DEMO: CRC4EVER: Cyclic Redundancy Check for Enhanced Verification and Efficient Routing

Everson Borges
UFES

Fabricio Rodriguez
Telefonica Research

Rafael Silva Guimarães
IFES

Magnos Martinello
UFES

Cristina K. Dominicini
IFES

Moises R. N. Ribeiro
UFES

Eduard Marin
Telefonica Research

Christian Rothenberg
Unicamp

ABSTRACT

This demonstration introduces Cyclic Redundancy Check for Enhanced Verification and Efficient Routing (CRC4EVER). We propose how the Residue Number System (RNS) — a number system that employs a shared secret scheme distributed across network nodes (nodeIDs)— can enable lightweight forwarding and proof-of-transit (PoT) in path-aware networks, relying solely on CRC operations. Our approach leverages an RNS-based source routing, where a routeID encodes the entire packet path. At each hop, the routeID is decoded via simple modulo operations, executed at line rate, by repurposing existing CRC hardware in programmable switches. Furthermore, the unique mapping between the routeID and its corresponding set of nodeIDs provides intrinsic path verifiability via CRC-based hash operations. A proof-of-concept was implemented on programmable Tofino switches, demonstrating the feasibility of executing both routing and path verification at line rate, through table-free CRC operations.

CCS CONCEPTS

• **Networks** → **Programmable networks**; **Protocol testing and verification**; **Network protocol design**;

ACM Reference Format:
Everson Borges, Fabricio Rodriguez, Rafael Silva Guimarães, Magnos Martinello, Cristina K. Dominicini, Moises R. N. Ribeiro, Eduard Marin, and Christian Rothenberg. 2025. DEMO: CRC4EVER: Cyclic Redundancy Check for Enhanced Verification and Efficient Routing . In *ACM SIGCOMM 2025 Conference (SIGCOMM Posters and Demos '25), September 8–11, 2025, Coimbra, Portugal.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3744969.3748446

1 INTRODUCTION

Path-Aware Networking (PAN) emerges as a paradigm that shifts the network control to end hosts. By empowering end hosts with the ability to select optimal paths based on application requirements and current network conditions, PAN creates valuable opportunities in application-driven traffic engineering. It enhances both reliability and efficient network resource utilization at the host level [4].

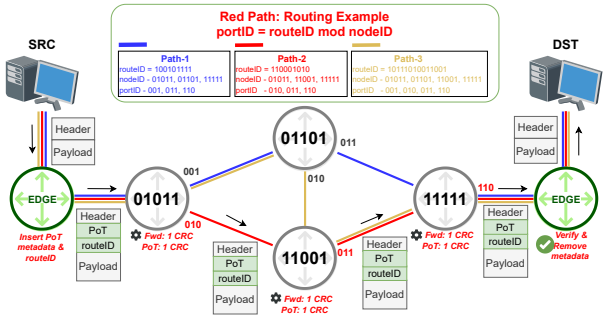

Figure 1: Conceptual Design of CRC4EVER

Despite the PAN boosting capabilities, its practical deployment still faces significant obstacles. Most existing methods for path verification such as ICING [6] and EPIC [5], are not compatible with programmable switches and typically impose high computational overhead on source or core nodes. These approaches often rely on chained MAC [3], or nested structures for path verification, but they deal with routing and verification as separate processes. In contrast, CRC4EVER has been designed for tightly integrating routing with path verification, keeping compatibility with programmable switches, resulting in a unified and efficient architectural design for modern networks.

This demo showcases CRC4EVER, a new way to design networking functions by using the mathematical properties of the Residue Number System (RNS) implemented by existing CRC mechanisms [2]. RNS operates as a secret-sharing scheme that can be distributed across network nodes (*nodeIDs*). We implement a source routing mechanism in which a *routeID* encodes the entire packet path. At each hop, this *routeID* is decoded using modulo operations, executed at line rate. The unique mapping between a *routeID* and its corresponding set of (*nodeIDs*) enables intrinsic path verifiability through RNS. By chaining CRC-based hash operations, we can create a path signature that supports proof-of-transit (PoT), confirming the packet followed the intended path[1].

