ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

# Anonymization

## COM-402: Information Security and Privacy

(Stevens le Blond and Linus Gasser)

# Outline

- **Motivation**
- Naive Anonymization
- k-anonymity
- Differential Privacy
- Confidentiality using Encryption
  - Property-Preserving Encryption - CryptDB
  - Building on Property-revealing EncrypTion - BoPET

# Anonymity - Privacy - Security




Source: https://highspeedexperts.com/online-security-privacy/anonymity-vs-privacy-vs-security/

# Anonymity - Privacy - Security

- Anonymity
  - This is essentially when you opt to have your online actions seen, but keep your identity hidden. In short, “they” can see what you do, but not who you are.
- Privacy
  - This primarily involves you controlling who (if anyone) sees what activities you engage in online. In other words, “they” can see who you are, but not what information or websites you access or seek out.
- Security
  - You give up privacy and anonymity towards a third party, but they should keep this information to themselves. E.g., internet security involves you’re being safeguarded while browsing sites or filling in a Web form.

Source: https://highspeedexperts.com/online-security-privacy/anonymity-vs-privacy-vs-security/

# Speakup

Room 71839: http://web.speakup.info/ng/room/58874fd89b3943d8ef7680c5

# Speakup - Anonymity

When do you mostly need **anonymity**?

A. When surfing on the web
B. Whistleblowing about a work situation
C. Watching a movie in streaming
D. Posting a dataset of PII data

# Speakup - Privacy

When do you mostly need **privacy**?

A. Going to a doctor
B. Buying something online
C. When surfing the web
D. Chatting with a friend

# Speakup - Security

When do you need **security**?

A. Watching a movie in streaming
B. If you have an implanted medical device
C. When surfing the web
D. Buying something online

# Growing importance of data anonymization

Examples of plaintext data:

- Humanitarian data (e.g., International Committee of the Red Cross)
- Bug reports (e.g., Microsoft)
- Search queries (e.g., AOL)
- Viewing habits (e.g., Netflix)

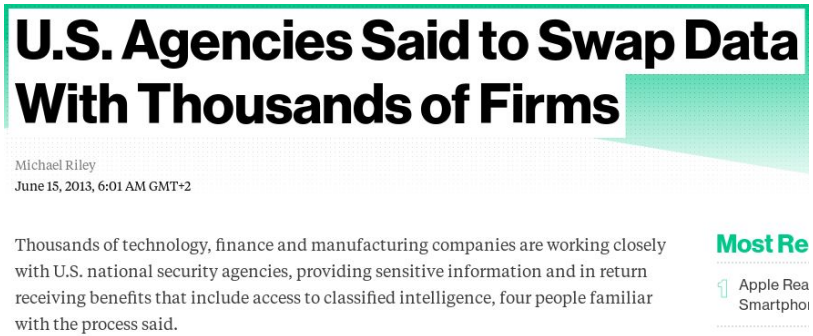
Example of encrypted data:

- Anonymous traffic (e.g., VPN, Tor)

# Accidental leakage of private data can have catastrophic consequences

Intelligence agencies exploited:

- Humanitarian data to track the whereabouts of Bin Laden
- Microsoft bug reports to identify zero-day vulnerabilities

# Naive anonymization can easily be broken

In 2006, AOL released search queries of 500,000 pseudonymous users

- Days later, New York Times reveals the identity of user 4417749
- CTO and researchers responsible of sharing data fired

The same year, Netflix revealed over 100 million movie ratings made by 500,000 users after removing personal details

- Researchers de-anonymize dataset by comparing it with publicly available ratings on Internet Movie Database (IMDB)

[Credit: New York Times]

# AOL Proudly Releases Massive Amounts of Private Data

Posted Aug 6, 2006 by **Michael Arrington** (@arrington)

**Yet Another Update:** AOL: "This was a screw up"

**Further Update:** Sometime after 7 pm the download link went down as well, but there is at least one mirror site. AOL is in damage control mode – the fact that they took the data down shows that someone there had the sense to realize how destructive this was, but it is also an admission of wrongdoing of sorts. Either way, the data is now out there for anyone that wants to use (or abuse) it.

**Update:** Sometime around 7 pm PST on Sunday, the AOL site referred to below was taken down. The direct link to the data is still live. A cached copy of the page is here.

AOL must have missed the uproar over the DOJ's demand for "anonymized" search data last year that caused all sorts of pain for Microsoft and Google. That's the only way to explain their release of data that includes 20 million web queries from 650,000 AOL users.

The data includes all searches from those users for a three month period this year, as well as whether they clicked on a result, what that result was and where it appeared on the result page. It's a 439 MB compressed download, expanded to just over 2 gigs. The data is available here (this link is directly to the file) and the output is in ten text files, tab delineated.

**The utter stupidity of this is staggering.** AOL has released very private data about its users without their permission. While the AOL username has been changed to a random ID number, the abilitiy to analyze all searches by a single user will often lead people to easily determine who the user is, and what they are up to. The data includes personal names, addresses, social security numbers and everything else someone might type into a search box.

## Crunchbase

**AOL** –

FOUNDED
1985

OVERVIEW
AOL Lifestream is a web-based application that enables users to keep track of all their comments on social networking sites. Integrated with AIM Express, AIM 7, and AIM for Mac, users can publish their statuses, reply to comments on networking sites from their Lifestream tab, and more. AOL Lifestream is a product of [AOL] (https://www.crunchbase.com/organization/aol#/entity)

LOCATION
New York, NY

CATEGORIES
Digital Media, Advertising Platforms, Content Creators, News

WEBSITE
http://www.aol.com

Full profile for AOL

# Metadata also leak important information about nature of encrypted traffic

Analysis of time series of encrypted packets can for example reveal:

- Skype caller/callee
- Netflix content with 99.99% accuracy
- Source IP address of VPN and Tor users

# Outline

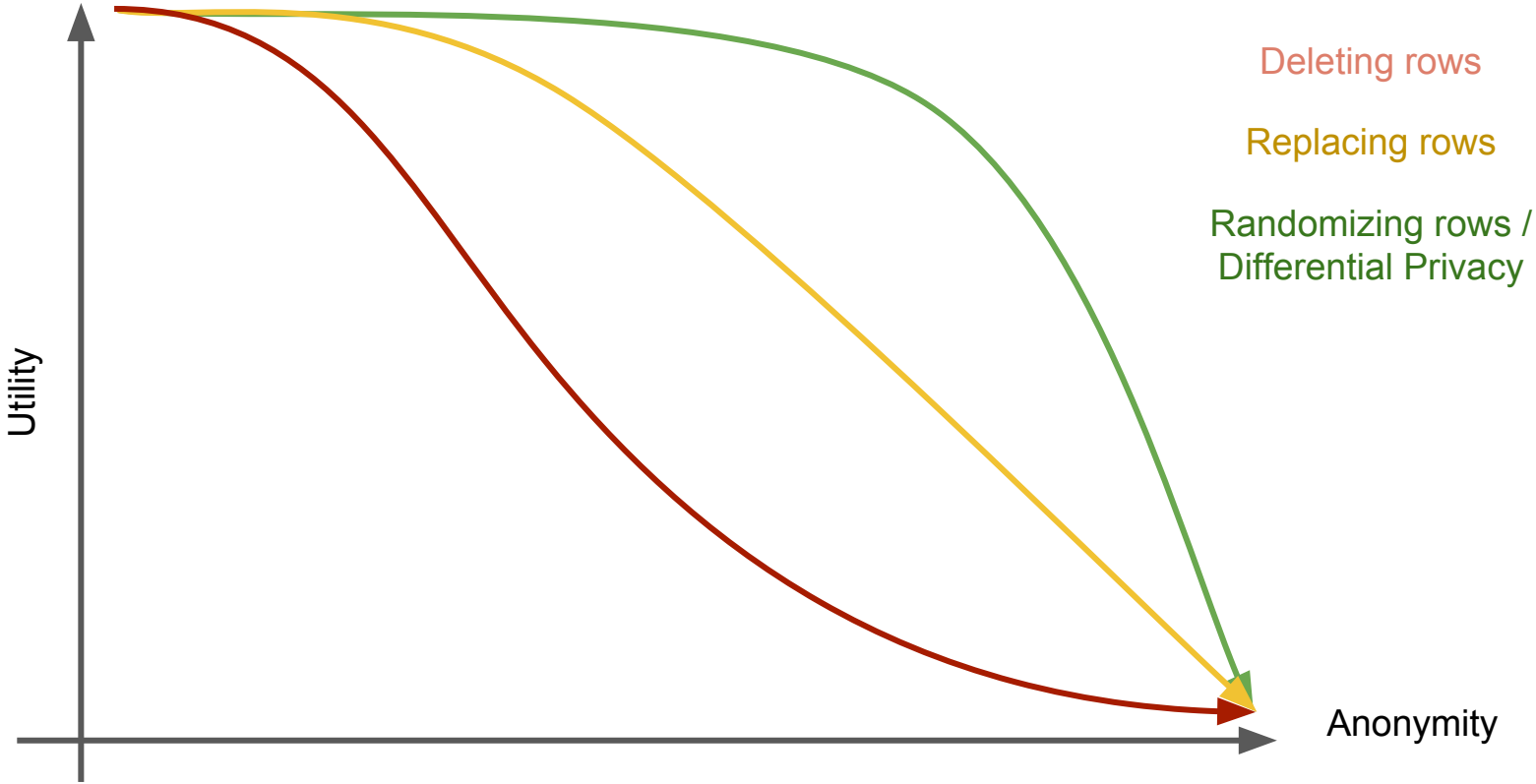
- Motivation
- **Naive Anonymization**
- k-anonymity
- Differential Privacy
- Confidentiality using Encryption
  - Property-Preserving Encryption - CryptDB
  - Building on Property-revealing EncrypTion - BoPET

