Nigel Paul Smart

Computing on Encrypted Data

How to do the impossible

KU Leuven

Dining Bankers (a.k.a. Millionaire’s Problem)

A set of bankers go to lunch.

They are celebrating their bonuses just being paid.

Each has been given a bonus of x_i dollars.

The one with the biggest bonus should pay.

But they do not want to reveal their bonus values.

Dining Bankers (a.k.a. Millionaire's Problem)

What they want to compute is the function

$$F(x_1,\ldots,x_n) = \{\ i : x_i \geq x_j \ \text{ for all } j\ \}$$

without revealing the x_i values.

This problem (Millionaires Problem) introduced by Andrew Yao in early 1980s.

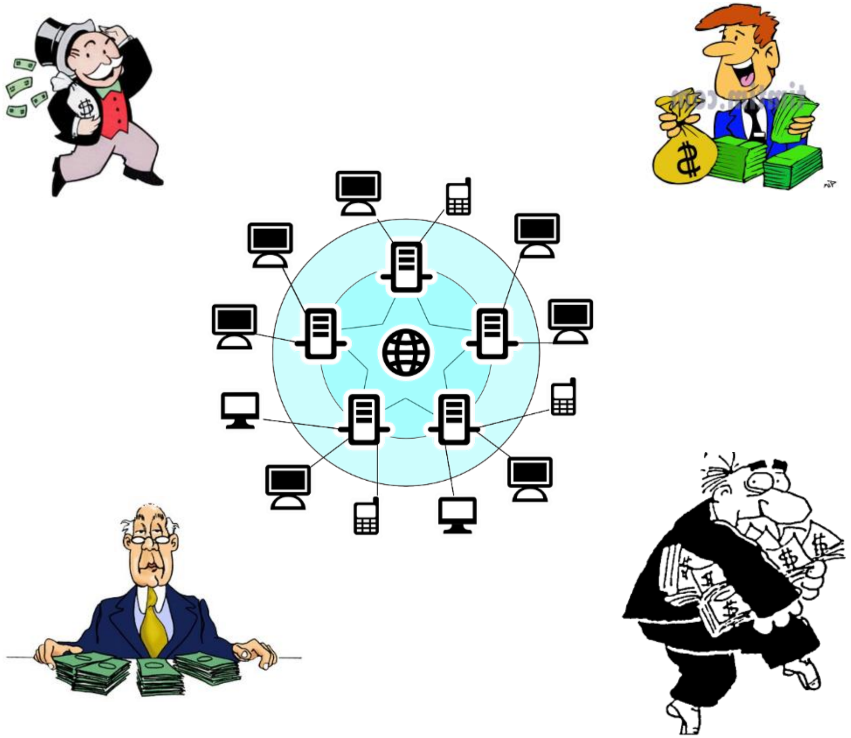
Andrew won the Turing Award for this and other work.

Dining Bankers (a.k.a. Millionaire’s Problem)

If the bankers had a person they trusted they could get this person to compute the answer to their problem for them.

They give the trusted person their bonus values and the trusted person computes who should pay for lunch.

Dining Bankers (a.k.a. Millionaire’s Problem)



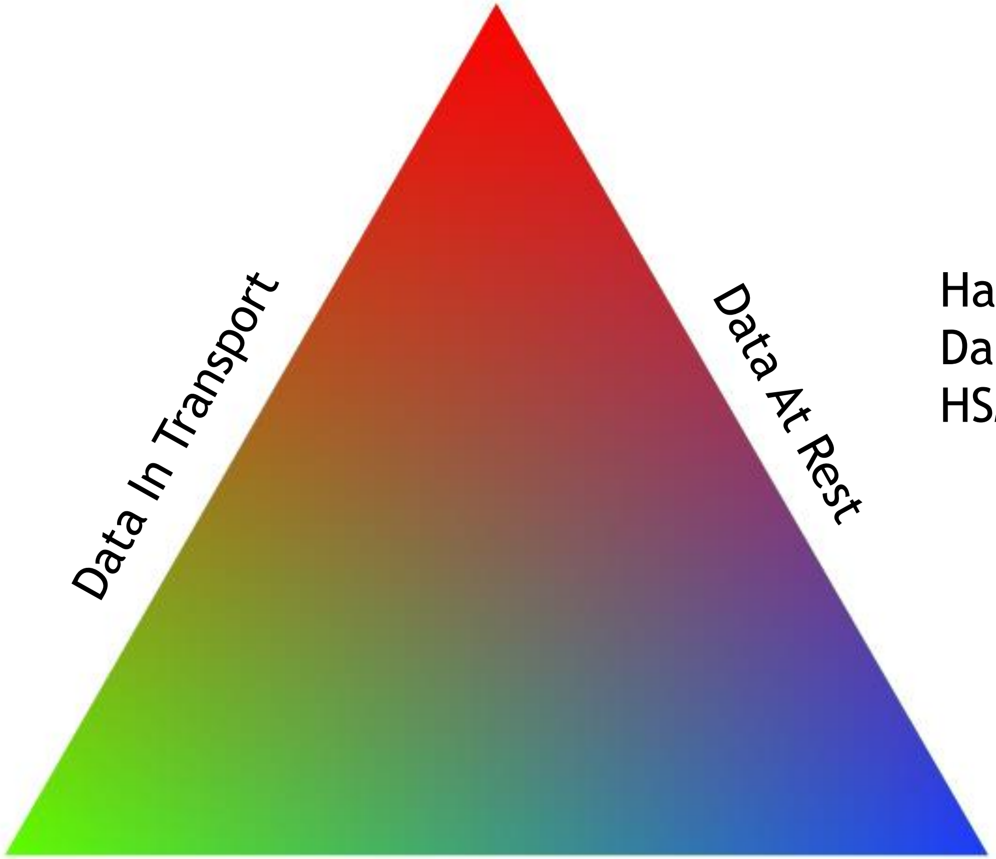
In real life such trusted people do not exist, or are hard to come by. So we want a protocol to compute the function securely. This is what MPC does.

It emulates a trusted party, enabling mutually distrusting parties to compute an arbitrary function on their inputs.

All that is revealed is what can be computed from the final output.

Securing Data

TLS/SSL
IPSec



Data In Transport

Data At Rest

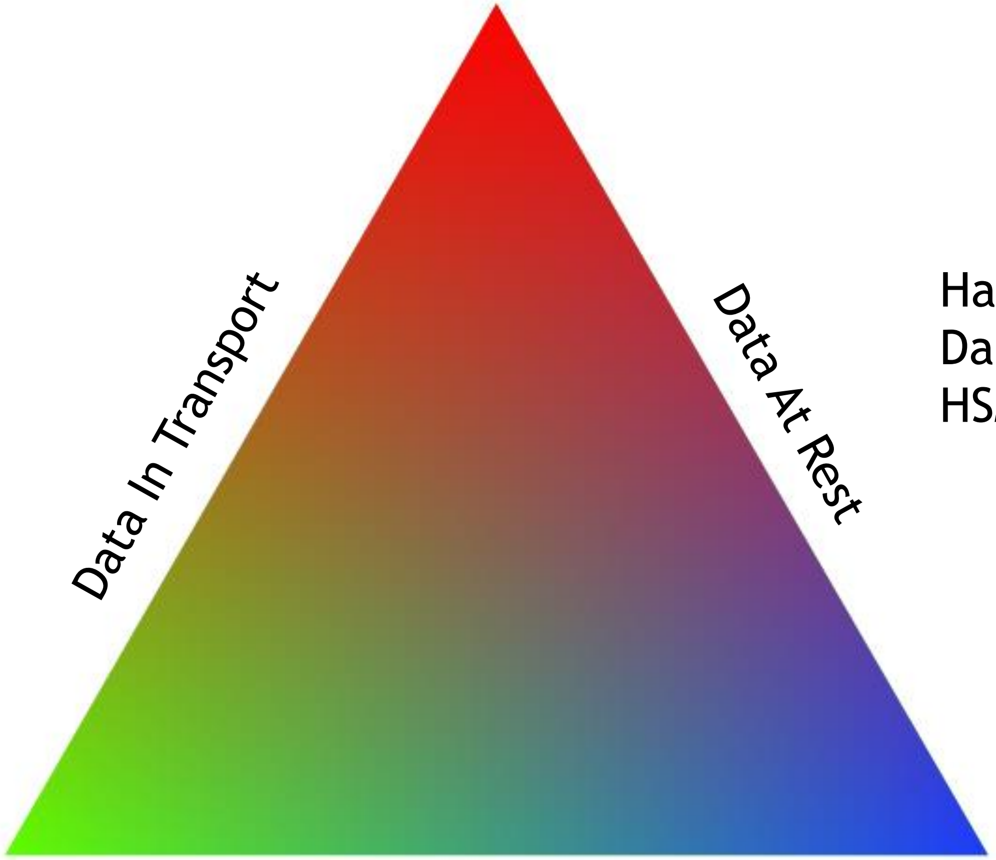
Hard disk encryption
Database encryption
HSM key storage

Data During Computation

???????????????????????????????????

Securing Data

TLS/SSL
IPSec



Data In Transport

Data At Rest

Hard disk encryption
Database encryption
HSM key storage

Data During Computation

???????????????????????????????????

Voting

GDPR

Genomics

Public
Policy

Citizen
Privacy

Two Technologies: MPC and FHE

- In MPC all parties engage in a protocol to compute the function securely
  - Relatively fast in computation
  - Expensive in communication
  - Enables a number of applications (see later)

- FHE the parties encrypt their data, a server computes the function in the encrypted domain, a designated party gets the output
  - Very very slow in computation
  - Relatively cheap in communication
  - Only possible (currently) for simple functions.

