Privacy-Preserving Speed-Limit Enforcement

Stefan Rass, Peter Schartner, and Patrick Horster
Institute of Applied Informatics, Universitaet Klagenfurt, Klagenfurt, Austria
Email: {stefan.rass, peter.schartner, patrick.horster}@aau.at

Alexander Abl
Kapsch TrafficCom AG, Klagenfurt, Austria
Email: alexander.abl@kapsch.net

***Abstract*—We describe a new speed-limit enforcement system with particular care for driver privacy and anonymity. Based on a novel application of identity-based encryption and homomorphic commitments, we demonstrate how a driver's identity can be processed in a way that is implicitly constrained to the mere detection of a speed limit violation. Using shared information that is distributed over the components of the system, we can assure protection of an identity even against insider attacks, up to the point where evidence of a speed limit violation is available.**

***Index Terms*—section control, speed-limit enforcement, traffic, safety, security, anonymity, privacy**

I. Introduction

Section control is the process of measuring the average driving speed on a road segment of fixed length. For that matter, two road-side systems, hereafter referred to as *gantries*, take notice of a bypassing vehicle and record the license-plate number and a time-stamp in order to recognize the vehicle at a later gantry. Radar, laser barriers, magnetic sensors or similar are standard methods to achieve this. The travel time is simply the difference between two time-stamps, and the average speed on the road segment can be calculated using the known distance between the two gantries. If this average speed exceeds the speed limit v_{max} on the road segment (section), then there must have been at least one point on the segment where the driver went faster than v_{max}, thus having committed a speed-limit violation. Notice this information is insufficient to tell *where* exactly the speed limit violation occurred, yet it proves that it must have occurred *somewhere* on the segment.

Related work: section control systems offer an interesting alternative to conventional Doppler-radar based measurements, which are limited to control a single point only, or radio-based transponder techniques [1], which require additional hardware in the vehicle (admittedly, this covers a wider range of violations than pure speed limit enforcement), which section control avoids. Most importantly, however, our system substantially improves on competing ones [2], [3] in terms of protecting the driver's privacy, as we will argue next. Besides the problem being highly debated in a legal context [4]-[8], the technical potential to protect the privacy of drivers has received surprisingly little attention so far.

A. Privacy vs. Legal Actions

Normally, the process of speed limit enforcement is done somewhat centralized, so that all evidence data is pooled at some server for processing. A particular challenge is the *protection of a driver's identity*, since unless there is evidence of a speed limit violation, processing of personal data is prohibited by law. Some patents on section control systems (e.g., [2][3]) try to hide the license-plate number (LPN) by encrypting it with a key that is derived from the LPN via a cryptographic hash-function. Without the additional use of random salt values to increase the entropy, this is far from sufficient to really protect an identity, since the search space is small enough to effectively brute-force search for the hidden identity. Speaking in numbers, even if we assume every person on the world as a car-owner, this makes approximately 8×10^9 possible license-plates, giving no more than $\log_2 8\times10^9 \approx 33$ bit of entropy. This is perfectly suitable for a brute-force search over all possibilities to find the pre-image of the hash-function, no matter if it is cryptographically secure. Unfortunately, the aforementioned patents refrain from using salt values, and are thus vulnerable to simple brute-force pre-image searches. Moreover, with fully homomorphic encryption not yet having come to much practicality, processing of data is infeasible once it is encrypted.

Our contribution: Legal regulations (to be detailed in the next section), impose somewhat paradoxical requirements of preventing personal data from processing without provable evidence of a speed-limit violation, whilst such evidence cannot be obtained without at least some data processing. This apparent contradiction between safety (speed limit enforcement) and security (data privacy) needs a resolution. To this end, we devise a new section control system for speed-limit enforcement that makes use of identity-based encryption in order to protect a driver's identity and allows explicit processing information only for the sake of a speed limit violation's detection. Notice that this is even an advantage over more powerful paradigms like fully homomorphic encryption,

Manuscript received January 2, 2014; revised March 27, 2014.

©2014 Engineering and Technology Publishing
doi: 10.12720/jtle.2.1.26-33

since by design, our system permits only the minimal set of operations needed to detect the incident. Similarly as for competing approaches, we use a key derived from some *vehicle identification information* (VID) to encrypt the related evidence data. Unlike hashing, however, we use a cryptographic *commitment* to derive the key, so that we can take advantage of homomorphic properties in order to match vehicles and calculate time differences hiddenly, without need to have the plain data available.

Before coming to the details in sections II and III, let us briefly summarize a selection of legal regulations for European section control systems, which will serve to derive the requirements to our newly developed system.

B. Selected European Legal Regulations

Throughout Europe, Austria and Germany can be taken as examples having very stringent laws concerning the processing of personnel data in the context of road safety enforcement. Therefore, we will use these federal legal regulations as reference from which we derive the requirements to the system. Austrian and German legal obligations, e.g., [6], [9]-[12], impose the following requirements on a section control system:

Requirement 1: Any data collected by a roadside system must only be used for determining whether or not a speed limit violation has happened. Any other or further processing is prohibited.