# Definition

Wikipedia on *Data anonymization*

> *Technology that converts clear text data into a non-human-readable and **irreversible** form, including preimage resistant hashes (e.g., one-way hashes) and encryption techniques in which the decryption key has been discarded."*
>
> *Data anonymization enables the **transfer of information across a boundary**, such as between two departments within an agency or between two agencies, while reducing the risk of unintended disclosure, and in certain environments in a manner that enables evaluation and analytics post-anonymization.*

# Anonymization 101

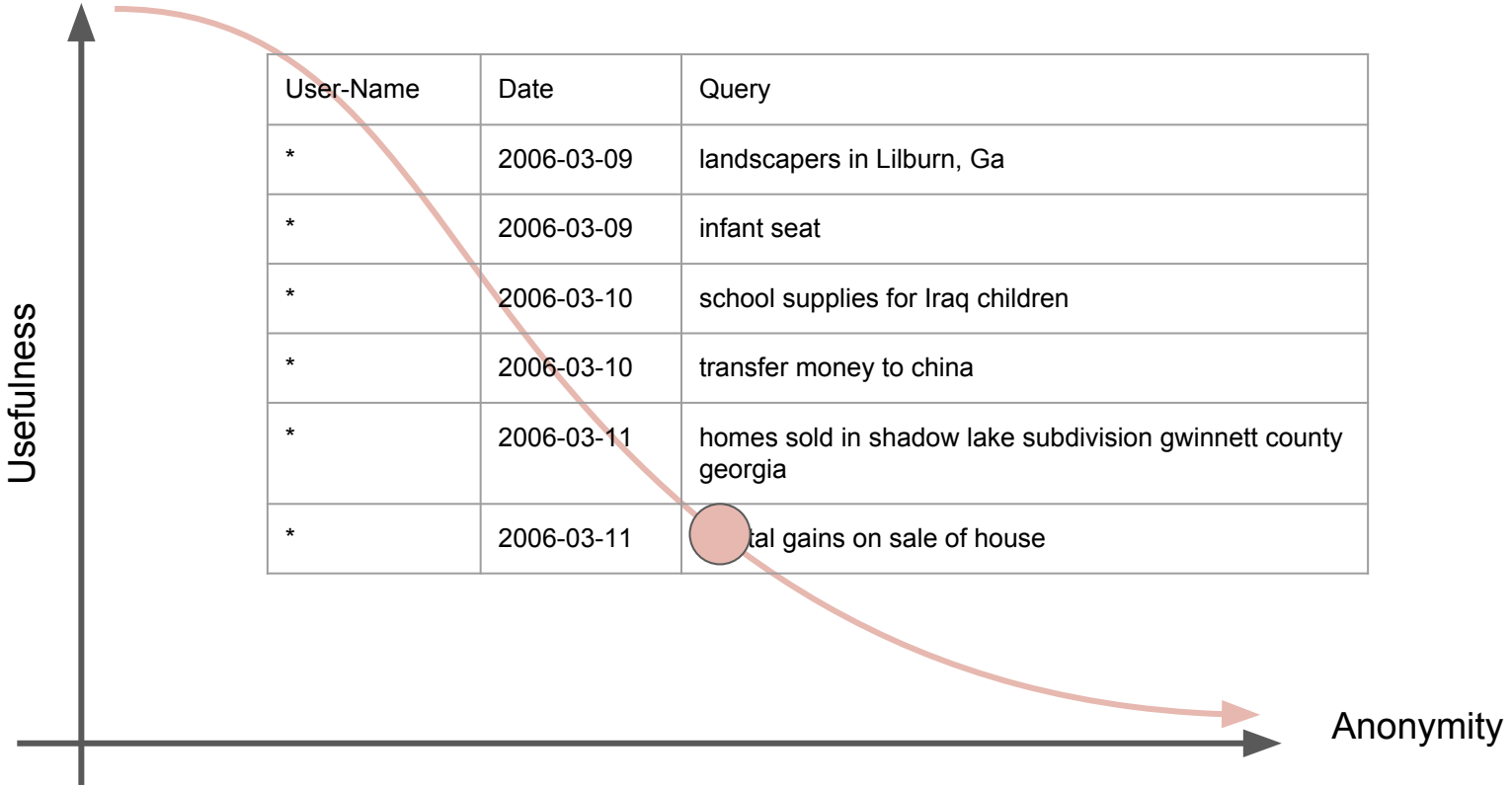


# Part of Original AOL Dataset

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

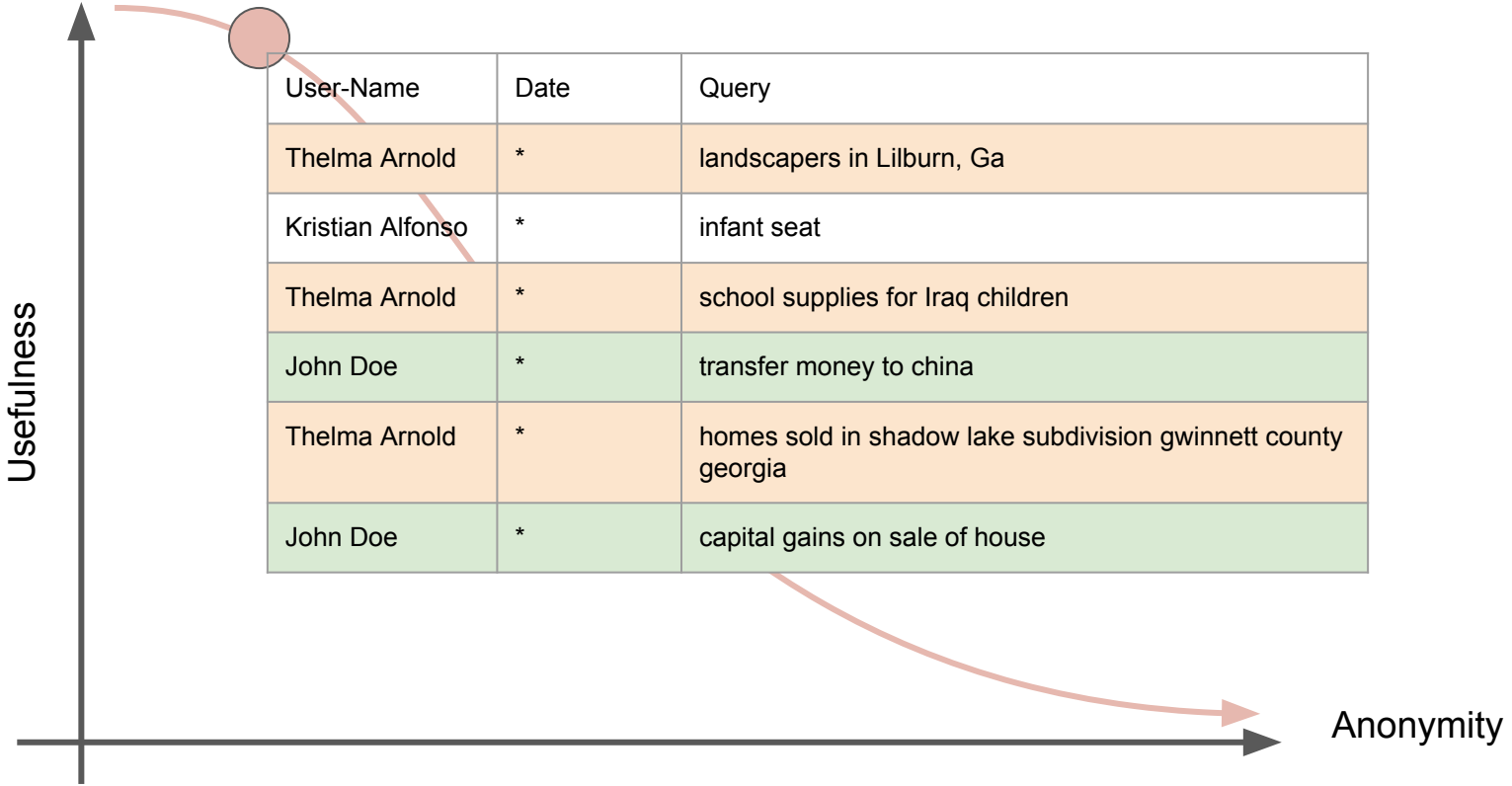
| User-ID | Date | Query |
|---|---|---|
| Thelma Arnold | 2006-03-09 | landscapers in Lilburn, Ga |
| Kristian Alfonso | 2006-03-09 | infant seat |
| Thelma Arnold | 2006-03-10 | school supplies for Iraq children |
| John Doe | 2006-03-10 | transfer money to china |
| Thelma Arnold | 2006-03-11 | homes sold in shadow lake subdivision gwinnett county georgia |
| John Doe | 2006-03-11 | capital gains on sale of house |