Basic Set Up

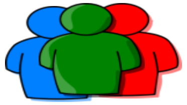
- We assume some data is being processed.
  - Think of genomic data, but it could be anything

- There are three basic groups of actors
  - Input Parties
  - Processing Parties
  - Output Parties

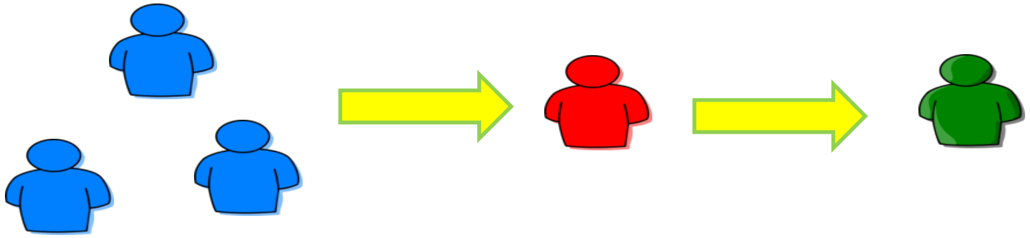
- In a traditional application there is one of each, and they are all the same person.
- We could however have very different scenarios…

Scenarios

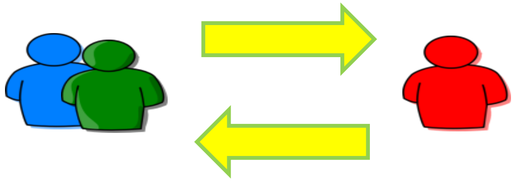
- Traditional



- Many Different Input Parties

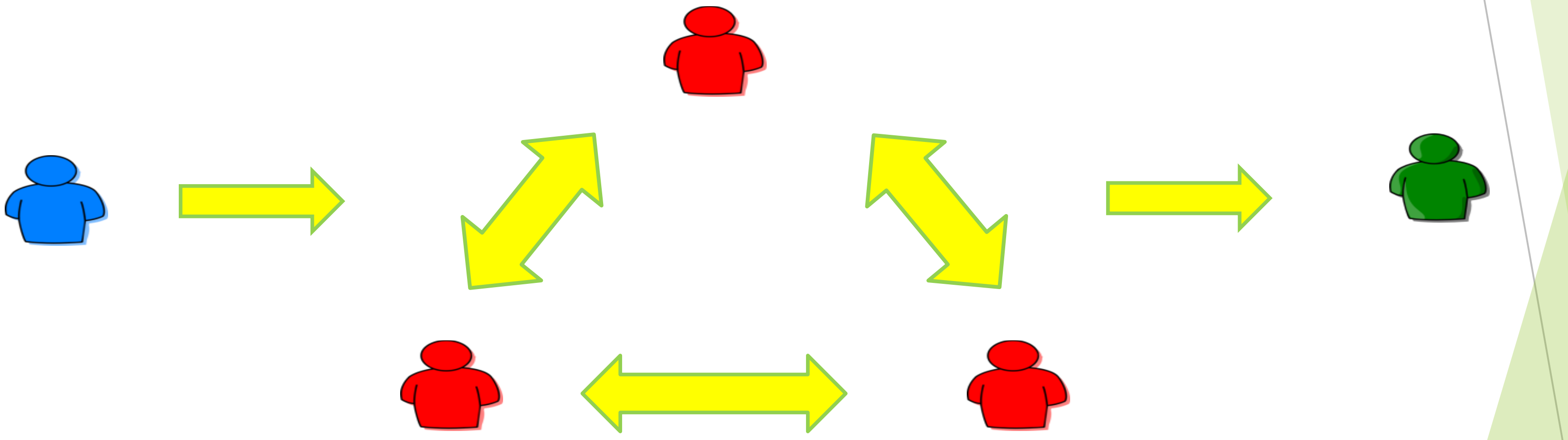


- Input Parties=Output Parties
  - Think of this as the usual paradigm for Cloud Computing



Scenarios

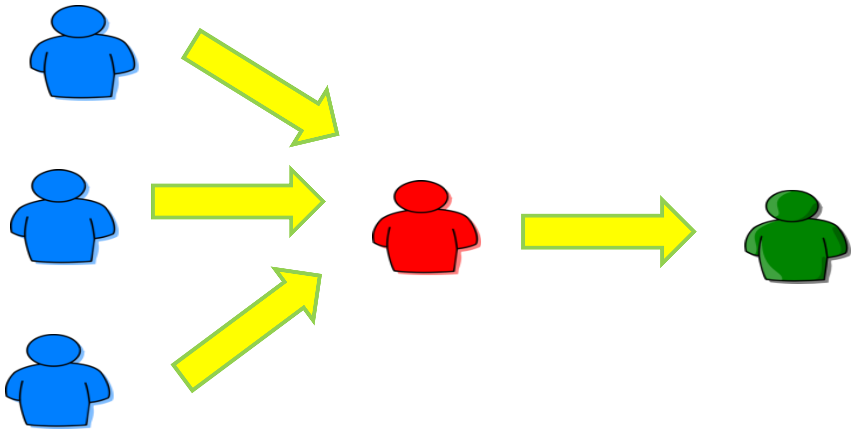
- Many computing parties



- And all other combinations of the above

Fully Homomorphic Encryption

- One computing party
- One or many input parties
- One output party (could be more)
- Output party != Computing party



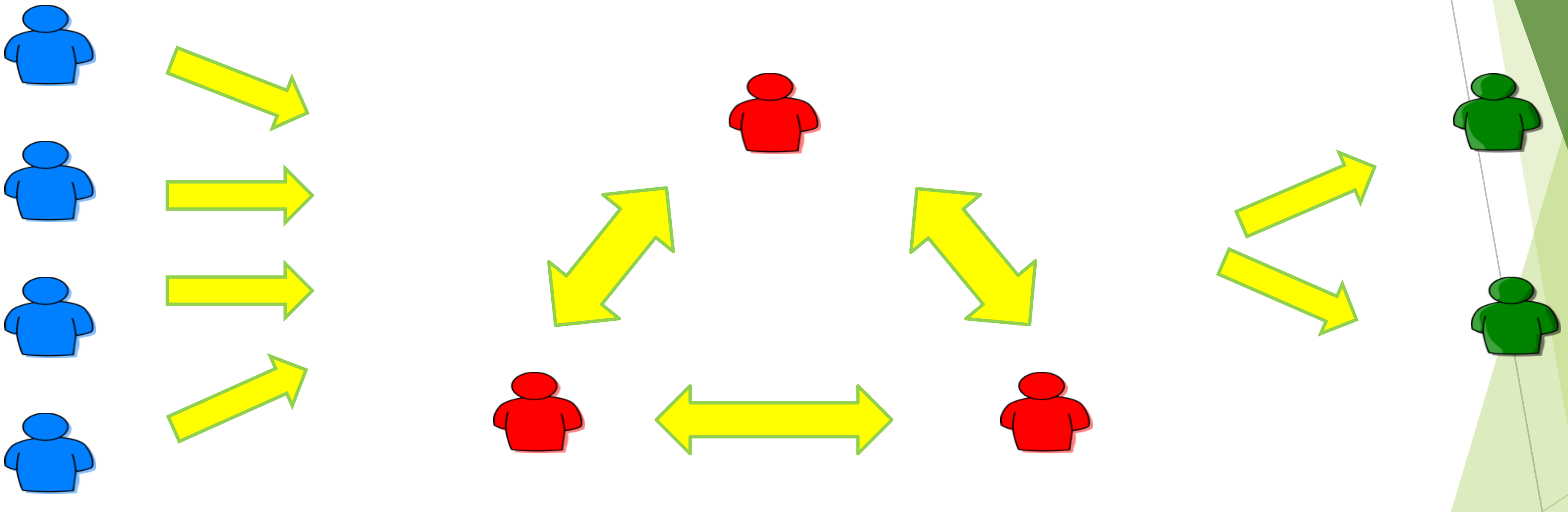
Fully Homomorphic Encryption

- Input parties encrypt their data
- Computing party evaluates the function on the encrypted data (without seeing the data)
- Output party performs the decryption

- First scheme 2008
- In theory can compute any function, with only a small overhead in cost
- In practice much more difficult

- Today this is practical for functions of low multiplicative depth
  - Think basic statistics, machine learning algorithms

Multi-Party Computation



FHE vs Multi-Party Computation

- The problem with FHE (i.e. the thing which made it hard to produce) was that we had only one computing party

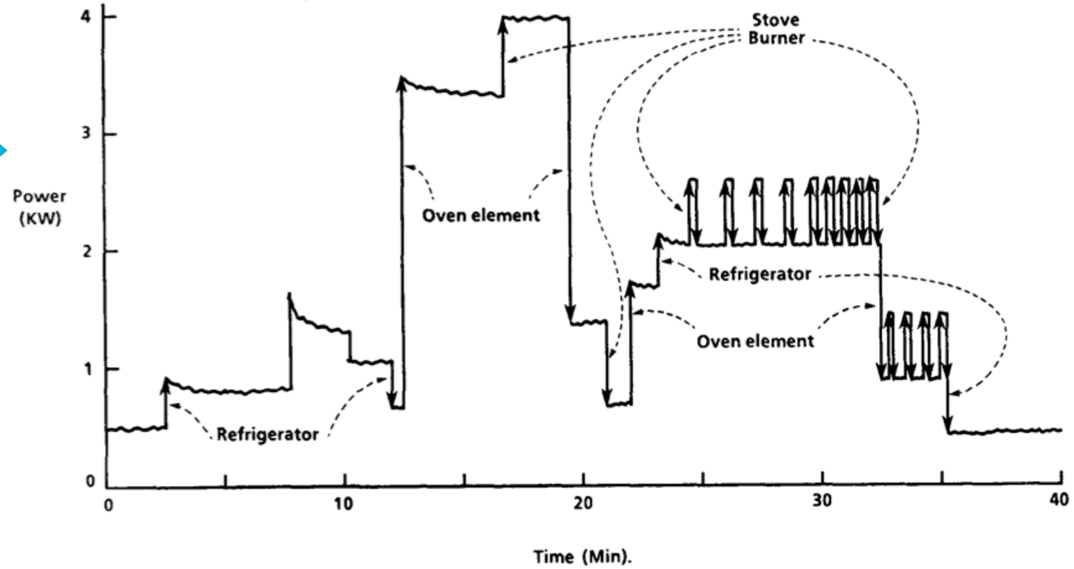
- With MPC we can have many input, computing and output parties, and indeed they could all be subsets of each other (or even exactly the same parties)