Requirement 2: Evidence data related to a driver's identity must not be stored permanently and must be destroyed immediately and without any traces if no speed limit violation has been discovered. Storage beyond this point in time is only permitted for those vehicles that have provably violated the speed limit.

Requirement 3: For the period in time in which the vehicle is between two roadside systems, the system must ensure that there is no way of extracting the license-plate number or any driver's identity from the data stored in the system.

Requirement 4: It must be impossible to discover that the same vehicle (even without knowing the license-plate number) has passed several roadside systems (this is to prevent the recording of travel profiles).

These requirements are mostly derive from the federal regulations of Austria and Germany, yet the European Union has published no less stringent obligations regarding privacy of data. We refer to [5], [7], [8] for an overview, including regulations that apply for the European Union in general. Also, [4] gives a list of more than 70 studies and publications related to road safety and speed limit enforcement.

II. Preliminaries

We use hybrid encryption to protect the driver's privacy up to the point where there is evidence of a speed-limit violation. Our solution is based on *identity-based encryption*, where the "identity", i.e. the public key, is a cryptographic commitment of the VID and the respective time-stamps. The VID is here considered as any information that uniquely identifies the vehicle within its locality, such as license-plate numbers, nationality, and perhaps additional features (we do not go further into this, as this may be have different legal definitions in different countries). Its size will be limited by the cryptographic setting and parameters, and concrete figures on how many bits are available (in our system) to encode the VID will be given in section III.A. We then take advantage of the homomorphic properties of the commitment in order to compare two public keys in terms of VID match and time-difference below a given threshold. This is expanded in detail in the following sections.

C. Parameters and Cryptographic Setting

For symmetric encryption, we will use standard AES, with key size $\ell_{\mathrm{AES}} \geq 128$. For the public-key encryption, we will use Boneh-Franklin identity-based encryption (IBE) [13], working in a subgroup of prime order q within an elliptic curve group $\mathrm{E}\left(\mathrm{GF}_{\mathrm{p}^2}\right)$, where $\mathrm{GF}_{\mathrm{p}^2}$ is a finite field of characteristic p (see [14] for details on how to make the appropriate choices). Our system employs hybrid encryption in which the IBE-scheme is only used to encrypt session keys. Therefore, the bitlength of q must strictly exceed the bitlength of the AES-key in charge, and be no less than recommended by standardization bodies like NIST. Adhering to [15] for that matter, the parameter q should thus have at least $\ell_{\mathrm{IBE}} = \max\left\{224, \ell_{\mathrm{AES}} + 1\right\}$ bits (other parameters are accordingly larger due to their construction).

Public identities (*keys*) for the IBE system will be created in a subgroup of prime order $\mathrm{q_G}$ within $\mathbb{Z}_{\mathrm{p_G}}$, where $\mathrm{p_G} = 2\mathrm{q_G} + 1$ is a (safe) prime. As for the IBE, we require $\mathrm{q_G}$ to have at least ℓ_{IBE} bits for security (which is ≥ 224 nowadays). Let g be a generator of the subgroup $\mathbb{Z}_{\mathrm{q_G}}$ within $\mathbb{Z}_{\mathrm{p_G}}$.

We denote an encryption of a string x using the AES with key K or IBE with public key PK as $\mathrm{AES_K}(\mathrm{x})$ and $\mathrm{IBE_{PK}}(\mathrm{x})$, respectively. A digital signature under a secret key SK is denoted as $\mathrm{Sign_{SK}}$. The signature algorithm may be the standardized DSA (digital signature algorithm), whose parameter choices and recommendations are the same as above (see [15]).

D. Assumptions

We assume the following available resp. doable when describing the system:

Assumption 1: Current road or weather conditions are authentically available (if traffic regulations depend on this), and can be compiled into the evidence data required for legal action.

Assumption 2: The gantries are equipped with synchronized clocks with sufficient precision for speed detection. In particular, it is reasonable (and in some countries also a legal obligation), to get the time from at least two independent sources. Usually, one of these is the GPS time (alternatively also international atomic time

©2014 Engineering and Technology Publishing

or radio time signals like DCF77), while the other comes from the network (network time protocol), or an internal clock.

Assumption 3: A gantry's sensory is sufficient to reliably detect the vehicle class and license-plate on a road section with perhaps multiple lanes, as well as it can collect any data required for potential legal action against the driver. For example, this could include the driver's face, time-stamps embedded in the photo, digital signature devices to create a proof of origin for the data, current weather conditions, etc.

Assumption 4: All components properly follow the protocols, but are potentially vulnerable to hacking attacks, except where stated otherwise. In that sense, we do not consider *actively cheating* insiders, but components (insiders) that may *leak* information (passively or without their knowledge).

