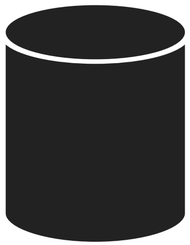
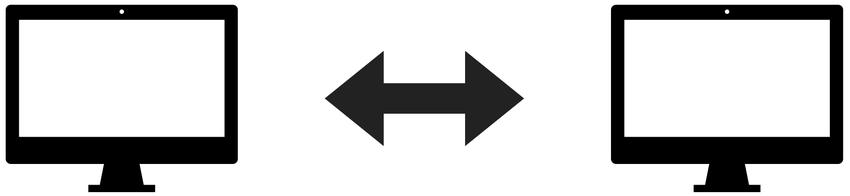
# SECRECY:
# *Secure collaborative analytics in untrusted clouds*

Vasiliki (Vasia) Kalavri
vkalavri@bu.edu
sites.bu.edu/casp


# ΟΛΟΚΛΗΡΩΜΕΝΗ ΠΡΟΣΤΑΣΙΑ ΔΕΔΟΜΕΝΩΝ (end-to-end)



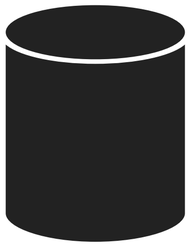
*Κατά την αποθήκευση*



*Κατά τη μεταφορά*

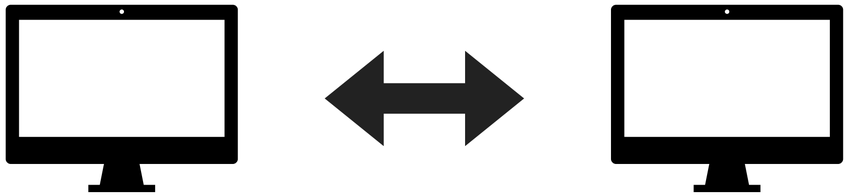
*Σε χρήση*

# ΟΛΟΚΛΗΡΩΜΕΝΗ ΠΡΟΣΤΑΣΙΑ ΔΕΔΟΜΕΝΩΝ (end-to-end)



*Κατά την αποθήκευση*

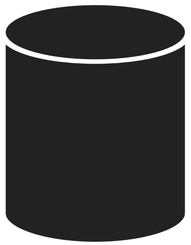
*Advanced Encryption Standard (AES)*



*Κατά τη μεταφορά*

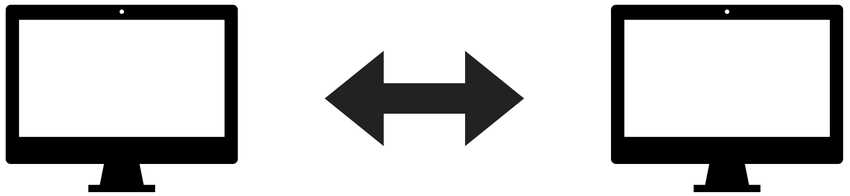
*Σε χρήση*

# ΟΛΟΚΛΗΡΩΜΕΝΗ ΠΡΟΣΤΑΣΙΑ ΔΕΔΟΜΕΝΩΝ (end-to-end)



*Κατά την αποθήκευση*

*Advanced Encryption Standard (AES)*

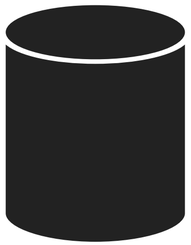


*Κατά τη μεταφορά*

*Transport Layers Security (TLS)*

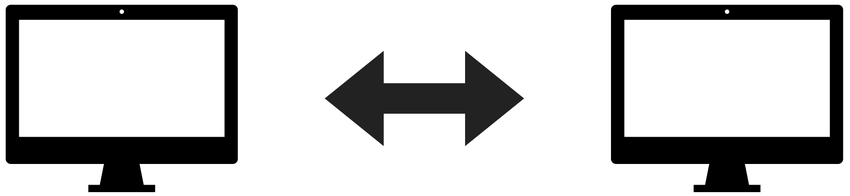
*Σε χρήση*

# ΟΛΟΚΛΗΡΩΜΕΝΗ ΠΡΟΣΤΑΣΙΑ ΔΕΔΟΜΕΝΩΝ (end-to-end)



*Κατά την αποθήκευση*

*Advanced Encryption Standard (AES)*



*Κατά τη μεταφορά*

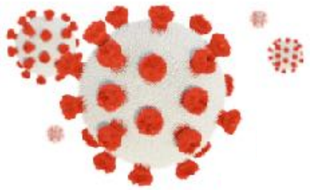
*Transport Layers Security (TLS)*

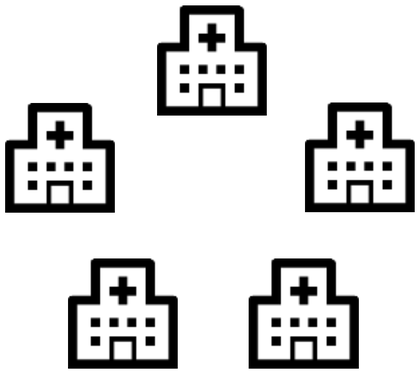
*Σε χρήση*

*Γιατί να προστατεύσουμε δεδομένα σε χρήση;*

# ΣΥΝΕΡΓΑΤΙΚΗ ΑΝΑΛΥΣΗ ΕΥΑΙΣΘΗΤΩΝ ΔΕΔΟΜΕΝΩΝ

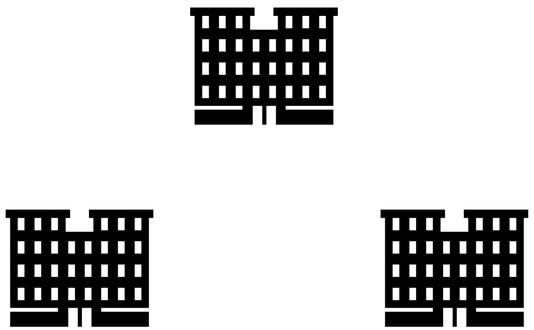
*Παρακολούθηση ασθενειών*





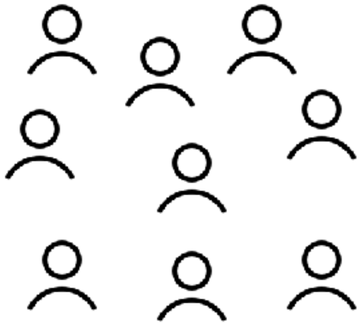
*Πάροχοι υγείας*

*Εξάλειψη μισθολογικού χάσματος*



*Εταιρείες*

*Προστασία προσωπικών δεδομένων στις διαφημίσεις*

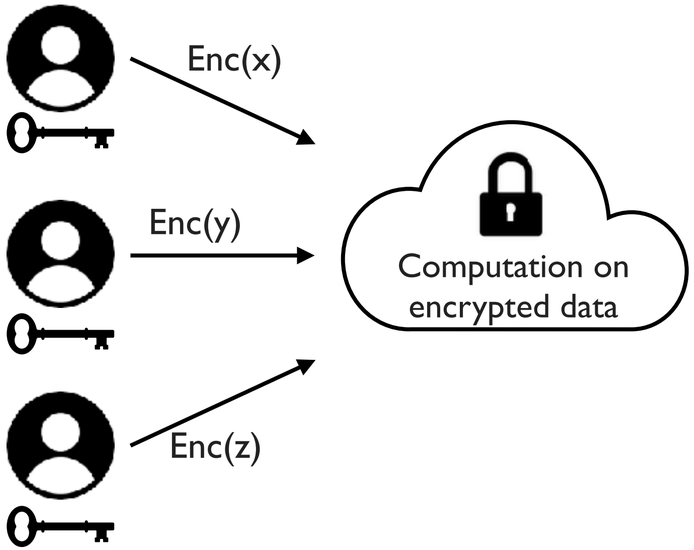


*Χρήστες*

# ΜΕΘΟΔΟΙ ΠΡΟΣΤΑΣΙΑΣ ΔΕΔΟΜΕΝΩΝ «ΣΕ ΧΡΗΣΗ»

# ΜΕΘΟΔΟΙ ΠΡΟΣΤΑΣΙΑΣ ΔΕΔΟΜΕΝΩΝ «ΣΕ ΧΡΗΣΗ»

Fully-Homomorphic Encryption (FHE)

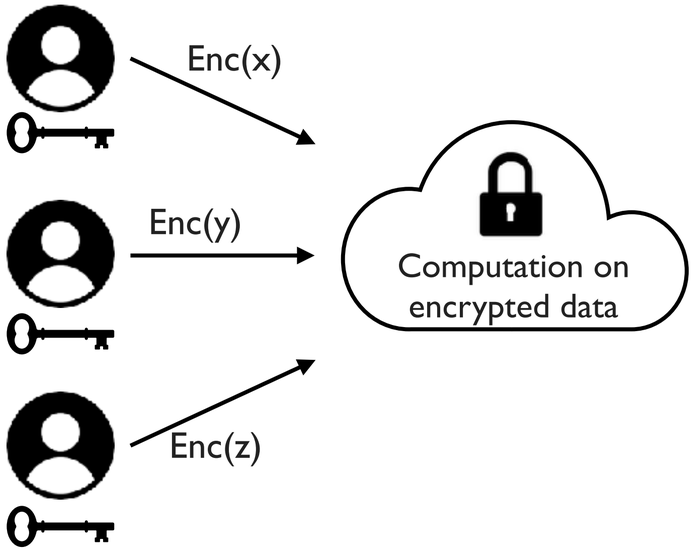



*Security via FH data encryption*

(very high computational cost)

# ΜΕΘΟΔΟΙ ΠΡΟΣΤΑΣΙΑΣ ΔΕΔΟΜΕΝΩΝ «ΣΕ ΧΡΗΣΗ»

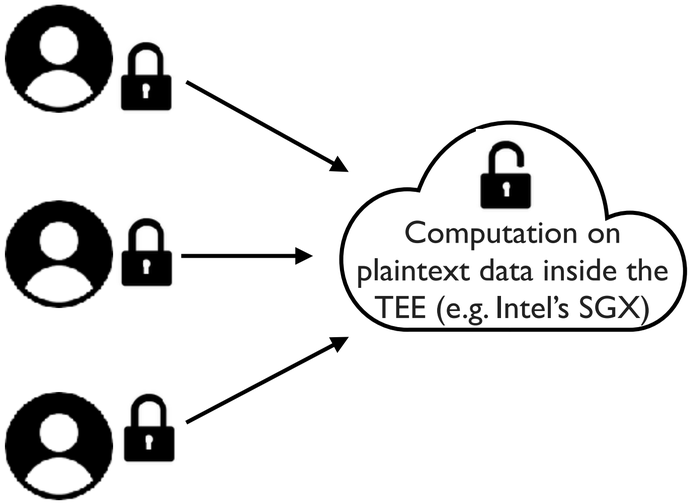
## Fully-Homomorphic Encryption (FHE)




*Security via FH data encryption*

(very high computational cost)

## Trusted Execution Environments (TEEs)

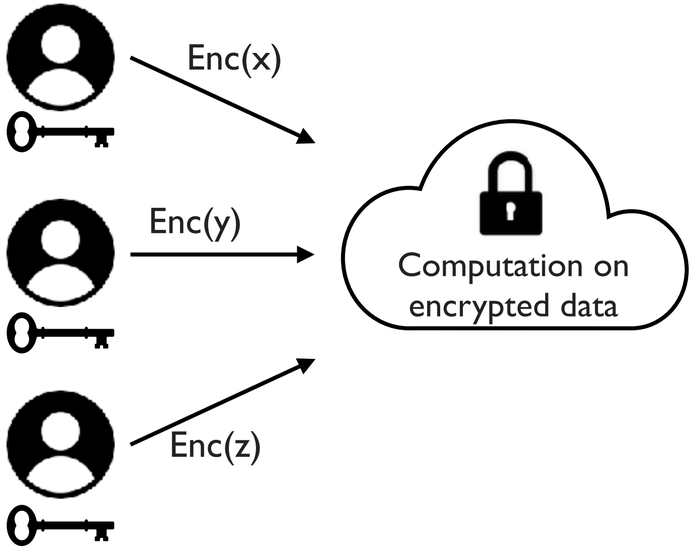



*Security via physically protected HW*

(prone to side-channel attacks)

# ΜΕΘΟΔΟΙ ΠΡΟΣΤΑΣΙΑΣ ΔΕΔΟΜΕΝΩΝ «ΣΕ ΧΡΗΣΗ»

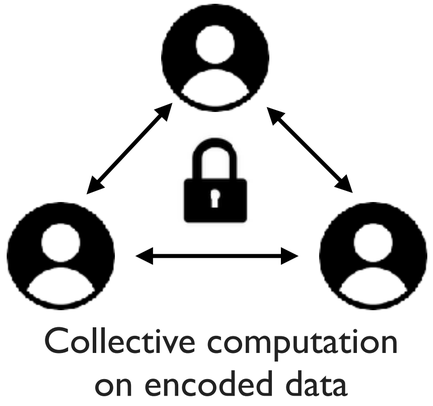
## Fully-Homomorphic Encryption (FHE)




*Security via FH data encryption*

(very high computational cost)

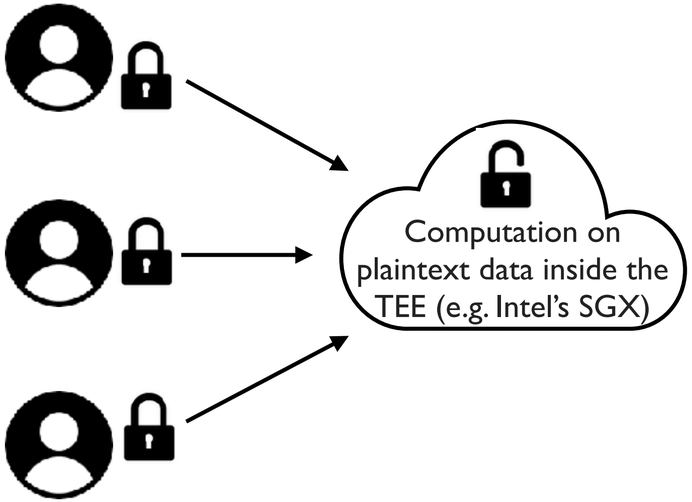
## Secure Multi-party Computation (MPC)




*Security via decentralized trust*

(high communication cost)

## Trusted Execution Environments (TEEs)

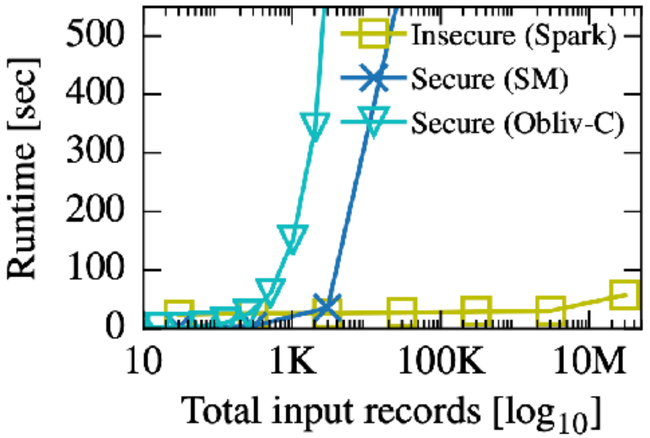



*Security via physically protected HW*

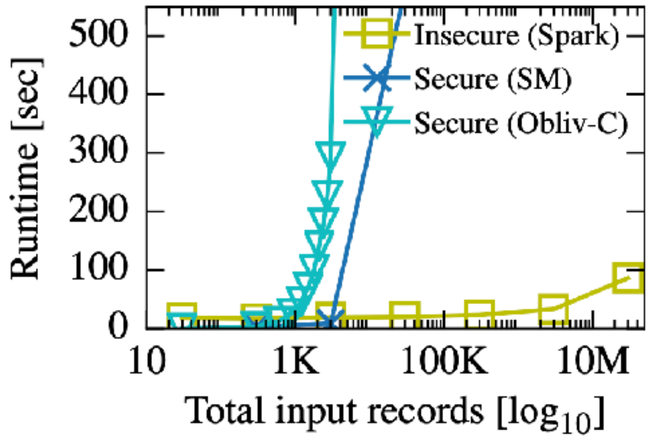
(prone to side-channel attacks)

# Η ΑΣΦΑΛΗΣ ΣΥΝΕΡΓΑΤΙΚΗ ΑΝΑΛΥΣΗ ΔΕΔΟΜΕΝΩΝ ΕΙΝΑΙ ΑΚΡΙΒΗ

Aggregation




Join




*"Running the query entirely under MPC […] fails to scale beyond 3,000 total records…"*

| Test | Plaintext | Secure | Slowdown |
|---|---|---|---|
| *Comorbidity* | 158 | 253,894 | 1,609X |
| *Recurrent C. Diff* | 165 | 159,145 | 967X |
| *Aspirin Count* | 193 | 8,195,317 | 43,337X |

*"The primary source of the slowdown arises from their join operators that have hundreds of input tuples…"*

[1] N. Volgushev, M. Schwarzkopf, B. Getchell, M. Varia, A. Lapets, and A. Bestavros. *Conclave: secure multi-party computation on big data*. EuroSys, 2019.

[2] J. Bater, G. Elliott, C. Eggen, S. Goel, A. N. Kho, and J. Rogers. *SMCQL: secure querying for federated databases*. PVLDB, 10(6):673–684, 2017.

# ΣΥΜΒΙΒΑΖΟΝΤΑΣ ΑΣΦΑΛΕΙΑ ΚΑΙ ΑΠΟΔΟΣΗ:

## Διαρροή πληροφοριών

Leaking **access patterns** or **intermediate results sizes** to untrusted parties, with or without noise

## Υβριδικός υπολογισμός

Require data owners to participate in the secure computation with **trusted local resources**

# ΣΥΜΒΙΒΑΖΟΝΤΑΣ ΑΣΦΑΛΕΙΑ ΚΑΙ ΑΠΟΔΟΣΗ:

## Διαρροή πληροφοριών

Leaking **access patterns** or **intermediate results sizes** to untrusted parties, with or without noise

Η διαρροή πληροφοριών μπορεί να διευκολύνει την **ανακατασκευή δεδομένων** και να αποκαλύψει ευαίσθητες πληροφορίες:

```
SELECT *
FROM EMPLOYEE
WHERE GENDER='F'
AND RANK='manager'
AND SALARY>50000
```

## Υβριδικός υπολογισμός

Require data owners to participate in the secure computation with **trusted local resources**

# ΣΥΜΒΙΒΑΖΟΝΤΑΣ ΑΣΦΑΛΕΙΑ ΚΑΙ ΑΠΟΔΟΣΗ:

## Διαρροή πληροφοριών

Leaking **access patterns** or **intermediate results sizes** to untrusted parties, with or without noise

Η διαρροή πληροφοριών μπορεί να διευκολύνει την **ανακατασκευή δεδομένων** και να αποκαλύψει ευαίσθητες πληροφορίες:

```
SELECT *
FROM EMPLOYEE
WHERE GENDER='F'
AND RANK='manager'
AND SALARY>50000
```

```
if (p1.salary)>(p2.salary) => gap
```

## Υβριδικός υπολογισμός

Require data owners to participate in the secure computation with **trusted local resources**

# ΣΥΜΒΙΒΑΖΟΝΤΑΣ ΑΣΦΑΛΕΙΑ ΚΑΙ ΑΠΟΔΟΣΗ:

## Διαρροή πληροφοριών

Leaking **access patterns** or **intermediate results sizes** to untrusted parties, with or without noise

Η διαρροή πληροφοριών μπορεί να διευκολύνει την **ανακατασκευή δεδομένων** και να αποκαλύψει ευαίσθητες πληροφορίες:

```
SELECT *
FROM EMPLOYEE
WHERE GENDER='F'
AND RANK='manager'
AND SALARY>50000
```

```
if (p1.salary)>(p2.salary) => gap
```

## Υβριδικός υπολογισμός

Require data owners to participate in the secure computation with **trusted local resources**

Οι κάτοχοι δεδομένων ενδέχεται να μην διαθέτουν **την απαραίτητη υποδομή ή την τεχνογνωσία** που απαιτείται για ασφαλή υπολογισμό, π.χ. νοσοκομεία, τελικοί χρήστες

***Στόχος***: *Πρακτική απόδοση*
*με αποδειξιμες εγγυήσεις ασφάλειας*
*χωρίς χρήση ιδιωτικών πόρων*
*(outsourced setting)*

# Secrecy

*Secure collaborative analytics in untrusted Clouds*

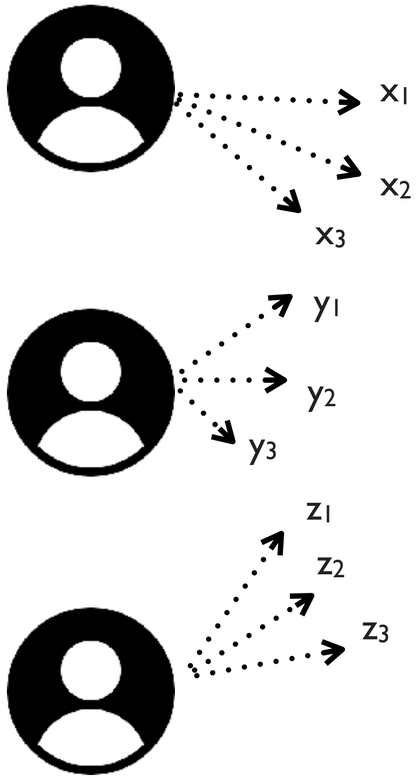
***No information leakage***

***No trusted resources***

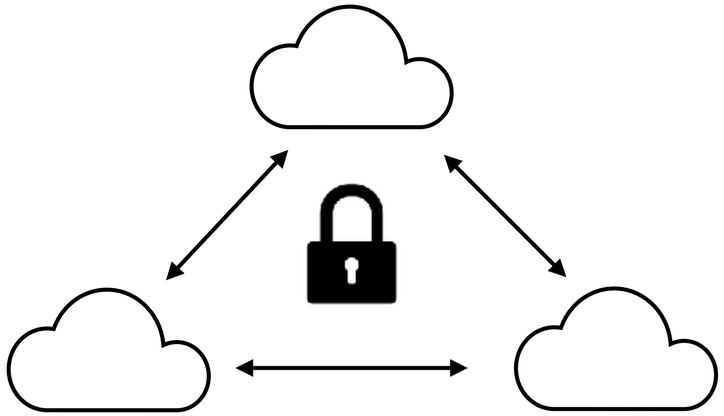
***Complex SQL queries***

# OUTSOURCED SECURE MULTI-PARTY COMPUTATION (MPC)

Κάτοχοι δεδομένων



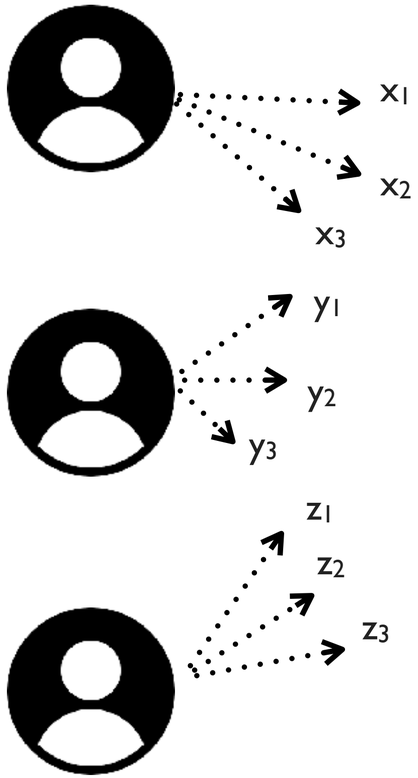

Υπολογιστικοί συμμέτοχοι



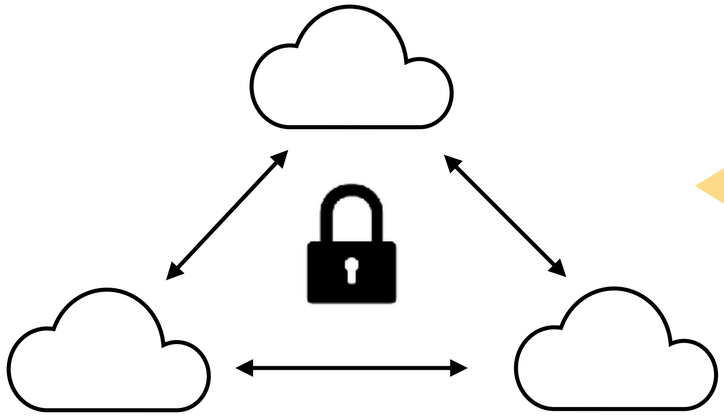
Μη-εμπιστεύσιμες και μη-συνεργαζόμενες οντότητες υπολογίζουν συλλογικά σε τυχαία "μερίδια δεδομένων"
( random data shares)

