ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

COM–402

An Example Application of Anonymous Credentials and Homomorphic Encryption

The case of privacy-preserving ride-hailing services

Many thanks to Anh Pham for producing several of the slides

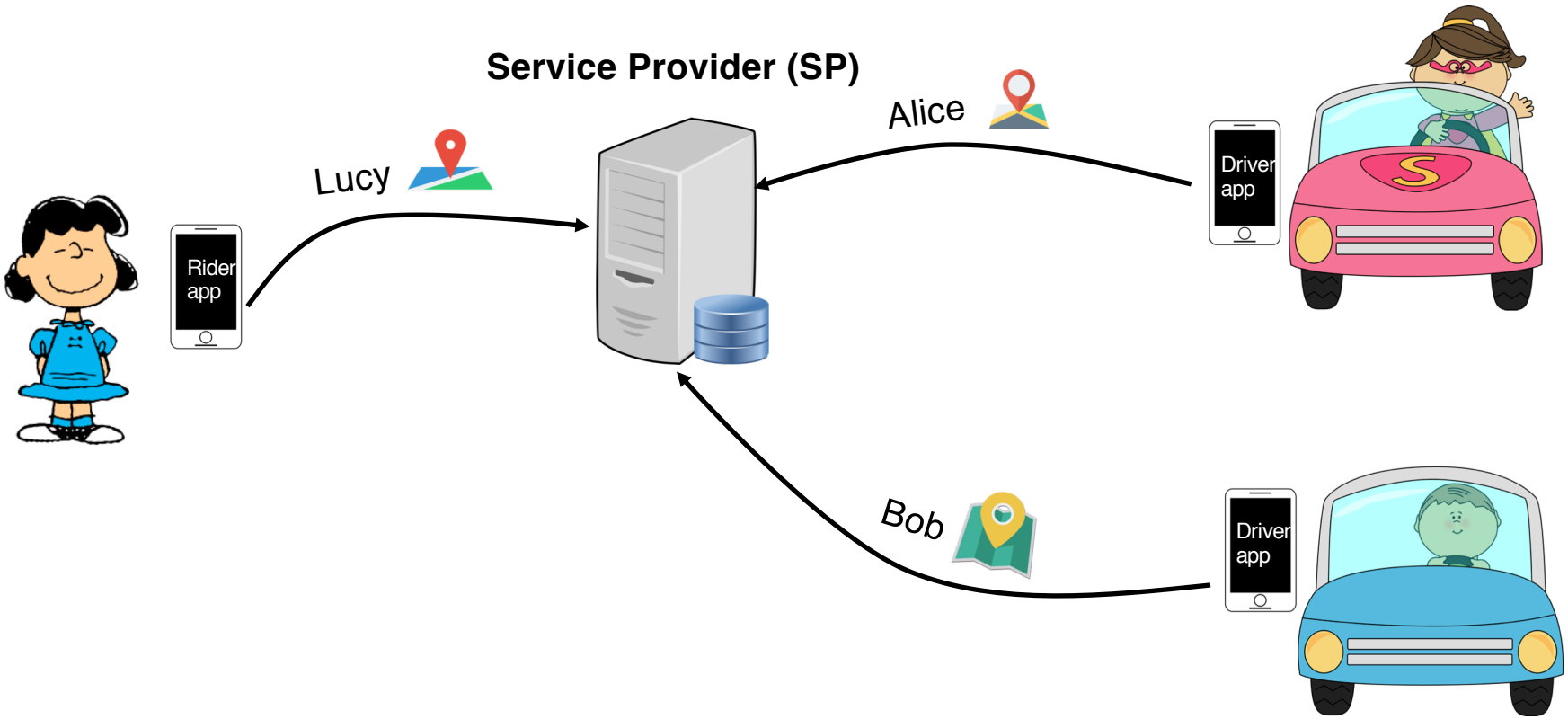
Brief reminder

- Anonymous credential (AC)
  - A cryptographic token that enables her holder to prove to a service provider that she is a legitimate user **without** having to reveal her identity

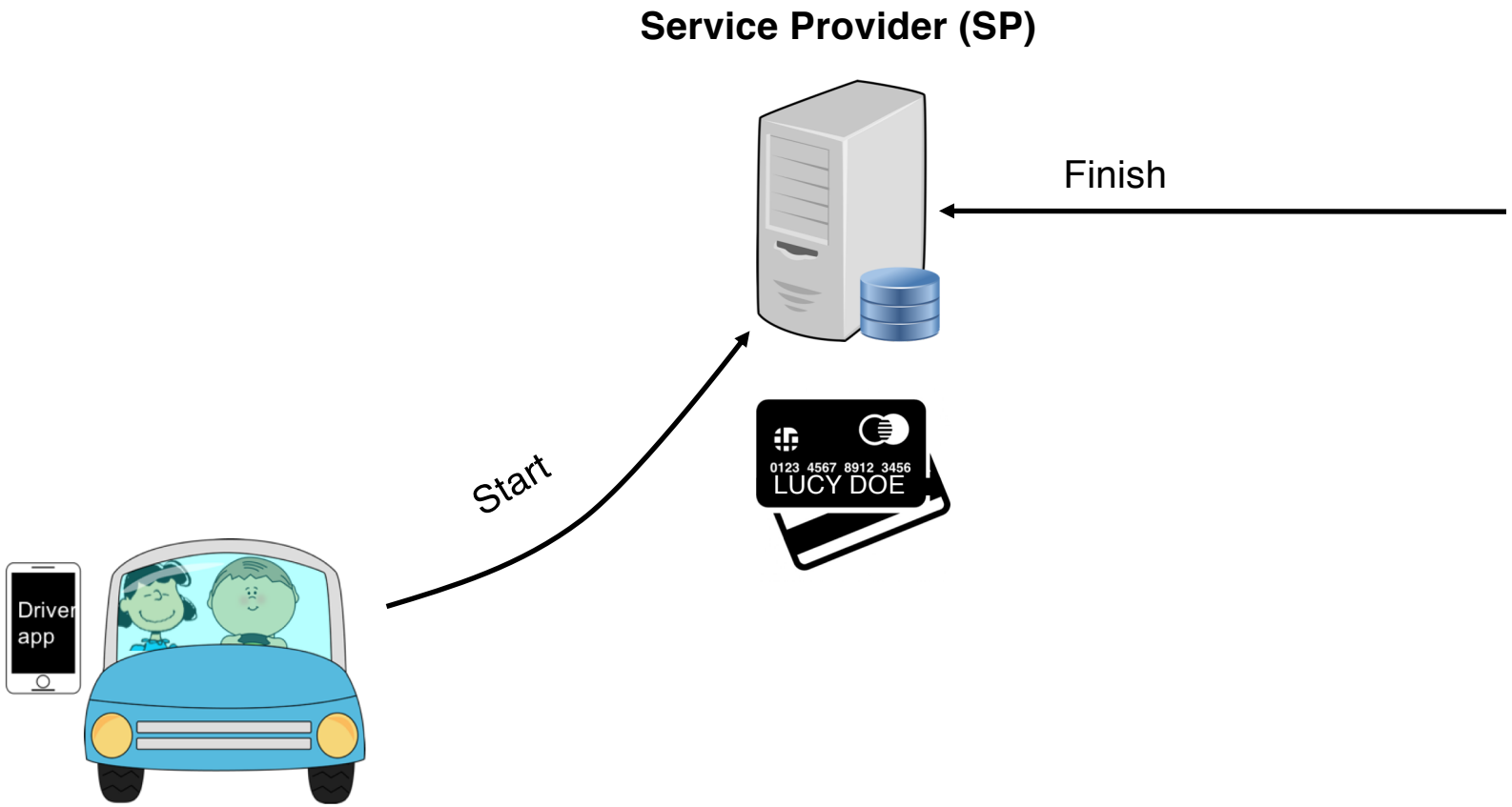
- Somewhat homomorphic encryption (SHE)
  - Supports a limited number of additions and multiplications on ciphertexts
  - Much faster than fully homomorphic encryption (FHE)

