Towards Effective yet Frugal Blockchain Network Emulation

Vincenzo Di Perna
University of Urbino
Urbino, Italy
vincenzo.diperna@unicam.it

ABSTRACT

Benchmarking blockchain systems is still a persistent issue. The Diablo benchmark suite allows strategically stressing and testing blockchain solutions thanks to parallel requests to their nodes network. Here, the efficiency of network infrastructure plays a paramount role. Kollaps, a research testbed, emulates network architecture, delivering accuracy levels on par with physical configurations. This research explores integrating Kollaps and Diablo to streamline the benchmarking process while reducing implementation costs through a cluster-based environment.

CCS CONCEPTS

• **Networks** → **Network simulations**; **Network measurement**; **Network performance analysis**; • **Computing methodologies** → *Distributed computing methodologies.*

KEYWORDS

blockchain, benchmark, network emulation

ACM Reference Format:
Vincenzo Di Perna. 2023. Towards Effective yet Frugal Blockchain Network Emulation. In *24th International Middleware Conference Demos, Posters and Doctoral Symposium (Middleware Demos, Posters and Doctoral Symposium '23), December 11–15, 2023, Bologna, Italy*. ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3626564.3629093

1 CONTEXT AND PROBLEM STATEMENT

The proliferation of blockchain infrastructures allows flexibility in choosing the most suitable one for specific implementation goals. However, benchmarking these technologies remains a resource-intensive challenge, lacking a well-defined framework, as seen in the existing literature.

In [4], Diablo emerges as a noteworthy framework that facilitates testing on blockchain platforms by efficiently distributing predefined workloads among multiple clients that will issue requests to nodes within the target blockchain network. Testing concludes with data collection for a thorough behavior analysis, where data veracity is upheld through real network infrastructures deployed on distributed machines spanning multiple regions via Amazon Web Services.

In considering the importance of implementation and costs of a network infrastructure, Kollaps figures as a complete network emulator executed across clusters and container technologies [3], delivering outcomes comparable to real-world physical nodes, offering superior accuracy compared to existing frameworks.

This paper aims to harness Diablo's benchmarking capabilities while concurrently mitigating the associated network implementation costs, and yielding benchmarking results on par with real-node scenarios (e.g., AWS). This endeavor seeks to replicate the tests conducted in [4], scrutinizing deviation percentages, all within the confines of a cluster-based environment.

Middleware Demos, Posters and Doctoral Symposium '23, December 11–15, 2023, Bologna, Italy

ACM ISBN 979-8-4007-0429-1/23/12.
https://doi.org/10.1145/3626564.3629093

2 BACKGROUND

Disparities between announced and observed blockchain performance raise questions about the (isolated) testing conditions. For instance, while Solana claims 250,000 TPS (Transactions Per Second) and sub-second finality [5], testing in [4] yielded much lower results – approximately 8,845 TPS with latency exceeding 12 seconds.

Diablo [4] is a versatile toolset for assessing diverse blockchain systems like Algorand, Ethereum, and Solana. It supports customizable workloads, from emulating exchange transactions to sustaining transaction throughputs, and facilitates replicating experiments across various blockchains based on WebSocket RPC interfaces. Diablo's strength lies in its enhanced distributed workload generation mechanism split on secondary element(s), orchestrated by a primary element, ensuring synchronized evaluations. The benchmark definition process involves parsing both benchmark and blockchain configuration files. Multiple configurations, employing up to 200 machines distributed globally, allow Diablo to assess blockchain systems' performance under conditions akin to those experienced by the Visa payment system [4].

At the same time, assessing network emulations involves considering execution location and node orchestration. Kollaps is a decentralized network emulator that aims to redefine large-scale application testing [3] by prioritizing end-to-end properties (e.g., latency, jitter, bandwidth, packet loss) while avoiding complex network element emulation. Its fully distributed model ensures scalability and agility in simulating dynamic events like link removals. Its main components include the Traffic Control Abstraction Layer (TCAL) and Deployment Generator that integrates seamlessly with container images and orchestrators like Docker Swarm and Kubernetes, ensuring consistent performance and cost-effectiveness in cluster deployments. Also, it matches full network state emulators like Mininet in accuracy [3].

3 PROPOSED APPROACH

Kollaps uses an xml file to define the desired network structure for emulation, including the dashboard, services within containers with execution instructions, links with attributes, and dynamic events. The goal is to integrate Diablo's primary and secondary components

