Measuring demographic disparities with groupwise private set intersection

A Federal Government Case Study

NIST Workshop on Privacy Enhancing Cryptography
9/25/24

xD
https://www.xd.gov

Tomo Lazovich, Ph.D. (they/them)
Emerging Technology Fellow
xD, U.S. Census Bureau (in partnership with GSA 10x)

United States Census Bureau

All statements are the author's personal views and do not necessarily reflect Census Bureau policy.

BEFORE DIVING IN – INTRODUCTION TO THE xD TEAM

xD is an **emerging technologies group** that's advancing the delivery of data-driven services through new and transformative technologies.

*We do this work by bringing on cohorts of **Emerging Technology Fellows** and by collaborating with others throughout the Census Bureau and beyond.*

PRIVACY ENHANCING TECHNOLOGIES (PETs): THE GOAL

How can we enable analysis and gain insights without revealing private information?

Add noise: differential privacy, synthetic data generation

Encrypt: ***secure multi-party computation***, fully homomorphic encryption, zero knowledge proofs, secure enclaves

IMPORTANCE OF DEMOGRAPHIC DATA FOR FEDERAL AGENCIES

JANUARY 20, 2021

Executive Order On Advancing Racial Equity and Support for Underserved Communities Through the Federal Government

BRIEFING ROOM ▸ PRESIDENTIAL ACTIONS

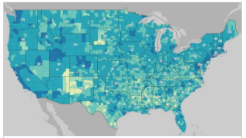
United States Census Bureau

Partners Researchers Educators Survey Respondents

Topics Data & Maps Surveys & Programs

// Census.gov / About the Bureau / What We Do / Advancing Equity with Data

Data Tools



ACCESS BROADBAND Dashboard

The ACCESS BROADBAND Dashboard displays maps for users to assess economic conditions in areas with changes in broadband availability and adoption.

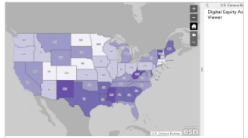
Data Tool

Census Business Builder (CBB)

Census Business Builder offers small business owners selected Census Bureau & other statistics to guide their research for opening or expanding their business.

Community Resilience Estimates (CRE) Tools

The CRE provide easily understood metrics for how socially vulnerable every neighborhood is to the impacts of disasters and other stressors.



Data Tool

Digital Equity Act Population Viewer

Interactive collection of maps that highlight various demographics and broadband internet availability and adoption by state.

DEMOGRAPHIC DATA IS NEEDED TO AUDIT AI/ML SYSTEMS IN GOVERNMENT

AI

IRS's AI system to flag returns for audit may include unintended bias, report finds

Following a report identifying racial disparities in audit selection, the GAO says the tax agency hasn't conducted a "comprehensive review" of the rules and filters in its Dependent Database.

BY MATT BRACKEN • MAY 23, 2024

A STAT INVESTIGATION

Denied by AI: How Medicare Advantage plans use algorithms to cut off care for seniors in need

By Casey Ross and Bob Herman March 13, 2023

—

How can we securely share data to enable measuring demographic disparities?

DISCLAIMER: I am *not a cryptographer*

INTRODUCTION TO PRIVATE JOIN AND COMPUTE PROTOCOL

PARTY 1

ID
1
2
3

PARTY 2

ID	Feature
2	700
3	800
901	60000

INPUTS

PJC protocol

Intersection size
2

Intersection size	Feature sum
2	1500

OUTPUTS

https://security.googleblog.com/2019/06/helping-organizations-do-more-without-collecting-more-data.html

PRIVATE JOIN AND COMPUTE PROTOCOL, MORE FORMALLY

Private Intersection Sum with Cardinality

Inputs:

P_1 : Set $V = \{v_i\}_{i=1}^{m_1}$ P_2 : Set of pairs $W = \{(w_i, t_i)\}_{i=1}^{m_2}$

Outputs:

$\mathsf{P}_1 : C = |\{i : w_i \in V\}|$ $\mathsf{P}_2 : C = |\{i : w_i \in V\}|, S = \sum_{i:w_i \in V} t_i$

Figure 1: F_{PIS-C} : The Private Intersection-Sum with Cardinality functionality.