How to use them to design privacy-preserving systems?

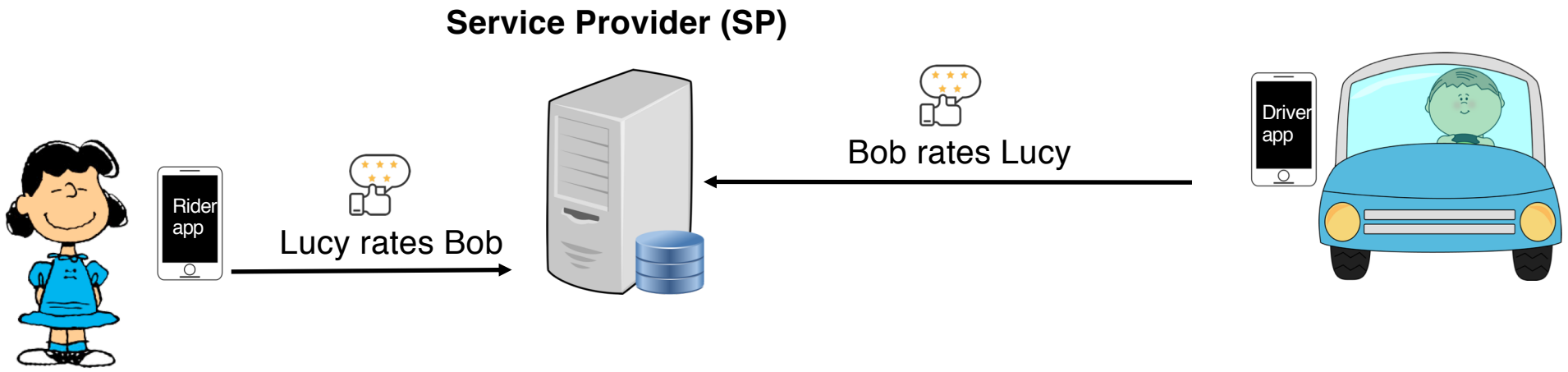
Ride-Hailing Service (RHS)



Ride-Hailing Service (RHS)



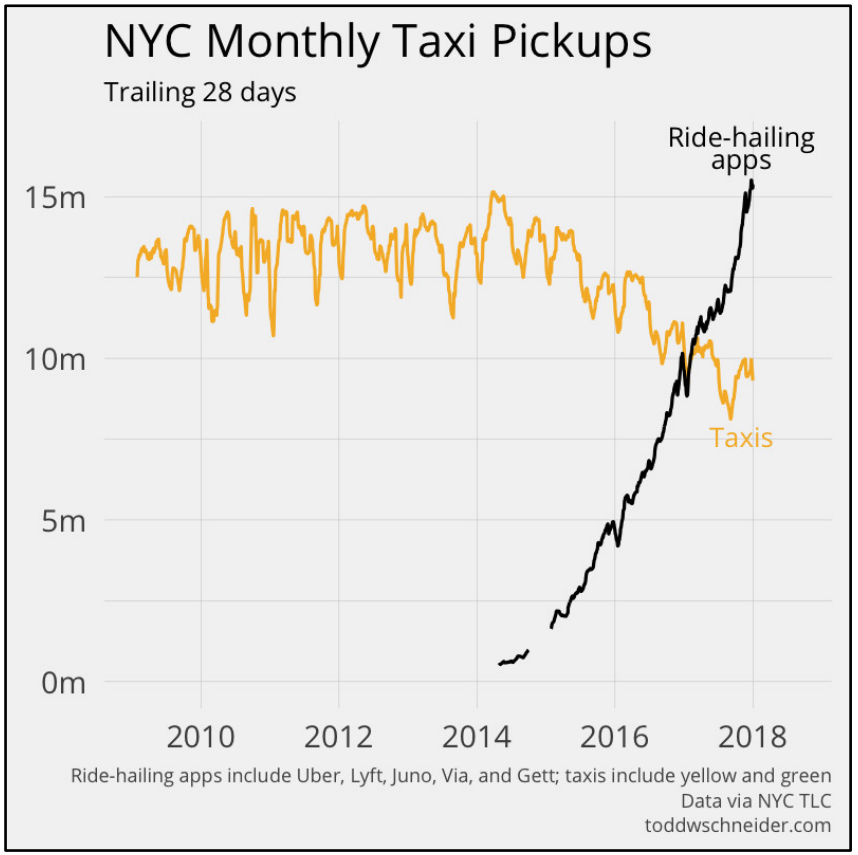
Ride-Hailing Service (RHS)