- Key point is that we have $n \geq 2$ computing parties

- In MPC we use a lot of communication though

FHE Example: Privacy in the Smart-Grid

Energy consumption

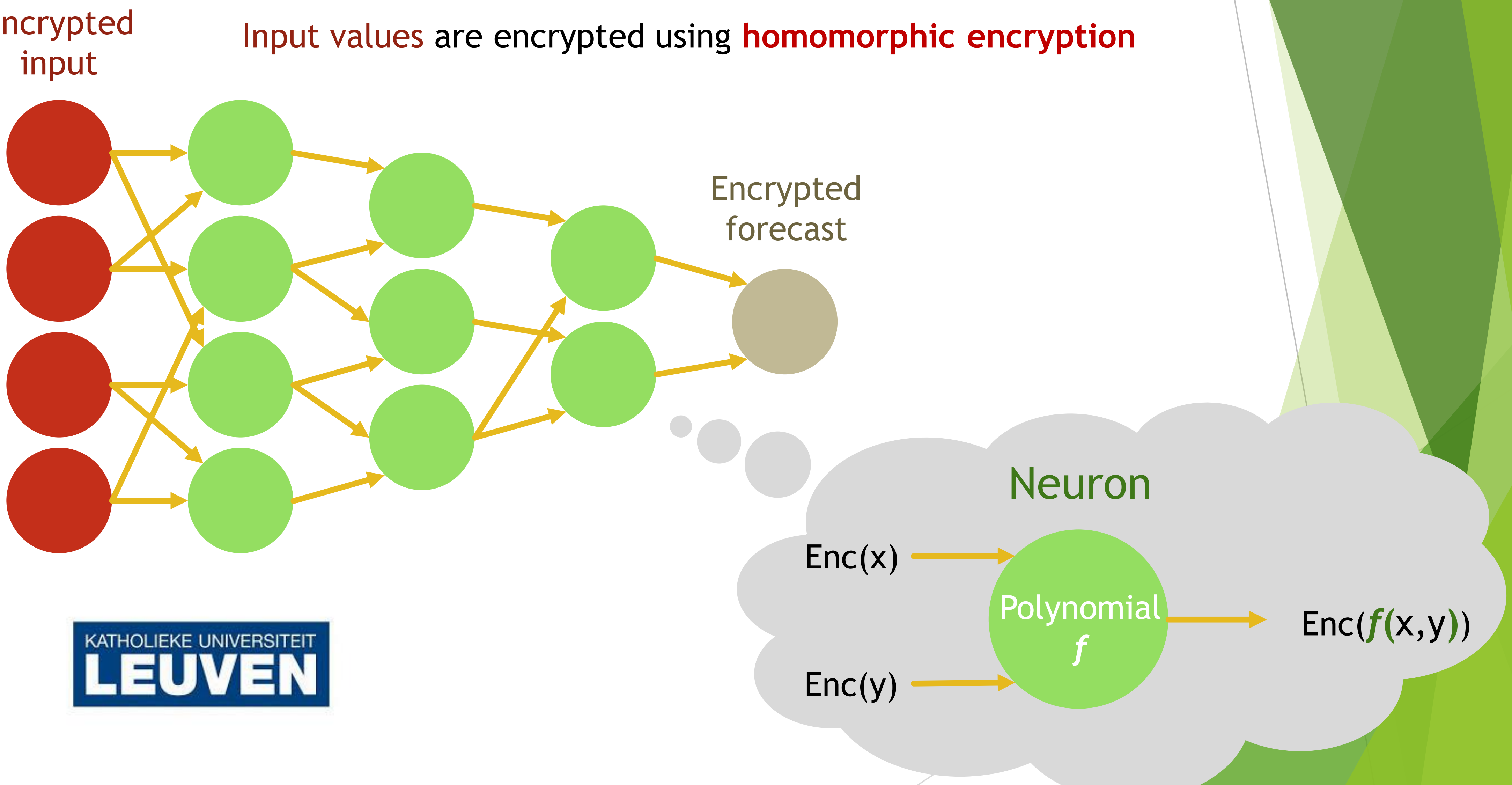


Power step changes due to individual appliance events

KATHOLIEKE UNIVERSITEIT LEUVEN

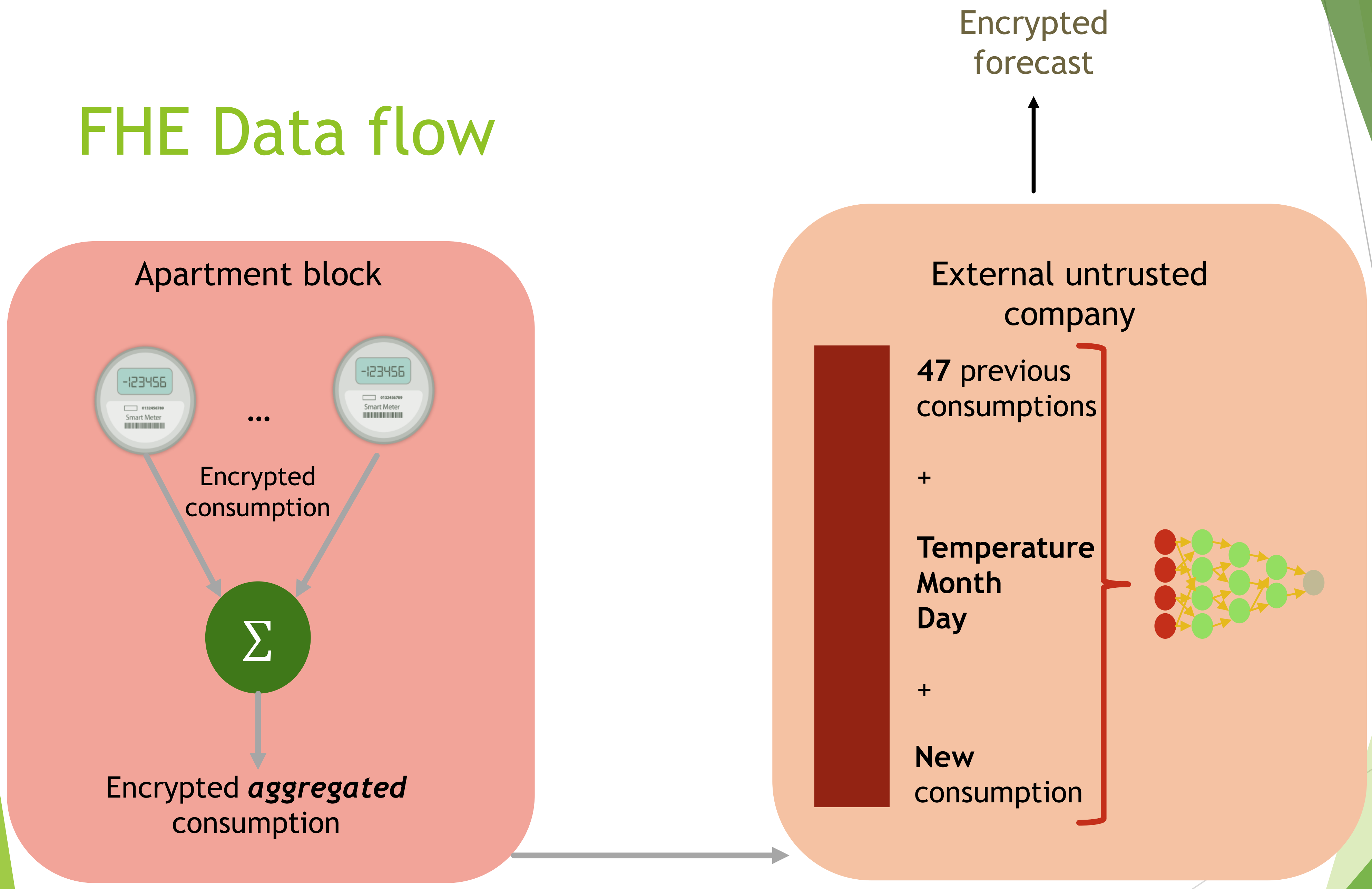
Privacy-friendly energy forecasting

Input values are encrypted using **homomorphic encryption**



KATHOLIEKE UNIVERSITEIT LEUVEN

FHE Data flow



Apartment block

…

Encrypted consumption

Encrypted ***aggregated*** consumption

External untrusted company

47 previous consumptions

+

Temperature
Month
Day

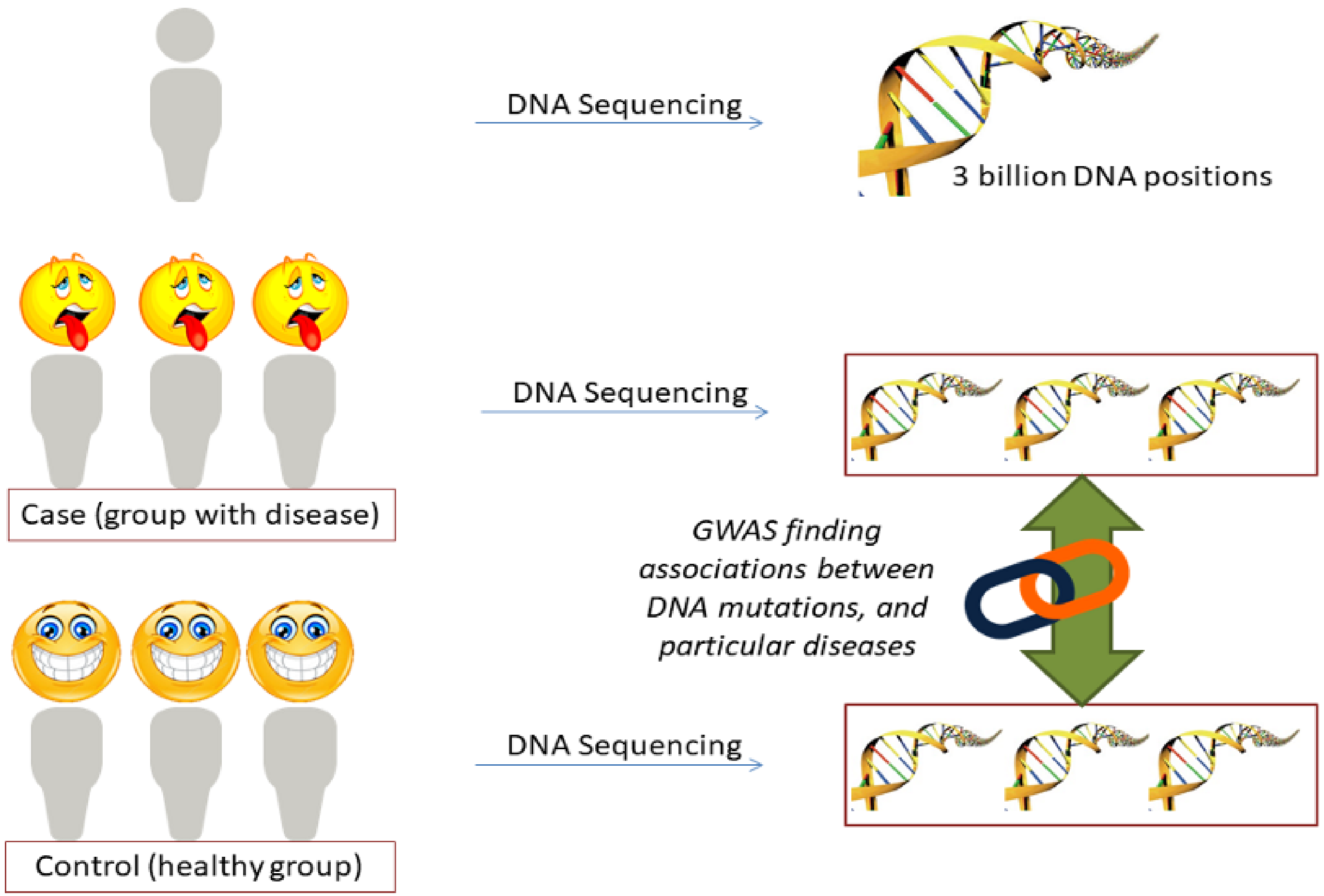
+

New consumption

Encrypted forecast

Prediction error for 10 houses: 23%

Genome Wide Association Study via FHE and MPC



DNA Sequencing

3 billion DNA positions

DNA Sequencing

Case (group with disease)

