations: abstracting routing nodes of each protocol layer and adding an edge between nodes if they have adjacency relationship within this layer, which carries the dependency conditions of communication between them. For example, in the EVPN model of Figure 1(b), we construct a direct edge between R1 and R2 according to their EVPN configurations, and add the dependency (reachability of their loopback addresses) to the edge. Then, routing operations can be performed through route propagation in their own *dependency topology*, carrying corresponding dependency conditions, avoiding interactions with other layers. This allows us to construct specific and independent models for each layer, and perform independent route computation under the assumption that dependencies are satisfied.

2.2 Verification Based on PDG

During property verification, RIB entries matched by packets may have dependencies on other layers, i.e., in Figure 1, whether r_1 can reach VRF1 depends on the reachability between IP1 and IP2. Besides, packets may need to be encapsulated and propagated with new headers in different layers as the example in challenge

2. Hence, the sub-properties decoupled from property to be verified are mainly from two categories: (i) dependencies of matched routing entries in this layer. (ii) dependencies of the propagation of encapsulated packets within other layers. To this end, we utilize *property dependency graph* to reveal these relationships:

First, we identify the property requiring verification and add it as a node to the PDG. During verification of it, when a packet needs to be encapsulated into a tunnel and propagated within another layer, we directly transmit the packet to the tunnel endpoint (i.e., vtep) and add a node corresponding to sub-property from category ii in the PDG. For example, in Figure 1(c), when verifying property P1, we directly transmit the packet with destination IP8 from R1 to R2, and add node P2 for the verification of the packet which is encapsulated with destination address IP2 by VXLAN. Besides, we also add nodes corresponding to sub-properties from category i to the PDG, if the routing entries matched by packets have dependencies on other layers, i.e., P3. Finally, we repeat this process until all properties have been successfully verified.

In this way, we achieve the following benefits in scalability and interpretability: (1) Avoiding complex encapsulation operations and redundant property verifications, i.e., when there is a new reachability property from VM2 to VM4 requiring verification, due to the same decoupling operations as P1, we just need to get the verification result of P2 from the PDG, avoiding redundant propagation. (2) Providing more precise error localization at the level of minimally violated property. For example, re-considering the error in Figure 1(a), we can ascertain that the unreachability between VM1 and VM3 is due to the violation of P6 which verifies the reachability between SRv6 segment identifiers (SIDs) of R3 and R4.

3 PRELIMINARY RESULTS

We have implemented a prototype of MEV in Java and evaluated it in a synthetic network whose topology is from the Topology Zoo [8]. The network runs BGP and EVPN, and consists of 174 nodes and 50 VXLAN tunnels. MEV finishes the verification in 20 seconds, and reduces time consumption by approximately 80% when verifying overlay configuration updates, i.e., adding tenants. In future, we plan to generalize MEV to more protocols and apply it to the NOC of FITI [12].

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation of China (No. 62102020, 62394322).

REFERENCES

- [1] Anubhavnidhi Abhashkumar, Aaron Gember-Jacobson, and Aditya Akella. 2020. Tiramisu: Fast multilayer network verification. In *17th USENIX Symposium on Networked Systems Design and Implementation (NSDI 20)*. 201–219.
- [2] Amazon. [n. d.]. Amazon EC2 secure and resizable compute capacity for virtually any workload. <https://aws.amazon.com/cn/ec2/>. ([n. d.]).
- [3] Tim Anderson. 2021. OVH blames hour-long global outage on human error during 'routine' network reconfiguration. https://www.theregister.com/2021/10/13/ovh_outage/. (2021).
- [4] Ryan Beckett and Aarti Gupta. 2022. Katra: Realtime verification for multi-layer networks. In *19th USENIX Symposium on Networked Systems Design and Implementation (NSDI 22)*. 617–634.
- [5] Ryan Beckett, Aarti Gupta, Ratul Mahajan, and David Walker. 2017. A general approach to network configuration verification. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*. 155–168.
- [6] Ari Fogel, Stanley Fung, Luis Pedrosa, Meg Walraed-Sullivan, Ramesh Govindan, Ratul Mahajan, and Todd Millstein. 2015. A general approach to network configuration analysis. In *12th USENIX Symposium on Networked Systems Design and Implementation (NSDI 15)*. 469–483.
- [7] Google. [n. d.]. Google cloud: Cloud computing services. <https://cloud.google.com/>. ([n. d.]).
- [8] Simon Knight, Hung X. Nguyen, Nickolas Falkner, Rhys Bowden, and Matthew Roughan. 2011. The Internet Topology Zoo. *IEEE Journal on Selected Areas in Communications* 29, 9 (2011), 1765–1775. <https://doi.org/10.1109/JSAC.2011.111002>
- [9] Xu Liu, Peng Zhang, Hao Li, and Wenbing Sun. 2023. Modular Data Plane Verification for Compositional Networks. *Proceedings of the ACM on Networking* 1, CoNEXT3 (2023), 1–22.
- [10] Mallik Mahalingam, Dinesh Dutt, Kenneth Duda, Puneet Agarwal, Lawrence Kreeger, T Sridhar, Mike Bursell, and Chris Wright. 2014. *Virtual extensible local area network (vxlan): A framework for overlaying virtualized layer 2 networks over layer 3 networks*. Technical Report.
- [11] C Perkins. 1996. RFC2003: IP Encapsulation within IP. (1996).
- [12] Global Times. 2023. World's first 1.2TB ultra-high-speed backbone network opens in China. <https://www.globaltimes.cn/page/202311/1301765.shtml>. (2023).
- [13] Jeremy Hartman Tom Strickx. 2022. Configuration Glitch Caused Microsoft's Jan. 25 Exchange Online Disruption. <https://blog.cloudflare.com/cloudflare-outage-on-june-21-2022>. (2022).
- [14] Fangdan Ye, Da Yu, Ennan Zhai, Hongqiang Harry Liu, Bingchuan Tian, Qiaobo Ye, Chunsheng Wang, Xin Wu, Tianchen Guo, Cheng Jin, et al. 2020. Accuracy, scalability, coverage: A practical configuration verifier on a global wan. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*. 599–614.
- [15] Peng Zhang, Dan Wang, and Aaron Gember-Jacobson. 2022. Symbolic router execution. In *Proceedings of the ACM SIGCOMM 2022 Conference*. 336–349.