Digitalization and Infectious Diseases: Improving patient outcome in the age of big data
Basel, 21 January 2020

Privacy-Conscious Data Sharing Data: Computing on Data without Moving Them

Jean-Pierre Hubaux

With contributions from Jean Louis Raisaro, Juan Troncoso-Pastoriza, Jacques Fellay and Nicolas Rosat
US Healthcare Official “Wall of Shame”

https://ocrportal.hhs.gov/ocr/breach/breach_report.jsf

Around 5 declared breaches per week, each affecting 500+ people
“Legal deterrence” and public shame are clearly not enough!

Millions of Americans’ Medical Images and Data Are Available on the Internet. Anyone Can Take a Peek.

Hundreds of computer servers worldwide that store patient X-rays and MRIs are so insecure that anyone with a web browser or a few lines of computer code can view patient records. One expert warned about it for years.

by Jack Gillum, Jeff Kao and Jeff Larson, Sept. 17, 12 a.m. EDT
## Technologies for Privacy and Security Protection

<table>
<thead>
<tr>
<th>Traditional Encryption</th>
<th>Homomorphic Encryption</th>
<th>Secure Multiparty Computation</th>
<th>Trusted Execution Environments</th>
<th>Differential Privacy</th>
<th>Distributed Ledger Technologies (Blockchains)</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Protects data at rest and in transit</td>
<td>• Protects computation in untrusted environments</td>
<td>• Protects computation in distributed environments</td>
<td>• Protects computation with Hardware Trusted Element</td>
<td>• Protects released data from inferences</td>
<td>• Strong accountability and traceability in distributed environments</td>
</tr>
<tr>
<td>• Cannot protect computation</td>
<td>• Limited versatility vs efficiency</td>
<td>• High communication overhead</td>
<td>• Requires trust in the manufacturer, vulnerable to side-channels</td>
<td>• Degrades data utility (privacy-utility tradeoff)</td>
<td>• Usually no data privacy</td>
</tr>
</tbody>
</table>
Multi-site Studies – Where to Store the Data?

a. Keep them at each site
   - Useful especially if the cloud is untrusted
   - Better control of the data

b. Upload them in the cloud
   - Take advantage of the well-known strengths of the cloud (see next slides)
Case 0: The Cloud is Fully Trusted – Storage in cleartext (never happens in practice)

- Data sharing is easy
- Computation in the cloud is easy
Case 1: The Cloud is Fully Trusted – It encrypts with keys it controls

- Data sharing is easy
- Computation in the cloud is easy

Encrypted Data

Secured pipe (that’s easy)
Case 2: The Cloud Is Untrusted – The user encrypts under their own keys

- Data sharing is tricky (key management)
- Computation in the cloud is impossible
- Some of the benefits of cloud computing are thus lost
- If the user loses their keys, they lose all their data

Encrypted Data

Secured pipe (that’s easy)
Case 3: The Cloud is Untrusted – The user homomorphically encrypts with keys it controls (1/3)

- Data sharing is doable
- Computation in the cloud is possible, but expensive
Case 3: The Cloud is Untrusted – The user homomorphically encrypts with keys it controls (2/3)

- Data sharing is doable
- Computation in the cloud is possible, but expensive
Case 3: The Cloud is Untrusted – The user homomorphically encrypts with keys it controls (3/3)

- The cloud can make computations on encrypted data, **for which it does not know the crypto keys**
- Hence computation in the cloud is possible (albeit expensive)
- Data sharing is doable

\[ \text{Encr}(3) \]
\[ \text{Encr}(5) \]
\[ \text{Encr}(3+5) = \text{Encr}(8) \]

Secured pipe

\[ \text{Encr}(3) \]
\[ \text{Encr}(5) \]
Homomorphic Encryption

\[ E(a), E(b) \xrightarrow{\text{Compute} (\circ)} E(a \circ b) \]

Homomorphic encryption enables computations directly on encrypted data.
Multi-site Studies: Keeping the Data at Each Site

Assume Sites do not trust each other

→ Solution we use: Secure Multi-Party Computation
Secure Multiparty Computation

Problem statement:

A set of players $\mathcal{P} = \{P_1, P_2, \ldots, P_N\}$ would like to compute a function $f(x_1, x_2, \ldots, x_N) = (y_1, y_2, \ldots, y_N)$ of their joint inputs.