# OUTSOURCED SECURE MULTI-PARTY COMPUTATION (MPC)

Κάτοχοι δεδομένων




Υπολογιστικοί συμμέτοχοι



Μη-εμπιστεύσιμες και μη-συνεργαζόμενες οντότητες υπολογίζουν συλλογικά σε τυχαία "μερίδια δεδομένων" ( random data shares)

Οι υπολογιστικοί συμμέτοχοι μπορεί να είναι:

- Ανταγωνιστικοί πάροχοι υπηρεσιών Cloud
- Πόροι διαχειριζόμενοι από διαφορετικές οντότητες σε ένα federated Cloud

# ΜΟΝΤΕΛΟ ΚΑΙ ΕΓΓΥΗΣΕΙΣ ΑΣΦΑΛΕΙΑΣ

Semi-honest model

- Οι συμμέτοχοι δεν αποκλίνουν από το προτόκολο (“honest but curious”)
- Ο “αντίπαλος” (adversary) επιτρέπεται να παρακολουθεί το δίκτυο και να αποκτήσει τον έλεγχο ενός από τους συμμέτοχους (*χωρίς να μεταβάλλει την εκτέλεση του*)

Security guarantees

Μη-εμπιστεύσιμες οντότητες δε θα μάθουν τίποτα σχετικό με:

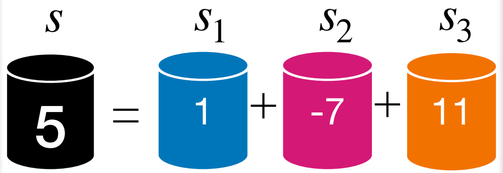
- Τα αρχικά δεδομένα, τα ενδιάμεσα αποτελέσματα και το μέγεθός τους
- Τα μοτίβα προσπέλασης δεδομένων κατά την εκτέλεση των ερωτημάτων

# ΠΑΡΑΔΕΙΓΜΑ: ΑΣΦΑΛΗΣ ΠΡΟΣΘΕΣΗ ΚΑΙ ΠΟΛΛΑΠΛΑΣΙΑΣΜΟΣ

Arithmetic sharing: $x = x_1 + x_2 + x_3 \pmod{2^{64}}$

Party 1

Κάτοχος A

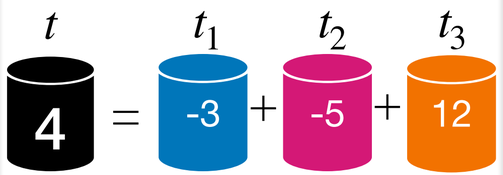



*Τυχαία μερίδια δεδομένων*

Party 3

Party 2

Κάτοχος B

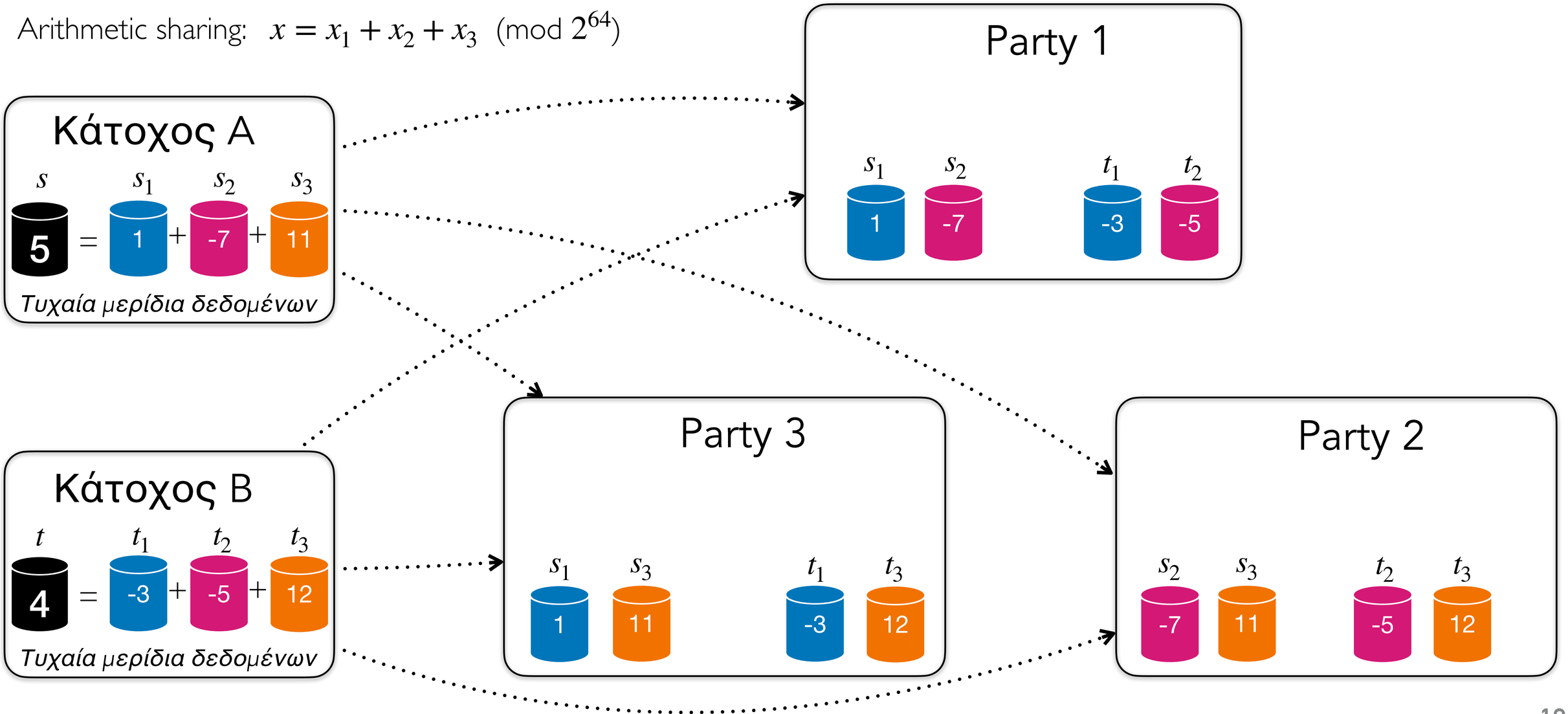



*Τυχαία μερίδια δεδομένων*

T. Araki, J. Furukawa, Y. Lindell, A. Nof, and K. Ohara. *High-Throughput Semi-Honest Secure Three-Party Computation with an Honest Majority*. CCS, 2016.

# ΠΑΡΑΔΕΙΓΜΑ: ΑΣΦΑΛΗΣ ΠΡΟΣΘΕΣΗ ΚΑΙ ΠΟΛΛΑΠΛΑΣΙΑΣΜΟΣ

Arithmetic sharing: $x = x_1 + x_2 + x_3 \pmod{2^{64}}$

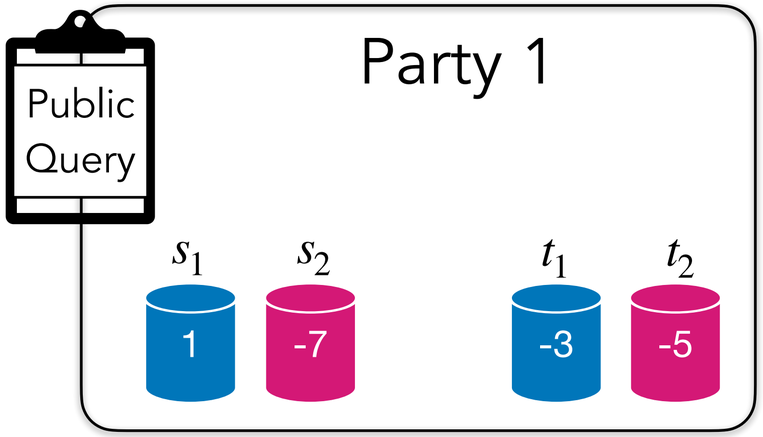



T. Araki, J. Furukawa, Y. Lindell, A. Nof, and K. Ohara. *High-Throughput Semi-Honest Secure Three-Party Computation with an Honest Majority*. CCS, 2016.

# ΠΑΡΑΔΕΙΓΜΑ: ΑΣΦΑΛΗΣ ΠΡΟΣΘΕΣΗ ΚΑΙ ΠΟΛΛΑΠΛΑΣΙΑΣΜΟΣ

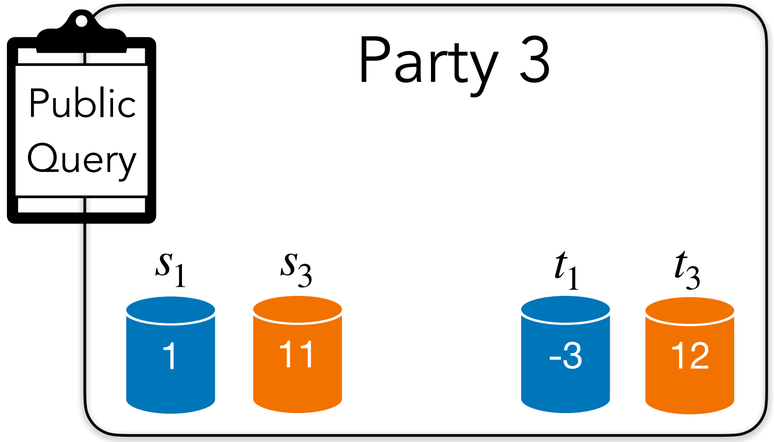
Arithmetic sharing: $x = x_1 + x_2 + x_3 \pmod{2^{64}}$

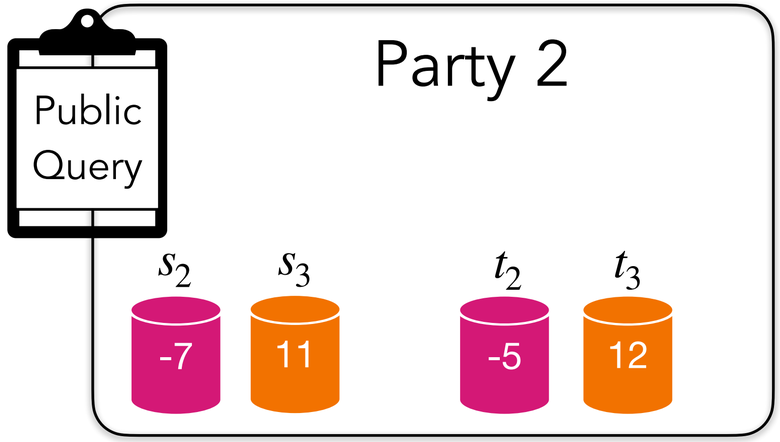





Query: $s + t$

$$s + t = (s_1 + t_1) + (s_2 + t_2) + (s_3 + t_3)$$

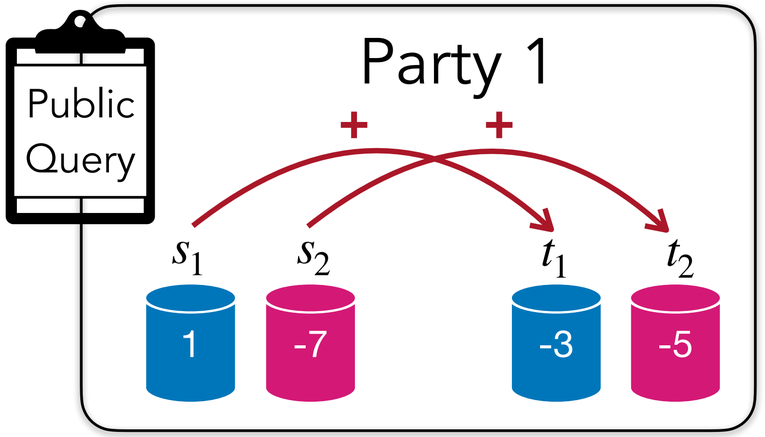



T. Araki, J. Furukawa, Y. Lindell, A. Nof, and K. Ohara. *High-Throughput Semi-Honest Secure Three-Party Computation with an Honest Majority*. CCS, 2016.

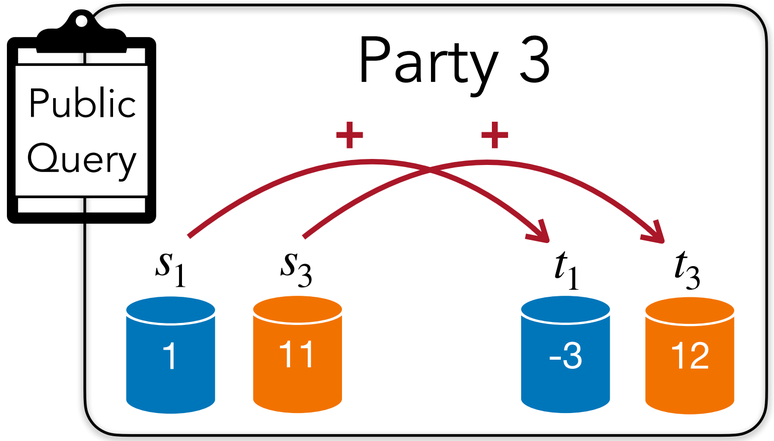
# ΠΑΡΑΔΕΙΓΜΑ: ΑΣΦΑΛΗΣ ΠΡΟΣΘΕΣΗ ΚΑΙ ΠΟΛΛΑΠΛΑΣΙΑΣΜΟΣ

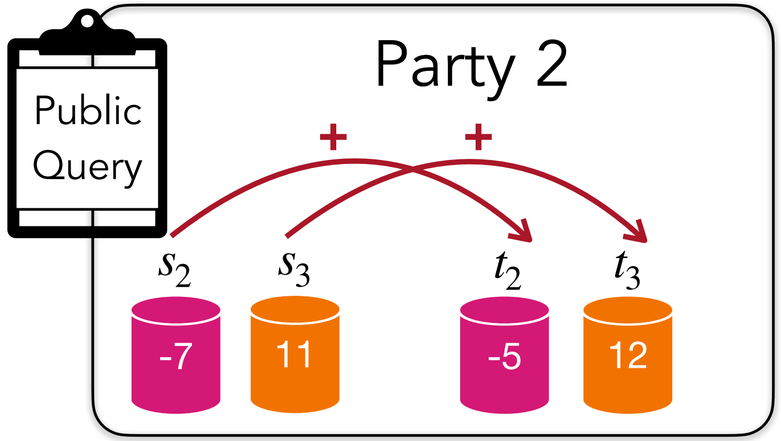
Arithmetic sharing: $x = x_1 + x_2 + x_3 \pmod{2^{64}}$


Query: $s + t$

$$s + t = (s_1 + t_1) + (s_2 + t_2) + (s_3 + t_3)$$

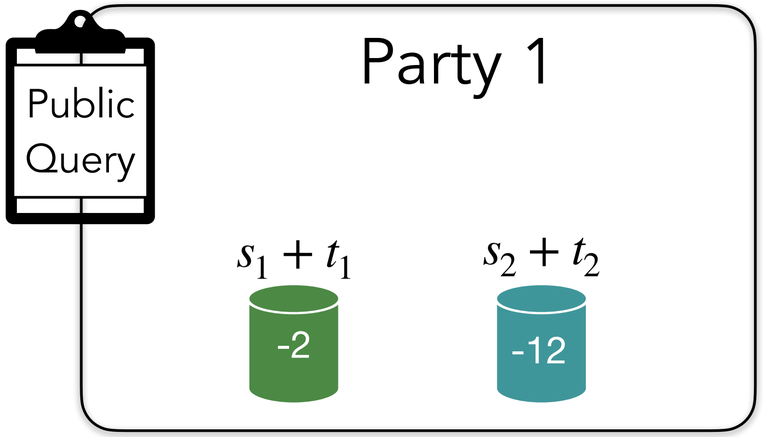




T. Araki, J. Furukawa, Y. Lindell, A. Nof, and K. Ohara. *High-Throughput Semi-Honest Secure Three-Party Computation with an Honest Majority*. CCS, 2016.

# ΠΑΡΑΔΕΙΓΜΑ: ΑΣΦΑΛΗΣ ΠΡΟΣΘΕΣΗ ΚΑΙ ΠΟΛΛΑΠΛΑΣΙΑΣΜΟΣ

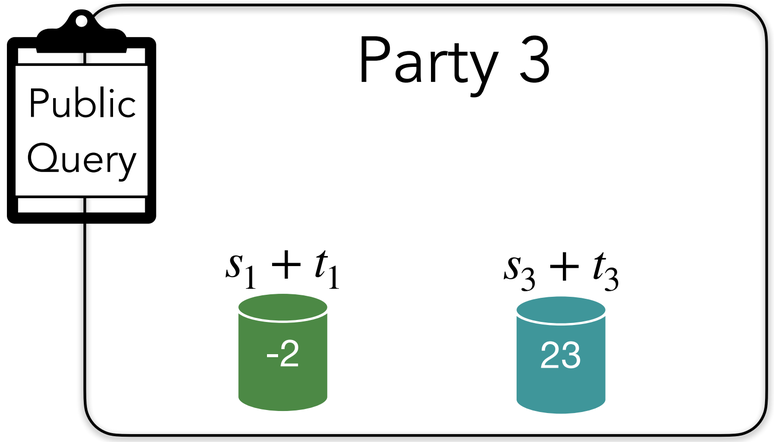
Arithmetic sharing: $x = x_1 + x_2 + x_3 \pmod{2^{64}}$

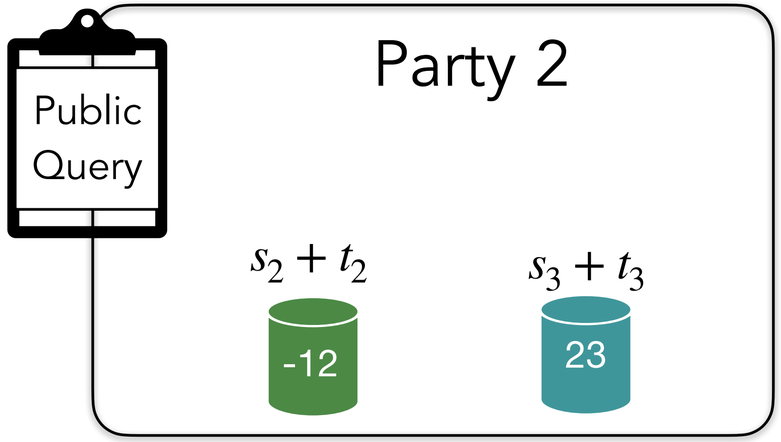





Query: $s + t$

$$s + t = (s_1 + t_1) + (s_2 + t_2) + (s_3 + t_3)$$



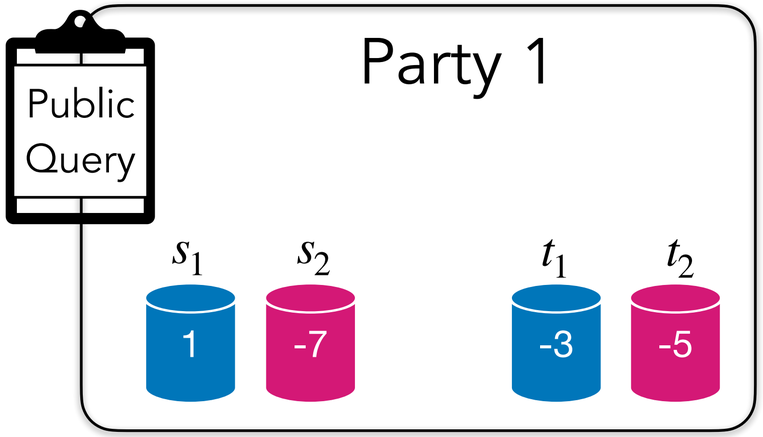




T. Araki, J. Furukawa, Y. Lindell, A. Nof, and K. Ohara. *High-Throughput Semi-Honest Secure Three-Party Computation with an Honest Majority*. CCS, 2016.

# ΠΑΡΑΔΕΙΓΜΑ: ΑΣΦΑΛΗΣ ΠΡΟΣΘΕΣΗ ΚΑΙ ΠΟΛΛΑΠΛΑΣΙΑΣΜΟΣ

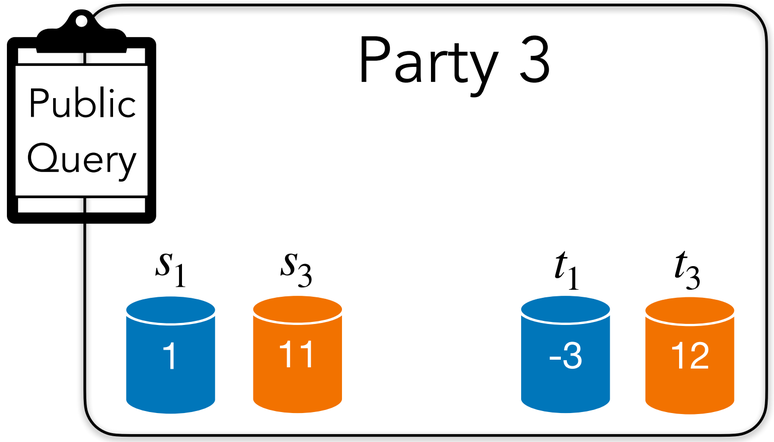
Arithmetic sharing: $x = x_1 + x_2 + x_3 \pmod{2^{64}}$

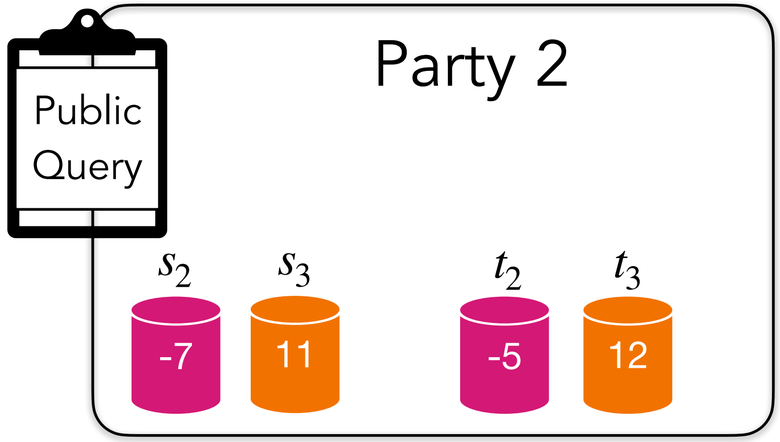





Query: $s \times t$

$$s \times t = (s_1 + s_2 + s_3) \cdot (t_1 + t_2 + t_3)$$

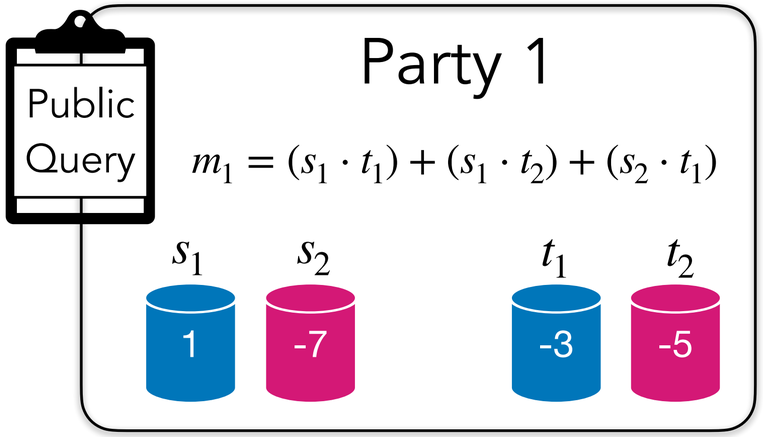



T. Araki, J. Furukawa, Y. Lindell, A. Nof, and K. Ohara. *High-Throughput Semi-Honest Secure Three-Party Computation with an Honest Majority*. CCS, 2016.