# Deleting row of user-names



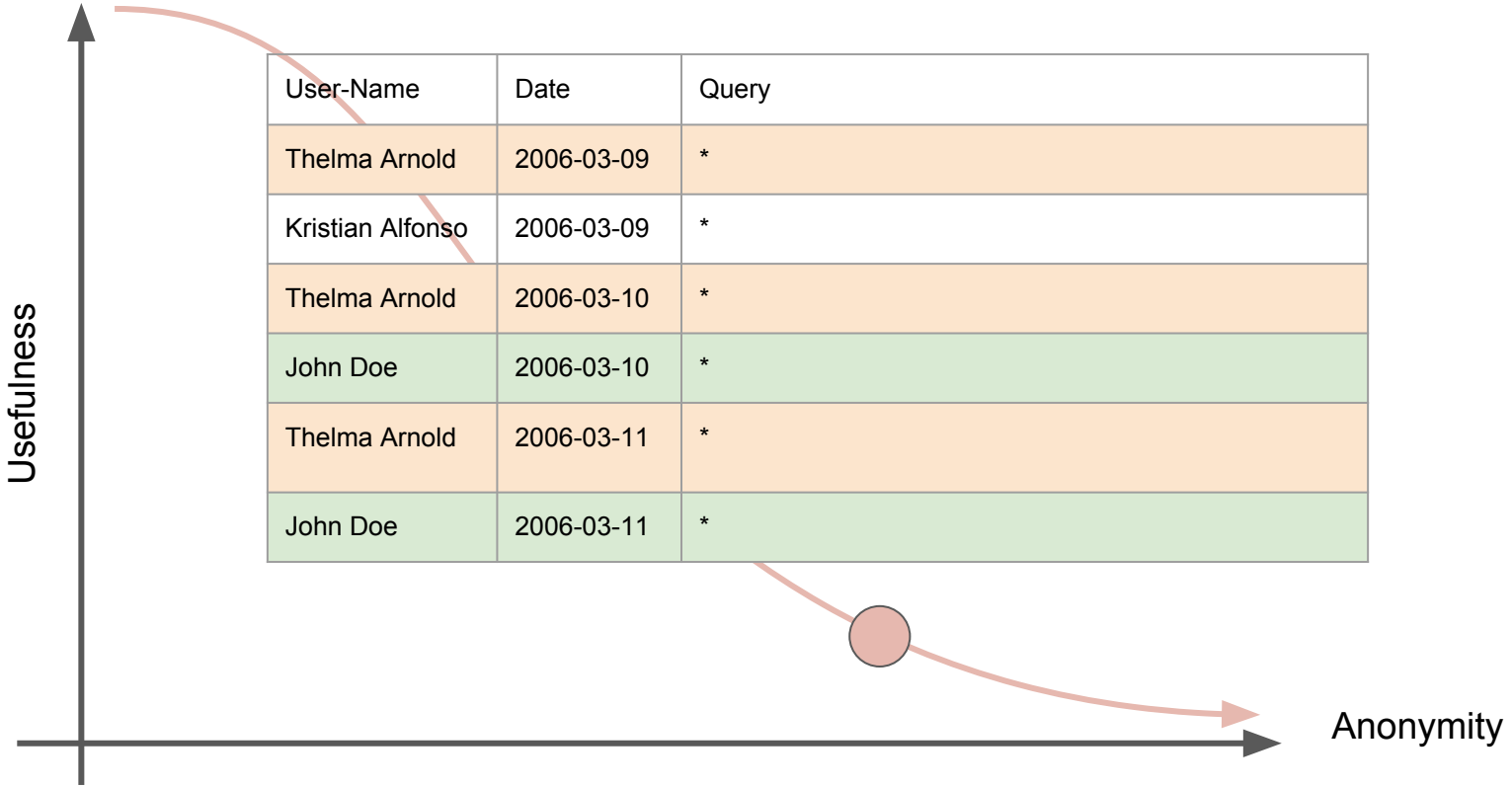

| User-Name | Date | Query |
|---|---|---|
| * | 2006-03-09 | landscapers in Lilburn, Ga |
| * | 2006-03-09 | infant seat |
| * | 2006-03-10 | school supplies for Iraq children |
| * | 2006-03-10 | transfer money to china |
| * | 2006-03-11 | homes sold in shadow lake subdivision gwinnett county georgia |
| * | 2006-03-11 | tal gains on sale of house |

# Remove row of Date



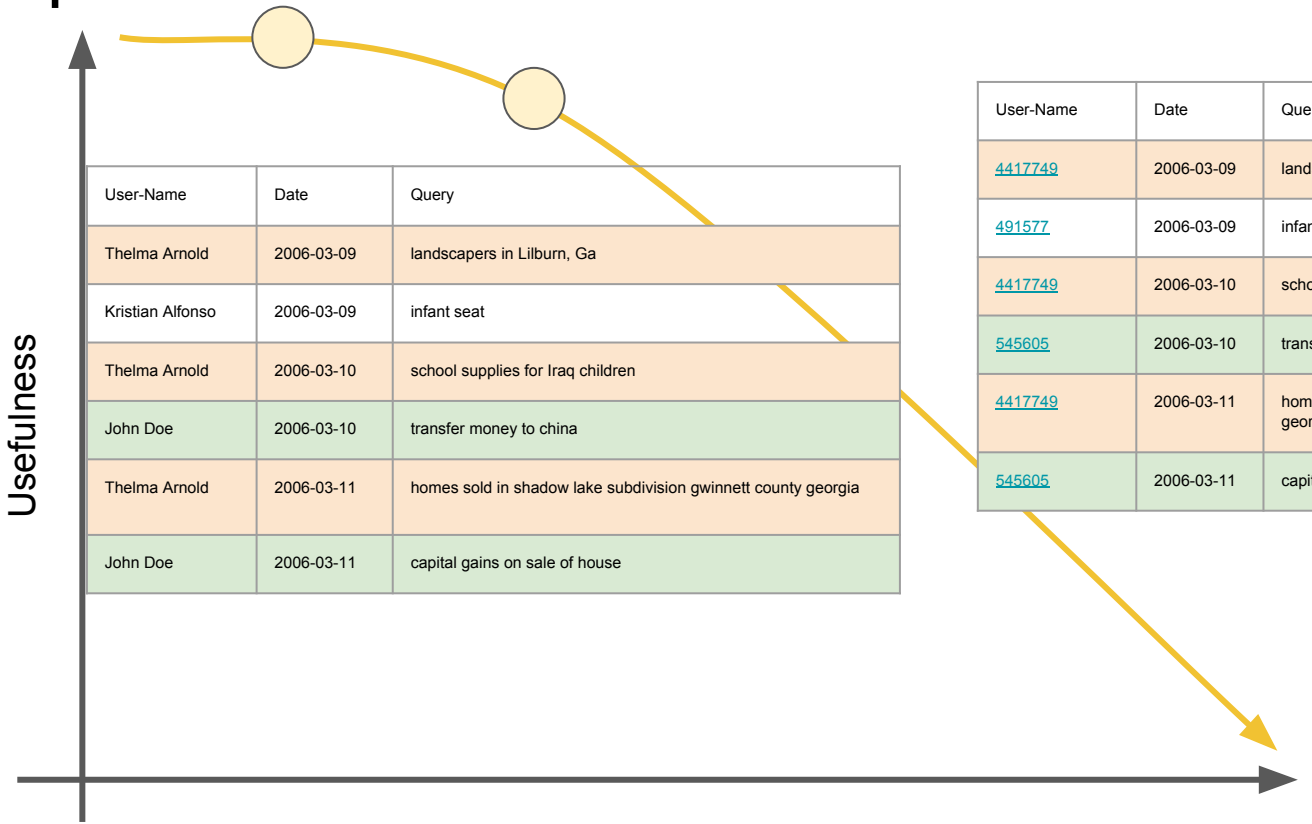

| User-Name | Date | Query |
|---|---|---|
| Thelma Arnold | * | landscapers in Lilburn, Ga |
| Kristian Alfonso | * | infant seat |
| Thelma Arnold | * | school supplies for Iraq children |
| John Doe | * | transfer money to china |
| Thelma Arnold | * | homes sold in shadow lake subdivision gwinnett county georgia |
| John Doe | * | capital gains on sale of house |

# Remove row of Query




| User-Name | Date | Query |
|---|---|---|
| Thelma Arnold | 2006-03-09 | * |
| Kristian Alfonso | 2006-03-09 | * |
| Thelma Arnold | 2006-03-10 | * |
| John Doe | 2006-03-10 | * |
| Thelma Arnold | 2006-03-11 | * |
| John Doe | 2006-03-11 | * |

# Replace User-Names with Random ID

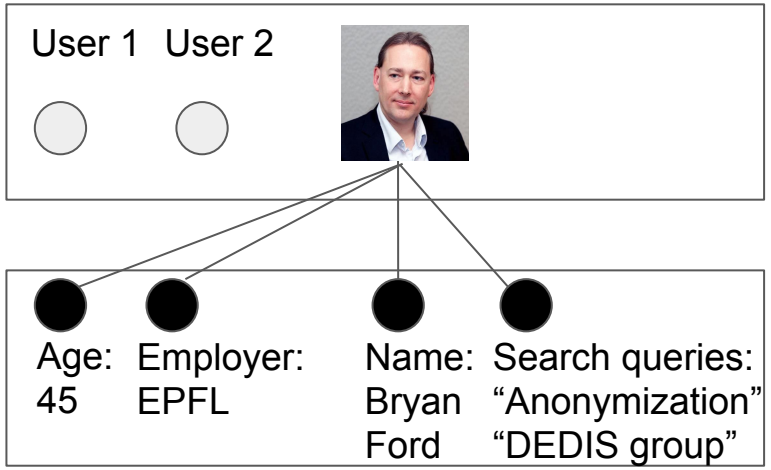



| User-Name | Date | Query |
|---|---|---|
| Thelma Arnold | 2006-03-09 | landscapers in Lilburn, Ga |
| Kristian Alfonso | 2006-03-09 | infant seat |
| Thelma Arnold | 2006-03-10 | school supplies for Iraq children |
| John Doe | 2006-03-10 | transfer money to china |
| Thelma Arnold | 2006-03-11 | homes sold in shadow lake subdivision gwinnett county georgia |
| John Doe | 2006-03-11 | capital gains on sale of house |

| User-Name | Date | Query |
|---|---|---|
| 4417749 | 2006-03-09 | landscapers in Lilburn, Ga |
| 491577 | 2006-03-09 | infant seat |
| 4417749 | 2006-03-10 | school supplies for Iraq children |
| 545605 | 2006-03-10 | transfer money to china |
| 4417749 | 2006-03-11 | homes sold in shadow lake subdivision gwinnett county georgia |
| 545605 | 2006-03-11 | capital gains on sale of house |

# For most practical purposes, anonymization consists of unlinking users from their attributes

General problem: Given a bipartite graph with links between users and attributes, how to effectively unlink the two without losing “utility?”

