Privacy Preserving Measurement (IETF 113)

Eric Rescorla
ekr@rtfm.com

2022-03-23

Overview

- Measurement scenarios
- Anonymous measurement
- MPC-based privacy-preserving measurement techniques
- Technical architecture

Many situations where we want to learn about people

- Public research (e.g., the census)
  - Demographics
  - Income
  - Medical issues
- Product development
  - Which features do they use/don't use?
  - How much do they use them?
  - Where/why are products failing?
- Behavioral measurements
  - Discovering new Web sites
  - Which information are people most interested in?

This information is very useful

- But can be very sensitive
  - Medical issues, income, sexual orientation, etc.
- Even "less" sensitive data can be very revealing
  - Especially when you put a lot of "less" sensitive data together


Feb 16, 2012, 11:02am EST

How Target Figured Out A Teen Girl Was Pregnant Before Her Father Did

Kashmir Hill Former Staff
Tech
Welcome to The Not-So Private Parts where technology & privacy collide

Follow

What do we really want to measure?

- Mostly we want *aggregates*
  - What is the distribution of people's income?
  - What is relationship between income and height?
  - What are the most popular Web sites?
- Need to slice the data multiple ways
  - Just look at a given region
  - Compare two variables
- Individual values are neither necessary nor useful
  - As long as we can compute the aggregates

Measurement Types

- Simple aggregates (mean, median, sum, histograms...)
- Relationships between multiple values (correlation, OLS, ...)
- Common strings ("heavy hitters")

Example Use Case: User Interests

- Useful to measure what kinds of sites users visit
  - Bucket sites by topic
  - Count the number of visits to/minutes spent on each topic
  - But... some topics are sensitive
- Problem statement: collect distribution of time spent on each type of site

Example: Use Case: Web Site Issues

- Web compatibility is a big problem
  - Some sites will not render properly in some browsers
  - Big problem for smaller browsers like Safari and Firefox
  - Often we can detect breakage on the client
- Many Web sites fingerprint users
  - Measure persistent properties to create a per-browser "fingerprint"
  - This can be used for tracking
  - Often detectable on the client
- No way to learn about these issues
  - We need to know the site
  - But browsing history is sensitive
- Problem statement: collect the sites where the client sees issues

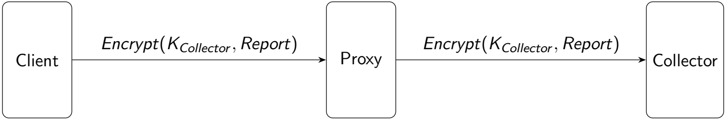
Privacy Threats

- Tying sensitive data directly to identifying information
    - Directly via user identifiers (E-mail, cookies, etc.)
    - Indirectly via metadata (IP address, E.164 number, etc.)
- Collecting sensitive data along with non-sensitive identifying information
    - Example: (birthday, zip code, initials) $\rightarrow$ income

It was found that 87% (216 million of 248 million) of the population in the United States had reported characteristics that likely made them unique based only on 5-digit ZIP, gender, date of birth. — Sweeney, 2014 [Swe00]

Anonymized Data Collection with OHAI

- Basic idea: collect user information without identifiers
- Practically speaking
  - Strip direct identifiers on the client side
  - Strip metadata using a proxy



- Example technologies:
  - Connection-level proxies (IPsec, RFC 2817 CONNECT, MASQUE)
  - Application-level proxies (OHAI)

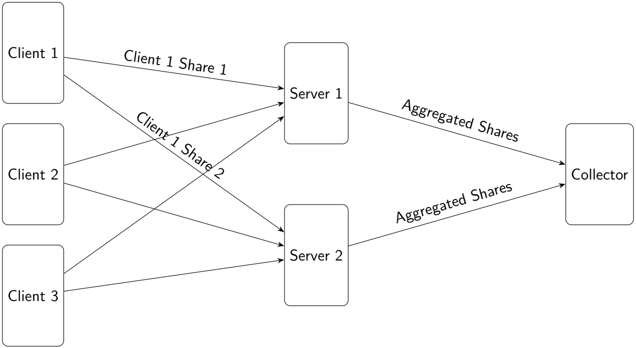
Good Use Cases for Anonymization

- Boosting the privacy of semi-sensitive data
  - Important: this requires that the proxy and server do not collude
  - Example: existing browser Telemetry is done with no privacy
- Individual values where you don't need to "dig into" the data
- Freeform data
  - E.g., JSON blobs
- Anything that needs an answer
  - DNS requests
  - Safe Browsing queries

Bad Use Cases for Anonymization

- High dimensionality data (statistical queries)
  - Multiple variables that need to be reported together
  - When you want to look at subgroups
  - Any time you want to do correlation/regression
  - *Anonymized data needs to be disaggregated to prevent de-anonymization*
- Collecting common values (heavy hitters)
  - The "top N" values common $> t$ users
  - *Anonymized data collects every value and depends on reporting only common values*

Cryptography to the Rescue



Solutions to both statistical queries and heavy hitters share a common framework:

- Split data between two servers
- Each server computes aggregated shares
- Aggregate shares combined to produce final value

Trust Model

- Client's requirement: The two servers do not collude
    - If they do, they can compute individual values
    - The servers enforce minimum batch sizes and query limits
- Collector's requirement: Both servers execute the protocol correctly
    - Either server can distort the results
- Difficult to verify this
    - Collusion can happen through side channels
    - If traffic goes through the leader, helper can just share keys
    - Point-in-time audits are possible

Cryptographic Protocol: Prio [CGB17]

- Useful for computing numeric aggregates (sum, mean, etc.)
- Each client i holds a value x_i, then secret shares it with each server
  - Generates random $R_i \leftarrow \mathbb{F}_p$
  - Sends $x_i - R_i (mod p)$ to server 1
  - Sends R_i to server 2
- Each server adds up their shares
  - Server 1: $\sum_i x_i - R_i$
  - Server 2: $\sum_i R_i$
- Now add these up:
  $\sum_i x_i - R_i + \sum_i R_i = \sum_i x_i + \sum_i R_i - \sum_i R_i = \sum_i x_i$

What else can Prio compute?