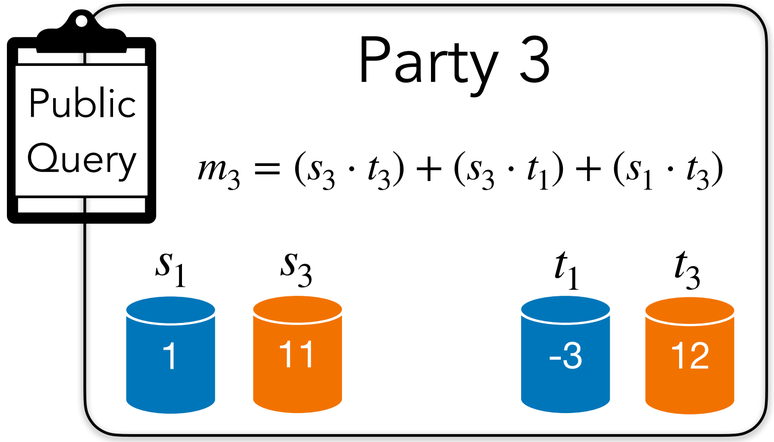
# ΠΑΡΑΔΕΙΓΜΑ: ΑΣΦΑΛΗΣ ΠΡΟΣΘΕΣΗ ΚΑΙ ΠΟΛΛΑΠΛΑΣΙΑΣΜΟΣ

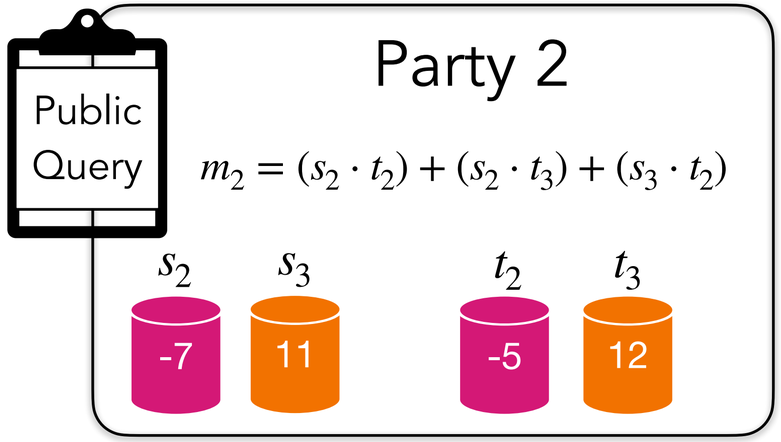
Arithmetic sharing: $x = x_1 + x_2 + x_3 \pmod{2^{64}}$


Query: $s \times t$

$$s \times t = (s_1 + s_2 + s_3) \cdot (t_1 + t_2 + t_3) = \cdots = m_1 + m_2 + m_3$$

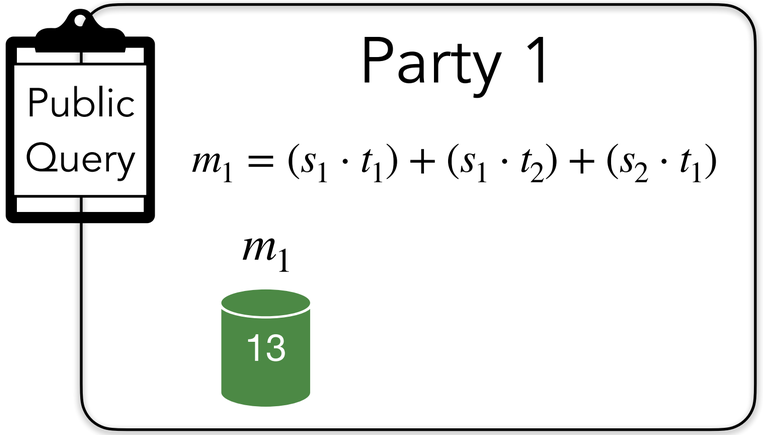




T. Araki, J. Furukawa, Y. Lindell, A. Nof, and K. Ohara. *High-Throughput Semi-Honest Secure Three-Party Computation with an Honest Majority*. CCS, 2016.

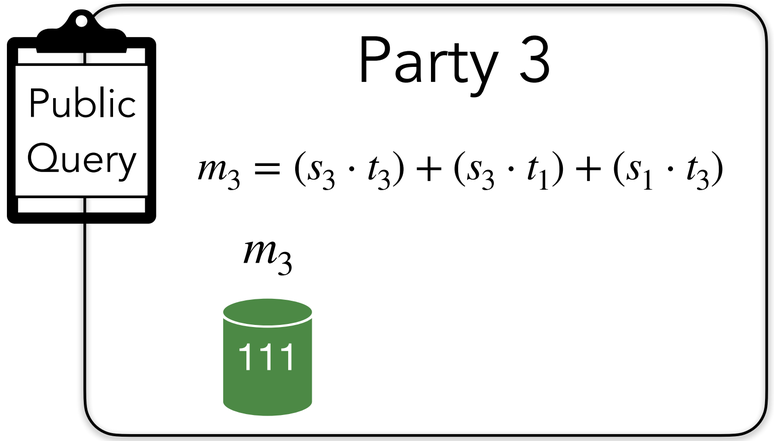
# ΠΑΡΑΔΕΙΓΜΑ: ΑΣΦΑΛΗΣ ΠΡΟΣΘΕΣΗ ΚΑΙ ΠΟΛΛΑΠΛΑΣΙΑΣΜΟΣ

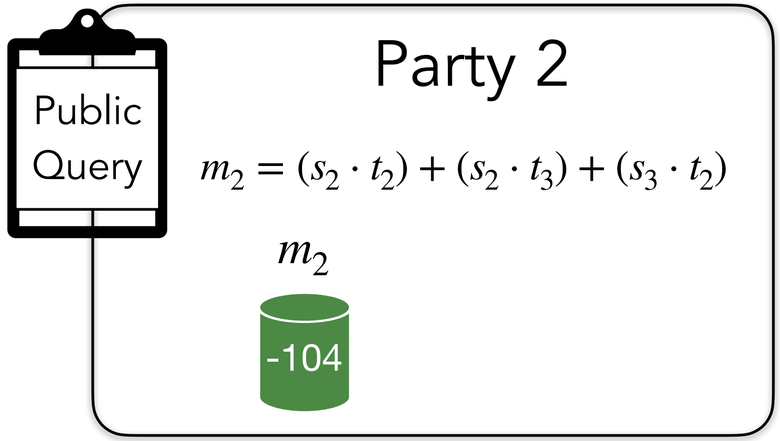
Arithmetic sharing: $x = x_1 + x_2 + x_3 \pmod{2^{64}}$


Query: $s \times t$

$$s \times t = (s_1 + s_2 + s_3) \cdot (t_1 + t_2 + t_3) = \cdots = m_1 + m_2 + m_3$$

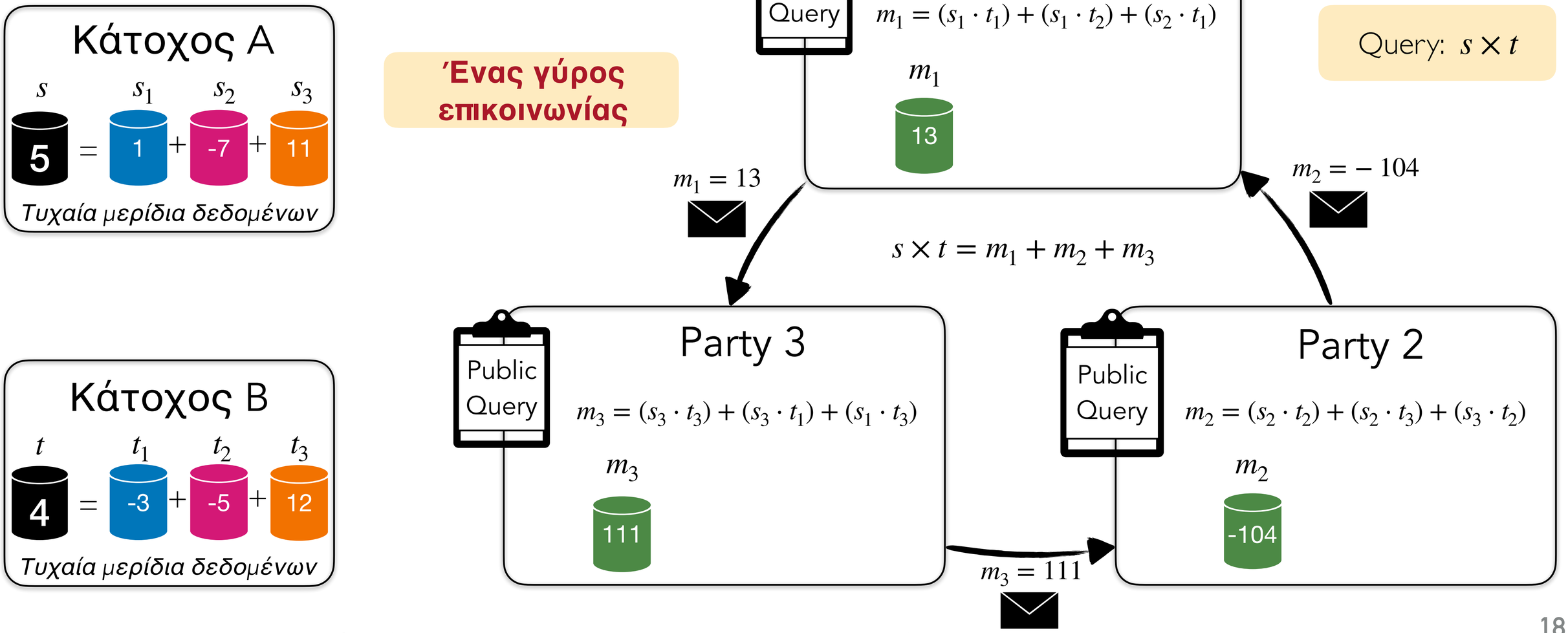




T. Araki, J. Furukawa, Y. Lindell, A. Nof, and K. Ohara. *High-Throughput Semi-Honest Secure Three-Party Computation with an Honest Majority*. CCS, 2016.

# ΠΑΡΑΔΕΙΓΜΑ: ΑΣΦΑΛΗΣ ΠΡΟΣΘΕΣΗ ΚΑΙ ΠΟΛΛΑΠΛΑΣΙΑΣΜΟΣ

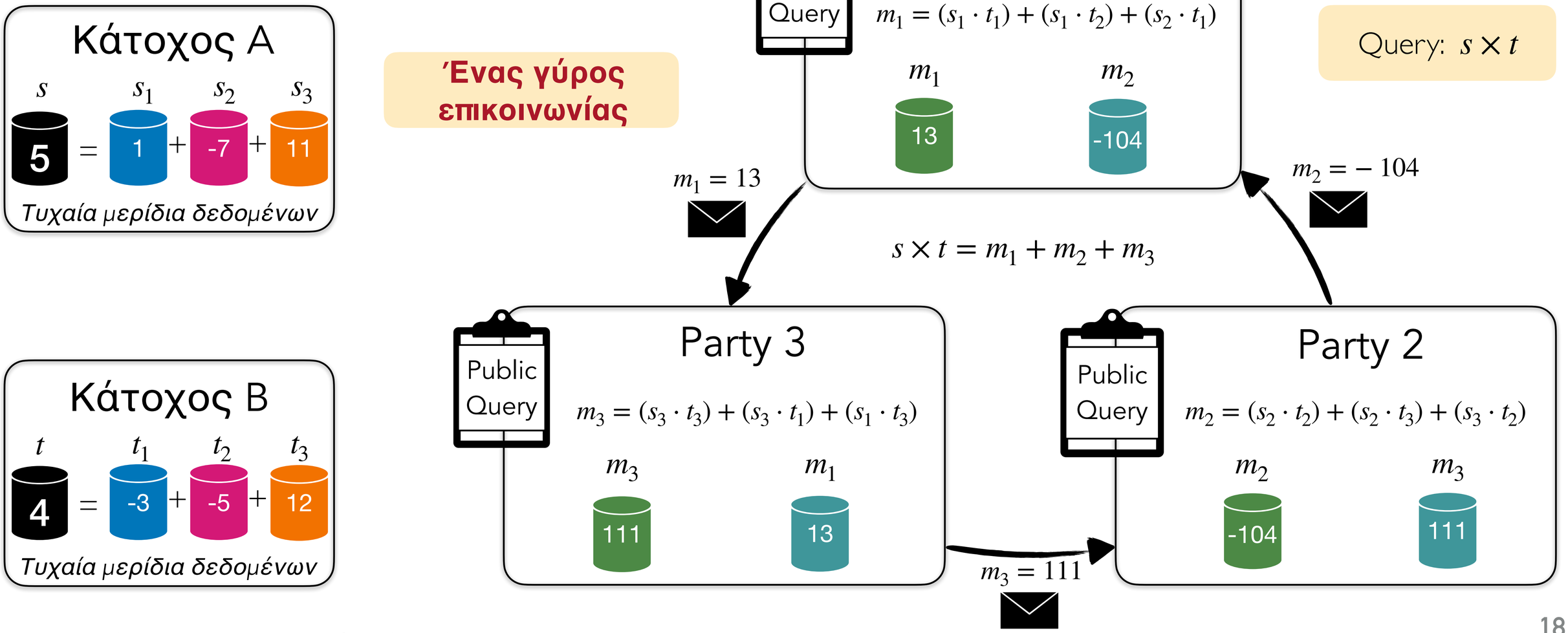
Arithmetic sharing: $x = x_1 + x_2 + x_3 \pmod{2^{64}}$




T. Araki, J. Furukawa, Y. Lindell, A. Nof, and K. Ohara. *High-Throughput Semi-Honest Secure Three-Party Computation with an Honest Majority*. CCS, 2016.

# ΠΑΡΑΔΕΙΓΜΑ: ΑΣΦΑΛΗΣ ΠΡΟΣΘΕΣΗ ΚΑΙ ΠΟΛΛΑΠΛΑΣΙΑΣΜΟΣ

Arithmetic sharing: $x = x_1 + x_2 + x_3 \pmod{2^{64}}$




T. Araki, J. Furukawa, Y. Lindell, A. Nof, and K. Ohara. *High-Throughput Semi-Honest Secure Three-Party Computation with an Honest Majority*. CCS, 2016.

# ΑΠΟ ΤΗΝ ΑΣΦΑΛΗ ΠΡΟΣΘΕΣΗ ΚΑΙ ΤΟΝ ΠΟΛΛΑΠΛΑΣΙΑΣΜΟ ΣΤΗ ΣΧΕΣΙΑΚΗ ΑΝΑΛΥΣΗ ΔΕΔΟΜΕΝΩΝ

Arithmetic sharing: $x = x_1 + x_2 + x_3 \pmod{2^{64}}$

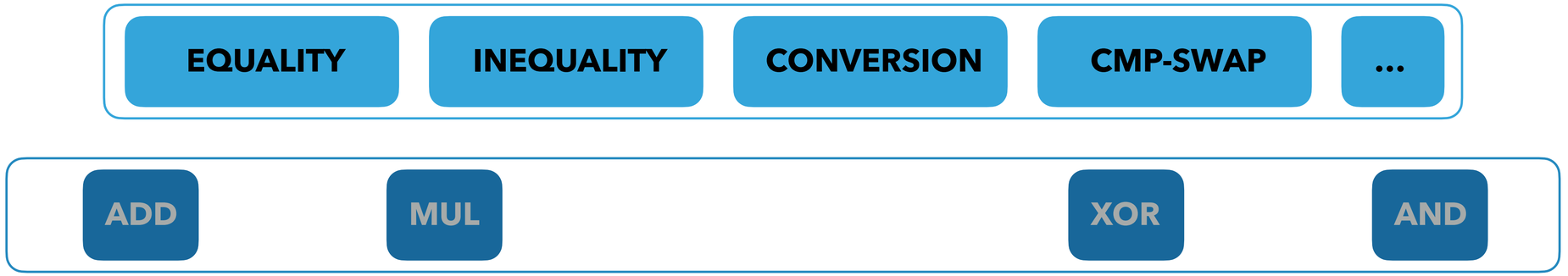
Boolean sharing: $x = x_1 \oplus x_2 \oplus x_3$

# ΑΠΟ ΤΗΝ ΑΣΦΑΛΗ ΠΡΟΣΘΕΣΗ ΚΑΙ ΤΟΝ ΠΟΛΛΑΠΛΑΣΙΑΣΜΟ ΣΤΗ ΣΧΕΣΙΑΚΗ ΑΝΑΛΥΣΗ ΔΕΔΟΜΕΝΩΝ


Arithmetic sharing: $x = x_1 + x_2 + x_3 \pmod{2^{64}}$

Boolean sharing: $x = x_1 \oplus x_2 \oplus x_3$

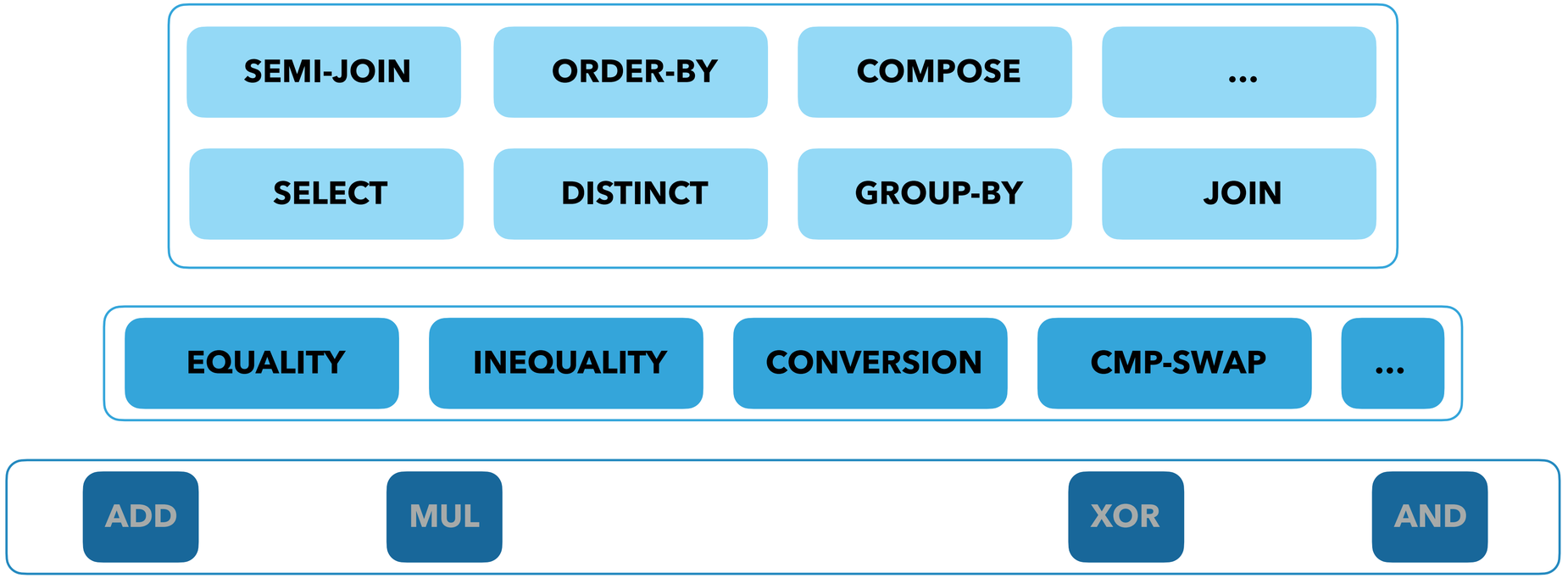
# ΑΠΟ ΤΗΝ ΑΣΦΑΛΗ ΠΡΟΣΘΕΣΗ ΚΑΙ ΤΟΝ ΠΟΛΛΑΠΛΑΣΙΑΣΜΟ ΣΤΗ ΣΧΕΣΙΑΚΗ ΑΝΑΛΥΣΗ ΔΕΔΟΜΕΝΩΝ




Arithmetic sharing: $x = x_1 + x_2 + x_3 \pmod{2^{64}}$

Boolean sharing: $x = x_1 \oplus x_2 \oplus x_3$

# ΑΠΟ ΤΗΝ ΑΣΦΑΛΗ ΠΡΟΣΘΕΣΗ ΚΑΙ ΤΟΝ ΠΟΛΛΑΠΛΑΣΙΑΣΜΟ ΣΤΗ ΣΧΕΣΙΑΚΗ ΑΝΑΛΥΣΗ ΔΕΔΟΜΕΝΩΝ




Arithmetic sharing: $x = x_1 + x_2 + x_3 \pmod{2^{64}}$

Boolean sharing: $x = x_1 \oplus x_2 \oplus x_3$

# OBLIVIOUS COMPUTATION

Οι συμμέτοχοι εκτελούν πανομοιότυπους υπολογισμούς που είναι **ανεξάρτητοι από τα χαρακτηριστικά των δεδομένων**

- Τα μοτίβα προσπέλασης δεν εξαρτώνται από τα δεδομένα εισόδου
- Δεν εκτελούνται συνθήκες και διακλαδώσεις
- Το μέγεθος των ενδιάμεσων δεδομένων δεν μειώνεται

"If the most significant bits are not the same, then $a$ is greater than $b$ when $a_2$ is set"

# OBLIVIOUS COMPUTATION

Οι συμμέτοχοι εκτελούν πανομοιότυπους υπολογισμούς που είναι **ανεξάρτητοι από τα χαρακτηριστικά των δεδομένων**

- Τα μοτίβα προσπέλασης δεν εξαρτώνται από τα δεδομένα εισόδου
- Δεν εκτελούνται συνθήκες και διακλαδώσεις
- Το μέγεθος των ενδιάμεσων δεδομένων δεν μειώνεται

If $(a > b)$ $\{\ldots\}$

"If the most significant bits are not the same, then $a$ is greater than $b$ when $a_2$ is set"

*Cleartext*

# OBLIVIOUS COMPUTATION

Οι συμμέτοχοι εκτελούν πανομοιότυπους υπολογισμούς που είναι **ανεξάρτητοι από τα χαρακτηριστικά των δεδομένων**

- Τα μοτίβα προσπέλασης δεν εξαρτώνται από τα δεδομένα εισόδου
- Δεν εκτελούνται συνθήκες και διακλαδώσεις
- Το μέγεθος των ενδιάμεσων δεδομένων δεν μειώνεται

* *Both $a$ and $b$ are secret-shared*

**For 3-bit numbers:** $a : a_2a_1a_0 \qquad b : b_2b_1b_0$

If $(a > b)\ \{\ldots\}$

$$\phi = a \overset{?}{>} b = (a_2 \oplus b_2) \wedge a_2$$
$$\oplus\ (a_2 \oplus b_2 \oplus 1) \wedge (a_1 \oplus b_1) \wedge a_1$$
$$\oplus\ (a_2 \oplus b_2 \oplus 1) \wedge (a_1 \oplus b_1 \oplus 1) \wedge ((b_0 \oplus 1) \wedge a_0)$$

"If the most significant bits are not the same, then $a$ is greater than $b$ when $a_2$ is set"

*Cleartext*

*Oblivious*

# OBLIVIOUS COMPUTATION

Οι συμμέτοχοι εκτελούν πανομοιότυπους υπολογισμούς που είναι **ανεξάρτητοι από τα χαρακτηριστικά των δεδομένων**

- Τα μοτίβα προσπέλασης δεν εξαρτώνται από τα δεδομένα εισόδου
- Δεν εκτελούνται συνθήκες και διακλαδώσεις
- Το μέγεθος των ενδιάμεσων δεδομένων δεν μειώνεται

$R$

| Employee | Salary |
|---|---|
| Kim | 2000 |
| Jane | 1500 |
| Alex | 4500 |

**all values are secret-shared*

# OBLIVIOUS COMPUTATION

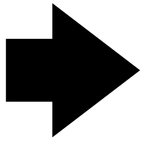
Οι συμμέτοχοι εκτελούν πανομοιότυπους υπολογισμούς που είναι **ανεξάρτητοι από τα χαρακτηριστικά των δεδομένων**

- Τα μοτίβα προσπέλασης δεν εξαρτώνται από τα δεδομένα εισόδου
- Δεν εκτελούνται συνθήκες και διακλαδώσεις
- Το μέγεθος των ενδιάμεσων δεδομένων δεν μειώνεται

