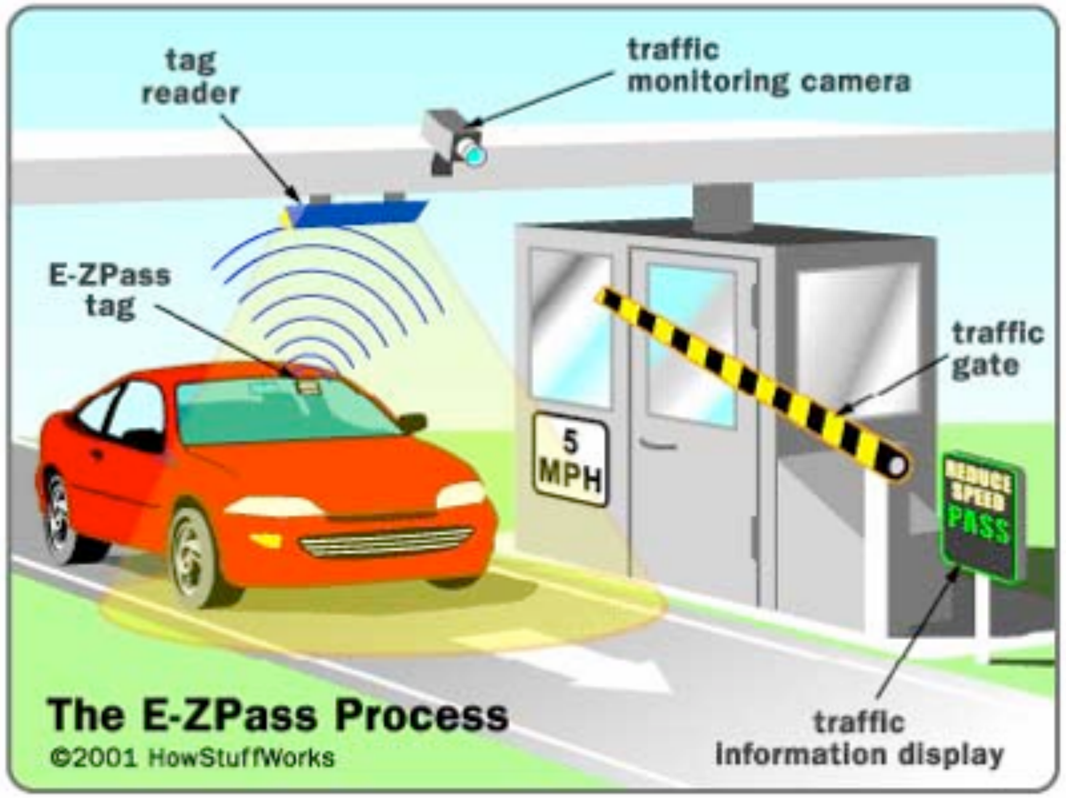
The Phantom Tollbooth: Privacy-Preserving Toll Collection in the Presence of Driver Collusion

Sarah Meiklejohn (UC San Diego)
Keaton Mowery (UC San Diego)
Stephen Checkoway (UC San Diego)
Hovav Shacham (UC San Diego)

Motivation: how tolling works today

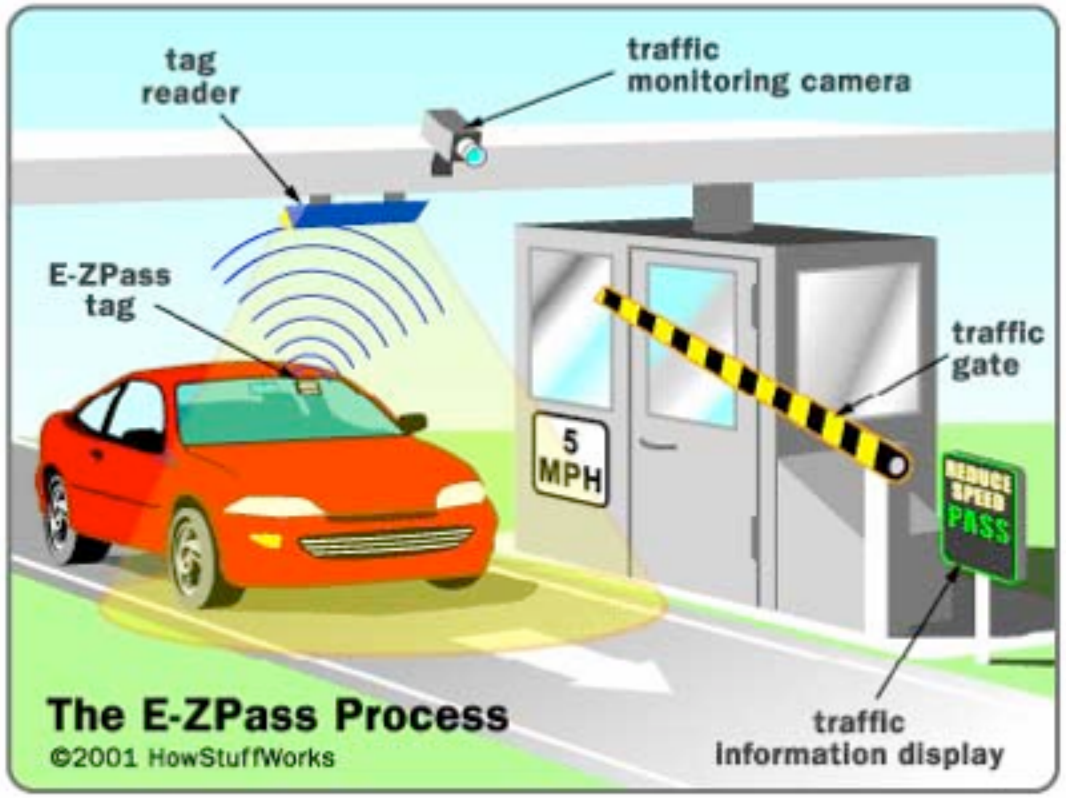
Motivation: how tolling works today

Motivation: how tolling works today



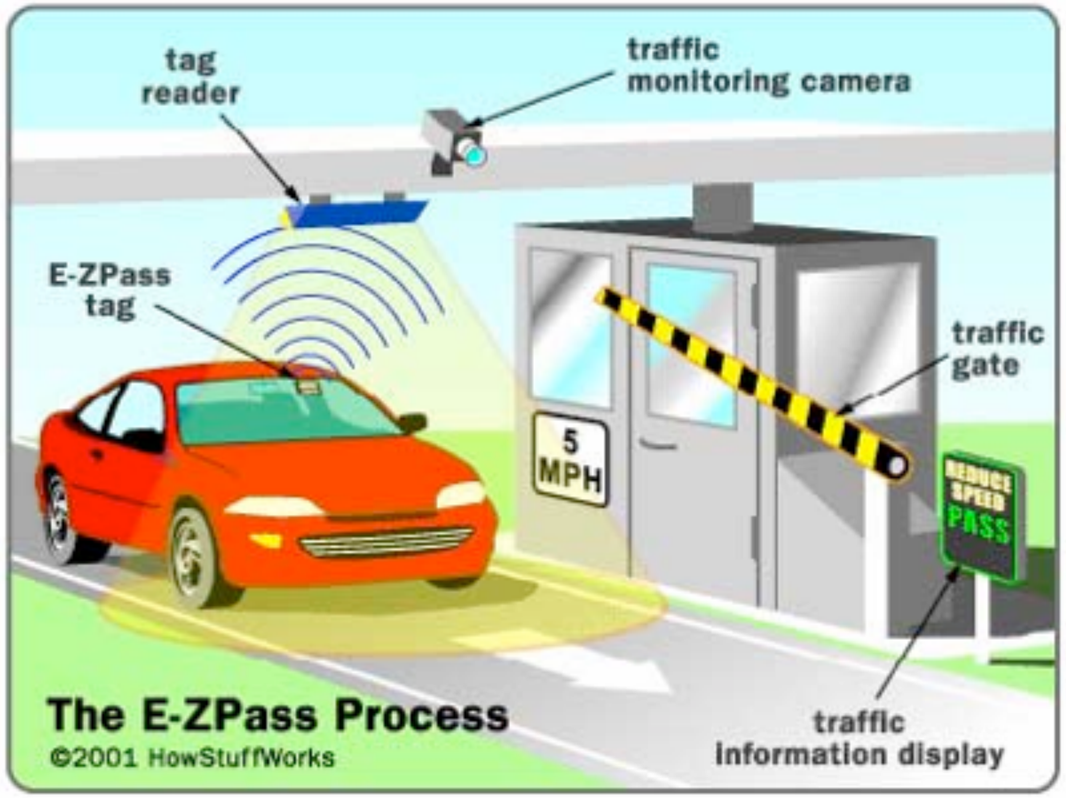
©2001 HowStuffWorks

Motivation: how tolling works today



This process leaves a lot to be desired in terms of flexibility:

Motivation: how tolling works today



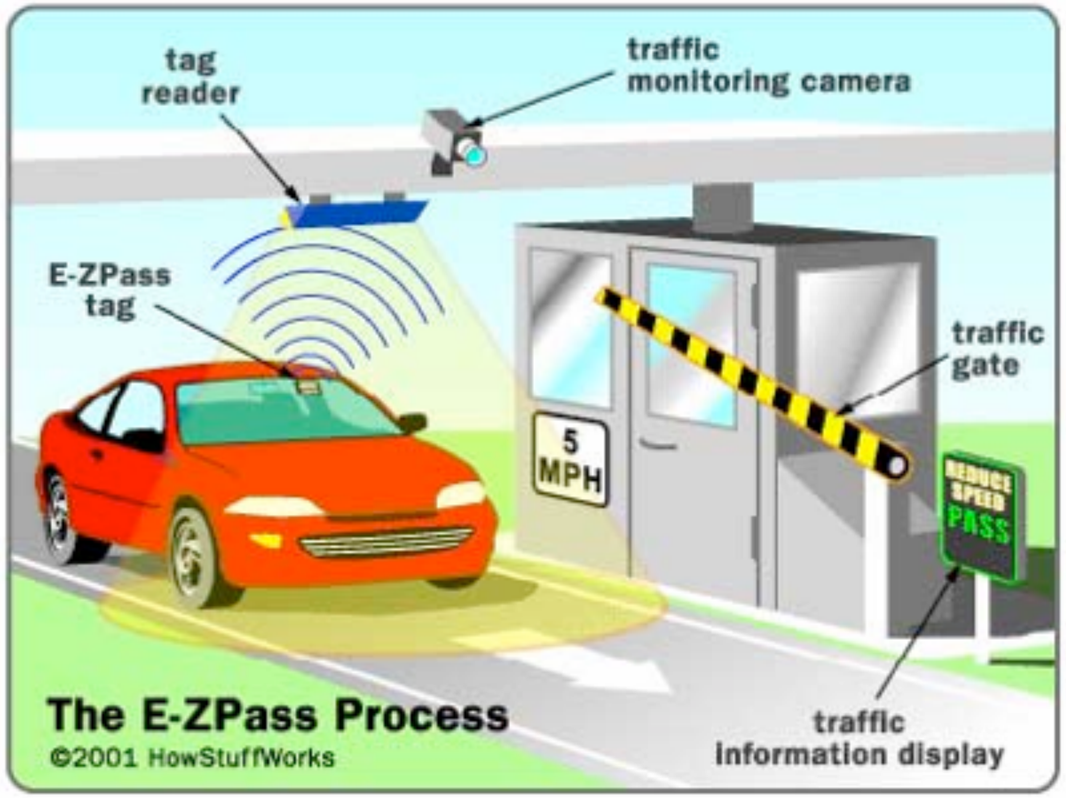
The E-ZPass Process

©2001 HowStuffWorks

This process leaves a lot to be desired in terms of flexibility:

- How do we charge more according to the time of day?

Motivation: how tolling works today



This process leaves a lot to be desired in terms of flexibility:

- How do we charge more according to the time of day?
- Or as drivers enter city centers?

Motivation: how tolling works today

Motivation: how tolling works today

Motivation: how tolling works today

Motivation: how tolling works today

Core tension between privacy and desire for more flexible toll pricing

Motivation: how tolling works today

Core tension between privacy and desire for more flexible toll pricing

- In this talk we'll see our system, Milo, which allows for fine-grained pricing policies without sacrificing drivers' privacy

Motivation: how tolling works today

Core tension between privacy and desire for more flexible toll pricing

- In this talk we'll see our system, Milo, which allows for fine-grained pricing policies without sacrificing drivers' privacy

- In the process, we strongly guarantee that drivers remain honest

Previous work [BKS05,BC06,TDKP07,dJJ08,...]

Previous work [BKS05,BC06,TDKP07,dJJ08,...]

USENIX Security 2009: VPriv [PBB]

Previous work [BKS05,BC06,TDKP07,dJJ08,...]

USENIX Security 2009: VPriv [PBB]

- Fine-grained policy: uses small road segments (where,when)

Previous work [BKS05,BC06,TDKP07,dJJ08,...]

USENIX Security 2009: VPriv [PBB]

- Fine-grained policy: uses small road segments (where,when)
- Privacy: uses Tor to maintain anonymity while driver uploads segments

Previous work [BKS05,BC06,TDKP07,dJJ08,...]

USENIX Security 2009: VPriv [PBB]

- Fine-grained policy: uses small road segments (where,when)
- Privacy: uses Tor to maintain anonymity while driver uploads segments
- Honesty: relies on audits wherein driver is asked to verify locations

Previous work [BKS05,BC06,TDKP07,dJJ08,...]

USENIX Security 2009: VPriv [PBB]

- Fine-grained policy: uses small road segments (where,when)
- Privacy: uses Tor to maintain anonymity while driver uploads segments
- Honesty: relies on audits wherein driver is asked to verify locations

USENIX Security 2010: PrETP [BRTPVG]

- Fine-grained policy: again uses small road segments
- Privacy: drivers commit to segments in a way that eliminates need for Tor
- Honesty: again relies on audits

Previous work [BKS05,BC06,TDKP07,dJJ08,...]

USENIX Security 2009: VPriv [PBB]

- Fine-grained policy: uses small road segments (where,when)
- Privacy: uses Tor to maintain anonymity while driver uploads segments
- Honesty: relies on audits wherein driver is asked to verify locations

USENIX Security 2010: PrETP [BRTPVG]

- Fine-grained policy: again uses small road segments
- Privacy: drivers commit to segments in a way that eliminates need for Tor
- Honesty: again relies on audits

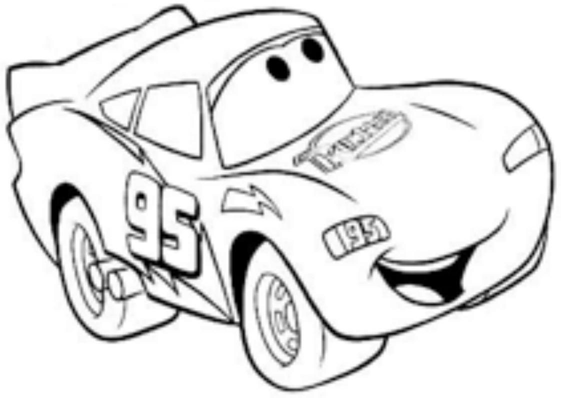
A potential problem: keeping colluding drivers honest

A potential problem: keeping colluding drivers honest

In these audits, we see a challenge/response behavior:

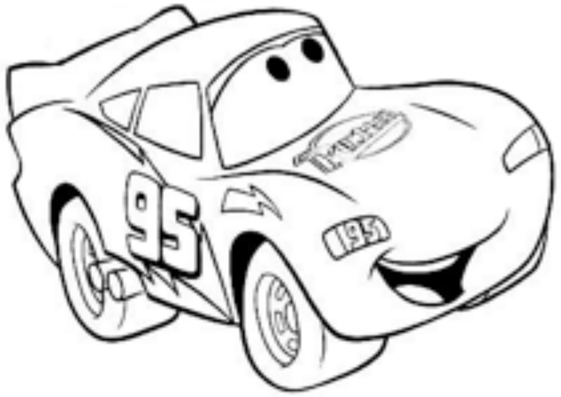
A potential problem: keeping colluding drivers honest

In these audits, we see a challenge/response behavior:



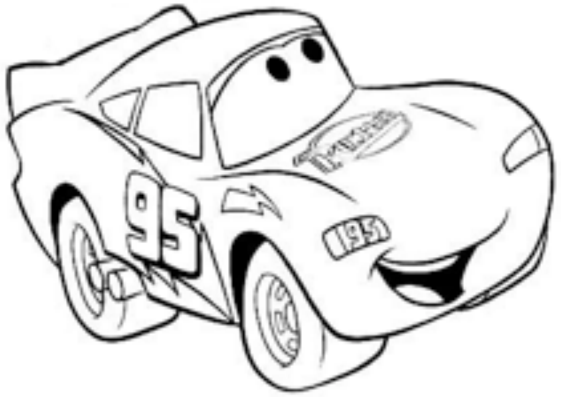
A potential problem: keeping colluding drivers honest

In these audits, we see a challenge/response behavior:



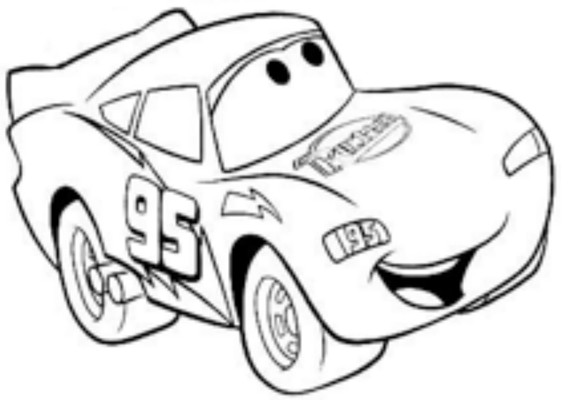
A potential problem: keeping colluding drivers honest

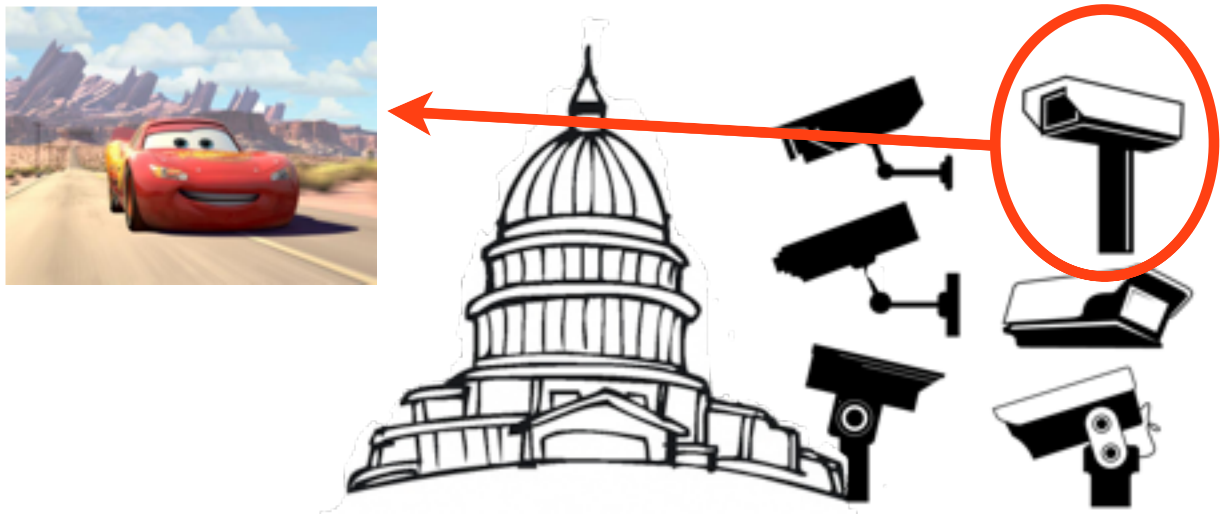
In these audits, we see a challenge/response behavior:



A potential problem: keeping colluding drivers honest

In these audits, we see a challenge/response behavior:



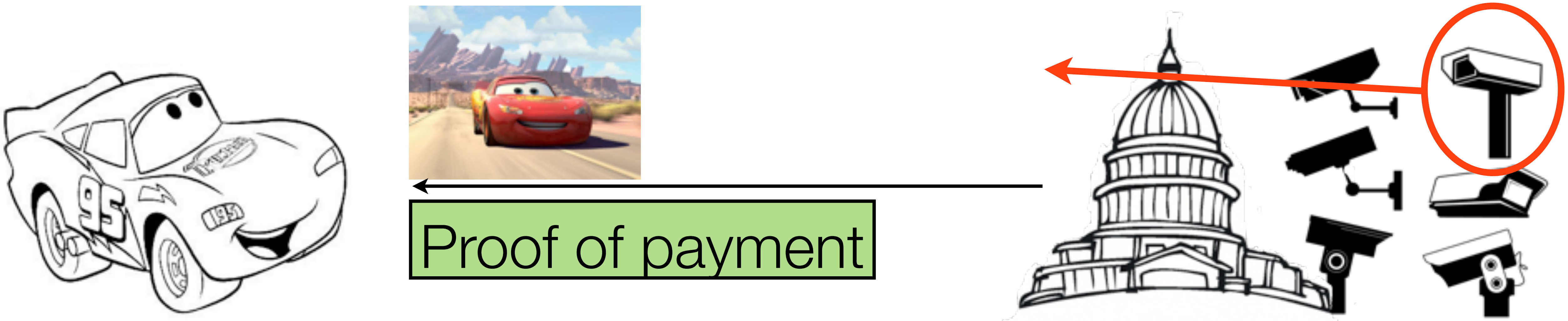


A potential problem: keeping colluding drivers honest

In these audits, we see a challenge/response behavior:

A potential problem: keeping colluding drivers honest

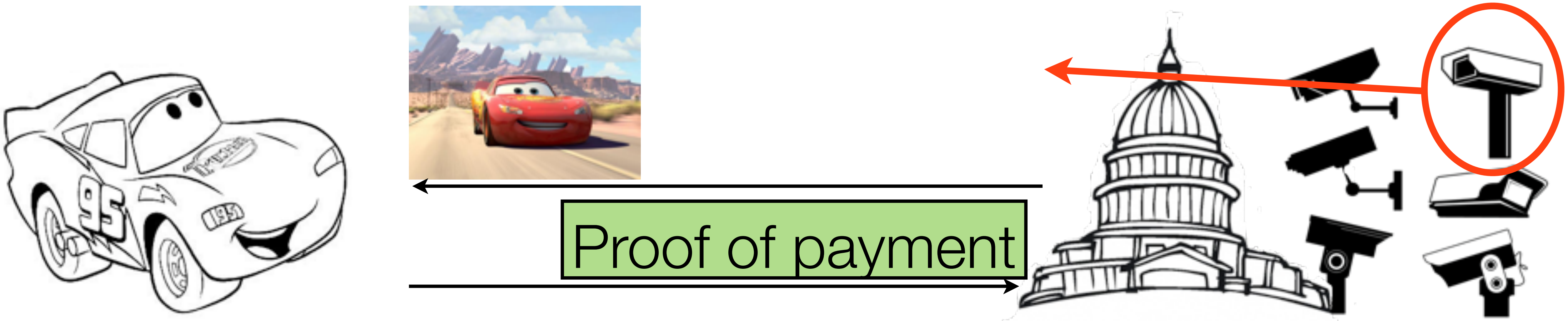
In these audits, we see a challenge/response behavior:



Proof of payment

A potential problem: keeping colluding drivers honest

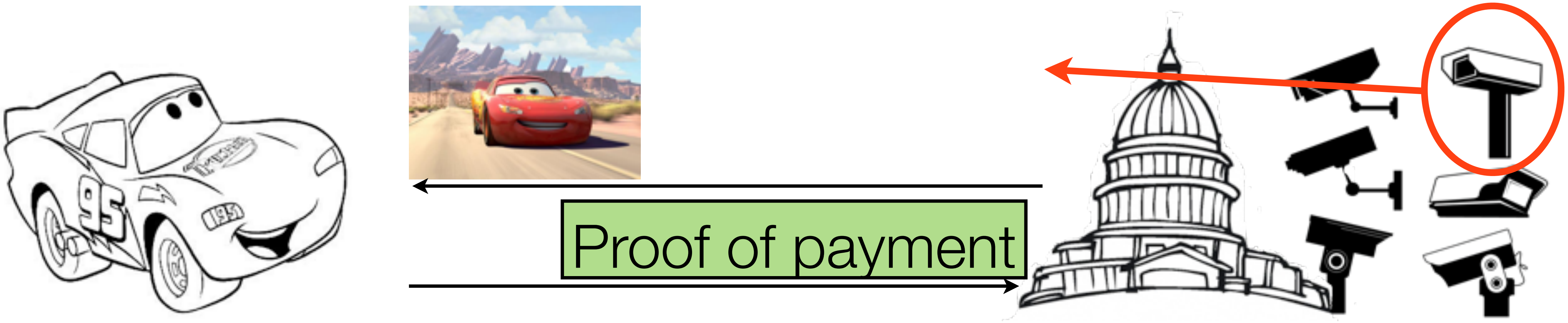
In these audits, we see a challenge/response behavior:



Proof of payment

A potential problem: keeping colluding drivers honest

In these audits, we see a challenge/response behavior:

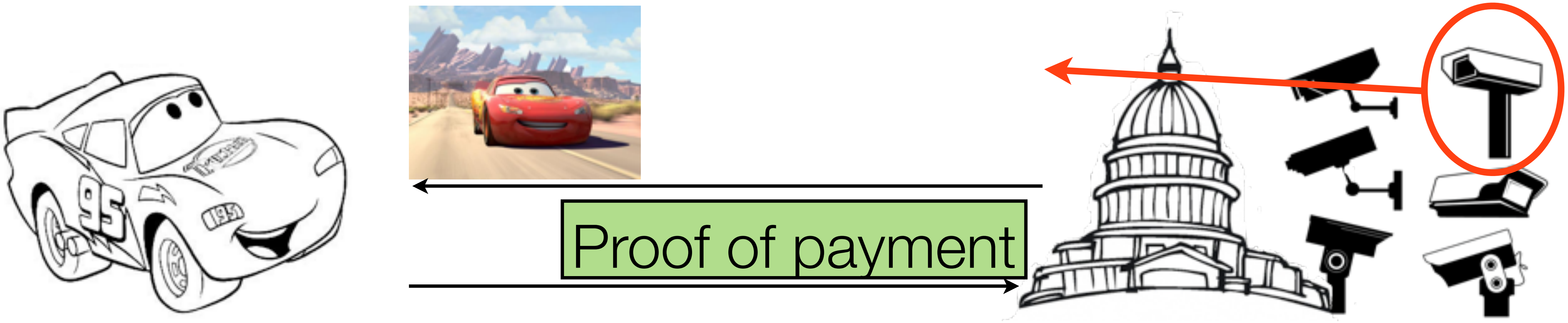


Proof of payment

So the authority reveals to the driver the segment in which he was seen! This information can then be shared to help drivers avoid cameras in the future

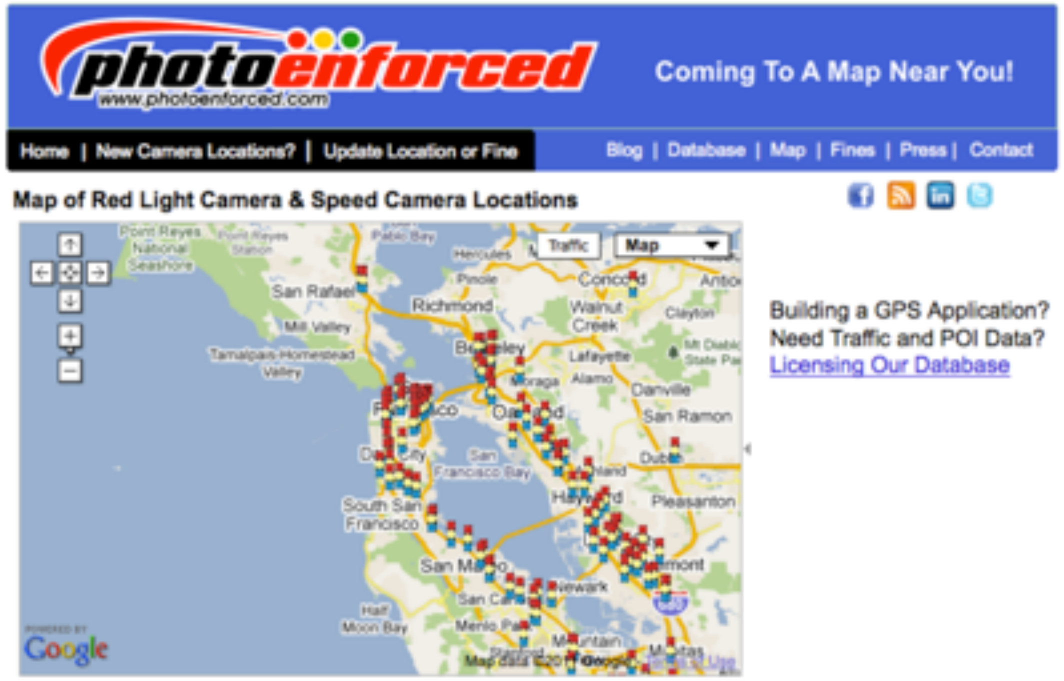
A potential problem: keeping colluding drivers honest

In these audits, we see a challenge/response behavior:



Proof of payment

So the authority reveals to the driver the segment in which he was seen! This information can then be shared to help drivers avoid cameras in the future



photoenforced
www.photoenforced.com

Coming To A Map Near You!