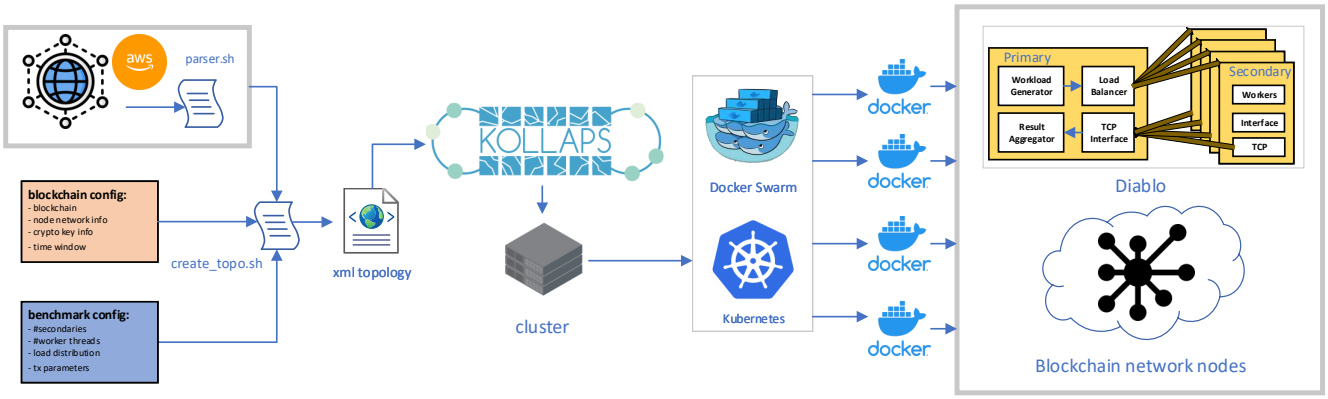

Figure 1: Workflow of the benchmarking process

as services and configure Kollaps to accommodate the blockchain-specific node network. A script processes a benchmark configuration file and a blockchain configuration file to yield an xml file that delineates the network configuration for the experiment. We want to enable Kollaps to faithfully replicate the network's characteristics by manipulating configurable attributes such as latency, packet loss, and upload and download speeds. Since obtaining data at hourly intervals becomes feasible, thereby capturing behavioral patterns, a secondary script may extract network properties associated with physical nodes dispersed across diverse regions (e.g., AWS), incorporating dynamic events into the xml file that faithfully mirror real-world occurrences.

Figure 1 depicts the structure of the benchmarking process streamflow. This approach enables Kollaps to efficiently deploy application containers and blockchain network nodes across cluster machines.

4 DISCUSSION AND CHALLENGES

The awareness that other tools, such as SimGrid [1], can provide additional assessments and that there are either no established standards for blockchain benchmarking or specific tool prevalence is well known. However, integrating Kollaps with Diablo will allow to emulate a network of nodes behaving like a real blockchain, potentially adaptable for testing over specific periods. In addition to reusing complex software tools like Kollaps and Diablo, we must address several other tasks such as recreating the same stress conditions and utilizing the same dataset while accommodating specific or dynamic events.

We will employ a Python script designed to generate a network graph aligned with Kollaps'xml file schema by extracting network information from the following public AWS datasets: *(i) iperf3* measurements from [4] and *(ii)* (RIPE Atlas) *ping* and *traceroute* measurements from [2]. Subsequently, another script will be utilized to merge the network data with the configuration files, resulting in the generation of the xml file intended for use with Kollaps to execute Diablo benchmarks.

We also have to implement and adapt the specific blockchain to the interface for integration with Diablo. Lastly, our tasks involve exporting and processing the results and conducting a comparative analysis involving Diablo with both AWS and Kollaps.

5 CONCLUSION AND FUTURE WORK

This initial work establishes the groundwork for the proposed solution, enabling a comparative analysis of Diablo and Kollaps implementations. The main objective is to evaluate network performance on physical nodes and explore the potential for equivalence with Kollaps and a cost-effective cluster-based network.

Even though the experiment was conducted using synthetic blockchain workloads, the plan for future tests is to collect and utilize real blockchain data from specific periods. While AWS data has already been obtained from measurements on real nodes, the goal is to be able to use temporal data to simulate dynamic network events (in Kollaps), including node failures, upload speed variations, and packet drops.

ACKNOWLEDGMENTS

The author is a student of the national PhD program in Blockchain and Distributed Ledger Technology (University of Camerino, Italy) supported by a PNRR scholarship ex D.M. 351/2022 and the PRIN 2020 project NiRvAna (University of Urbino, Italy). He is supervised by Prof. Marco Bernardo (University of Urbino, Italy) and Prof. Francesco Fabris (University of Trieste, Italy). This research is based on collaborations with Prof. Valerio Schiavoni (University of Neuchâtel, Switzerland) and the development teams of *Kollaps* and *Diablo*.

REFERENCES

[1] H. Casanova, A. Giersch, A. Legrand, M. Quinson, and F. Suter. 2013. SimGrid: A Sustained Effort for the Versatile Simulation of Large Scale Distributed Systems. arXiv:cs.DC/1309.1630

[2] L. Corneo, M. Eder, N. Mohan, A. Zavodovski, S. Bayhan, W. Wong, P. Gunningberg, J. Kangasharju, and J. Ott. 2021. Surrounded by the Clouds: A Comprehensive Cloud Reachability Study. In *Proceedings of the Web Conference 2021 (WWW '21)*. Association for Computing Machinery, New York, NY, USA, 295–304. https://doi.org/10.1145/3442381.3449854

[3] P. Gouveia, J. Neves, C. Segarra, L. Liechti, S. Issa, V. Schiavoni, and M. Matos. 2020. Kollaps: Decentralized and Dynamic Topology Emulation. In *Proceedings of the Fifteenth European Conference on Computer Systems (EuroSys '20)*. Association for Computing Machinery, New York, NY, USA, Article 23, 16 pages. https://doi.org/10.1145/3342195.3387540

[4] V. Gramoli, R. Guerraoui, A. Lebedev, C. Natoli, and G. Voron. 2022. Diablo-v2: A Benchmark for Blockchain Systems. (2022), 14. http://infoscience.epfl.ch/record/294268

[5] A. Yakovenko. 2021. Solana: A New Architecture for a High Performance Blockchain v0.8.13. https://solana.com/solana-whitepaper.pdf Accessed: 2023-09-26.