While MPLS-style label stacks offer similar forwarding behavior without the complexity of modulo-based decoding, their flat labels lack inherent path semantics. In contrast, CRC4EVER employs a

[1]https://github.com/nerds-ufes/CRC4EVER

SIGCOMM Posters and Demos '25, September 8–11, 2025, Coimbra, Portugal

ACM ISBN 979-8-4007-2026-0/2025/09…$15.00
https://doi.org/10.1145/3744969.3748446

route identifier that represents a unique path, meaning that all switches and their interfaces along the path are explicitly known, enabling additional features such as path verification.

To demonstrate the CRC4EVER approach, we deployed it on a single Tofino switch configured with multiple logical switches, and emulated it using the P7 testbed [7] enhanced with PINT-BoX [1]. P7 with PINT-BoX enables dynamic path control, runtime reconfiguration, and real-time observation of latency, loss, and throughput. CRC4EVER distinguishes itself by introducing a novel forwarding and path verification mechanism that relies exclusively on CRC operations, enabling lightweight, table-free path verification at line rate. This setup allows users to trace packets hop-by-hop across logical paths, verify the selected route, and gain intuitive insight into the path-aware behavior of CRC4EVER.

2 ARCHITECTURE & DEMO

The CRC4EVER architecture is specifically designed for path verification and routing using a unique encoding scheme. It operates within programmable switches, leveraging CRC-based operations at its core. Figure 1 presents the conceptual design, illustrating the step-by-step packet flow through a sequence of network nodes in a path-aware network.

At the source node, a Type of Service(ToS) field is used to map the flow to its corresponding *routeID* and *PoT* metadata inserted into the packet header at the edge. At each hop, CRC operations are employed both for packet forwarding and for updating the PoT. Packet forwarding follows the PolKA technique [2], which enables commodity switches to compute polynomial modulo operations using two SHIFT, one CRC, and two XOR operations. For example, the blue path at the first node, the *routeID=100101111* mod *nodeID=01011* gives *portID=001*, following the computation steps below:

(1) $G = \texttt{nodeID} = 01011$, so $r = \deg(G) = 3$
(2) $D = \texttt{routeID} \div 2^r = 100101111 \gg 3 = 100101$ ***(SHIFT RIGHT)***
(3) $\texttt{dif} = \texttt{routeID} - D \cdot 2^r = 100101111 \oplus (100101 \ll 3)$
$= 100101111 \oplus 100101000 = 111$ ***(SHIFT LEFT, XOR)***
(4) $R = \langle D \cdot 2^r \rangle_G = \langle 100101000 \rangle_{01011} = 110$ ***(CRC)***
(5) $\texttt{portID} = \texttt{dif} \oplus R = 111 \oplus 110 = \boxed{001}$ ***(XOR)***

For the PoT, a unique mapping between the routeID and the sequence of node identifiers (nodeIDs) generates a path signature, as shown in Equation 1. This mechanism enables intrinsic path verification by supporting end-to-end PoT through chained CRC computations. At the egress edge, the path verification process validates the PoT metadata.

$$PoT_i = CRC(nodeid(i) \,||\, portid(i) \,||\, PoT_{i-1}) \quad (1)$$

Figure 2 illustrates a PoC deployment of CRC4EVER on a single core switch, demonstrating how multiple logical switches can coexist through pipeline isolation. The design splits processing into two separate pipelines: one for packet forwarding and another for PoT computation. The packet header is inserted at the ingress edge (step 1). Packets enter through the ingress pipeline, where the *routeID* and PoT header are processed. A CRC8 operation is used to decode the *routeID* by computing its modulo with the *nodeID* (step 2), determining the *portID* to the next logical switch. The packet is then recirculated and re-enters through the ingress pipeline, (step 3). In the verification pipeline, a CRC32 operation updates the PoT