Home | New Camera Locations? | Update Location or Fine

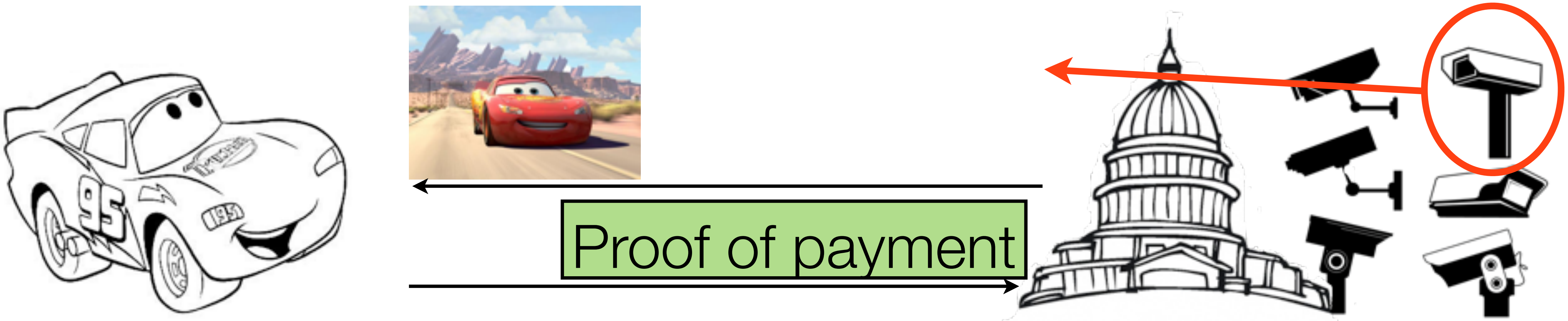
Blog | Database | Map | Fines | Press | Contact

Map of Red Light Camera & Speed Camera Locations

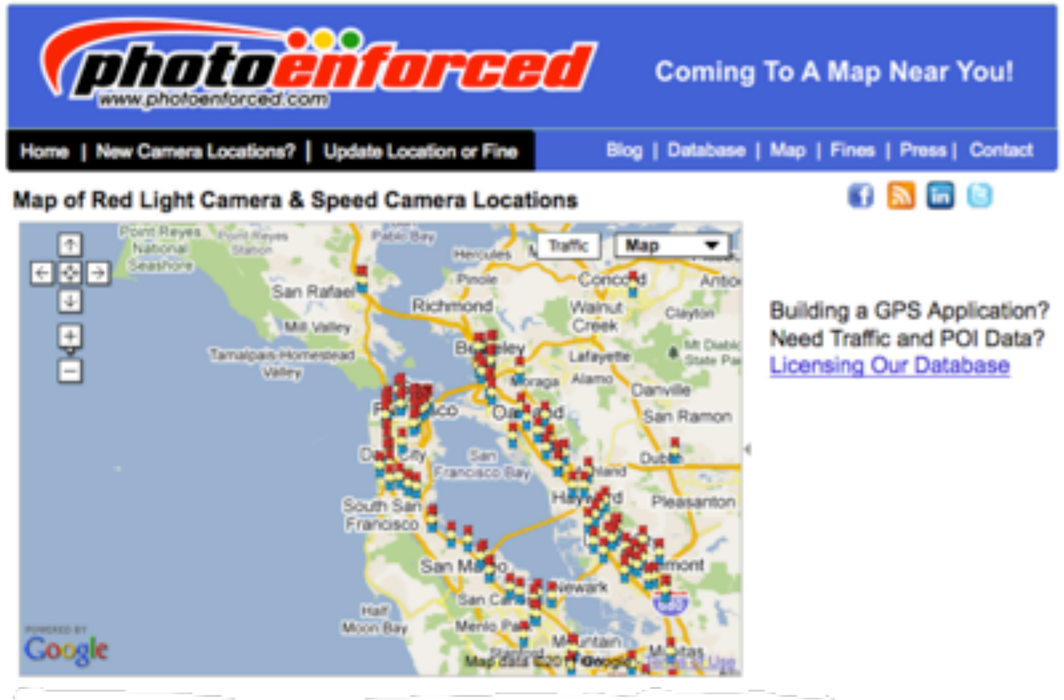
Building a GPS Application?
Need Traffic and POI Data?
Licensing Our Database

A potential problem: keeping colluding drivers honest

In these audits, we see a challenge/response behavior:



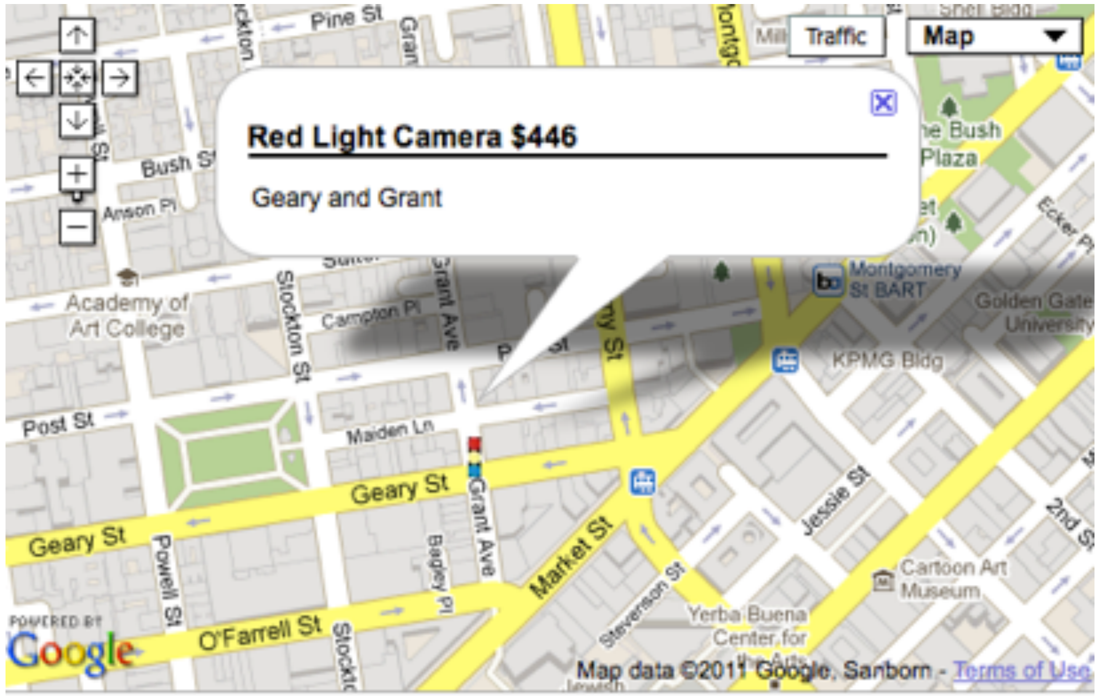
So the authority reveals to the driver the segment in which he was seen! This information can then be shared to help drivers avoid cameras in the future


photoenforced
www.photoenforced.com
Coming To A Map Near You!

Home | New Camera Locations? | Update Location or Fine
Blog | Database | Map | Fines | Press | Contact

Map of Red Light Camera & Speed Camera Locations

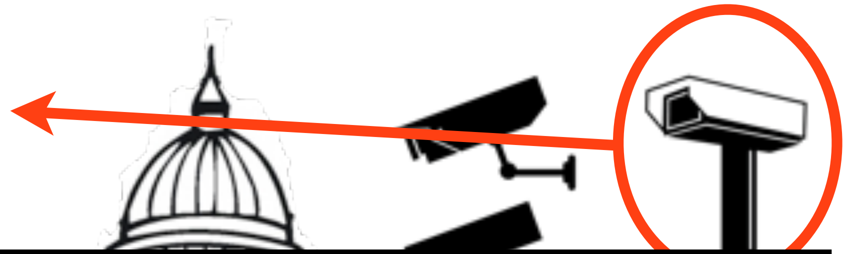
Building a GPS Application? Need Traffic and POI Data? Licensing Our Database


Red Light Camera $446

Geary and Grant

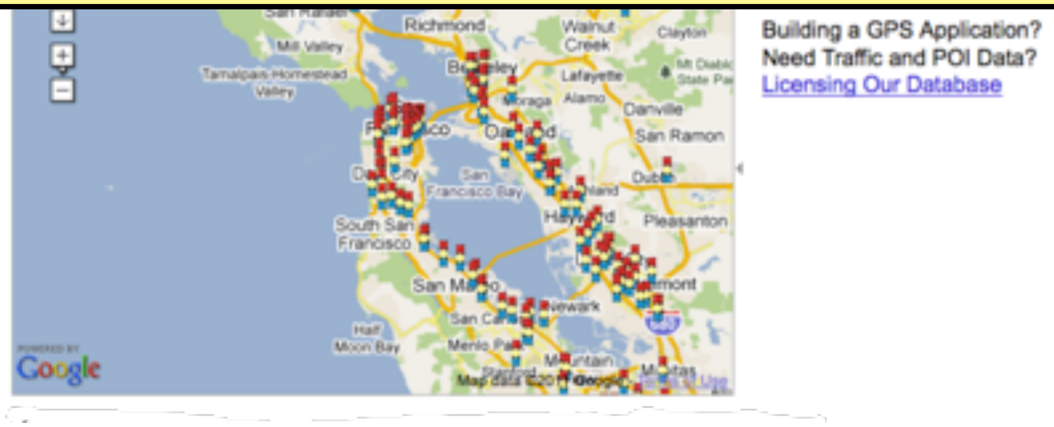
A potential problem: keeping colluding drivers honest

In these audits, we see a challenge/response behavior:



USENIX Security 2011: Milo

- Fine-grained policy: uses same small road segments (where,when)
- Privacy: drivers commit to segments in a way similar to PrETP
- Honesty: audit protocol no longer reveals locations to drivers



Building a GPS Application?
Need Traffic and POI Data?
Licensing Our Database

Outline

Outline

Cryptographic background

Outline

Cryptographic background

Milo

Outline

Cryptographic background

Milo

Evaluation

Outline

Cryptographic background

Milo

Evaluation

Conclusions

Outline

Cryptographic background

Commitment schemes

Zero-knowledge proofs

Blind identity-based encryption

Milo

Evaluation

Conclusions

Commitments [BCC88,P91]

Commitments [BCC88,P91]

Commitments [BCC88,P91]

My favorite number is 42

Commitments [BCC88,P91]

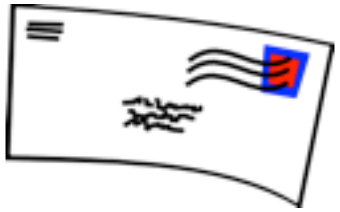
My favorite number is 42

42

Commitments [BCC88,P91]

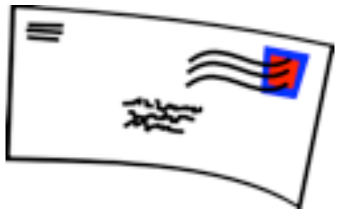
My favorite number is 42

42

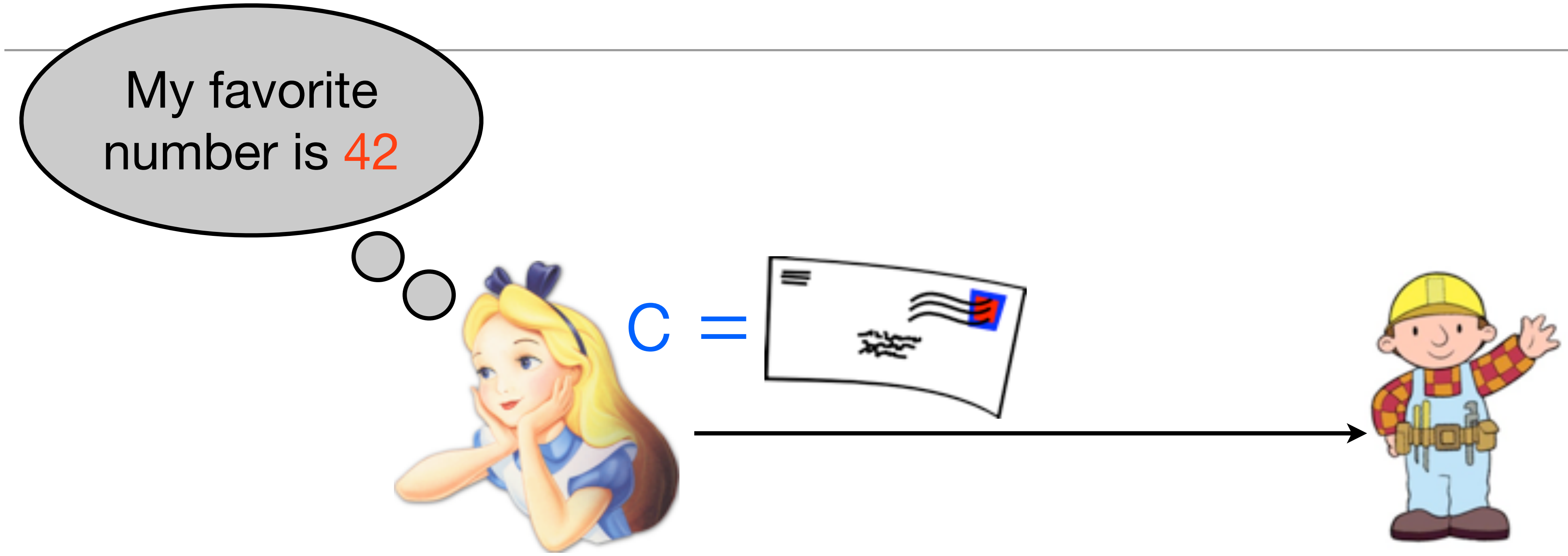
c = 

Commitments [BCC88,P91]

My favorite number is 42

C = 

Commitments [BCC88,P91]



My favorite number is 42

C =

Commitments [BCC88,P91]

My favorite number is 42

C =

Commitments [BCC88,P91]

My favorite number is 42

c =

Open(c)

Commitments [BCC88,P91]

My favorite number is 42

42

c =

Open(c)

Commitments [BCC88,P91]

My favorite number is 42

42

c =

Open(c)

There are two important properties of commitments:

Commitments [BCC88,P91]

My favorite number is 42

42

c =

Open(c)

There are two important properties of commitments:

- Hiding: Bob didn't know the value in c until Alice gave him Open(c)

Commitments [BCC88,P91]

My favorite number is 42

42

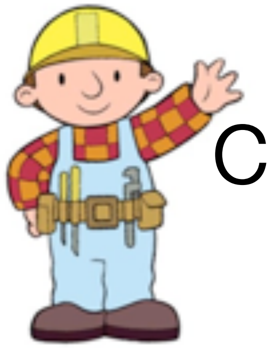
c =

Open(c)

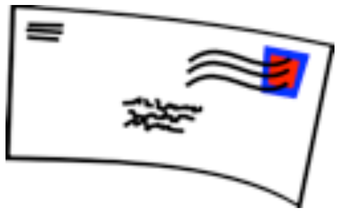
There are two important properties of commitments:

- Hiding: Bob didn’t know the value in c until Alice gave him Open(c)
- Binding: Alice couldn’t change the value in c after giving Bob the envelope

Zero-knowledge proofs [GMR89,BdSMP91]

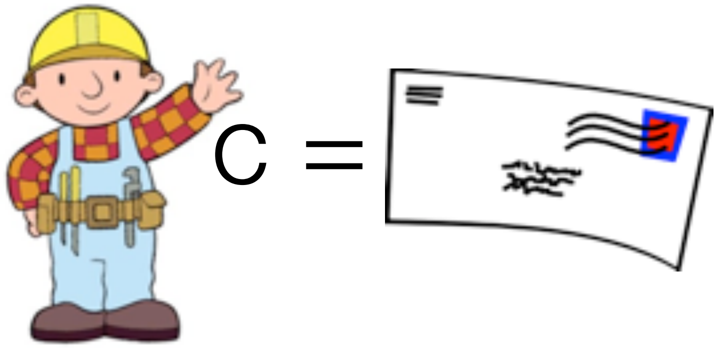


c =



Zero-knowledge proofs [GMR89,BdSMP91]

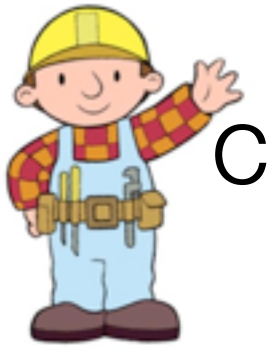
The value in
c is between 0
and 100

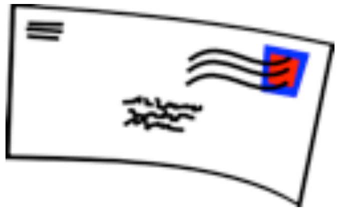


Zero-knowledge proofs [GMR89,BdSMP91]

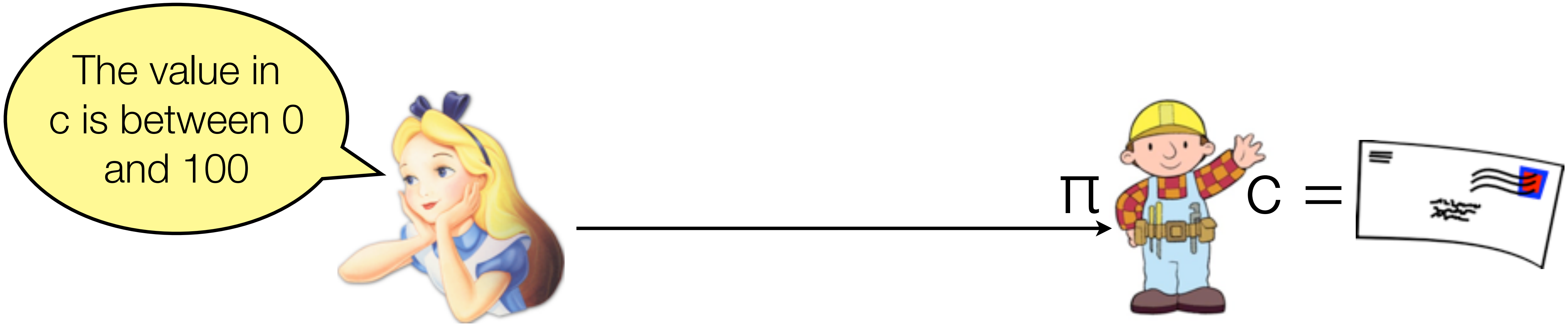
The value in c is between 0 and 100

π



$c =$ 

Zero-knowledge proofs [GMR89,BdSMP91]

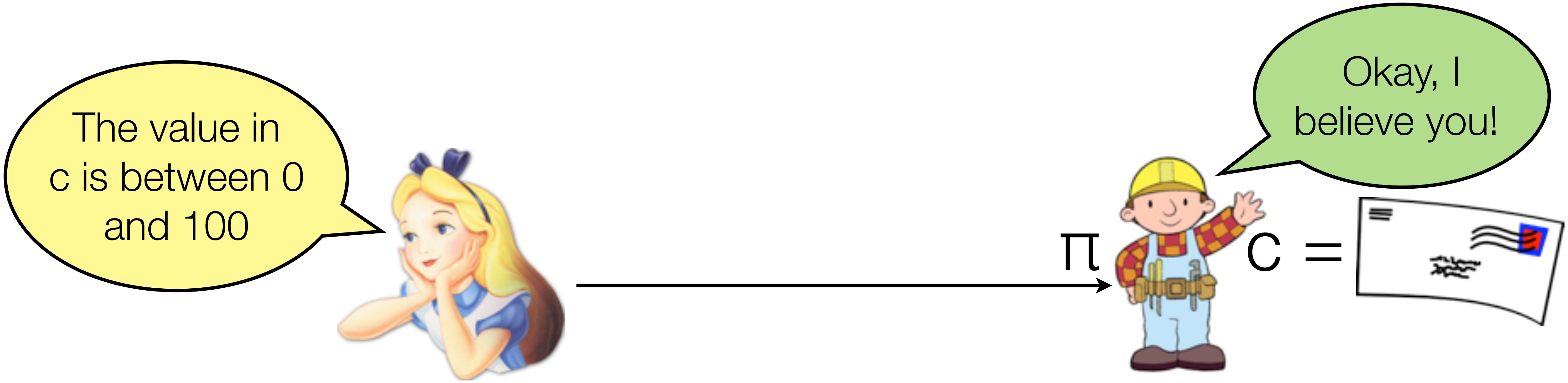


The value in c is between 0 and 100

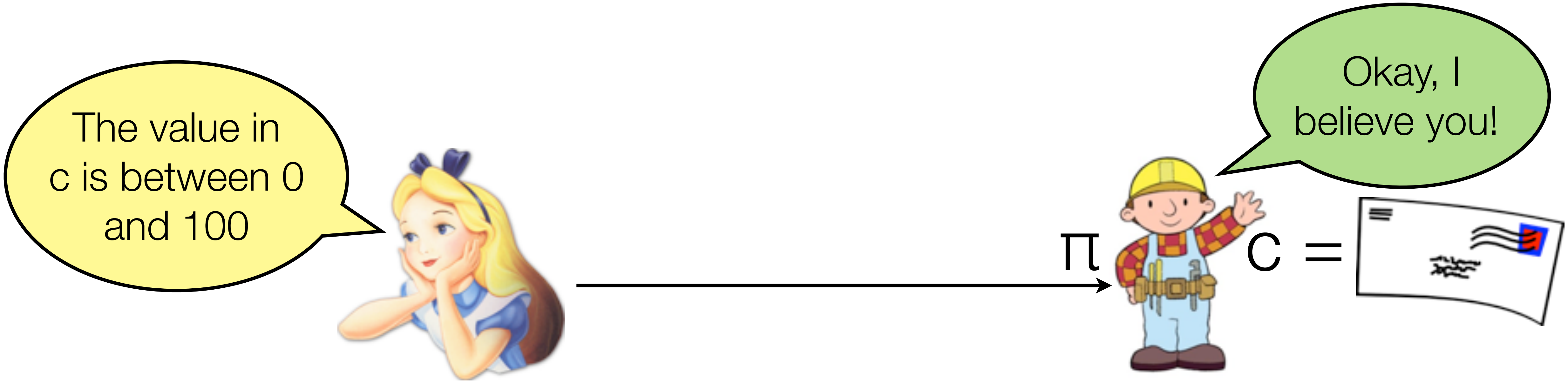
π

c =

Zero-knowledge proofs [GMR89,BdSMP91]



Zero-knowledge proofs [GMR89,BdSMP91]



The value in c is between 0 and 100

Okay, I believe you!

There are two important properties of zero-knowledge proofs:

Zero-knowledge proofs [GMR89,BdSMP91]

The value in c is between 0 and 100

π

Okay, I believe you!

$c =$

There are two important properties of zero-knowledge proofs:

- Soundness: Alice can't convince Bob of something that isn't true

Zero-knowledge proofs [GMR89,BdSMP91]

The value in c is between 0 and 100
π
Okay, I believe you!
c =

There are two important properties of zero-knowledge proofs:

- Soundness: Alice can't convince Bob of something that isn't true
- Zero knowledge: Bob doesn't learn anything about Alice's exact number

Zero-knowledge proofs [GMR89,BdSMP91]

The value in c is between 0 and 100
π
Okay, I believe you!
c =

There are two important properties of zero-knowledge proofs:

- Soundness: Alice can’t convince Bob of something that isn’t true
- Zero knowledge: Bob doesn’t learn anything about Alice’s exact number

Zero-knowledge proofs are much more general than this, but this range proof is the only type we will need

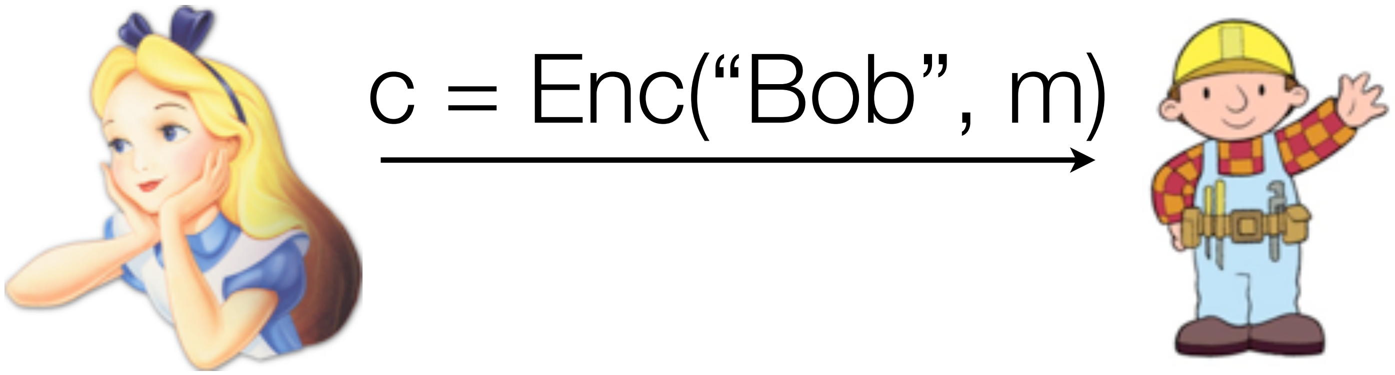
Blind identity-based encryption (IBE)

Blind identity-based encryption (IBE)

Regular [S84,BF01,C01]:

Blind identity-based encryption (IBE)

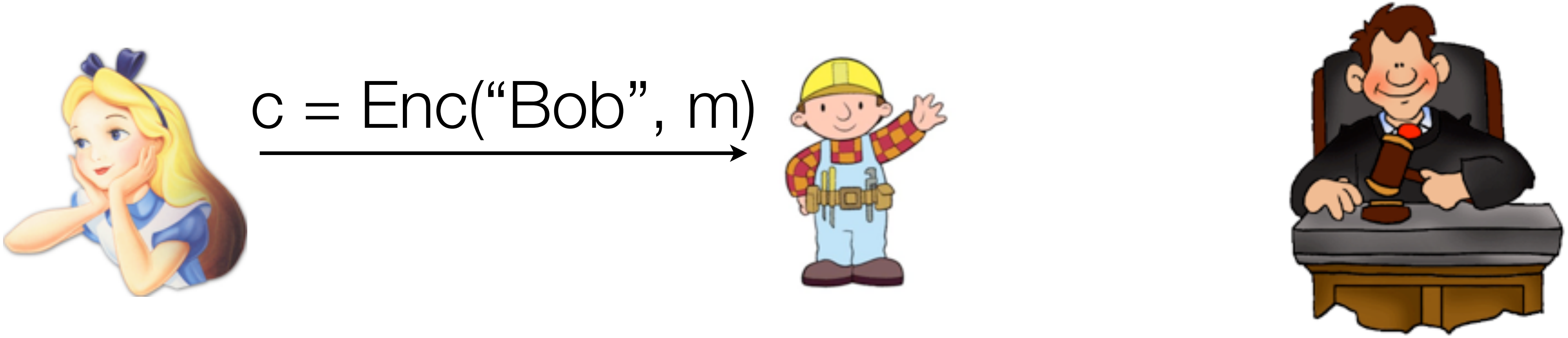
Regular [S84,BF01,C01]:



c = Enc(“Bob”, m)

Blind identity-based encryption (IBE)

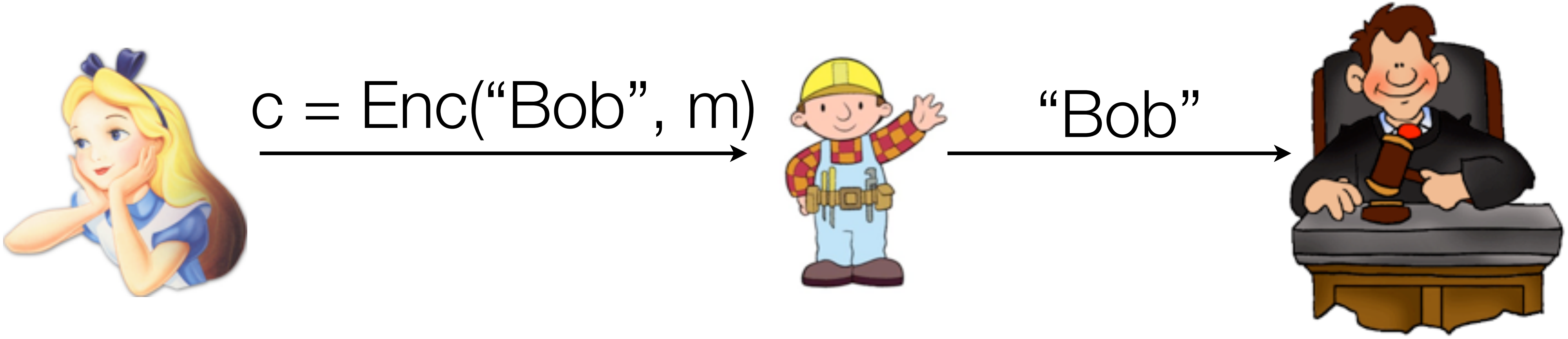
Regular [S84,BF01,C01]:



c = Enc(“Bob”, m)

Blind identity-based encryption (IBE)

Regular [S84,BF01,C01]:

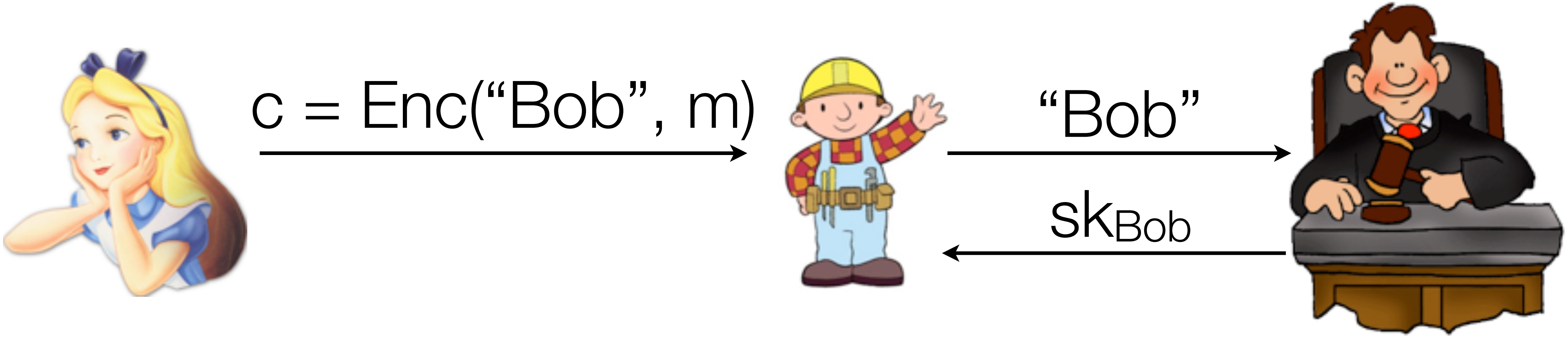


c = Enc(“Bob”, m)

“Bob”

Blind identity-based encryption (IBE)

Regular [S84,BF01,C01]:



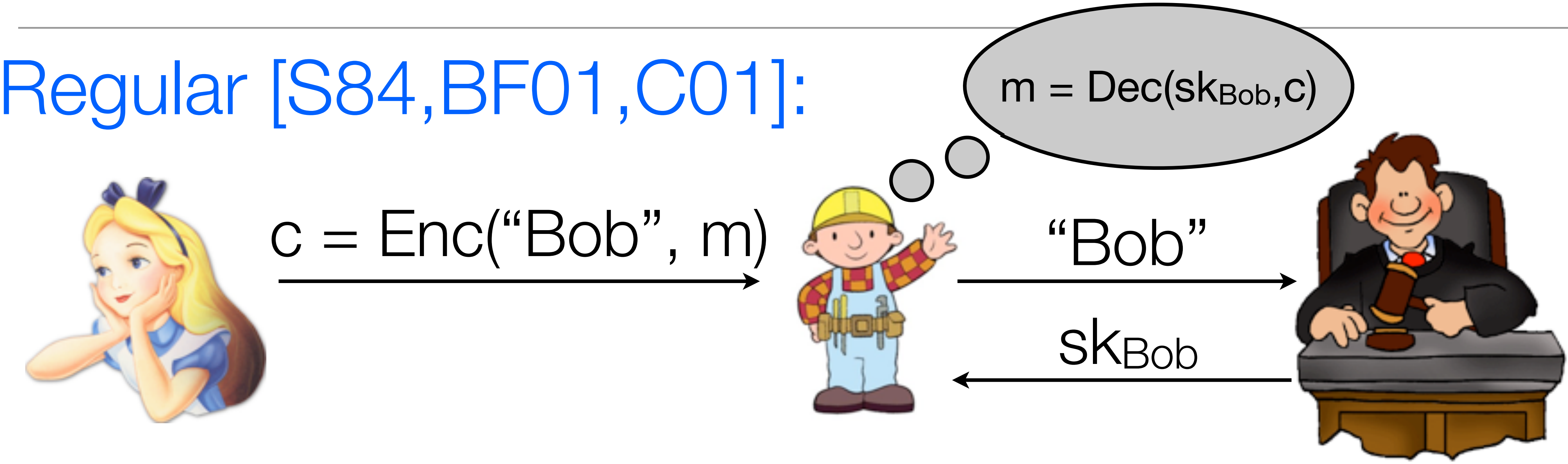
c = Enc(“Bob”, m)

“Bob”

sk_{Bob}

Blind identity-based encryption (IBE)

Regular [S84,BF01,C01]:



$m = \mathrm{Dec}(sk_{Bob}, c)$

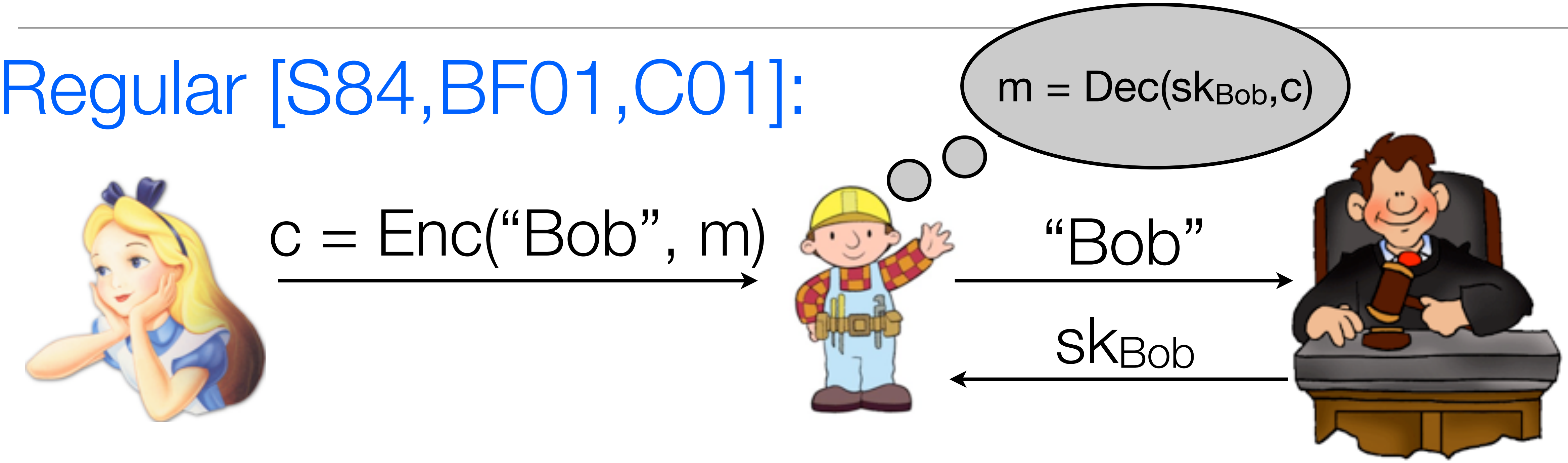
$c = \mathrm{Enc}(\text{“Bob”}, m)$

“Bob”

sk_{Bob}

Blind identity-based encryption (IBE)

Regular [S84,BF01,C01]:



$m = \mathrm{Dec}(sk_{Bob}, c)$

$c = \mathrm{Enc}(\text{“Bob”}, m)$

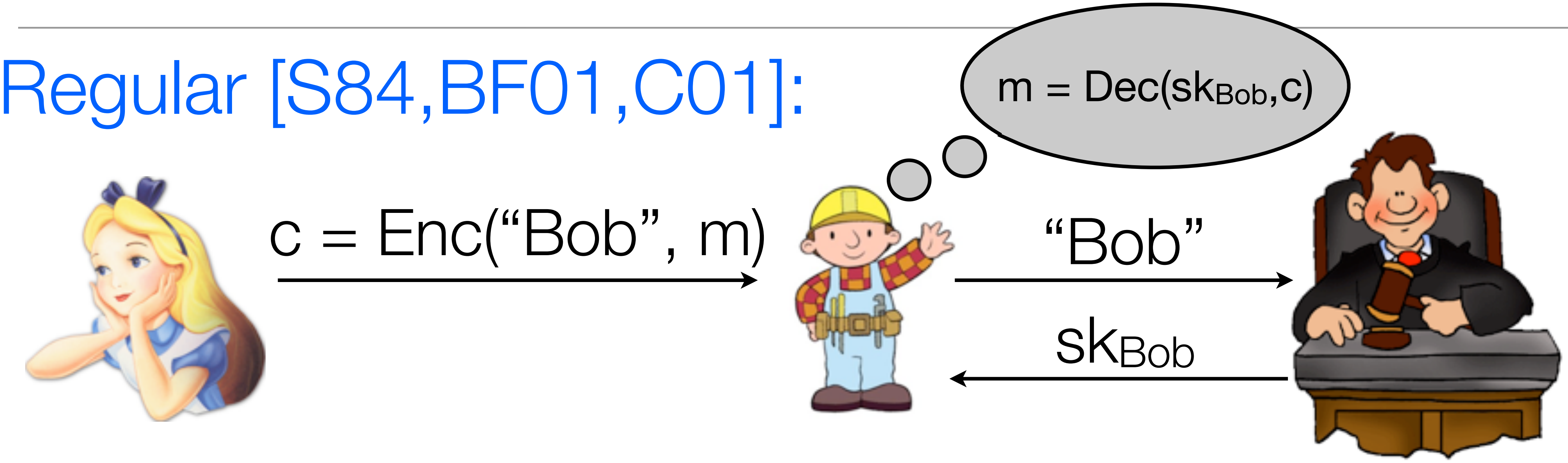
“Bob”

sk_{Bob}

Blind [GH07]:

Blind identity-based encryption (IBE)

Regular [S84,BF01,C01]:



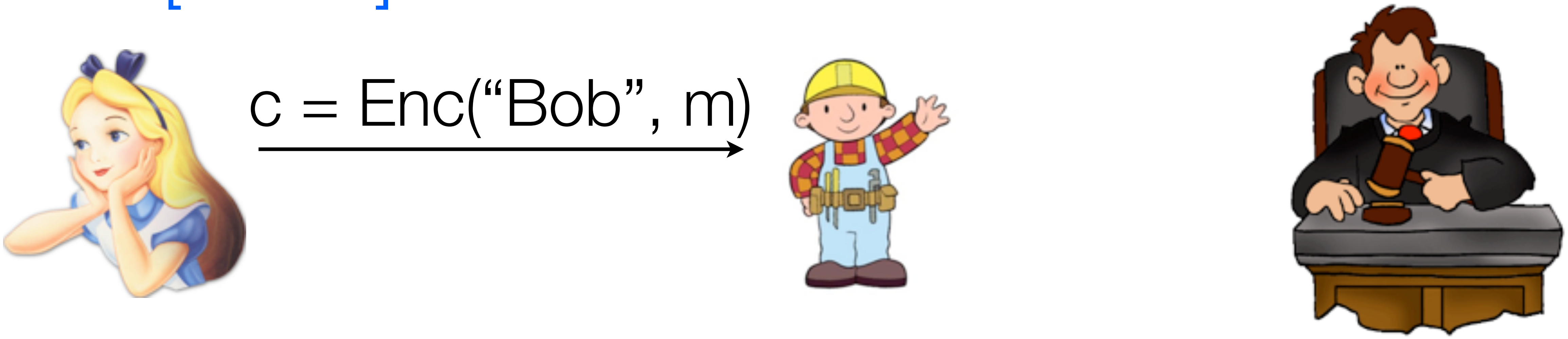
m = Dec(sk_{Bob},c)

c = Enc(“Bob”, m)

“Bob”

sk_{Bob}

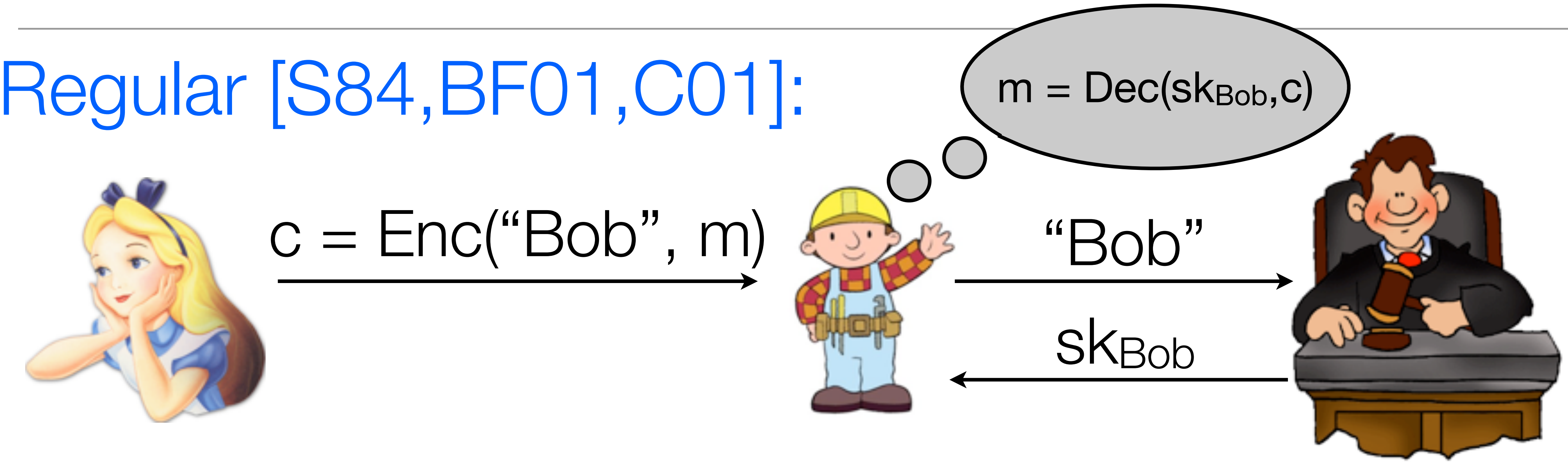
Blind [GH07]:



c = Enc(“Bob”, m)

Blind identity-based encryption (IBE)

Regular [S84,BF01,C01]:



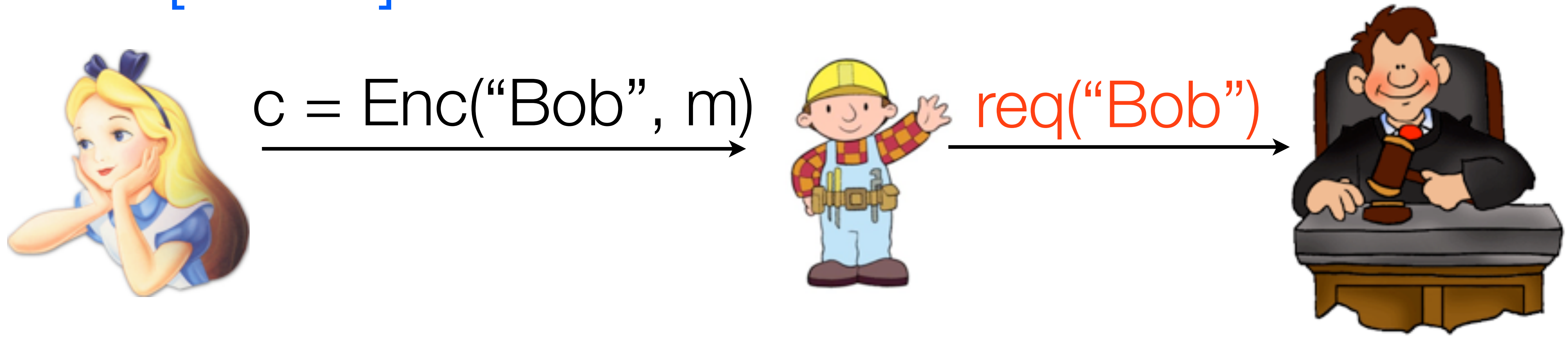
$m = \mathrm{Dec}(sk_{Bob}, c)$

$c = \mathrm{Enc}(\text{“Bob”}, m)$

“Bob”

sk_{Bob}

Blind [GH07]:

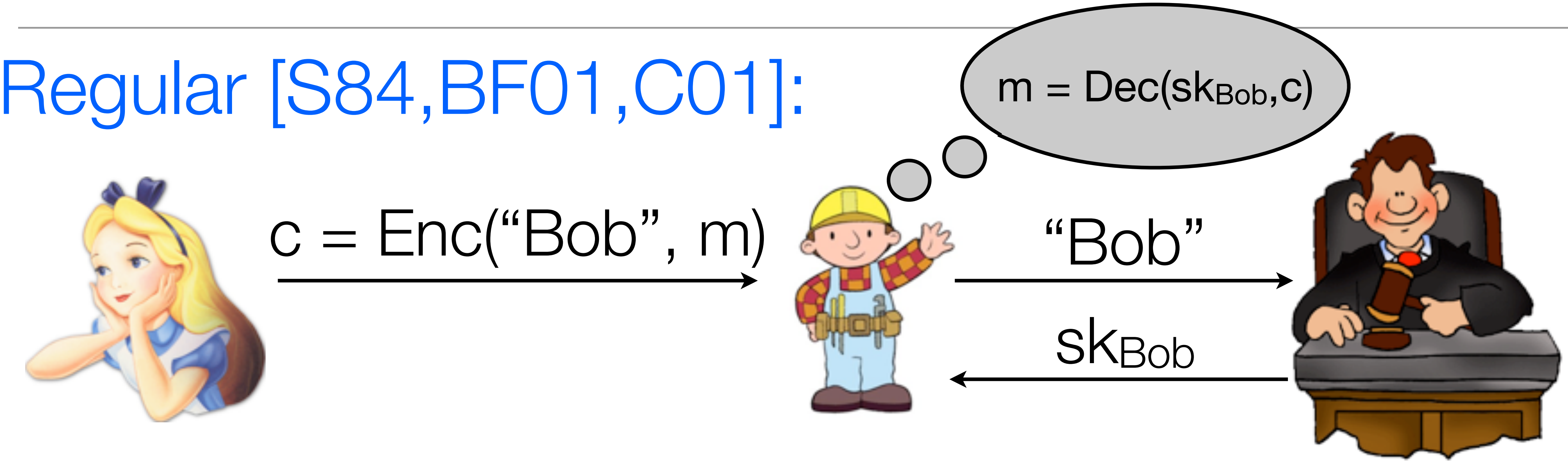


$c = \mathrm{Enc}(\text{“Bob”}, m)$

req(“Bob”)

Blind identity-based encryption (IBE)

Regular [S84,BF01,C01]:



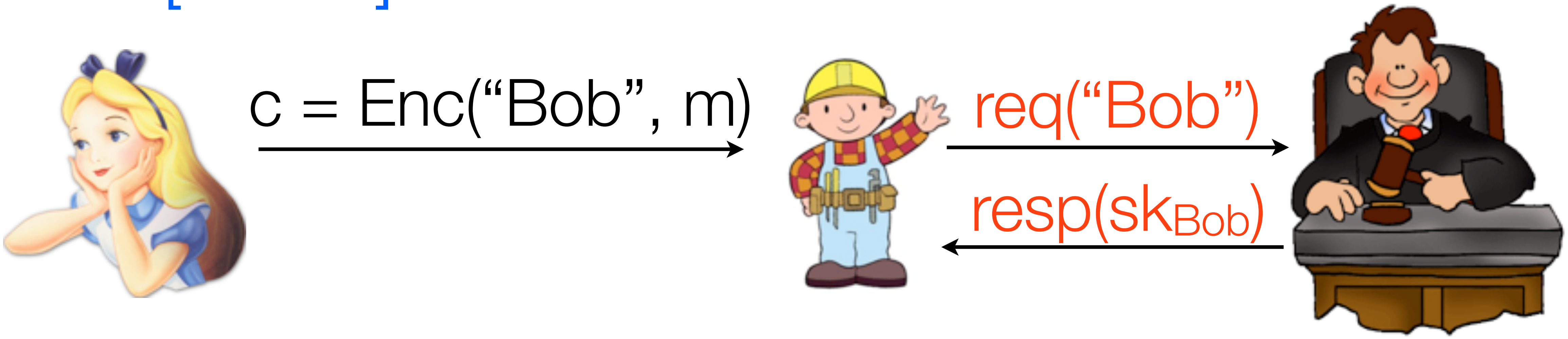
$m = \mathrm{Dec}(sk_{Bob}, c)$

c = Enc(“Bob”, m)

“Bob”

sk_{Bob}

Blind [GH07]:



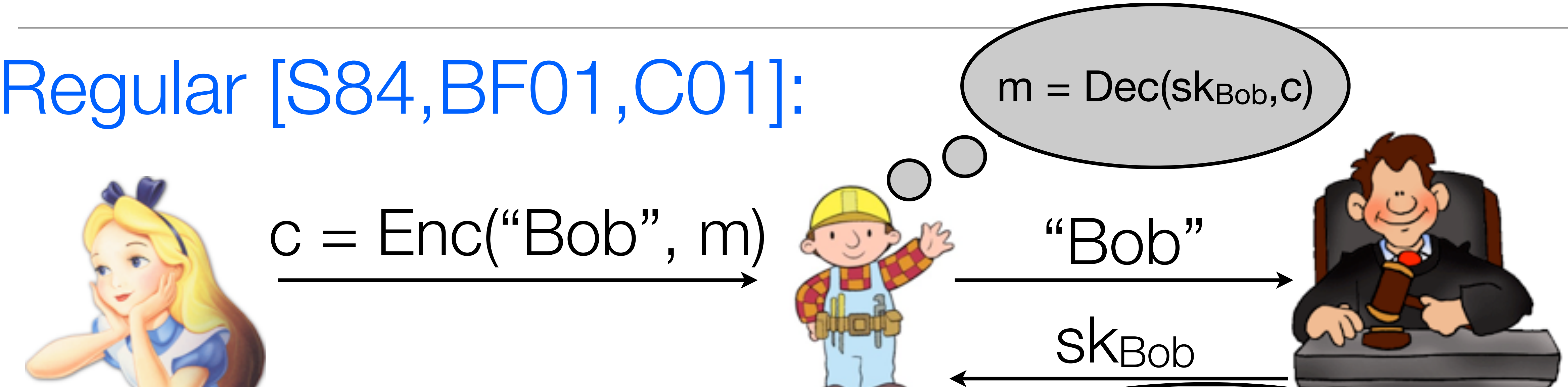
c = Enc(“Bob”, m)

req(“Bob”)

resp(sk_{Bob})

Blind identity-based encryption (IBE)

Regular [S84,BF01,C01]:



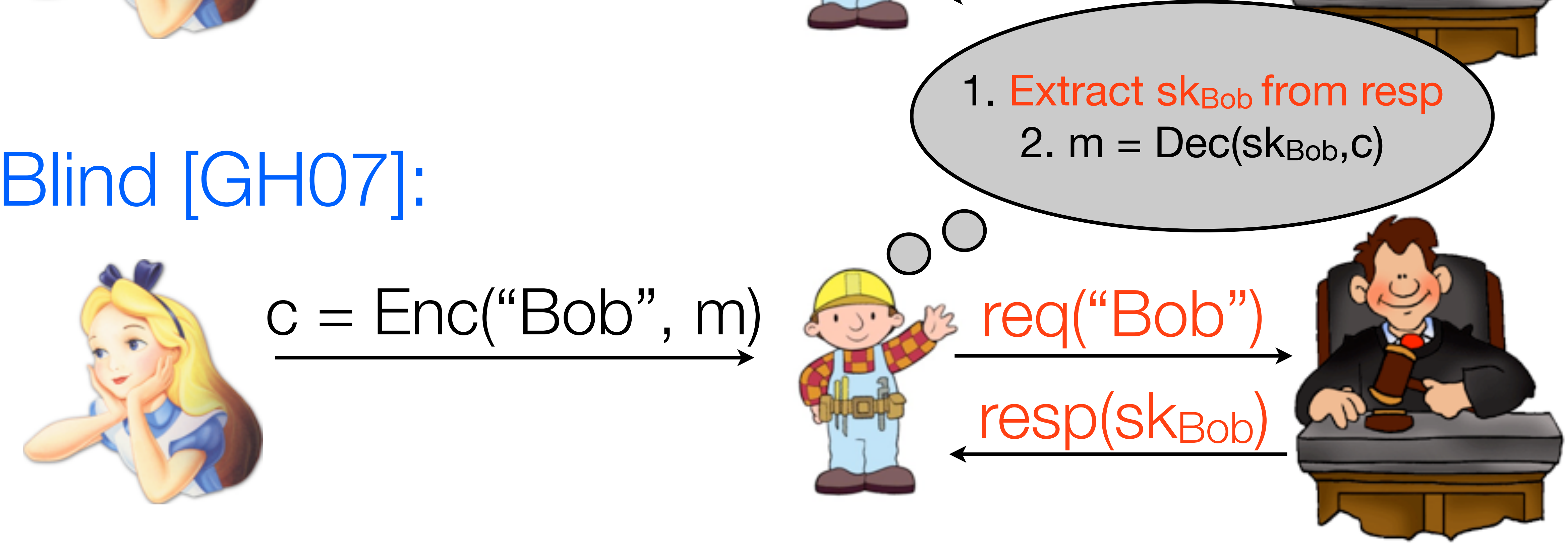
$m = \mathrm{Dec}(sk_{Bob}, c)$

$c = \mathrm{Enc}(\text{“Bob”}, m)$

“Bob”

sk_{Bob}

Blind [GH07]:



1. Extract sk_{Bob} from resp
2. $m = \mathrm{Dec}(sk_{Bob}, c)$

$c = \mathrm{Enc}(\text{“Bob”}, m)$

req(“Bob”)

resp(sk_{Bob})

Blind identity-based encryption (IBE)

Regular [S84,BF01,C01]:

c = Enc("Bob", m) →

m = Dec(sk_{Bob},c)

"Bob" →

← sk_{Bob}

Blind [GH07]:

1. Extract sk_{Bob} from resp
2. m = Dec(sk_{Bob},c)

c = Enc("Bob", m) →

req("Bob") →

← resp(sk_{Bob})

So the authority doesn't learn which key is being extracted

Outline

Cryptographic background

Milo

A generic toll collection system

A look back at (adapted) PrETP

A new Audit protocol

Evaluation

Conclusions

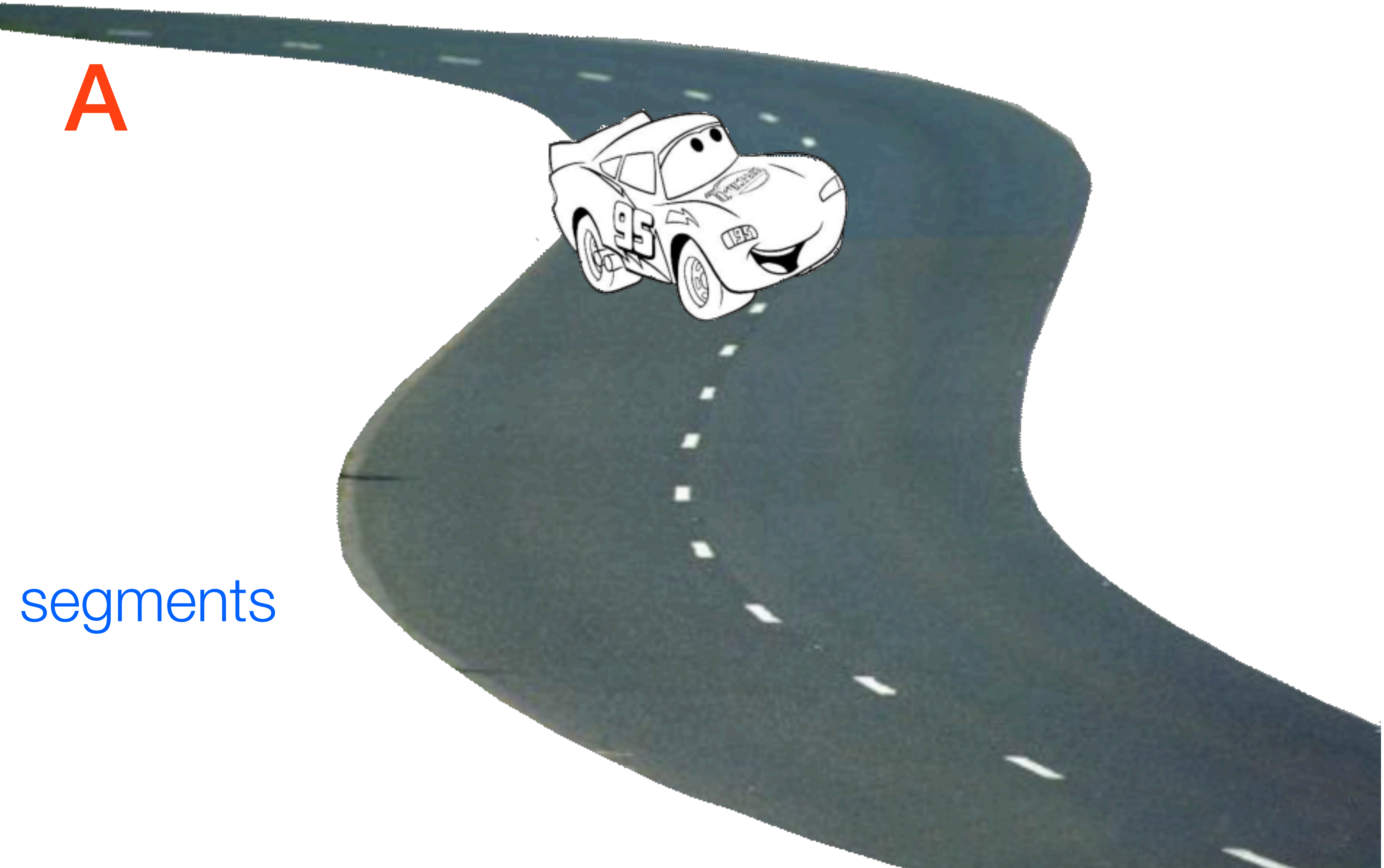
privacy-preserving toll pricing works

privacy-preserving toll pricing works

segments

y privacy-preserving toll pricing works

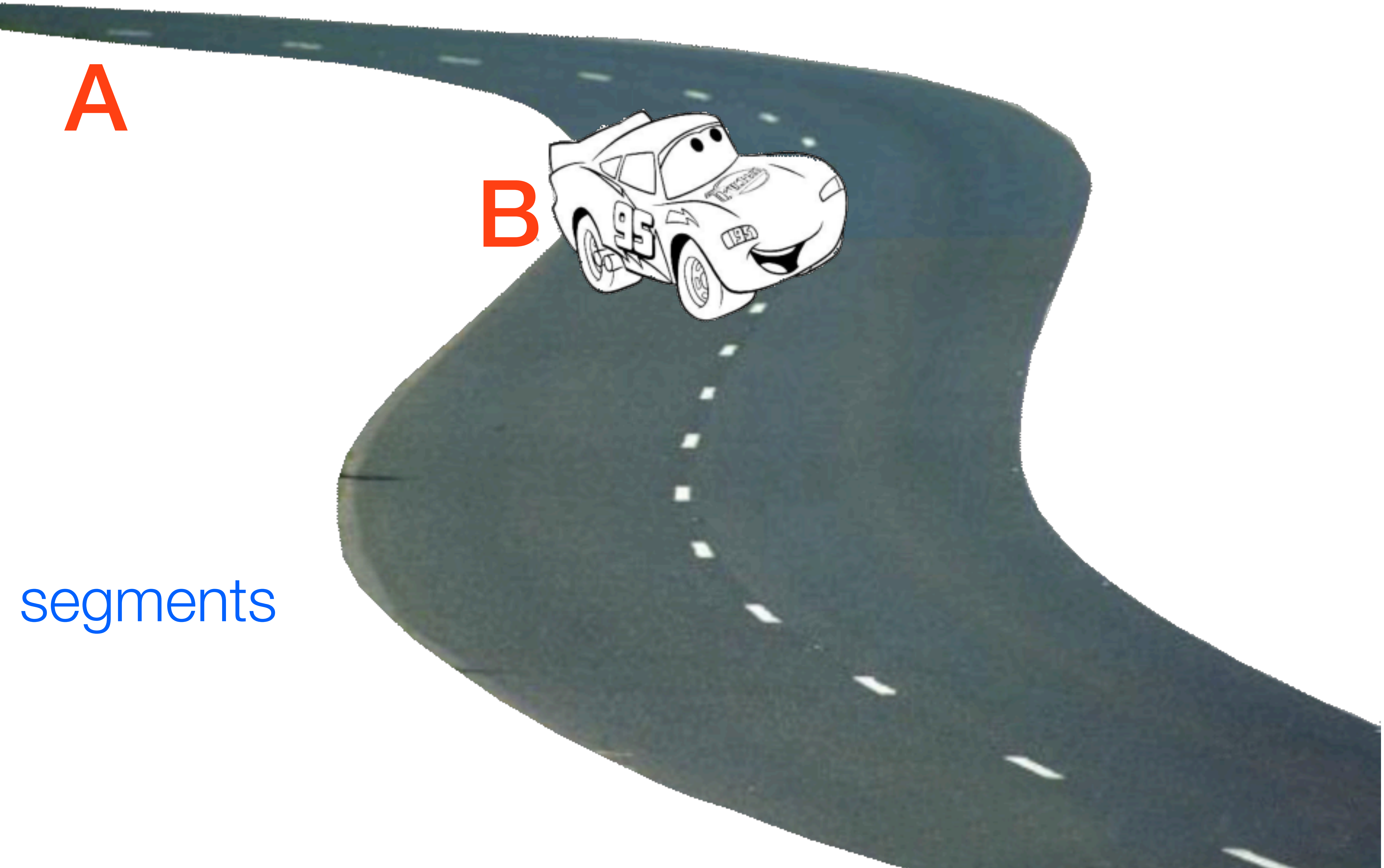
How privacy-preserving toll pricing works



