A Demonstration of Sterling: A Privacy-Preserving Data Marketplace

Nick Hynes[1, 2] David Dao[2, 3] David Yan[1] Raymond Cheng[1, 2] Dawn Song[1, 2]

[1]Oasis Labs [2]UC Berkeley [3]ETH Zurich

{nhynes, daviddao, david.yan, ryscheng, dawnsong}@oasislabs.com

ABSTRACT

In this work, we demonstrate Sterling, a decentralized marketplace for private data. Sterling enables privacy-preserving distribution and use of data by using *privacy-preserving smart contracts* which run on a permissionless blockchain. The privacy-preserving smart contracts, written by data providers and consumers, immutably and irrevocably represent the interests of their creators. In particular, we provide a mechanism for data providers to control the use of their data through automatic verification of data consumer contracts, allowing providers to express constraints such as pricing and differential privacy. Through smart contracts and trusted execution environments, Sterling enables privacy-preserving analytics and machine learning over private data in an efficient manner. The resulting economy ensures that the interests of all parties are aligned.

For the demonstration, we highlight the use of Sterling for training machine learning models on individuals' health data. In doing so, we showcase novel approaches to automatically appraising training data, verifying and enforcing model privacy properties, and efficiently training private models on the blockchain using trusted hardware.

PVLDB Reference Format:
Nick Hynes, David Dao, David Yan, Raymond Cheng, Dawn Song. A Demonstration of Sterling: A Privacy-Preserving Data Marketplace. *PVLDB*, 11 (12): 2086-2089, 2018.
DOI: https://doi.org/10.14778/3229863.3236266

1. INTRODUCTION

Machine learning (ML) systems benefit from large quantities of diverse training data. Currently, collecting high-quality datasets is challenged by data privacy requirements. In this work, we propose a marketplace in which mutually-distrusting parties can share and use private data without sacrificing privacy.

There have been several attempts at creating distributed AI and data marketplaces for public datasets, some of which are implemented as *smart contracts* on distributed ledgers known as blockchains. Although smart contracts enable reaching consensus on the result of a computation, current mechanisms for verifying correctness requires public disclosure of contract inputs and state. This poses a difficulty for data marketplaces since any user of the blockchain can directly view and copy the data and models. Furthermore, even in the benign case, there is no way to ensure that data are not used in a manner that conflicts with its provider's constraints (e.g., using biometric data to train ad-serving models).

To simultaneously address these issues, we propose Sterling, a data marketplace for private datasets. Our approach combines blockchain smart contracts, trusted execution environments (e.g., Intel SGX [2], Sanctum [13], Keystone [11]), and differential privacy, to offer strong security and privacy guarantees for user data and machine learning models. Smart contracts allow the enforcement of data providers' constraints on how their data is used. For example, they can require analytics performed on their data to be differentially private. Smart contracts also enable users to define payments and rewards. By leveraging privacy-preserving smart contracts running in trusted execution environments, we can compute analytics and train machine learning models while keeping all data and models private. Sterling thus enables mutually distrusting parties to collaboratively train privacy-preserving machine learning models, compensating parties while keeping their data private.

We make the following technical contributions:

1. a framework supporting generic data provider and data consumer smart contracts which uphold their creators' interests;
2. a method to encode and automatically enforce flexible constraints on the use of data,
3. a method for running machine learning pipelines while ensuring privacy of both data and models,
4. a concrete demonstration of the above contributions on the task of medical diagnosis.

2. THE STERLING MARKETPLACE

Consider the motivating example of a medical researcher wishing to train a predictive model of disease. Currently, this would require a lengthy process of negotiating with hospitals for data [18]. Obtaining a truly representative dataset may require collaborations with clinics across the globe. Instead Sterling, a privacy-preserving data marketplace, allows individuals to provide their EHR data for direct use by researchers and organizations. Thus, individuals can realize the economic value of their data without compromising privacy.

Generally, we seek to provide the following workflow (Figure 1):

1. A data provider, U_d, uploads encrypted data to a centralized or decentralized storage service (e.g., AWS, IPFS, Swarm) . U_d publishes a smart contract C_d containing the address of the data and, optionally, constraints like payment or privacy requirements. U_d provisions C_d with a data decryption key which is privately stored by the contract.

