HyperProv: Decentralized Resilient Data Provenance at the Edge with Blockchains

Petter Tunstad
Department of Computer Science
UiT The Arctic University of Norway
Tromsø, Norway
pettertunstad@gmail.com

Amin M. Khan
Department of Computer Science
UiT The Arctic University of Norway
Tromsø, Norway
amin.khan@uit.no

Phuong Hoai Ha
Department of Computer Science
UiT The Arctic University of Norway
Tromsø, Norway
phuong.hoai.ha@uit.no

Abstract

Data provenance and lineage are critical for ensuring integrity and reproducibility of information in research and application. This is particularly challenging for distributed scenarios, where data may be originating from decentralized sources without any central control by a single trusted entity. We present HyperProv, a general framework for data provenance based on the permissioned blockchain Hyperledger Fabric (HLF), and to the best of our knowledge, the first system that is ported to ARM based devices such as Raspberry Pi (RPi). HyperProv tracks the metadata, operation history and data lineage through a set of built-in queries using smart contracts, enabling lightweight retrieval of provenance data. HyperProv provides convenient integration through a NodeJS client library, and also includes off-chain storage through the SSH file system. We evaluate HyperProv's performance, throughput, resource consumption, and energy efficiency on x86-64 machines, as well as on RPi devices for IoT use cases at the edge.

CCS Concepts. • **Computer systems organization → Distributed architectures**.
Keywords. Blockchain, Data Provenance, Edge Computing

1 Introduction

Over the last decades, the size of data utilized in research has increased significantly, highlighting the importance of data provenance systems [7, 11, 15] in order to ensure the quality and integrity of the information, and to counteract accidental or malicious data manipulation and corruption. We present HyperProv, a permissioned blockchain based provenance system using Hyperledger Fabric (HLF) [1] to provide guarantees for provenance and lineage of data by storing the provenance metadata in a tamper-proof ledger. We consider the use case of Internet of Things (IoT) data at the edge to demonstrate utility and applicability of HyperProv. We evaluate in detail [17**?**] throughput, resource consumption and energy efficiency of HyperProv both on x86-64 commodity hardware and Raspberry Pi (RPi) ARM64 devices, in order to compare its performance with other recent blockchain-based systems [9, 13].

To the best of our knowledge, we are the first to run a provenance system featuring HLF's first long term release, and are also the first one to run provenance system based on HLF on ARM devices. Our contributions in porting HLF for ARM devices have already generated significant uptake and recognition in the community, with more than 500 downloads in the first 2-3 months[1,2]. We believe HyperProv demonstrates the feasibility of using edge devices such as RPi for data provenance through blockchains. We hope that by building and releasing Docker images for ARM, we can pave way for other innovative solutions employing HLF for edge computing. HyperProv's NodeJS client library hides away the complexity of working with HLF, allowing for easily plugging in HyperProv with other domain-specific data provenance systems.

2 Related Work

Many projects have tackled domain-specific as well as general purpose data provenance, such as PAAS [11], Chimera [4], MyGrid [18], CMCS [12], ESSW [5], and Trio [2], among others [7, 15]. We limit our focus here on recent data provenance systems that employ blockchains [3, 9, 13], and HyperProv is distinguished from them as it uses permissioned blockchains like HLF that have much less resource requirements compared to public blockchains [16], limits recording only provenance metadata in the blockchain while moving actual data to off-chain storage, and demonstrates practical applications of using devices such as RPi at the edge without relying on constant connectivity to the cloud. With respect to HyperProv's implementation, other recent works not focused on provenance but related to HLF and RPi are also of interest to us. Vegvisir [8] handles network partitions, which is critical to successful deployment in IoT and edge scenarios, FastFabric [6] improves on the throughput of HLF, while [14] demonstrates feasibility and utility of HLF for edge scenarios like wireless mesh networks. We leave the detailed comparison to the technical report [17**?**].

3 HyperProv Data Provenance System

HyperProv follows the features from the Open Provenance Model [10], while still supporting extensions to support