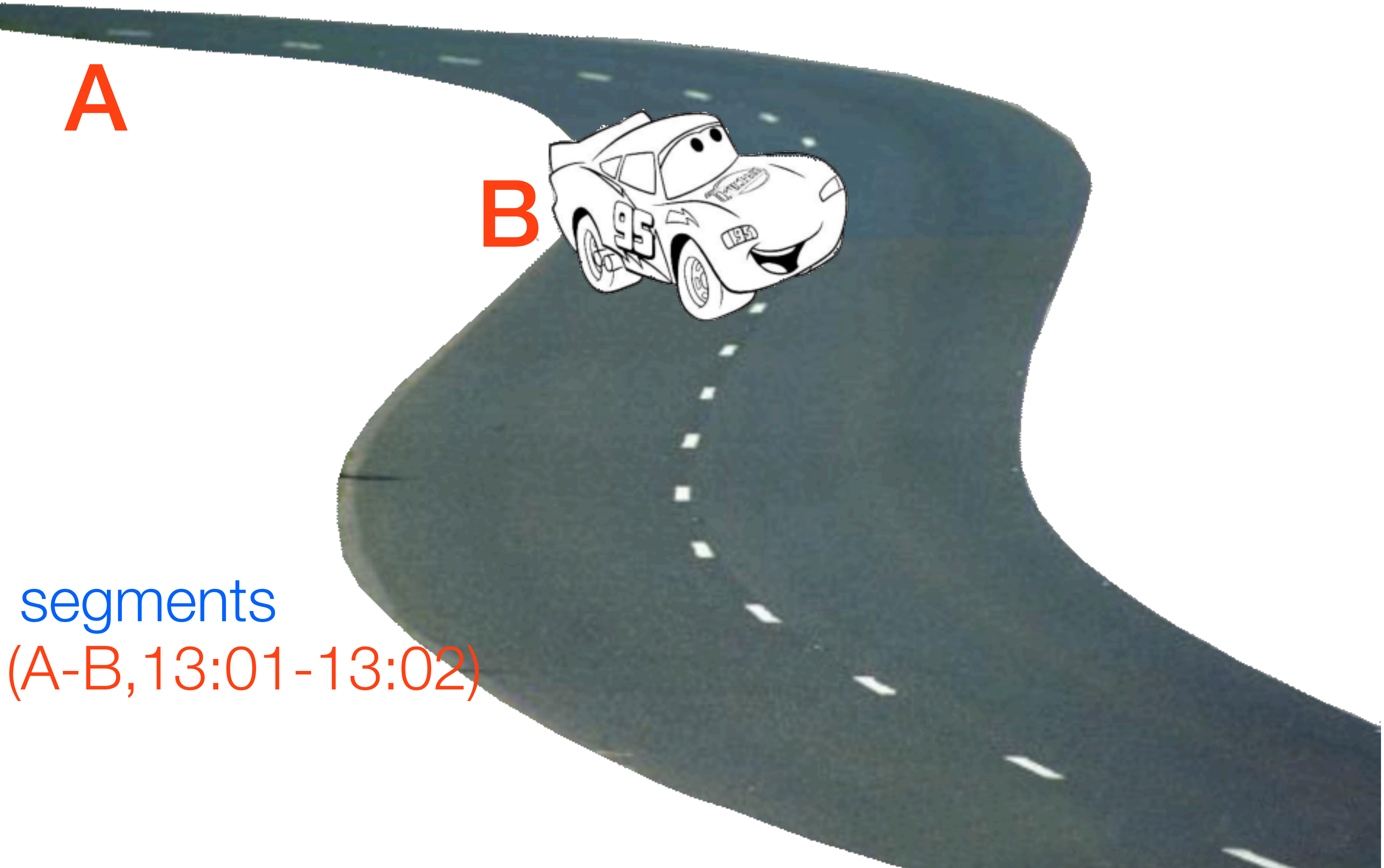
A

segments

How privacy-preserving toll pricing works



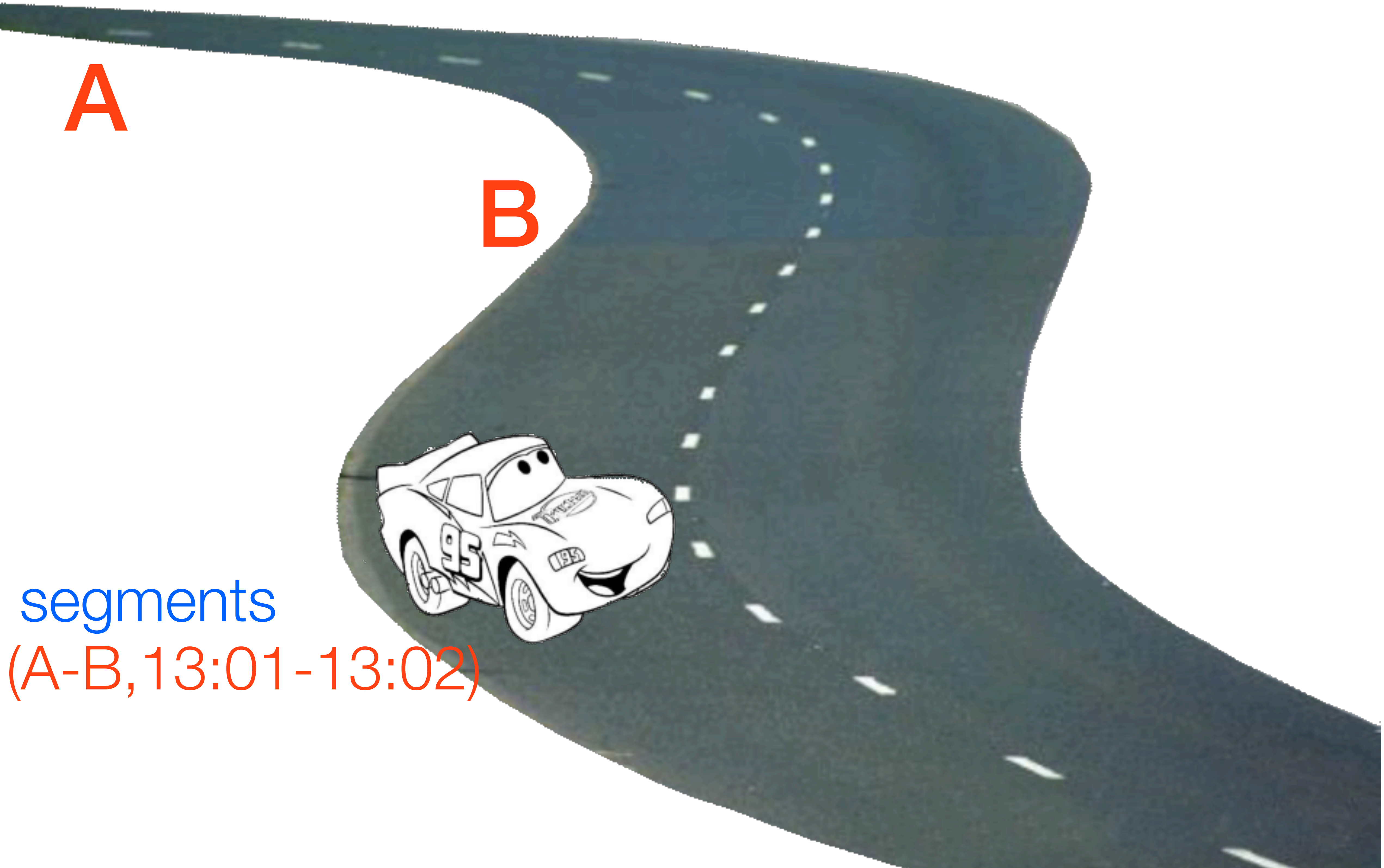
How privacy-preserving toll pricing works



segments

(A-B,13:01-13:02)

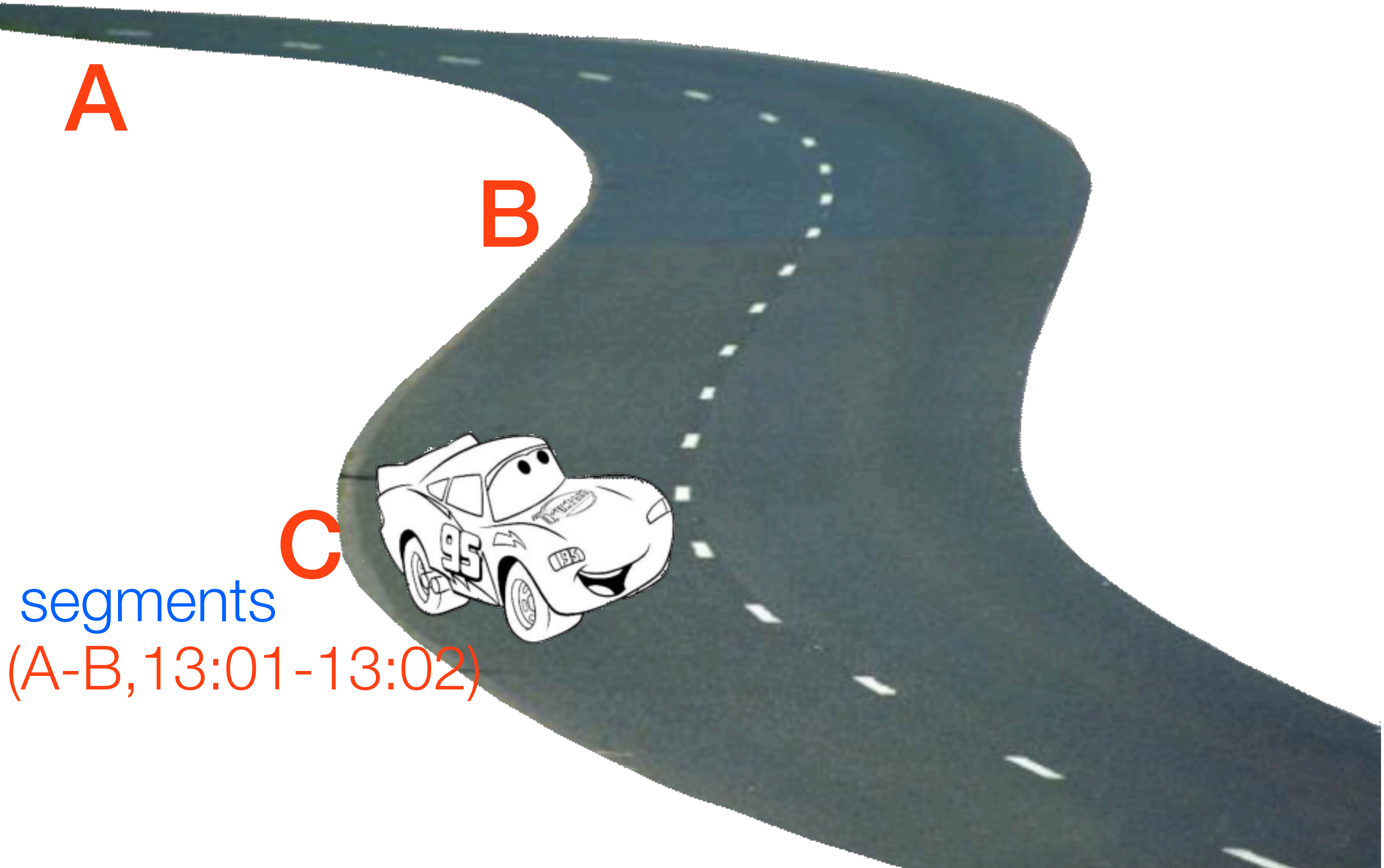
How privacy-preserving toll pricing works



segments

(A-B,13:01-13:02)

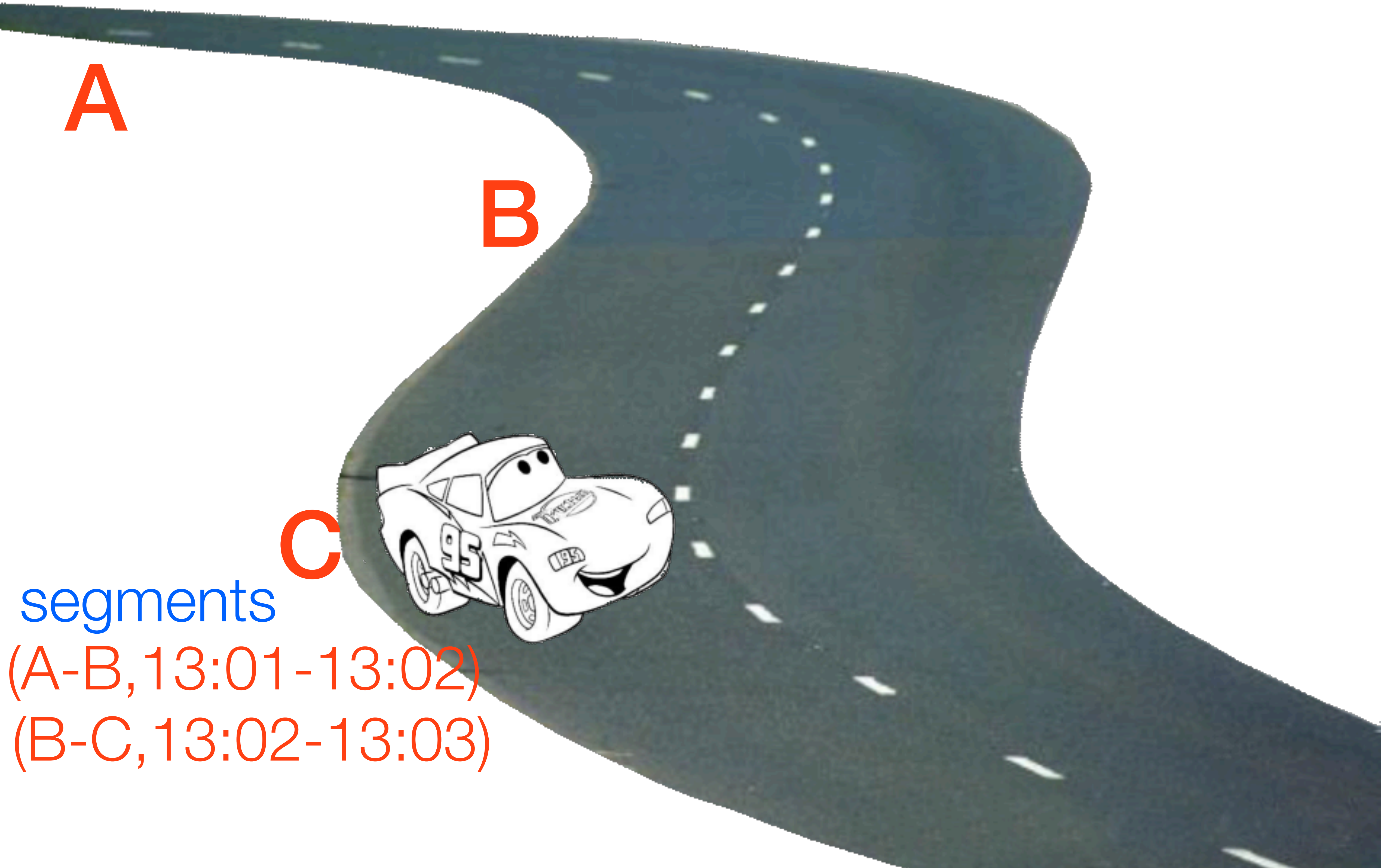
How privacy-preserving toll pricing works



segments

(A-B,13:01-13:02)

How privacy-preserving toll pricing works

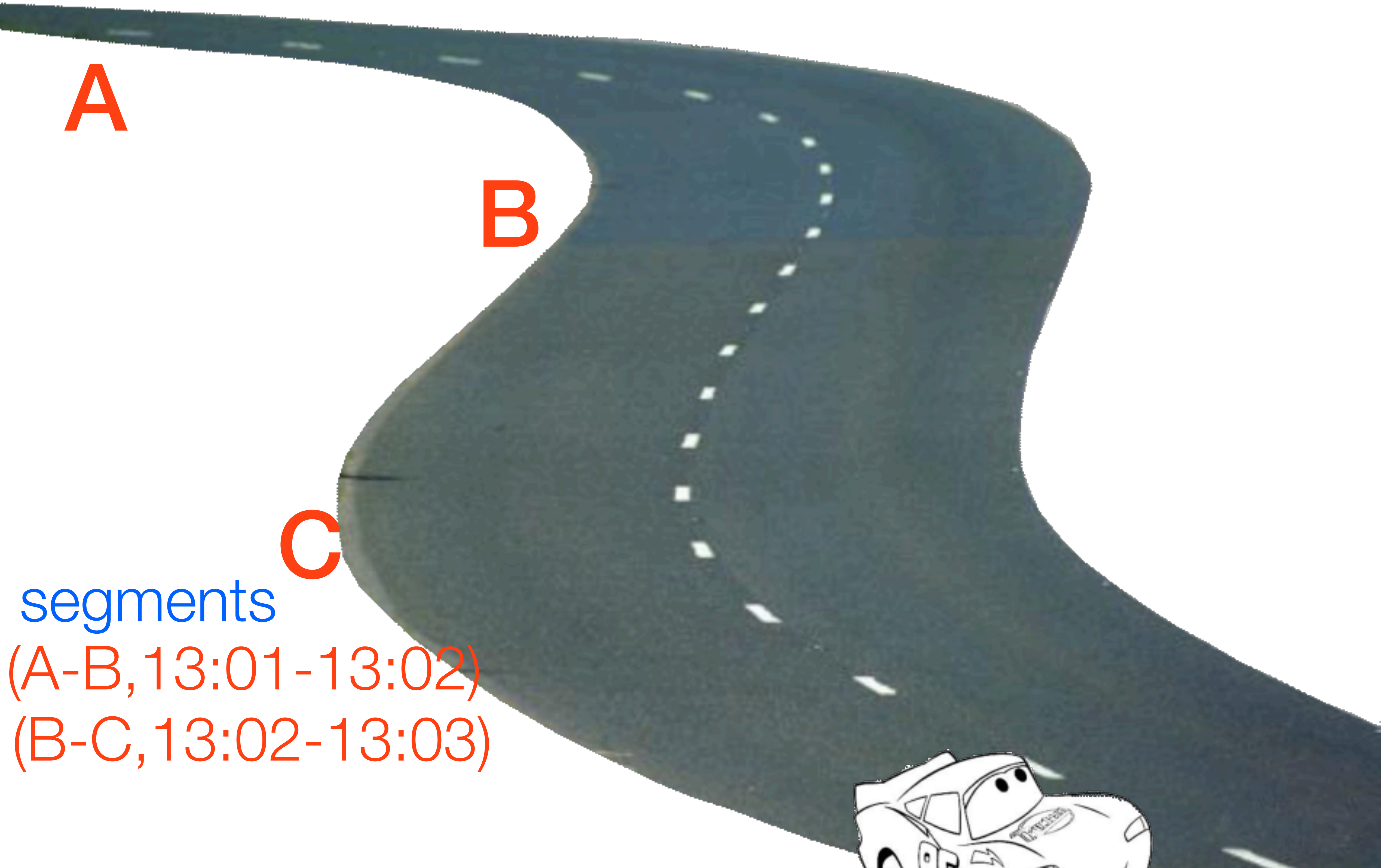


segments

(A-B,13:01-13:02)

(B-C,13:02-13:03)

How privacy-preserving toll pricing works

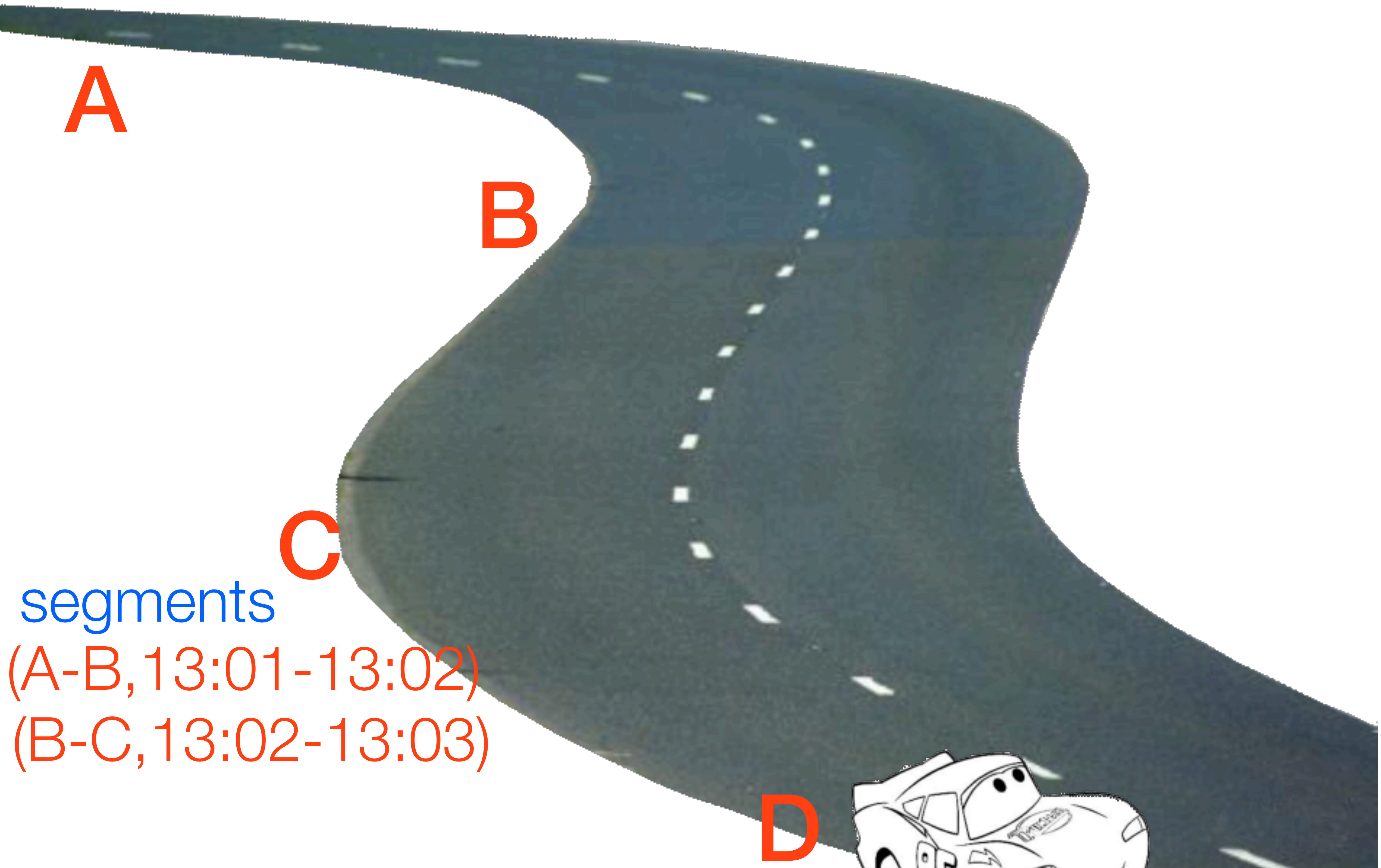


segments

(A-B,13:01-13:02)

(B-C,13:02-13:03)

How privacy-preserving toll pricing works

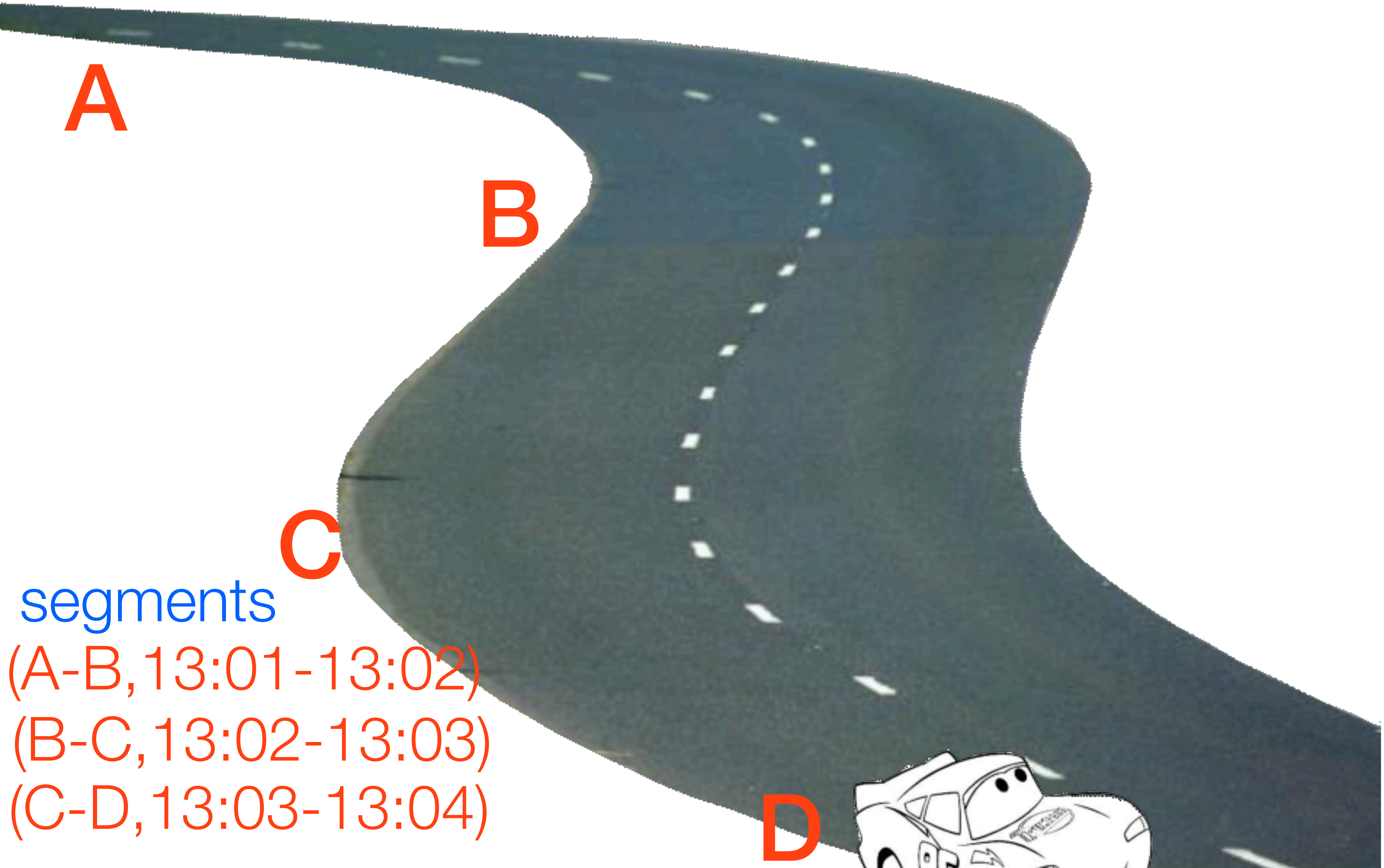


segments

(A-B,13:01-13:02)

(B-C,13:02-13:03)

How privacy-preserving toll pricing works



segments

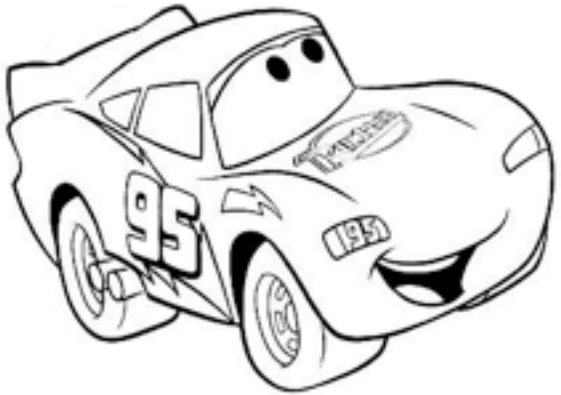
(A-B,13:01-13:02)

(B-C,13:02-13:03)

(C-D,13:03-13:04)

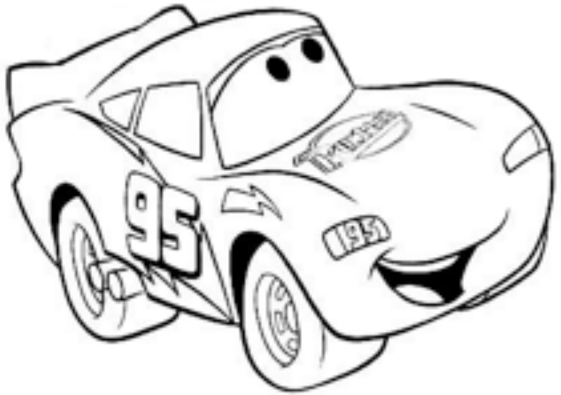
How privacy-preserving toll pricing works