III. High-Level Architecture

A high-level system diagram is displayed in Fig. 1. It consists of two gantries, denoted by G_1 and G_2, which are located at a known distance to each other. Furthermore, there is a human operator, whose duty is to finally judge the potential speed limit violation alerted by the gantries, before legal action can be taken. Notice that these parts in the system are considered as *vulnerable*, in the sense that an attacker may conquer a part of the system in an attempt to extract individual-related information from the system. We stress that the system is *not* designed to prevent an attacker from learning which vehicles pass a section (such information can trivially be obtained by anyone watching the road), but shall protect a driver from prosecution or investigation *before* evidence of a speed-limit violation is found.

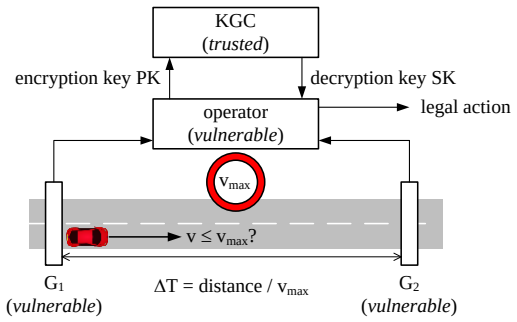

Figure 1. High-level architecture

The use of identity-based encryption (IBE) allows us conceal the link between the evidence data and the identity of the respective driver by means of hybrid encryption. The *key-generation center* (KGC) is responsible for providing the decryption keys upon a request from the operator. This is the *only* part of the system that needs to be *trusted* not to leak its secret master key to any other entity, except when the operator legitimately requests a secret decryption key. More specifically, the information required to associate evidence data with a person's identity is distributed over the components of the system, much like in a threshold cryptographic setting (though the technical implementation is fundamentally different here). The design is such that no component on its own can gain any information on the identities of any driver in the system. In brief, this separation of duties and information works as follows (cf. also Fig. 2.):

- A gantry collects the evidence data and symmetrically encrypts it with a randomly chosen session key K. This session key is IBE-encrypted using a key PK that is derived from the vehicle identification information VID. The VID and evidence data are reliably and safely discarded after the encryption, so that only the encrypted session key and encrypted evidence data remain inside the gantry. Decrypting any of these data items requires help from the key-generation center. Hence, the information inside the gantry is insufficient to link the evidence data to some person's identity.
- The key-generation center, on the other hand, would be able to decrypt the gantry's information, yet does not get to see it because the operator sits in between.
- The operator does get information from the gantries, however, cannot decrypt it unless the key-generation center agrees to help.

It follows that none of the involved components can, by itself, establish any link between an identity and the available evidence data.

The speed limit on the road section is v_{max}, which gives a minimum permissible travel time of $\Delta T = \text{distance} / v_{max}$. In detail, the process of capturing a vehicle and measuring its average speed runs as follows:

Step 1: A vehicle passes gantry G_1 at time t_1. It collects the identification information VID and creates a public key PK (see section III.A for details) that serves two purposes:

- It hides the identity VID from the eyes of an adversary, but
- Permits determining whether or not a speed limit violation has occurred.

The gantry stores all its public keys for a limited time in a temporary memory. A public key of age larger than ΔT units of time is discarded. The constant ΔT is the minimum permissible travel time between this gantry and the next one on the road section. For example, if the next gantry is 5 km ahead, and the speed limit is 130 km/h, then $\Delta T \approx 139$ seconds. Any vehicle passing the next gantry after this time cannot have gone faster – on average – than the speed limit on this section permits. Observe that we cannot detect peak speeds beyond the limit, taken on somewhere between the gantries. However, this is technically beyond the capabilities of conventional section control, and hence of no further interest here (see [16] for a more comprehensive introduction to speed limit enforcement).

Fig. 2 presents the process as a data flow diagram. Data items are shown as boxes, processing steps are boxes with rounded edges.

©2014 Engineering and Technology Publishing

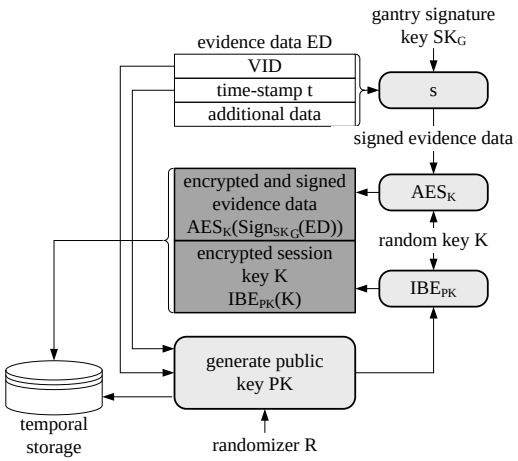

Figure 2. Data flow within a gantry

A. Key Derivation from an Identity

Let VID denote a vector of features that uniquely identify a vehicle. Usually, this comprises the license-plate number, plus additional properties as needed. When a vehicle passes a gantry, the license-plate and time of passage are recorded by the gantry, both of which are fused into a public key that *hides* this information, but permits certain processing of the committed data via homomorphic properties.