This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-nd/4.0/. For any use beyond those covered by this license, obtain permission by emailing info@vldb.org.
Proceedings of the VLDB Endowment, Vol. 11, No. 12
Copyright 2018 VLDB Endowment 2150-8097/18/8.
DOI: https://doi.org/10.14778/3229863.3236266

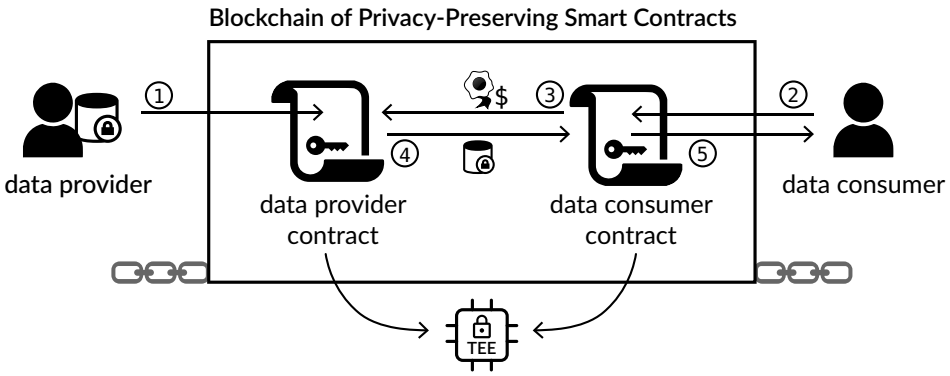

Figure 1: Diagram of the interaction between data producers and consumers in the Sterling marketplace. The economic and privacy interests of each party is mediated and enforced using privacy-preserving smart contracts. The circled numbers refer to steps of the workflow described in Section 2.

2. A data consumer, U_c, desiring to use provided data writes a smart contract C_c which satisfies the constraints of C_d.
3. U_c invokes C_c which sends a signed request, attesting to its identity, to C_d.
4. C_d automatically verifies that C_c satisfies the constraints and securely returns a data decryption key.
5. C_c performs its computation on the decrypted data and U_d is compensated according to the terms of use.

Enabling a secure protocol like the above is challenging. In Sterling, we propose an approach which effectively addresses these challenges. In the remainder of this paper, we describe our approach for automatically enforcing the constraints of data providers (Section 2.1), the resulting data economy (Section 2.2), and our implementation of an end-to-end privacy-preserving machine learning pipeline (Section 2.3). We then describe the demonstration scenario of training a predictive model of disease from electronic health records (Section 3). We conclude with an overview of the system.

2.1 Automatically Enforcing Terms of Use

A primary contribution of the Sterling marketplace is the ability for data providers to impose *terms of use*, or constraints, on the use of their data. In our system, the privacy-preserving smart contracts are programmed in a general-purpose language (e.g., Rust, JavaScript). Thus, data providers can encode flexible requirements within the Sterling framework. Perhaps the simplest term of use is requiring payment for each use of the data. Recalling the motivating example, a more nuanced term might be that a consumer contract bearing the cryptographic signature of a hospital receives the EHR data for free.

By executing the contracts on a blockchain, Sterling ensures correct autonomous execution of smart contracts. Sterling is designed to be compatible with existing blockchains, inheriting the assumptions of the underlying blockchain for achieving availability and integrity. Currently we use the Oasis blockchain platform [16] which extends Ekiden [4] and provides privacy-preserving smart contracts.

2.1.1 Terms of Use for Training ML Models

In our initial system, we focus on the constraints of payment and differential privacy [6] of models trained on the data. In both cases, our approach relies on static analysis to ensure that the data consumer contract satisfies the constraints of data provider contract.

For ensuring differential privacy, we provide functionality for training differentially private ML models like logistic regression and neural networks using stochastic gradient descent [1]. We use techniques from Optio [15] to perform privacy-aware type checking of a consumer contract's model definition, so to ensure that it satisfies differential privacy. We further describe differential privacy constraints in Section 2.3.2.

Since Sterling supports flexible logic (which includes calls to other contracts), a data provider can straightforwardly create additional, custom constraints within the general framework.

2.2 Data Economics

In general, the data economy is governed by the terms of use set by data providers. Since data providers are free to create additional provider contracts, they can re-share data under modified terms of use–for instance, lowering the price to reflect other providers' actions in the marketplace.