$R$

| Employee | Salary |
|---|---|
| Kim | 2000 |
| Jane | 1500 |
| Alex | 4500 |

$\sigma(Salary > 3000)$

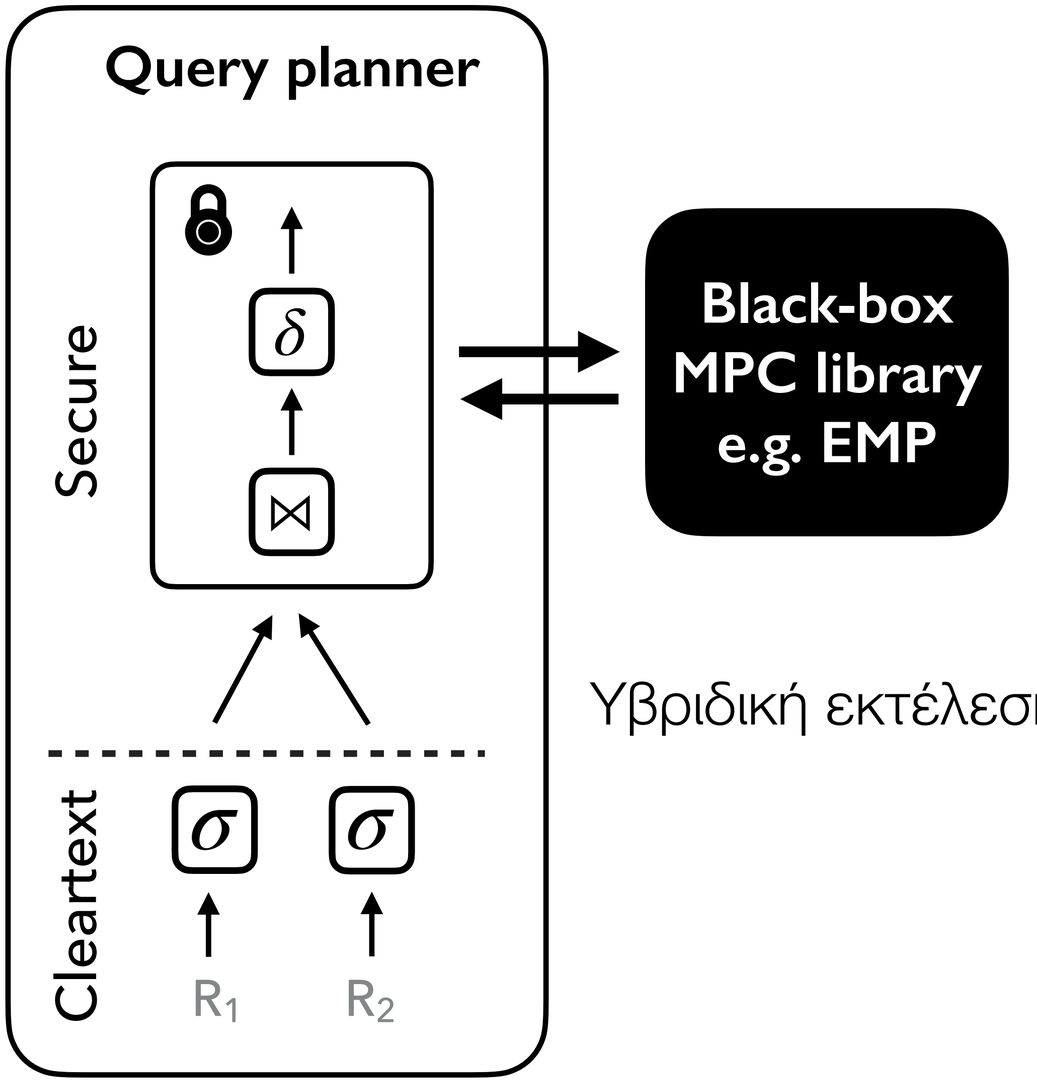


$R'$

| Employee | Salary | $\phi$ |
|---|---|---|
| Kim | 2000 | 0 |
| Jane | 1500 | 0 |
| Alex | 4500 | 1 |

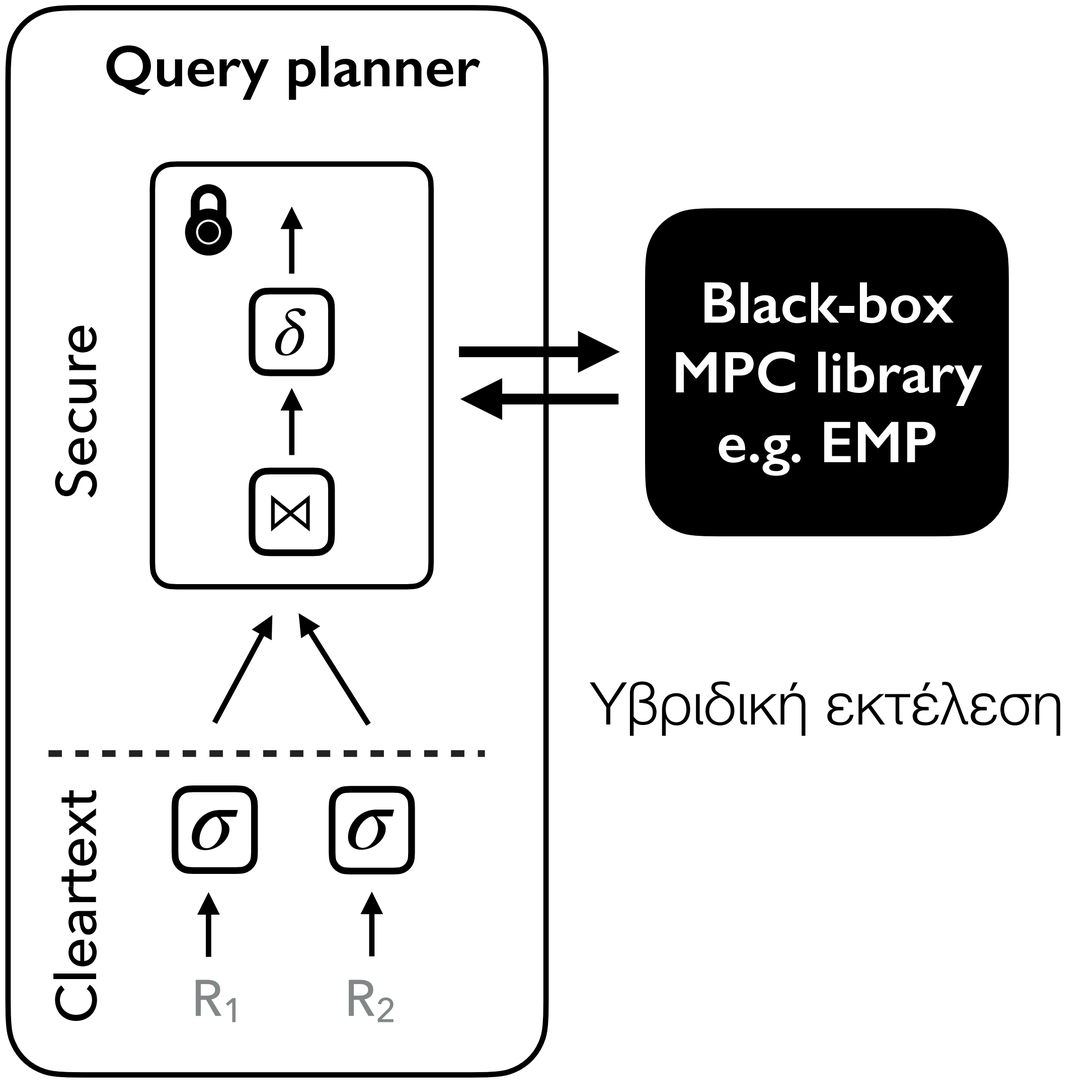
*all values are secret-shared

# Ανοίγοντας το “μαύρο κουτί” του MPC




**Black-box MPC library e.g. EMP**

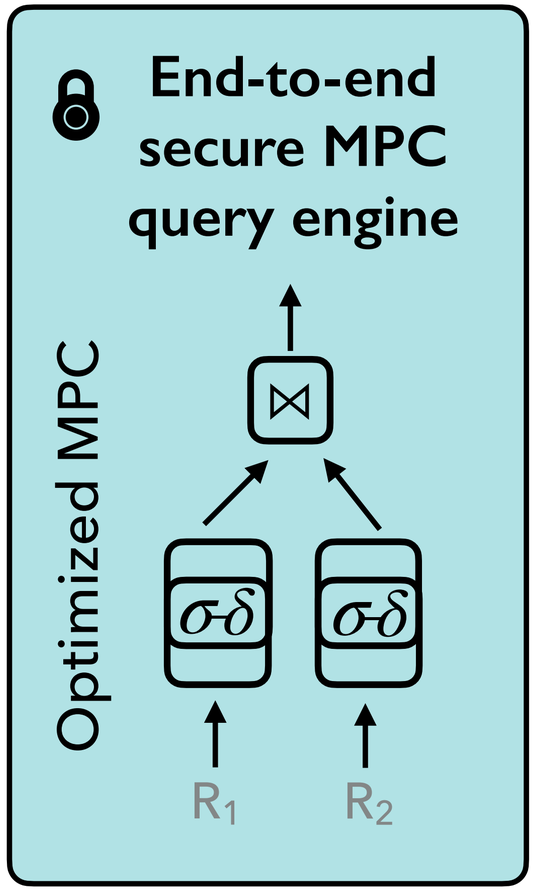
Υβριδική εκτέλεση







Υβριδική εκτέλεση



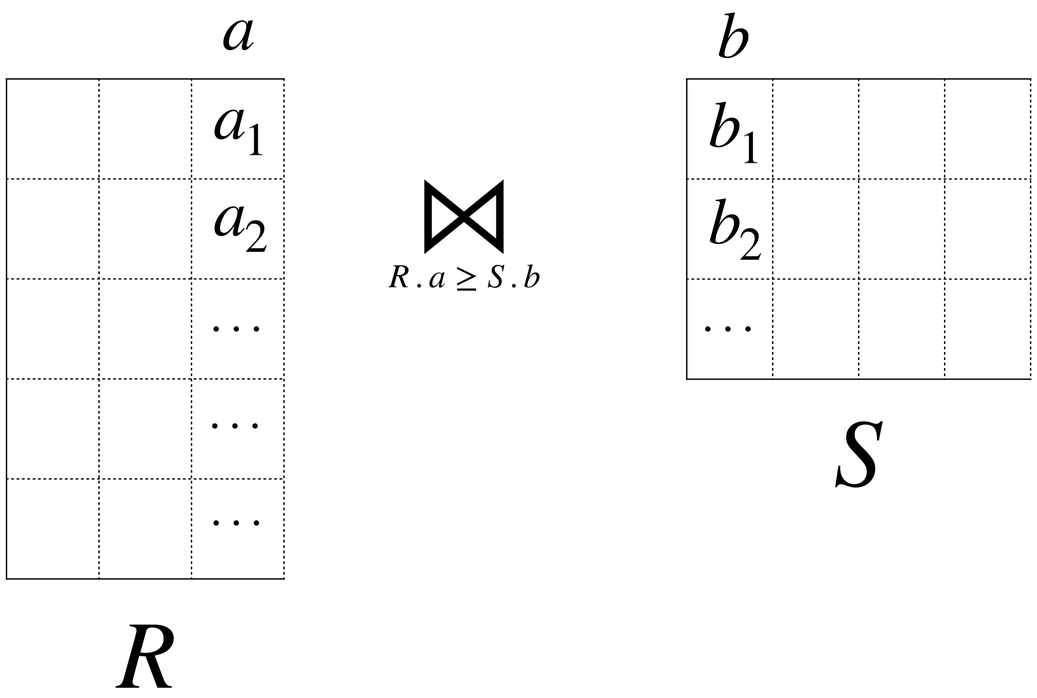

Secrecy

# ΤΕΧΝΙΚΕΣ ΣΥΝΕΙΣΦΟΡΕΣ ΤΟΥ SECRECY

1. ***Επανασχεδιασμός βασικών λειτουργιών MPC για αποδοτική χρήση σε σχεσιακά δεδομένα***

- Απόσβεση δικτυακού κόστους I/O
- Ανταγωνιστική απόδοση σε δίκτυα με υψηλή καθυστέρηση (high-latency WAN)

# ΠΑΡΑΔΕΙΓΜΑ: ΟΜΑΔΟΠΟΙΗΣΗ ΜΗΝΥΜΑΤΩΝ (batching)




$$[\theta_1 \leftarrow a_1 \overset{?}{\geq} b_1]$$

$$[\theta_2 \leftarrow a_1 \overset{?}{\geq} b_2]$$

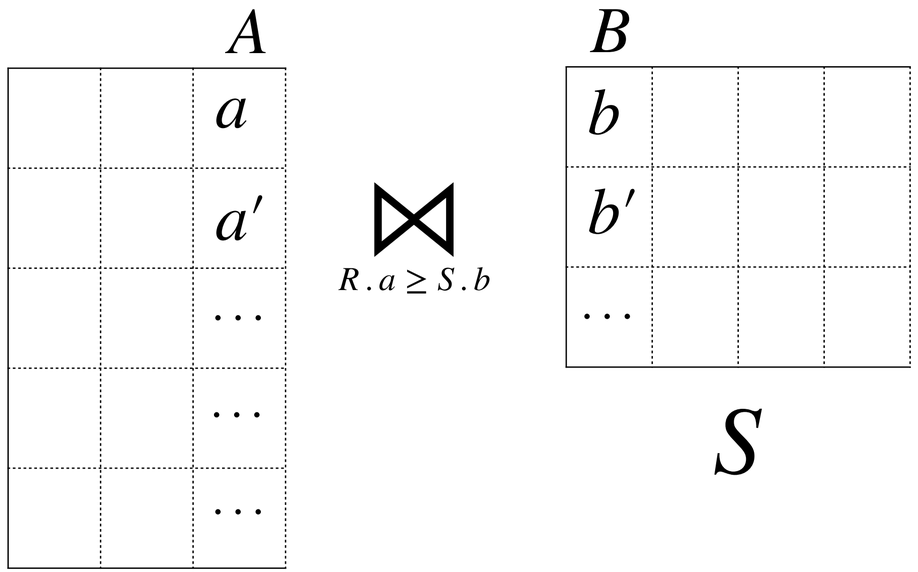
$$\cdots$$

$$[\theta_\kappa \leftarrow a_2 \overset{?}{\geq} b_1]$$

$$[\theta_{\kappa+1} \leftarrow a_2 \overset{?}{\geq} b_1]$$

$$\cdots$$

# ΠΑΡΑΔΕΙΓΜΑ: ΟΜΑΔΟΠΟΙΗΣΗ ΜΗΝΥΜΑΤΩΝ (batching)




$\bowtie_{R.a \geq S.b}$




$$a \overset{?}{\geq} b \iff (a_3 \oplus b_3) \wedge a_3 \oplus ((a_3 \oplus b_3) \oplus 1) \wedge (a_2 \oplus b_2) \wedge a_2 \oplus ((a_3 \oplus b_3) \oplus 1) \wedge ((a_2 \oplus b_2) \oplus 1) \wedge (a_1 \oplus b_1) \wedge a_1 \oplus ((a_3 \oplus b_3) \oplus 1) \wedge ((a_2 \oplus b_2) \oplus 1) \wedge ((a_1 \oplus b_1) \oplus 1) \wedge ((a_0 \oplus 1) \wedge b_0)$$

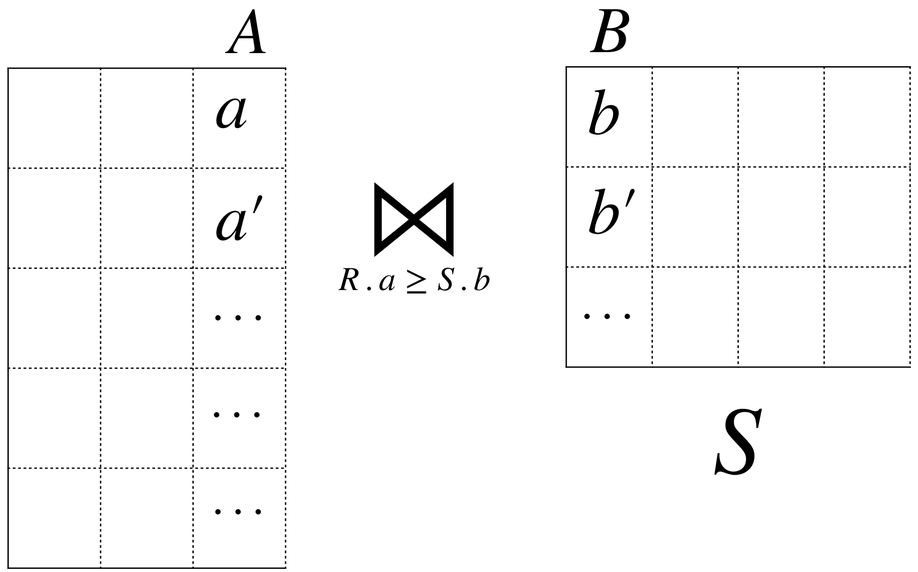
$$a \overset{?}{\geq} b' \iff (a_3 \oplus b'_3) \wedge a'_3 \oplus ((a_3 \oplus b'_3) \oplus 1) \wedge (a_2 \oplus b'_2) \wedge a_2 \oplus ((a_3 \oplus b'_3) \oplus 1) \wedge ((a_2 \oplus b'_2) \oplus 1) \wedge (a_1 \oplus b'_1) \wedge a_1 \oplus ((a_3 \oplus b'_3) \oplus 1) \wedge ((a_2 \oplus b'_2) \oplus 1) \wedge ((a_1 \oplus b'_1) \oplus 1) \wedge ((a_0 \oplus 1) \wedge b'_0)$$

$$a' \overset{?}{\geq} b \iff \cdots$$

$\ell = 4$ *bits*

$$\underbrace{a_3a_2a_1a_0}_{a} \quad \underbrace{b_3b_2b_1b_0}_{b} \quad \underbrace{a'_3a'_2a'_1a'_0}_{a'} \quad \underbrace{b'_3b'_2b'_1b'_0}_{b'}$$

# ΠΑΡΑΔΕΙΓΜΑ: ΟΜΑΔΟΠΟΙΗΣΗ ΜΗΝΥΜΑΤΩΝ (batching)




$\bowtie_{R.a \geq S.b}$




$\ell = 4$ *bits*

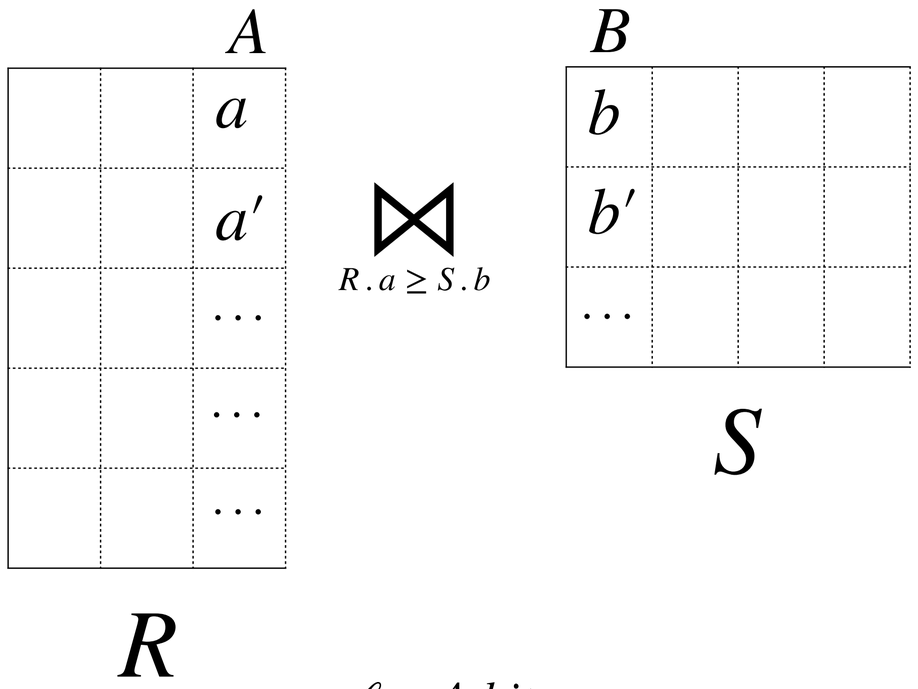
$\underbrace{a_3a_2a_1a_0}_{a}$ $\underbrace{b_3b_2b_1b_0}_{b}$ $\underbrace{a'_3a'_2a'_1a'_0}_{a'}$ $\underbrace{b'_3b'_2b'_1b'_0}_{b'}$

$$a \overset{?}{\geq} b \iff (a_3 \oplus b_3) \wedge a_3 \oplus ((a_3 \oplus b_3) \oplus 1) \wedge (a_2 \oplus b_2) \wedge a_2 \oplus ((a_3 \oplus b_3) \oplus 1) \wedge ((a_2 \oplus b_2) \oplus 1) \wedge (a_1 \oplus b_1) \wedge a_1 \oplus ((a_3 \oplus b_3) \oplus 1) \wedge ((a_2 \oplus b_2) \oplus 1) \wedge ((a_1 \oplus b_1) \oplus 1) \wedge ((a_0 \oplus 1) \wedge b_0)$$

$$a \overset{?}{\geq} b' \iff (a_3 \oplus b'_3) \wedge a'_3 \oplus ((a_3 \oplus b'_3) \oplus 1) \wedge (a_2 \oplus b'_2) \wedge a_2 \oplus ((a_3 \oplus b'_3) \oplus 1) \wedge ((a_2 \oplus b'_2) \oplus 1) \wedge (a_1 \oplus b'_1) \wedge a_1 \oplus ((a_3 \oplus b'_3) \oplus 1) \wedge ((a_2 \oplus b'_2) \oplus 1) \wedge ((a_1 \oplus b'_1) \oplus 1) \wedge ((a_0 \oplus 1) \wedge b'_0)$$

$$a' \overset{?}{\geq} b \iff \cdots$$

# ΠΑΡΑΔΕΙΓΜΑ: ΟΜΑΔΟΠΟΙΗΣΗ ΜΗΝΥΜΑΤΩΝ (batching)




$\ell = 4$ *bits*

$\underbrace{a_3a_2a_1a_0}_{a}$ $\underbrace{b_3b_2b_1b_0}_{b}$ $\underbrace{a'_3a'_2a'_1a'_0}_{a'}$ $\underbrace{b'_3b'_2b'_1b'_0}_{b'}$

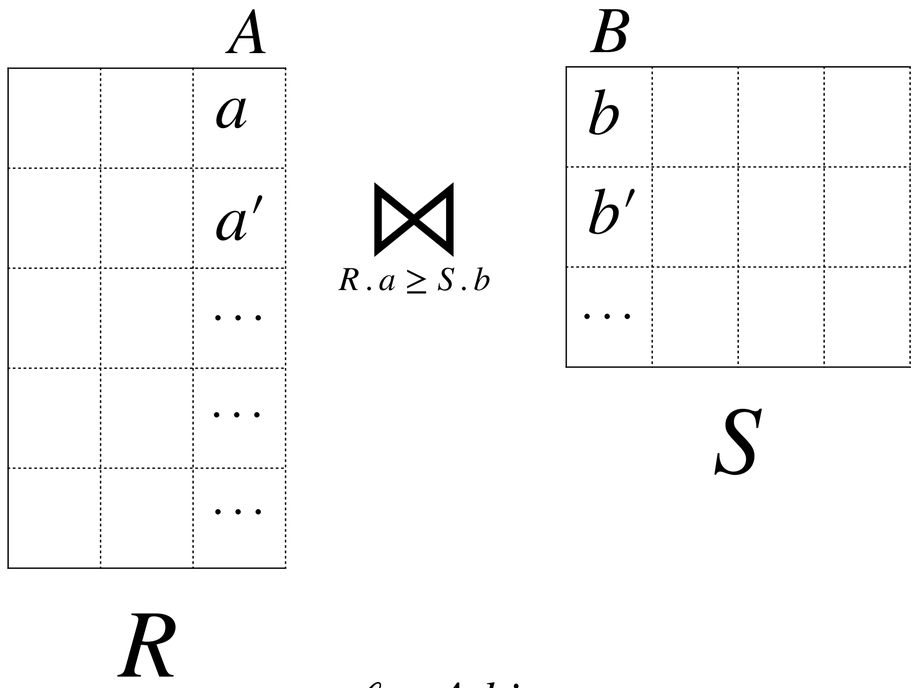
$$a \overset{?}{\geq} b \iff c_1 \wedge a_3 \oplus c_2 \wedge c_3 \wedge a_2 \oplus c_4 \wedge c_5 \wedge c_6 \wedge a_1 \oplus c_4 \wedge c_5 \wedge c_7 \wedge (c_8 \wedge b_0)$$

$$a \overset{?}{\geq} b' \iff c'_1 \wedge a_3 \oplus c'_2 \wedge c'_3 \wedge a_2 \oplus c'_4 \wedge c'_5 \wedge c'_6 \wedge a_1 \oplus c'_4 \wedge c'_5 \wedge c'_7 \wedge (c_8 \wedge b'_0)$$

$$a' \overset{?}{\geq} b \iff \cdots$$

1 γύρος επικοινωνίας

# ΠΑΡΑΔΕΙΓΜΑ: ΟΜΑΔΟΠΟΙΗΣΗ ΜΗΝΥΜΑΤΩΝ (batching)




$$a \overset{?}{\geq} b \iff c_1 \wedge a_3 \oplus c_2 \wedge c_3 \wedge a_2 \oplus c_4 \wedge c_5 \wedge c_6 \wedge a_1 \oplus c_4 \wedge c_5 \wedge c_7 \wedge (c_8 \wedge b_0)$$

$$a \overset{?}{\geq} b' \iff c'_1 \wedge a_3 \oplus c'_2 \wedge c'_3 \wedge a_2 \oplus c'_4 \wedge c'_5 \wedge c'_6 \wedge a_1 \oplus c'_4 \wedge c'_5 \wedge c'_7 \wedge (c_8 \wedge b'_0)$$