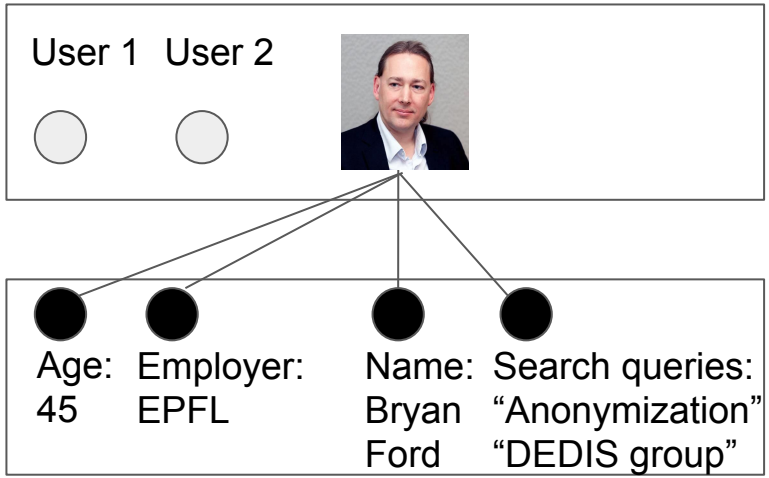


# Strawman 1: Obfuscate users' identities

General problem: Given a bipartite graph with links between users and attributes, how to effectively unlink the two without losing "utility?"

Let's make users pseudonymous!

I can still use their attributes!

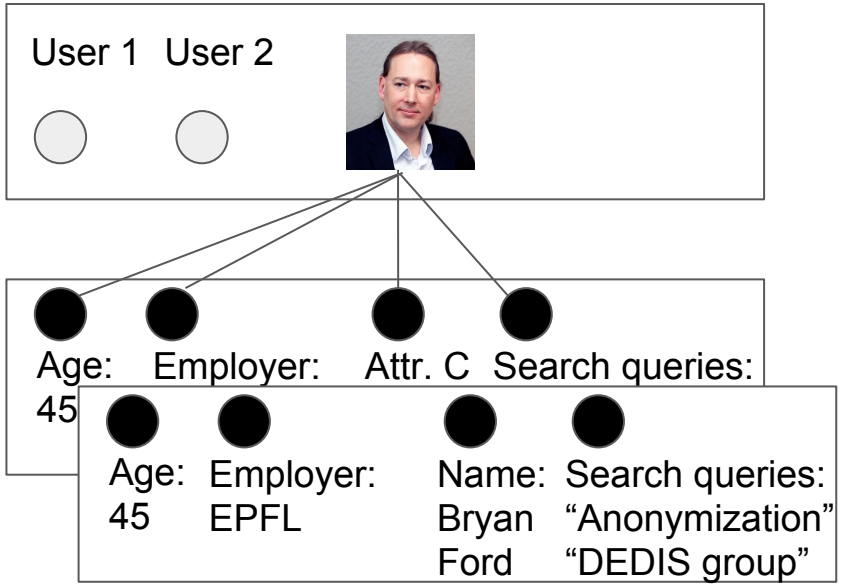


# Strawman 2: Obfuscate users' identities and some attributes

General problem: Given a bipartite graph with links between users and attributes, how to effectively unlink the two without losing "utility?"

Let's also obfuscate some attributes!

I can still correlate them with public data!



# Strawman 3: Aggregation

General problem: Given a bipartite graph with links between users and attributes, how to effectively unlink the two without losing “utility?”

OK, let’s give up on the graph...

You now have limited utility!

| Age: | Employer: | Search queries: |
|---|---|---|
| <10 : 0 | EPFL: 900 | “Anonymization”: 10 |
| 10-20: 0 | | |
| 20-30: 200 | | |
| 30-40: 200 | | |
| 40-50: 200 | | |
| 50-60: 200 | | |
| >60 : 100 | | |

# Other Data that Should be Anonymous

Previous anonymizations can be applied to:

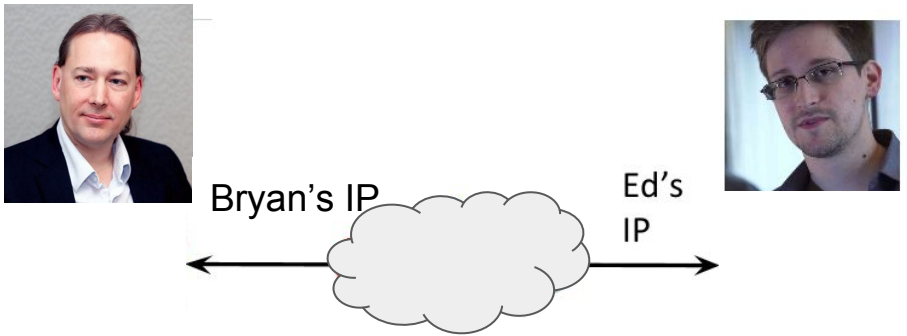
- User-data, log files
- Public records of hospitals

Other techniques apply to networks:

- Onion-routing: Tor

# Problem: Communications on the Internet are not Anonymous

Given a simple graph with links between communicating users, how to effectively unlink users without losing "utility?"



# Strawman 4: Obfuscate the IPs

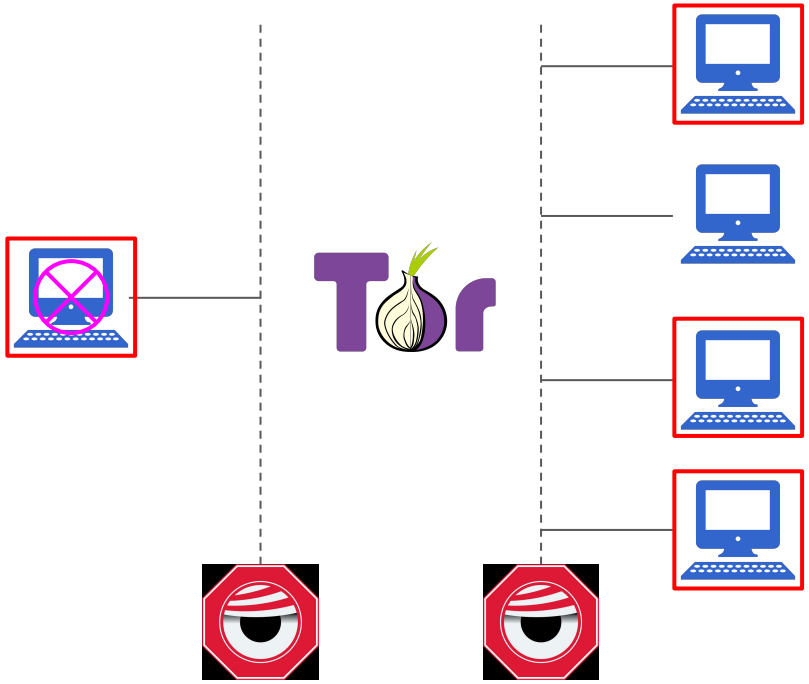
Given a simple graph with links between communicating users, how to effectively unlink users without losing "utility?"

Let's obfuscate IPs using a series of relays!

# Strawman 4': Example of intersection attack

Given a simple graph with links between communicating users, how to effectively unlink users without losing "utility?"

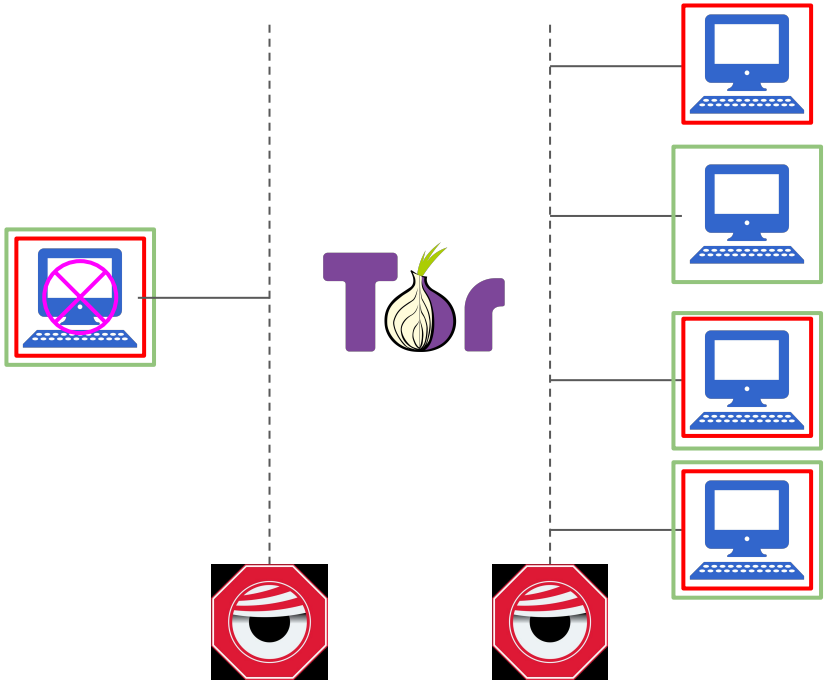
Let's obfuscate IPs using a series of relays!



# Strawman 4': Example of intersection attack

Given a simple graph with links between communicating users, how to effectively unlink users without losing "utility?"

Let's obfuscate IPs using a series of relays!