How privacy-preserving toll pricing works



OBU

How privacy-preserving toll pricing works

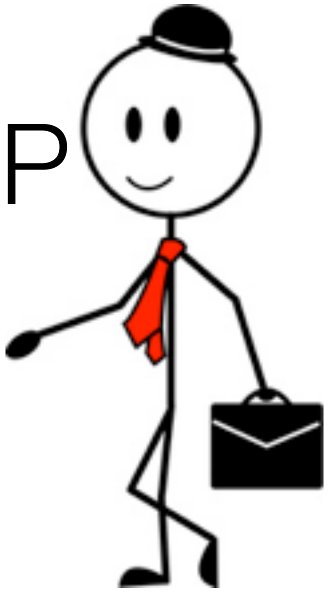


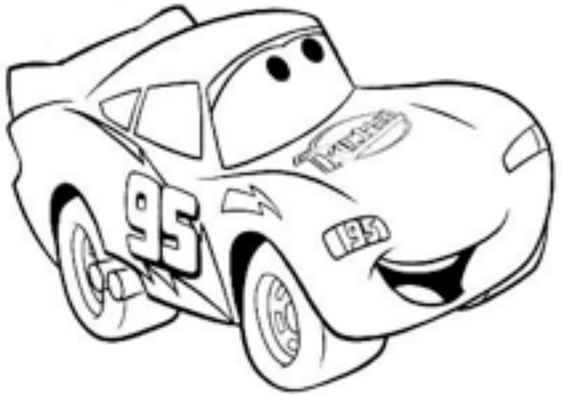
OBU

segments

How privacy-preserving toll pricing works

TSP



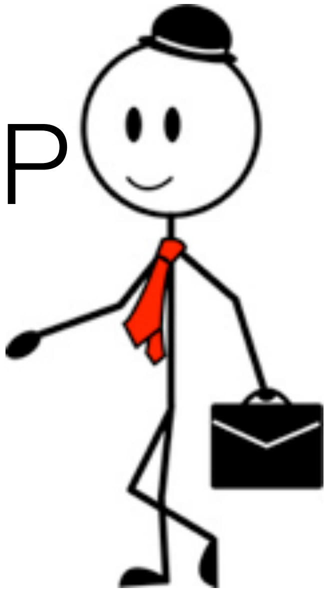


OBU

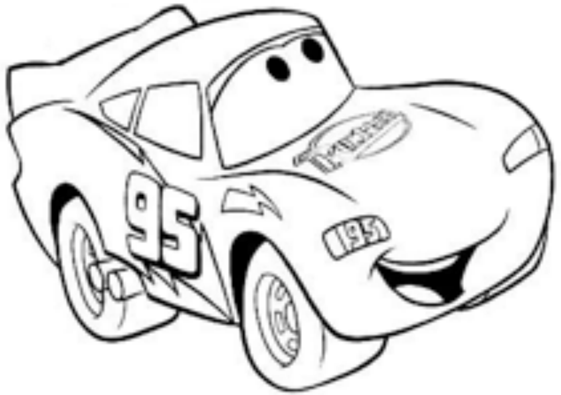
segments

How privacy-preserving toll pricing works

TSP



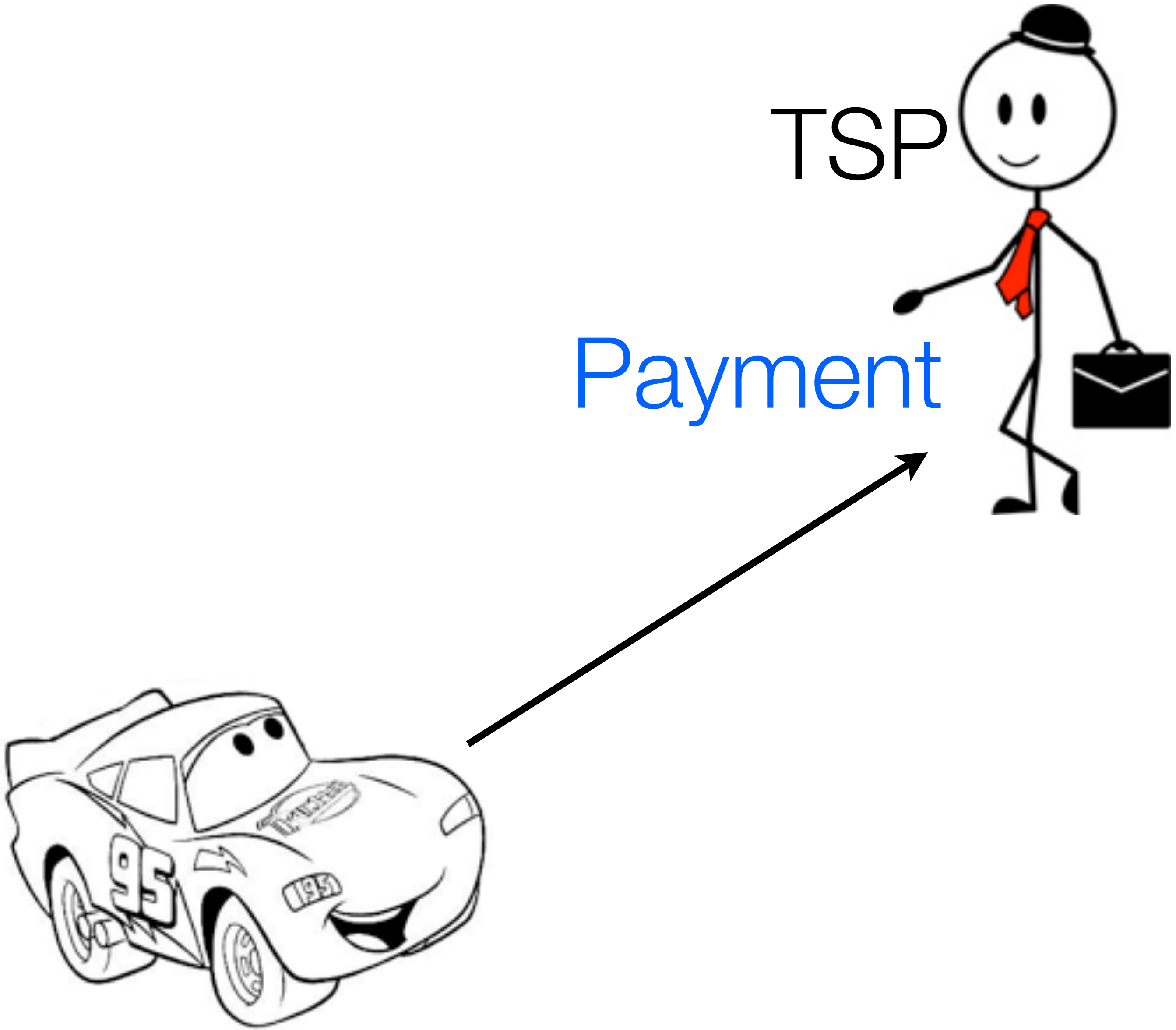
Payment



OBU

segments

How privacy-preserving toll pricing works



TSP

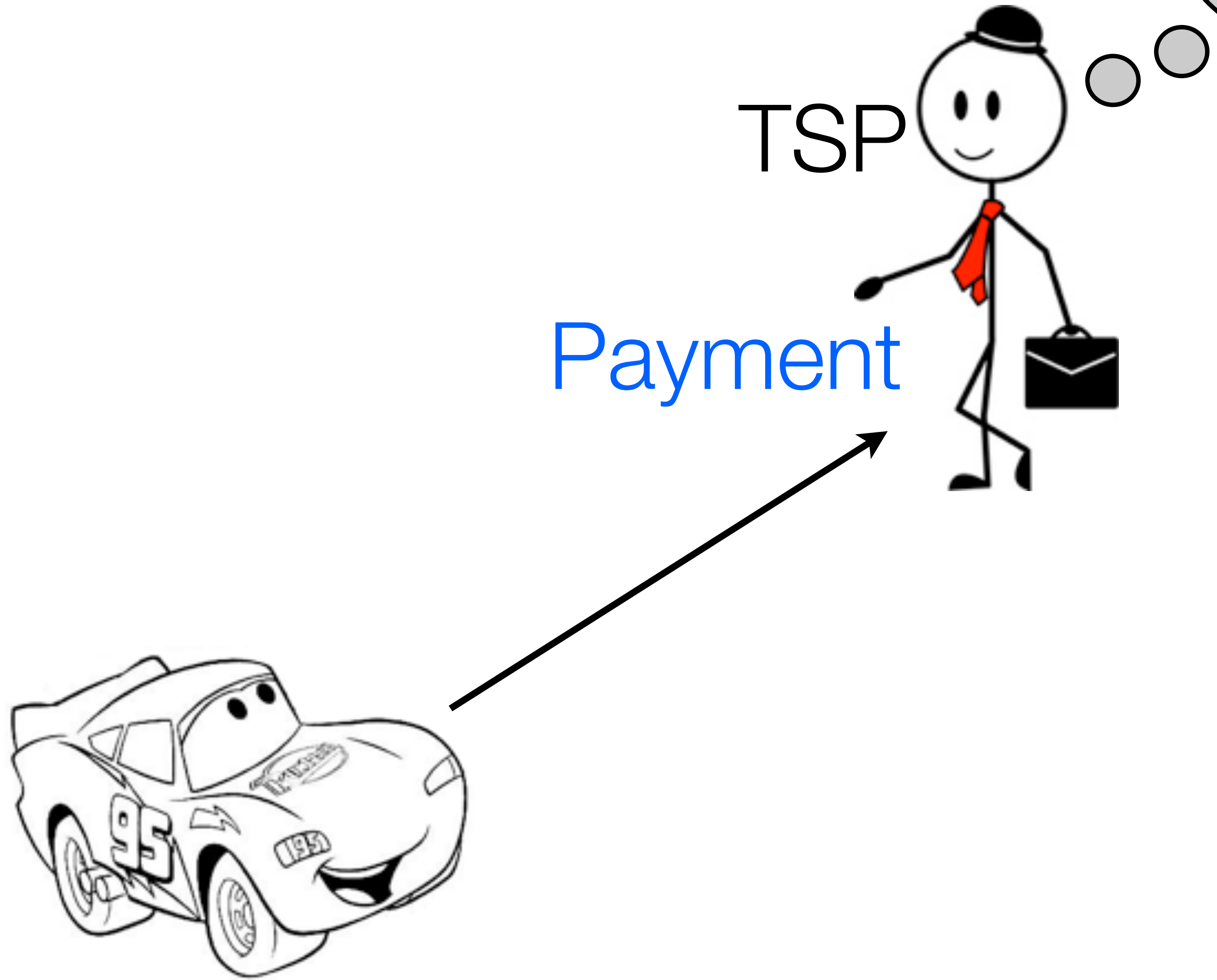
Payment

OBU

segments

How privacy-preserving toll

Check information and charge driver what they owe



TSP

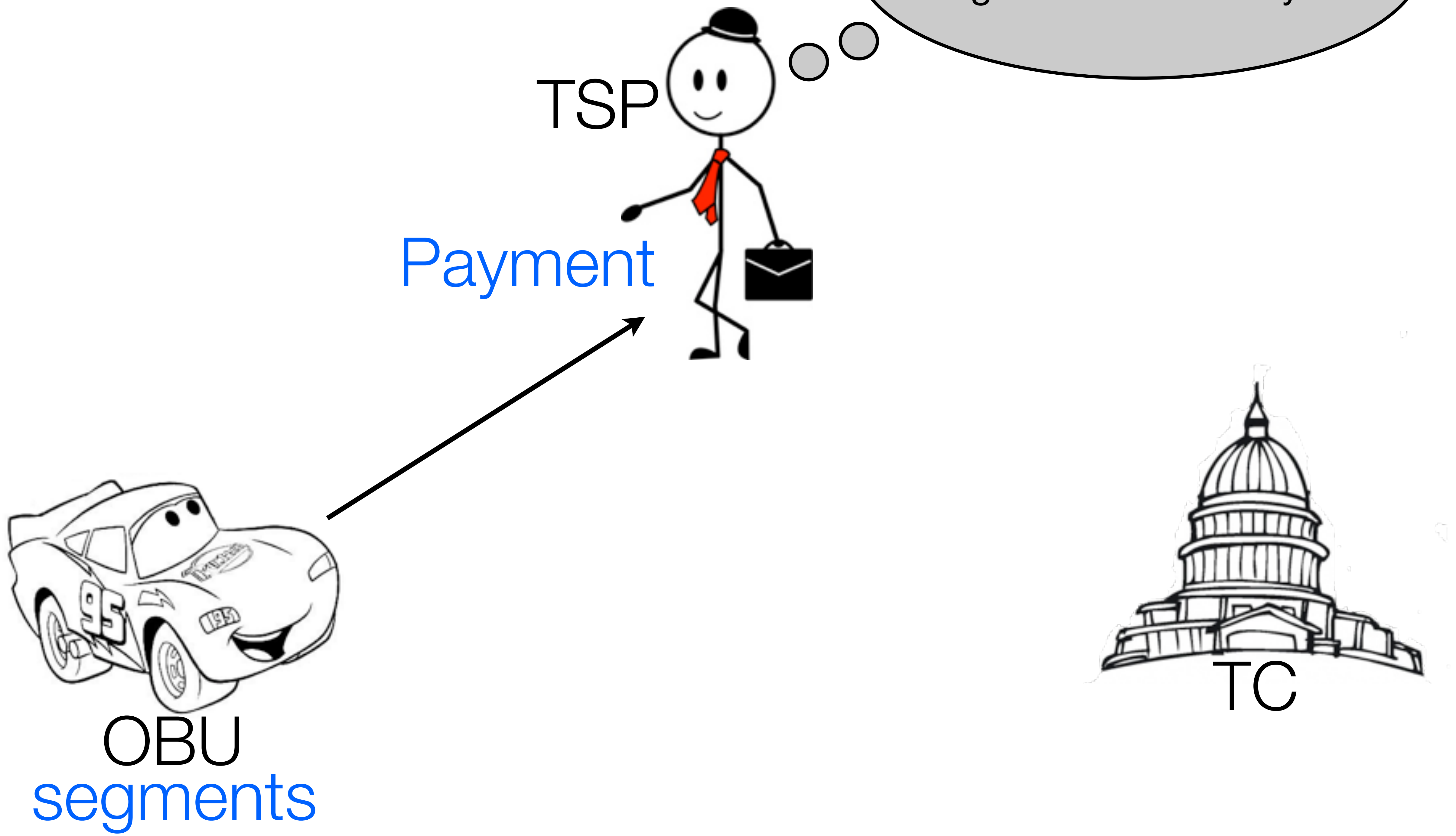
Payment

OBU

segments

How privacy-preserving toll

Check information and charge driver what they owe



TSP

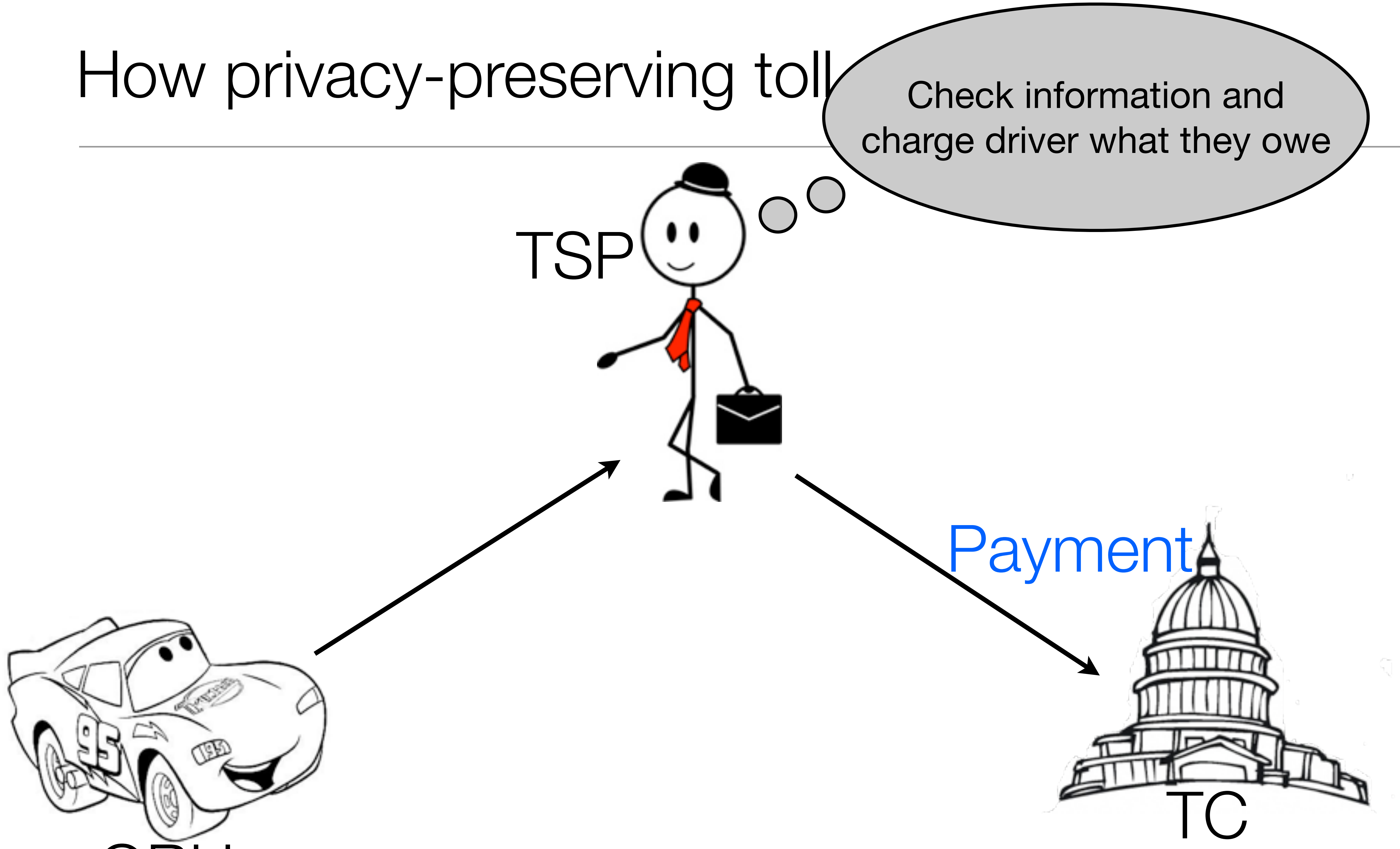
Payment

OBU

segments

TC

How privacy-preserving toll



Check information and charge driver what they owe

TSP

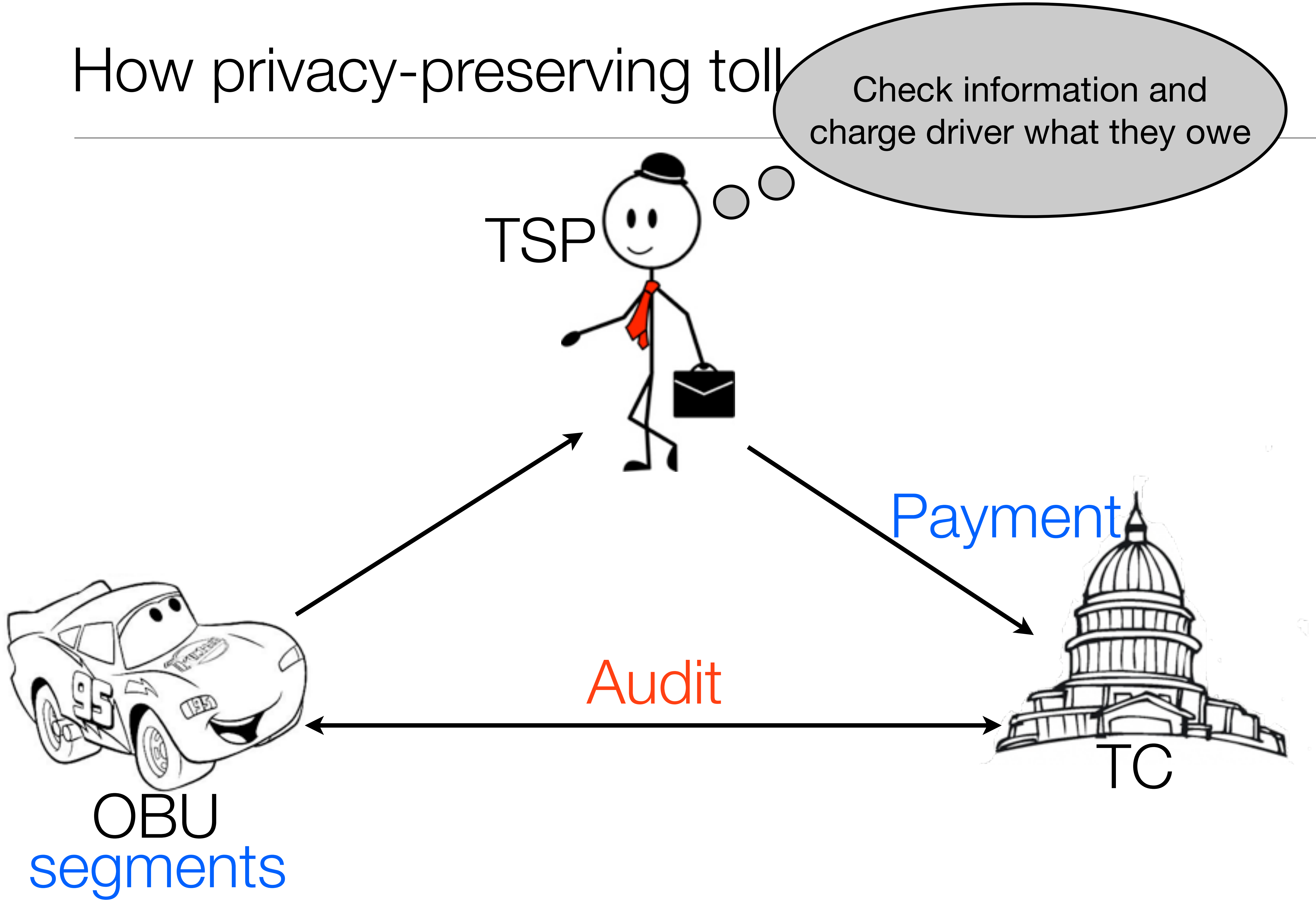
Payment

TC

OBU

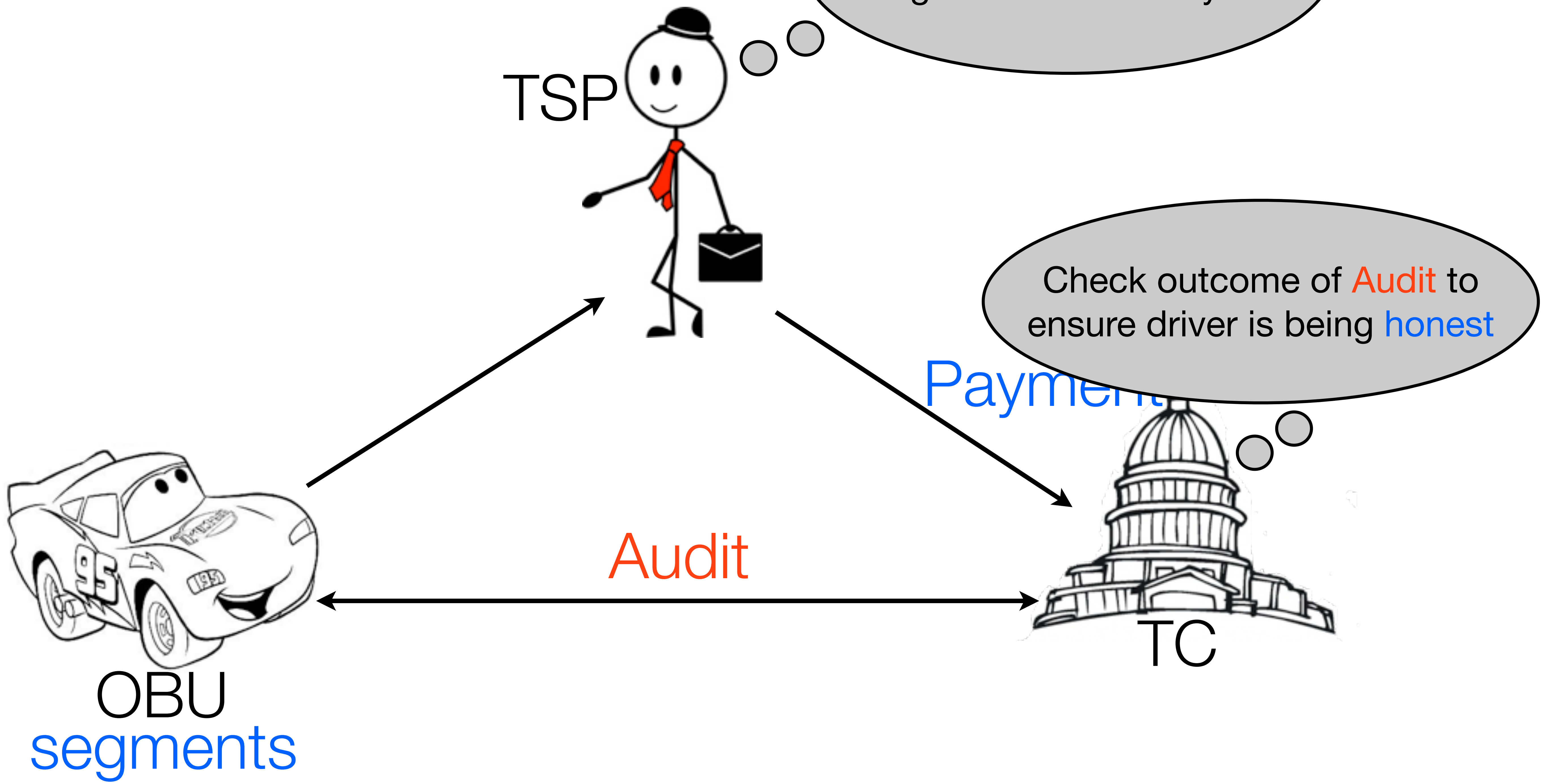
segments

How privacy-preserving toll

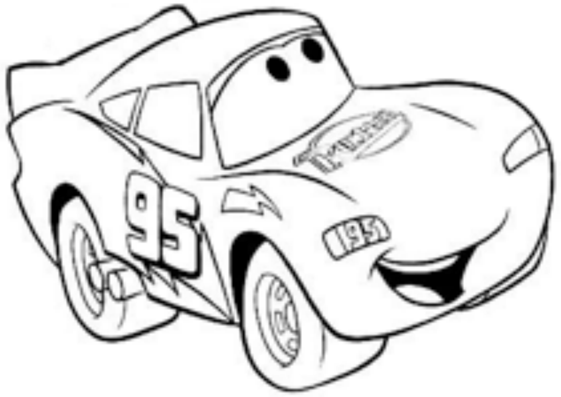


How privacy-preserving toll

Check information and charge driver what they owe

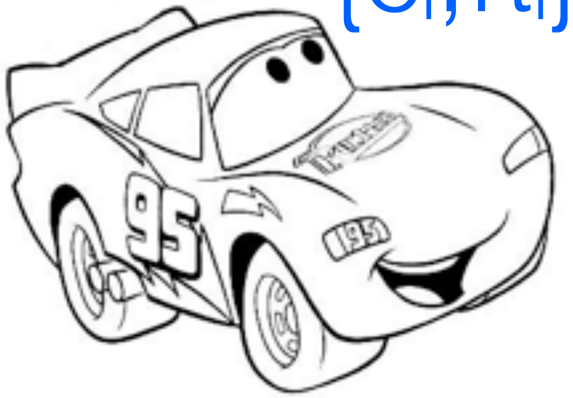


An adapted version of PrETP



An adapted version of PrETP