With the vehicle identification VID and a time-stamp t, the public key takes the form of a commitment as

$$PK := g^{([VID\|pad]\oplus R_j)\|t} \bmod p_G \qquad (1)$$

where pad is a (publicly known) padding and R_j is a randomizer that thwarts a brute-force search to open the commitment. We assume that the time-stamp t can be encoded using 64 bits (like coordinated universal time or GPS time). By construction, the element g has order q_G, which has ≥ 224 bits. This leaves a total of $224-64=160$ bits to encode the vehicle identifying information VID (remember that the license-plate number takes up no more than approximately 33 bits of this). Notice that we should nevertheless should leave some slack (for padding), as a wraparound modulo q_G within the exponent must be prevented.

Step 2: Gantry G_1 collects all potentially necessary evidence data (e.g. a picture of the vehicle and similar) and chooses a random session key K to symmetrically encrypt and digitally signs the evidence data, using its secret signature key SK_G.

Step 3: The chosen session key K is encrypted using the public key PK with some identity-based encryption scheme (see [17] for a well-written overview and introduction). The encrypted session key is attached to the encrypted evidence data and saved to the temporal storage device.

Step 4: Once the vehicle passes the next gantry G_2, this one does as G_1 and sends its public key PK' to G_1 for verification of speed limit obedience. This process is described in section III.B. Once G_1 notifies G_2 about a speed limit violation, both send their public keys along with the encrypted evidence data to the operator for manual verification and potential legal action.

Step 5: The operator files in a signed request at the key-generation center to obtain the respective decryption keys for the public keys obtained from both gantries. With the decryption keys, he opens the encrypted session keys and extracts the evidence data for manual verification. After then, legal actions can be initiated.

The choice of the randomizer is not constant and the randomizer is a shared secret between every two consecutive gantries. That is, two adjacent gantries always share a common randomizer, but no three gantries share the same value. Moreover, this value changes periodically. We come back to this in section III.C.

Each gantry stores a list of recent public keys in a ring-buffer, such that aging keys are automatically discarded (securely), after a predefined time to live. This assures that information is not stored permanently, so as to meet the respective legal requirement.

We stress that we *do not* rely on the binding property of the public key. The "commitment" is never opened, as it is only used to detect a speed-limit violation. Hence, calling the public key a commitment is technically correct, yet one must not interpret it as a commitment in the standard way these things are used.

B. Detecting a Speed-Limit Violation

When a speed limit violation shall be detected, gantry G_2 sends its newly created public key PK to its predecessor G_1 for verification. Now, suppose that gantry G_1 maintains a list of recent public keys, which we denote here as $L=\{PK_1,\ldots,PK_n\}$.

To check for a speed limit violation, G_1 now iterates through its list L and calculates the quotient

$$Q \equiv PK \cdot PK_i^{-1} \pmod{p_G},\ \text{for } i=1,2,\ldots,n,$$

and looks up the result Q as the key in a hashtable, whose structure (content) is displayed in Table I.

TABLE I. HASHTABLE FOR SPEED LIMIT VIOLATION DETECTION

key	$g^0 \bmod p_G$	…	$g^i \bmod p_G$	…	$g^{\Delta T} \bmod p_G$
value	0	…	i	…	ΔT

Technically, the gantry calculates the discrete logarithm of the quotient, which is feasible if and only if a speed limit violation has occurred *and* the two VID-values matched. To see this, let us take a closer look at the quotient Q. Suppose that PK as provided by G_2 encodes the value VID at time t, and PK_i as provided by G_1 contains VID_i at time t_i. Moreover, assume the

©2014 Engineering and Technology Publishing

randomizers to be equal (the case in which there has been a change of randomizers between passage of G_1 and G_2 is discussed later in section III.C. By (1), the quotient evaluates to

$$\left.\begin{aligned} Q &\equiv PK \cdot PK_i^{-1} \\ &\equiv g^{([VID\|pad]\oplus R_j)\|t} \cdot g^{-([VID_i\|pad]\oplus R_j)\|t_i} \\ &\equiv g^{x\|y} \pmod{p_G}, \end{aligned}\right\} \quad (2)$$

where the partitioning $x \| y$ is such that y has the same bitlength as the time-stamps. The value Q is the key to look up the value in the hashtable (Table I), leading to two possible cases:

Case 1: The table lookup comes back negative for all indices $i = 1, 2, \ldots, n$, i.e. none of the quotients has been found in the hashtable. This can only happen if $x\|y > \Delta T$. In turn, this either means that $x \neq 0$, which implies that $VID \neq VID_i$ for all i, or otherwise, if $x = 0$ in case of a vehicle identity match, then $y = t - t_i > \Delta T$. So, if $x = 0$, then the vehicle might indeed be the same, but it has not committed a speed limit violation. The latter is assured, thus there are no false-negative alarms raised by the system.