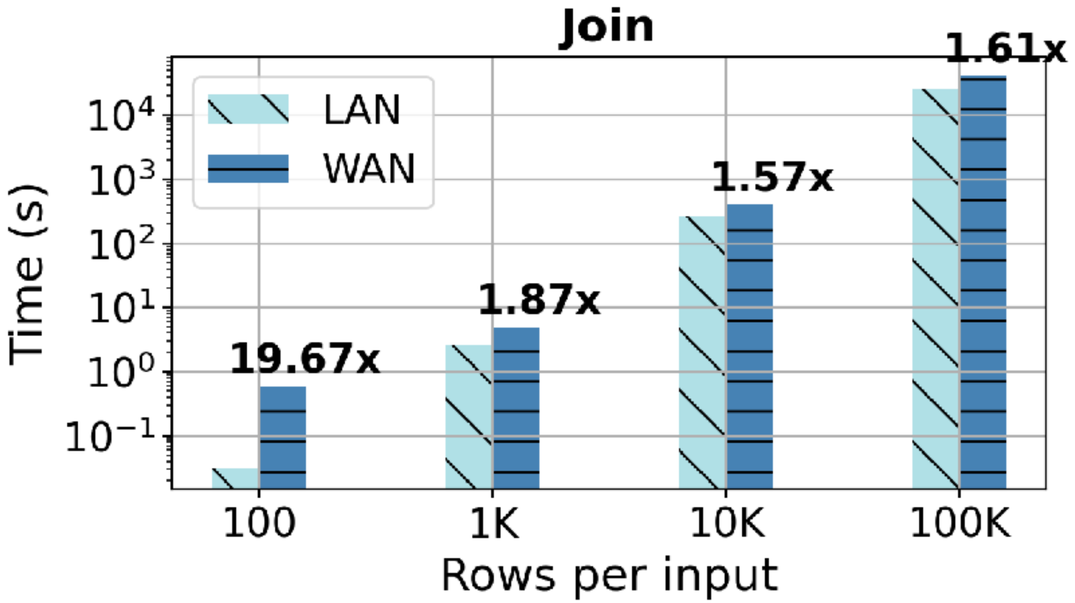
$$a' \overset{?}{\geq} b \iff \cdots$$

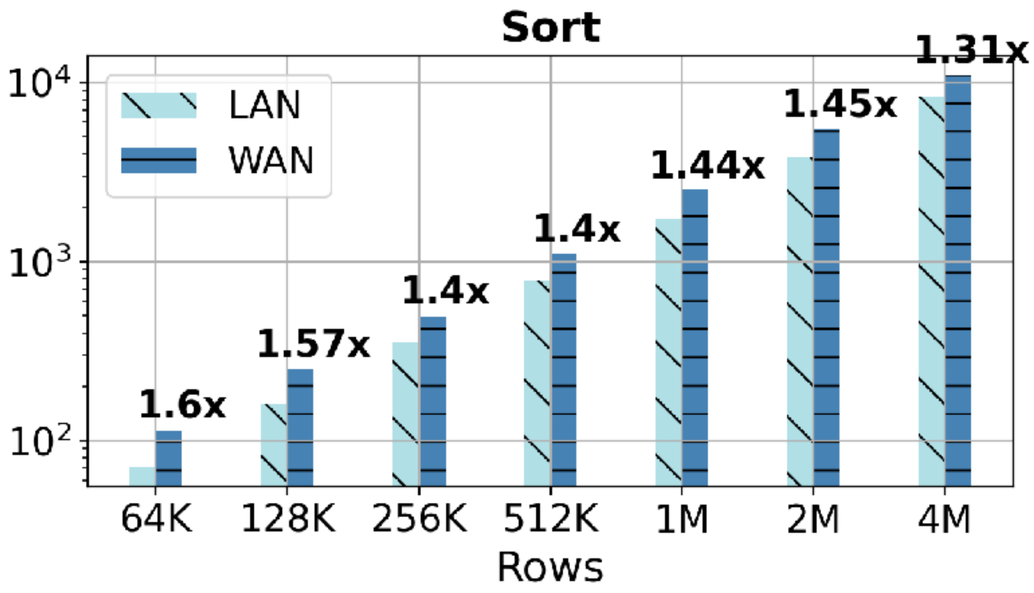
$\ell = 4$ *bits*

$$\underbrace{a_3a_2a_1a_0}_{a} \quad \underbrace{b_3b_2b_1b_0}_{b} \quad \underbrace{a'_3a'_2a'_1a'_0}_{a'} \quad \underbrace{b'_3b'_2b'_1b'_0}_{b'}$$

$\log \ell + 1$ γύροι επικοινωνίας *συνολικά* για του υπολογισμό του join

# ΕΠΙΔΡΑΣΗ ΤΗΣ ΟΜΑΔΟΠΟΙΗΣΗΣ ΜΗΝΥΜΑΤΩΝ ΣΤΗΝ ΑΠΟΔΟΣΗ




**Απόσβεση του κόστους επικοινωνίας του MPC**
**Η τεχνική secret-sharing καθίσταται πρακτική και σε WAN**

* LAN: Parties deployed on three VMs in the same AWS region

* WAN: Parties deployed in three AWS regions: us-east-2 (Ohio), us-east-1 (Virginia), and us-west-1 (California)

# ΤΕΧΝΙΚΕΣ ΣΥΝΕΙΣΦΟΡΕΣ ΤΟΥ SECRECY

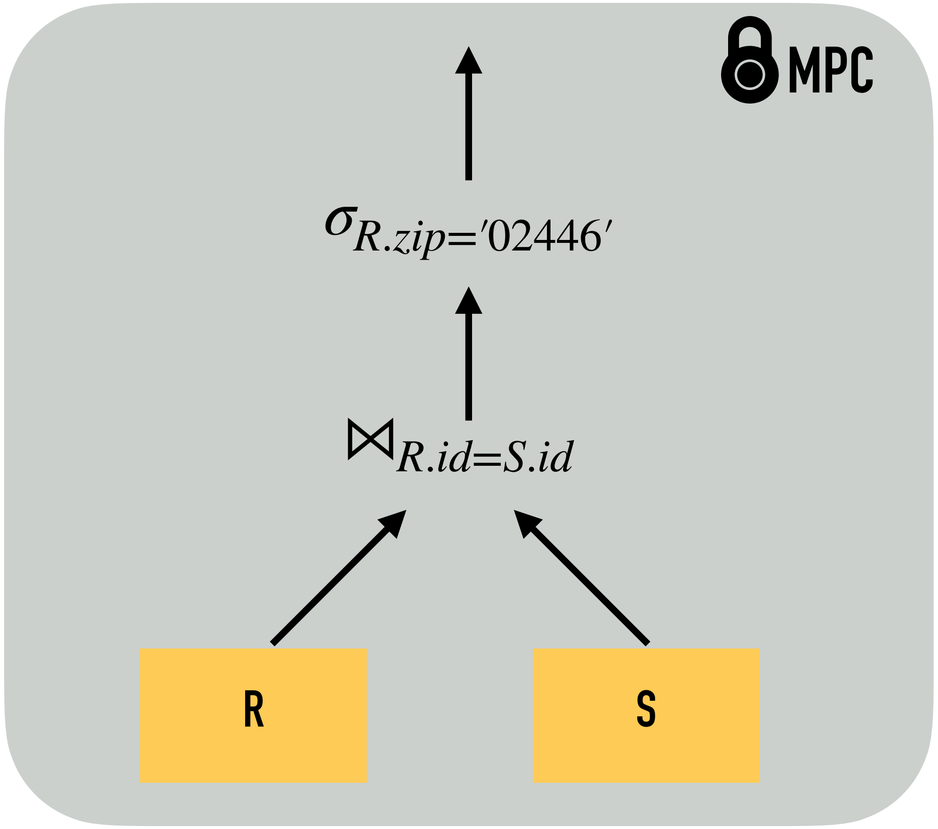
1. ***Επανασχεδιασμός βασικών λειτουργιών MPC για αποδοτική χρήση σε σχεσιακά δεδομένα***
   - Απόσβεση δικτυακού κόστους I/O
   - Ανταγωνιστική απόδοση σε δίκτυα με υψηλή καθυστέρηση (high-latency WAN)

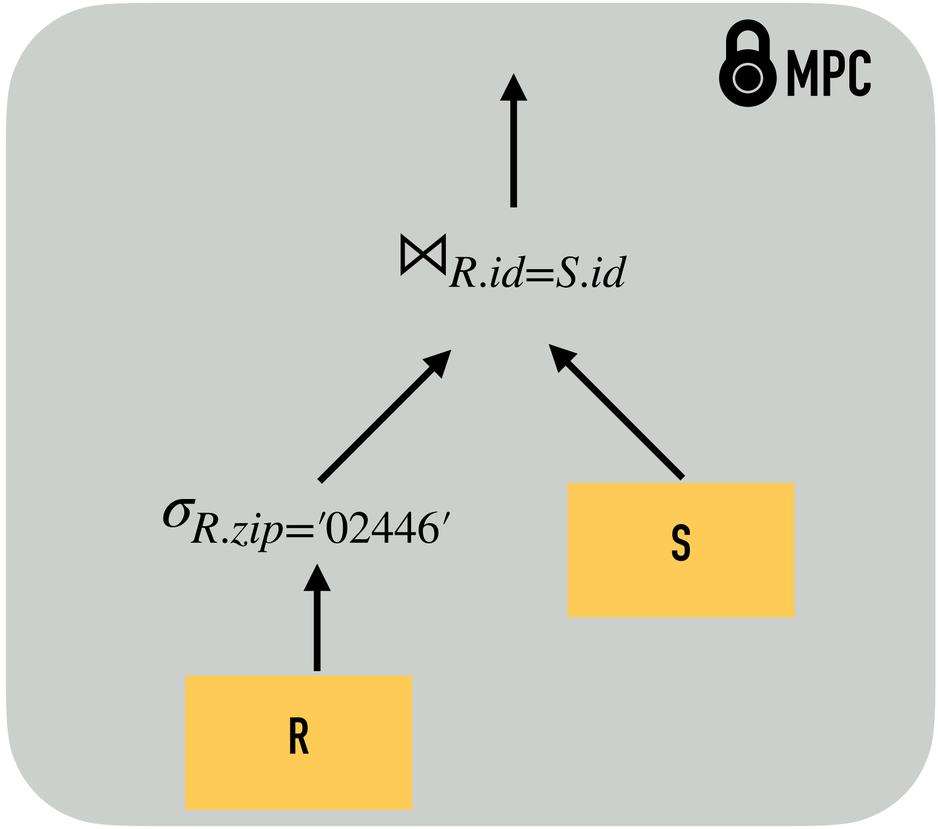
# ΤΕΧΝΙΚΕΣ ΣΥΝΕΙΣΦΟΡΕΣ ΤΟΥ SECRECY

1. ***Επανασχεδιασμός βασικών λειτουργιών MPC για αποδοτική χρήση σε σχεσιακά δεδομένα***
   - Απόσβεση δικτυακού κόστους I/O
   - Ανταγωνιστική απόδοση σε δίκτυα με υψηλή καθυστέρηση (high-latency WAN)

2. ***Ανάλυση του κόστους σχεσιακών πλάνων ως συνάρτηση υπολογισμών και επικοινωνίας των βασικών λειτουργιών MPC***
   - Operation cost (πλήθος βασικών λειτουργιών MPC)
   - Synchronization cost (πλήθος γύρων)
   - Composition cost (επιπλέον κόστος σύνθεσης τελεστών υπό MPC)

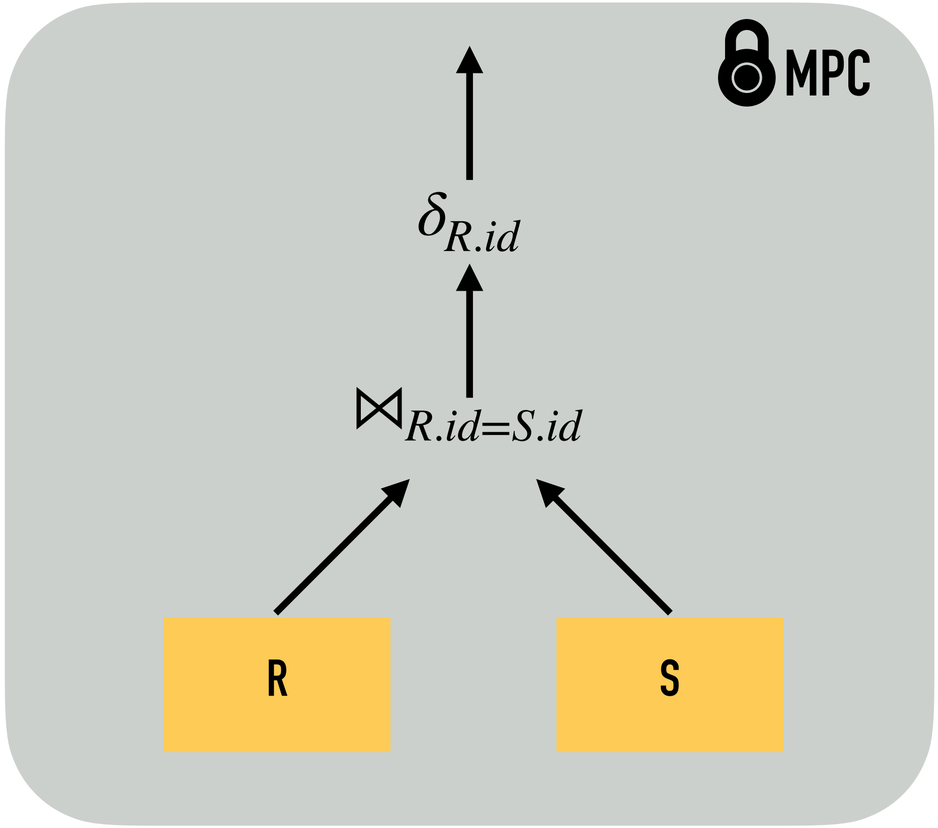
# ΟΙ ΚΛΑΣΙΚΕΣ ΜΕΘΟΔΟΙ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ ΔΕΝ ΕΙΝΑΙ ΠΑΝΤΑ ΑΠΟΤΕΛΕΣΜΑΤΙΚΕΣ ΣΤΟ MPC



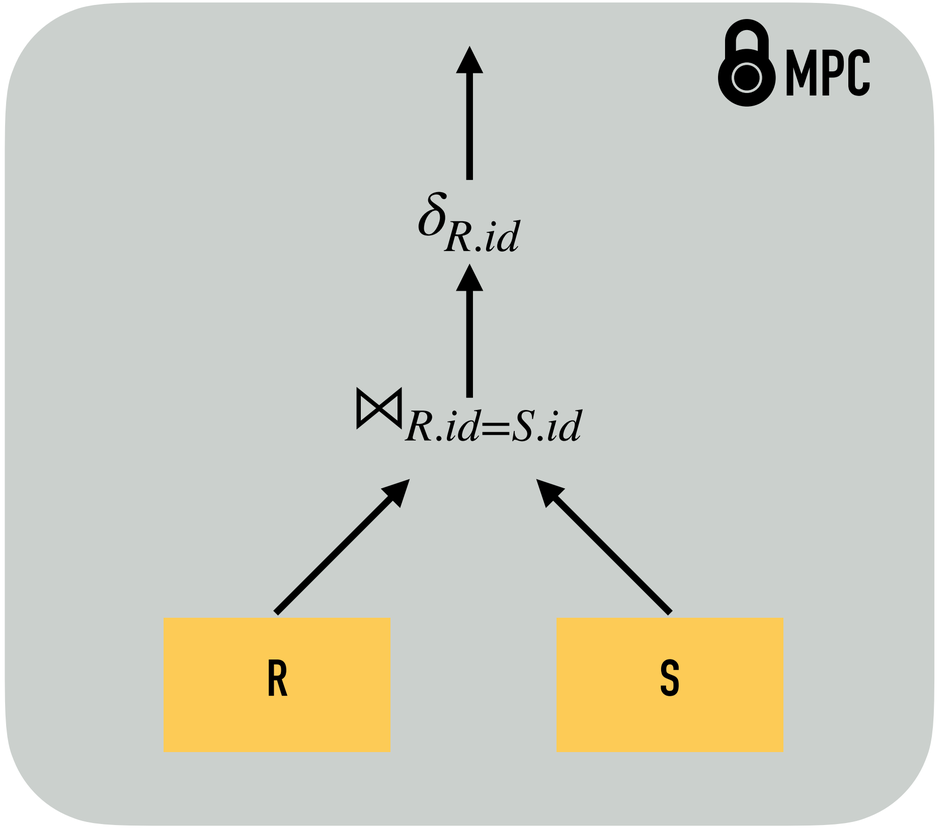

*Η αναδιάταξη του πλάνου (selection pushdown) δε βελτιώνει την απόδοση του* `JOIN` *υπό MPC, αφού το φίλτρο δε μειώνει το πλήθος των ενδιάμεσων εγγραφών της σχέσης R*

# Η ΑΝΑΔΙΑΤΑΞΗ ΤΕΛΕΣΤΩΝ ΥΠΟ MPC ΜΠΟΡΕΙ ΝΑ ΜΕΙΩΣΕΙ ΤΟ ΚΟΣΤΟΣ




```
SELECT DISTINCT R.id
FROM R, S
WHERE R.id = S.id
```

# Η ΑΝΑΔΙΑΤΑΞΗ ΤΕΛΕΣΤΩΝ ΥΠΟ MPC ΜΠΟΡΕΙ ΝΑ ΜΕΙΩΣΕΙ ΤΟ ΚΟΣΤΟΣ




```
SELECT DISTINCT R.id
FROM R, S
WHERE R.id = S.id
```

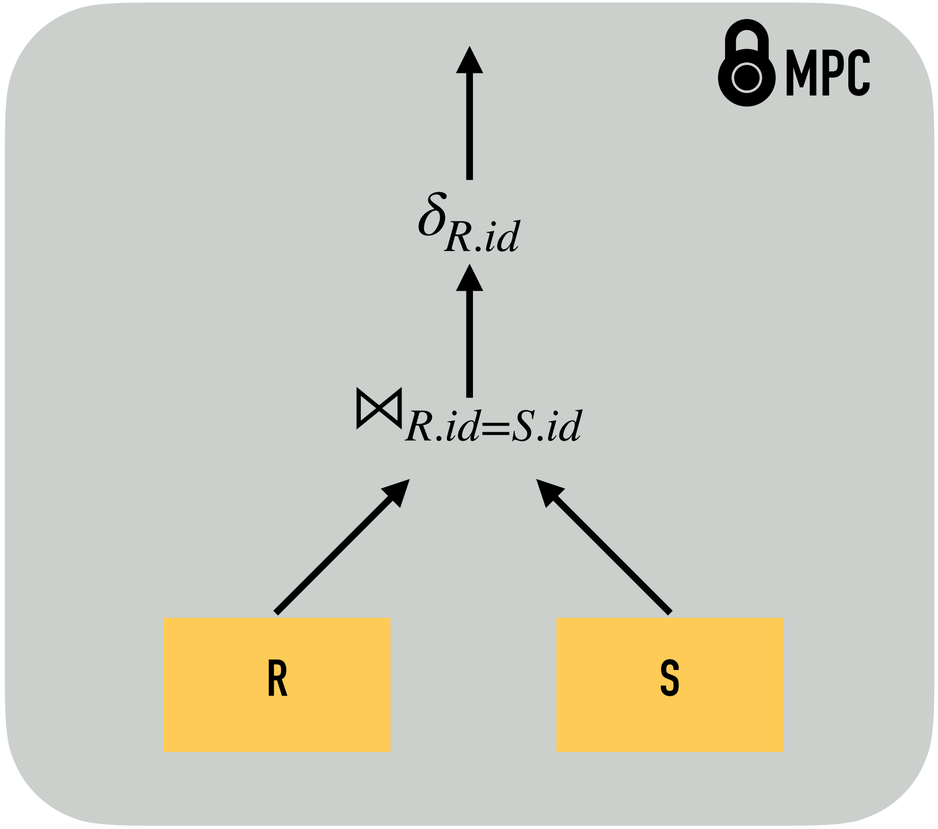
$|R| = |S| = n$

$O(n^2 \log^2 n)$ operations / messages

$O(\log^2 n)$ rounds $O(n^2)$ space

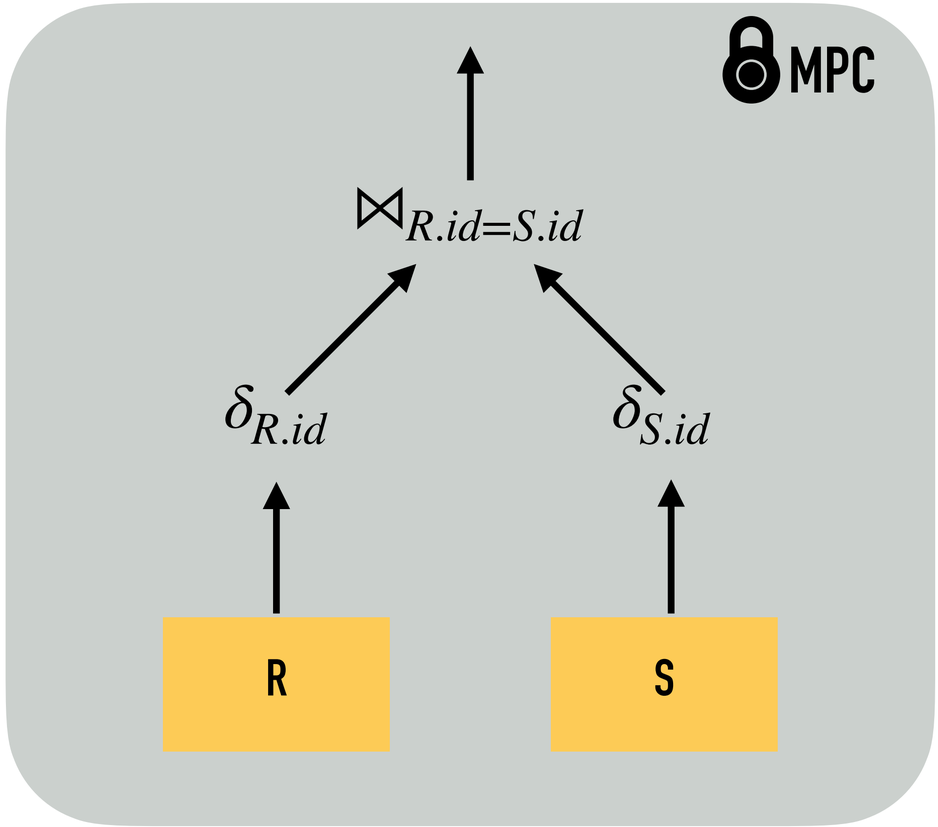
* Assuming the distinct operator is based on a sorting network

