I. ETP Background

ETP, generally
- Replace “flat road tax” by “distance related pricing”
- Pricing may depend on:
  - type of road
  - type of car (esp. emission)
  - time of day
- Aims, apart from fairness,
  - congestion steering/reduction
  - environmental impact reduction
ETP setup via OBU, planned in NL

- Cars get a special box, called **OBU**, for “on-board unit”
- ...which can at least:
  - determine its own position, via GPS or Galileo
  - communicate with backoffice, via GSM, Wifi, ...
  - calculate & store data
- Tariff map needed for fee calculation on basis of “trajectory parts”

Big Question

- Where to store trajectory information:
  - in the back office of the Pricing Authority (PA)
  - in vehicle (i.e. in OBU)
- Architectural decision about information flow
- But also about division of power in society (balance citizen – state)
- **Architecture is Politics!**
  (Mitchell Kapor, EFF founder)

Other concerns

- Reliability
- Cost-effectivity (aim in NL: overhead < 10%)
- Ease of use / transparency
- Fraud resistance (e.g. GPS can be manipulated)
- Ease of enforcement
- Ease of dispute resolution
- User acceptance, requiring trust!

This paper

- Focus on privacy aspects
- Novel solution
  (originally due to WdJ; joint elaboration)
- Main ideas will be presented; details may be implemented in several ways
- More general applicability, see later

Hostile users are to be expected
II. “Thin” and “Fat”

An on-board unit (OBU) is “thin” if:

- it does not itself calculate fees
- Obvious implementation, via centralisation:
  - OBU only computes (somehow) trajectory parts . . .
  - . . . and sends them to the back office of the PA . . .
  - . . . which calculates the fees (and sends bills)
- Easy enforcement via spot checks: take pictures and compare them to PA data

Pros and cons of centralisation

- Simple and transparent architecture
- Simple and cheap OBUs
- Extremely privacy-unfriendly
- Central database introduces high risks
  - data compromise may embarrass people
    (look for politicians who visited prostitute areas)
  - data protection relevant for personal security
    (e.g. whereabouts of people under threat)
  - single point of failure / bottleneck

An OBU is “fat” or “thick” if:

- it calculates fees itself
  (and passes them on to the PA back office)
- OBU must thus contain tariff map
  (which must be securely updated, occasionally)
- Spot checks complicated:
  - Two-way communication, while driving by
    requires integrity & authenticity of OBU
  - requesting most recent trajectory data
  - noticeable: generate warning to other drivers
Pros and cons of “fat”

- Privacy-friendly (via decentralisation)
- Complicated and expensive OBU
- OBU must be trusted
  (Successful attack on OBU is catastrophic)
- Complicated spot checks

III. Novel solution

Essentials

(I). Fee calculation does not require identity

- Anyone may do it, as long as it can be checked
- May be a (distributed) service

(II). Vehicles can commit to trajectories, without revealing them: via secure hash!

(III). Spot checks & fee checks via revealing of pre-images

Trajectory reporting

- Each OBU calculates, say each minute $i$, a trajectory part $TP_i$, from GPS+timing data
- 1440 TPs per day
- At the end it sends the “hash of the day” to the PA back office:
  $$h( h^2(TP_1) \parallel \cdots \parallel h^2(TP_{1440}) )$$

($h^2$ used for fee verification; not essential now)

- SMS size message, that completely fixes trajectories
Trajectory verification

- Assume photo spot check on day $d$ at time $t$
- PA asks for pre-image of hash of day $d$:
  $$h^2(TP_1) \parallel \cdots \parallel h^2(TP_{1440})$$
- and asks for pre-images of trajectory parts $TP_i$ around time $t$ — using fixed size of hashes
- Car owner may demand actual photo as evidence — in order to control/limit verification attempts

Fee calculation

- Assumptions:
  - tariff map is publicly available
  - tariff depends on time of day, type of road, 
  - fee for trajectory parts determined by public rules
  - “subfees” should be hidden, as much as possible
- Anyone can calculate fees, eg.
  - one or more (distributed) services
  - own PC, with open source software
  - OBU itself (“fat” version)

Fee reporting I

- Assume fee reports need to be sent quarterly
- The Pricing Authority (PA) should be able to:
  - check that subfees add up correctly
  - fees of selected trajectory parts are correct
  - ...in combination with trajectory verification
- In FAST paper realised via nested hashes
  Here via homomorphic encryption

Fee reporting II

- Assume finite group generator $g$
  Use $f \mapsto g^f$ as hash of fee $f$
- Multiplication of hashes is sum of fees:
  $$g^{f_1} \cdot g^{f_2} = g^{f_1+f_2}$$
- Fees are small numbers that can be tried out: random $R$ needed for blinding in $g^{f+R}$.
Fee reporting III; rough version

- A quarterly fee report by driver: \( \langle \text{Fee}, \ H, \ R \rangle \)
- where:
  - \( \text{Fee} \) is total due, over \( N = \pm 90 \) days
  - \( H \) is hash \( h(g_{a1} \parallel \cdots \parallel g_{aN}) \) of hashed day fees \( a_j \) (including random)
  - \( R \) is sum of blindings, with correct sum:
    \[ g^{\text{Fee}+R} = \prod_i g^{a_i} \]
  - \( a_j = \sum_{i \leq 1440} \text{Fee}_i + h(\text{TP}_i) \)
  - first hash \( h(\text{TP}_i) \) of trajectory part is both “blinder” and “binder”

Fee verification

- Assume PA wants to check time \( t \) at day \( d \) in fee report \( \langle \text{Fee}, \ H, \ R \rangle \), after spot check
- PA asks for pre-image day reports \( g^{a_1}, \ldots, g^{aN} \) and checks hash \( H \) and total sum, via multiplication (and \( R \))
- PA asks for day \( d \) all 1440 hashed amounts \( g^{\text{Fee}_i+h(\text{TP}_i)} \), which must multiply to \( g^{ad} \)
- PA asks trajectory parts \( \text{TP}_i \) around \( t \), checks occurrence, and corresponding fees \( \text{Fee}_i \).
- This can be fully automated!

IV. Perspective & conclusions

- Both privacy-friendly and “thin” possible
- But also “fat”, and more variations (including integration with commercial services)
- Simple spot checking (observation only; should not be noticeable)
- Failure of OBU protection not catastrophic: just increase intensity of spot checks
- Individual remains in control of own data
- Submission of daily hashes compulsory, to enable trajectory (and fee) verification
Perspective

- General issue “secure & privacy-friendly metering”
- Consumer uses valuables continuously; continuous monitoring by provider privacy-unfriendly architecture
- Alternative: consumer controls own usage data, but frequently sends hashes; provider monitors occasionally & checks hashes
- Applicable in many more situations
- What do we prefer? Architecture is politics!