For details of the protocol, see Ion et al., *On Deploying Secure Computing: Private Intersection-Sum-With-Cardinality*, *https*://eprint.iacr.org/2019/723.pdf

PRIVATE MODEL ACCURACY COMPUTATION

PARTY 1

ID
1
2
3

PARTY 2

ID	Prediction = ground truth?
2	1
3	0
901	0

INPUTS

PJC protocol

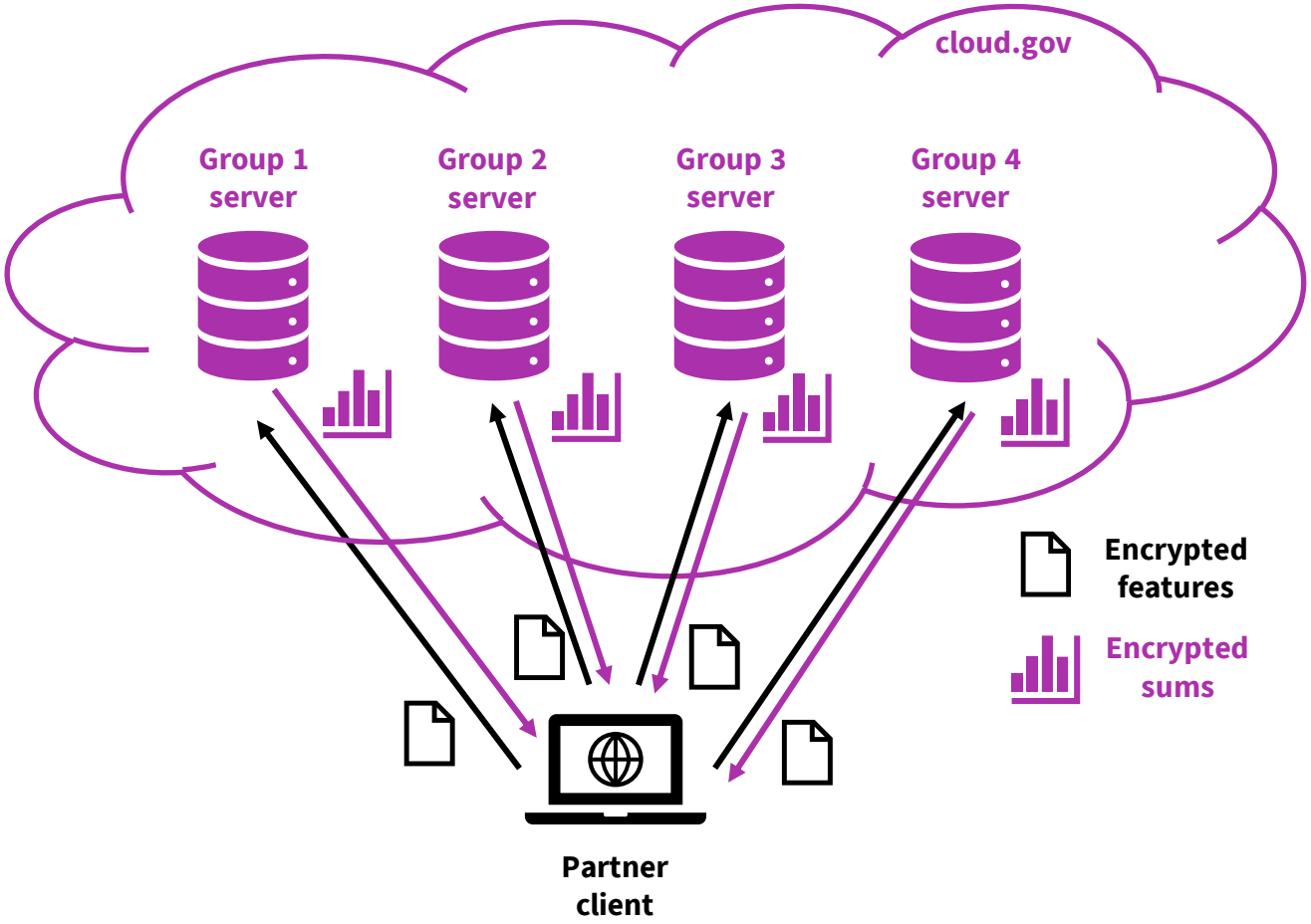