# Η ΑΝΑΔΙΑΤΑΞΗ ΤΕΛΕΣΤΩΝ ΥΠΟ MPC ΜΠΟΡΕΙ ΝΑ ΜΕΙΩΣΕΙ ΤΟ ΚΟΣΤΟΣ




$O(n^2 \log^2 n)$ operations / messages

$O(\log^2 n)$ rounds $O(n^2)$ space

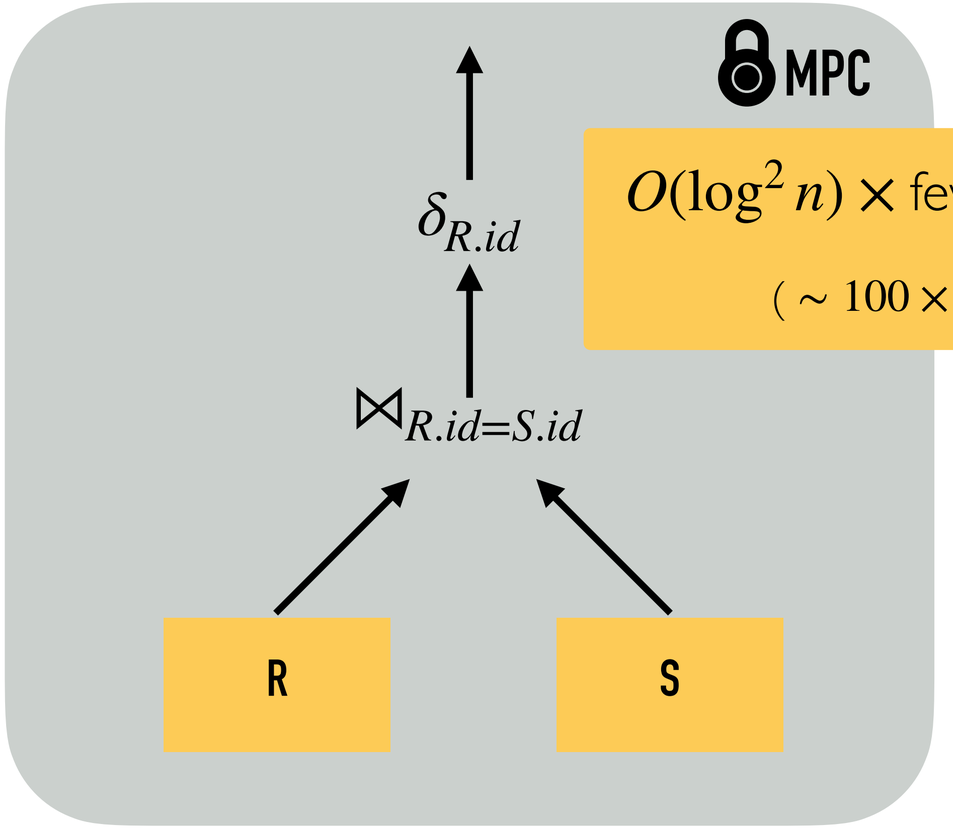



$O(n^2)$ operations / messages

$\sim 4\times$ fewer rounds $O(n)$ space

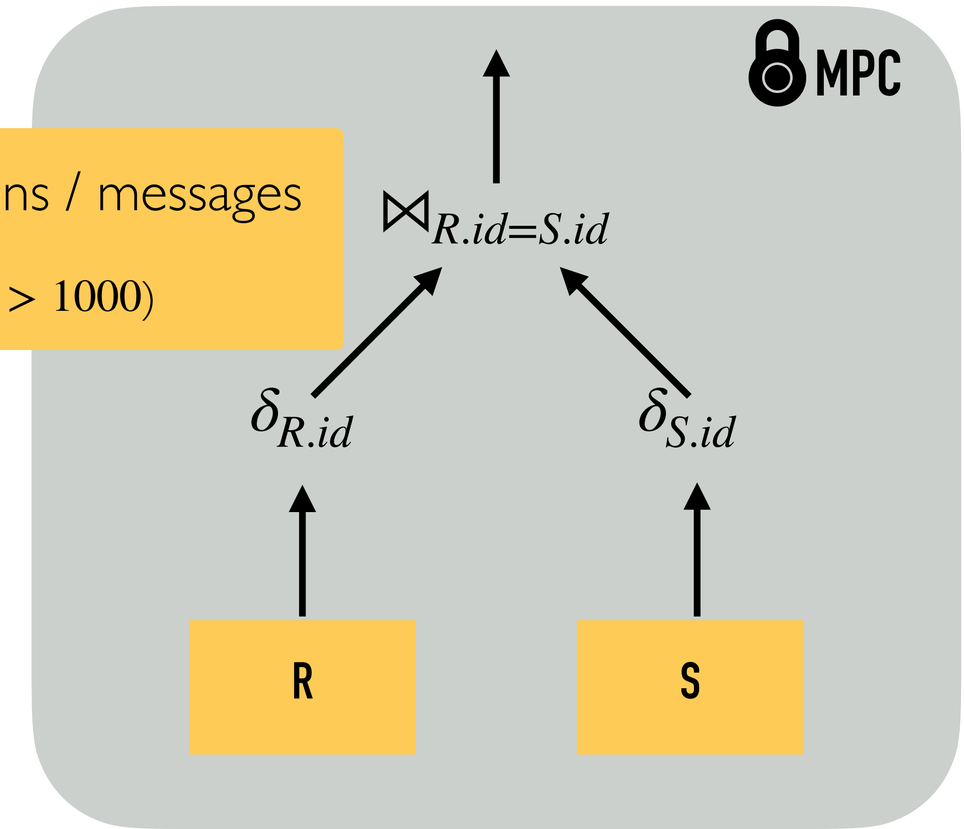
* Assuming the distinct operator is based on a sorting network

# Η ΑΝΑΔΙΑΤΑΞΗ ΤΕΛΕΣΤΩΝ ΥΠΟ MPC ΜΠΟΡΕΙ ΝΑ ΜΕΙΩΣΕΙ ΤΟ ΚΟΣΤΟΣ




$O(\log^2 n) \times$ fewer operations / messages

( $\sim 100 \times$ βελτίωση για $n > 1000$)




$O(n^2 \log^2 n)$ operations / messages

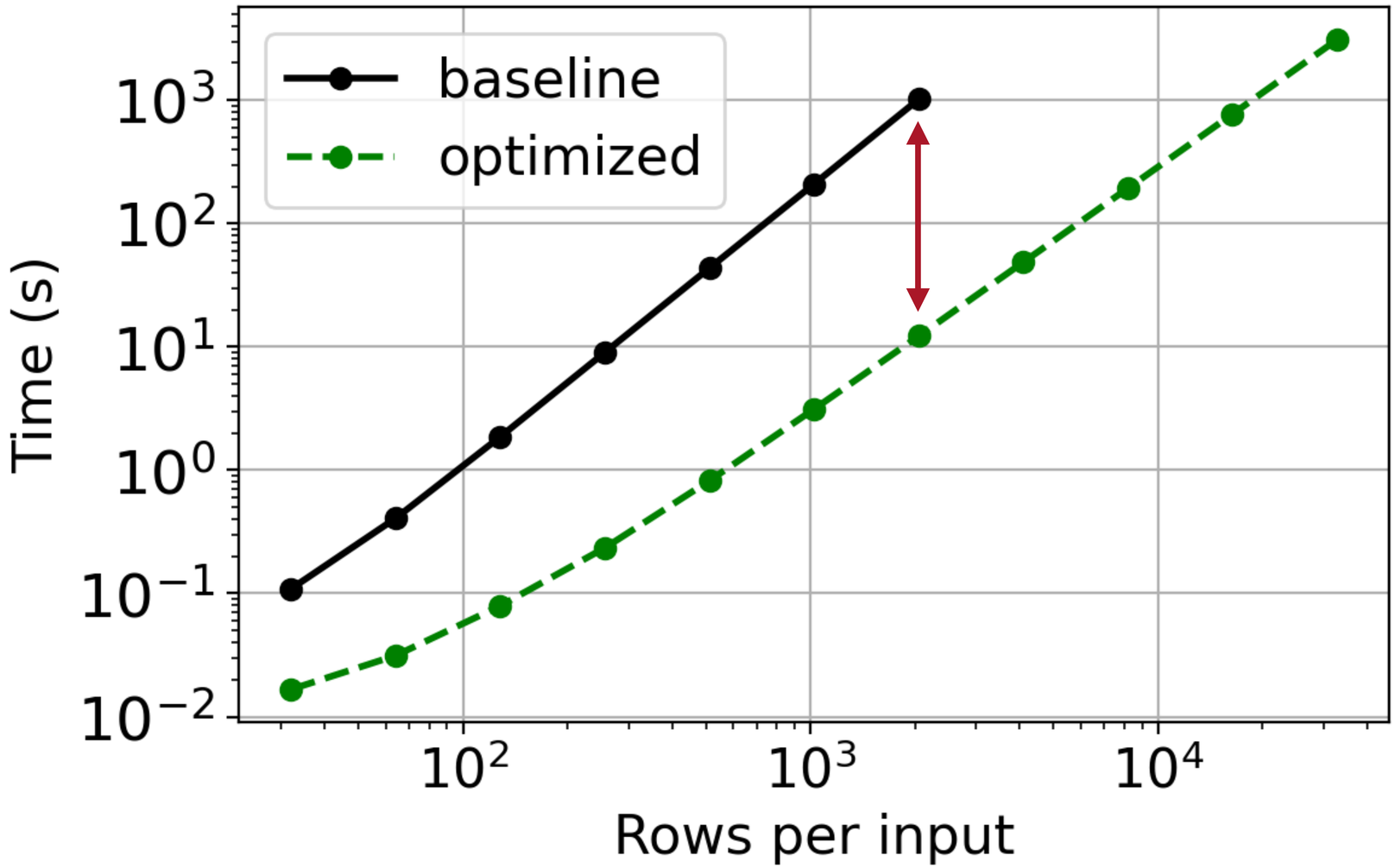
$O(\log^2 n)$ rounds $O(n^2)$ space

$O(n^2)$ operations / messages

$\sim 4 \times$ fewer rounds $O(n)$ space

* Assuming the distinct operator is based on a sorting network

# ΕΠΙΔΡΑΣΗ ΤΗΣ ΤΕΧΝΙΚΗΣ "DISTINCT PUSH-DOWN" (LAN)

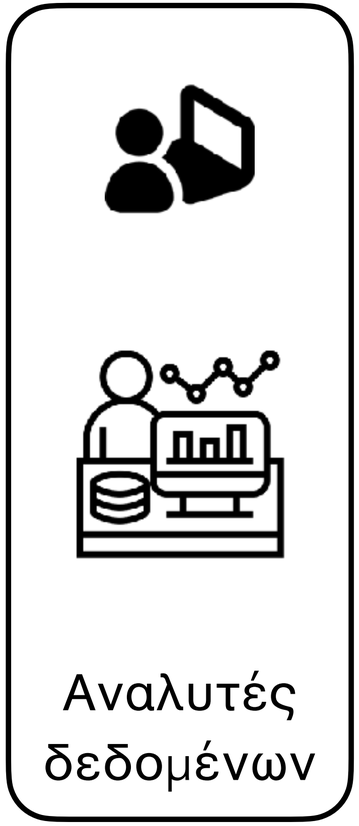



* Parties deployed on AWS EC2 r5.xlarge instances (us-east-2)

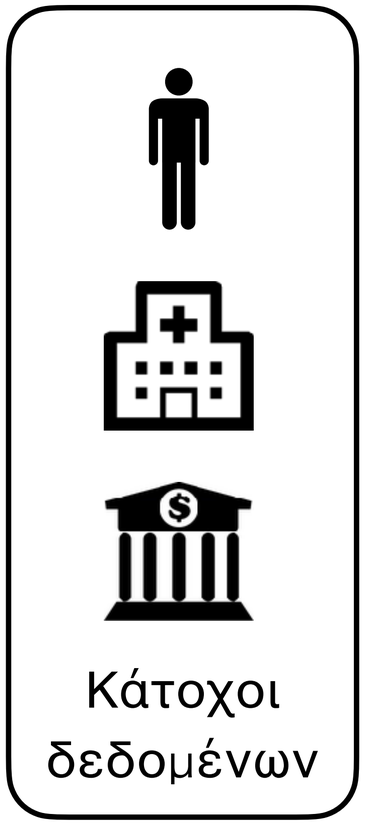
# ΤΕΧΝΙΚΕΣ ΣΥΝΕΙΣΦΟΡΕΣ ΤΟΥ SECRECY

1. ***Επανασχεδιασμός βασικών λειτουργιών MPC για αποδοτική χρήση σε σχεσιακά δεδομένα***
   - Απόσβεση δικτυακού κόστους I/O
   - Ανταγωνιστική απόδοση σε δίκτυα με υψηλή καθυστέρηση (high-latency WAN)
2. ***Ανάλυση του κόστους σχεσιακών πλάνων ως συνάρτηση υπολογισμών και επικοινωνίας των βασικών λειτουργιών MPC***
   - Operation cost (πλήθος βασικών λειτουργιών MPC)
   - Synchronization cost (πλήθος γύρων)
   - Composition cost (επιπλέον κόστος σύνθεσης τελεστών υπό MPC)
3. ***Διανυσματικός εκτελεστής ερωτημάτων MPC "Volcano-style"***
   - Λογικές βελτιστοποιήσεις (π.χ. αναδιάταξη τελεστών, αποσύνθεση)
   - Φυσικές βελτιστοποιήσεις (π.χ. σύντηξη τελεστών)
   - Βελτιστοποιήσεις για συγκεκριμένα πρωτόκολλα (π.χ. dual sharing)