[1] https://github.com/Tunstad/Hyperprov
[2] https://hub.docker.com/u/ptunstad

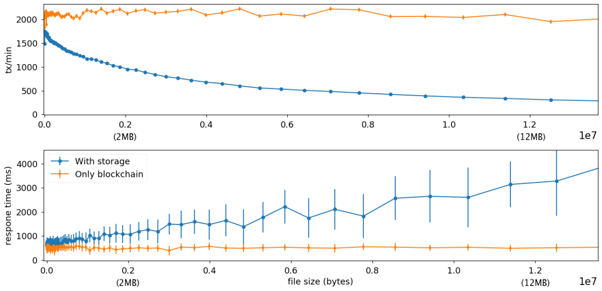

Figure 1. Throughput and response times for desktop

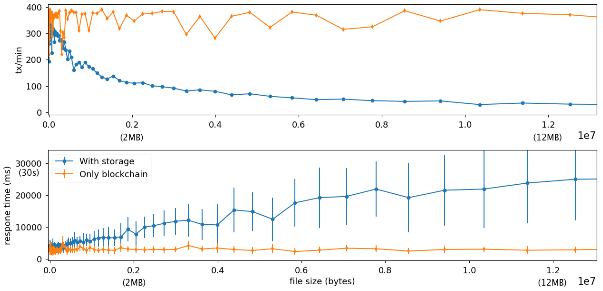

Figure 2. Throughput and response times for RPi

domain-specific provenance metadadata. To this end, HyperProv stores the checksum, editors, operations, data ownership, and data pointers in the blockchain, while delegating the storage of actual data to a separate *off-chain* storage system. HyperProv consists of HLF-framework running in Docker containers and the NodeJS based client library, and is supplemented by an off-chain storage system built using SSHFS. The goal of the system is to enable seamless storage of provenance metadata in a tamper-proof blockchain framework when accessing and storing data in a *pluggable* storage service. Peer nodes are responsible for hosting the chaincode, which is the executable logic that can append or query data stored in the ledger. The chaincode consists of a few functions that are mirrored and available across all peer nodes. The core data currently stored in the blockchain is the checksum of every data item, the data location, a certificate pertaining to who stored the data, a list of other data items that were used to create an item, and a custom field for any additional metadata. The HyperProv's client library enables the use of the HyperProv system for a wide range of functionality with only a few limited operators, such as `Init`, `Post`,`Get`, `StoreData`, `GetData`, etc., see [17] for the details.

4 Evaluation

We perform extensive evaluation [17] using two different setups of the same network. The first consists of desktop nodes and the other consists of RPi devices. The desktop setup has 4 machines: 2 Intel Xeon E5-1603 CPU 2.80GHz, 1 Intel Core i7-4700MQ CPU 2.40GHz, and 1 Intel Corei3-2310M CPU 2.10GHz. All have Ubuntu 16.04 OS and are equipped with SSD storage. Each of the four nodes run peer docker containers, whereas one Xeon machine runs the orderer, and all use the official HLF docker images. The second setup consists of 4 ARM-based RPi 3B+ 1.4GHz Cortex-A53 devices, interconnected on the same network switch. The RPi runs the unofficial RaspberryPi 3 Debian Buster 64-bit OS, since the newer HLF versions require 64-bit support. Due to the lack of other supported docker images, we have compiled our own images for the ARM64 architecture for RPi. Off-chain storage component based on SSH file system always runs on a separate node. The measurements are performed using our custom benchmarking program. The energy consumption is measured using an ODROID V3 power meter placed between the device and the power source.

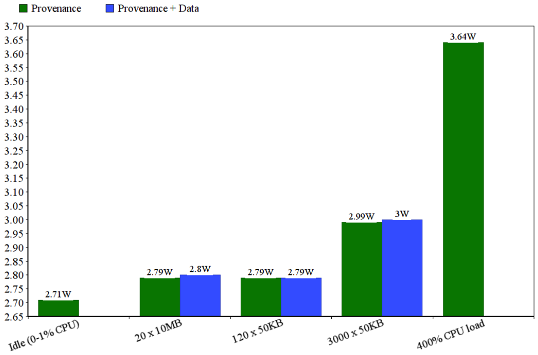

Figure 3. Energy consumption on RPi, 10-minute intervals

Fig. 1 shows how increasing the size of data items impacts both throughput and response times, when off-chain storage is involved for desktop machines which incurs the overhead of data transfer and checksum calculation. Fig. 2 shows similar trend for throughput and response times for RPi though greater variation, however absolute performance for RPi is lower than desktop machines as expected owing to the limited hardware capacity. Measurements of the energy consumption of RPi devices running both peer and client processes for 10 minutes (as shown in Fig. 3) highlight that running HyperProv without any active transactions barely consumes any power (2.71W) compared to an idle RPi running without HLF, while at the peak load level consumes only 10.7% more as compared to idle, and maximum up to 3.64W.

Acknowledgments

This work is funded by the Research Council of Norway under PREAPP project (grant n° 231746) and eX3 project (grant n° 270053).

References

[1] Elli Androulaki, Yacov Manevich, Srinivasan Muralidharan, Chet Murthy, Binh Nguyen, Manish Sethi, Gari Singh, Keith Smith, Alessandro Sorniotti, Chrysoula Stathakopoulou, Marko Vukolić, Artem Barger, Sharon Weed Cocco, Jason Yellick, Vita Bortnikov, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Christopher Ferris, and Gennady Laventman. 2018. Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains. In *EuroSys '18*. https://doi.org/10.1145/3190508.3190538

[2] Y. Cui and J. Widom. 2000. Practical lineage tracing in data warehouses. In *Proceedings of 16th International Conference on Data Engineering (Cat. No.00CB37073)*. 367–378. https://doi.org/10.1109/ICDE.2000.839437

[3] Andrey Demichev, Alexander Kryukov, and Nikolai Prikhodko. 2018. The Approach to Managing Provenance Metadata and Data Access Rights in Distributed Storage Using the Hyperledger Blockchain Platform. *CoRR* abs/1811.12706 (2018).