GWAS finding associations between DNA mutations, and particular diseases

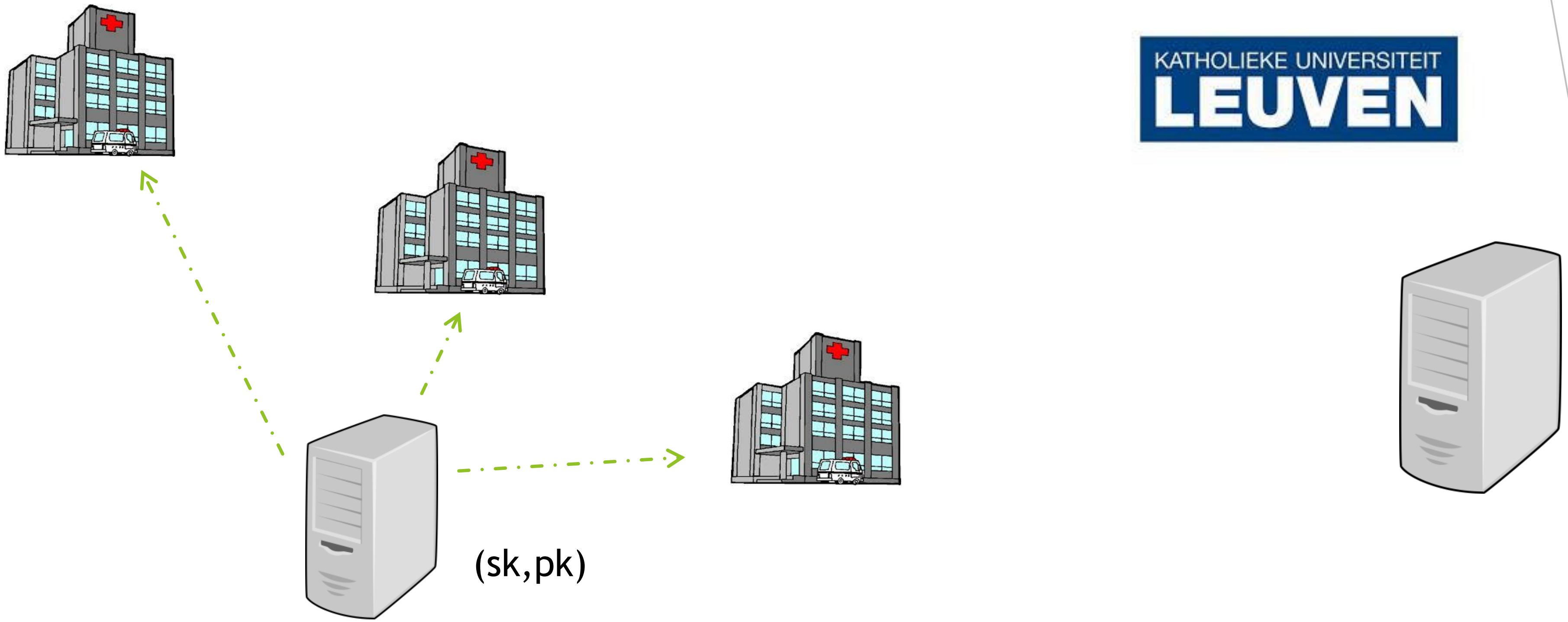
DNA Sequencing

Control (healthy group)

KATHOLIEKE UNIVERSITEIT LEUVEN

Homomorphic Encryption Variant

KATHOLIEKE UNIVERSITEIT LEUVEN

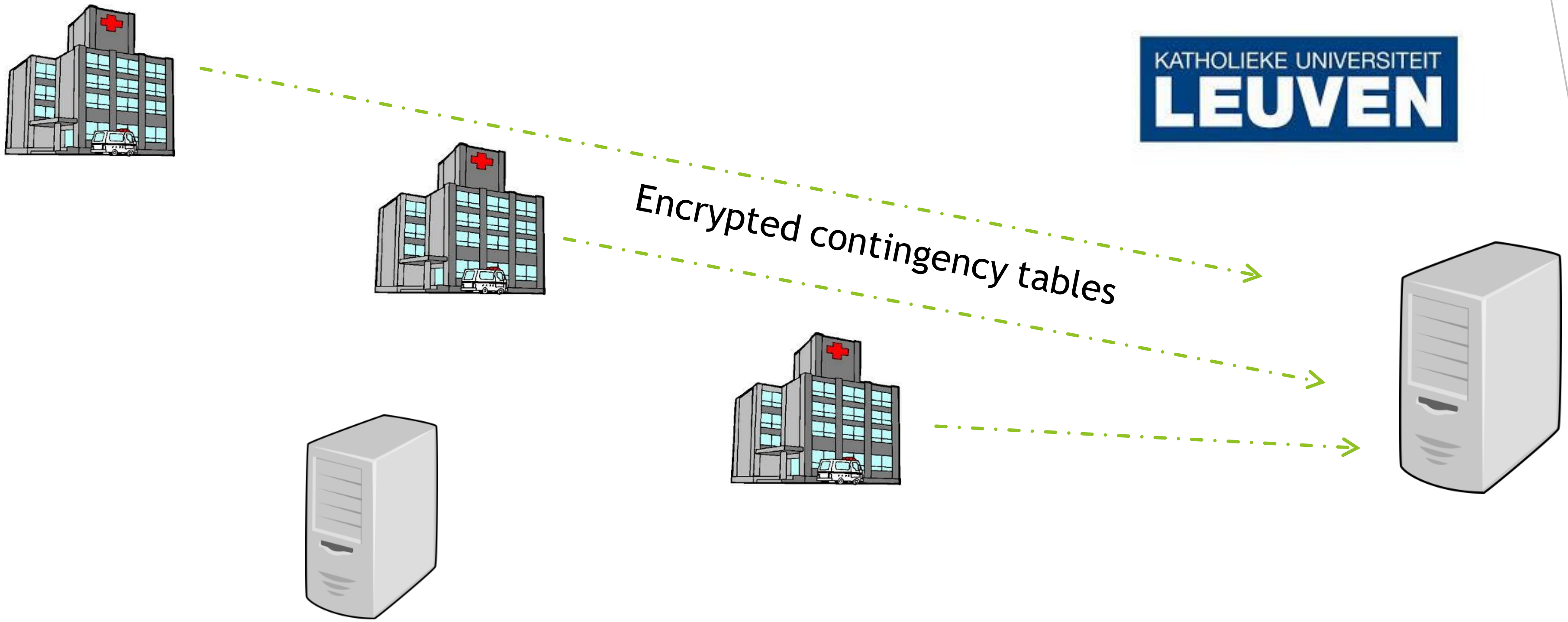


Two servers : One compute (right), one decryptor (left)