Intersection size
2

Intersection size	Number of correct predictions
2	1

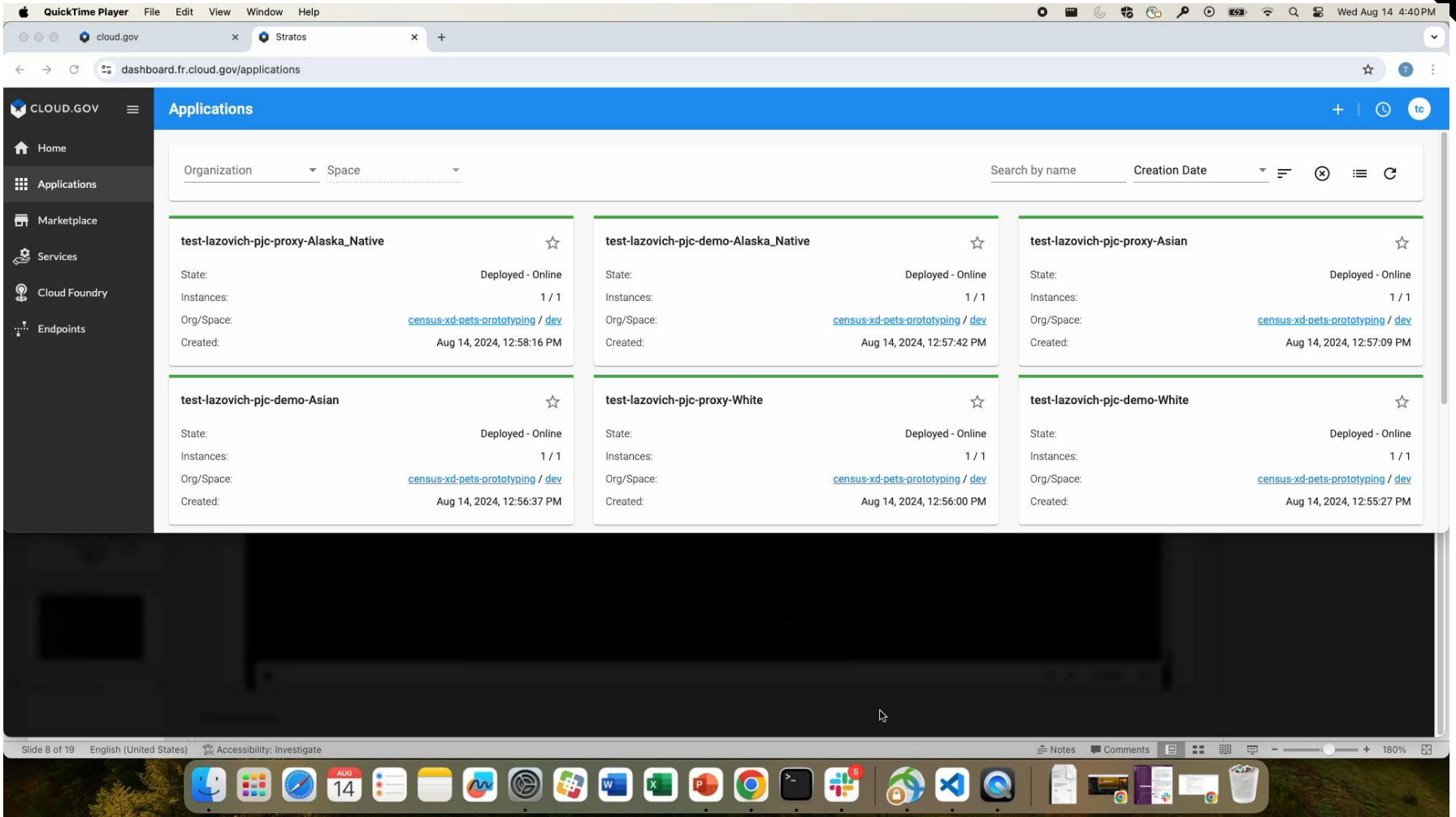
Accuracy = 1 / 2 = 50%

OUTPUTS

COMPUTING ACROSS DEMOGRAPHIC GROUPS



Demo – ML model performance across demographic groups



—

What engineering details did it take to make it work?