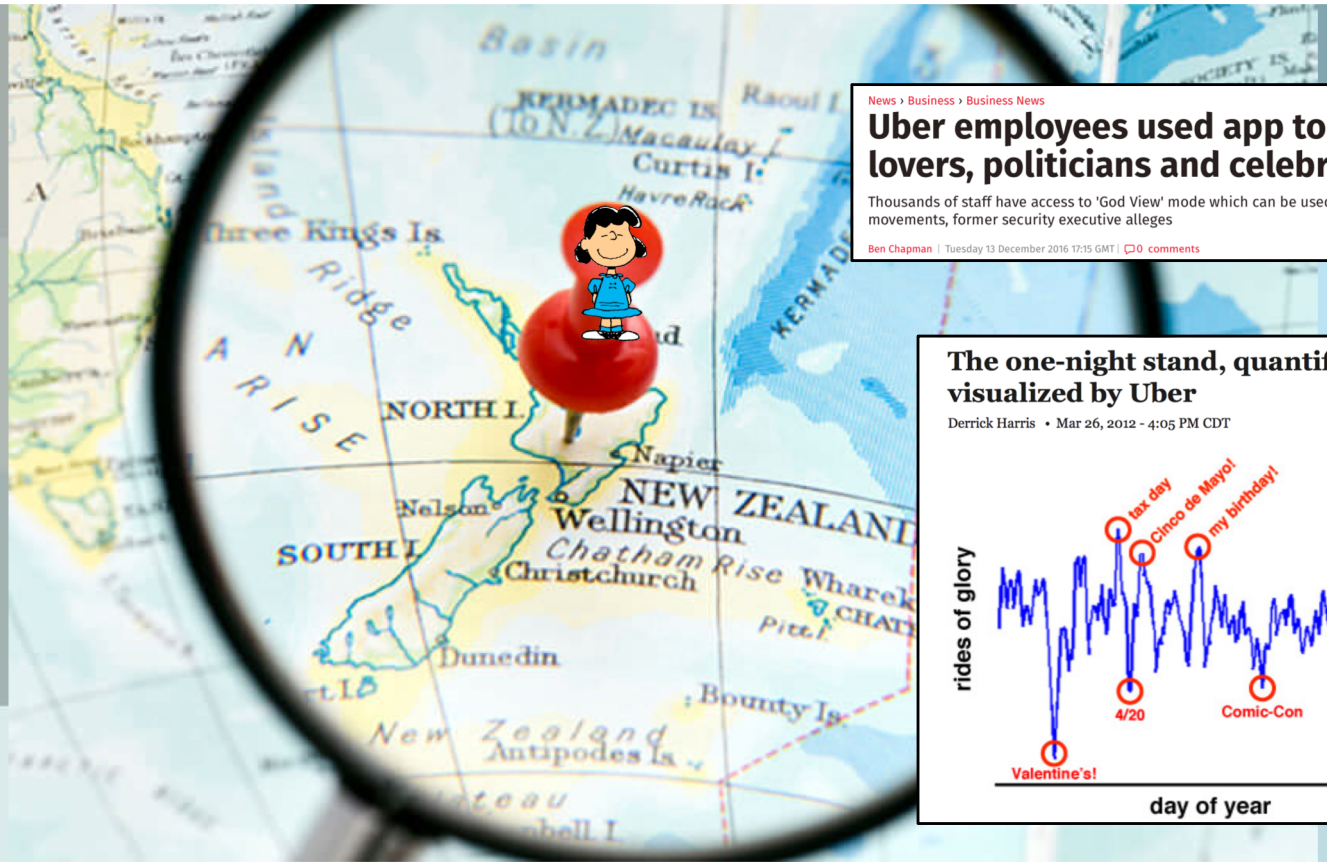
RHSs Are Increasingly Popular

Uber powered four billion rides in 2017. It wants to do more — and cheaper — in 2018.

Johana Bhuiyan • Jan 5, 2018, 4:09pm EST



Privacy Problems



News › Business › Business News

Uber employees used app to 'spy on ex-lovers, politicians and celebrities'

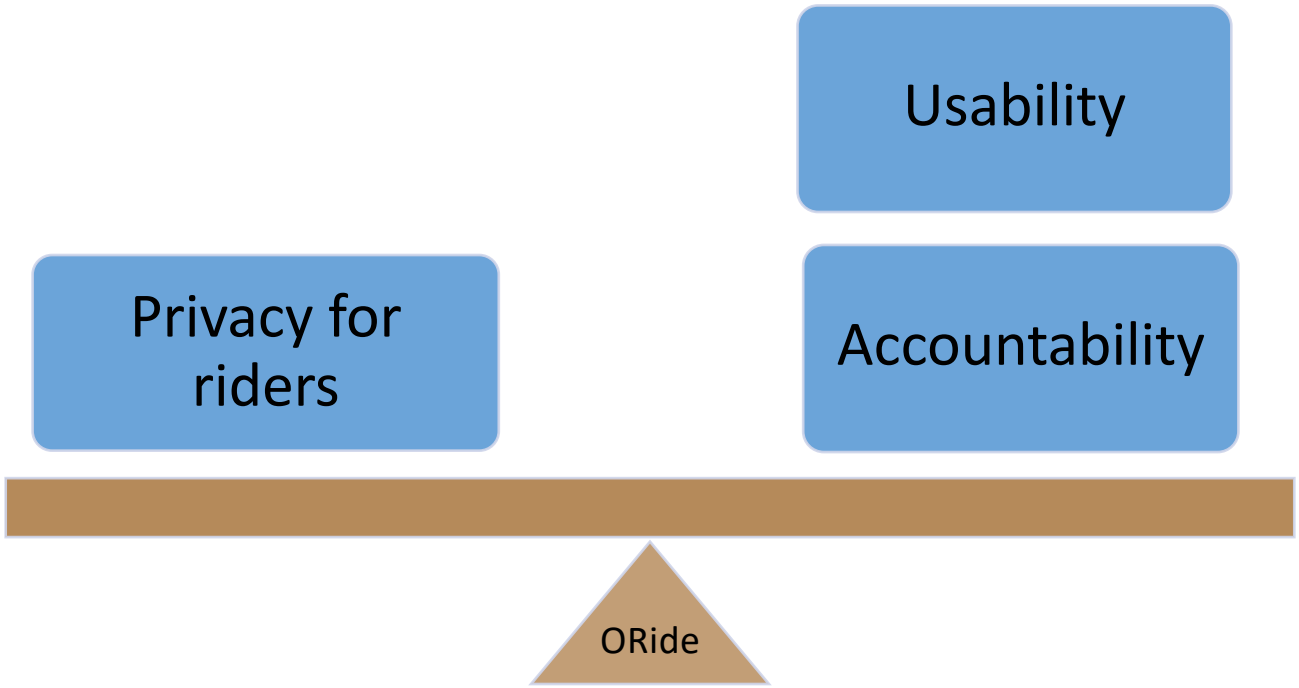
Thousands of staff have access to 'God View' mode which can be used to track customers' movements, former security executive alleges