A main challenge of working with private data is that the consumer is unable to determine ahead of time that the data are of value. In a benign case, a data provider may simply offer poor documentation. An adversarial provider, however, may attempt to defraud buyers by submitting random noise or even plausible fake data. Thus it would be advantageous–indeed essential–for a data consumer contract to automatically determine the value of the data it receives. Generally, appraising data requires domain specific knowledge of what constitutes *good* data. In the section to follow, we present as examples techniques usable in machine learning applications.

Assuming that data consumers are able to verify the utility of data, the economics of the market ensure that the objectives of providers and consumers are aligned. For example, an adversary might submit fake data with the constraint that payment be made upfront, but no rational data consumer would use the data without first verifying its utility. Conversely, an honest data provider would not want their data to be used without payment, so they might require that a data consumer contract not reveal the results of its computation until a payment is made. Since each party's terms are immutably and irrevocably encoded in a privacy-preserving smart contract, Sterling guarantees that all parties requirements are fulfilled.

2.2.1 Economics of ML Models & Data

For the specific use case of machine learning, we draw on techniques from *data valuation* [10] and adapt them for use on the blockchain. For a given utility function, computing the exact value of data requires training many models on varying subsets of the full dataset. Since the blockchain cost model makes extensive retraining prohibitively expensive, we need to use approximations. Specifically, we use an approximation to *influence functions* which,

themselves, approximate the influence a training point had on a model's prediction [12]. Importantly, computing influence functions does not require re-training the model. As an additional benefit, this technique enables providing fine-grained payments for data: if a fee is charged per prediction, payment can be distributed to the providers whose data were most influential for the prediction.

2.3 Privacy-Preserving Machine Learning

To protect the contents of data and models and ensure their fair use, we must guarantee that the complete machine learning pipeline remains privacy-preserving–from data loading to evaluation of the trained model. To this end, we use the unique combination of trusted execution environments and differential privacy.

2.3.1 Trusted Execution Environments

In Sterling, trusted execution environments (TEEs) can serve as the foundation for secure computation. The ML pipeline begins with the TEE remotely attesting to the veracity of the consumer smart contract. Once verified, the TEE runs the smart contract while keeping the program state safe from external observation or manipulation. The consumer contract is then able to obtain encrypted data via the provider smart contract, decrypt it, and use it to directly update the model parameters, inside the TEE. Even with the overhead of memory encryption and privacy-preserving context switches, this approach is significantly more efficient than direct cryptographic methods like homomorphic encryption or secure multi-party computation. Indeed, machine learning in TEEs has performance comparable to non-private CPU-based training [9].

The TEE threat model does not include side-channel attacks, however. We address this using *data-oblivious* implementations of common training algorithms [17] which do not depend on the values of input data. Moreover, the threat model does not aim to protect the *host* from the computation. For example, since an TEE can directly access host RAM, a malicious smart contract could probe host memory for sensitive information like private keys. To counter such an attack, we sandbox the smart contracts by running them within a WebAssembly interpreter which provides complete memory isolation and limits the resources available to the computation.

Having established a secure way to operate on ML models, we now turn to ensuring that the model does not learn the exact values of the training data.

2.3.2 Differential Privacy

Even if data and model parameters are secured within a TEE, naive implementations of machine learning algorithms can memorize and later reveal training data [3].

Differential privacy (DP), in essence, provides strong theoretical guarantee that the risk to a data provider's privacy is not significantly increased by the use of the data. In other words, applying a DP *mechanism* ensures that the results of analyzing the data are relatively insensitive to the exact values of any particular provider's data. A simple and intuitive DP mechanism is the addition of noise to the model's gradients during training. The trade-off between privacy and precision is controlled by the *privacy budget*. An important element of DP is that making queries of the data (e.g., through model training or inference) "spends" the privacy budget.

The Sterling framework allows the data provider contract to specify the differential privacy parameters as terms of use. We make the novel contribution of an automatic tracker for privacy budget expenditure which does not require trust assumptions (c.f. [14]): the privacy requirements of every consumer request is automatically determined by analyzing its computation graph [15]. When the consumer contract uses the data, the provider smart contract's privacy budget is correspondingly reduced; when the budget reaches zero, the contract ceases to yield data and consumer contracts admit no further queries. Sterling permits an economy to develop around privacy budget by allowing providers to require payment in proportion to privacy usage, perhaps using principles from the literature [8, 7].