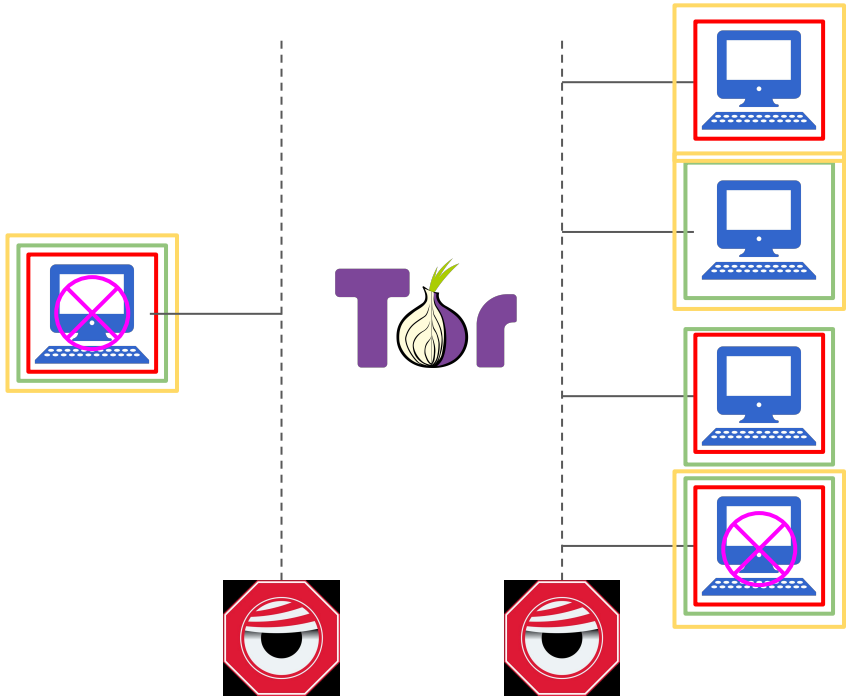

2nd call

# Strawman 4': Example of intersection attack

Given a simple graph with links between communicating users, how to effectively unlink users without losing "utility?"

Let's obfuscate IPs using a series of relays!

I can still correlate availability and/or time series of encrypted packets!




3rd call

# FBI agents tracked Harvard bomb threats despite Tor

74

by Russell Brandom | @russellbrandom | Dec 18, 2013, 12:55pm EST

SHARE TWEET LINKEDIN


via farm1.staticflickr.com

This week, Harvard was rocked by an unsigned bomb threat, originating from a burner email address and timed to disrupt final exams. It was a seemingly anonymous threat, but just two days later, authorities managed to trace it back to sophomore Eldo Kim, who's now awaiting trial in federal court. Kim used two separate anonymity tools to cover his tracks — the routing service Tor, which covered his web traffic, and the temporary mail service Guerrilla Mail, which offered a one-time email — but neither one was enough to throw authorities off the trail.

Kim's mistake, it turns out, was connecting through Harvard's wireless network. The FBI quickly traced the emails back to Guerrilla Mail, which in turn indicated that the service had been accessed through Tor. Security researcher Runa Sandvik points out that the originating IP address would have been revealed in the email header, which would have indicated Tor usage. Suspecting a Harvard student was behind the threats, agents checked to see if anyone had accessed Tor through the local wireless networks. That led them to Kim, who promptly confessed.

# Outline

- Motivation
- Naive Anonymization
- **k-anonymity**
- Differential Privacy
- Confidentiality using Encryption
  - Property-Preserving Encryption - CryptDB
  - Building on Property-revealing EncrypTion - BoPET

# k-anonymity

- k-Anonymity: attributes are suppressed or generalized until each user is identical with at least k-1 other users.
- k-Anonymity thus prevents definite database linkages. At worst, the data released narrows down an individual entry to a group of k users.
- Methods for achieving k-anonymity
  - Suppression – can replace individual attributes with a *
  - Generalization – replace individual attributes with a broader category Example: (Age: 26 => Age: [20-30])
  - Randomization - add a random value with a gaussian distribution to the Age

# Example

The following database:

| First | Last | Age | Race |
|---|---|---|---|
| Harry | Stone | 34 | Afr-Am |
| John | Reyser | 36 | Cauc |
| Beatrice | Stone | 34 | Afr-Am |
| John | Delgado | 22 | Hisp |

Can be 2-anonymized with suppression as follows:

| First | Last | Age | Race |
|---|---|---|---|
| * | Stone | 34 | Afr-Am |
| John | * | * | * |
| * | Stone | 34 | Afr-Am |
| John | * | * | * |

Note:
Rows 1 and 3 are identical and
Rows 2 and 4 are identical

# Outline

- Motivation
- Naive Anonymization
- k-anonymity
- **Differential Privacy**
- Confidentiality using Encryption
  - Property-Preserving Encryption - CryptDB
  - Building on Property-revealing EncrypTion - BoPET

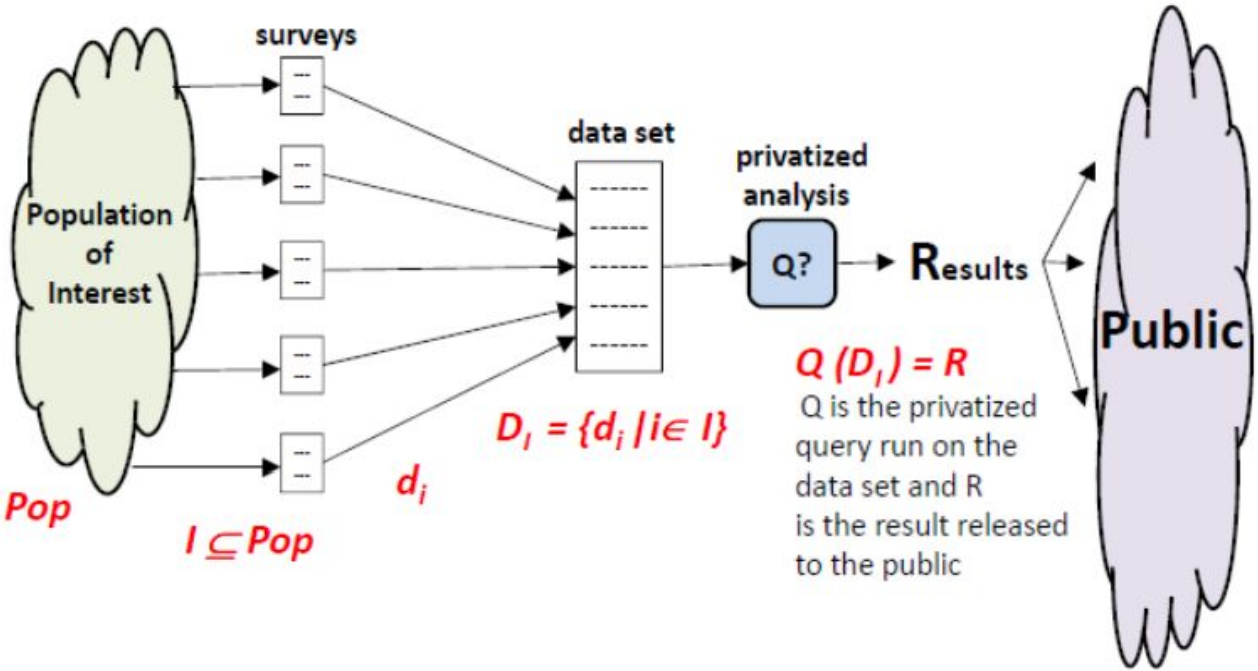
# Differential Privacy

Given a survey like

1. Do you use streaming to watch movies?
2. If yes, how many movies did you stream this year?
3. What is your gender?
4. What is your age?

What would make you feel safe to participate in that survey?

# Notations



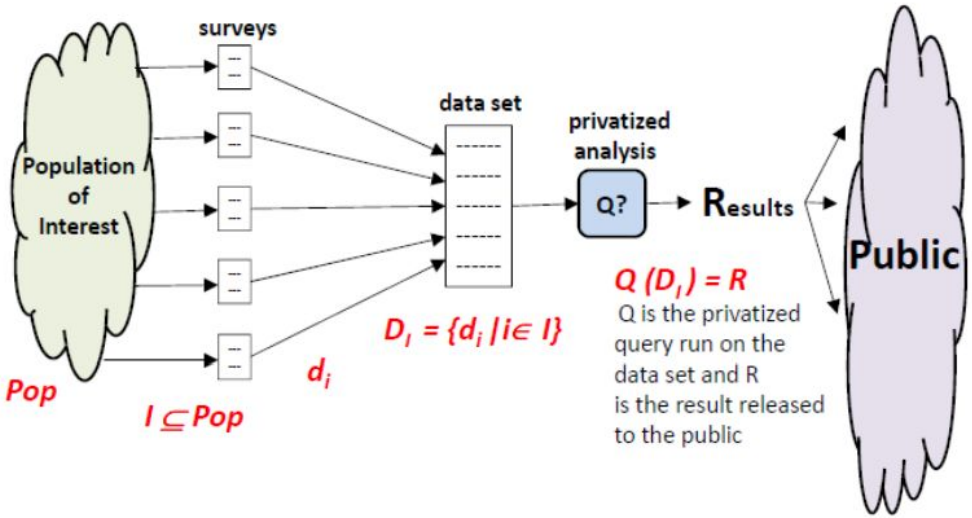

Wang Yuxiang
https://www.cs.cmu.edu/~yuxiangw/docs/Differential%20Privacy.pdf

# I would feel safe if:

1. My answers would not impact the results:
   a. $Q(D_{I\text{-}me}) = Q(D_I)$

2. An attacker looking at the results R could not learn anything about me
   a. $P(secret(me) \mid R) = P(secret(me))$

Does that make sense?



# Does it work?

1. If the answers don't affect the result, there's no point in the survey:
   a. $Q(D_{I\text{-}me}) = Q(D_I) \rightarrow Q(D_I) = Q(D_\varnothing)$
2. If there's a strong trend in the population, it will probably reflect in me, too:
   a. Prob(secret(me) | secret(pop)) > Prob(secret(me))
      If I'm 25 and male, and most men between 20 and 30 use streaming, I'm using it probably, too.
3. If the attacker knows a *relation* between me and the general population, he'll be able to infer the absolute data:
   a. I'm twice the average age of streaming-users, and the average age is 25 -> I'm 50!
      This works even if I don't participate in the survey!