Step 1: Decryptor generates FHE keys and sends public keys to the hospitals

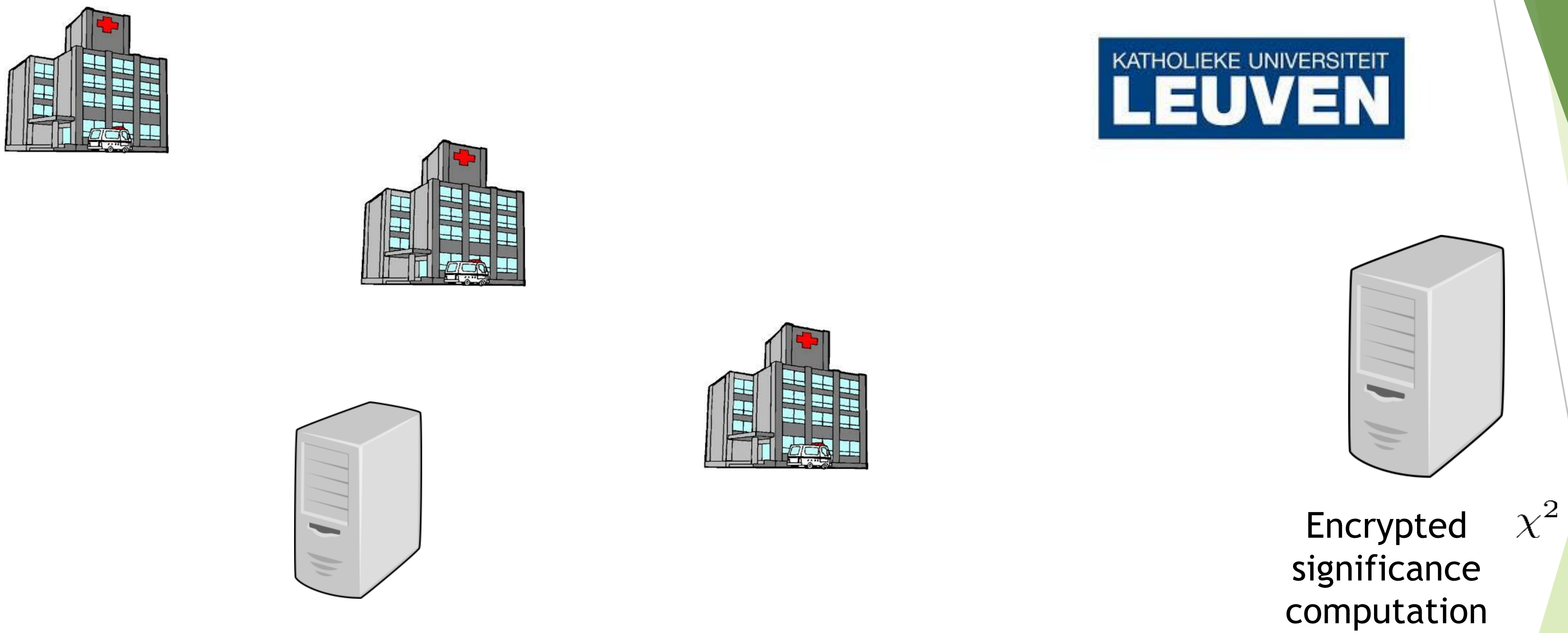
Homomorphic Encryption Variant

KATHOLIEKE UNIVERSITEIT LEUVEN



Step 2: The hospitals encrypt their contingency tables to the compute server

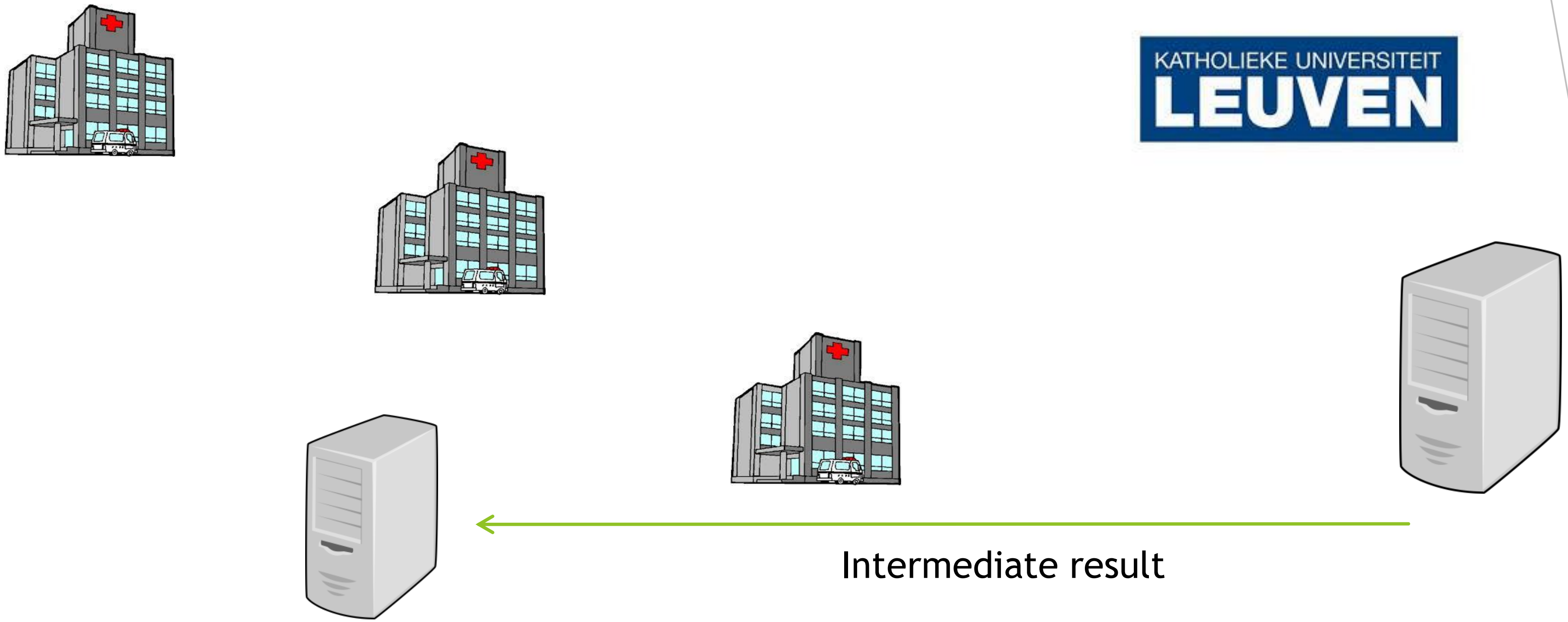
Homomorphic Encryption Variant



Encrypted significance computation χ^2

Step 3: The compute server (partially) performs the chi-squared computation

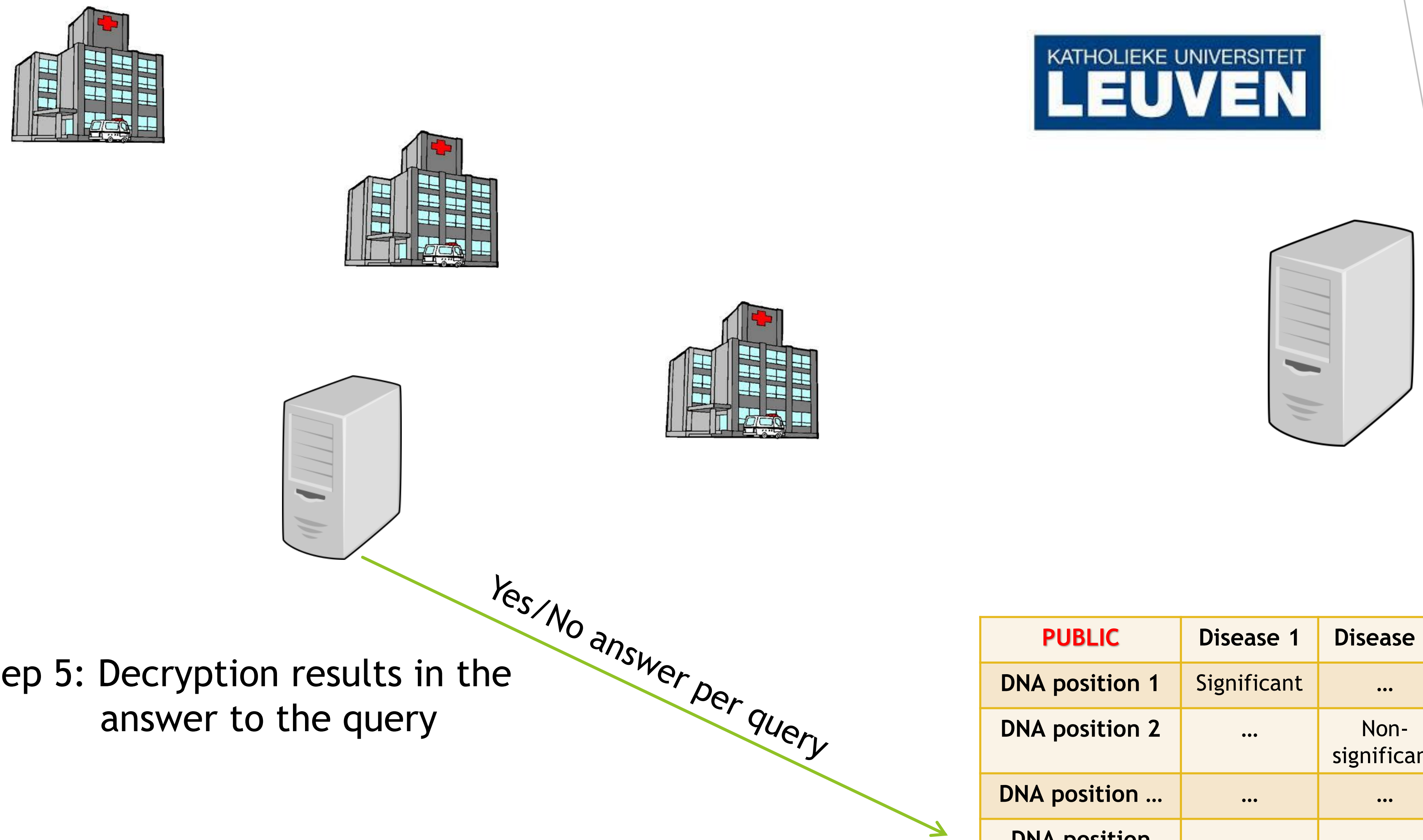
Homomorphic Encryption Variant



Step 4: Intermediate results are passed back to the the decryption server in a blinded form

So upon decryption only the result is obtained

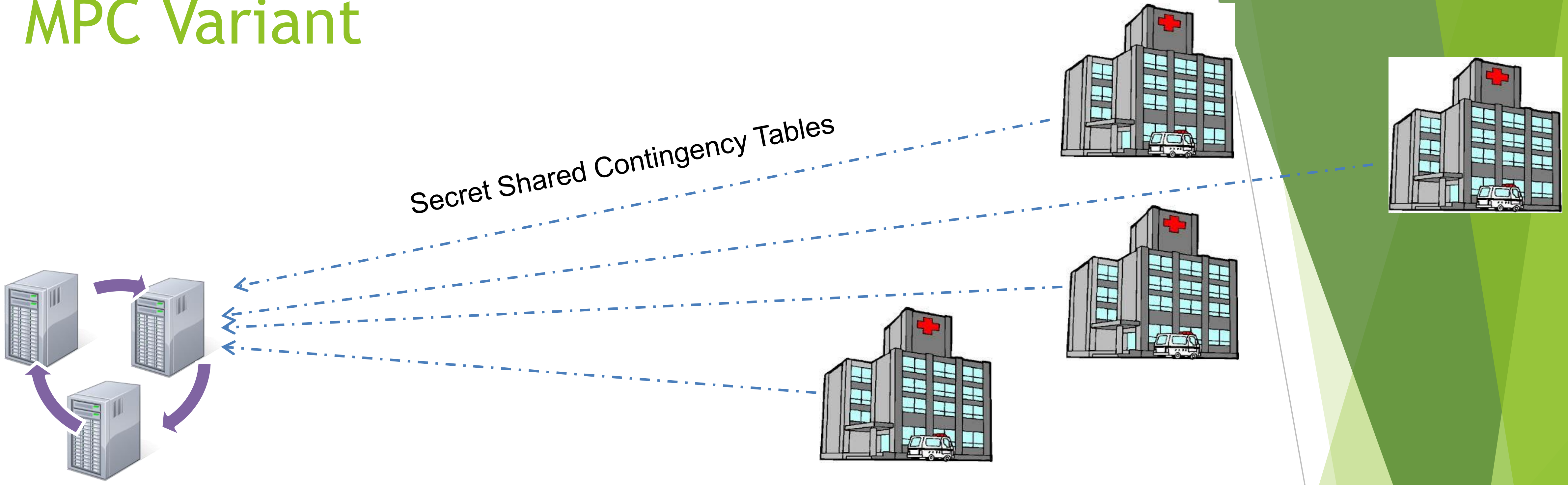
Homomorphic Encryption Variant



Step 5: Decryption results in the answer to the query

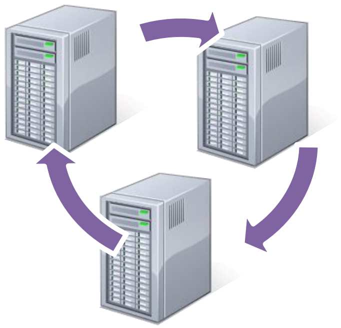
PUBLIC	Disease 1	Disease 2	Disease …	Disease 11.000
DNA position 1	Significant	…	…	…
DNA position 2	…	Non-significant	…	…
DNA position …	…	…	…	…
DNA position 3.000.000.000	…	…	…	…