Engineering workarounds to run “Private Join and Compute” on cloud.gov

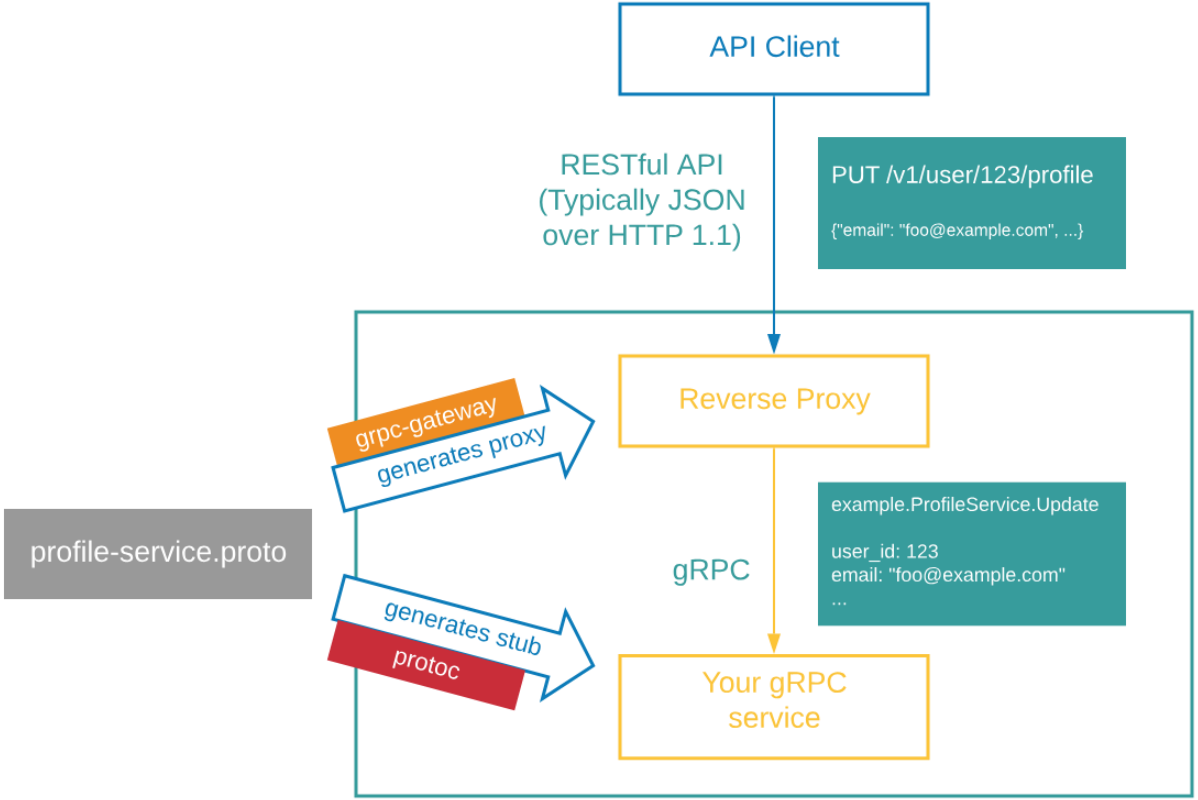
Cross-compiling Rust implementation from OS X to Linux

Generating a REST API proxy around gRPC communication protocol

Modifying client to call REST API and serialize JSON correctly

Internal cloud.gov networking for proxy communication with gRPC server

gRPC -> REST API - Overview



https://grpc-ecosystem.github.io/grpc-gateway/

gRPC -> REST API – generated code

```
service PJC {
  rpc KeyExchange(Init) returns (InitAck) {
    option (google.api.http) = {
      post: "/v1/key_exchange"
      body: "*"
    };
  }
  rpc RecvUCompanyKeys(ServiceResponse) returns (stream Payload) {
    option (google.api.http) = {post: "/v1/recv_u_company_keys"};
  }
  rpc SendECompanyKeys(stream Payload) returns (ServiceResponse) {
    option (google.api.http) = {post: "/v1/send_e_company_keys"};
  }
  rpc SendUPartnerKeys(stream Payload) returns (ServiceResponse) {
    option (google.api.http) = {post: "/v1/send_u_partner_keys"};
  }
  rpc SendUPartnerFeature(stream Payload) returns (ServiceResponse) {
    option (google.api.http) = {post: "/v1/send_u_partner_feature"};
  }

  rpc RecvStats(Commitment) returns (Stats) {
    option (google.api.http) = {post: "/v1/recv_stats"};
  }
}
```

Add annotations to proto file

```
func request_PJC_KeyExchange_0(ctx context.Context, marshaler runtim
    var protoReq Init
    var metadata runtime.ServerMetadata

    if err := marshaler.NewDecoder(req.Body).Decode(&protoReq); err
        return nil, metadata, status.Errorf(codes.InvalidArgument, "
    }

    fmt.Println(protoReq)
    msg, err := client.KeyExchange(ctx, &protoReq, grpc.Header(&meta
    return msg, metadata, err

}
```

Generated Go proxy code

Modified Rust client - example

```
169 -   let mut u_company_keys = TPayload::new();
170 -   let _ = rpc_client::recv(
171 -       ServiceResponse {
172 -           ack: Some(Ack::InitAck(init_ack.clone())),
173 -       },
174 -       "u_company_keys".to_string(),
175 -       &mut u_company_keys,
176 -       &mut client_context,
177 -   )
178 -   .await?;
    173 +
    174 +   let resp = http_client.post(
    175 +       format!("{}/v1/recv_u_company_keys", &host_pre.unwrap())
    176 +   ).send().await?.json::<KeyResponse>().await?;
    177 +
    178 +   let byte_array : Vec<ByteBuffer> = resp.result.payload.iter().map(|e| ByteBuffer{buffer: e.to_vec()}).collect();
    179 +
    180 +   let mut u_company_keys = TPayload::from(byte_array);
    181 +
    182 +   println!("{:?}", u_company_keys);
```