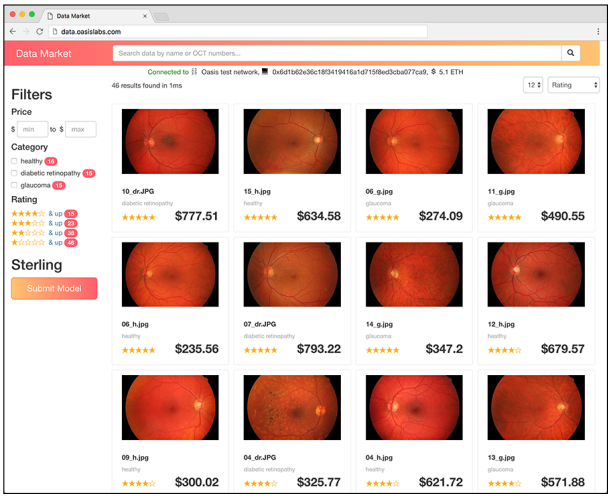

Figure 2: Data providers upload encrypted data and list it on Sterling. Data consumers can browse and purchase those data which satisfy their requirements.

3. DEMONSTRATION

To demonstrate the utility of the Sterling data marketplace, we implement the disease modeling scenario described in the beginning of Section 2. The concrete application is diagnosis of diabetic retinopathy from fundus (back of eye) images. Examples of which are shown in Figure 2).

We simulate multiple data providers by splitting a public dataset of fundus images among several data providers. Each provider contract will offer a randomly sized partition of the data and have its own privacy and payment requirements. Demo participants then assume the role of medical researchers and design consumer smart contracts, through our web interface, which train and evaluate privacy-preserving models on providers' data. As a basis for customization, we provide several examples of models including logistic regression and deep neural networks.

In this setting of medical diagnosis, we provide a walkthrough which highlights the key features of Sterling. Namely:

1. the ability of a data provider to specify a rich set of constraints, like payment and privacy, on privately shared data,
2. the ability of a data consumers to, via a web interface (shown in Figures 2 and 3), browse the marketplace, assemble a custom dataset, and create contracts which satisfy the constraints of all selected providers,
3. efficient, secure training of differentially private ML models, and automatic appraisal of training data and the resulting model.

To yield insight into the otherwise opaque blockchain operations, we develop a blockchain explorer that displays events like pending transactions and model training progress (shown in Figure 4).

Overall, the demonstration offers a preview of a realistic end-to-end workflow for buying and selling data in a privacy-preserving data marketplace.

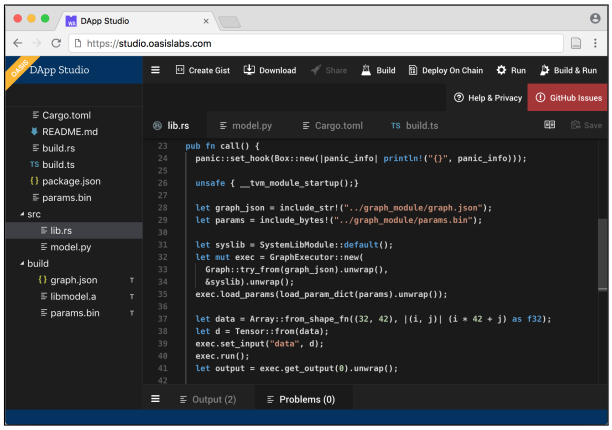

Figure 3: Web IDE for creating and editing smart contracts. From here, data providers can specify precise constraints on the use of their data; and data consumers can design machine learning models which use the data.

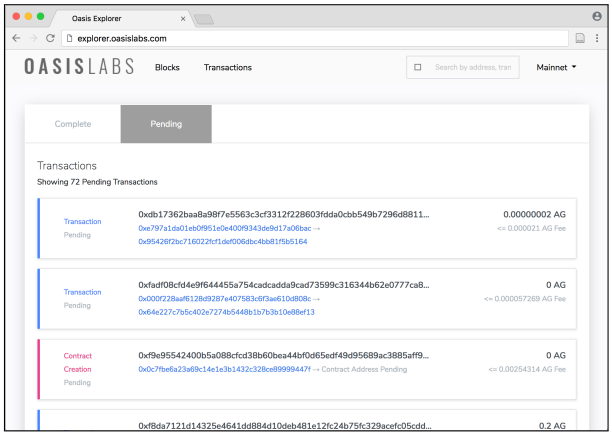

Figure 4: Blockchain explorer which provides visibility into Sterling transactions. Here, we see two requests for data and the creation of a new provider contract.