Under the previous assumptions I cannot feel safe, even without participating!

# I would feel safe - 2nd try

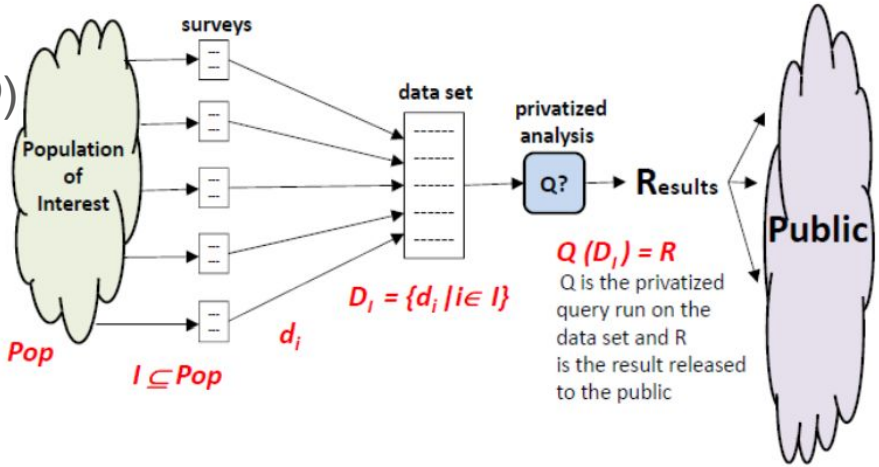
If I knew the chance that the privatized released result would be R was nearly the same, whether or not I submitted my information.

Definition: $\epsilon$-Differential Privacy

$$\frac{Pr(Q(D)=R)}{Pr(Q(D_{+-i})=R)} < e^{\epsilon}$$

For any $|D_{\pm i} - D| \leq 1$ and any $R \in Range(Q)$

$\epsilon$ is called the *privacy budget*. It is linked to the amount of information you release. A small $\epsilon$ protects the data, while a big $\epsilon$ gives away more data.



# Differential Privacy

- Limits the harm of the results to the results themselves
- Ensures the released results give minimal evidence whether any given individual participated or not
- If individuals only provide information about themselves and not others, it protects PII to the chosen level $\epsilon$

# DP applies to

- Approved information gathering
  - Surveys submitted by users
  - Medical data
- Information gathered automatically
  - Bug-reports
  - Search-queries
  - Viewing habits

Apple is using it to “Collect Your Data—But Not Your Data”. Others will follow.

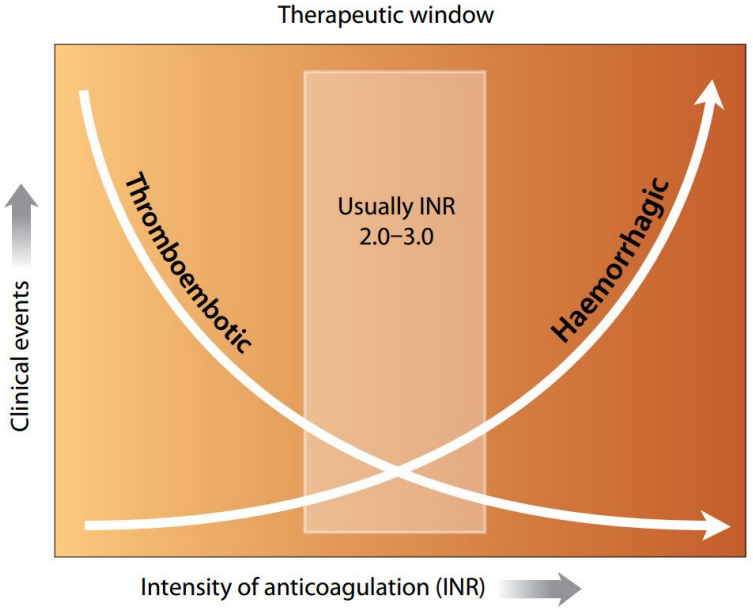
# DP at the Example of Warfarin

Warfarin is used to thin blood, which

- kills rats by bleeding them to death
- prevents strokes for people with illness

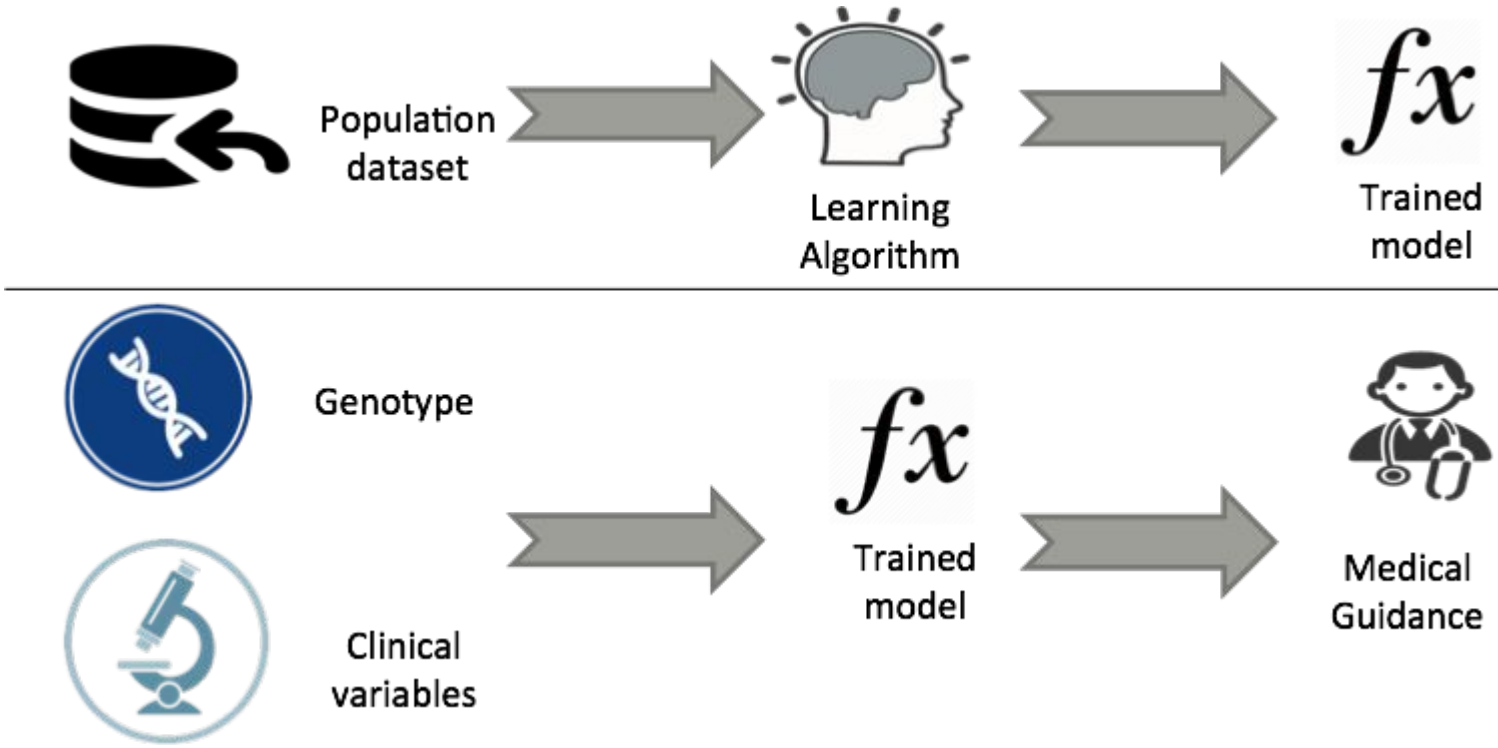
Finding the right level of Warfarin-dosage is difficult:

- not enough -> patient might get another stroke
- too much -> the patient might bleed internally