Convert RPC calls to REST API calls

Internal cloud.gov networking

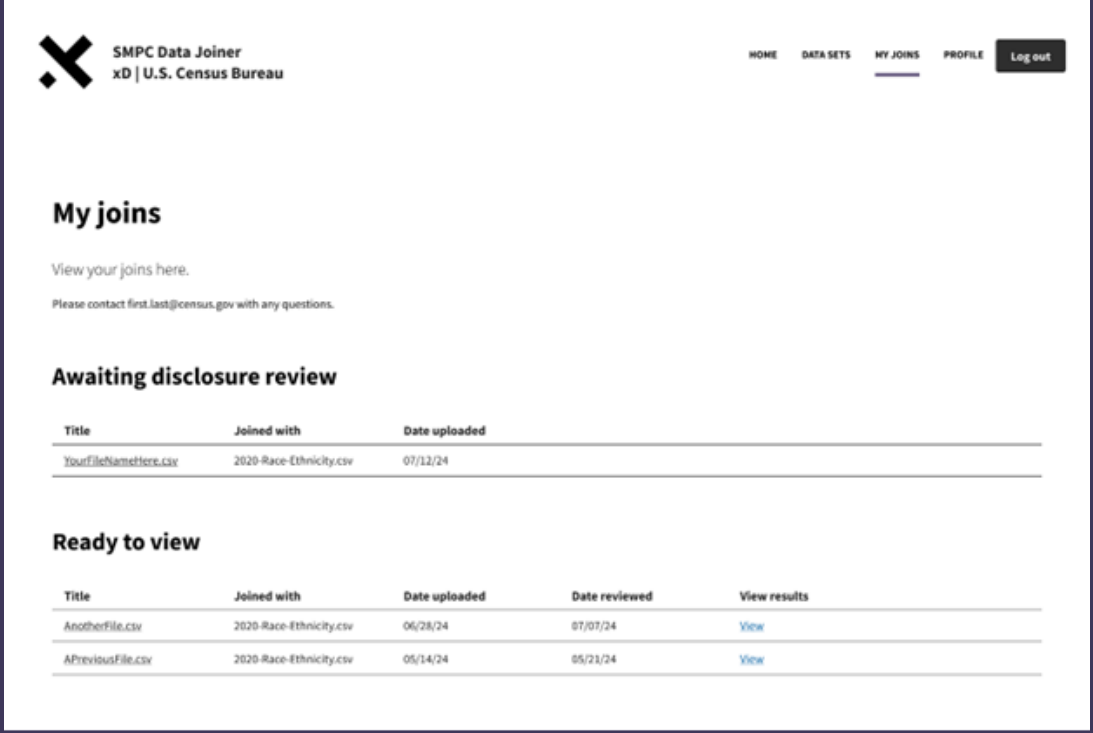
Create internal route to gRPC server app

```
cf map-route test-lazovich-binary-pjc apps.internal --hostname test-lazovich-binary-pjc --app-protocol http2
```

Allow traffic between Go proxy app and gRPC server app

```
cf add-network-policy test-lazovich-pjc-proxy test-lazovich-binary-pjc -s dev -o census-xd-pets-prototyping --protocol tcp --port 8080
```

Future vision: Demographic Disparities as a Service



We're always looking for new partnerships!