Arithmetic mean	$\sum_i x_i/i$
Product	$exp(\sum_i log(x_i))$
Geometric mean	From product
Variance and stddeviation	From $\sum_i x_i$ and $\sum_i (x_i)^2$
Boolean OR, AND	...
MIN, MAX	...
Ordinarily least squares (OLS)	...

The trick is finding the right encoding

What about bogus data?

- Plausible but false
  - "I am 180cm tall" when I am actually 175cm
  - A problem with any surveying technique
  - Solution: live with somewhat noisy data
- Completely ridiculous
  - "I am 1km tall" (or worse, "I am -1km tall")
  - Easy to remove with standard systems by filtering
    - ... but with Prio the data is encrypted
  - Solution: each submission comes with a zero-knowledge proof of validity
    - "This height report is between 100 and 200cm"
    - Servers work together to validate the proof
    - Only aggregate submissions with valid proofs

Collecting User Interests with Prio

- Each user interest gets a bucket
- Client reports the time T spent in each bucket (including 0s) with Prio
- Use Prio to sum them up[1]
- Servers only learn aggregates, not values for each category

[1] Bonus: report T^2 to compute standard deviation

Cryptographic Protocol: Heavy Hitters ("Hits") [BBCG+21]

- Each client submits a string (e.g., a URL)
  - Report the N most frequent strings
- Servers jointly can compute the number of strings with prefix p
  - Can use binary search to compute the most common strings
  - "How many strings have prefix $p||0$ versus $p||1$

Collecting Broken Sites with Hits

- Client creates one report for each site which is broken
- Use Hits to determine the top sites
- Servers only learn the most important sites, not who reported them

Subset Queries

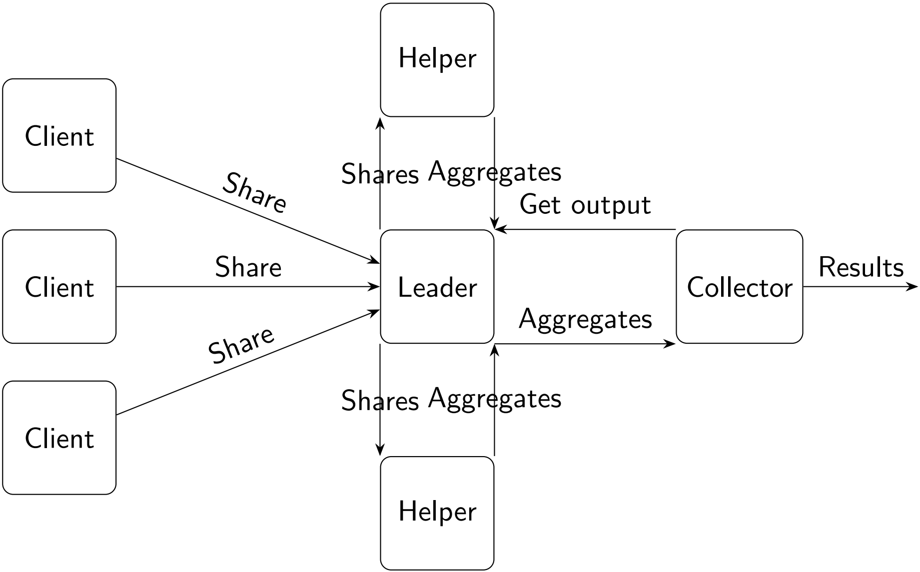
- Submissions can be tagged with demographic data
  - Example: (birthday, zip code, initials) $\rightarrow$ Encrypted(income)
  - This is safe because the sensitive information is encrypted
  - Servers can then compute aggregates over subsets
- Repeated queries can be used to determine individual values
  - Querying for S and $S \setminus I$ reveals I's value
  - Defenses
    - Minimum batch size
    - Anti-replay
    - Differential privacy randomization

Privacy Preserving Measurement Protocol

draft-gpew-priv-ppm-00

- A generic, modular, protocol for privacy-preserving measurement
    - Initially implementing Prio and Hits
    - Compatible with multiple cryptographic algorithms ("verifiable distributed aggregation functions" – see CFRG presentation for details)
- Build on top of HTTPS
    - Easy to implement with existing services infrastructure

PPM System Architecture



Questions?

Dan Boneh, Elette Boyle, Henry Corrigan-Gibbs, Niv Gilboa, and Yuval Ishai.
Lightweight techniques for private heavy hitters.
Cryptology ePrint Archive, Report 2021/017, 2021.
https://eprint.iacr.org/2021/017.

Henry Corrigan-Gibbs and Dan Boneh.
Prio: Private, robust, and scalable computation of aggregate statistics.
In *14th USENIX Symposium on Networked Systems Design and Implementation (NSDI 17)*, pages 259–282, Boston, MA, March 2017. USENIX Association.

Latanya Sweeney.
Simple demographics often identify people uniquely.
Health (San Francisco), 671(2000):1–34, 2000.