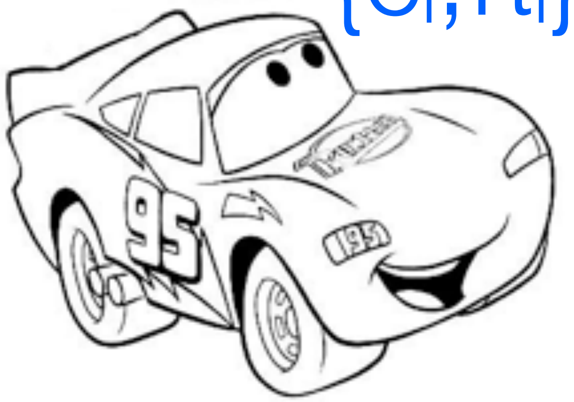
$\{c_i,\pi_i\}_i$



An adapted version of PrETP

Commitment to segment price p_i

$\{c_i, \pi_i\}_i$

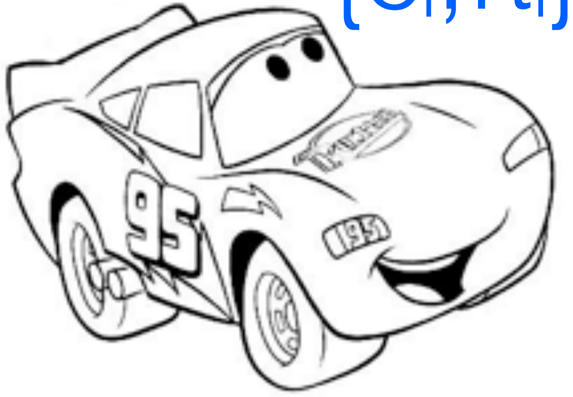


An adapted version of PrETP

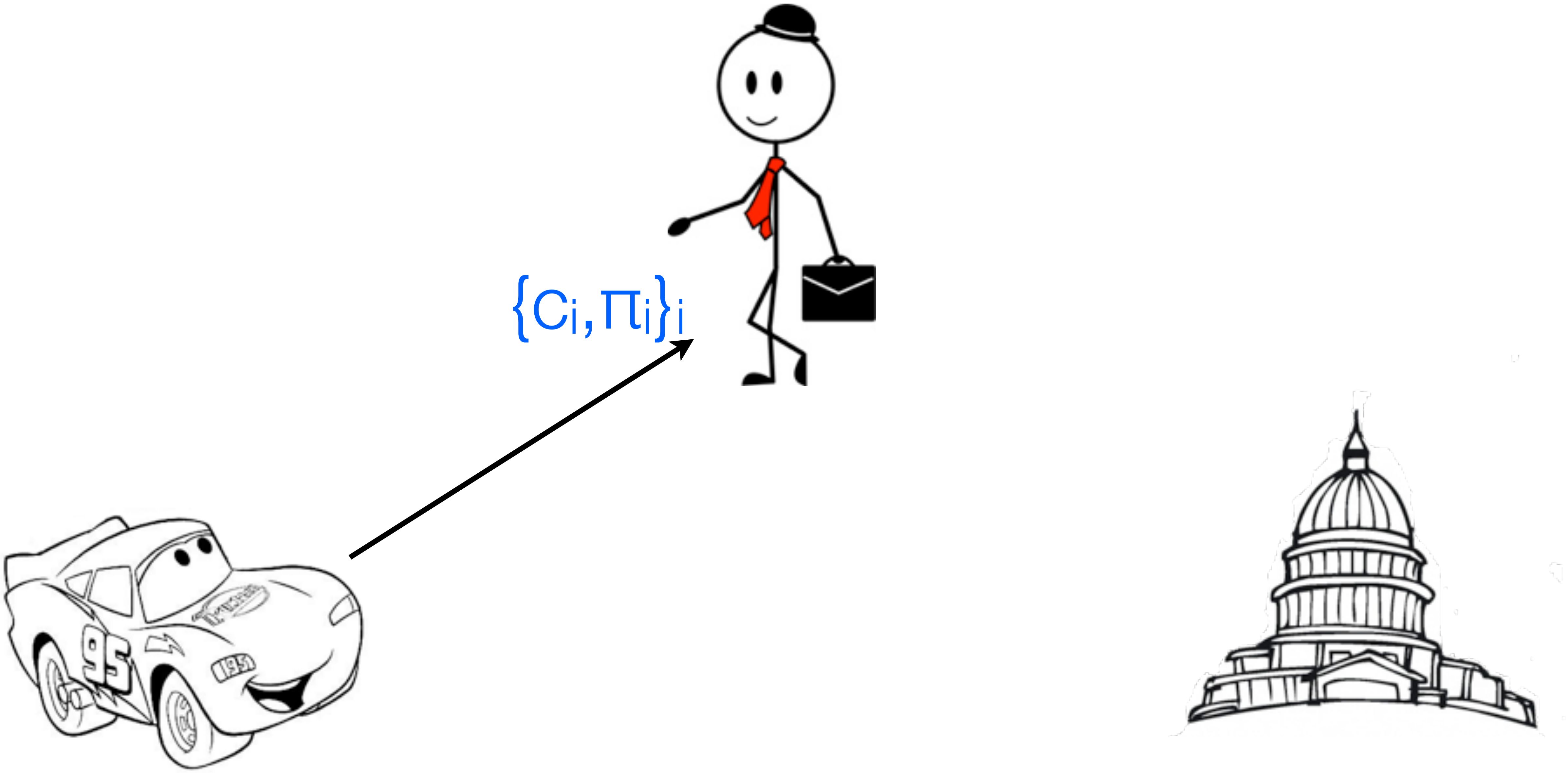
Commitment to segment price p_i

NIZK that the value in c_i is in the proper range

$\{c_i, \pi_i\}_i$

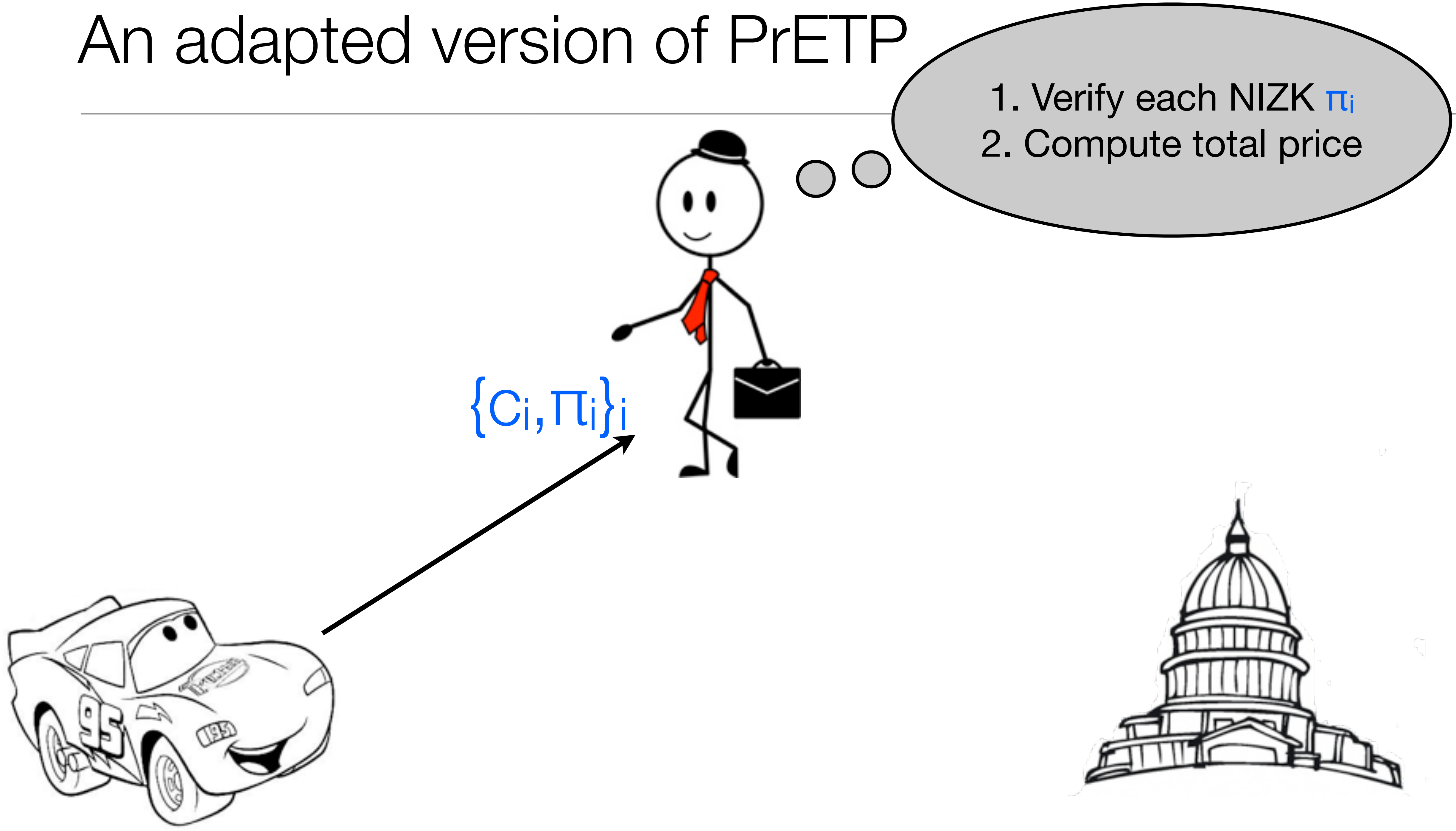


An adapted version of PrETP



$\{c_i, \pi_i\}_i$

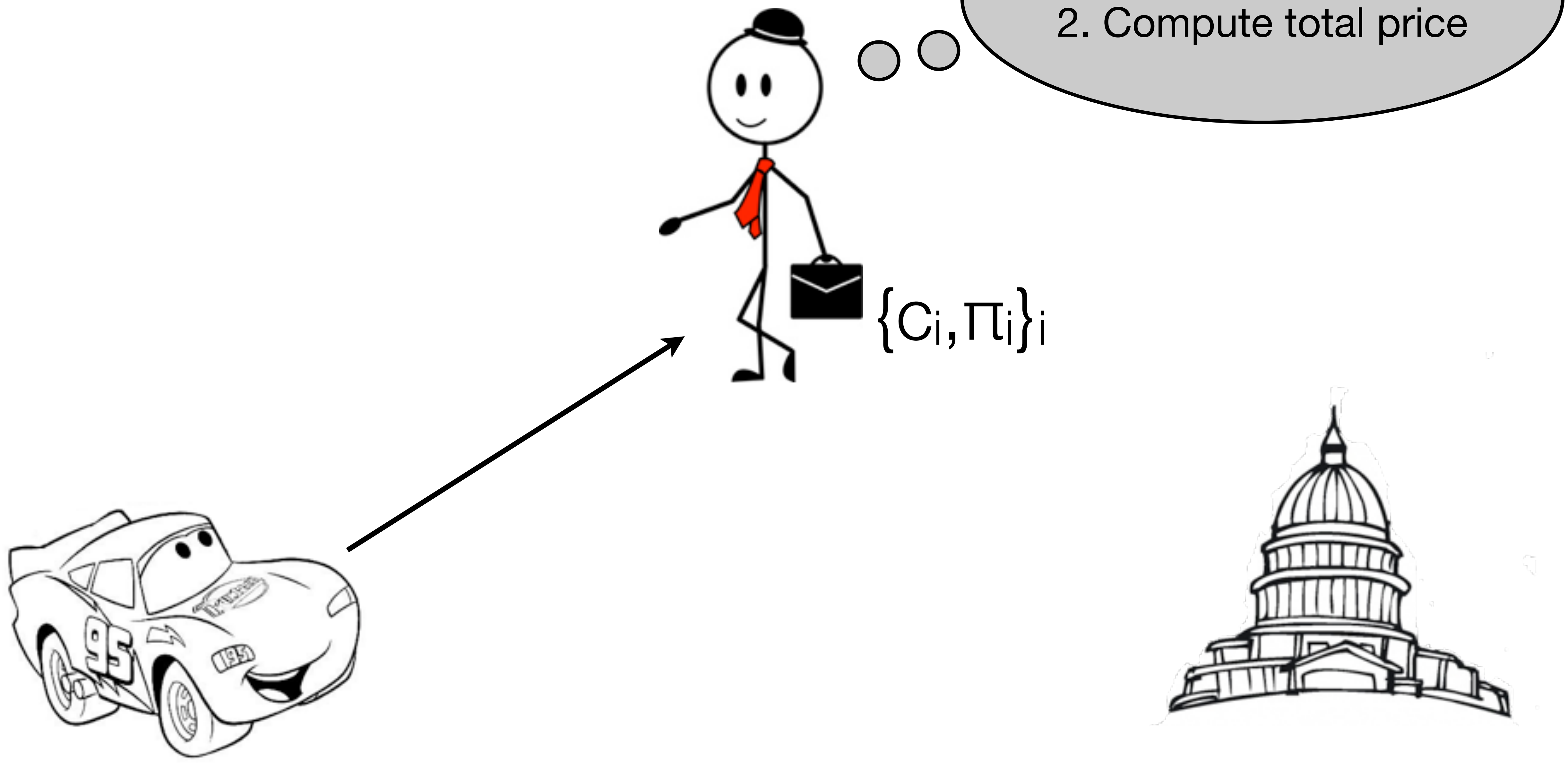
An adapted version of PrETP



1. Verify each NIZK π_i
2. Compute total price

$\{c_i, \pi_i\}_i$

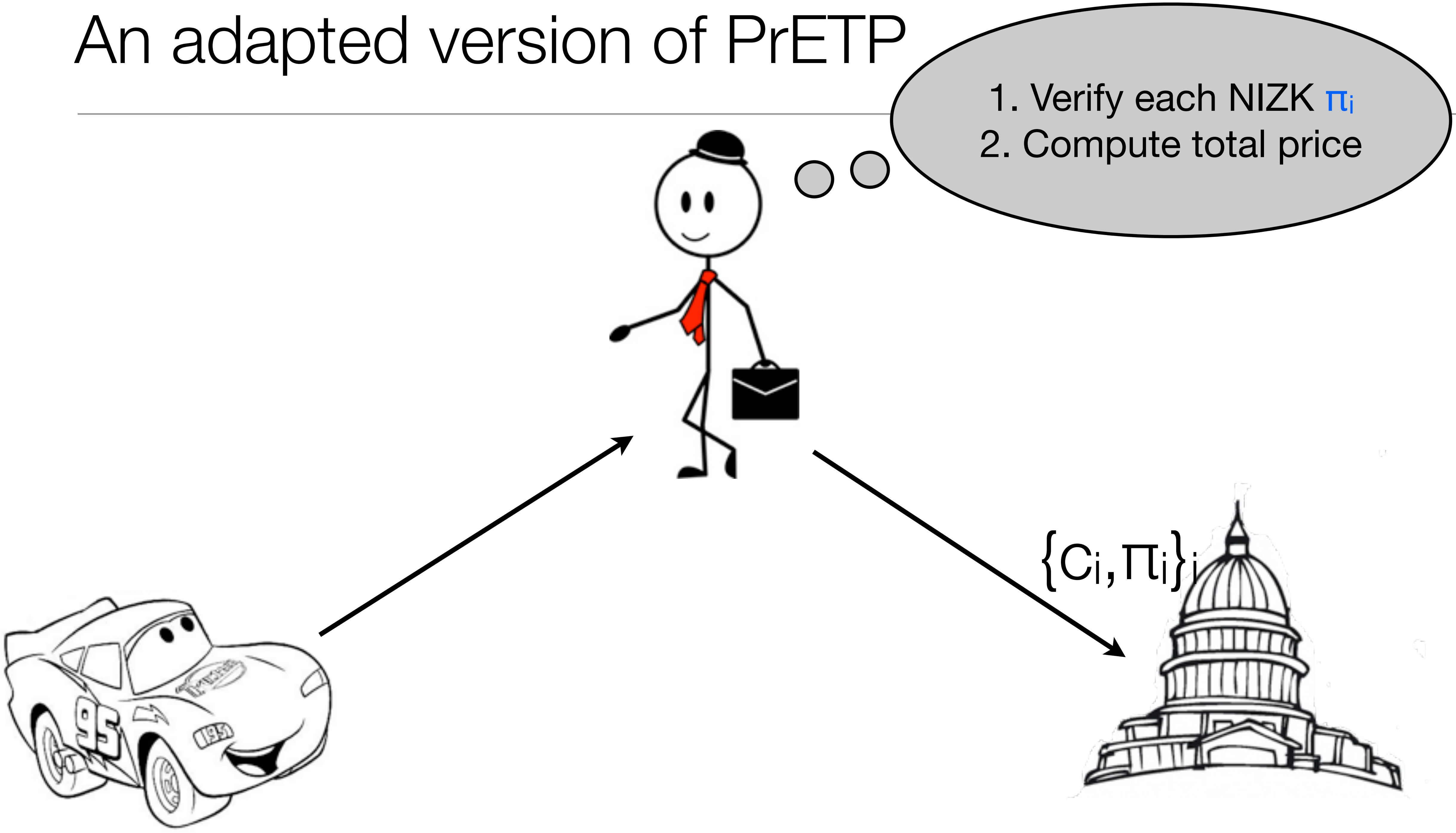
An adapted version of PrETP



1. Verify each NIZK π_i
2. Compute total price

$\{c_i, \pi_i\}_i$

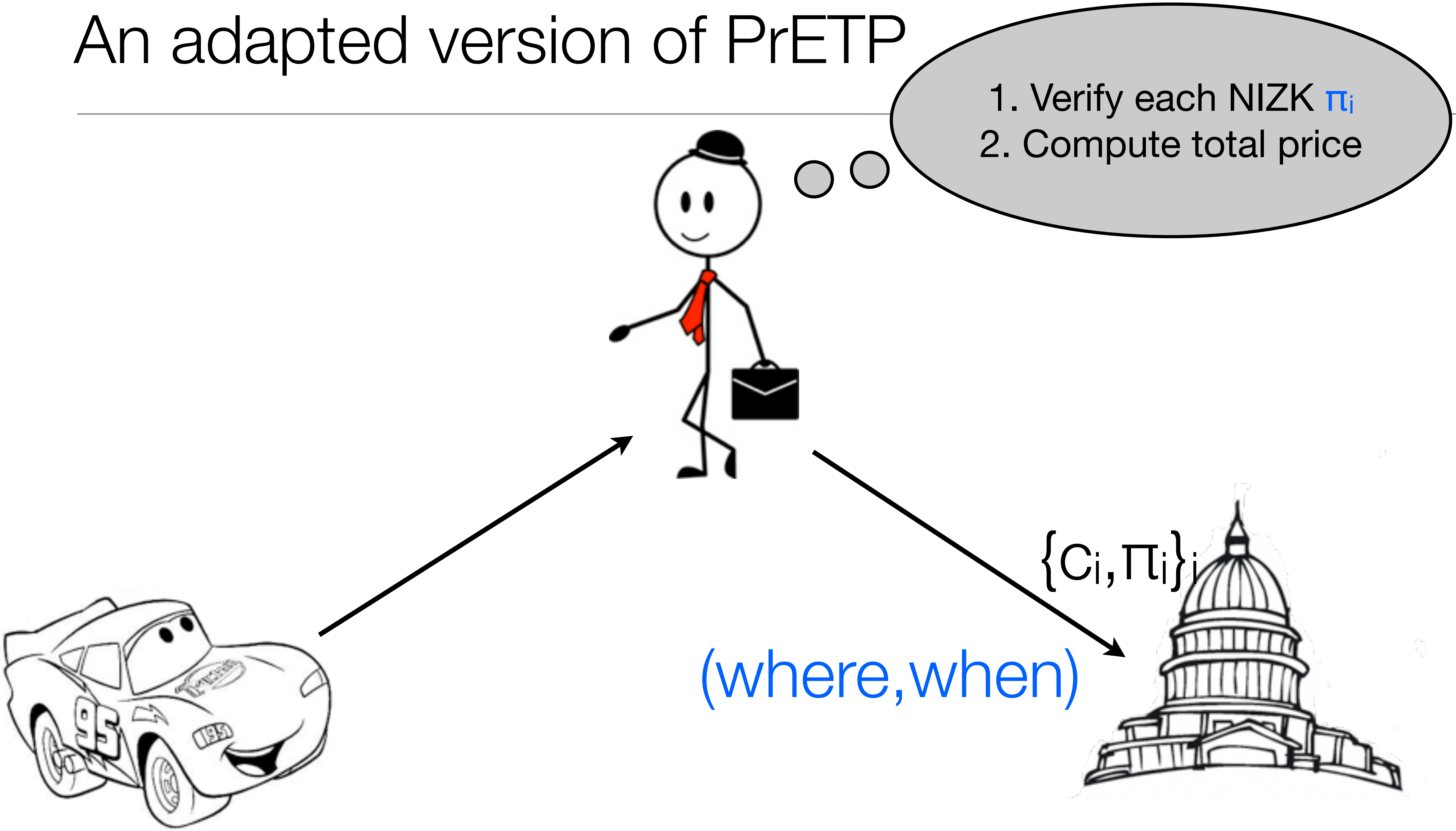
An adapted version of PrETP



1. Verify each NIZK π_i
2. Compute total price

$\{c_i,\pi_i\}_i$

An adapted version of PrETP

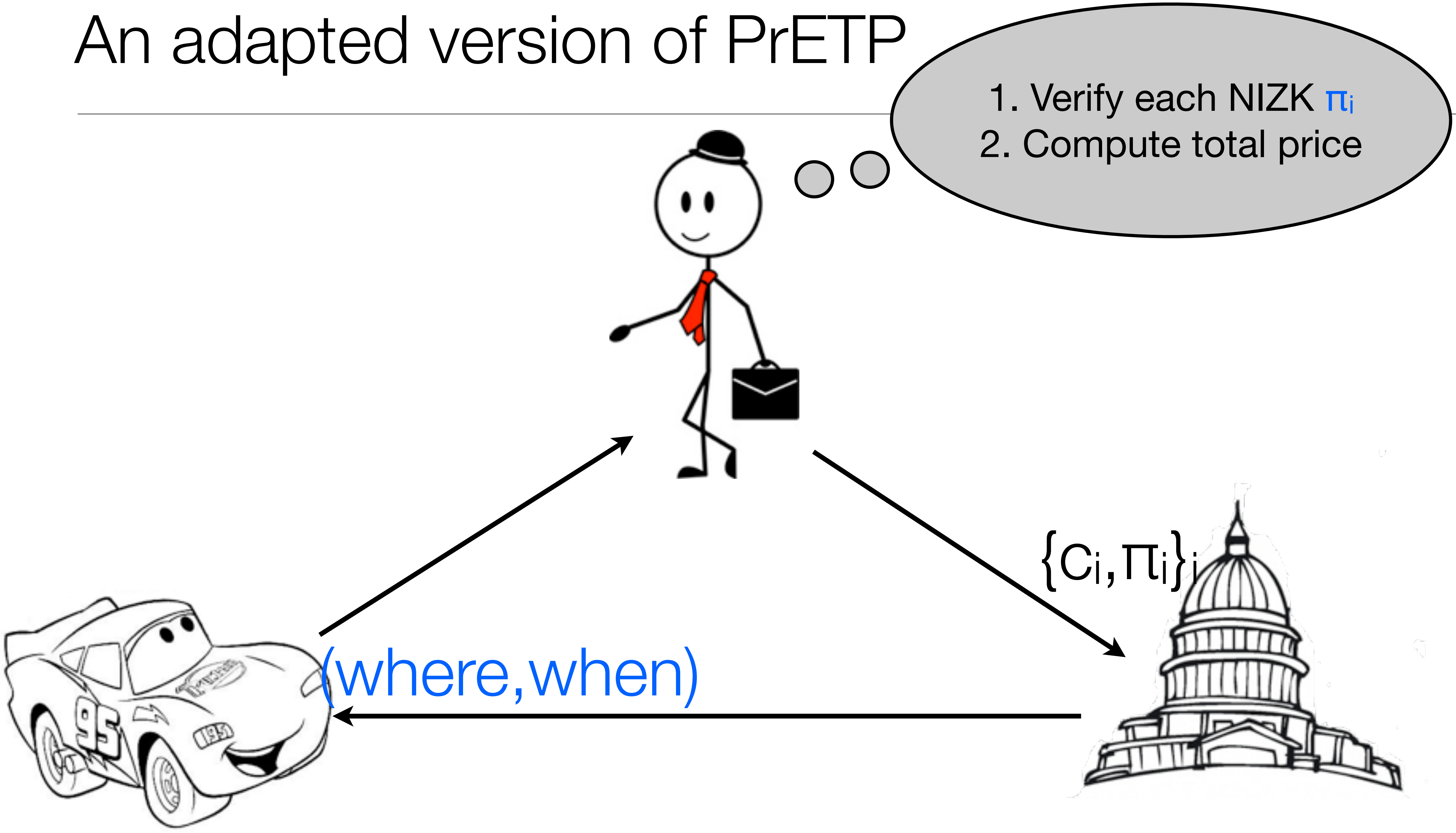


1. Verify each NIZK π_i
2. Compute total price

$\{c_i,\pi_i\}_i$

(where,when)

An adapted version of PrETP

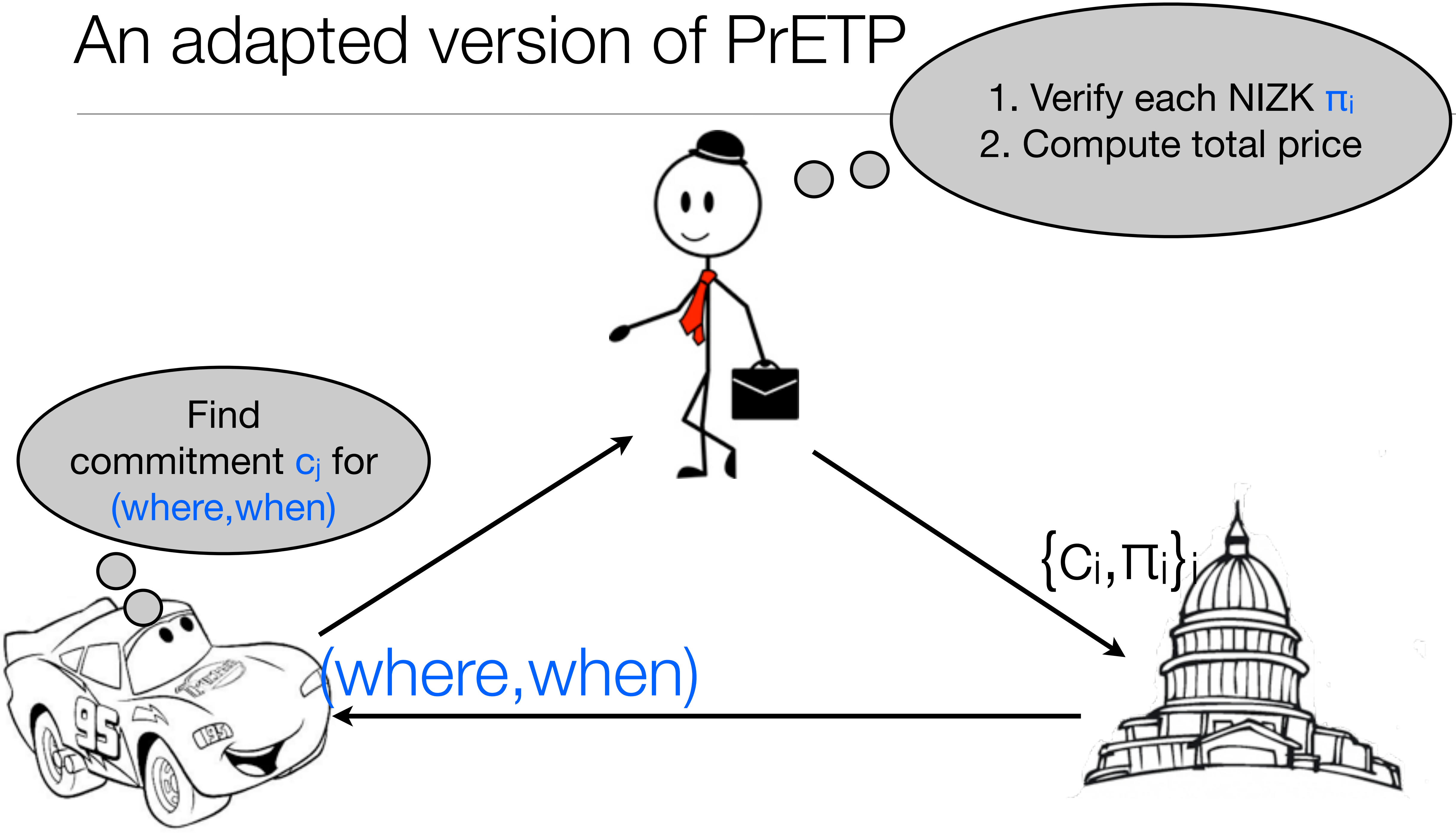


1. Verify each NIZK π_i
2. Compute total price

$\{c_i,\pi_i\}_i$

(where,when)

An adapted version of PrETP



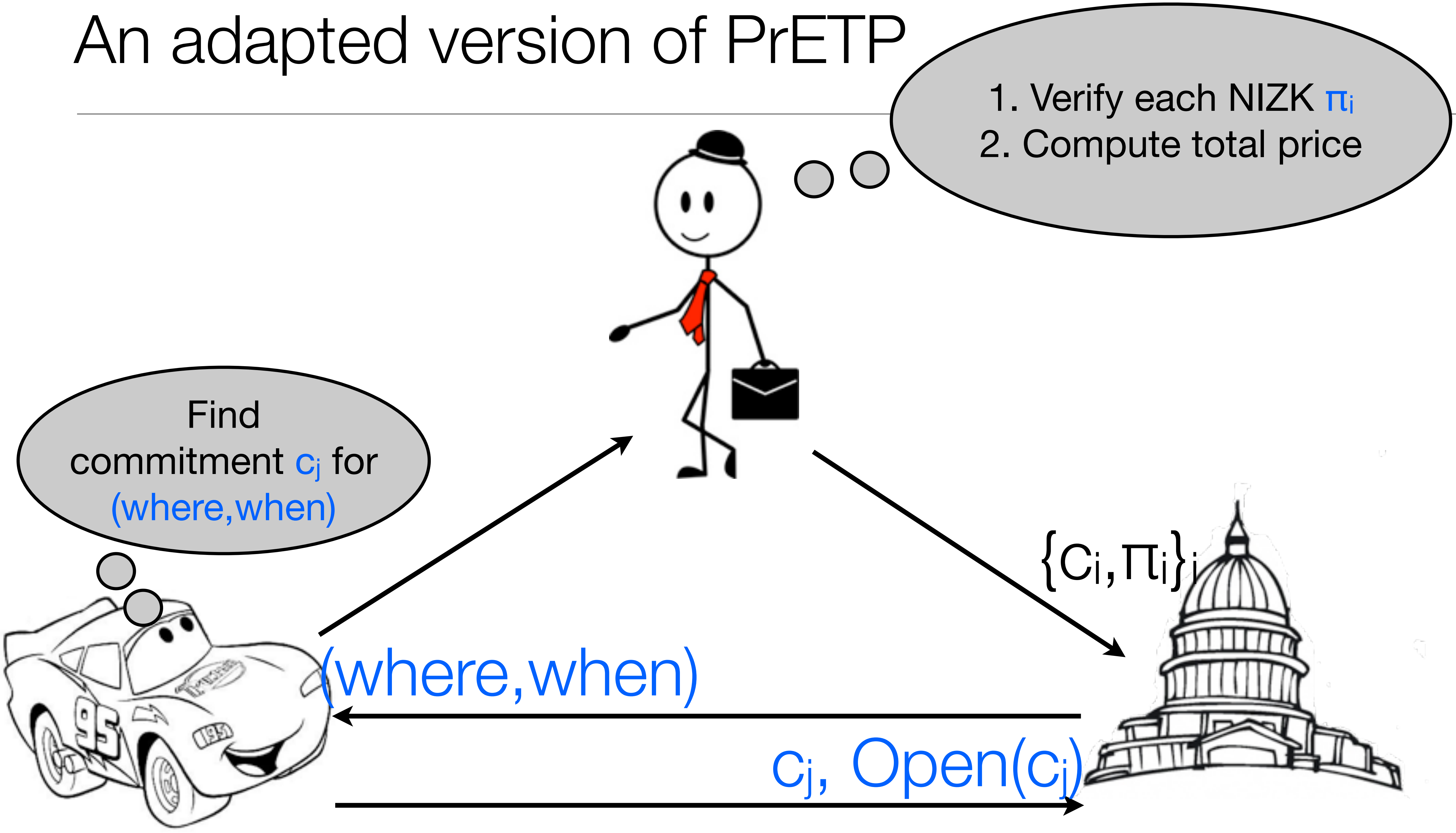
1. Verify each NIZK π_i
2. Compute total price

Find commitment c_j for (where,when)

$\{c_i,\pi_i\}_i$

(where,when)