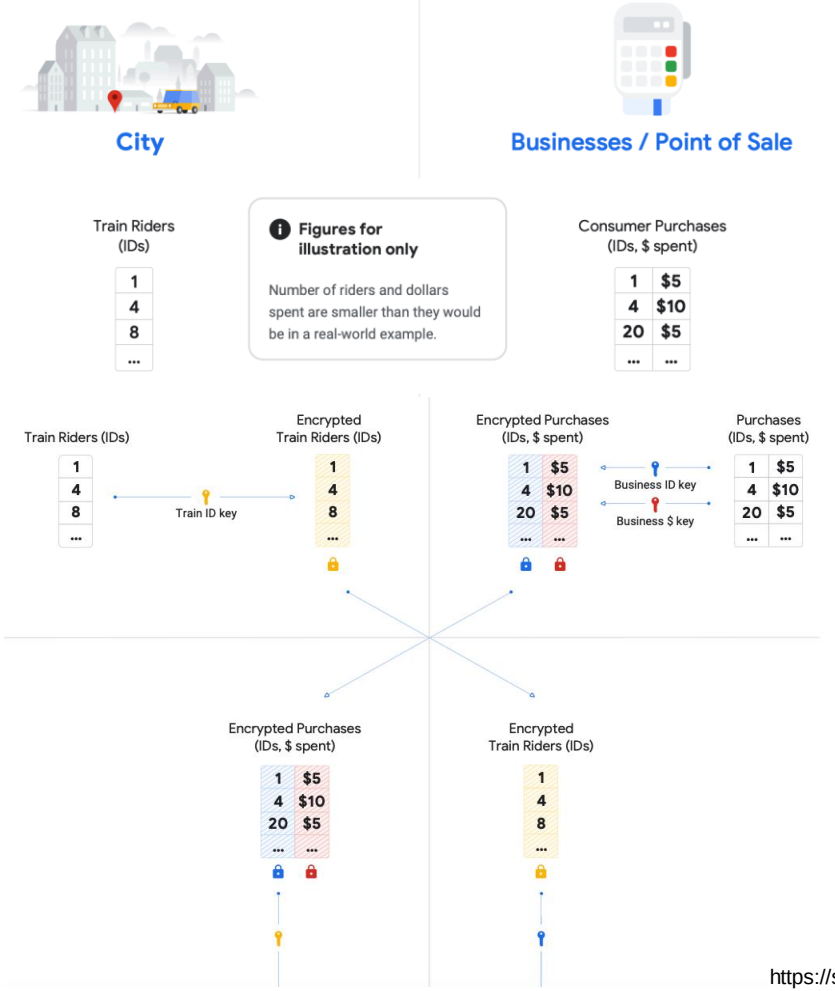
—

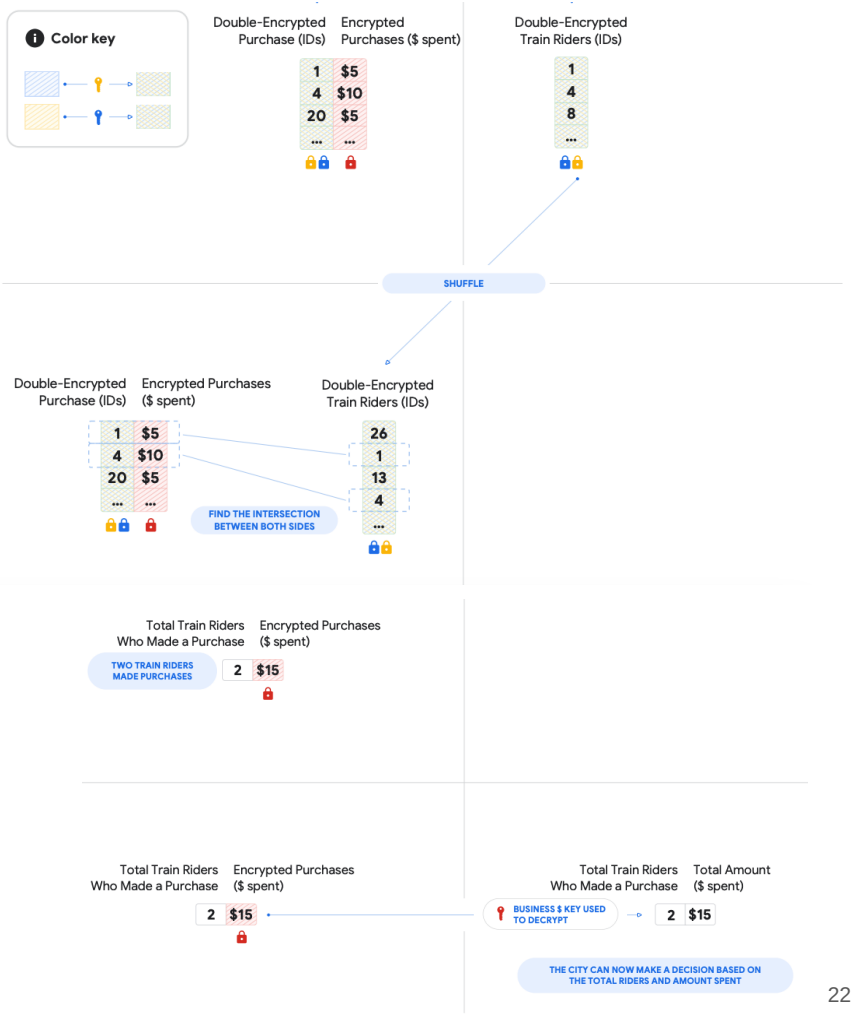
Get in touch – inquiries@xd.gov

Tomo Lazovich tomo.lazovich@census.gov

Backup

Private Join and Compute Protocol





https://security.googleblog.com/2019/06/helping-organizations-do-more-without-collecting-more-data.html

Workflow GUI Mockup

SMPC Data Joiner
xD | U.S. Census Bureau

HOME | DATA SETS | MY JOINS | PROFILE | Log out

Welcome to SMPC Data Joiner!

Data Joiner allows you to securely join your data sets with US Census Bureau data sets using Secure Multi-Party Computation (SMPC). Both party's data is double-encrypted, ensuring that Personally Identifiable Information (PII) is kept secure and inaccessible from the beginning to end of the process.