Requirements:

1. **Privacy**
   No party should learn anything more than its prescribed output
2. **Correctness**
   Each party is guaranteed that the output that it receives is correct

Realization:

A multiparty cryptographic protocol
Data Protection in Personalized Health

- 4 research groups across the ETH domain + SDSC (Swiss Data Science Center)
- Funding: 3 Millions CHFrs
- Duration: 3 years (4/2018 - 3/2021)
- Funding Program: ETH PHRT (Personalized Health and Related Technologies)

Project goals:
- Address the main privacy, security, scalability, and ethical challenges of data sharing for enabling effective P4 medicine
- Define an optimal balance between usability, scalability and data protection
- Deploy an appropriate set of computing tools
Data Protection in Personalized Health

https://dpdp.ch
DPPH: Envisioned Nation-Wide Deployment

Q2: What is the survival rate for cancer patients undergoing a given chemotherapy?

Q1: How many patients with BRCA1 and breast cancer?
MedCo is the first operational system that makes sensitive medical data available for research in a simple, private and secure way. It enables hundreds of clinical sites to collectively protect their data and securely share them with investigators without single points of failure.

To achieve this, MedCo applies advanced privacy-enhancing technologies, such as collective homomorphic encryption, secure multi-party computation, blockchain and differential privacy. MedCo can be used with state-of-the-art cohort explorers. Today, it is already supported by D3a3, the most widespread cohort explorer used in more than 200 clinical cohort worldwide.

https://medco.epfl.ch/
Information security technologies used by MedCo

Homomorphic encryption
- Enables computations on encrypted data

Secure multi-party computation
- Enables decentralization of trust thus avoiding single points of failure

Differential privacy
- Enables to mitigate inference risks stemming from the release of aggregate data

Private blockchain
- Enables to immutably log actions performed in the network for traceability, auditability and reproducibility

Honest-but-curious adversary
- Malicious adversary
MedCo software stack: re-using the best of medical informatics and information security

Common Data model
Privacy-preserving computing framework
Modern GUI

Collective protection of medical data
MedCo: Consortium and project goals

- **Funding:** SPHN + PHRT
- **Budget:** 530K CHF
- **Start date:** April 1st 2019
- **Duration:** 18 months
- **First application:** oncology: O. Michielin,…
- **Goal(s):**
  1. Bringing MedCo from an “academic” prototype to “hospital-compliant” operational system
  2. Deploy and test MedCo in (at least) 3 Swiss University Hospitals
  3. Validate MedCo with end-users

N. Rosat  J. Fellay  D. Cavin  A. Leichte

JP Hubaux
LDS, C4DT

EPFL
MedCo Project Timeline

- **April 2019**: Academic prototype
- **September 2020**: Hospital-compliant system

- **WP1s**: Requirements Elicitation – Hospital compliance
- **WP2s**: Deployment and Benchmarking
- **WP3s**: User study & Validation

- **WP1p**: System Development and Adaptation
- **WP2p**: Packaging and Final Release

- **Now M10**:
Requirements for hospital compliance

- Security policies for development of Web applications (based on OWASP recommendation)
  - HUG
  - CHUV
  - Insel
First use case: Swiss Molecular Tumor Board

- Joint work with the Swiss Personalized Oncology (SPO) representatives
Foreseen user experience: (step 1) cohort selection
Foreseen user experience: (step 2) survival analysis
First use case: melanoma

- Pseudonymized test cohort from the Swiss Personalized Oncology pilot project on melanoma
  - Encounters, basic demographics,
  - Diagnosis, Tumor staging,
  - Specimen, genetic test, genetic variants,
  - Treatment (drug, radiotherapy, surgery),
  - Treatment outcome,
  - Adverse-event, follow-up

18 variables
MedCo-Explore scalability tests

Population: 150,000 individuals
Observations/individual: (15,000, 200,000)
Dataset size: up to 28 billion observations
Query size: (1,50) terms
Resulting set: (100, 1511) individuals/node
#servers: (3, 12)
28 B data points
1511 matching patients
10 query terms
MedCo-Analysis

Decentralized, Secure, Verifiable System for Statistical Queries and Machine Learning on Distributed Databases [1]

Functionality: Enable queries on a set of distributed databases while protecting individuals privacy and data confidentiality.