An adapted version of PrETP



1. Verify each NIZK π_i
2. Compute total price

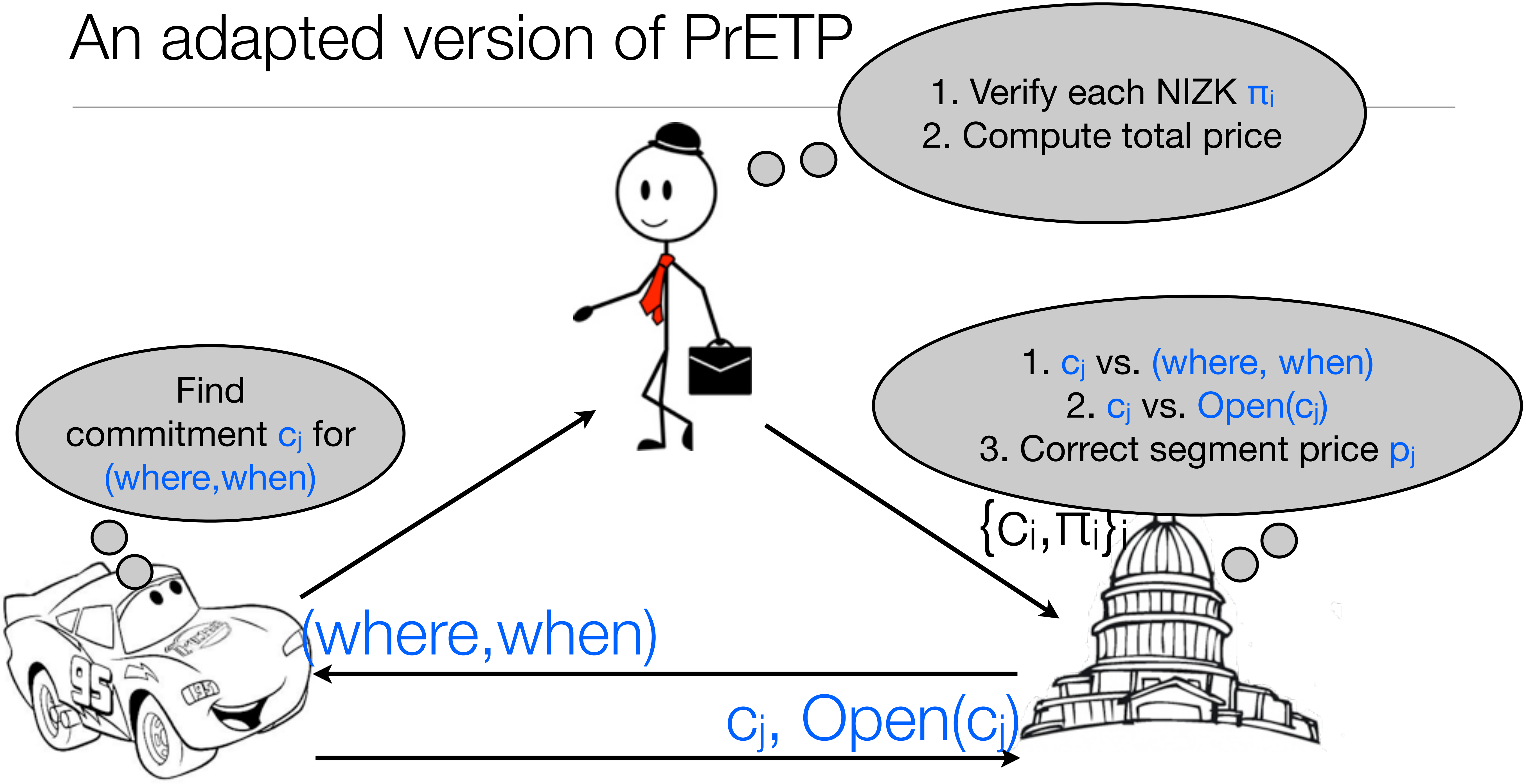
Find commitment c_j for (where,when)

$\{c_i, \pi_i\}_i$

(where,when)

c_j, Open(c_j)

An adapted version of PrETP



1. Verify each NIZK π_i
2. Compute total price

Find commitment c_j for (where,when)

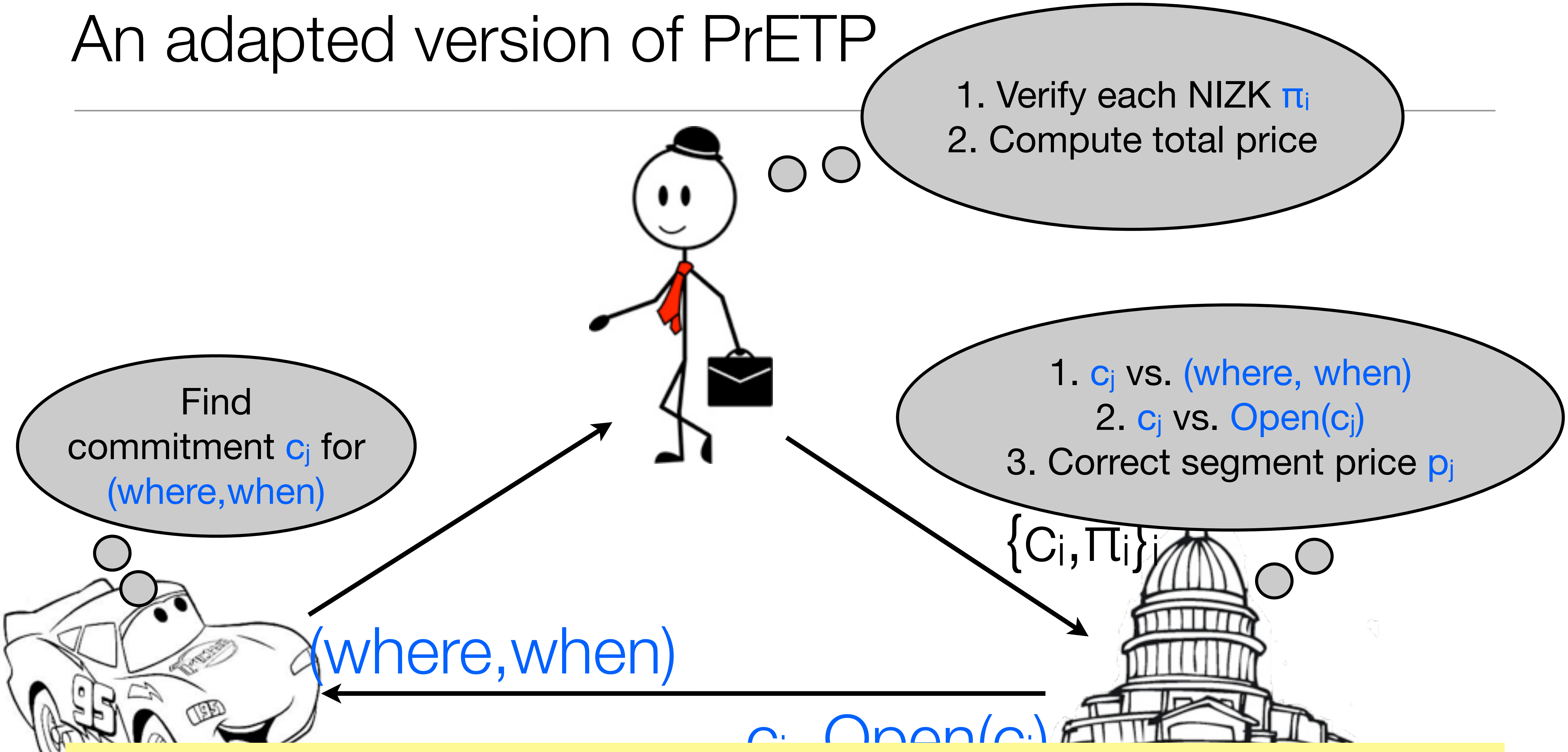
1. c_j vs. (where, when)
2. c_j vs. Open(c_j)
3. Correct segment price p_j

$\{c_i,\pi_i\}_i$

(where,when)

c_j, Open(c_j)

An adapted version of PrETP

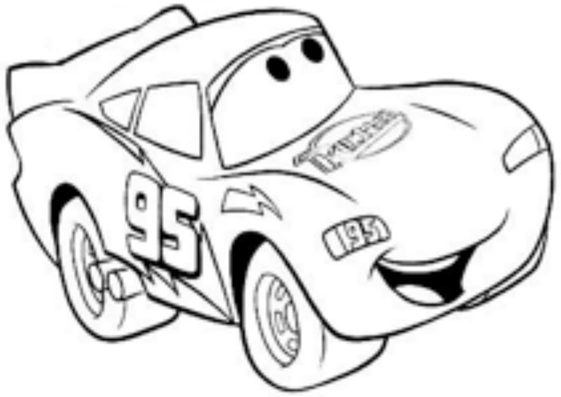

1. Verify each NIZK π_i
2. Compute total price
Find commitment c_j for (where,when)
1. c_j vs. (where, when)
2. c_j vs. Open(c_j)
3. Correct segment price p_j
$\{c_i,\pi_i\}_i$
(where,when)
c_j, Open(c_j)

NIZK zero knowledge and commitment hiding guarantee driver privacy

NIZK soundness guarantees price p_i is in the right range (e.g., non-negative)

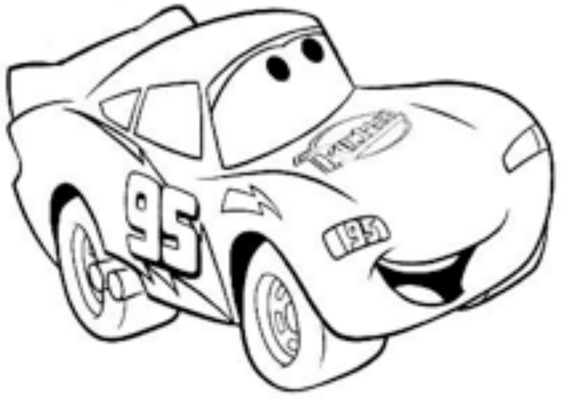
Commitment binding guarantees c_j is the right commitment for (where,when)

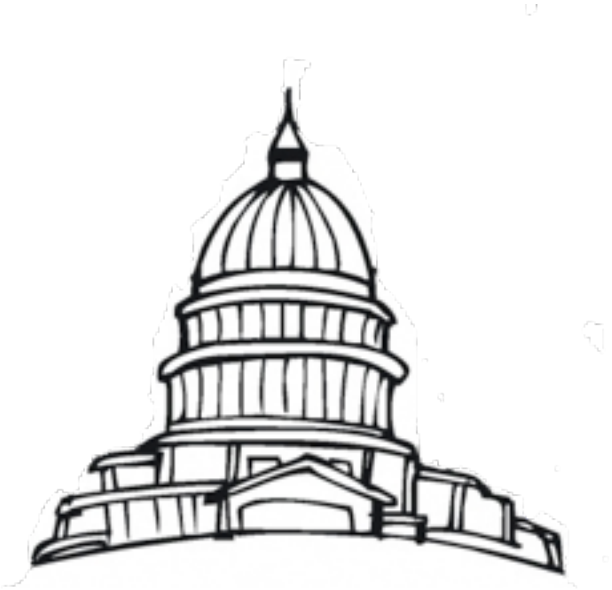
“PrETP with sugar on top”: our new Audit protocol



“PrETP with sugar on top”: our new Audit protocol

$\{c_i, C_i, \pi_i\}_i$

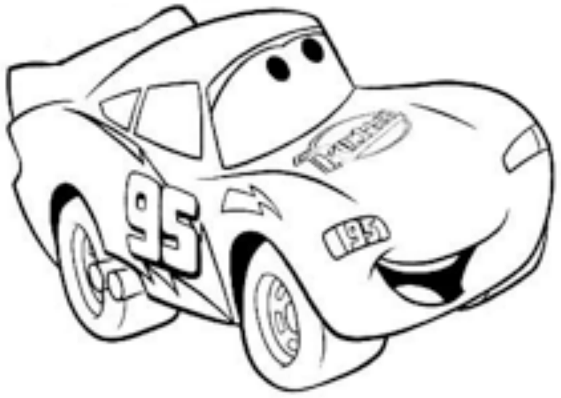




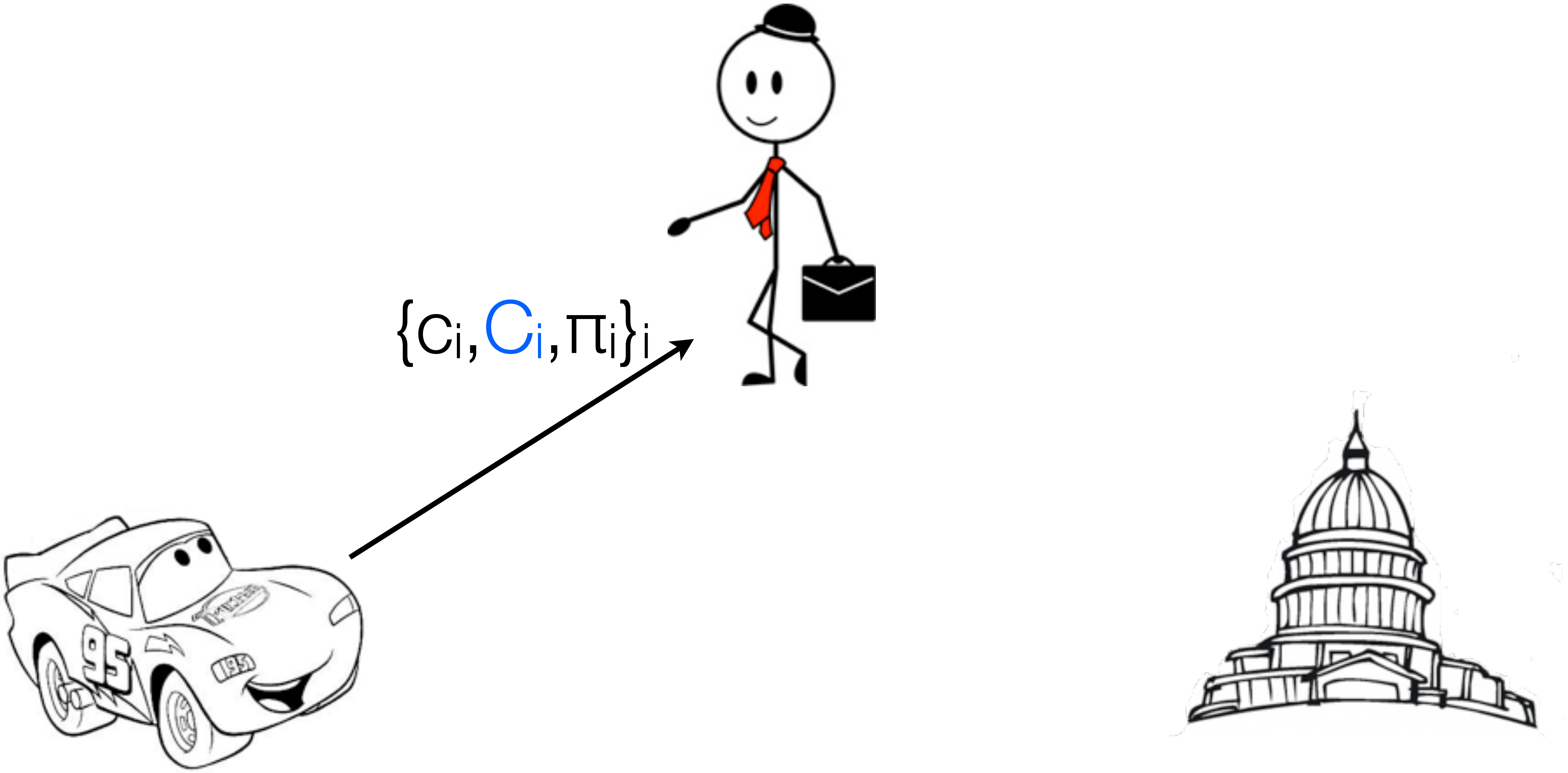
“PrETP with sugar on top”: our new Audit protocol

Blind IBE of the opening to c_i, using (where,when) as identity

$\{c_i, C_i, \pi_i\}_i$

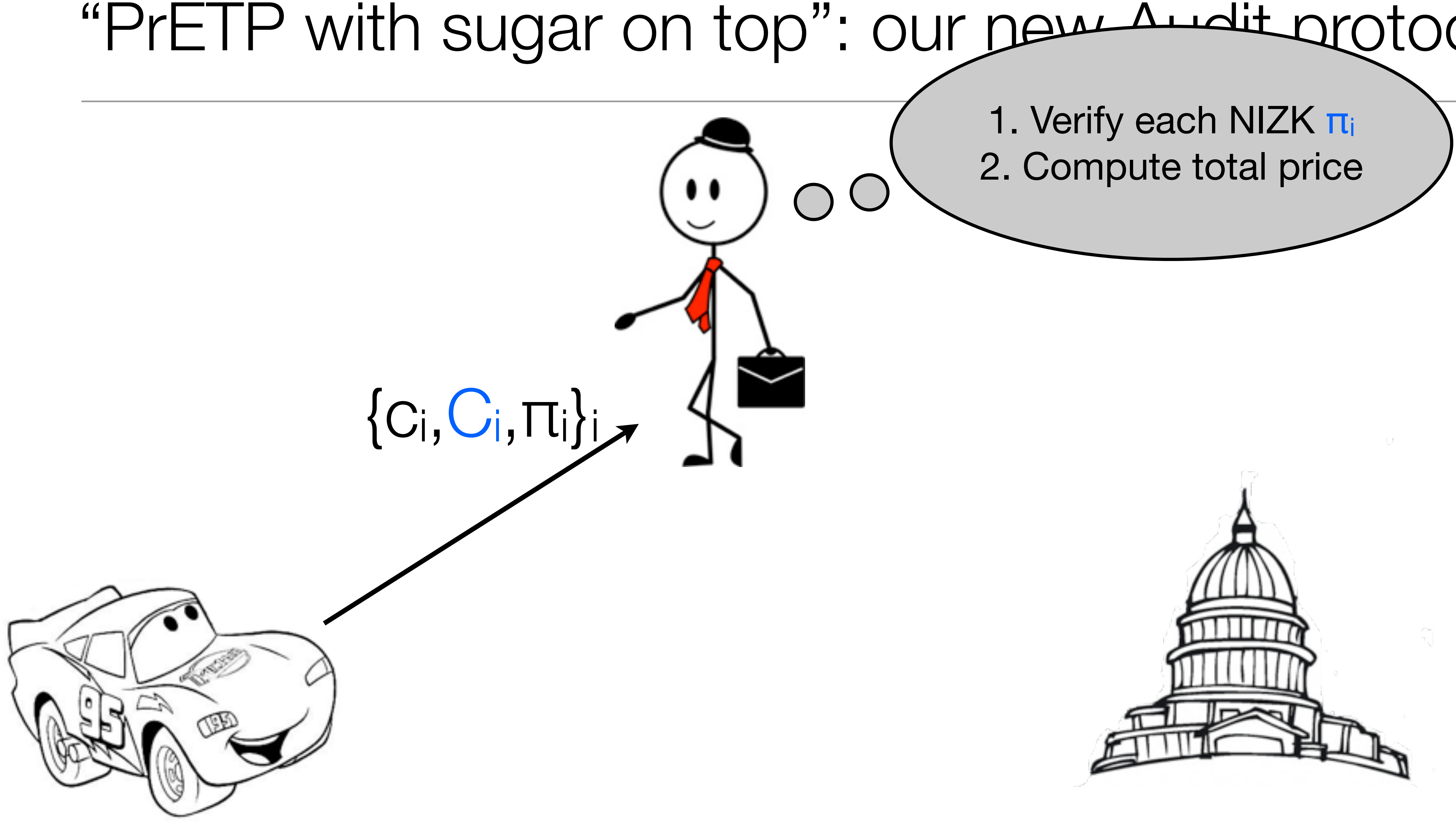


“PrETP with sugar on top”: our new Audit protocol



$\{c_i, C_i, \pi_i\}_i$

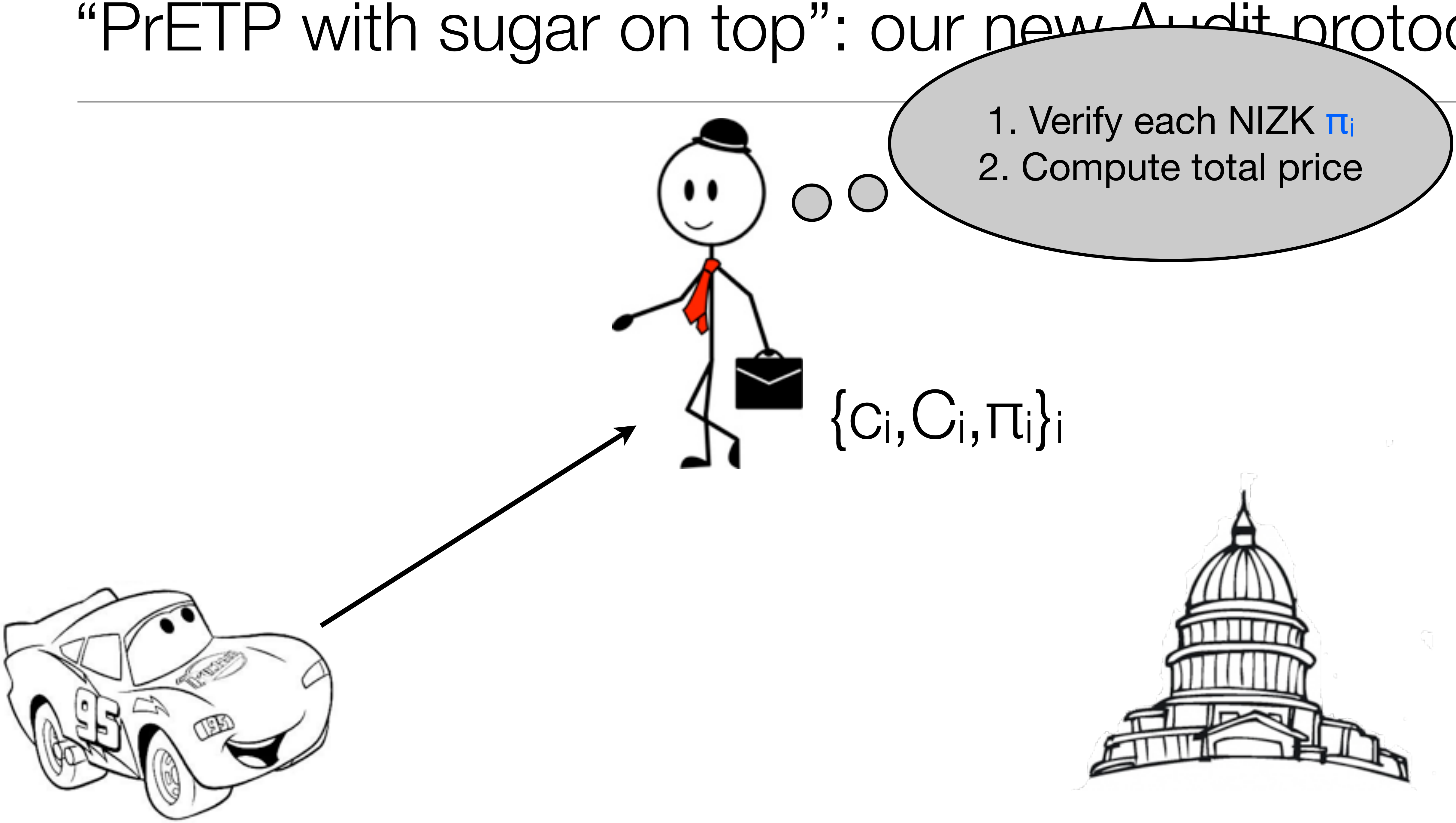
“PrETP with sugar on top”: our new Audit protocol



1. Verify each NIZK π_i
2. Compute total price

$\{c_i, C_i, \pi_i\}_i$

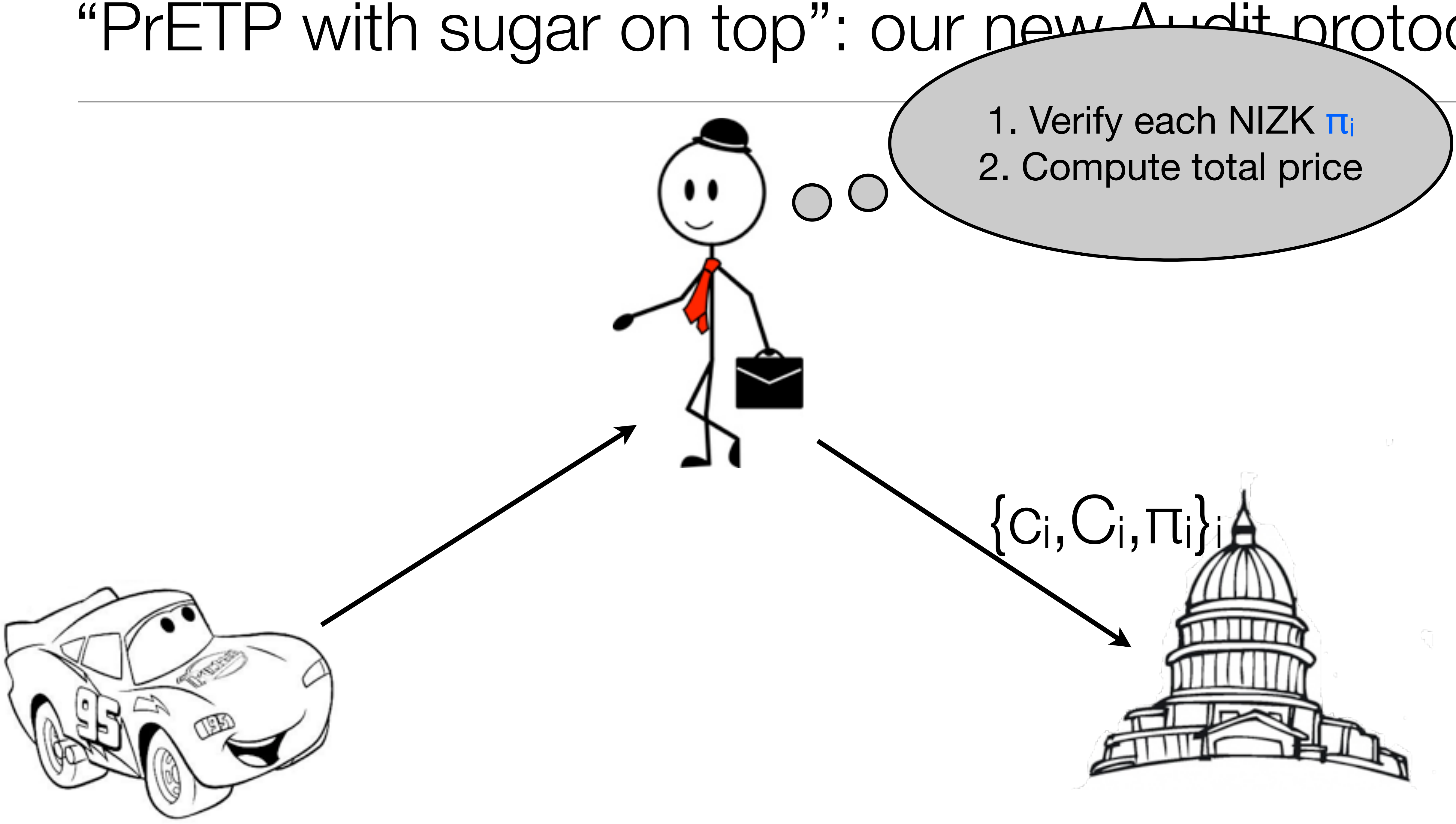
“PrETP with sugar on top”: our new Audit protocol



1. Verify each NIZK π_i
2. Compute total price

$\{c_i, C_i, \pi_i\}_i$

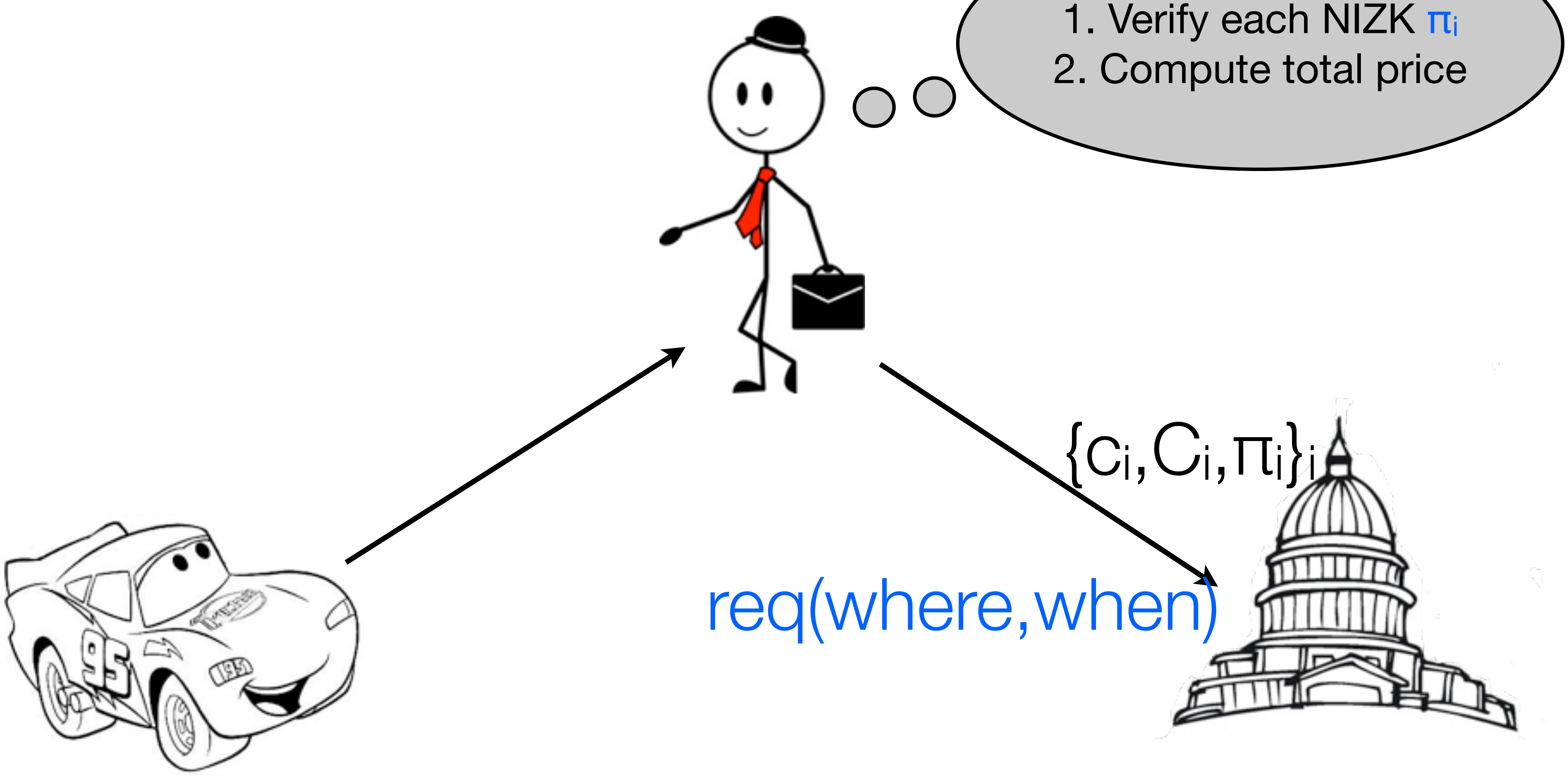
“PrETP with sugar on top”: our new Audit protocol