Get started

How Secure Multi-Party Computation (SMPC) works:

1. **Each party has its own data**
   The US Census Bureau has several data sets you can choose to join your data set with.
2. **Encrypting each party's data**
   First, both you and the US Census Bureau both encrypt your respective data with private keys so that it's not accessible or decipherable to anyone else.
3. **Exchanging encrypted data**
   Then, each party's encrypted data is sent to the other party.
4. **Double encrypting**
   Both party's data are encrypted with their own private keys, resulting in double-encrypted data.

   The double-encrypted IDs can be compared but can't be decrypted by either party individually.
5. **Finding intersections**
   The US Census Bureau can send your double-encrypted data back to you in shuffled order.

SMPC Example

Click to view a example from Google that illustrates how SMPC works in practice.

View example

SMPC Data Joiner
xD | U.S. Census Bureau

HOME | DATA SETS | MY JOINS | PROFILE | Log out

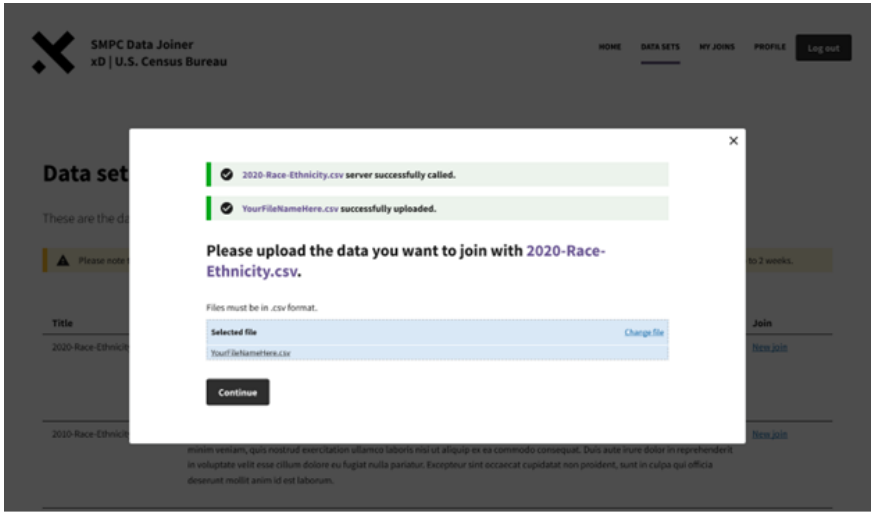
Data sets

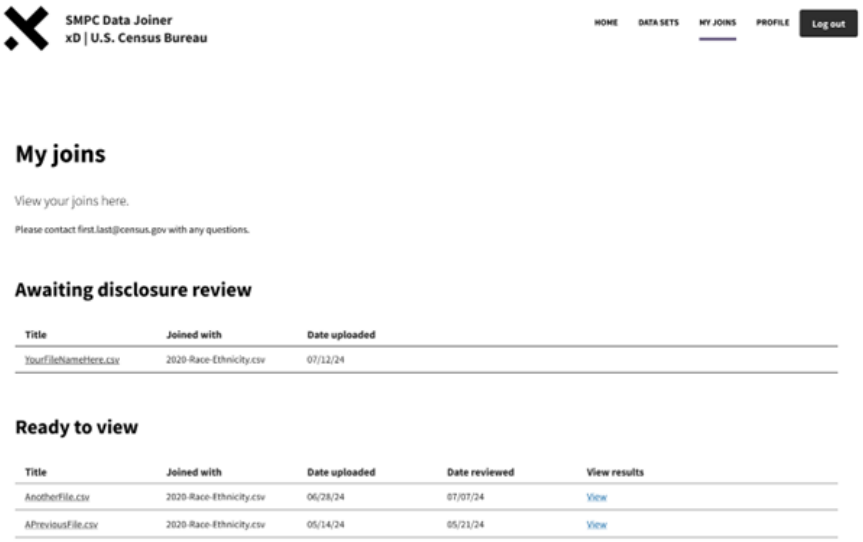
These are the data sets currently offered by the US Census Bureau. Click on one to begin the SMPC join process.

⚠ Please note that if you perform a join through this process, the results will have to go through the Disclosure Review process before you can view them. This can take up to 2 weeks.

