Lab 10 – Privacy via remote data analytics

Course: Big Data Analytics
Instructor: David Johnson
Student: Ruth Ihunanya Chimezuru Obere
Date: 17 January 2025

Reflection on the Hello Syft Notebook

The Hello Syft notebook demonstrates **privacy-preserving remote data analytics**. PySyft allows a data scientist to **send computation code** to the data owner's environment instead of receiving the raw data. The computation runs **remotely**, and only the **results** are returned. This ensures that the data owner's private information is **never exposed**.

While working through the notebook, I observed the following:

- Running the `Submit code` cell (data scientist sending computation) **did not produce any visible output**. This is expected, as the data scientist never has access to the raw data — only the results are returned.

- Running the `Load data` cell (data owner's environment) **did produce output**, because the data owner has access to the private dataset locally.

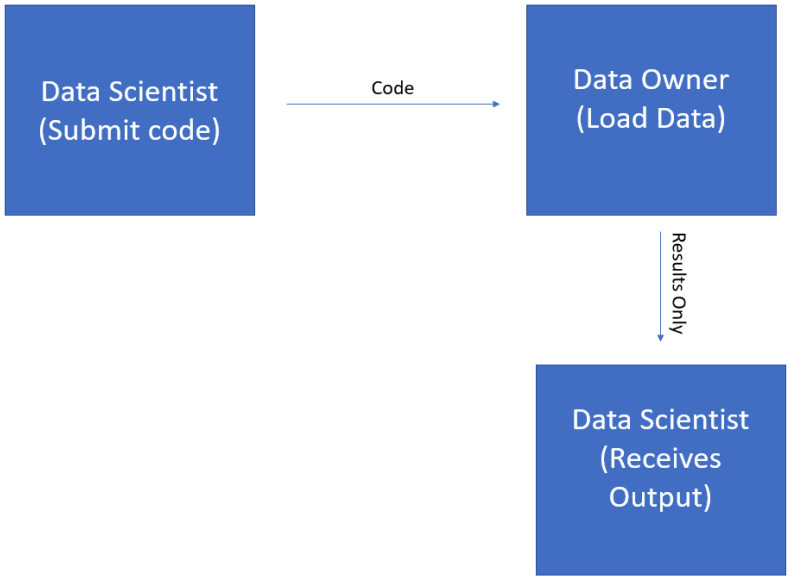
This behavior **demonstrates the core privacy principle**: the data **remains with the owner**, and the data scientist **cannot see the raw data**, only the computation results.

Some cells (like `load data`) were not executed in full due to library version issues, but the notebook still clearly illustrates the privacy concept.

Diagram 1 – Hello Syft Conceptual Flow

Description:
This diagram illustrates the interaction between a single data scientist and a single data owner in the Hello Syft notebook.

Data movement:



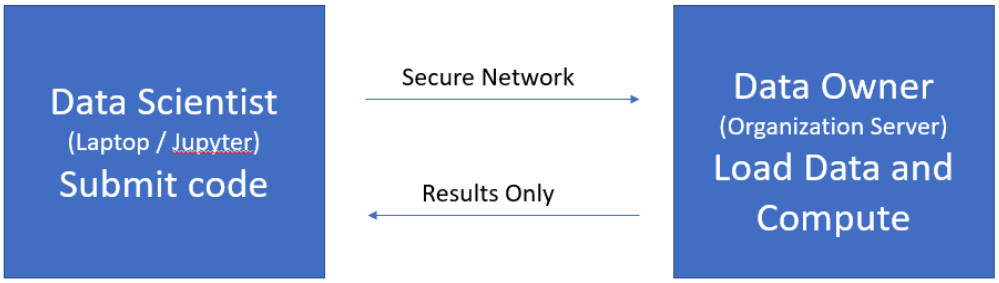
Explanation:

- **Data Scientist (Submit code):** Writes Python code to analyze the data and sends it to the data owner.
- **Data Owner (Load data):** Holds the private data and executes the computation.
- **Flow:** Only results are returned to the data scientist; raw data never leaves the owner.

Diagram 2 – Real Deployment of PySyft

Description:
This diagram shows how PySyft could be set up in a real-world scenario with one data owner and one data scientist.

Data movement:



Explanation:

- **Data Scientist:** Uses Python and a PySyft client to send analysis code.
- **Data Owner Organization:** Runs computations on a secure server holding private data.
- **Secure Network:** Ensures communication is encrypted.
- **Flow:** Only computation results are returned to the data scientist; raw data never leaves the organization.

My Observation and Conclusion

- The notebook clearly demonstrates **privacy-preserving remote computation**.
- The **data owner keeps full control** of their data.
- The **data scientist cannot see raw data**, only the computation results.
- Differences in output from `Submit code` and `Load data` cells show how privacy is enforced:
    - `Submit code` → no output (data scientist cannot see raw data)
    - `Load data` → shows output (data owner can access data locally)

Even though some cells could not be fully executed due to library version issues, the **core concept of privacy is fully illustrated**.

Key Takeaway: PySyft ensures that computation is sent to the data, not the other way around, keeping sensitive information secure.