MPC Variant

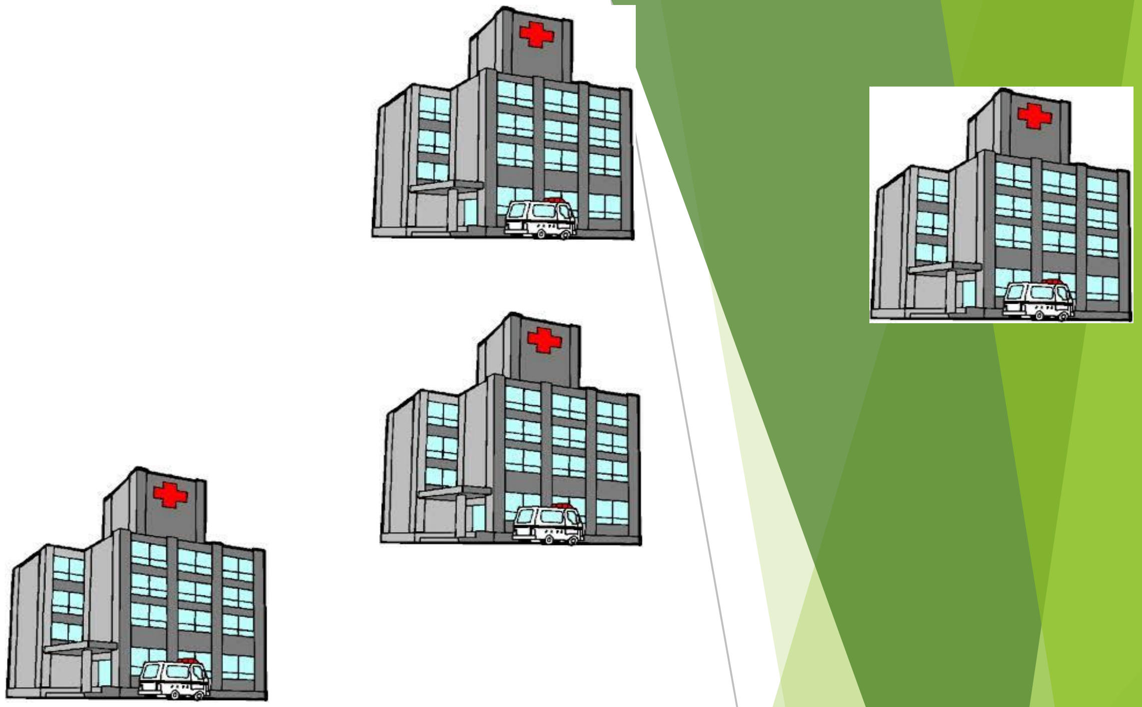


Step 1: The hospitals secret share their contingency tables to the MPC engine

MPC Variant

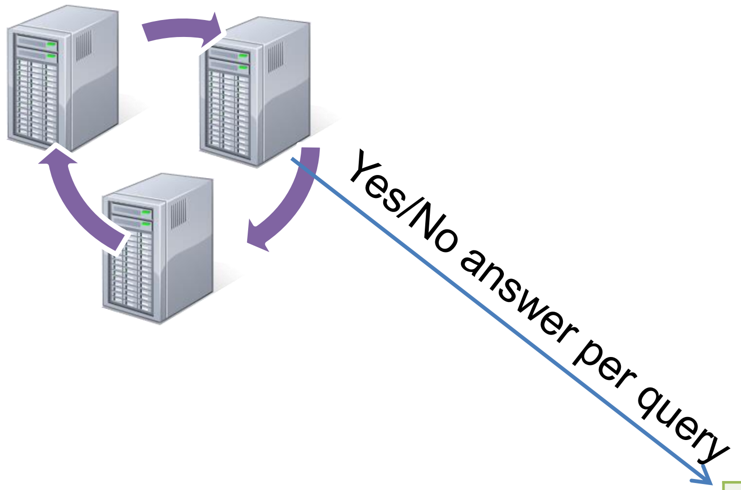


Privacy-preserving significance computation



Step 2: The MPC engine performs on the computation on the secret shared data

MPC Variant



Step 3: Answers are reconstructed and the relevant secret shares are opened.

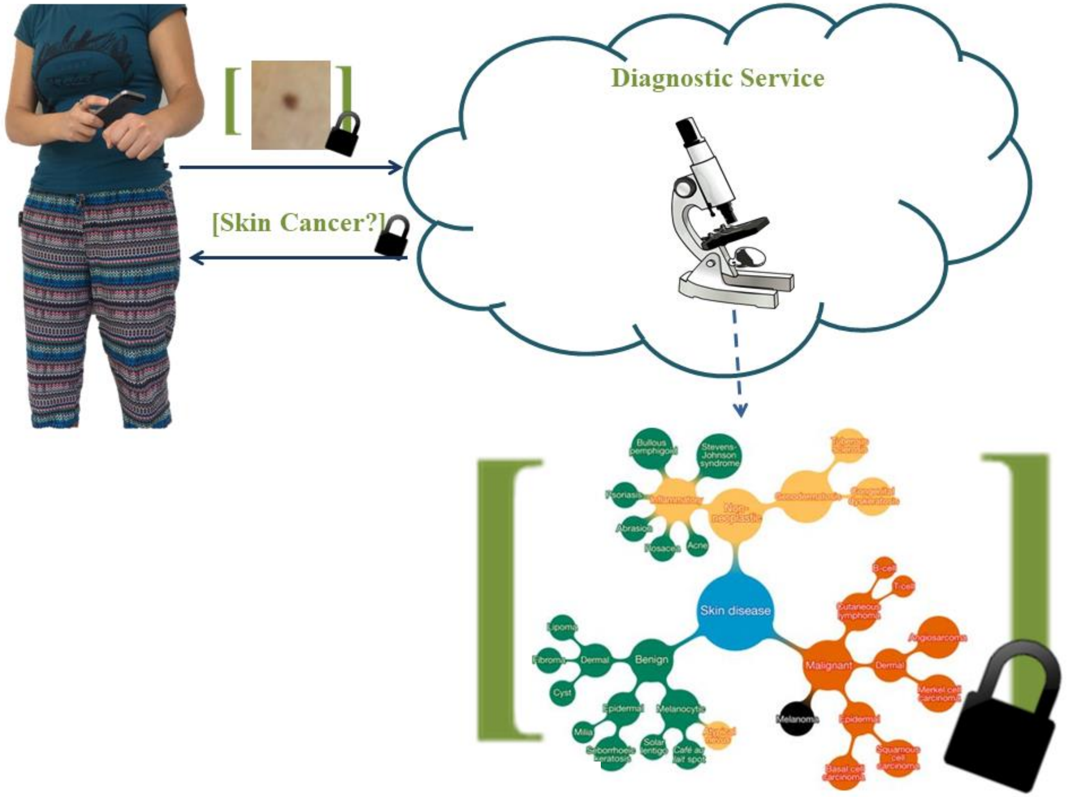
PUBLIC	Disease 1	Disease 2	Disease …	Disease 11.000
DNA position 1	Significant	…	…	…
DNA position 2	…	Non-significant	…	…
DNA position …	…	…	…	…
DNA position 3.000.000.000	…	…	…	…

EPIC MPC Based Image Recognition

Basic problem is how can one keep the image private AND the model being applied to the image

An image clearly has privacy issues.

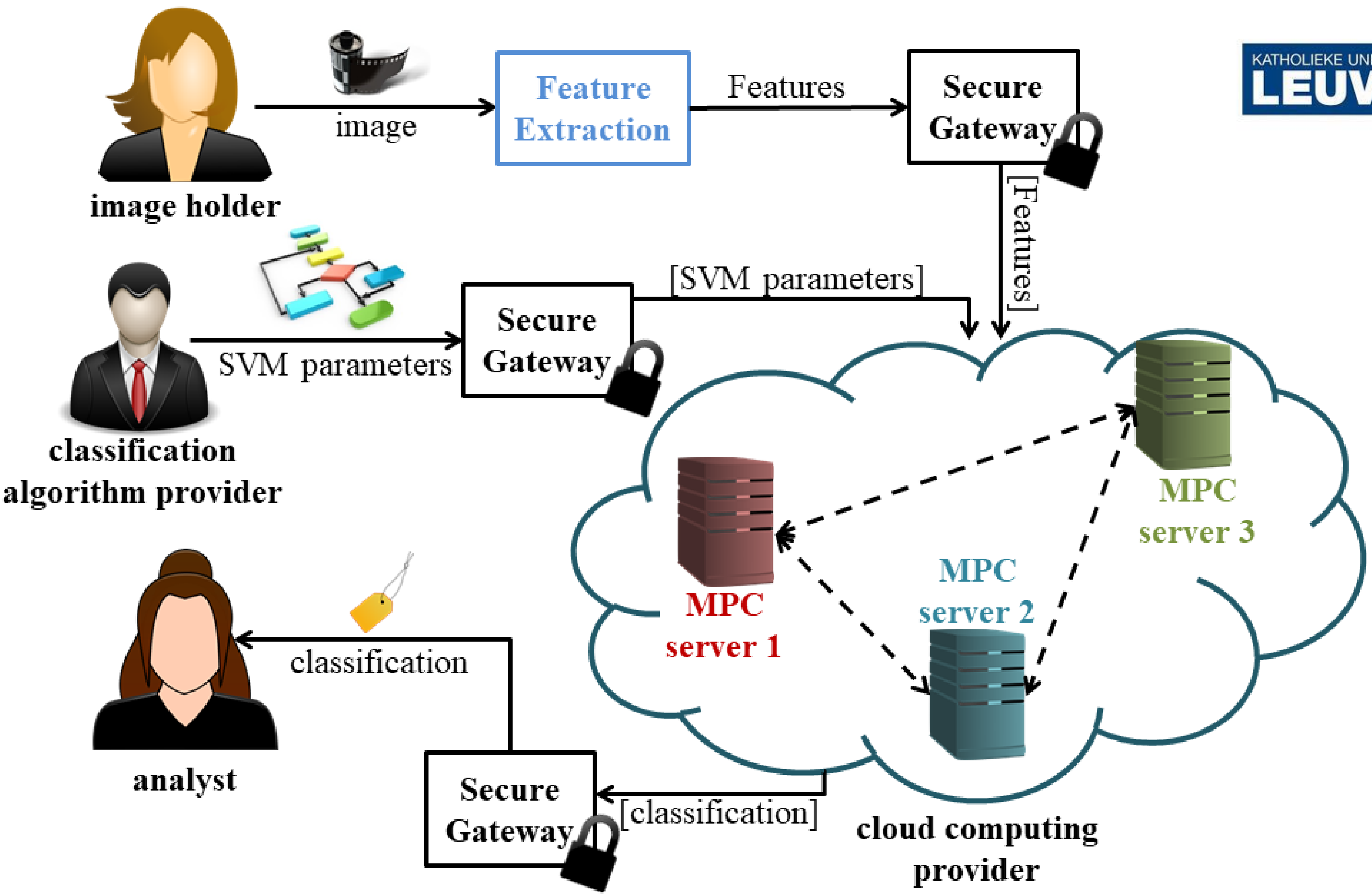
But so does a model, as it could contain sensitive commercial imformation.