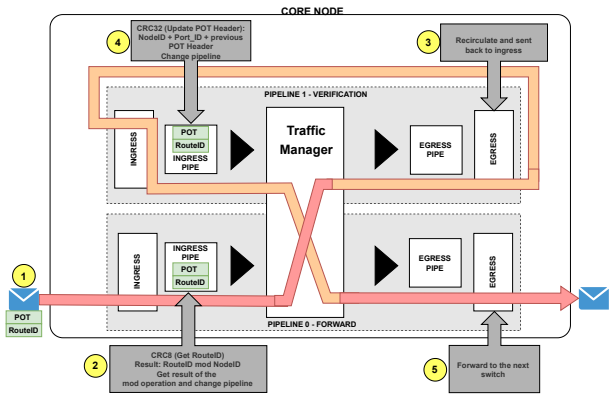

Figure 2: CRC4EVER deployment with multiple logical switches in a single Tofino box

metadata header using the Equation 1 *nodeID*, *portID*, and the previous PoT value (step 4). Finally, the packet is forwarded to the next logical switch (step 5). This architecture enables accurate forwarding and path verification entirely within a single Tofino 1 device, using the modularity of CRC operations.

During the demo: To validate the implementation and create a realistic test environment, we will use P7 with PINT-BoX, which emulates a network with link-metric instrumentation and P4 support. P7 lets users define custom topologies and—even better—modify the environment at run time. For the demonstration, attendees will start from a predefined topology and select one of three available paths. Demo attendees will be able to watch CRC-based forwarding in action and inspect the switch-by-switch PoT. Real-time packet-level visualization at each switch will let participants confirm the effectiveness of CRC4EVER on the fly. Finally, we will showcase P7's run-time ability to reconfigure node IDs and link metrics while the experiment is running.

3 CONCLUSIONS AND FUTURE WORK

The CRC4EVER demo showcases an RNS-based routing and PoT mechanism using CRC operations, enabling table-free, line-rate path processing and verification. The PoC on Tofino switches confirms its feasibility design and efficiency for path-aware networking.

Future work. We envision designing network protection mechanisms against link failures by enabling packet deflections using additional nodes encoded in the routeID. We also intend to carry out a comprehensive security analysis to identify and understand potential attack vectors.

ACKNOWLEDGEMENTS

Porvir-5G Research Project (Grant 20/05182-3), and Fapes (941/2022, 732/2024). Also, this work was supported by Ericsson Telecomunicações Ltda and by the Sao Paulo Research Foundation (FAPESP), grant 2021/00199-8, CPE SMARTNESS. This study was partially funded by *CAPES*, Brazil - Finance Code 001 and CNPq fellow (Grant 312058/2023-3). This work has been partially supported by the European Union's Horizon Europe project under grant agreement No. 101070473 (FLUIDOS).

REFERENCES

[1] Everson S. Borges et al. 2024. PINT-BoX: Path-aware networking IN a Tofino BoX. In *2024 IEEE NFV-SDN*. 1–2. https://doi.org/10.1109/NFV-SDN61811.2024.10978976

[2] Cristina Dominicini et al. 2020. PolKA: Polynomial Key-based Architecture for Source Routing in Network Fabrics. In *NetSoft*. 326–334. https://doi.org/10.1109/NetSoft48620.2020.9165501

[3] Jonathan Katz and Andrew Y. Lindell. 2008. Aggregate Message Authentication Codes. In *Topics in Cryptology – CT-RSA 2008*, Tal Malkin (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 155–169.

[4] Thorben Krüger and David Hausheer. 2021. Towards an api for the path-aware internet. In *Proceedings of the ACM SIGCOMM 2021 Workshop on Network-Application Integration*. 68–72.

[5] Markus Legner et al. 2020. EPIC: every packet is checked in the data plane of a path-aware internet. In *Proceedings of the 29th USENIX Conference on Security Symposium (SEC'20)*. USENIX, USA, Article 31, 18 pages.

[6] Jad Naous et al. 2011. Verifying and enforcing network paths with ICING. In *Proceedings of the Seventh CoNEXT*. 1–12.

[7] Fabricio Rodriguez et al. 2022. P4 programmable patch panel (P7): an instant 100g emulated network on your tofino-based pizza box. In *Proceedings of the SIGCOMM '22 Poster and Demo Sessions (SIGCOMM '22)*. ACM, NY, USA, 4–6. https://doi.org/10.1145/3546037.3546046