[4] I. Foster, J. Vockler, M. Wilde, and Yong Zhao. 2002. Chimera: a virtual data system for representing, querying, and automating data derivation. In *Proceedings 14th International Conference on Scientific and Statistical Database Management*. 37–46. https://doi.org/10.1109/SSDM.2002.1029704

[5] J. Frew and R. Bose. 2001. Earth System Science Workbench: a data management infrastructure for earth science products. In *Proceedings Thirteenth International Conference on Scientific and Statistical Database Management. SSDBM 2001*. 180–189. https://doi.org/10.1109/SSDM.2001.938550

[6] Christian Gorenflo, Stephen Lee, Lukasz Golab, and S. Keshav. 2019. FastFabric: Scaling Hyperledger Fabric to 20,000 Transactions per Second. In *IEEE International Conference on Blockchain and Cryptocurrency (ICBC '19)*. IEEE.

[7] Melanie Herschel, Ralf Diestelkämper, and Houssem Ben Lahmar. 2017. A Survey on Provenance: What for? What Form? What from? *The VLDB Journal* 26, 6 (Dec. 2017), 881–906. https://doi.org/10.1007/s00778-017-0486-1

[8] K. Karlsson, W. Jiang, S. Wicker, D. Adams, E. Ma, R. van Renesse, and H. Weatherspoon. 2018. Vegvisir: A Partition-Tolerant Blockchain for the Internet-of-Things. In *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*. 1150–1158. https://doi.org/10.1109/ICDCS.2018.00114

[9] X. Liang, S. Shetty, D. Tosh, C. Kamhoua, K. Kwiat, and L. Njilla. 2017. ProvChain: A Blockchain-Based Data Provenance Architecture in Cloud Environment with Enhanced Privacy and Availability. In *2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*. 468–477. https://doi.org/10.1109/CCGRID.2017.8

[10] Luc Moreau, Ben Clifford, Juliana Freire, Joe Futrelle, Yolanda Gil, Paul Groth, Natalia Kwasnikowska, Simon Miles, Paolo Missier, Jim Myers, Beth Plale, Yogesh Simmhan, Eric Stephan, and Jan Van den Bussche. 2011. The Open Provenance Model core specification (v1.1). *Future Generation Computer Systems* 27, 6 (2011), 743 – 756. https://doi.org/10.1016/j.future.2010.07.005

[11] Kiran-Kumar Muniswamy-Reddy, Peter Macko, and Margo I Seltzer. 2010. Provenance for the Cloud. In *USENIX FAST'10*.

[12] Carmen Pancerella, John Hewson, Wendy Koegler, David Leahy, Michael Lee, Larry Rahn, Christine Yang, James D. Myers, Brett Didier, Renata McCoy, Karen Schuchardt, Eric Stephan, Theresa Windus, Kaizar Amin, Sandra Bittner, Carina Lansing, Michael Minkoff, Sandeep Nijsure, Gregor von Laszewski, Reinhardt Pinzon, Branko Ruscic, Al Wagner, Baoshan Wang, William Pitz, Yen-Ling Ho, David Montoya, Lili Xu, Thomas C. Allison, William H. Green, Jr., and Michael Frenklach. 2003. Metadata in the Collaboratory for Multi-scale Chemical Science. In *Proceedings of the 2003 International Conference on Dublin Core and Metadata Applications: Supporting Communities of Discourse and Practice—Metadata Research & Applications (DCMI '03)*. Dublin Core Metadata Initiative, Article 13, 9 pages.

[13] Aravind Ramachandran and Murat Kantarcioglu. 2018. SmartProvenance: A Distributed, Blockchain Based DataProvenance System. In *CODASPY '18*. ACM, 35–42. https://doi.org/10.1145/3176258.3176333

[14] Mennan Selimi, Aniruddh Rao Kabbinale, Anwaar Ali, Leandro Navarro, and Arjuna Sathiaseelan. 2018. Towards Blockchain-enabled Wireless Mesh Networks. In *CryBlock '18*. ACM Press, Munich, Germany, 13–18. https://doi.org/10.1145/3211933.3211936

[15] Yogesh L. Simmhan, Beth Plale, and Dennis Gannon. 2005. A Survey of Data Provenance in e-Science. *SIGMOD Rec.* 34, 3 (Sept. 2005), 31–36. https://doi.org/10.1145/1084805.1084812

[16] Parth Thakkar, Senthil Nathan, and Balaji Viswanathan. 2018. Performance Benchmarking and Optimizing Hyperledger Fabric Blockchain Platform. In *MASCOTS '18*. 264–276. https://doi.org/10.1109/MASCOTS.2018.00034

[17] Petter Tunstad. 2019. *Hyperprov: Blockchain-based Data Provenance using Hyperledger Fabric*. Master's thesis. UiT The Arctic University of Norway. https://hdl.handle.net/10037/15780

[18] Jun Zhao, Carole Goble, Robert Stevens, and Sean Bechhofer. 2004. Semantically Linking and Browsing Provenance Logs for E-science. In *Semantics of a Networked World. Semantics for Grid Databases*, Mokrane Bouzeghoub, Carole Goble, Vipul Kashyap, and Stefano Spaccapietra (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 158–176.