Title	Information	Join
2020-Race-Ethnicity.csv	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.	New join
2010-Race-Ethnicity.csv	Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.	New join

Workflow GUI Mockup





Demo 2 – ML model performance across demographic groups

```
2018HU1278462,4
2018HU1297422,4
2018HU0707826,4
2018HU0225572,4
2018HU0393953,4
2018HU0677879,4
2018HU1244338,4
2018HU0623582,4
2018HU0873650,4
2018GQ0017689,4
```

Demographic server

CSV with unique ID and demographic group

```
2018HU1296546,0
2018HU0797135,1
2018GQ0056212,1
2018HU1278462,1
2018HU0143803,0
2018HU1199963,1
2018HU0474613,1
2018HU1053291,1
2018HU1256208,1
2018HU0144706,1
```

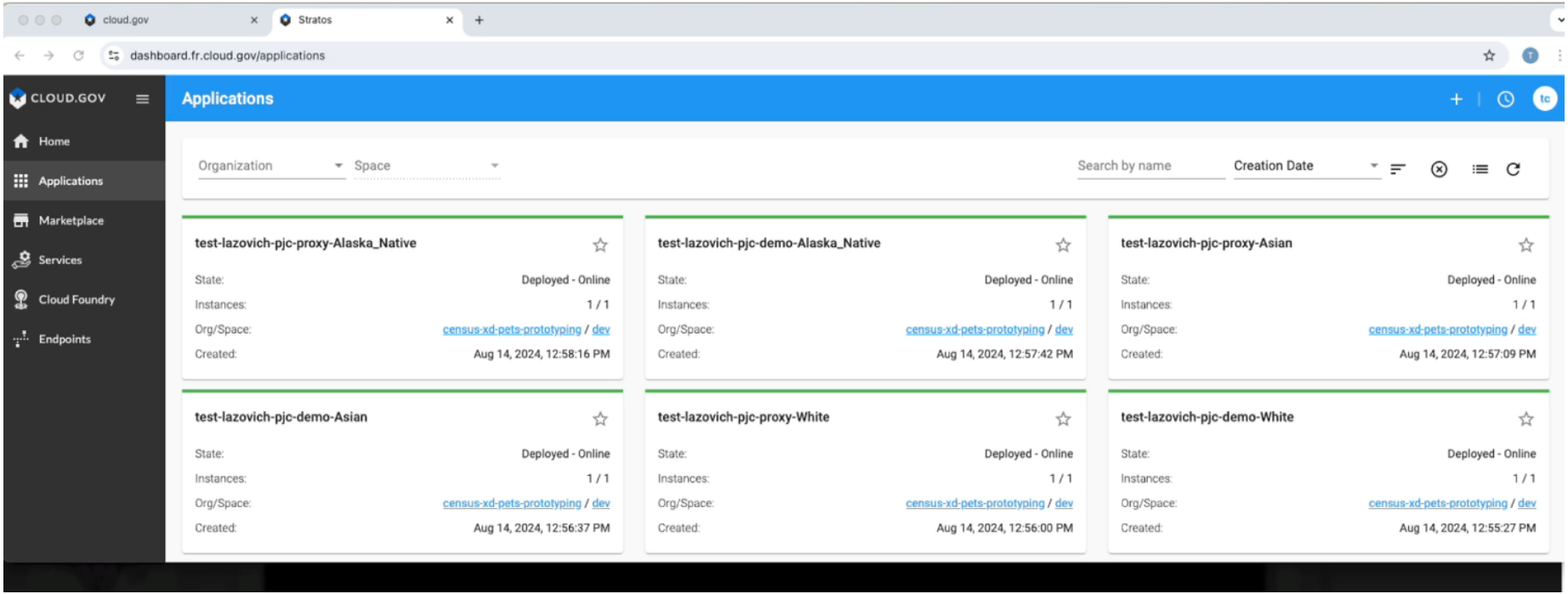
Partner client

CSV with unique ID and model outcome

Demo 2 – ML model performance across demographic groups

```bash
#!/bin/bash
set -e

declare -a demo_groups=("White" "Asian" "Alaska_Native")

results=()

for grp in "${demo_groups[@]}"
do
    echo "Running demographic group ${grp}"
    outfile=results_$grp.log

    env RUST_LOG=info cargo run --release --bin pjc-client -- --company https://test-lazovich-pjc-proxy-$grp.app.cloud.gov \
    --input etc/example/model_results.csv --stdout --no-tls >& $outfile

    num=$(cat $outfile | awk "/Sum/" | grep -o "\w*$")
    denom=$(cat $outfile | awk "/Intersection/" | grep -o "\w*$");
    ratio=$(bc -l <<< "${num} / ${denom}")

    results+=("$ratio")
done

arraylength=${#results[@]}

echo ""
echo "======== RESULTS ========"
# use for loop to read all values and indexes
for (( i=0; i<${arraylength}; i++ ));
do
  echo "Group ${demo_groups[$i]}, result: ${results[$i]}"
done
```

Demo – ML model performance across demographic groups



Demo – ML model performance across demographic groups

```
(workenv) lazov001@MD-K57DW9FJKM Private-ID % ./run-demographic-client.sh
Running demographic group White
Running demographic group Asian
Running demographic group Alaska_Native

======== RESULTS ========
Group White, result: .82142857142857142857
Group Asian, result: .77205882352941176470
Group Alaska_Native, result: .80604534005037783375
```