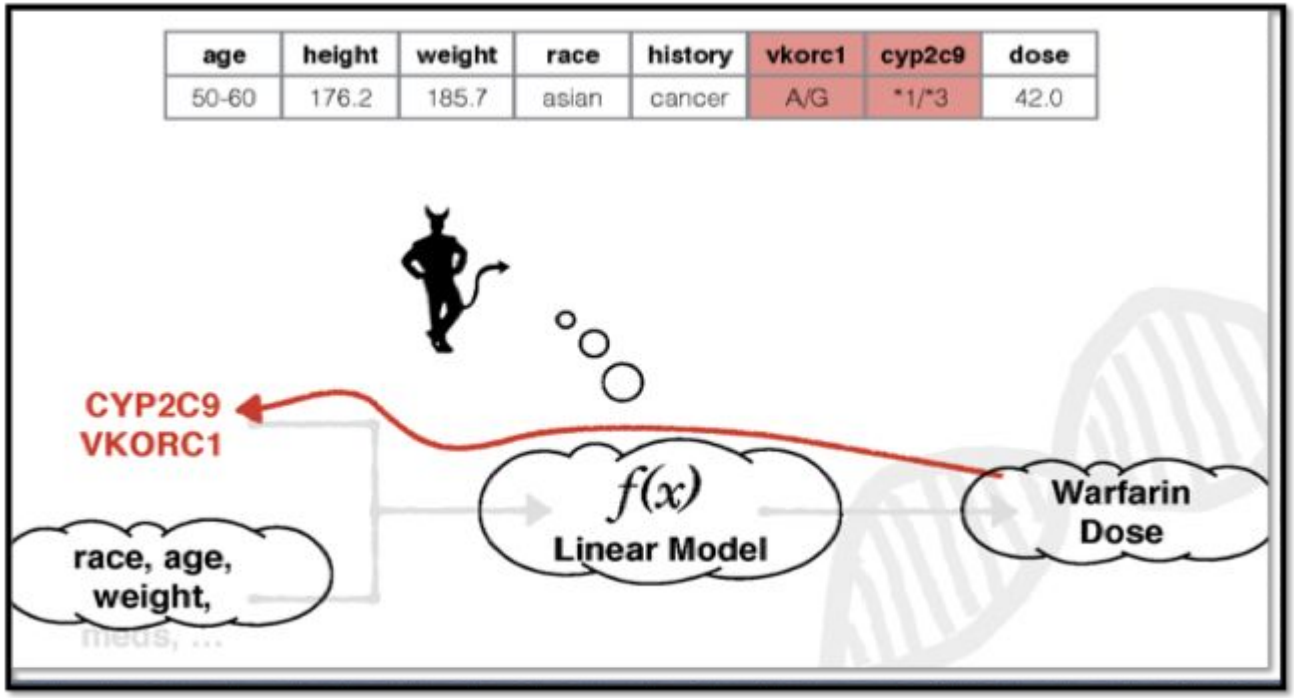

http://www.bpac.org.nz/BT/2010/November/inr.aspx

# How Machine Learning can Help




https://www.slideshare.net/alanoudsalqoufi/privacy-in-pharmacogenetics

# But the Model can be Inverted



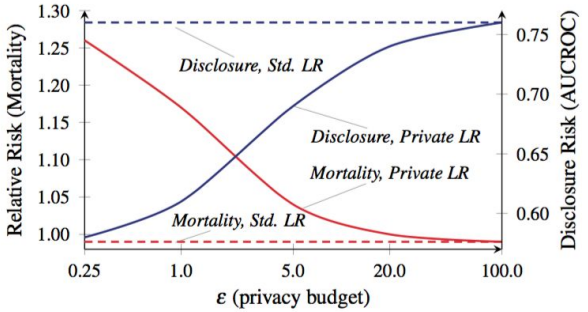

https://www.slideshare.net/alanoudsalgoufi/privacy-in-pharmacogenetics

# Differential Privacy to the Rescue

- Introducing some error to the model
+ Makes it more difficult to invert the model
- The dosage might be wrong and kill patients

Best paper award from Usenix Security 2014 shows:

- This much privacy kills this many people!




Even though there is a catch: is it really a problem if you can go from the Warfarin dosage to some knowledge about genetic phenotypes?

# Speakup - Differential Privacy

Which of the following might be made more private using Differential Privacy?

A. An online survey
B. Documents stored on the cloud
C. Automatic bug reports
D. Statistics on web usage

# Overview

- Motivation
- Naive Anonymization
- k-anonymity
- Differential Privacy
- **Confidentiality using Encryption**
  - **Property-Preserving Encryption - CryptDB**
  - Building on Property-revealing EncrypTion - BoPET

# Property-Preserving Encryption

- Used mostly in databases
- Leaves some information intact for
  - Indexing (equality tests)
  - Range queries (inequality tests)
  - Data clustering
  - Keyword search
  - General computations
- Can also be used for anonymization, as long as the key is not made public

# PPE setup

A property P is a function of arity *k* that returns 0 or 1:

$$P(m_1, m_2,...,m_k) = \{0,1\}$$

PPE can be used for different properties:

- OPE - Order Preserving Encryption
- DET - Deterministic Encryption

Because we only have 0 or 1, we cannot use the comparison used by many sort algorithms that need -1, 0 and 1.

# OPE Example

A simple example of how OPE works in practice:

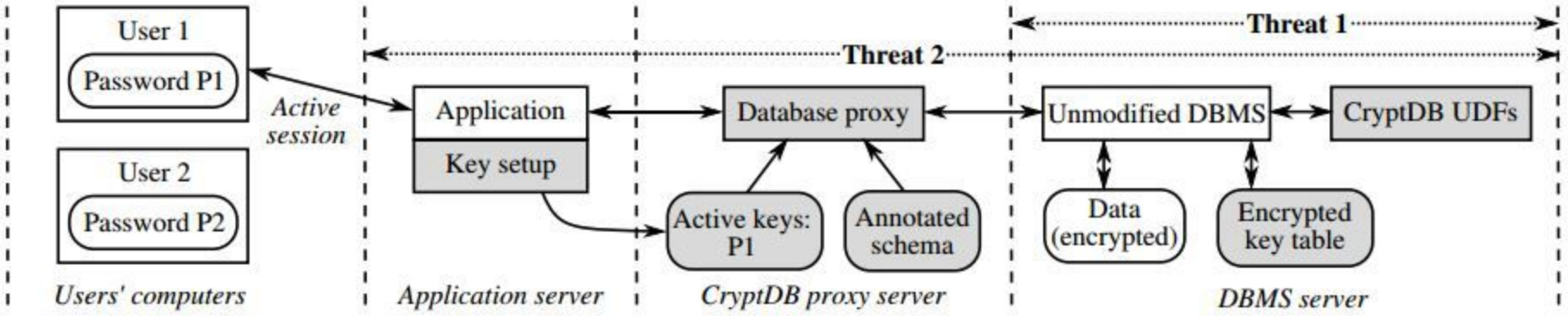
| Cleartext | Encryption |
|---|---|
| Europa | 5e182591 |
| Ganymede | 5ee60e3c |
| Callisto | 4e84cab1 |
| Himalia | c369fd1 |

| Encryption sorted | Corresponding Cleartext |
|---|---|
| 4e84cab1 | Callisto |
| 5e182591 | Europa |
| 5ee60e3c | Ganymede |
| c369fd1 | Himalia |

# CryptDB

One of the first databases to use encryption. Set up in three parts:

1. DBMS server - trusted - plaintext data
2. Proxy server - trusted - encrypts the data and holds keys
3. Application server - untrusted - should survive an attack



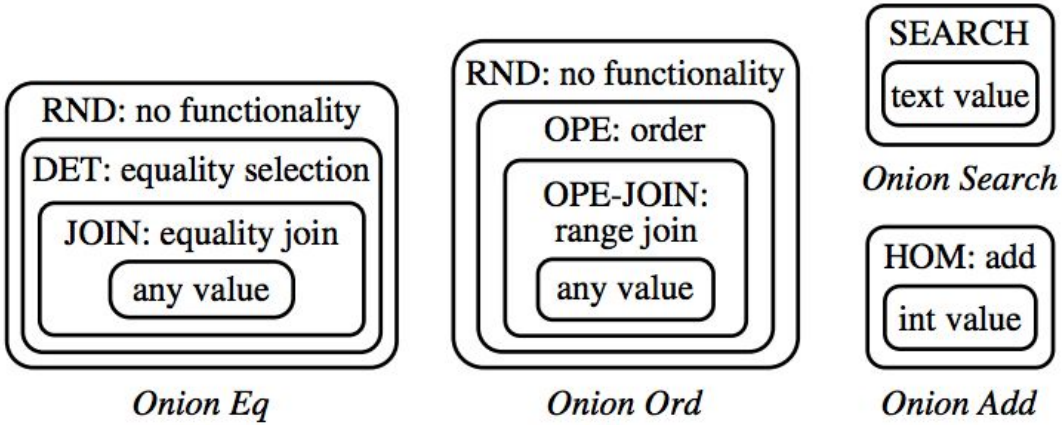
# Onion encryption

Every row/column intersection is encrypted with different PPE-schemes:

- RND: maximum security
- DET: deterministic - only the same for two equivalent plaintexts
- OPE: order-preserving
- HOM: homomorphic encryption for simple computations
- JOIN and OPE-JOIN: special encryption to support join-functions
- SEARCH: per-word padded encryption for word searches

Only the required onion is returned from the proxy to the application server.

Outer layers have more protection but less usability.



# Speakup - DET implementation

How would you implement the DET onion for comparisons?

A. Homomorphic encryption using a static key
B. El Gamal encryption using a static key
C. AES encryption using a static key
D. RSA encryption using a static key
E. Replace with a random value

# Attacking CryptDB

Inference Attacks on Property-Preserving Encrypted Databases

- Deterministic Encryption (DTE)
  If two encrypted message-texts are the same, their plain-text messages are the same
  - Frequency analysis
  - L2-optimization, which is an improved version of the frequency analysis
- Order Preserving Encryption (OPE)
  There is a *Test*-function that returns the order of two encrypted elements
  - Sorting attack, works well for dense columns
  - Cumulative attack, extended version for non-dense columns

# Results of Attacks

Goal: de-anonymize encrypted databases (EDBs) of medical data from the National Inpatient Sample (NIS) database.

- $l_p$-optimization / frequency analysis: recovered the mortality risk and patient death attributes for 100% of the patients
- sorting attack: recovered the admission month and mortality risk of 100% of patients for at least 90% of the 200 largest hospitals.
- cumulative attack
  - Largest 200 hospitals: recovered disease severity, mortality risk, age, length of stay, admission month, and admission type of at least 80% of the patients
  - For 200 small hospitals, the attack recovered admission month, disease severity, and mortality risk for 100% of the patients for at least 99.5% of the hospitals.

# Takeaway on PPE

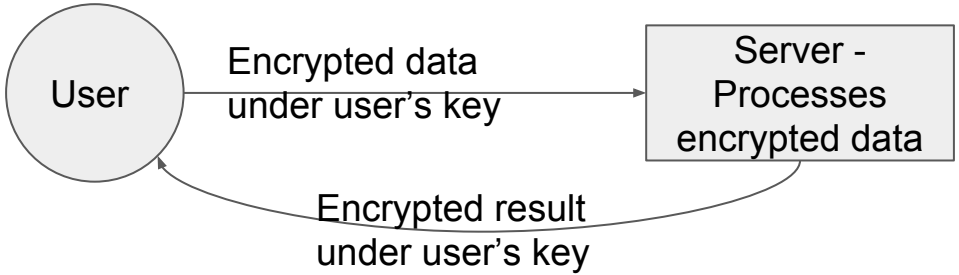
- Raises the bar on an attack a little
- Doesn’t anonymize if a similar reference dataset is available
  - Medical data under PPE, and publicly available names and addresses
  - Employee data under PPE, and publicly available list of employees
  - This even works for sparsely linked PPE and publicly available datasets
- CryptDB did not find the holy grail

# Overview

- Motivation
- Naive Anonymization
- Differential Privacy
- **Confidentiality using Encryption**
  - Property-Preserving Encryption - CryptDB
  - **Building on Property-revealing EncrypTion - BoPET**

# Building on Property-revealing Encryption

The main idea behind BoPET is to encrypt users' data before uploading it to the server using special property revealing encryption (PRE). The server can then execute its part of the application's functionality on encrypted data.