Ben Chapman | Tuesday 13 December 2016 17:15 GMT | 0 comments

The one-night stand, quantified and visualized by Uber

Derrick Harris • Mar 26, 2012 - 4:05 PM CDT

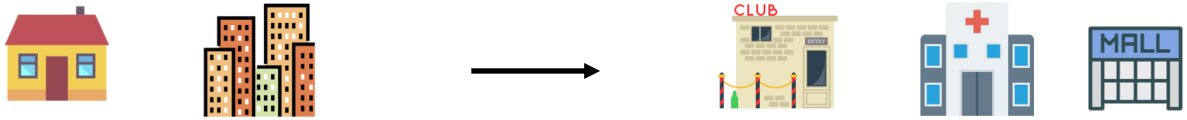
ORide: Goals



Goal 1

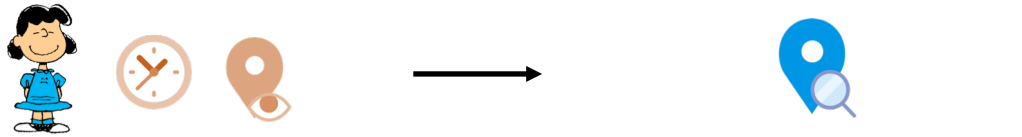
Privacy for riders

- Protect riders against large-scale privacy-sensitive inferences by the SP

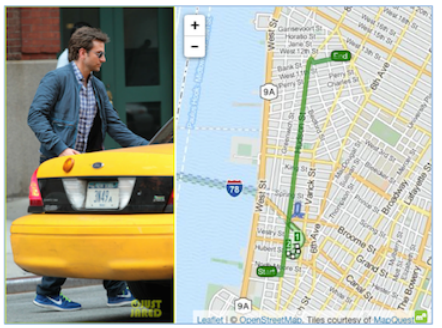


Home/work addresses → Sensitive activities

- Protect riders against targeted attacks by the SP



Pick-up location and time → Drop-off location



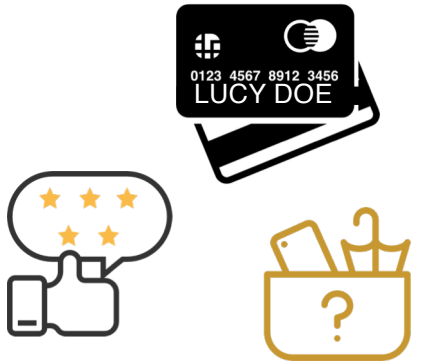
https://research.neustar.biz/2014/09/15/riding-with-the-stars-passenger-privacy-in-the-nyc-taxicab-dataset/

Goals 2 + 3

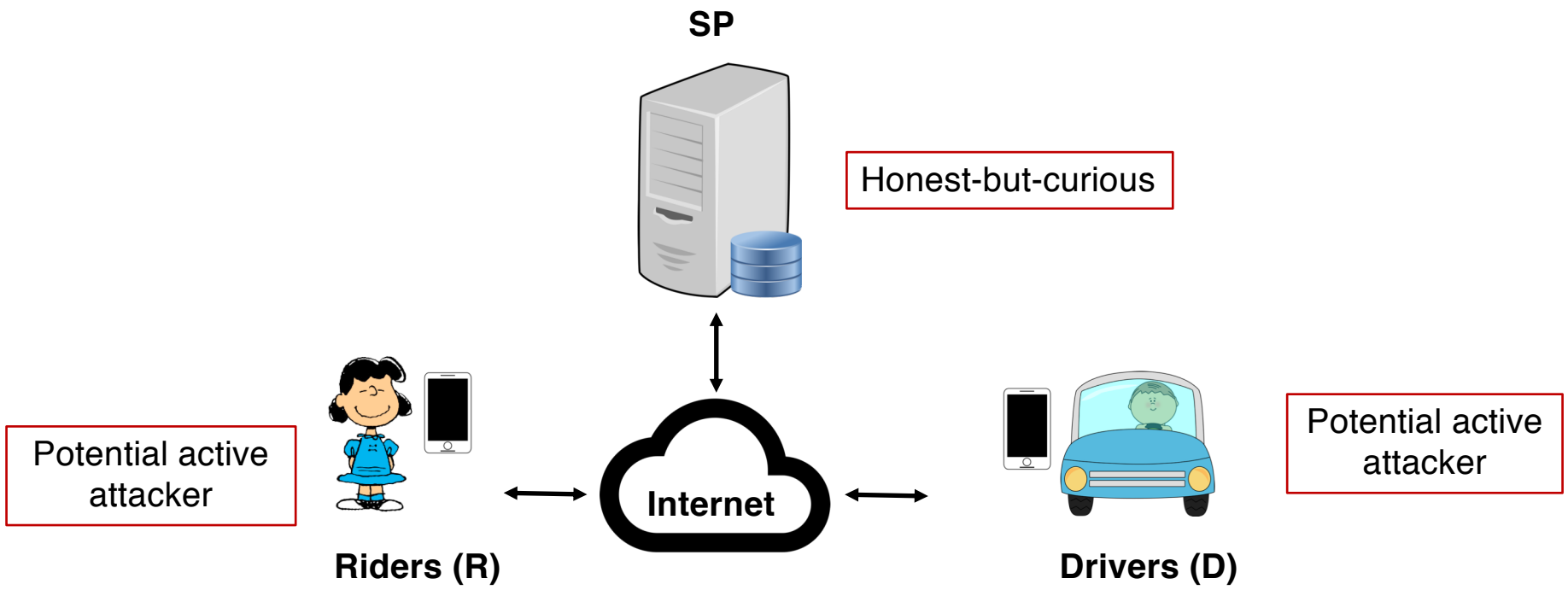
Accountability

Usability

- SP can identify misbehaving riders and drivers
  - Riders and drivers might assault each other
  - Riders might not want to pay for the ride
  - Dispute on the fare
- Usability:
  - Easy payment
  - Reputation rating
  - Retrieval of lost items



Adversarial Assumptions



ORide Overview

Notations:
ACs: Anonymous credentials
cert: Digital certificate

Riders (anonymous)

SP

Drivers (anonymous)

Ride set-up

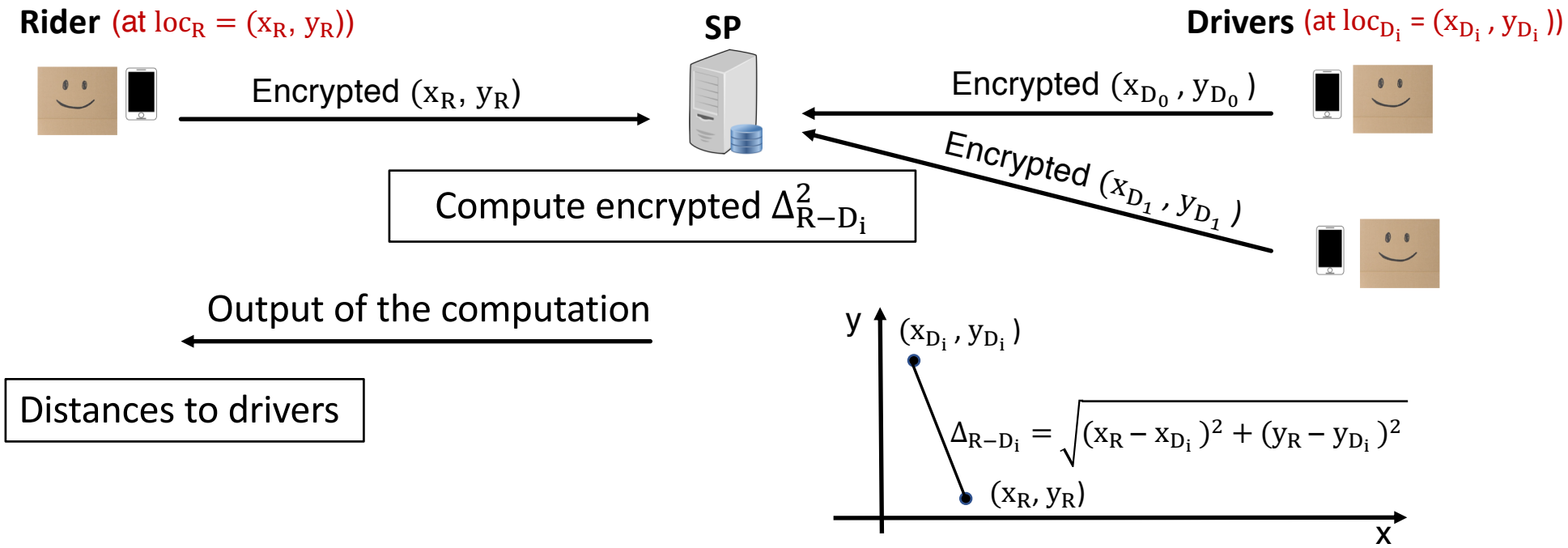
- Oblivious ride matching
- Pick-up preparation