1. Verify each NIZK π_i
2. Compute total price

$\{c_i,C_i,\pi_i\}_i$

“PrETP with sugar on top”: our new Audit protocol

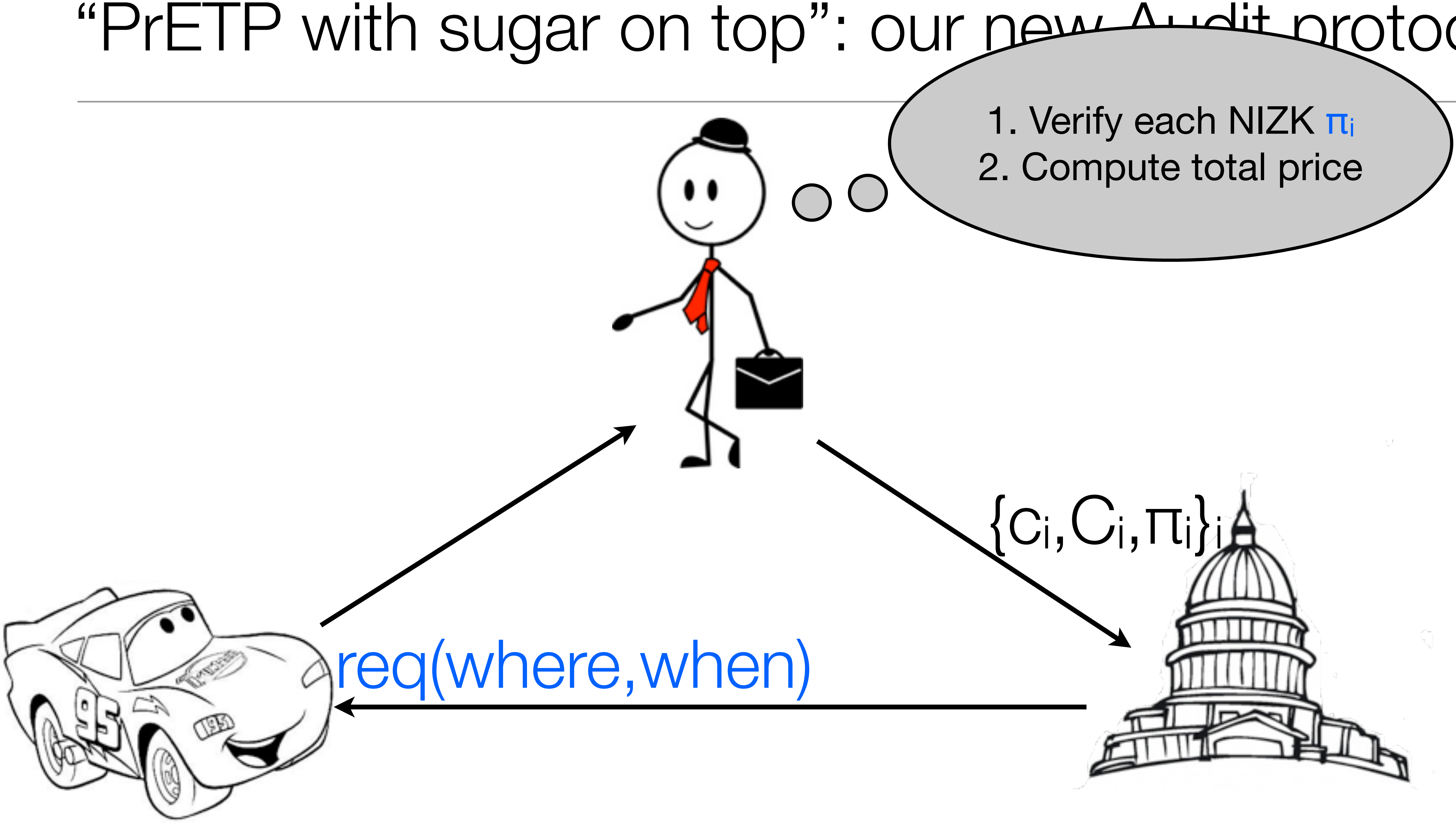


1. Verify each NIZK π_i
2. Compute total price

$\{c_i, C_i, \pi_i\}_i$

req(where,when)

“PrETP with sugar on top”: our new Audit protocol

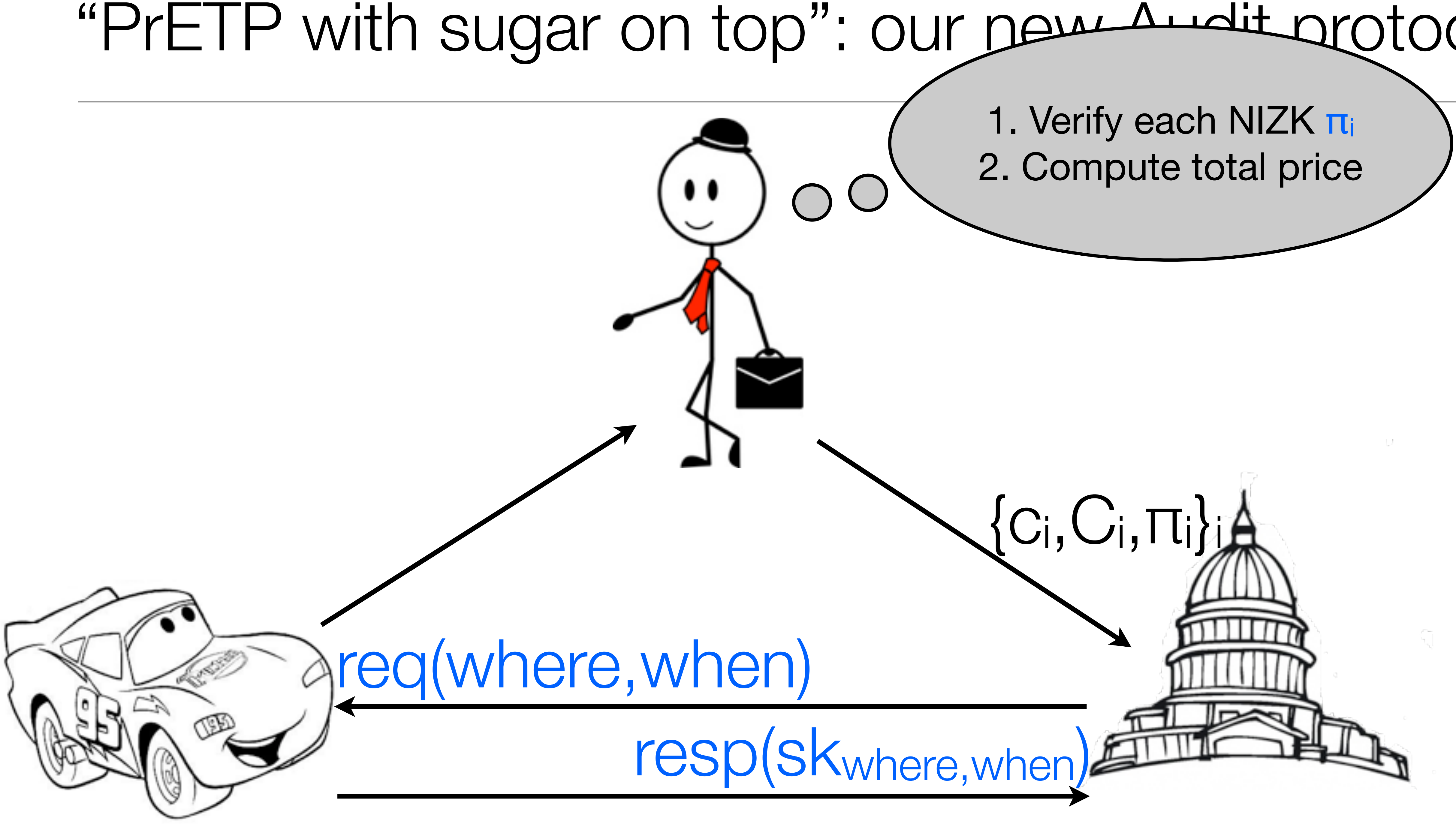


1. Verify each NIZK π_i
2. Compute total price

$\{c_i, C_i, \pi_i\}_i$

req(where,when)

“PrETP with sugar on top”: our new Audit protocol



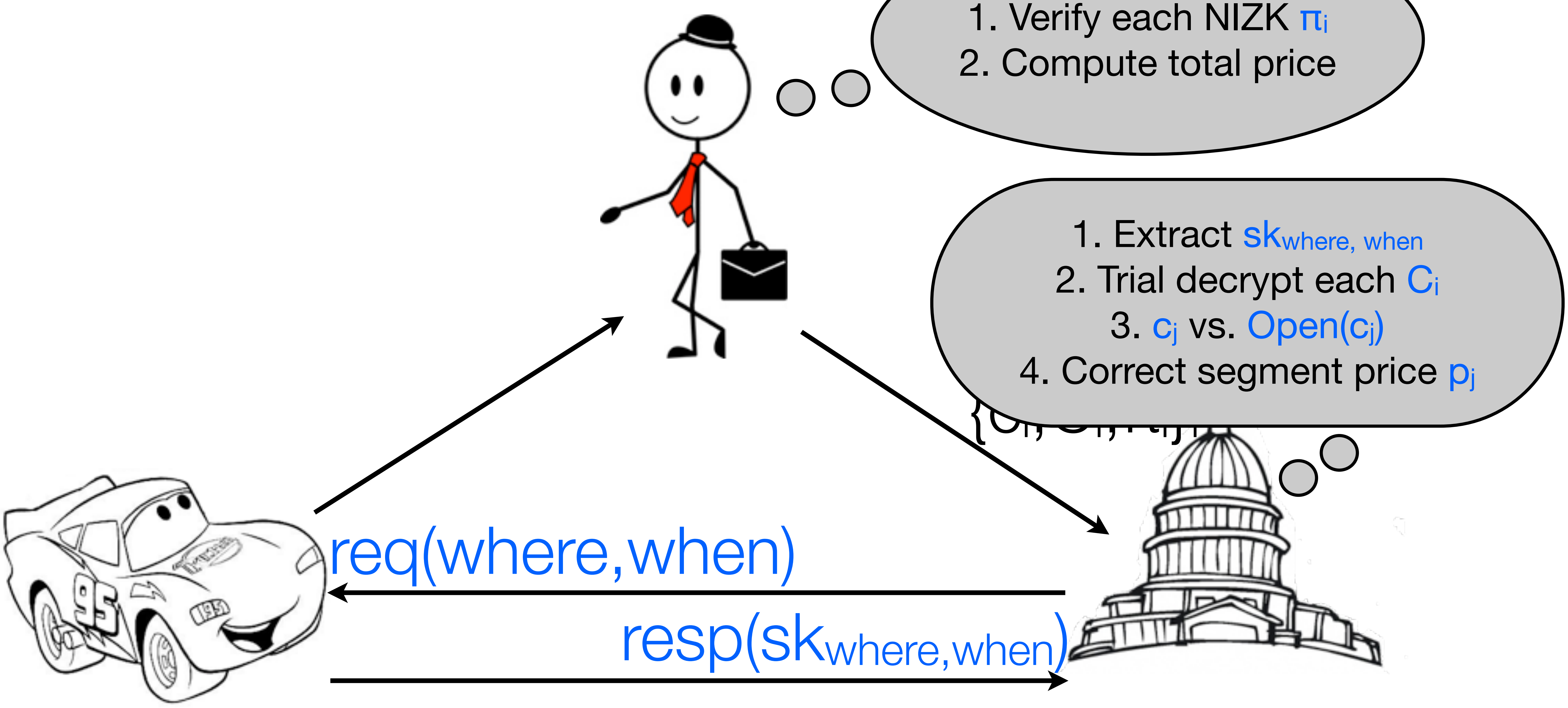
1. Verify each NIZK π_i
2. Compute total price

$\{c_i, C_i, \pi_i\}_i$

req(where,when)

resp($sk_{where,when}$)

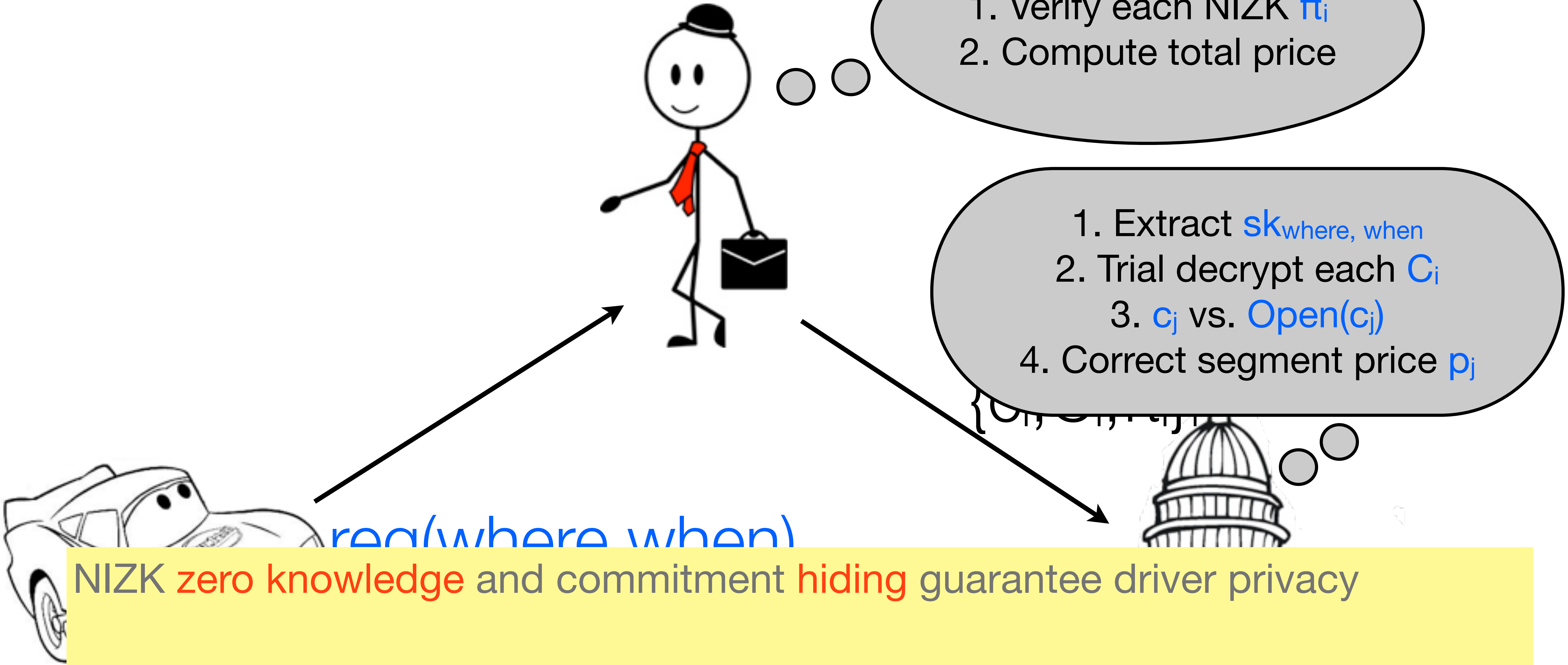
“PrETP with sugar on top”: our new Audit protocol



1. Verify each NIZK π_i
2. Compute total price

1. Extract $sk_{where, when}$
2. Trial decrypt each C_i
3. c_j vs. Open(c_j)
4. Correct segment price p_j

“PrETP with sugar on top”: our new Audit protocol



1. Verify each NIZK π_i
2. Compute total price

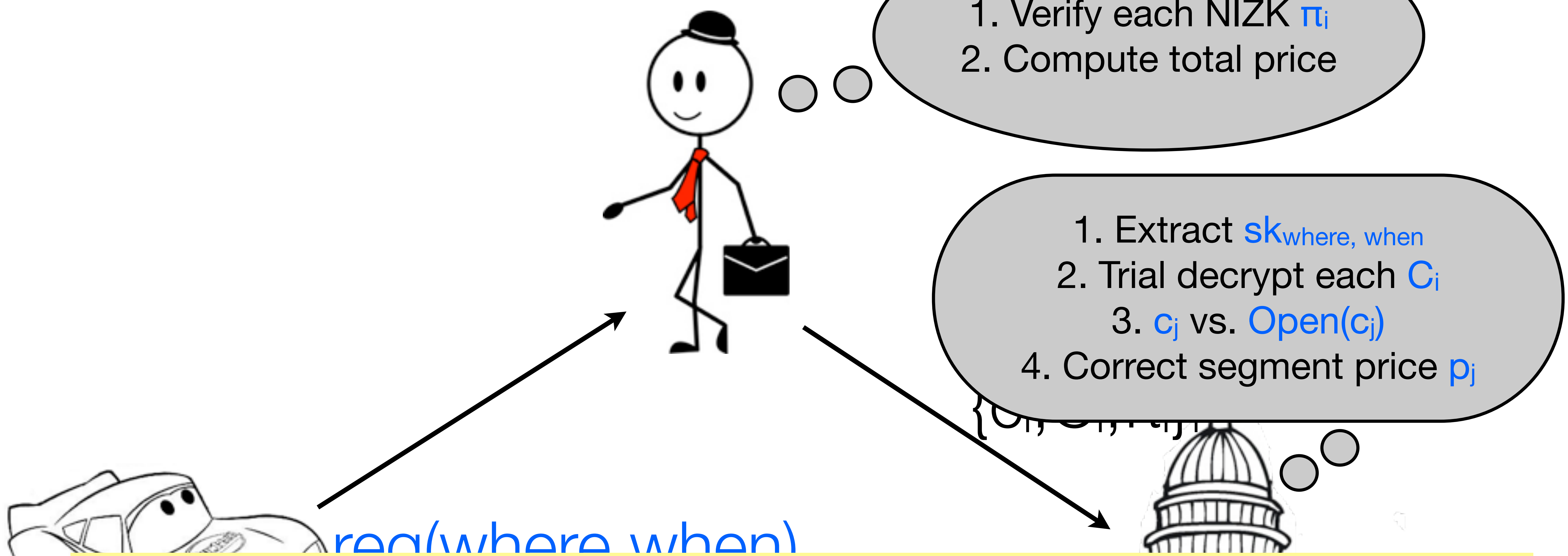
1. Extract $sk_{where, when}$
2. Trial decrypt each C_i
3. c_j vs. Open(c_j)
4. Correct segment price p_j

NIZK zero knowledge and commitment hiding guarantee driver privacy

NIZK soundness guarantees price p_i is in the right range (e.g., non-negative)

Commitment binding guarantees c_j is the right commitment for (where,when)

“PrETP with sugar on top”: our new Audit protocol



1. Verify each NIZK π_i
2. Compute total price

1. Extract $sk_{where, when}$
2. Trial decrypt each C_i
3. c_j vs. $Open(c_j)$
4. Correct segment price p_j

NIZK zero knowledge and commitment hiding guarantee driver privacy

NIZK soundness guarantees price p_i is in the right range (e.g., non-negative)

Commitment binding guarantees c_j is the right commitment for (where,when)

IBE blindness guarantees that driver doesn’t learn segment (where,when)

Outline

Cryptographic background

Milo

Evaluation

Implementation details

Milo’s performance

Conclusions

Implementation

Implementation

Used MIRACL [Scott] for blind IBE, ZKPDL [MEKHL’10] for commitments and NIZKs

Implementation

Used MIRACL [Scott] for blind IBE, ZKPDL [MEKHL’10] for commitments and NIZKs

Collected timing information on both a MacBook Pro (acting as the TC) and an ARM v5TE (acting as the OBU)

Implementation

Used MIRACL [Scott] for blind IBE, ZKPDL [MEKHL’10] for commitments and NIZKs

Collected timing information on both a MacBook Pro (acting as the TC) and an ARM v5TE (acting as the OBU)

When are blind IBE operations happening?

Implementation

Used MIRACL [Scott] for blind IBE, ZKPDL [MEKHL’10] for commitments and NIZKs

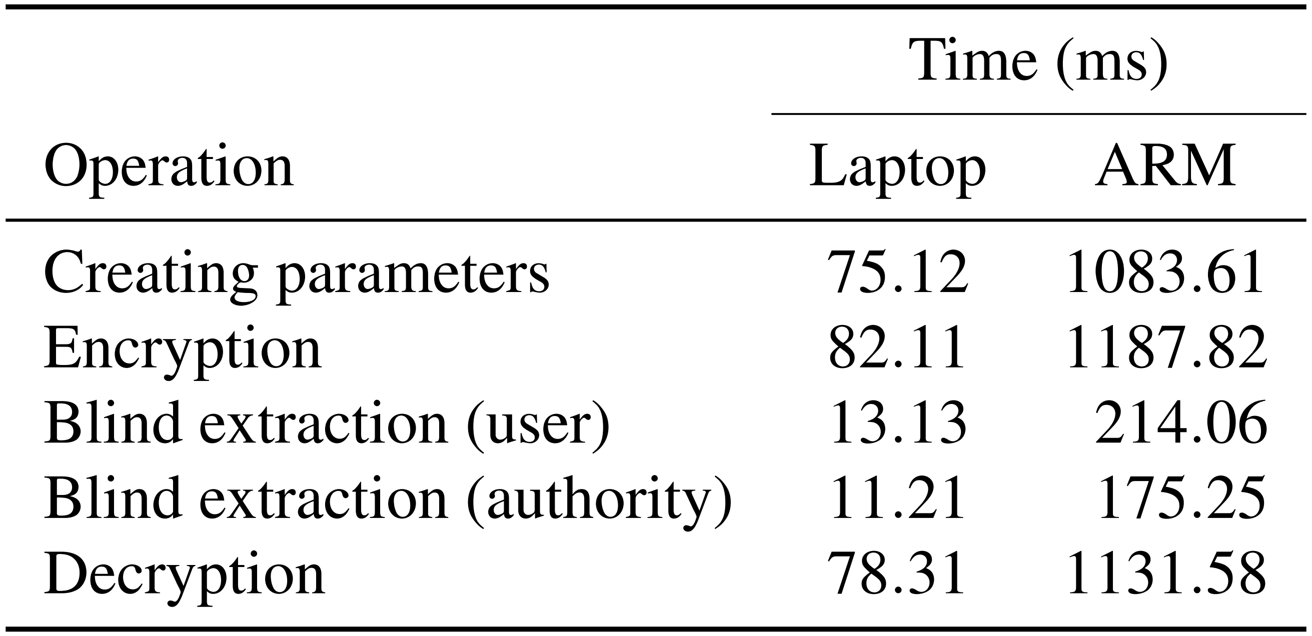
Collected timing information on both a MacBook Pro (acting as the TC) and an ARM v5TE (acting as the OBU)

When are blind IBE operations happening?

- Encryption: during Payment process
- Extraction: during Audit (OBU as authority, TC as user)
- Decryption: during Audit (TC needs to trial decrypt each ciphertext)

Various measurements: time and space

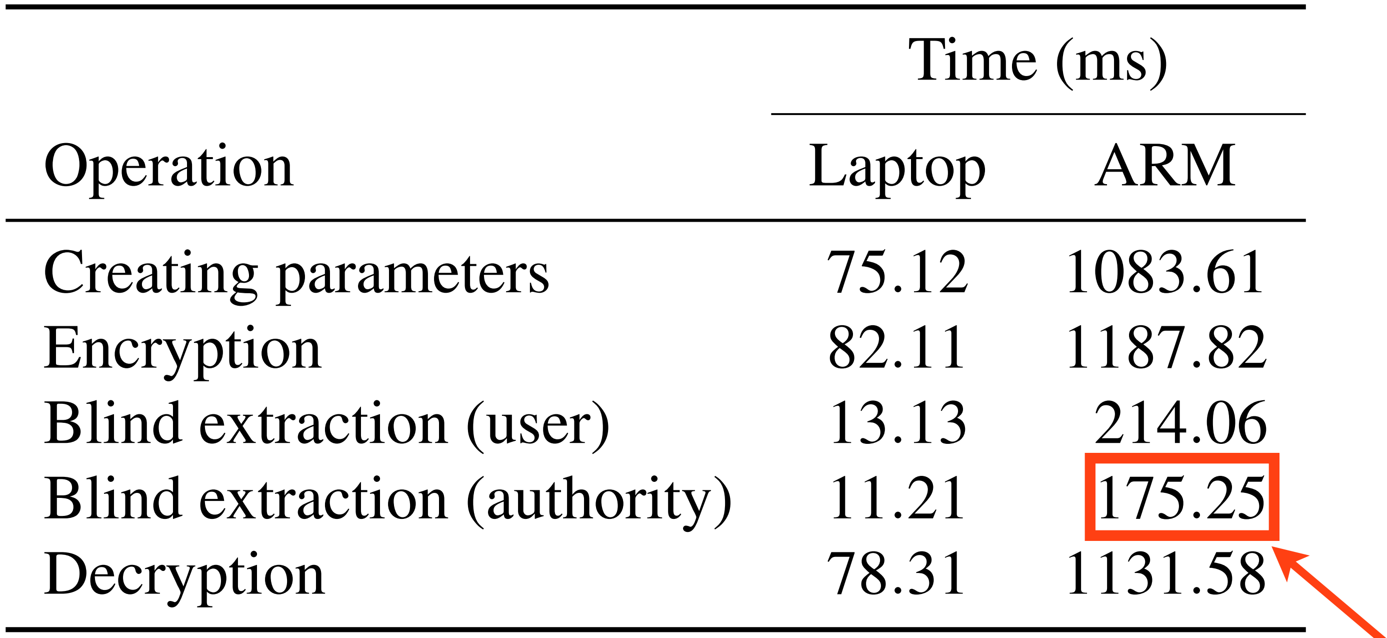
Various measurements: time and space



	Time (ms)	
Operation	Laptop	ARM
Creating parameters	75.12	1083.61
Encryption	82.11	1187.82
Blind extraction (user)	13.13	214.06
Blind extraction (authority)	11.21	175.25
Decryption	78.31	1131.58

Time for blind IBE

Various measurements: time and space



	Time (ms)	
Operation	Laptop	ARM
Creating parameters	75.12	1083.61
Encryption	82.11	1187.82
Blind extraction (user)	13.13	214.06
Blind extraction (authority)	11.21	175.25
Decryption	78.31	1131.58

Time for blind IBE

cost for OBU during
Audit is reduced

Various measurements: time and space

	Time (ms)	
Operation	Laptop	ARM
Creating parameters	75.12	1083.61
Encryption	82.11	1187.82
Blind extraction (user)	13.13	214.06
Blind extraction (authority)	11.21	$\boxed{175.25}$
Decryption	78.31	1131.58

Time for blind IBE

Object	Size (B)
NIZK	5455
Commitment	130
Ciphertext	366
Total Pay segment	5955
Audit message	494

Size for messages

cost for OBU during
Audit is reduced

Various measurements: time and space

	Time (ms)	
Operation	Laptop	ARM
Creating parameters	75.12	1083.61
Encryption	82.11	1187.82
Blind extraction (user)	13.13	214.06
Blind extraction (authority)	11.21	175.25
Decryption	78.31	1131.58

Time for blind IBE

Object	Size (B)
NIZK	5455
Commitment	130
Ciphertext	366
Total Pay segment	5955
Audit message	494

NIZK size dominates total size

Size for messages

cost for OBU during Audit is reduced

Various measurements: time and space

	Time (ms)	
Operation	Laptop	ARM
Creating parameters	75.12	1083.61
Encryption	82.11	1187.82
Blind extraction (user)	13.13	214.06
Blind extraction (authority)	11.21	175.25
Decryption	78.31	1131.58

Time for blind IBE

Object	Size (B)
NIZK	5455
Commitment	130
Ciphertext	366
Total Pay segment	5955
Audit message	494

NIZK size dominates total size

Size for messages

cost for OBU during Audit is reduced

Time for TC to perform Audit

Length	Time step	Segments	Time for TC (s)
1 mile	1 minute	2000	55.68
1 mile	1 hour	1000	33.51
2 miles	1 hour	500	10.45

Various measurements: time and space

	Time (ms)	
Operation	Laptop	ARM
Creating parameters	75.12	1083.61
Encryption	82.11	1187.82
Blind extraction (user)	13.13	214.06
Blind extraction (authority)	11.21	175.25
Decryption	78.31	1131.58

Time for blind IBE

Object	Size (B)
NIZK	5455
Commitment	130
Ciphertext	366
Total Pay segment	5955
Audit message	494

NIZK size dominates total size

Size for messages

cost for OBU during Audit is reduced

time to iterate dominates cost for TC

Time for TC to perform Audit

Length	Time step	Segments	Time for TC (s)
1 mile	1 minute	2000	55.68
1 mile	1 hour	1000	33.51
2 miles	1 hour	500	10.45

Outline

Cryptographic background

Milo

Evaluation

Conclusions

Conclusions

Conclusions

We presented Milo, a privacy-preserving electronic toll collection system

Conclusions

We presented Milo, a privacy-preserving electronic toll collection system

- Guarantees honesty even in the face of driver collusion
- Did so using blind IBE
- Found that computational overhead was manageable, significantly cheaper than certain alternatives

Conclusions

We presented Milo, a privacy-preserving electronic toll collection system

- Guarantees honesty even in the face of driver collusion
- Did so using blind IBE
- Found that computational overhead was manageable, significantly cheaper than certain alternatives

Future work:

- Possibly formalizing security definitions
- Find cheaper methods for achieving same security properties

Conclusions

We presented Milo, a privacy-preserving electronic toll collection system

- Guarantees honesty even in the face of driver collusion
- Did so using
- Found that c gnificantly cheaper tha

Thanks!
Any questions?

Future work:

- Possibly formalizing security definitions
- Find cheaper methods for achieving same security properties