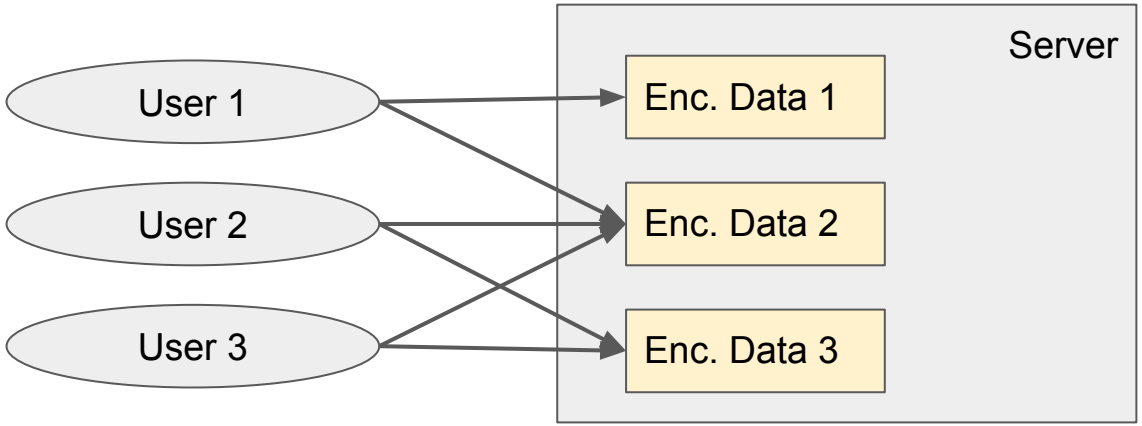



Property Revealing Encryption includes Property Preserving Encryption, but can give more information than PPE, which is restricted to 1 bit revelations.

# Multi-key Searchable Encryption (MKSE)

A main building block of BoPETs is Multi-key Searchable Encryption:

- The users send encrypted data to the server
- The server
  - Does not have access to the keys
  - Can do searches / comparisons on the encrypted data



# Attacking Mylar

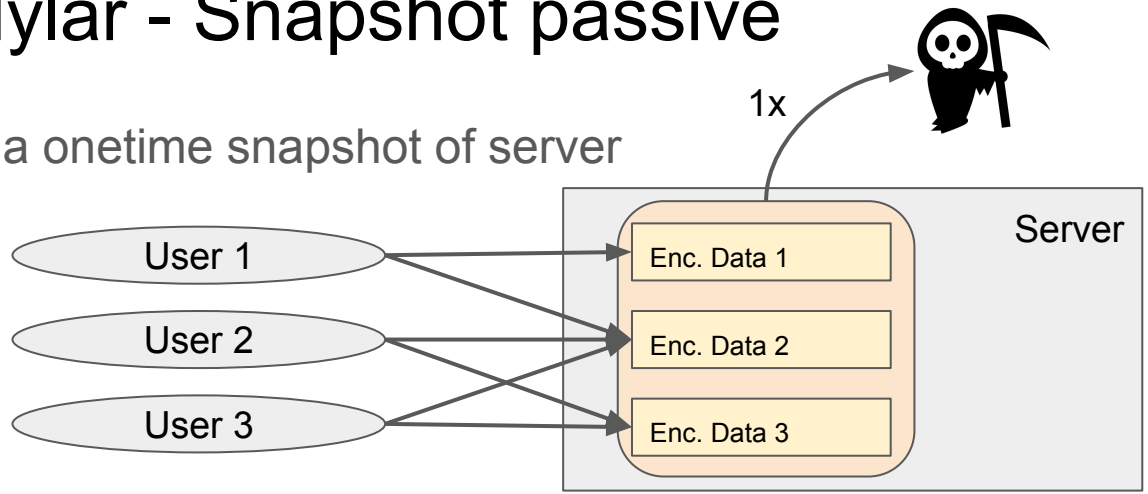
https://eprint.iacr.org/2016/920.pdf

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

- Setup
  - App communicates encrypted data to a server using BoPET
- Example applications
  - kChat - done by the Mylar developers themselves
  - Ported by the paper's authors:
    - MDaisy - medical appointment app
    - MeteorShop - eCommerce site
    - OpenDNA - enables users to search for risks based on DNA-sequences
- Threat model
  - Snapshot passive - Attacker captures a onetime snapshot of server
  - Persistent passive - Attacker records server operations over a period of time
  - Active - Server can arbitrarily misbehave, collude with users

# Attacking Mylar - Snapshot passive

Attacker captures a onetime snapshot of server

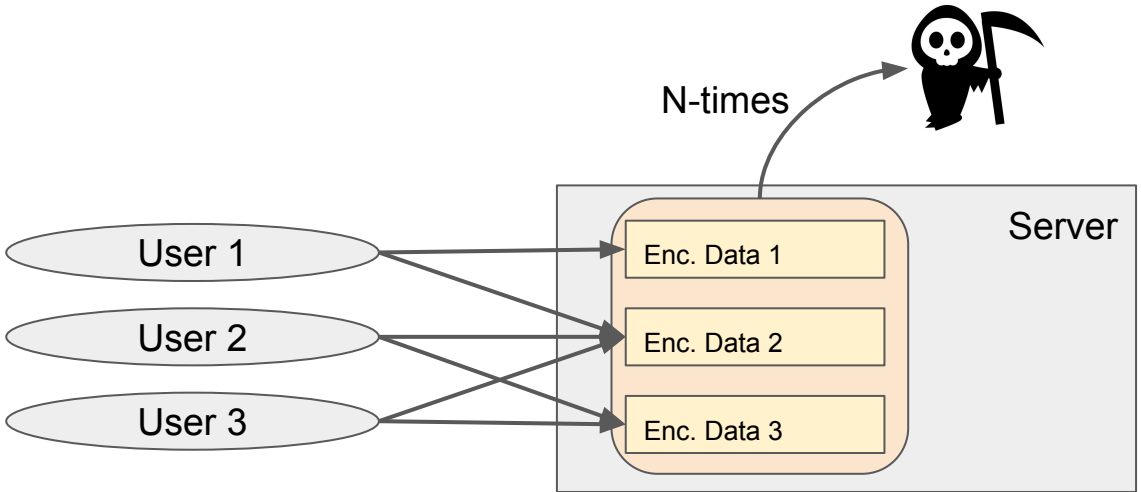



- kChat: names of principals (representation object in Mylar) leak information about chat room topics
- MDaisy: metadata about relationships between documents leaks patients' medical information; size of users' profiles leaks their roles
- OpenDNA: size of encrypted DNA leaks which risk groups the user is searching for

# Attacking Mylar - Persistent passive

Attacker records server operations over a period of time

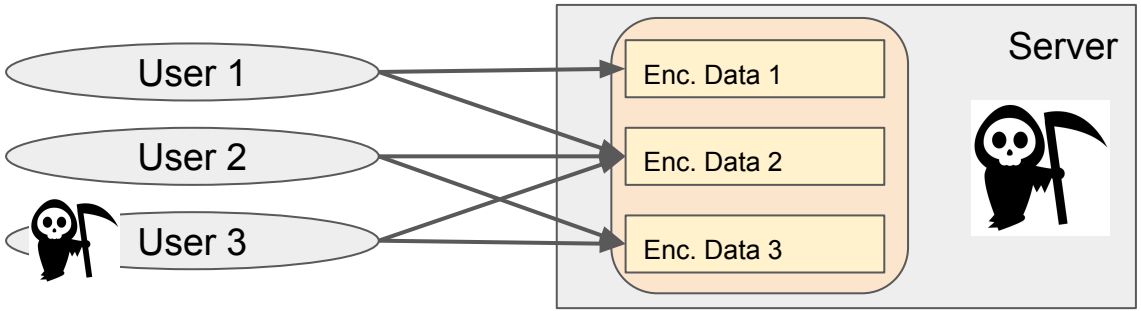
- MDaisy: server can cluster patients by their medical procedures; if one patient reveals information, entire cluster is compromised
- MeteorShop: users' recently viewed items leak their encrypted shopping carts



# Attacking Mylar - Active

Server can arbitrarily misbehave, collude with users

- Any Mylar app: malicious server can perform brute-force dictionary attacks on any past, present, or future search query over any server-hosted content
- OpenDNA: malicious server can search users' DNA for arbitrary SNPs



# Takeaway on Mylar

- Encryption alone is not enough
- As with PPE, MKSE raises the bar a little
- Given a reference dataset, MKSE is breakable, too
- With high-incidence PII information, neither PPE nor MKSE protect it enough

# Speakup - Best Privacy Protection

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

As an engineer, which of the following best practices brings the biggest gain in privacy protection for your users?

A. Encrypt everything on the server
B. Only collect relevant data
C. Anonymize data as soon as it exits the server
D. Always aggregate data when possible

# Conclusion

- Creating good anonymity and privacy is hard!
- Limit what is revealed
  - K-anonymity reduces the set of comparable users
  - Differential Privacy reduces the amount of shared data
- Limit what is collected in the first place
- Releasing anonymized sets needs good preparation