Ride termination

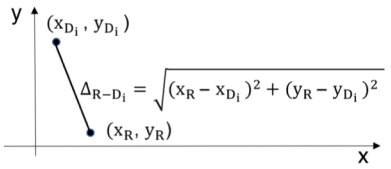
- Payment
- Reputation rating

Oblivious Ride Matching - Intuition

- **Goal:** Rider can select the closest driver w/o revealing her locations to SP



Oblivious Ride Matching



- **Goal:** Rider can select the closest driver w/o revealing their locations to SP

Rider (at $\text{loc}_R = (x_R, y_R)$)

SP

e.g., Lausanne

Drivers (at $\text{loc}_{D_i} = (x_{D_i}, y_{D_i})$)

(1) zone_{D_i}

Gen. (k_p, k_s)

(2) zone_R, $\mathbf{E}_{k_p}\{x_R\}$, $\mathbf{E}_{k_p}\{y_R\}$, k_p

(3) k_p

(4) $\mathbf{E}_{k_p}\{x_{D_i}\}$, $\mathbf{E}_{k_p}\{y_{D_i}\}$

Driver D_i

Compute $\mathbf{E}_{k_p}\{\Delta^2_{R-D_i}\}$

Use k_s to decrypt
$\mathbf{E}_{k_p}\{\Delta^2_{R-D_1}\}$
...
$\mathbf{E}_{k_p}\{\Delta^2_{R-D_n}\}$

(5) $\mathbf{E}_{k_p}\{\Delta^2_{R-D_1}\}$,..., $\mathbf{E}_{k_p}\{\Delta^2_{R-D_n}\}$

(6) Index ind

(7) Notify driver with index ind

Explanation of previous slide

The drivers frequently send the zones of their locations to the SP. When a rider wants to request a ride, she generates an *unauthenticated* ephemeral public/private key pair. She uses the public key to encrypt her coordinates and she sends to the SP a message consisting of the ephemeral public key, the cipher texts and the zone of her location. The SP then broadcasts this ephemeral public key to all drivers who are in the same zone with the rider.

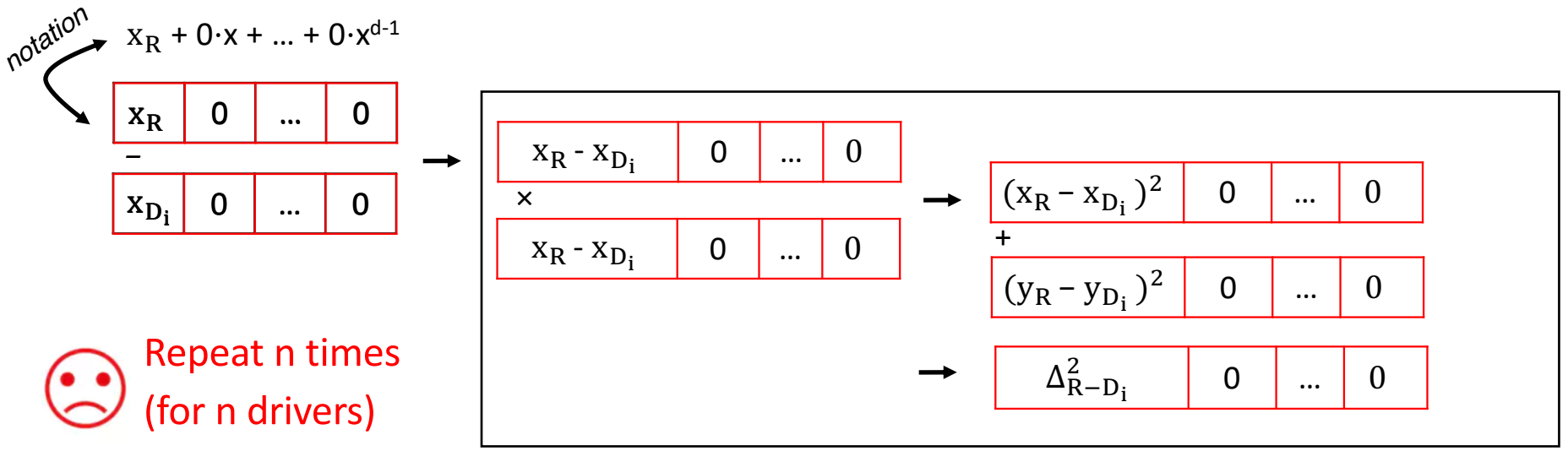
In short, it's not that the drivers send their zone information to the SP *right before* the rider's request, but the drivers *frequently* do so.

k_p is not authenticated during the ride-matching phase. But there is no concern about man-in-the-middle-attacks here, because later on, the rider and the driver set up their secure channel and then short-range communication channel, and via these two channels, they can check the integrity of the key k_p that they received during the ride-matching phase.

SP Computes $\mathbf{E}_{k_p}\{\Delta^2_{R-D_i}\}$

- Use Somewhat-Homomorphic Encryption (SHE)

$$\begin{cases} R_t = Z_t[X]/(X^d + 1) & \text{Plaintexts} \\ R_q = Z_q[X]/(X^d + 1) & \text{Ciphertexts and keys} \end{cases}$$

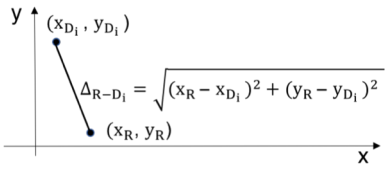


Repeat n times
(for n drivers)

Explanation of previous slide

- d, t and q: integer numbers chosen depending on the bit-security levels. Examples of values: t = 2^20, q = 2^124, and d = 4096. We'll come back to this later.
- We want to support both addition and multiplication (in order to compute the squared Euclidian distance), and therefore we need to use somewhat homomorphic encryption. They are typically based on polynomial quotient rings.
- Before the rider and driver encrypt their coordinate, for example the x coordinate, they have to represent it as a polynomial of degree d-1 with only one non-zero coefficient at degree 0 of the polynomial.