Case 2: The table lookup comes back positive for some index i. Then $x \| y$ has been obtained and necessarily, we have $x = 0$. This almost surely implies that the same vehicle has passed both gantries within the time limit ΔT, thus providing evidence of a speed limit violation. The likelihood of a false-positive verification can be approximated as follows: a false-positive can occur if and only if the exponents within PK and PK_i coincide. Assuming an approximate uniform distribution due to the randomization, we can estimate the chance for such a coincidence using the birthday paradox, which we will exemplify now.

For 5 km distance between two gantries, with a speed limit of 130 km/h, the minimal allowed travel time would be $\Delta T \approx 139$ seconds. Given a temporal resolution of 0.01 seconds, the hashtable needs to store ≈ 13900 entries (notice that this number is independent of the chosen encoding of the time-stamp). According to nowadays cryptographic standards (cf. [15], [18]), the exponent would need at least 224 bit, so that the chance for a false-positive among 13900 entries with 224 bits each, by the birthday paradox, roughly 3.58×10^{-60}. This can be considered negligible for practical purposes, especially as the ultimate judgment is up to a human operator.

C. Randomizer Synchronization

The randomizer within the public key (1) is needed to prevent an adversary from brute-force searching over all license-plates and possible time-stamps. It must be synchronized between two neighboring gantries. So, particular care has to be taken if the randomizer is switched after a vehicle passes a gantry G_1 and before it passes the next gantry G_2.

Randomizer switching is needed for security reasons, since keeping this information constant over the entire lifetime of the system would allow an adversary to recognize identical VIDs over a certain period of time (perhaps drivers regularly taking a route to work or similar). So, we propose switching the randomizer periodically. In the following, let t_{switch} be the time when R is changed into R'. The sequence of randomizers is a hash-chain and thus pseudorandom:

$$R_{j+1} = \mathrm{Hash}(R_j) \quad \text{for } j = 1, 2, 3, \ldots, \quad (3)$$

where the validity period of R_j is strictly larger than ΔT on the particular road section (for reasons that will become obvious below). Alternatively to a hash-chain, one can also use cryptographic key agreement protocols to safely establish a fresh randomizer between any two gantries. However, this requires additional communication and maintenance of respective signature keys within the gantries and perhaps unnecessarily complicates the system beyond our "offline solution". Nevertheless, synchronization from scratch is inevitable in case of power-failure or at system startup.

If the randomizer has been switched, then the check via calculating the quotient (2) will fail even though the same vehicle ID is hidden inside the public key. This is tackled in the following way: first, the switching interval must be much larger than ΔT, which is the minimal travel time between the two gantries. Second, within a period of $[t_{switch} - \Delta T, t_{switch}]$, the gantry must use the current *and* previous randomizer for checking, in order to recognize an identical vehicle despite the yet different randomizer.

To properly distinguish and resolve each possible case, Fig. 3. displays the different scenarios and respective actions taken by each gantry.

IV. Security

Legal obligations require personal information to remain protected and hidden, unless there is evidence of a speed limit violation. An attacker in our system is an external entity with access to the system in an attempt to discover the personal data from the drivers. Note that our security treatment here assumes the system components to follow the protocol specifications, yet all internal information from a component may leak in case of a successful intrusion.

We complete the system description by discussing several attack scenarios and deriving appropriate counter-measures along the way. Hence, this section is to be considered with a strong reference to sections II and III, which it will complement partially. First, we note that all communication within the system is encrypted and digitally signed, so as to avoid straightforward insertion, blockage or manipulation of data packets on the communication channels. We will therefore not further discuss the required public key infrastructure in the background, and focus on *insider attacks* in the following.

©2014 Engineering and Technology Publishing

As outlined above, the privacy of a driver is protected by means of distributing the information properly across the system so as to avoid a single instance becoming able to disclose a driver's identity hidden in some public key. To see this, we distinguish several scenarios of an adversary gaining access to one of the components. As an overview and for later reference, Table II displays who is in possession of which data item. The column "covertly known" shows data items that are technically in possession of the respective entity, yet are buried inside tamper-proof devices to protect them from unauthorized access. The table is exhaustive w.r.t. what an entity knows; hence anything *not* listed is unknown to the respective component, except for one's own public encryption or signature verification keys. We refer to this list in order to illustrate the security of our system in the upcoming paragraphs.