KATHOLIEKE UNIVERSITEIT
LEUVEN

EPIC: Efficient Private Image Classification

KATHOLIEKE UNIVERSITEIT LEUVEN



Efficiency compared to state-of-the-art

Previous state of the art was a system called Gazelle (USENIX 2018)

- EPIC vs. Gazelle on CIFAR-10:
  - 34 times faster runtime;
  - 50 times improvement of communication cost;
  - 7% higher classification accuracy.
- EPIC vs. Gazelle with the same accuracy:
  - 700 times faster runtime;
  - 500 times improvement of communication cost.
- Appeared at CT-RSA 2019

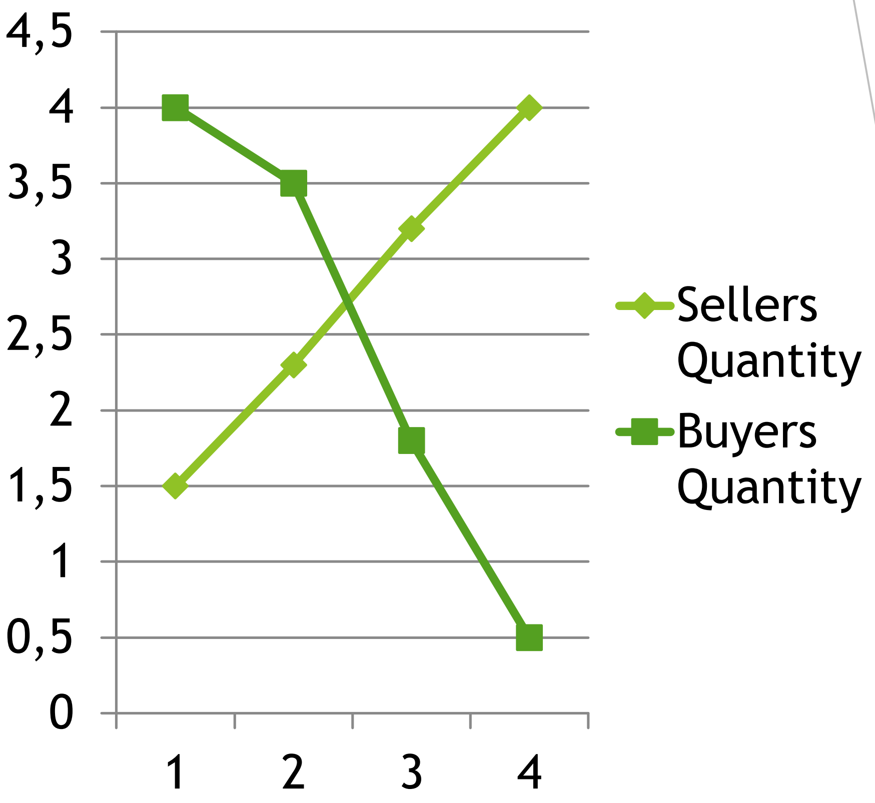
Auction Example

Similar example occurs in a sealed bid auction

- □ Buyers/sellers want to determine clearing price
- □ Single one off auction (not continuous as in stock markets)

Partisia (a Danish company) pioneered work in this area

- □ First MPC auction done in mid 2000's for Danish Sugar Beet



PARTISIA

Dark Market Example

Consider a “Dark” stock market

- Buyers/sellers bids kept in dark to avoid major swings in price
- Common for large trades to be done in this way
- The dark market operator acts as a god figure
- But they can cheat (actually happened in 2017)
- Can replace the dark operator by an MPC protocol

- We looked into the most efficient way of doing this, appeared ASIA-CCS 2019
  - Questions related to exactly how to deal with the real time nature of such markets
  - Examining different mechanisms used in real Dark markets to see which can be transferred to the MPC arena.

Dark Market Experiments

KATHOLIEKE UNIVERSITEIT LEUVEN

Using our SCALE-MAMBA system....

- Continuous Double Auction Method
  - Two Party Online Throughput : 60-250 orders per second
  - Three Party Online Throughput : 30-140 orders per second
- Volume Matching Auction Method
  - Two Party Online Throughput : 2000 orders per second
  - Three Party Online Throughput : 1000 orders per second

- Two Party here means using the SPDZ protocol
  - Uses a combination of SHE and MPC
- Three Party here means using Shamir 1-out-of-3 sharing
  - Optimized for online efficiency
- Both actively secure MPC protocols

Statistics

Suppose you want to analyse two databases

  - E.g. Combine customer data from different banks to produce a better credit scoring model
  - Privacy concerns mean you cannot share the data
  - But using MPC you could be able to produce a combined credit score

- Similar situation occurs in other databases
  - City of Boston gender equality survey
  - Estonian Tax+Education analysis CYBERNETICA
  - US Gov move for more student outcomes data for colleges “Know before you go”
  - Evidence based policy making initiative of Senator Wyden and others

Statistics + Differential Privacy

Question is whether a query reveals information

- ❑ Allowing salary average data output can reveal an individuals salary
- ❑ Theory of differential privacy: Add noise to remove this link

KU Leuven working in DARPA program Brandeis to produce the Jana database which works on encrypted data, and adds differential privacy based noise.

Looking at applications in US Census and potential UN applications

KATHOLIEKE UNIVERSITEIT LEUVEN + |galois|

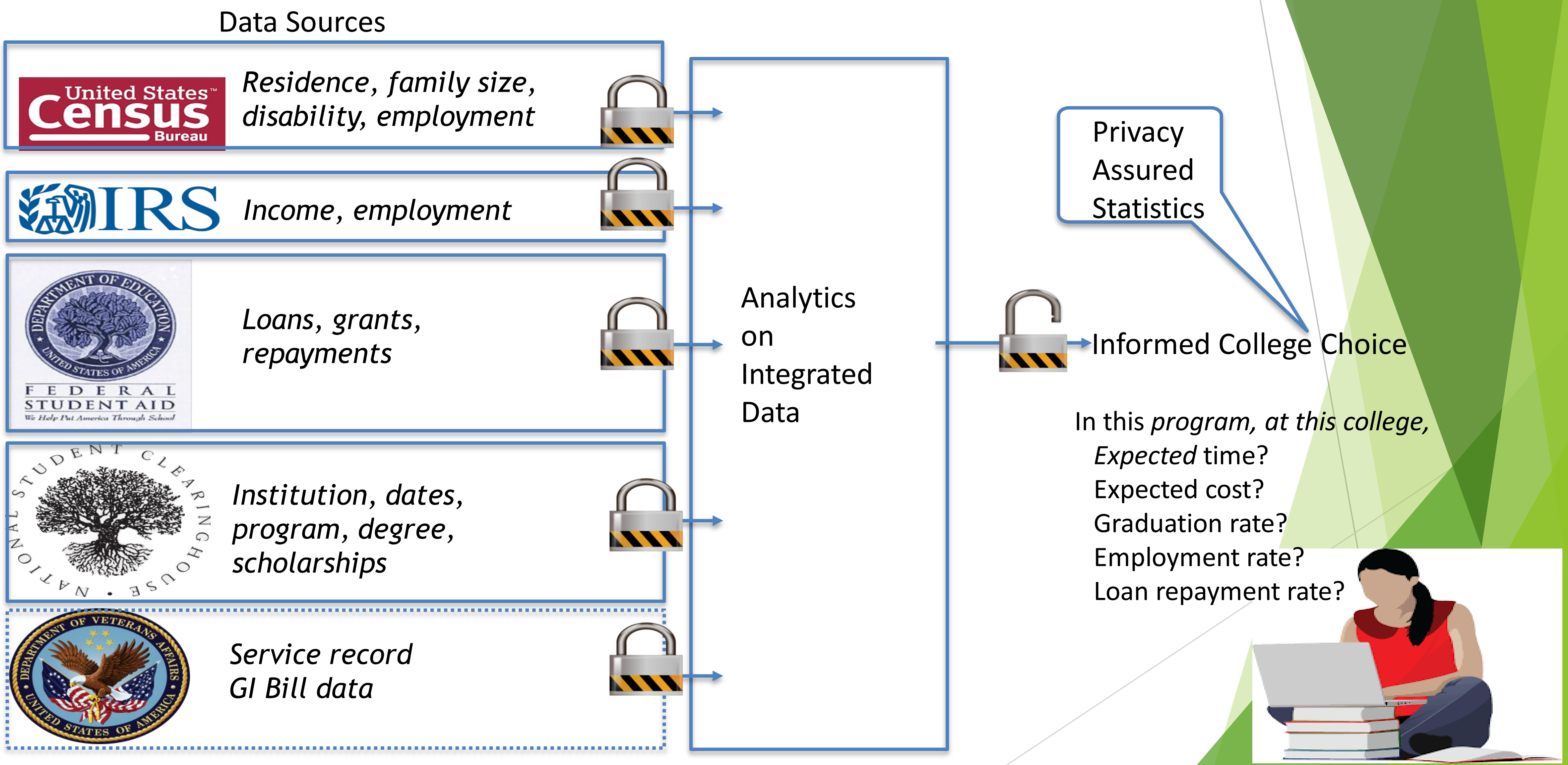
Jana

KATHOLIEKE UNIVERSITEIT LEUVEN

|galois|

- Combines the SCALE-MAMBA system from KU Leuven with a query re-writer from Galois
- SQL queries are dynamically re-written into SCALE bytecodes and executed.
- Differential privacy is added to results
- Number of US gov style applications being created
- Been influential in pushing MPC as a way of creating “knowledge based policy” in the US
    - e.g. “Know before you go” (see next slide)
    - Senator Wyden pushing for MPC in a number of application areas.

“Know Before You Go”



Data Sources

Residence, family size, disability, employment

Income, employment

Loans, grants, repayments

Institution, dates, program, degree, scholarships

Service record GI Bill data

Analytics on Integrated Data

Privacy Assured Statistics

Informed College Choice

United States Census Bureau
IRS
Department of Education United States of America
Federal Student Aid
We Help Put America Through School
National Student Clearinghouse
Department of Veterans Affairs United States of America

In this *program, at this college,*

Expected time?

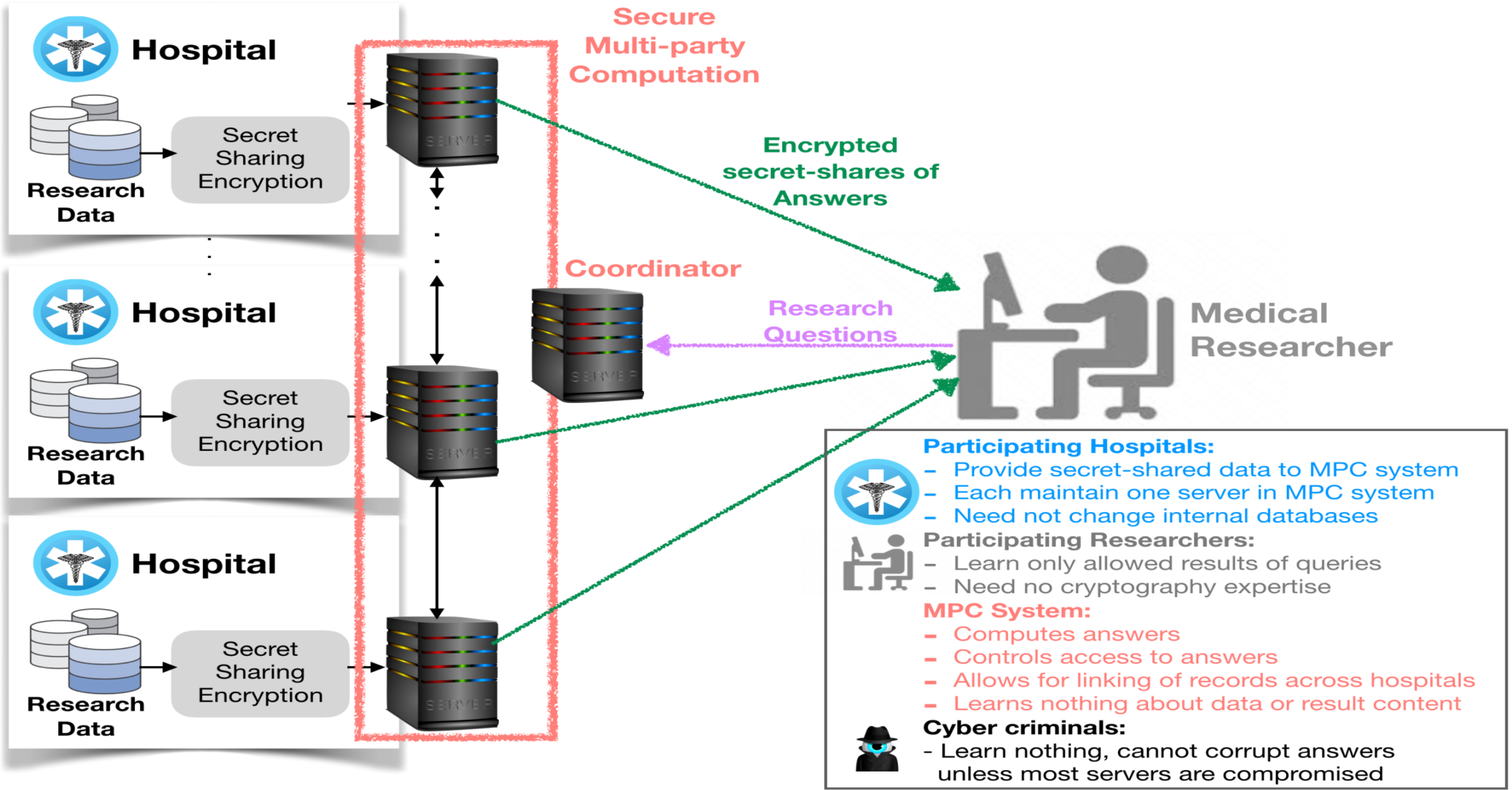
Expected cost?

