# Safe Collections and Stewardship on Cloud Kotta

Yadu N. Babuji, Kyle Chard, Eamon Duede, Ian Foster
Computation Institute
University of Chicago and Argonne National Laboratory
{yadunand,chard,eduede,foster}@uchicago.edu

***Abstract*—To address these needs we present CLOUD KOTTA, a cloud-based architecture for the secure management and analysis of social science data. CLOUD KOTTA leverages reliable, secure, and scalable cloud resources to deliver capabilities to users, and removes the need for users to manage complicated infrastructure.CLOUD KOTTA implements automated, cost-aware models for efficiently provisioning tiered storage and automatically scaled compute resources.CLOUD KOTTA has been used in production for several months and currently manages approximately 10TB of data and has been used to process more than 5TB of data with over 75,000 CPU hours. It has been used for a broad variety of text analysis workflows, matrix factorization, and various machine learning algorithms, and more broadly, it supports fast, secure and cost-effective research.**

## I. INTRODUCTION

Today, researchers are increasingly dependent on the ability to store, manage and analyze vast amounts of data. The data-sets come in varying size, structure and sensitivity. In a typical research lab, collaborators from multiple institutions often share data and techniques for analyzing the data. Given the sensitivity of data, the security of the data is a primary concern.

Labs engaged in data-science often use a wide range of data-sets. While many data-sets are public the most valuable ones are often considered private due to legal or contractual obligations under which the data is shared, or even governmental regulations due to the nature of the data. The limited and proprietary nature of the data further increases the perceived value to the researcher. However this also adds the burden of ensuring the security of the data to the researchers.

Here are some critical problems researchers face in this space:

- Data-sets are large, posing problems for storage, compute and cost.
- Varying degrees of sensitivity in data-sets require configurable security
- Share access, analyses across individuals and groups.

## II. BACKGROUOND AND MOTIVATION

There are two major camps that try to attempt to solve the problem of computing on sensitive datasets without disclosing the underlying sensitive micro-data. Data enclaves attempt to provide a secure environment to apply computation over sensitive data with security enforced via controlling what enters and exits the system often via human oversight. Formal privacy techniques on the other hand state privacy assumptions and guarantees and the privacy loss from analyses are formally defined.

[Yadu: Kyle: We need a review that expands on the following]

Current limitation of data-enclaves: The need for switching to Cloud Democratization - Anyone can have an enclave Cost effective - Demise of the campus cluster model Legal constraints?

How to secure the outputs that are generated ?

Limitations of formal methods:

Show the current scale of compute (0.25M core hours last year). Any system that adds to this overhead can be significantly expensive.

Homomorphic encryption overheads are too high.

Discuss the 3 flavors of data: 1) Public: no sensitive information and no protection required 2) Confidential: contains private information and legal/contractual obligation to control access and usage. 3) Regulated: contains sensitive data that can be widely damaging if disclosed. Usually controlled by govt. regulations. Eg HIPAA, SSNs etc

Cloud Kotta is a data-enclave designed to enable computation over sensitive data-sets in a cost effective and secure manner. In prior work we've shown that Cloud Kotta offers a scalable solution to match the compute needs of a distributed research network. This model involved two classes of users: 1) the administrators who take on the role of managing users, approving privileges and handling the transfer of data onto the data stores. 2) the analysts who perform the role of developing analytical pipelines and running analyses on the data-sets.

The workflow of an analyst on Cloud Kotta involves requesting access to a data-set, submitting analyses tasks via the various interfaces to interrogate the data and tune the analyses until results are generated. These results are available by requesting the UUID4 url of the submitted task and not tied to any user. This was designed to allow for sharing of analyses codes and results, but this introduces the risk of unintended, unapproved access to the generated results. More importantly since there is no disclosure controls on the results generated, any sensitive data that makes it to the results can be exported.

[Yadu: Ref https://tools.ietf.org/html/rfc4122.html ?]

## III. ARCHITECTURE AND IMPLEMENTATION

To simplify and address concerns outlined in the previous section we introduce two major changes. We introduce safe collections, an abstraction that encapsulates a data-set and the set of policies that govern it's use. Seconondly, in order to separate administative responsibilities pertaining to the safe-collection from the infrastructure admins, we create a new


Fig. 1. Safe collection schema

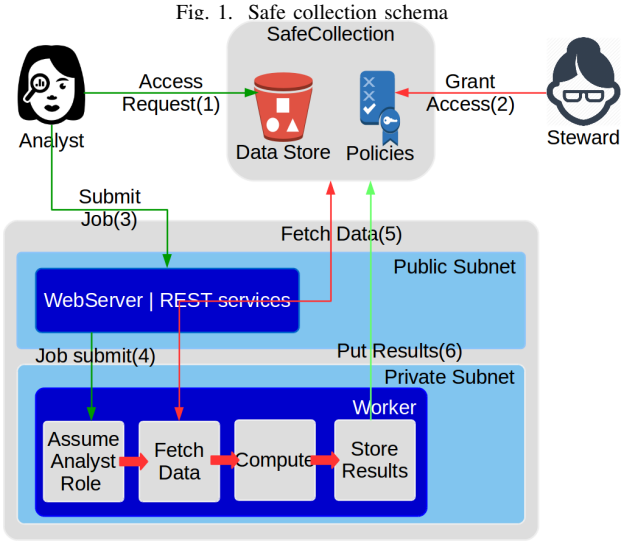


Fig. 2. Safe collection schema

class of privileged users called Stewards. In this design, administrators create a safe collection and hand over all administrative privileges to the stewards. This transfers all administrative responsibilities regarding access and export of data to the stewards who now take ownership and responsibility for the safe collection.

### A. *Safe Collection*

Data-sets come with various levels of sensitivity and the selecting protocols to ensure adequate security requires a case-by-case treatement. To closely integrate such policies with the data-set we have created safe-collections. Our implementation supports a range of protocols to be enforced on the data-set in three key areas: Access control, Export control and Link control. These controls can be described through a document.


## ACKNOWLEDGMENTS

Ack - Klab, DSaPP, Tristan


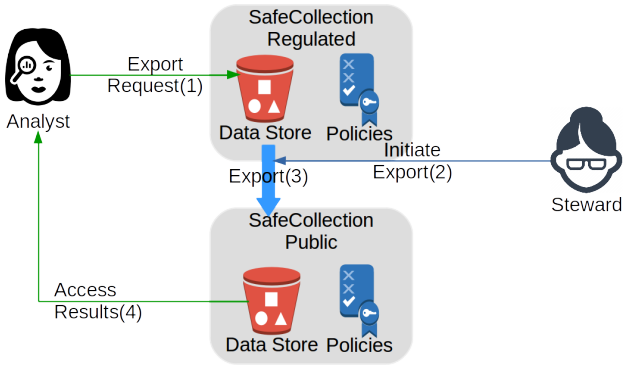


Fig. 3. Safe collection schema