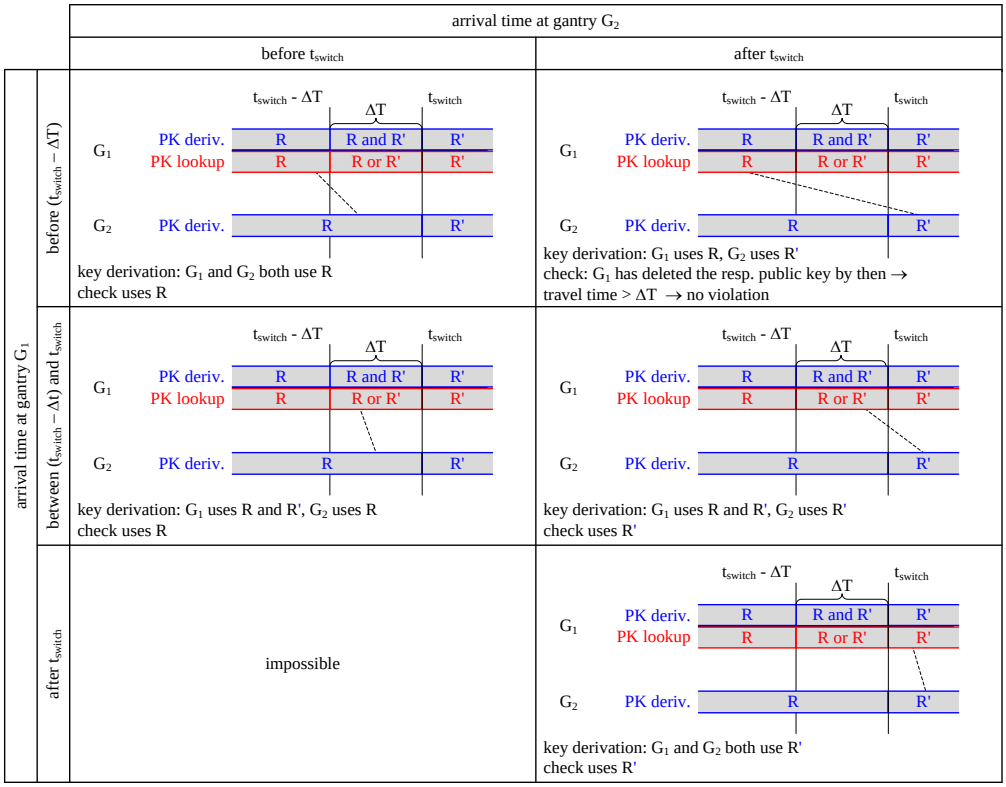

Figure 3. Synchronization and usage of randomizer

A. *Trust and Security Requirements for the Components*

From Table II, it is immediate to derive the security requirements concerning each component and parameter of the system. For the sake of brevity, we do not elaborate on the visibility of keys or parameters, as all system parameters and public keys are to be authenticated yet require no concealing, as opposed to the respective secret keys (especially the KGC's master-key), which must be kept confident. In brief, public key cryptography is used mostly to encrypt and authenticate all communication from the gantries to the operator and from the operator to the KGC (and both stages back). These are the only keys that require protection in the system, as the IBE's secret key that corresponds to a vehicle's specific public key PK does not even exist in the system until the operator requests it from the KGC.

Physical protection of secret keys in hardware (e.g., by putting certain computations inside a smartcard or a hardware security module) is only required to protect covertly known information. As Table II shows, the gantry as such needs no protection beyond the storage and protection of the secret signature keys. Likewise, the operator's only crucial piece of information is the signature key that enables him to query the key-generation center. Equally evident and due to the use of identity-based encryption, is the fact that the key-generation center is actually *the* neuralgic spot in the system. Compromising the KGC in fact defeats all security assurances that our scheme can provide, which implies that the KGC should reside inside high-security premises, say a full-fledged data center with strong physical and logical security precautions.

TABLE II. DATA OWNERSHIP

Component	Known information	Covertly known information
Gantries	public keys, encrypted session keys, encrypted evidence data	secret signature creation keys
Operator	the KGC's public encryption key	secret signature creation keys for requests to the KGC
KGC	operator's public key for signature verification	IBE master-key to create decryption keys, secret signature keys

©2014 Engineering and Technology Publishing

B. Vulnerable System Components

Insecure Gantries: Inside a gantry, an adversary would only find the encrypted session key, encrypted evidence data and public keys. However, no cryptographic keys like the secret signature key SK_G are accessible, provided that these cryptographic operations happen within a tamper-proof device. Decryption of these data items is impossible without knowledge of the IBE master-key, which is only known to the key-generation center.

Moreover, extracting the identity from the public key (1) is infeasible, provided that nowadays cryptographic standards are obeyed when implementing the system. A brute-force attempt to extract the VID from a public key of the form (1) will fail thanks to the randomizer R. This prohibits a brute-force search overall VIDs, since R effectively hides $\mathrm{VID} \| \mathrm{pad}$. Accessing the current randomizer is prevented if the public-key is generated inside a tamper-proof device (e.g., hardware security module; c.f. e.g. [19]). Hence, the value R never becomes visible to the adversary, and as well must not be stored within the gantry. It is as well subject to safe destruction.