SP Computes $\mathbf{E}_{k_p}\{\Delta^2_{R-D_i}\}$



- **Optimization:** 1 computation per request for the riders and the SP

Naïve

$\Delta^2_{R-D_0}$	0	…	0
⋮			
$\Delta^2_{R-D_1}$	0	…	0
⋮			
$\Delta^2_{R-D_{n-1}}$	0	…	0

Optimized

$\Delta^2_{R-D_0}$	$\Delta^2_{R-D_1}$	…	$\Delta^2_{R-D_{n-1}}$	0	…	0

Coefficient i contains $\Delta^2_{R-D_i}$

Optimization: Packing at Coefficient Level

- Rider: uses all coefficients of the plaintext polynomial

x_R	x_R	…	x_R	…	x_R

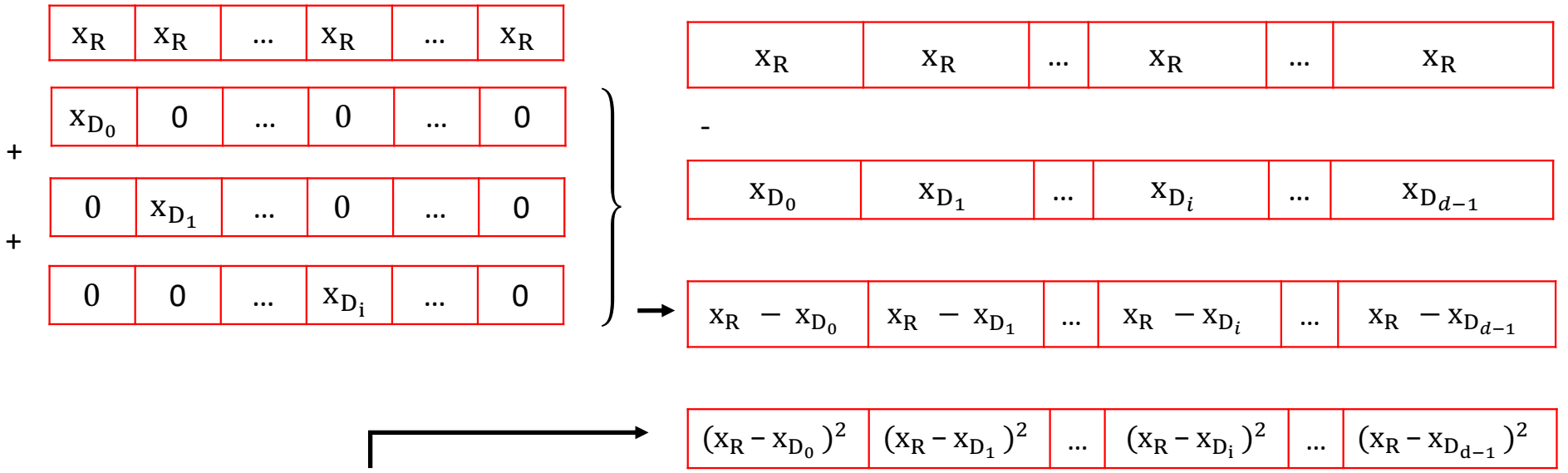
- Each driver is assigned an index by the SP
  - Driver with index i: uses the coefficient at degree i

Driver 0	x_{D_0}	0	…	0	…	0
Driver 1	0	x_{D_1}	…	0	…	0
Driver i	0	0	…	x_{D_i}	…	0

Assumption: i < d-1

Optimization: Packing at Coefficient Level

Ciphertexts received by the SP



SP wants to compute

Needs element-wise multiplication

Element-Wise Multiplication

- Use Number-Theoretic Transform (NTT): the finite ring version of a Discrete Fourier Transform (DFT); NTT^{-1}: Inverse NTT
- Let A and B be two polynomials; the rider computes the following:

$$\underline{\mathrm{NTT}}(\underline{\mathrm{NTT}^{-1}(\mathrm{A}) \times \mathrm{NTT}^{-1}(\mathrm{B})}) = \mathrm{A} \cdot \mathrm{B}$$

Rider and drivers apply iNTT on their plaintext polynomials **before** encryption

After decrypting the ciphertext received from the SP, the rider applies NTT on the plaintext polynomial

Explanation of the previous slide

- NTT is the finite ring version of a Discrete Fourier Transform (DFT). No need to know how the transform works
- Important: the property that we get from NTT: considering two polynomials A and B, we take the *convolutional product* of their inverse-NTT, and we apply the NTT on the result of the convolutional product. In this way we obtain the *element-wise* product of the two polynomials.

  We use that property to support element-wise multiplications of two ciphertexts.

ORide Overview

Notations:
ACs: Anonymous credentials
cert: Digital certificate

Riders (anonymous)

SP

Drivers (anonymous)