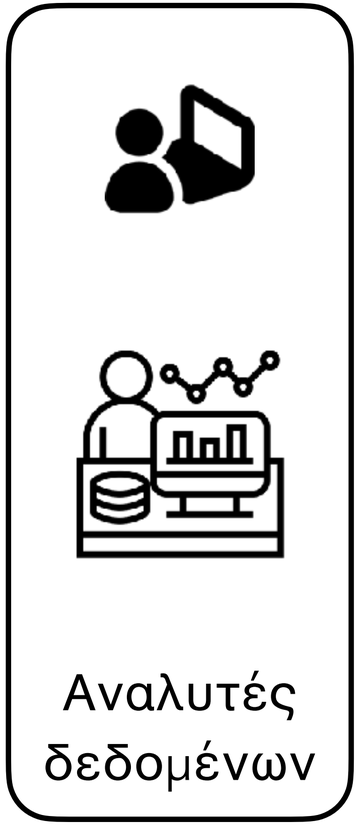
# SECRECY AS A SERVICE

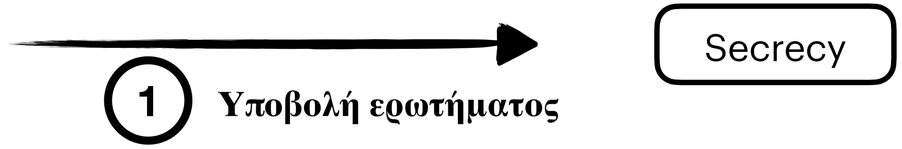



Secrecy

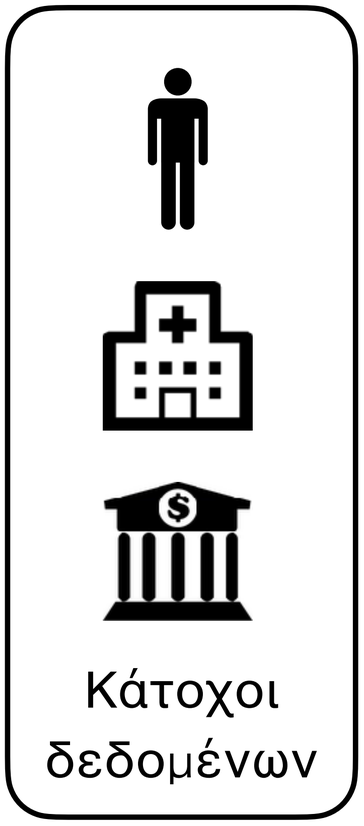


# SECRECY AS A SERVICE

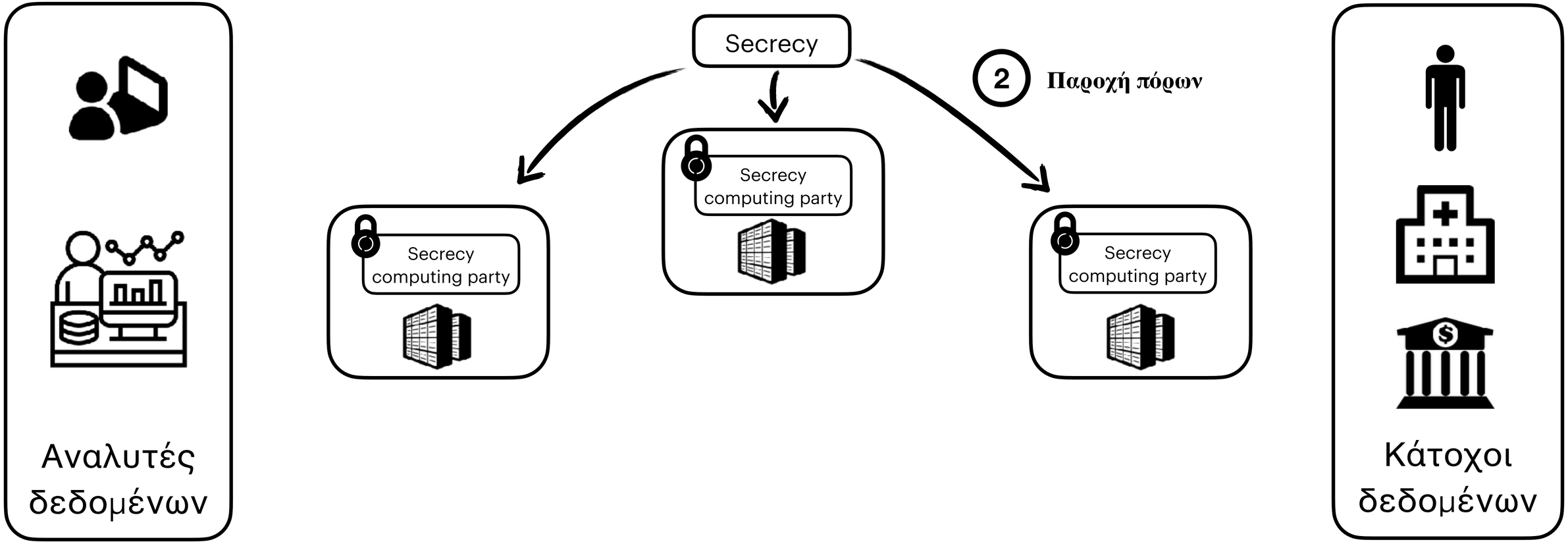



Secrecy

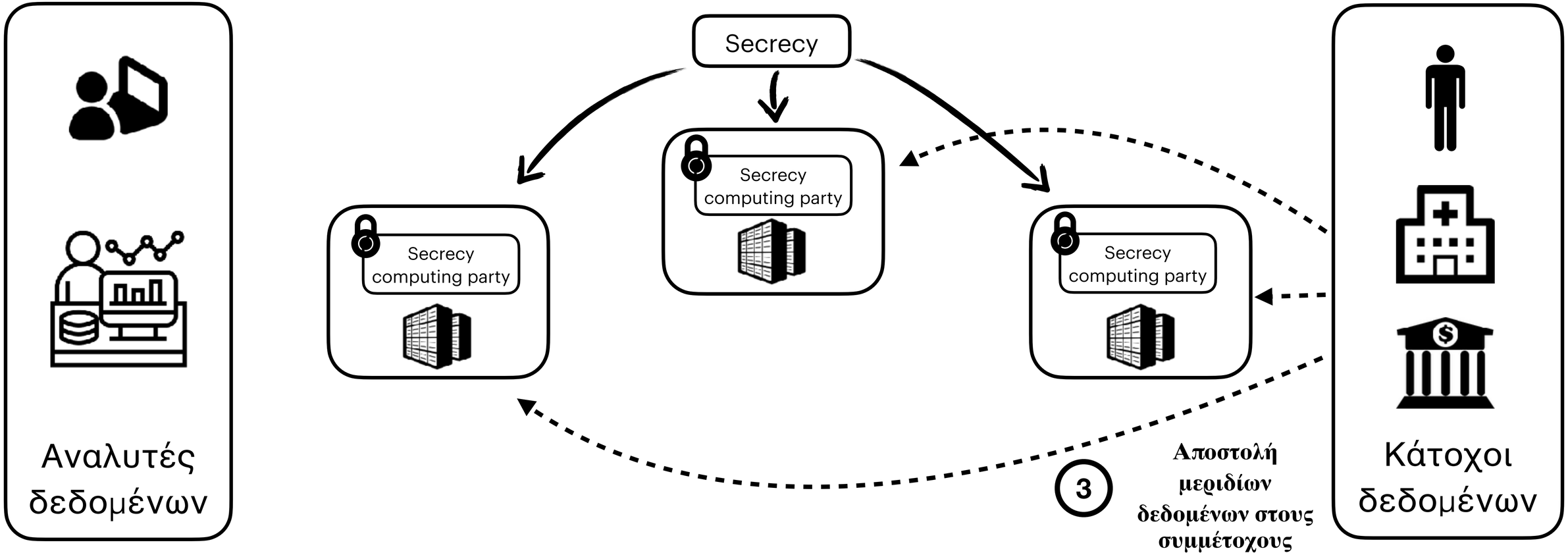


# SECRECY AS A SERVICE

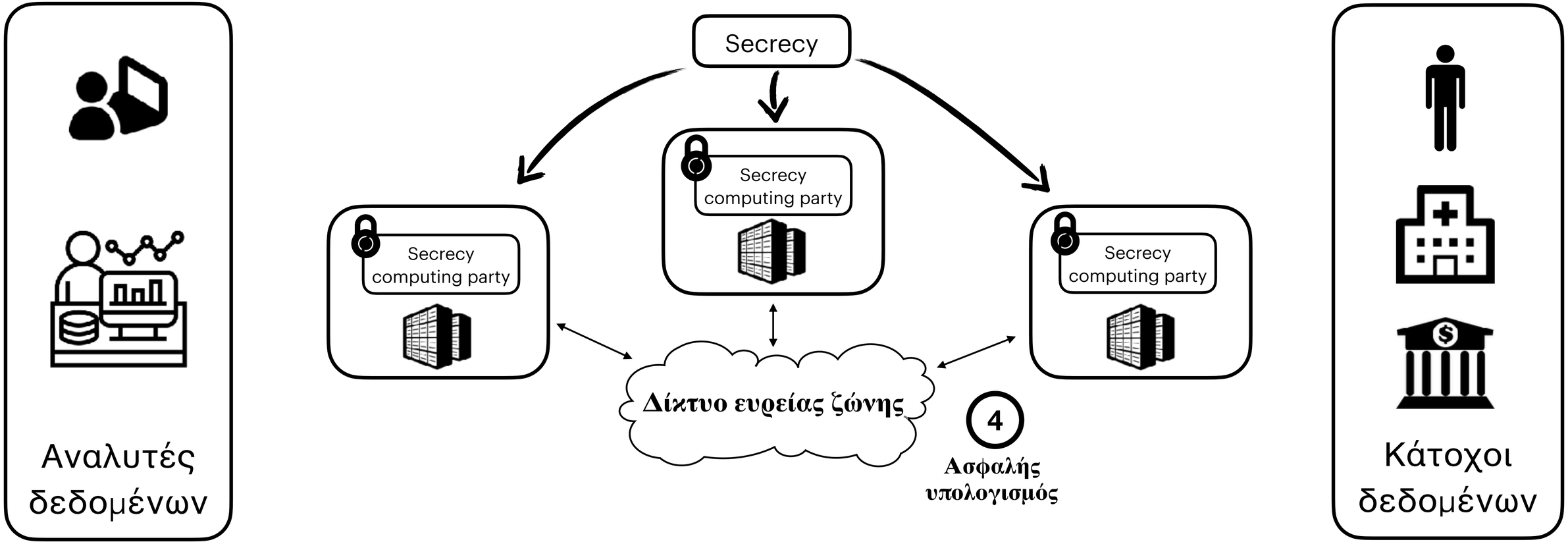



**Supported Cloud providers:**

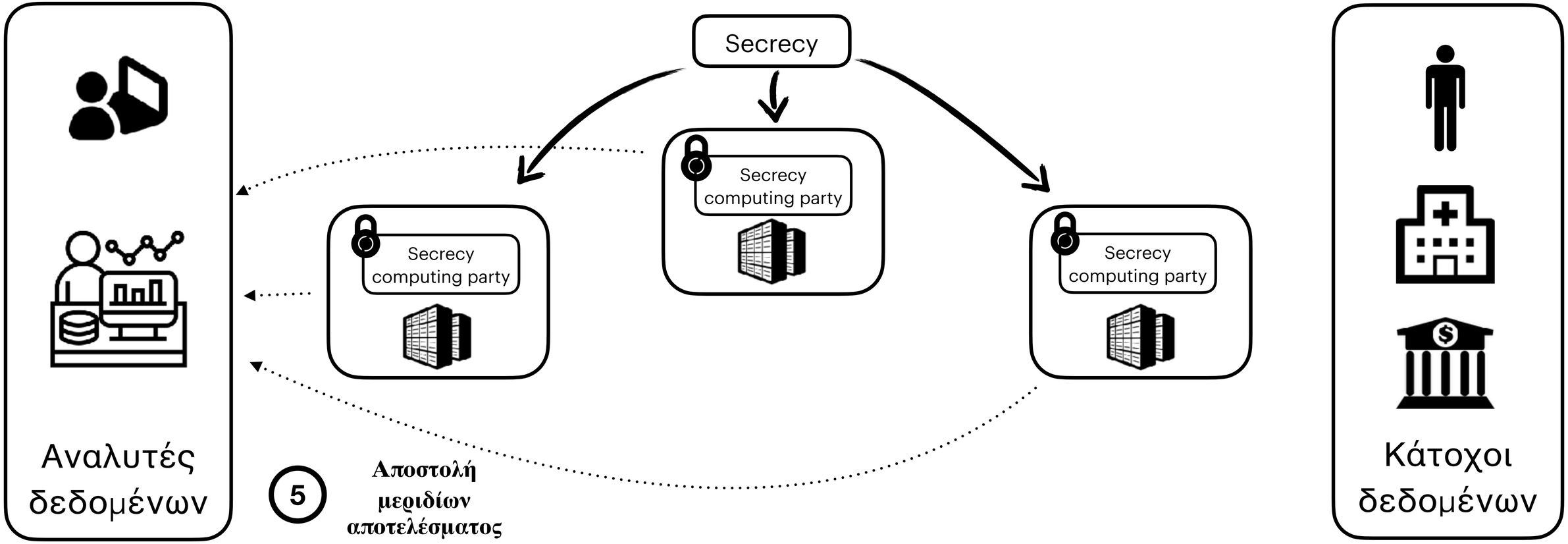
# SECRECY AS A SERVICE



# SECRECY AS A SERVICE

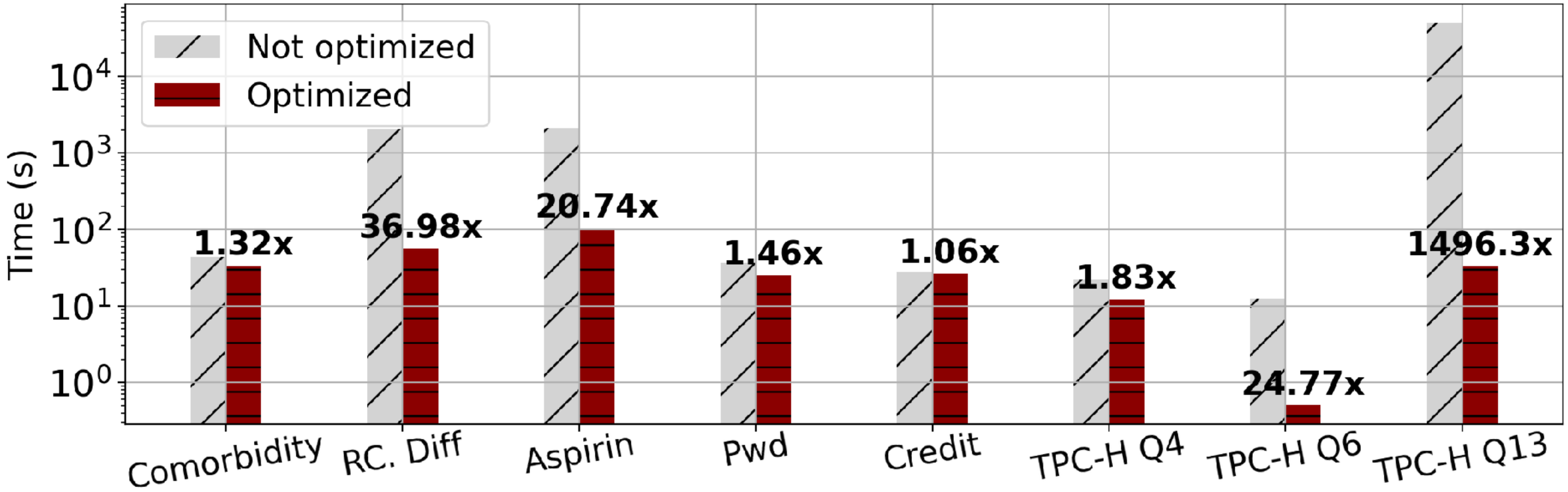


# SECRECY AS A SERVICE



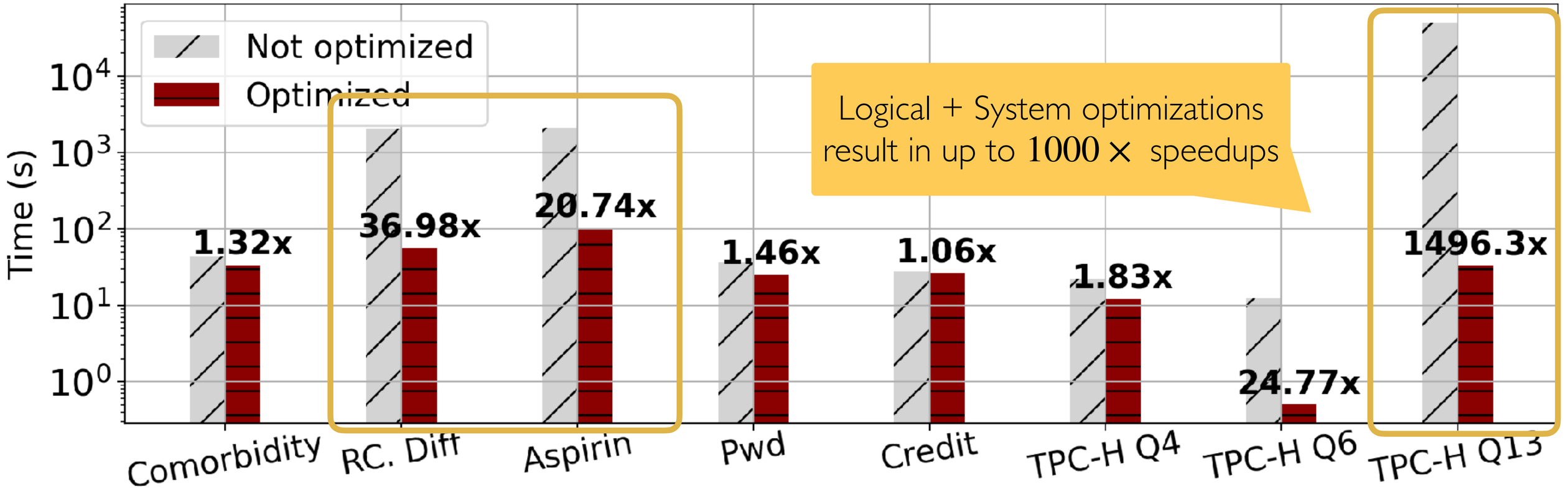
# Πειραματικά αποτελέσματα

# ΑΠΟΤΕΛΕΣΜΑΤΑ ΤΩΝ ΒΕΛΤΙΣΤΟΠΟΙΗΣΕΩΝ ΣΕ ΠΡΑΓΜΑΤΙΚΑ ΕΡΩΤΗΜΑΤΑ (WAN)



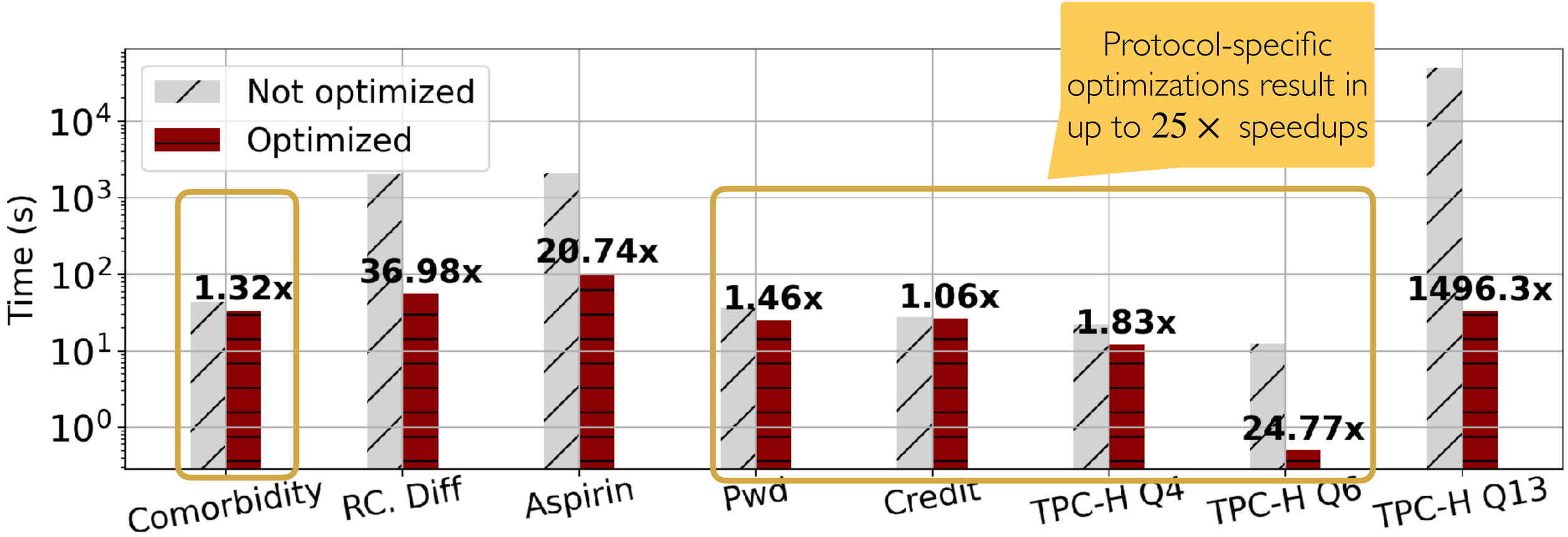

* Parties deployed in three AWS regions: us-east-2 (Ohio), us-east-1 (Virginia), and us-west-1 (California)
* Reported times are for 1000 rows per input relation
* Not optimized plans use message batching too (otherwise the cost of MPC is prohibitive)

# ΑΠΟΤΕΛΕΣΜΑΤΑ ΤΩΝ ΒΕΛΤΙΣΤΟΠΟΙΗΣΕΩΝ ΣΕ ΠΡΑΓΜΑΤΙΚΑ ΕΡΩΤΗΜΑΤΑ (WAN)



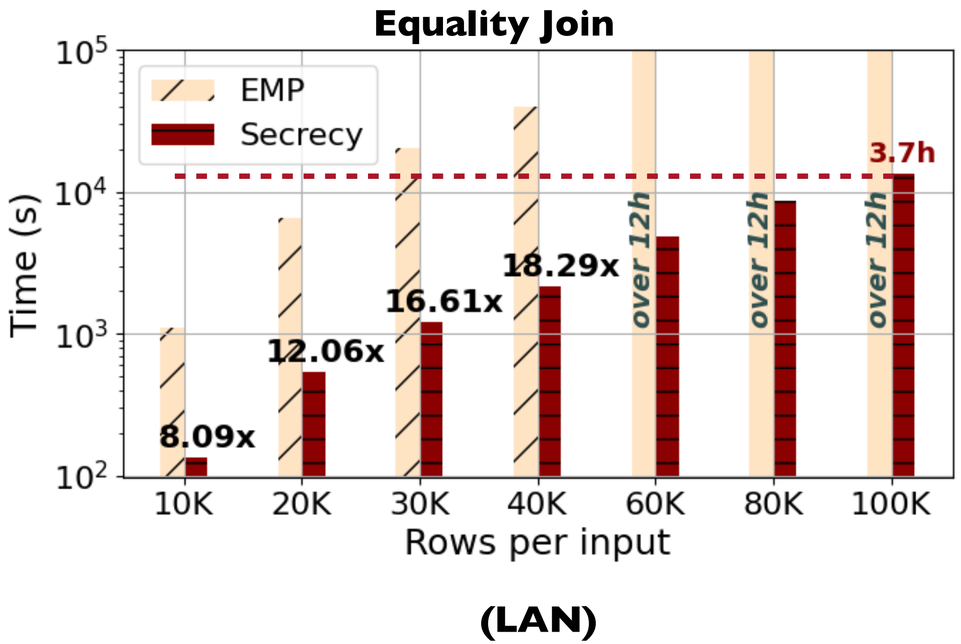

* Parties deployed in three AWS regions: us-east-2 (Ohio), us-east-1 (Virginia), and us-west-1 (California)
* Reported times are for 1000 rows per input relation
* Not optimized plans use message batching too (otherwise the cost of MPC is prohibitive)

# ΑΠΟΤΕΛΕΣΜΑΤΑ ΤΩΝ ΒΕΛΤΙΣΤΟΠΟΙΗΣΕΩΝ ΣΕ ΠΡΑΓΜΑΤΙΚΑ ΕΡΩΤΗΜΑΤΑ (WAN)




* Parties deployed in three AWS regions: us-east-2 (Ohio), us-east-1 (Virginia), and us-west-1 (California)
* Reported times are for 1000 rows per input relation
* Not optimized plans use message batching too (otherwise the cost of MPC is prohibitive)

# ΣΥΓΚΡΙΣΗ ΜΕ ΤΟ EMP (βιβλιοθήκη MPC γενικού σκοπού)



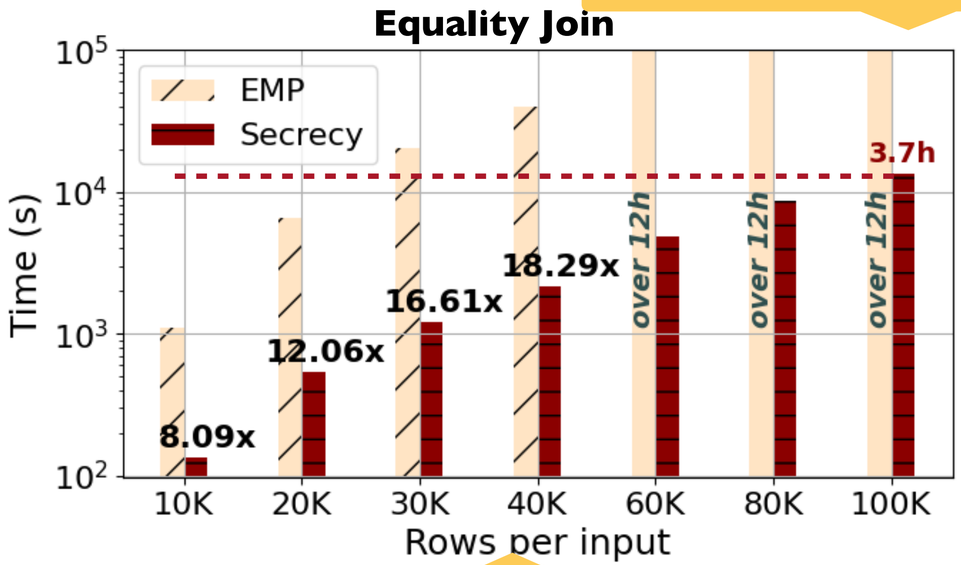

**(LAN)**

* Parties deployed on AWS EC2 r5.xlarge instances (us-east-2)

[1] X. Wang, A. J. Malozemoff, and J. Katz. *EMP-toolkit: Efficient MultiParty computation toolkit*, 2016. https://github.com/emp-toolkit

# ΣΥΓΚΡΙΣΗ ΜΕ ΤΟ EMP (βιβλιοθήκη MPC γενικού σκοπού)

Secrecy executes the join on 100K rows per input in ~3.7h



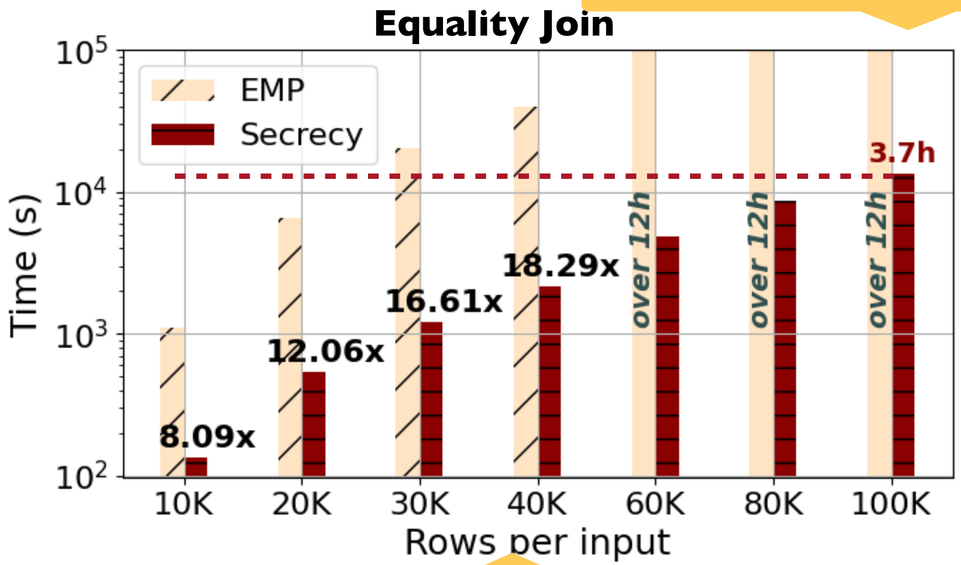

EMP requires ~11h for 40K rows per input

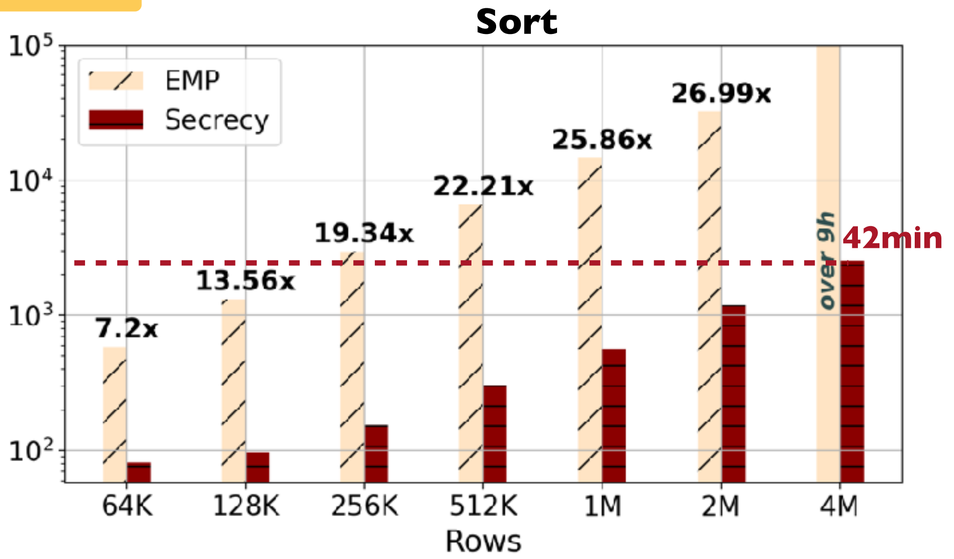
* Parties deployed on AWS EC2 r5.xlarge instances (us-east-2)

[1] X. Wang, A. J. Malozemoff, and J. Katz. *EMP-toolkit: Efficient MultiParty computation toolkit*, 2016. https://github.com/emp-toolkit

# ΣΥΓΚΡΙΣΗ ΜΕ ΤΟ EMP (βιβλιοθήκη MPC γενικού σκοπού)

Secrecy executes the join on 100K rows per input in ~3.7h




(WAN)

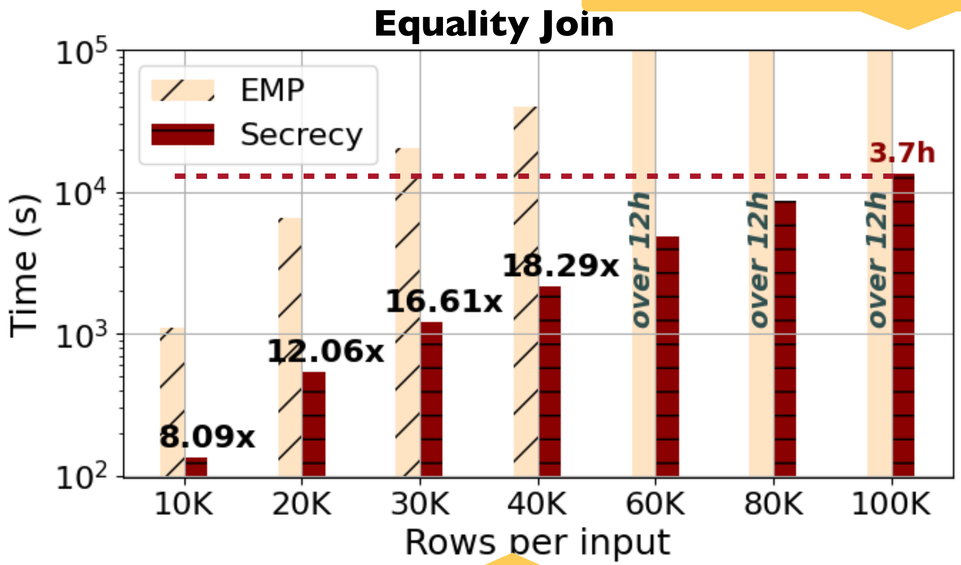
EMP requires ~11h for 40K rows per input

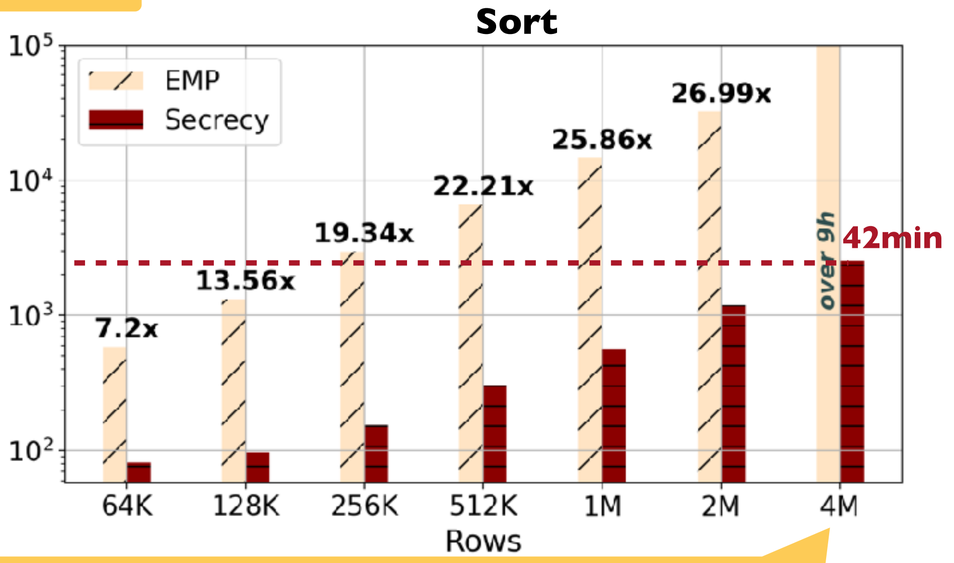
* Parties deployed on AWS EC2 r5.xlarge instances (us-east-1, us-east-2, us-west-1)

[1] X. Wang, A. J. Malozemoff, and J. Katz. *EMP-toolkit: Efficient MultiParty computation toolkit*, 2016. https://github.com/emp-toolkit

# ΣΥΓΚΡΙΣΗ ΜΕ ΤΟ EMP (βιβλιοθήκη MPC γενικού σκοπού)

Secrecy executes the join on 100K rows per input in ~3.7h



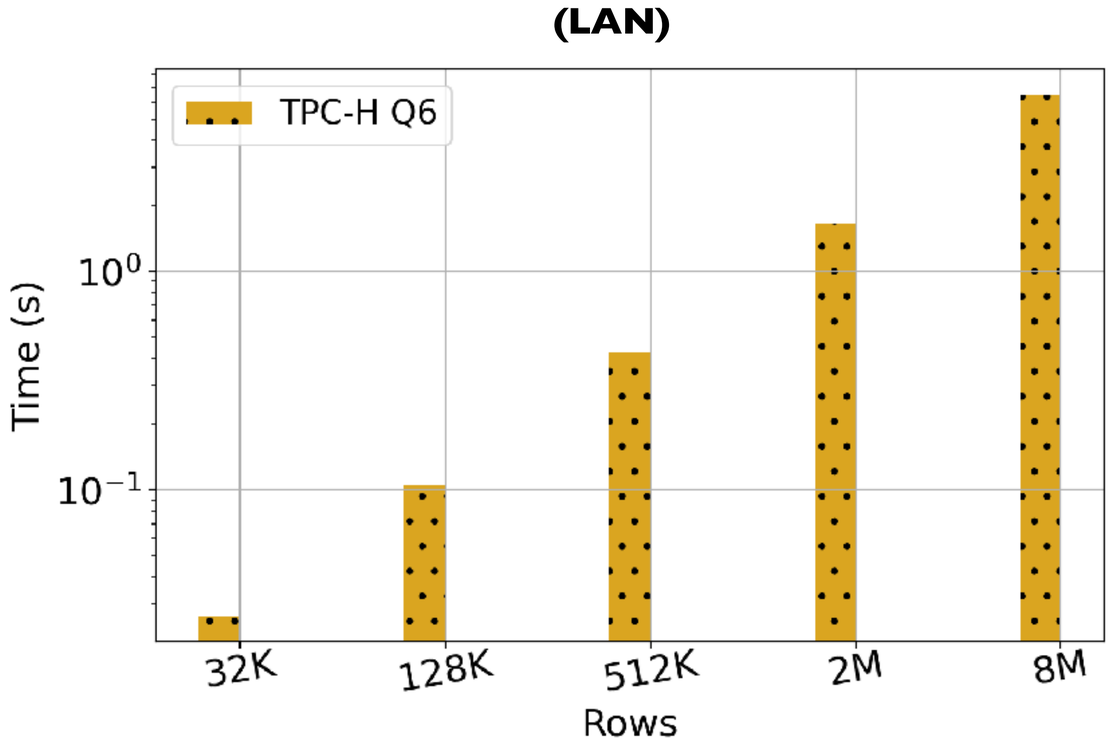

EMP requires ~11h for 40K rows per input

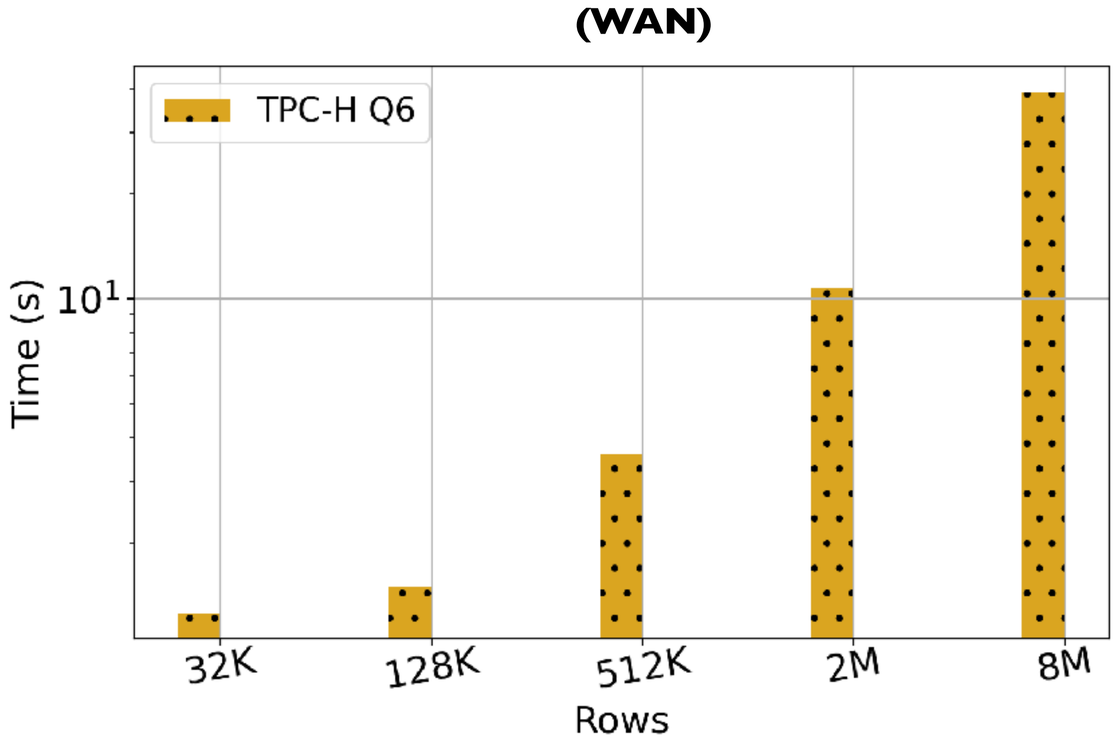
Secrecy sorts 4M rows in 42min whereas EMP requires more than 9h