4. CONCLUSION & FUTURE WORK

In this demo proposal, we introduce Sterling, a data marketplace based on privacy-preserving smart contracts, which allows participants to exchange and use private data without revealing the data or the analytics performed thereon to untrusted parties. These interactions are mediated through novel data provider and consumer smart contracts; each automatically enforces the terms-of-use set by its creator. Upon this generic platform, we build a market for privacy-preserving machine learning data and models. In this context, models are kept from leaking the training data by automatic verification of differential privacy. In this way Sterling enables applications including credit scoring, smart home automation, and medical diagnosis. Indeed, the medical use case is the focus of our demo, which highlights the usability and security of our system.

As a follow-up to this demonstration, we aim to deploy Sterling and conduct a formal observational study on its real-world utility. Further lines of inquiry might explore exotic terms of use, privacy-preservation via secure multi-party computation, or even non-ML use cases like decentralized ad serving or customer relationship management. We hope that Sterling makes a step towards amplifying the value of heretofore unshareable data.

Acknowledgements

We would like to thank Ce Zhang, Dan Alistarh, and Claudiu Musat for their helpful feedback and discussion.

References

[1] Martın Abadi et al. “Deep Learning with Differential Privacy”. In: *CCS*. 2016, pp. 308–318.

[2] Ittai Anati et al. “Innovative technology for CPU based attestation and sealing”. In: *HASP*. Vol. 13. 2013.

[3] Nicholas Carlini et al. “The Secret Sharer: Measuring Unintended Neural Network Memorization & Extracting Secrets”. In: *arXiv:1802.08232* (2018).

[4] Raymond Cheng et al. “Ekiden: A Platform for Confidentiality-Preserving, Trustworthy, and Performant Smart Contract Execution”. In: *arXiv:1804.05141* (2018).

[5] D. Dao et al. “DataBright: Towards a Global Exchange for Decentralized Data Ownership and Trusted Computation”. In: *arXiv:1802.04780* (2018).

[6] Cynthia Dwork, Guy N. Rothblum, and Salil P. Vadhan. “Boosting and Differential Privacy”. In: *FOCS*. 2010, pp. 51–60.

[7] Lisa Fleischer and Yu-Han Lyu. “Approximately optimal auctions for selling privacy when costs are correlated with data”. In: *EC*. 2012, pp. 568–585.

[8] Justin Hsu et al. “Differential Privacy: An Economic Method for Choosing Epsilon”. In: *CSF*. 2014, pp. 398–410.

[9] Nick Hynes, Raymond Cheng, and Dawn Song. “Efficient Deep Learning on Multi-Source Private Data”. In: *arXiv 1801.07860* (2018).

[10] Ruoxi Jia et al. “‘How Much is My Data Worth?”: Data Valuation with Efficient Shapley Value Estimation”. 2018.

[11] *Keystone Project*. 2018. URL: `https://keystone-enclave.org/`.

[12] Pang Wei Koh and Percy Liang. “Understanding Black-box Predictions via Influence Functions”. In: *ICML*. 2017, pp. 1885–1894.

[13] Ilia A. Lebedev, Kyle Hogan, and Srinivas Devadas. “Secure Boot and Remote Attestation in the Sanctum Processor”. In: *IACR*. 2018, p. 427.

[14] Frank McSherry. “Privacy integrated queries: an extensible platform for privacy-preserving data analysis”. In: *SIGMOD*. 2009, pp. 19–30.

[15] Joe Near et al. “Optio: Differential Privacy for Machine Learning Pipelines, Statically and Automatically”. 2018.

[16] *Oasis Blockchain*. 2018. URL: `https://www.oasislabs.com/`.

[17] Olga Ohrimenko et al. “Oblivious Multi-Party Machine Learning on Trusted Processors”. In: *USENIX Security*. 2016, pp. 619–636.

[18] Alvin Rajkomar et al. “Scalable and accurate deep learning for electronic health records”. In: *arXiv:1801.07860* (2018).