Ride set-up

- ✓ Oblivious ride matching
- → Pick-up preparation

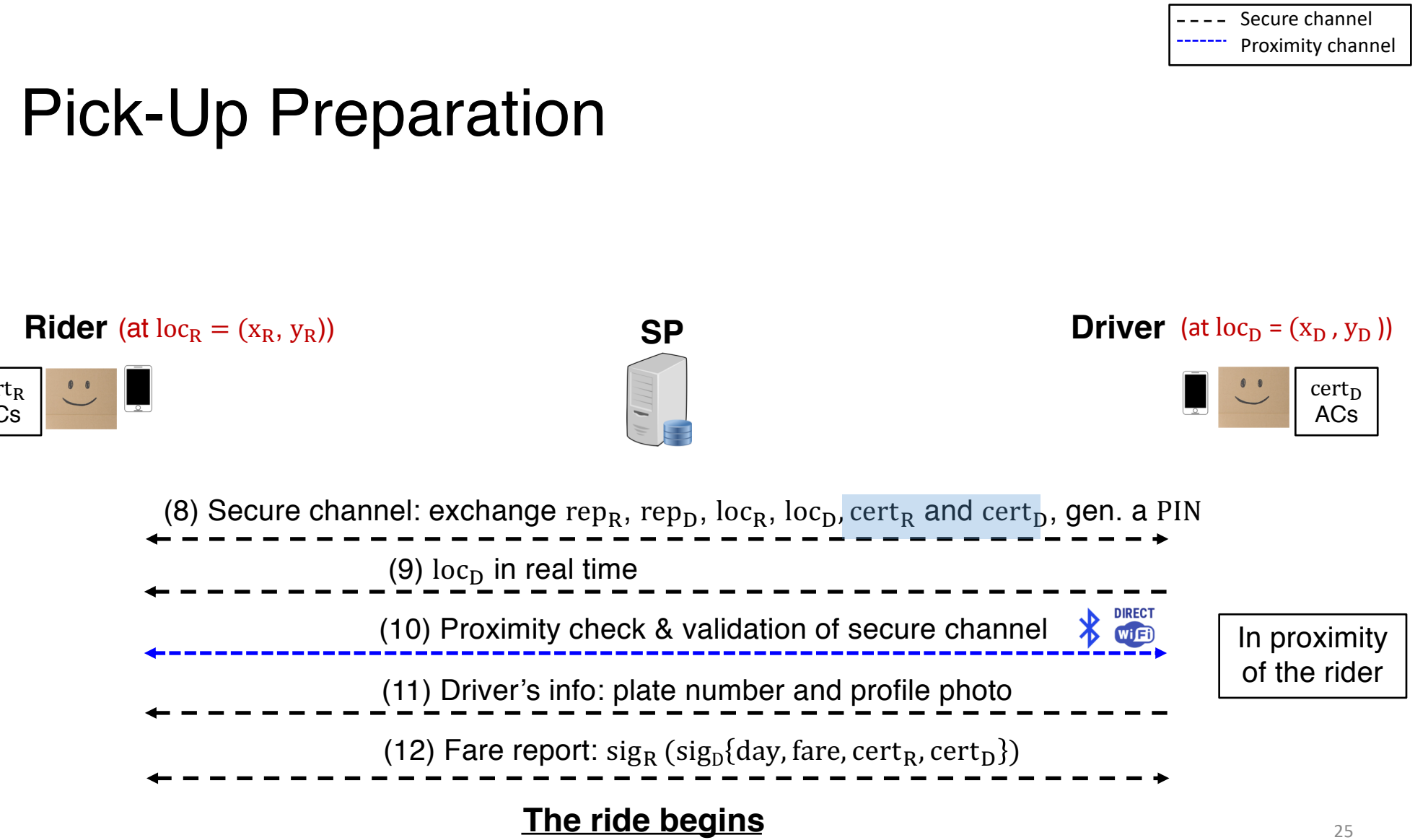
Ride termination

- Payment
- Reputation rating

Pick-Up Preparation - Goals

- The rider and the driver share their locations with each other

- Accountability
  - Physical attacks
  - The rider might not want to pay for the ride
  - Dispute on the fare


Secure channel
Proximity channel
Pick-Up Preparation
Rider (at $\mathrm{loc_R} = (\mathrm{x_R}, \mathrm{y_R})$)
SP
Driver (at $\mathrm{loc_D} = (\mathrm{x_D}, \mathrm{y_D})$)
$\mathrm{cert_R}$ ACs
$\mathrm{cert_D}$ ACs
(8) Secure channel: exchange $\mathrm{rep_R}$, $\mathrm{rep_D}$, $\mathrm{loc_R}$, $\mathrm{loc_D}$, $\mathrm{cert_R}$ and $\mathrm{cert_D}$, gen. a PIN
(9) $\mathrm{loc_D}$ in real time
(10) Proximity check & validation of secure channel
DIRECT
Wi Fi
In proximity of the rider
(11) Driver's info: plate number and profile photo
(12) Fare report: $\mathrm{sig_R}$ ($\mathrm{sig_D}${day, fare, $\mathrm{cert_R}$, $\mathrm{cert_D}$})
The ride begins

Explanation of previous slide

The rider gets to see the reputation of the driver after the latter has been assigned to her. If the rider is not happy about the driver's reputation, she can choose the second-closest driver from her set of distances to the drivers (in the plaintext polynomial she already has). Rider and driver can see each other’s reputations only *after* being matched together, because knowledge of the reputations could help the SP to de-anonymize the users.

ORide Overview

Notations:
ACs: Anonymous credentials
cert: Digital certificate

Riders (anonymous)

SP

Drivers (anonymous)

Ride set-up

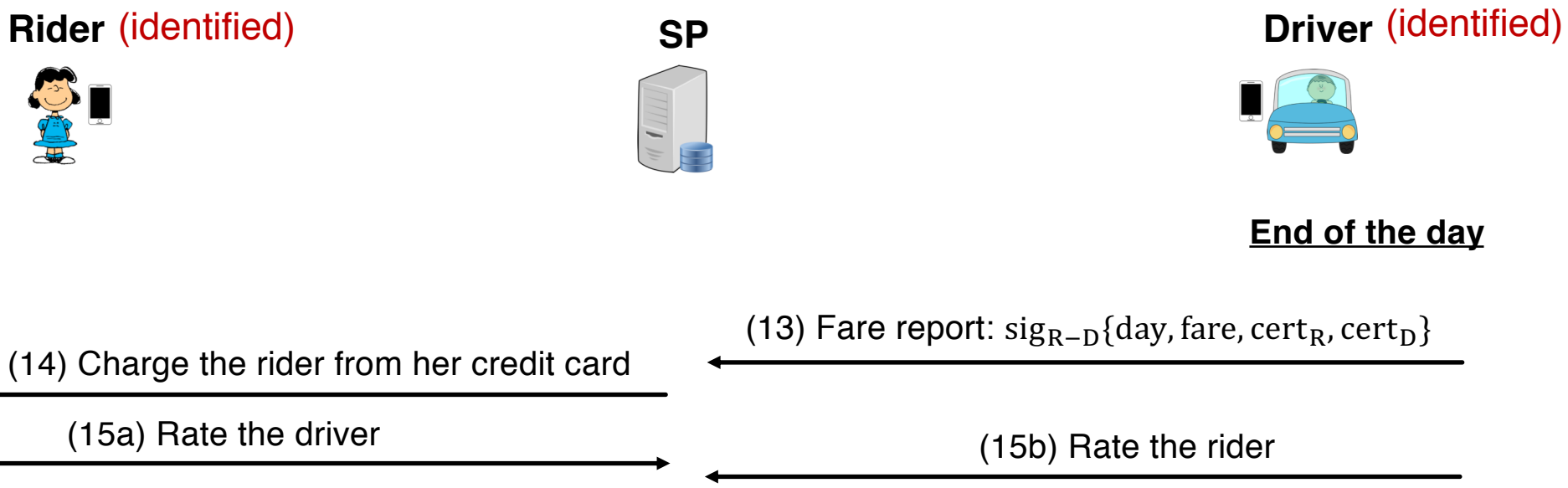
- ✓ Oblivious ride matching
- ✓ Pick-up preparation

Ride termination

- ➡ Payment
- ➡ Reputation rating

Payment and Reputation Rating

- **Goal:** Preserve the anonymity of the ride



ORide Overview

Notations:
ACs: Anonymous credentials
cert: Digital certificate

Riders (anonymous)

SP

Drivers (anonymous)