**Statistics**
- sum/count/frequency count
- and/or, max/min
- variance/standard deviation
- Set intersection/union
- Cosine similarity

**Machine Learning**
- linear regression
- logistic regression
- Neural networks

MedCo-Analysis: Query Workflow

1. The querier defines the **query**: training of a linear regression model on specific attributes.

2. Each data provider $i$ (DP$_i$) **locally computes** a function $\sigma$ on its local database $d_i$.

3. Each data provider **encrypts** its result.

4. The DPs collectively **aggregate** the encrypted results.

5. The querier can **decrypt and compute** the final result.
Example: Linear/Logistic Regression

Background:

Goal: Find the line (defined by $b_0$ and $b_1$) that best fits the dots $(x_i, y_i)$.

Generic Method to find the best $b_0$ and $b_1$: gradient descent is used to find the $b_0$, $b_1$ that give the minimum error.
Example: Linear/Logistic Regression

**Background:**

**Goal:** Find the line (defined by \( b_0 \) and \( b_1 \)) that best fits the dots \((x_i, y_i)\).

**Generic Method to find the best \( b_0 \) and \( b_1 \):**

Gradient descent is used to find the \( b_0, b_1 \) that give the minimum error.
Example: **Distributed Linear/Logistic Regression**

**Problem:** the data providers have to collaborate during the gradient descent, otherwise they can find different minimums.

![Diagram](image_url)

- $b_0$ and $b_1$ are the parameters to be estimated.
- The error is the difference between the actual and predicted values.
- The correct minimum is the point where the error is minimized.
- Gradient descent steps are shown as arrows leading to the minimum achieved by DP1.
Example: Distributed Linear/Logistic Regression

**Solution:** the data providers collaborate to enable a joint gradient descent while protecting their privacy

1. DPs create encrypted summary of their data
2. DPs’ summaries are collectively aggregated
3. The aggregated summary encryption is switched to the querier’s key
4. The querier decrypts the final summary
5. The querier performs the gradient descent on the final data summary
Distributed Logistic Regression - Evaluation

LBW = Low birth weight dataset. 10 features [6]
PCS = Prostate Cancer Study. 10 features [12]
Pima = Pima Indians Diabetes 8 features [10]

Parameters:
6 Compute Nodes, 7 Verifying Nodes
60 DPs
80% training; 20% testing
Scaling factor $10^2$;
learning rate 0.1;
k = 2;
l2-regularization factor = 1;

<table>
<thead>
<tr>
<th>Data.</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>LBW</td>
<td>Centralized 69.31% MedCo 70.26%</td>
</tr>
<tr>
<td>PCS</td>
<td>Centralized 74.60% MedCo 75.13%</td>
</tr>
<tr>
<td>Pima</td>
<td>Centralized 80.5% MedCo 77.55%</td>
</tr>
<tr>
<td>SPECTF</td>
<td>Centralized 78.9% MedCo 74.87%</td>
</tr>
</tbody>
</table>

Main features and guarantees of the toolset

Functionalities:

- **“MedCo-Explore”** [1]:
  - Count and select patients based on Boolean combination of inclusion/exclusion genetic and clinical criteria on data distributed at different locations

- **“MedCo-Analysis”** [2]:
  - Compute distributed basic statistics and set operations (variance, stdev, set intersection/union, cosine similarity)
  - Perform distributed basic machine learning tasks (train and run linear regression, logistic regression models)

Security/Privacy guarantees:

- End-to-end data protection (at rest, in transit and during computation) against unauthorized data access
- No single point of failure
- Only the investigator can obtain the query end-result
- Minimal risk of re-identification from aggregate data released (differential privacy)

---


BLOCKCHAIN.
We use a closed ("permissioned") blockchain, unlike Bitcoin that uses a public ("non-permissioned") blockchain.
Conclusion

- It is possible to securely make computations on decentralized data without moving them
- We have developed a unique know-how and provably secure open-source software solution for privacy-conscious medical data sharing
- The proposed solutions are based on modern crypto and generate reasonable overhead, at least for low- to medium-complexity operations
- We are currently deploying these solutions at 3 Swiss university hospitals (Bern, Geneva and Lausanne)
- First applications are in oncology, but the same tools can be used for infectious diseases as well

https://dpph.ch
https://medco.epfl.ch
jean-pierre.hubaux@epfl.ch