Even if a randomizer gets somehow into the hands of an adversary, this does not help to open any past commitments, since the adversary would need to traverse the hash-chain (3) *backwards*. Nowadays standardized hash-functions, such as SHA-3 (Keccak), are all pre-image resistant, so this attack will practically fail.

Malicious operator: The operator cannot decrypt evidence data at will, because the gantry only submits the respective information in case of a suspected speed limit violation. Hence, the operator cannot ask for decryption keys for records related to drivers who obeyed the speed-limit. Otherwise, however, it is his duty to decrypt the information and disclose the hidden identity. An adversary attempting to impersonate the operator will not succeed, since all queries to the KGC must be digitally signed with a key stored within a personal smartcard for the operator.

Malicious key-generation center: The KGC stores the master-key and can therefore theoretically decrypt all the information contained in a gantry. However, it cannot directly access this information within the gantries, since it must use the operator as proxy.

Linkability and travel profiles: Using randomized commitments has the positive effect of making two public keys for the same VID look different, even though the time-stamps might differ only slightly. As an example, consider a person driving to work, roughly at the same time every day. Without a randomizer, the respective public keys could be associated quite efficiently, allowing an adversary to recognize the same vehicle based on the public key only.

The resilience of the system against travel profile extraction relies on the usage of common randomizers between two adjacent gantries only. Consider two public-keys created on two *different* but adjacent road-sections for the *same* vehicle VID. So, our vehicle passes G_1, G_2 and G_3 in exactly this order. From equation (1), we get the two public keys $PK_1 = g^{\lceil (\mathrm{VID} \| \mathrm{pad}) \oplus R \rceil \| t_1}$ and $PK_1 = g^{\lceil (\mathrm{VID} \| \mathrm{pad}) \oplus R' \rceil \| t_3}$, where t_1, t_3 are the times at which the vehicle passes G_1 and G_3, and R, R' are the different randomizers shared between G_1 and G_2 (randomizer R) and between G_2 and its successor roadside system G_3 (randomizer R'). No algorithm is known to extract VID from either public-key, and algebraic manipulations in the exponent are limited to scalar multiplication and addition (based on the difficulty of the discrete logarithm problem and the Diffie-Hellman problem). Both operations appear insufficient to extract or brute-force search for VID. Hence, it is infeasible to establish a relation between PK_1 and PK_3 in order to derive a travel profile for any particular vehicle.

Timing attacks: An easy way to trick the system into revealing all the driver's identities would be manipulating the local clocks towards making all drivers apparently exceed the speed limit. This attack is usually avoided by the (legal) requirement and availability of at least two independent sources for the time, one of which would be GPS, so that no manipulation other than interference with the signal would be expected. However, the clocks would in any case be required to remain synchronized over the lifetime of the system, which is yet another reason why GPS time is so attractive for that matter. An attacker could, potentially, interfere with the gantry to cut it off the GPS signal, in order to manipulate the second (perhaps external) time source (e.g., coming from the network). To thwart this, one may go with a local high-precision clock residing inside the tamper-proof components of the gantry, and acting as a reliable source to bridge periods of such adversarial interference with the system. Moreover, timing attacks can hardly be launched on specific vehicles, but would automatically apply to all vehicles passing the gantry over a certain period. Such incident would likely raise an alarm at the operator's side, due to an unusually high number of speed-limit violations. Hence, the timing attack would probably not go undetected for long.

V. Outlook

Speed limit enforcement under the requirement of not processing any data before a violation has occurred is a challenge under seemingly contradictive constraints. We presented a possible solution by a novel application of identity-based encryption and homomorphic commit-ments, which may be of independent interest.

Several modifications to the above scheme are imaginable. For example, one may simplify the system by having the session keys encrypted under the KGC's public key. In that instance, the KGC may simply decrypt the session key rather than return a secret key that is specific for the driver. However, this destroys the notable feature of our system, by which the decryption key to open a driver's personal record *does not exist* unless a speed-limit violation has occurred. Hence, legal

©2014 Engineering and Technology Publishing

compliance may become an issue under such a simplification. In another variation, IBE may be replaced by certificateless encryption [20], so as to relax the trust assumption on the KGC, and to improve security by sharing the decryption abilities between the KGC and the operator. An exploration of this extension is subject of future considerations, as is the possibility of doing the KGC's computations in a distributed fashion by multiparty computation [21].

REFERENCES

[1] A. Blumberg, L. Keeler, and A. Shelat, "Automated traffic enforcement which respects driver privacy," in *Proc. Intelligent Transportation Systems*, 2005, pp. 941–946.