Ride set-up

- ☑ Oblivious ride matching
- ☑ Pick-up preparation

Ride termination

- ☑ Payment
- ☑ Reputation rating

Protocol Analysis

- Information observed by the SP

Ride DB

Zone of the pick-up	**Obfuscated** pick-up time

Payment DB

Rider's ID	Driver's ID	Fare	Day of the ride

- Targeted attacks by the SP

Pick-up location and time of a specific rider

Drop-off location

Drop-off locations and times are never reported to the SP

Protocol Analysis

- Information observed by the SP

Ride DB

Zone of the pick-up	**Obfuscated** pick-up time

Payment DB

Rider's ID	Driver's ID	Fare	Day of the ride

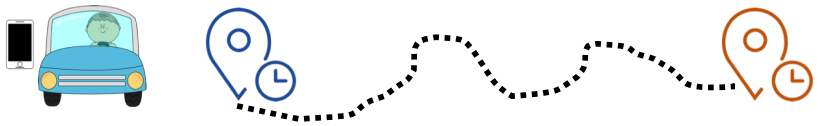
- Large-scale inferences by the SP

Home/work addresses → Sensitive activities

Lower-bound anonymity set = No. of rides from the same zone on the same day

Evaluation

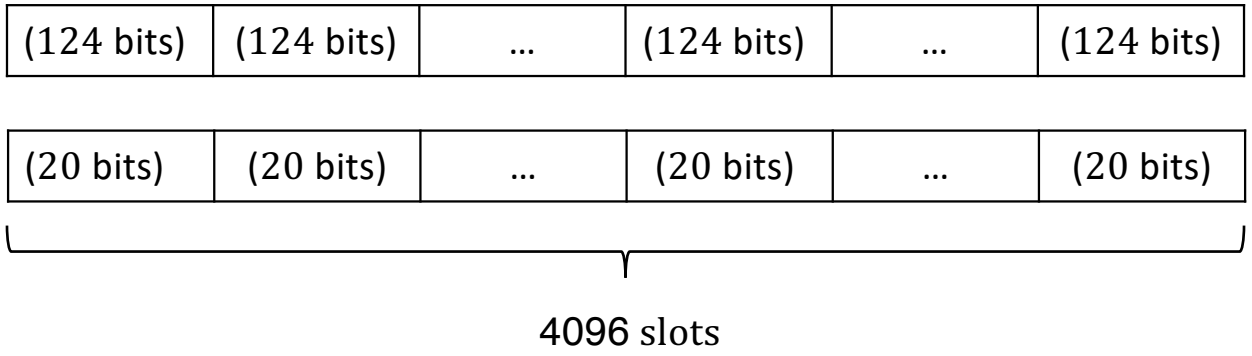
- Data-set: taxi rides in NYC in Oct. 2013 (15 million rides)



- How practical and efficient is ORide?
  - Per-ride overhead
  - Riders' anonymity vs. drivers' bandwidth
  - Effect of Euclidean distance on ride-matching optimality

Implementation

- Feature the oblivious ride matching algorithm
- SHE parameters: > 112 bits of security
  - Ciphertexts and keys
  - Plaintexts



- In C++ using NFLlib [1]
- No optimizations for Intel processors

NFLlib Is an open-source C++ library dedicated to lattice-based cryptography

[1] https://github.com/quarkslab/NFLlib

Per-Ride Overhead

S1: Naïve approach
S2: Optimized approach

- Measured on Intel i5-4200U, 2.6 GHz, 6 GB RAM (assuming it for Rider, Driver and SP)
- Bandwidth overhead

Setting	Rider		Driver	
Algorithm	Upload (KB)	Download (KB)	Download (KB)	Upload (KB)
S1	372	761856	124	248
S2	372	186	124	248

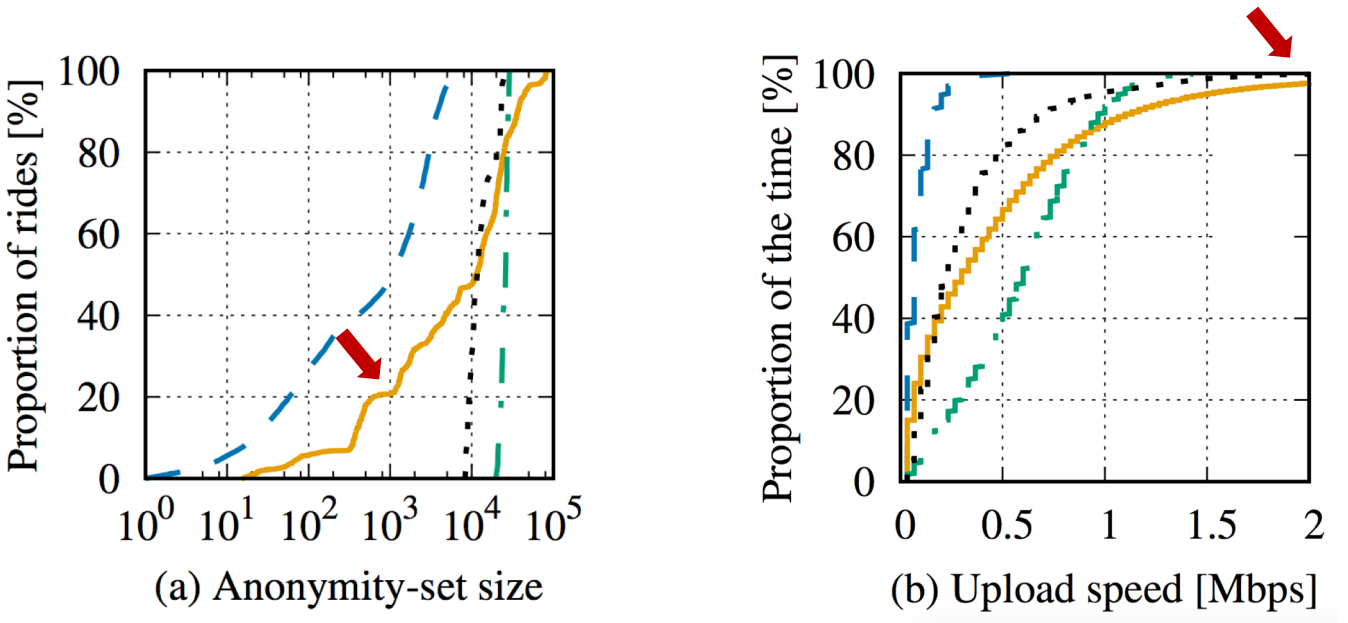
- Computation time

Setting	Rider			Driver		SP	
Algorithm	Gen. keys (ms)	Encrypt (ms)	Decrypt (ms)	Load key (ms)	Encrypt (ms)	Load key (ms)	Compute Dist. (ms)
S1	1.51	2.6	7823.4	0.53	2.6	0.53	113868.8
S2	1.51	2.6	2.2	0.53	2.6	0.53	208.9

The optimized approach (S2) significantly reduces bandwidth and computation overhead

Riders' Anonymity Set and Drivers' Bandwidth

- Zones: Queens + Bronx; Brooklyn + Staten Island; Manhattan CTs (Census Tracts); Manhattan NTAs (Neighborhood Tabulation Areas)



Fairly large anonymity set; reasonable bandwidth requirements

Explanation of previous slide

- The more ride requests are generated in a zone, the more anonymity a rider will enjoy (bigger anonymity sets). Yet, it requires drivers to respond to more requests, hence necessitating more bandwidth.
- CT: Census Tract and NTA: Neighbourhood Tabulation Area. A census tract is a statistical division of NYC for an area with a population of around 3000-4000 people, and a NTA is a set of census tracts (i.e., bigger than a census tract).
- For the suburbs, the splits are as follows. Boroughs of Queens and Bronx are merged into a zone, Brooklyn and Staten Island are one zone. For Manhattan, each census tract is a zone (in case of Manhattan CTs), or each NTA is a zone (in case of NTAs).

ORide: Conclusion

- ORide: practical and privacy-preserving
  - Strong privacy guarantees
  - Negligible overhead

- Still offers key ride-hailing service features:
  - Accountability
  - Easy payment
  - Reputation scores

http://oride.epfl.ch