Graduation rate?

Employment rate?

Loan repayment rate?

US Forward Act: Funding MPC Demo in Heath Care


Hospital
Research Data
Secret Sharing Encryption
Hospital
Research Data
Secret Sharing Encryption
Hospital
Research Data
Secret Sharing Encryption
SERVER
Secure Multi-party Computation
Encrypted secret-shares of Answers
Coordinator
Research Questions
Medical Researcher

Participating Hospitals:
- Provide secret-shared data to MPC system
- Each maintain one server in MPC system
- Need not change internal databases

Participating Researchers:
- Learn only allowed results of queries
- Need no cryptography expertise

MPC System:
- Computes answers
- Controls access to answers
- Allows for linking of records across hospitals
- Learns nothing about data or result content

Cyber criminals:
- Learn nothing, cannot corrupt answers unless most servers are compromised

Securing Cryptographic Keys

Another major applications comes from looking at things in reverse

Major problem in organizations is to secure long term cryptographic data

- ❑ Cryptographic keys for payment operations (EMV system, CAP, etc)
- ❑ Keys for website authentication
- ❑ Password protection mechanisms
- ❑ Hot wallet private signing keys in cryptocurrencies
- ❑ Signing keys for authenticating provisioned blockchains
- ❑ Code signing keys for updates

Securing Cryptographic Keys

The traditional way to do this is via Hardware Security Modules (HSMs) ...

HSMs meant to keep your keys safe

- Only access the key via a specific API
- Key never leaves the hardware module
- HSMs go through validation to ensure they meet minimum requirements

Problems with HSMs

Expensive

Not that secure (update issues, API issues, ...)

Lowest common denominator security requires extra management

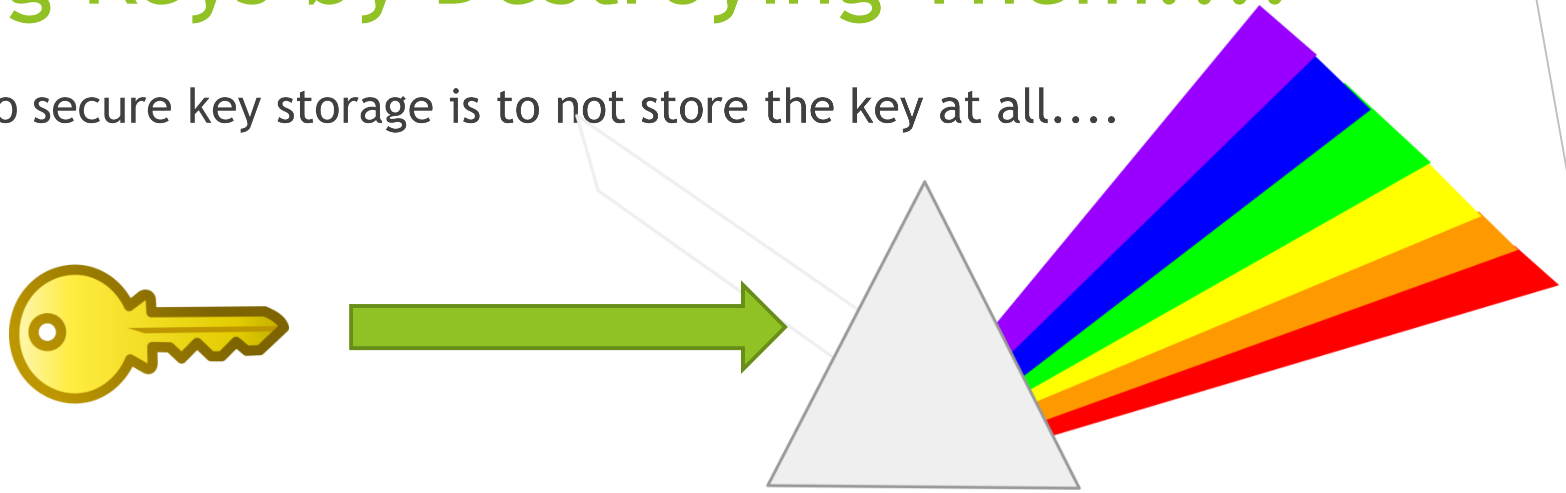
Huge footprint needed for peak load (non-elastic)

Very inflexible API

Not integrated into authorization infrastructure (issue with code-signing)

Securing Keys by Destroying Them....

Another way to secure key storage is to not store the key at all....



Take the key and split it into “shares”

- ❑ The shares reveal no information about the key
- ❑ The shares are never brough back together
- ❑ Required computation is done using MPC

Unbound Tech

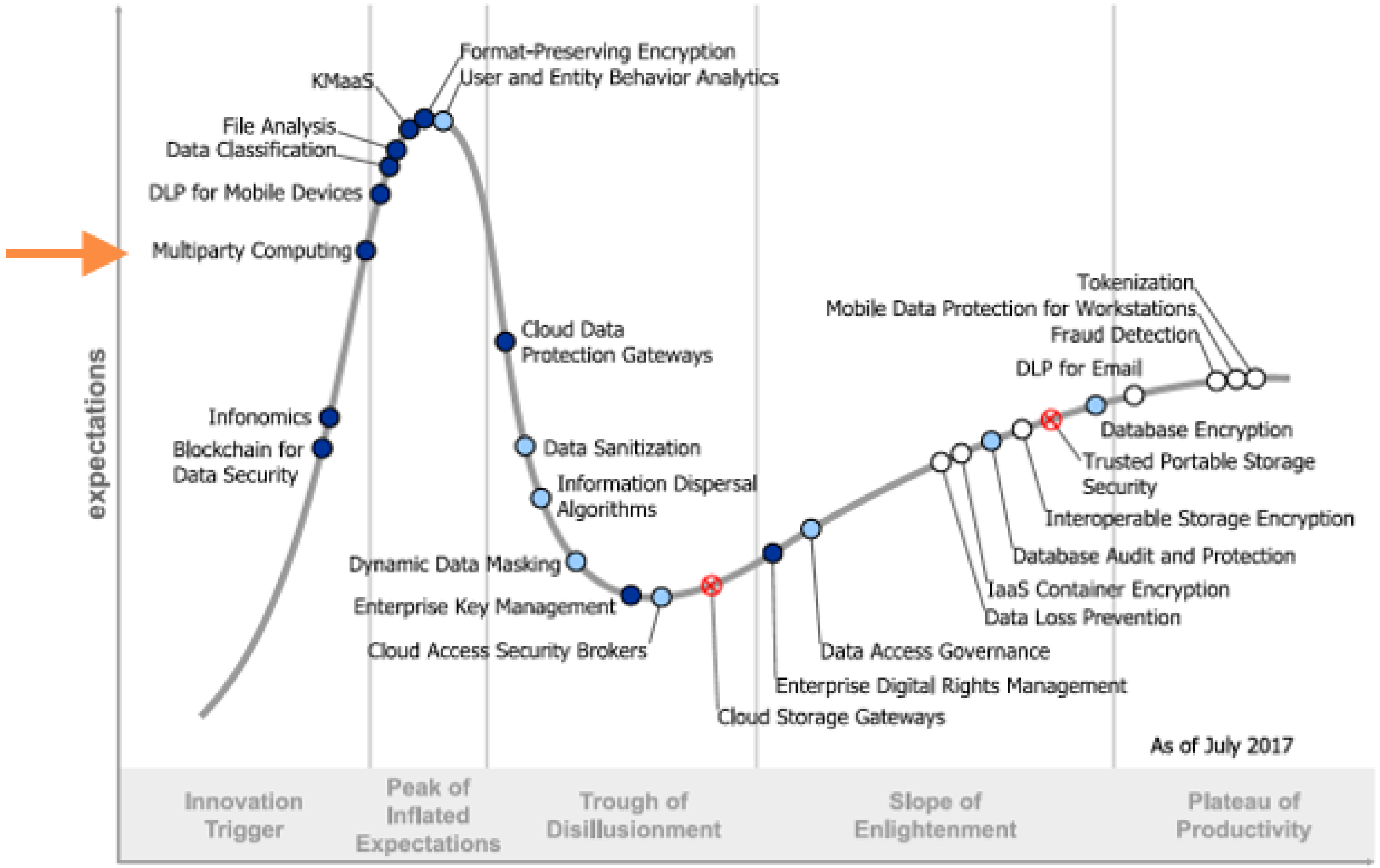
- ❑ Unbound Tech produce a virtual HSM which uses MPC to do this precise thing
- ❑ Enables financial (and other) organizations to move away from inflexible HSMs
- ❑ The first MPC solution to get US government FIPS approval - FIPS-Level 2
- ❑ Major installations in various financial institutions
- ❑ Usage for code-signing by a major computer company
- ❑ Usage for crypto-custodian service for a number of major crypto-exchanges

UNB( )UND

(MATH OVER MATTER)

Gartner Hype Cycle....

Figure 1. Hype Cycle for Data Security, 2017



In five such Gartner reports in 2018

Number of Companies Now in This Space….

baffle

cosmian

DUALITY TECHNOLOGIES

cryptonumerics

BOSCH

Fireblocks

ENVEIL
ENCRYPTED VE

intel®

SEPIOR

|galois|

inpher

Microsoft

PARTISIA

sharemind
BY CYBERNETICA

UNBOUND
(MATH OVER MATTER)

ZenGo

QUESTIONS?