[2] G. Fally, L. Rohrecker, and G. Schreiber, "Verfahren und anordnung zum anonymisierten erfassen und auswerten von fahrzeugreisedaten," patent DE 102005036562 A1, German, February 15, 2007.

[3] J. Birchbauer, J. Hatzl, and M. Hennecke, "Verfahren und vorrichtung zur erfassung einer geschwindigkeits ubertretung eines Fahrzeugs," patent DE 102007059 346A1, German, February 15, 2009.

[4] European Commission. Road Safety References. (October 29, 2013). [Online]. Available: http://ec.europa.eu/justice/data-protection/index en.htm

[5] European Commission. (October 2013). Protection of personal data – justice. [Online]. Available: http://ec.europa.eu/justice/data-protection/indexen.htm

[6] Austrian Federal Chancellery, "Federal act concerning the protection of personal data (DSG 2000)," *Federal Law Gazette I No. 165/1999*, May 2012.

[7] European Parliament. (October 2013). Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data. [Online]. Available: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:31995L0046:EN:HTML

[8] European Parliament. (October 2013). Directive 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector (Directive on privacy and electronic communications). [Online]. Available: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32002L0058:EN:HTML

[9] Austrian Data Protection Commission. (October 2013). Austrian Laws and Ordinances. [Online]. Available: http://www.dsk.gv.at/site/6274/default.aspx

[10] L. Bauer, "Section control – no violation of right to data privacy by automatic speed measuring systems of road traffic," *ICL Journal*, vol. 1, no. 1, pp. 59–63, 2007.

[11] F. Albrecht, "Section control in deutschland – rahmenbedingungen für die einführung der abschnittbezogenen geschwindigkeitsüberwachung," *SVR*, vol. 5, pp. 161-167, 2009.

[12] M. Hoffer, "Section control – datenschutz bei verkehrsueberwachung – bitte warten?" *Deutsches Autorecht (DAR) – Rechtszeitschrift des ADAC*, vol. 79, no. 1, pp. 23–26, 2009.

[13] D. Galindo, "Boneh-franklin identity based encryption revisited," in *Proc. 32nd International Conference on Automata, Languages and Programming*, Berlin, Heidelberg: Springer, 2005, pp. 791–802.

[14] D. Boneh and M. Franklin, "Identity based encryption from the weil pairing," *SIAM J. of Computing*, vol. 32, no. 3, pp. 586–615, 2003.

[15] D. Giry. (October 2011). Bluecrypt – cryptographic key length recommendation. [Online]. Available: http://www.keylength.com/

[16] Committee for Guidance on Setting and Enforcing Speed Limits, "Managing speed – review of current practice for setting and enforcing speed limits," Transportation Research Board, Special Report 254, 1998.

[17] L. Martin, *Introduction to Identity-Based Encryption*, Artech House, 2008.

[18] E. Barker, W. Barker, W. Burr, W. Polk, and M. Smid, "NIST SP800-57: Recommendation for key management Part 1: General (Revised)," *National Institute of Standards and Technology, Tech. Rep.*, Mar 2007.

[19] Rittal. Modul Safe. (2013). [Online]. Available: http://www.rittal.de

[20] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *ASIACRYPT*, 2003, pp. 452–473.

[21] M. Hirt, "Multi-party computation: Efficient protocols, general adversaries, and voting," Ph.D. dissertation, ETH Zürich, 2001.

Stefan Rass received a double master degree in mathematics and computer science in 2005, and got a PhD in mathematics from the Universität Klagenfurt in 2009. His main research topics cover security infrastructures, risk management, complexity theory, game-theory, statistics, (applied) system security and cryptography.
He has been a member of various national and international projects, and is currently a member of the System Security Research Group at the Universität Klagenfurt.

Peter Schartner received his master degree in Telematics (with a focus on information security) from the Technical University of Graz in 1997 and his PhD in computer science (with a focus on security tokens) from the Universität Klagenfurt in 2001. His research interests include key management, security infrastructures and applications for security tokens, especially smartcards and smartphones. He is currently an associate professor in the system security research group at the Universität Klagenfurt.

Prof. Dr. Patrick Horster is the head of the system security research group at the Universität Klagenfurt. His scientific and research interest includes key-management, computer science and system security; in particular public-key-infrastructures as well as the design and analysis of cryptographic mechanisms.

Alexander Abl is the head of a group of requirements engineers and business analysts in the Kapsch TrafficCom system engineering center developing solutions for the electronic tolling business. In addition to Alexander is lecturing at the University of Applied Sciences in Klagenfurt in the area of Logistics and Transportation. Before joining Kapsch, Alexander worked for Frequentis (Vienna/Austria) as solution architect. Alexander gained considerable experience in the field of traffic telematics (including vehicle tolling), fleet management and command and control center engineering especially in the public safety sector. Alexander Abl studied Communication Engineering at the University of Applied Sciences Klagenfurt and the Universität Klagenfurt.

©2014 Engineering and Technology Publishing