* Parties deployed on AWS EC2 r5.xlarge instances (us-east-1, us-east-2, us-west-1)

[1] X. Wang, A. J. Malozemoff, and J. Katz. *EMP-toolkit: Efficient MultiParty computation toolkit*, 2016. https://github.com/emp-toolkit

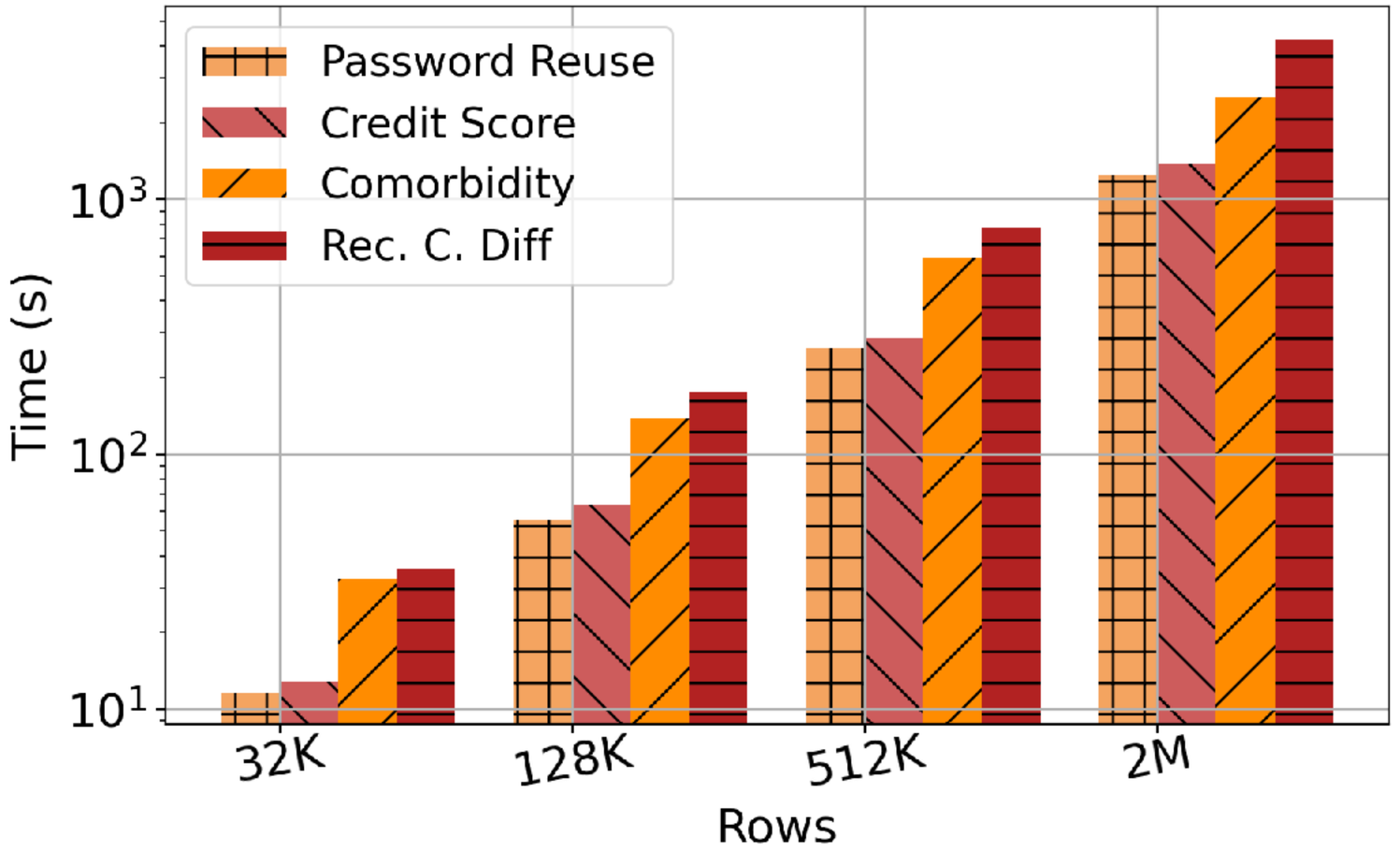
# TPC-H Q6: 5 SELECTIONS + GLOBAL AGGREGATION



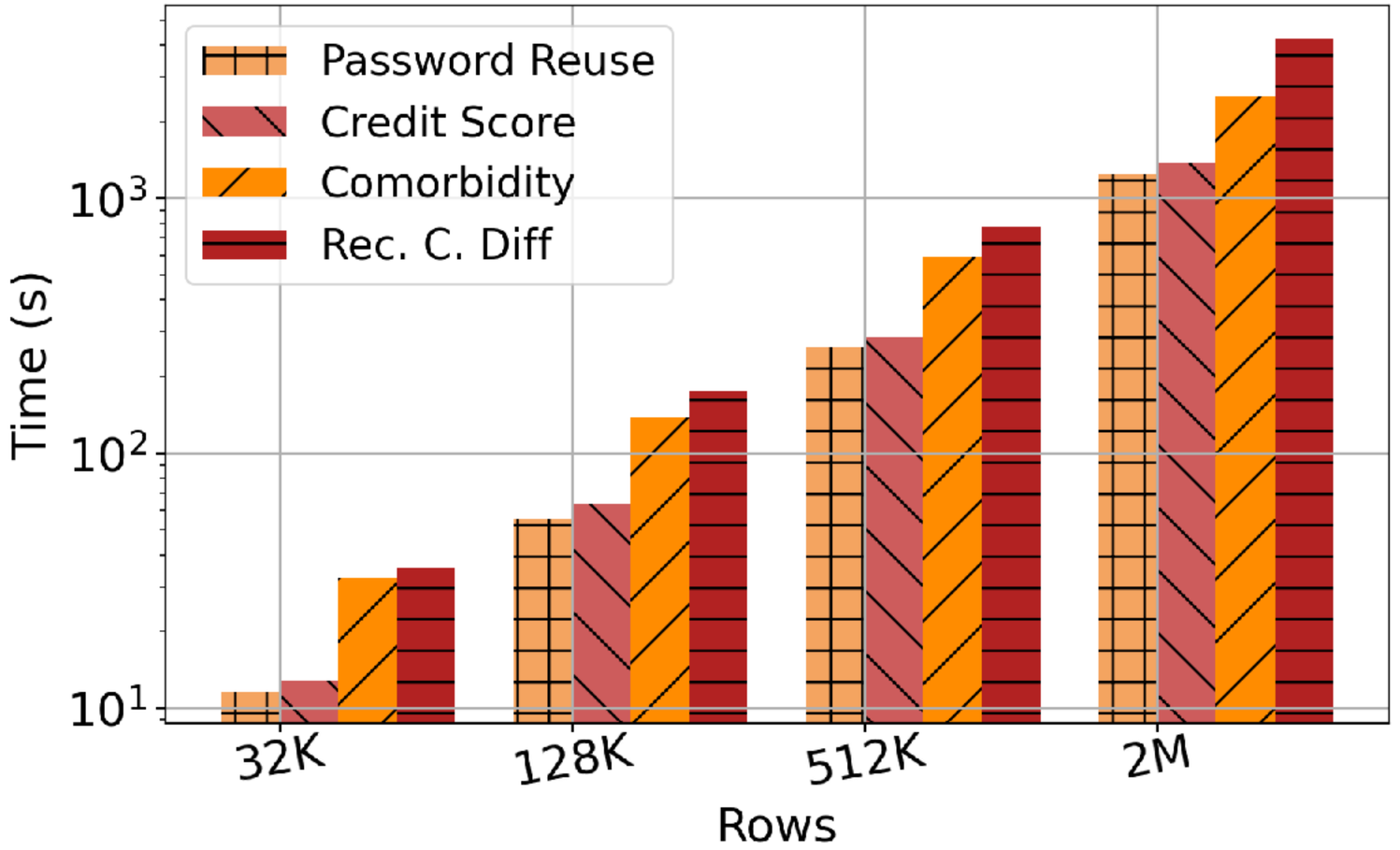

* Parties deployed on AWS EC2 r5.xlarge instances (us-east-1, us-east-2, us-west-1)

# QUERIES WITH GROUP-BY + AGGREGATION, SEMI-JOINS




* Parties deployed on AWS EC2 r5.xlarge instances (us-east-2)

# QUERIES WITH GROUP-BY + AGGREGATION, SEMI-JOINS




Rec. C. Diff scales to 2 million rows in ~1.2h

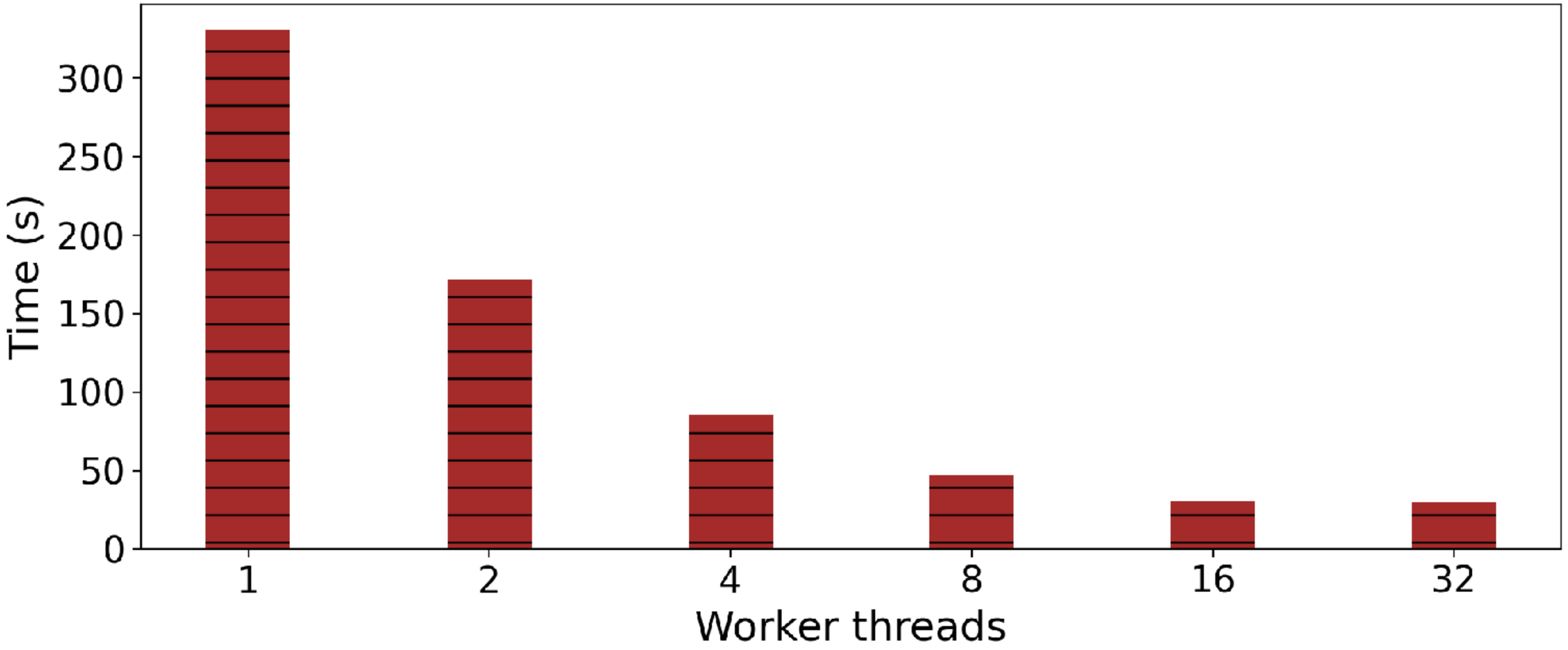
```
WITH rcd AS (
    SELECT pid, time, row_no() OVER
    (PARTITION BY pid ORDER BY time)
    FROM diagnosis
    WHERE diag=cdiff)
SELECT DISTINCT pid
FROM rcd r1 JOIN rcd r2 ON r1.pid = r2.pid
WHERE r2.time - r1.time >= 15 DAYS
AND r2.time - r1.time <= 56 DAYS
AND r2.row_no = r1.row_no + 1
```

*"Find the distinct ids of patients who have been diagnosed with cdiff and have two consecutive infections between 15 and 56 days apart"*

* Parties deployed on AWS EC2 r5.xlarge instances (us-east-2)

# Τρέχουσες και μελλοντικές κατευθύνσεις

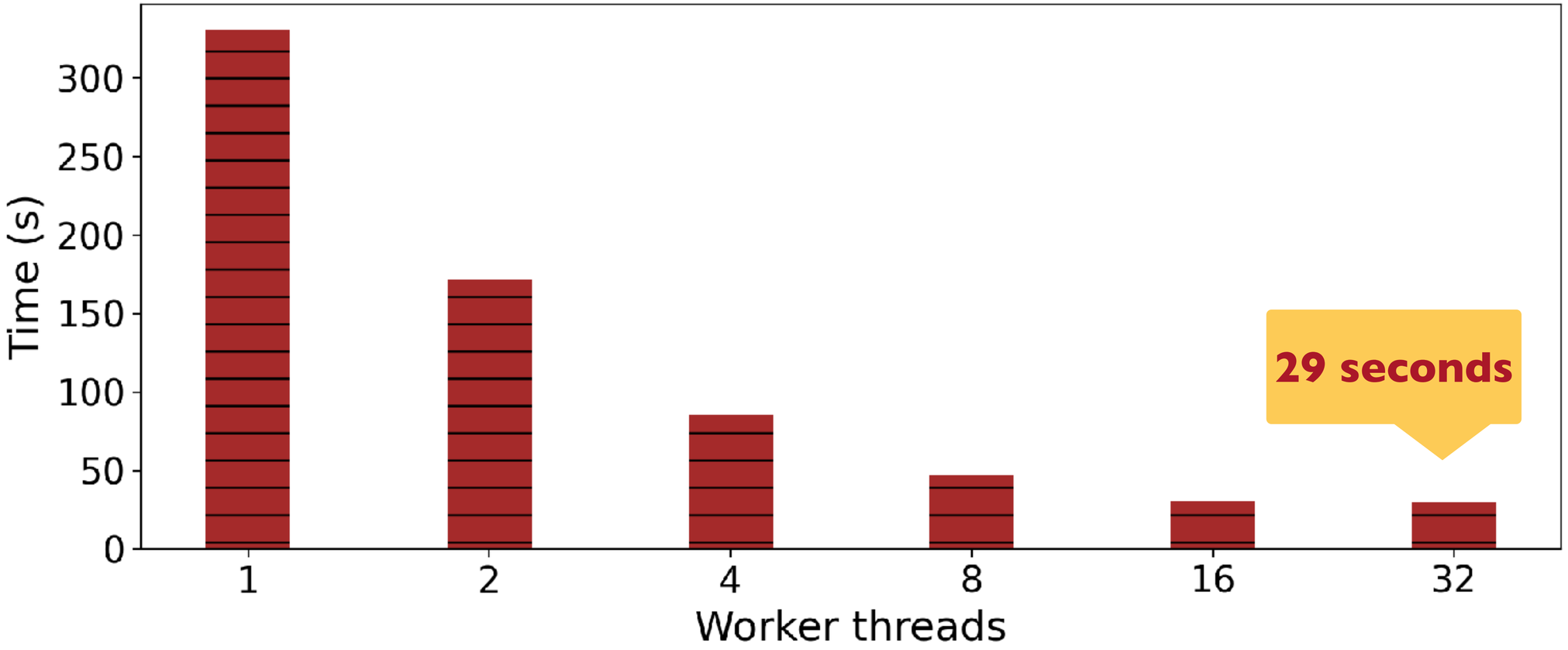
# PARALLEL OBLIVIOUS SORT ON SECRECY (LAN)




* Reported times are for 2M input rows

* Parties deployed on AWS EC2 r5.xlarge instances (us-east-2)

# PARALLEL OBLIVIOUS SORT ON SECRECY (LAN)



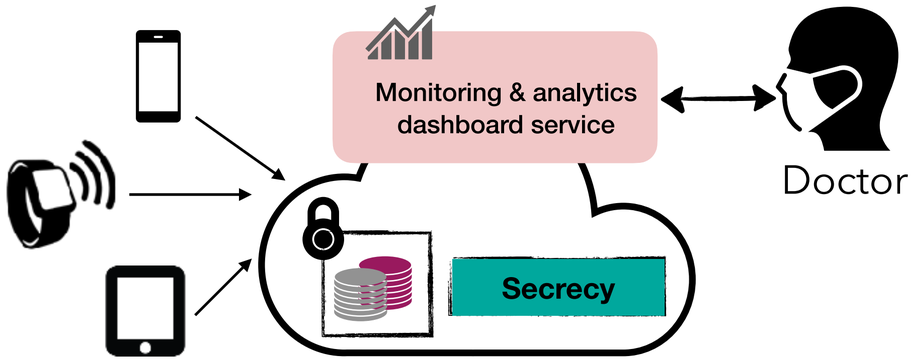

* Reported times are for 2M input rows

* Parties deployed on AWS EC2 r5.xlarge instances (us-east-2)

# ΠΡΑΓΜΑΤΙΚΕΣ ΕΦΑΡΜΟΓΕΣ ΤΟΥ SECRECY

Digital Health Analytics for vulnerable populations

(BU Medical & Hariri Institute for Computing)

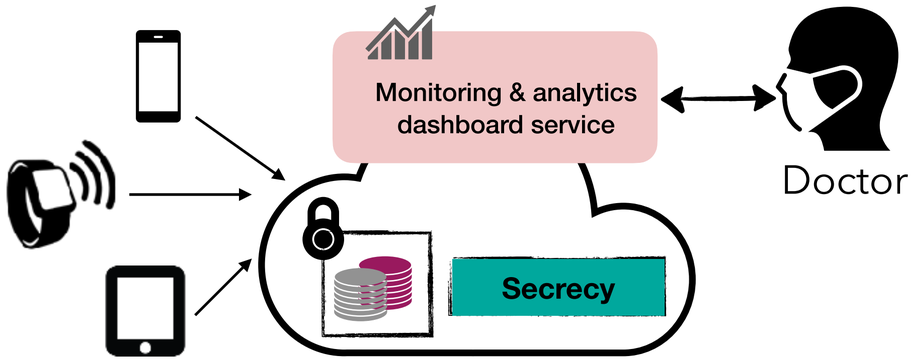



https://www.bu.edu/hic/research/focused-research-programs/continuous-analysis-of-mobile-health-data-among-medically-vulnerable-populations/

# ΠΡΑΓΜΑΤΙΚΕΣ ΕΦΑΡΜΟΓΕΣ ΤΟΥ SECRECY

## Digital Health Analytics for vulnerable populations

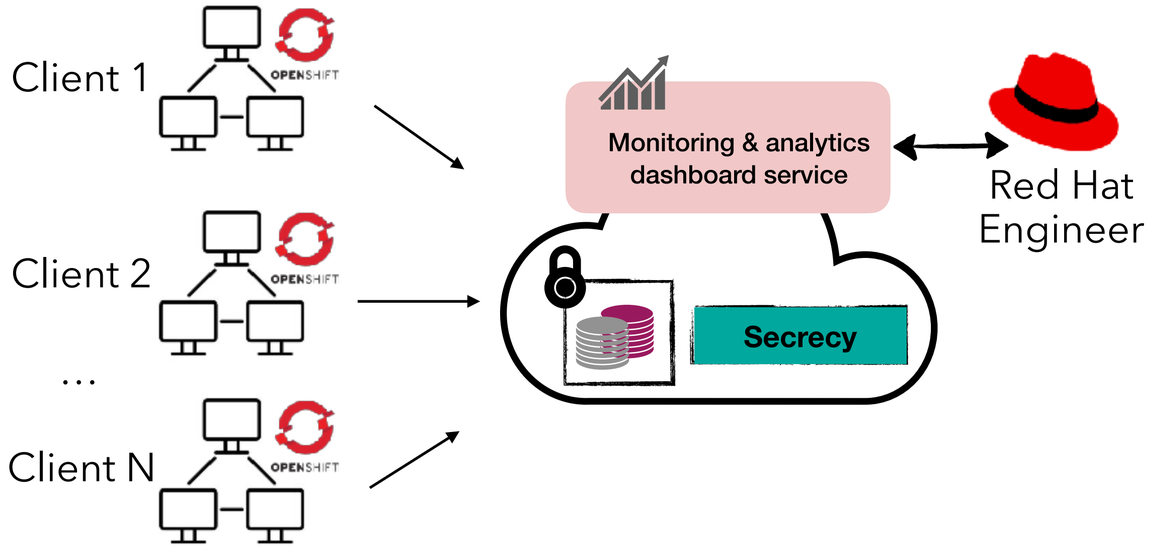
(BU Medical & Hariri Institute for Computing)




https://www.bu.edu/hic/research/focused-research-programs/continuous-analysis-of-mobile-health-data-among-medically-vulnerable-populations/

## Secure cross-site analytics on OpenShift logs

(BU RedHat Collaboratory)




https://www.bu.edu/rhcollab/projects/security-privacy/secure-cross-site-analytics-on-openshift-logs/

# WHAT'S NEXT?

- Παραλληλοποίηση και περαιτέρω βελτιστοποίησης απόδοσης
- Προσθήκη πρωτοκόλλων MPC για ευέλικτη επιλογή εγγυήσεων ασφάλειας
- Βελτιστοποίηση ερωτημάτων με multi-way joins
- Ανάπτυξη βιβλιοθηκών MPC  για εφαρμογές σε time series & machine learning

# Η ΟΜΑΔΑ ΤΟΥ SECRECY

John Liagouris

Vasia Kalavri

Muhammad Faisal

Mayank Varia

Jerry Zhang

Ian Saucy

Jingyu Su

Derek Ewers

Xavier Ruiz

Past contributors (BSc/MSc):

Pierre-Francois Wolf, Henry Wu,
Lucas Ou, Bang Tran, Hasnain Abdul Rehman,
Samyak Jain, Suli Hu, Yufeng Lin, Eric Chen

**Secrecy paper (NSDI'23):** https://sites.bu.edu/casp/files/2022/10/secrecy-nsdi23.pdf

# We are hiring students and postdocs

https://sites.bu.edu/casp/

## with generous support from

BOSTON UNIVERSITY

NSF

BOSCH

# SECRECY:
# *Secure collaborative analytics in untrusted clouds*

Vasiliki (Vasia) Kalavri
vkalavri@bu.edu
sites.bu.edu/casp