Interoperable Privacy-Aware E-Participation within Smart Cities

Constantinos Patsakis, University of Piraeus
Paul Laird, Michael Clear, and Mélanie Bouroche, Trinity College Dublin
Agustí Solanas, Rovira i Virgili University

Sharing data via information and communications technology is a fundamental goal of e-government, but data fusion and data mining could reveal sensitive personal information. A proposed cryptographic protocol guarantees citizens’ privacy through data aggregation and fosters e-participation in a scalable and interoperable way.

E-government uses information and communication technology to facilitate access to information, services, and expertise and ensure citizen participation in and satisfaction with the governmental process. ICT simplifies and expedites transactions among people, businesses, and government agencies in a transparent and cost-efficient way.

The adoption of e-government offers benefits at different levels, but it arguably has the greatest impact at the municipal level, that closest to the people. This can be seen in the growing popularity of smart cities, those "strongly founded on information and communication technologies that invest in human and social capital to improve the quality of life of their citizens by fostering economic growth, participatory governance, wise management of resources, sustainability, and efficient mobility, whilst they guarantee the privacy and security of citizens." ICT enables citizens to play an active role in collecting and sharing information—in other words, as intelligent sensors. Thanks to such e-participation, local governments can make more informed decisions and thereby improve quality of life. In this model, illustrated in Figure 1, information flows from the citizens to the government and then back to the citizens in a closed loop.

To encourage municipalities to adopt e-government and foster e-participation, it is necessary to provide citizens with the ability to contribute detailed information efficiently using a wide range of interoperable devices and protocols. The data’s granularity and immediacy, as well as the vast amount of information sources, require infrastructures with very demanding financial, technical, and personnel requirements. However, an additional, nonmone
tary, cost of even greater concern to citizens is privacy.

Citizens might be reluctant to share information
depending on its nature. However, even if this is not the case, a great deal of private information could be revealed through data mining and data fusion techniques. In many instances, some data considered redundant could reveal sensitive personal details when fused with other information. E-government services therefore must be privacy-aware for citizens to trust and embrace them.\textsuperscript{10,11}

Toward this end, we have developed a cryptographic protocol designed to manage the huge amount of personal information that could be generated through e-participation in a scalable, interoperable manner that guarantees citizens’ privacy.

**CHALLENGES AND OBJECTIVES**

Any e-participation solution for municipalities in which numerous citizens will interact with e-government faces four challenges:

- **Performance:** The system’s algorithms and infrastructures should be able to accommodate time-dependent data.
- **Interoperability:** Citizens should be able to interact with the system using various types of personal devices that follow open e-government standards.
- **Privacy:** The system should guarantee the privacy of personal information it collects and stores.
- **Scalability:** The system must be able to cope with a large and progressively growing number of citizens.

Several researchers have examined these challenges. For example, one recent paper describes the three core functionalities of municipal-scale ICT platforms as urban communications abstraction, unified urban information models, and open urban services development.\textsuperscript{12} While these functionalities highlight the need for interoperability, their urban character underscores the importance of scalability. Performance and privacy are combined under the umbrella of communications abstraction, where both are considered the default for communications because every user presumes that his or her provider will try to increase performance without decreasing privacy. Unfortunately, the three well-known Vs—volume, velocity, and variety—impose serious limitations that prevent current solutions from addressing all of the challenges simultaneously. For instance, a solution can be interoperable, scalable, and high-performing but not provide privacy, or it might provide performance, privacy, and interoperability but not scale efficiently.

We believe that e-government can manage fine-grained data in a manner that addresses all four challenges at the same time. The system can either aggregate data or provide anonymous statistics, both of which are invaluable, especially if the data can be delivered in or near real time. Our proposed cryptographic protocol focuses on e-government services that exploit urban sensing information stemming from the fixed sensors of a smart city’s infrastructure, such as weather, pollution, traffic, and mobile sensors pre-equipped with GPS, as well as information gathered from citizens through participatory sensing. Information from fixed sensors can be considered public, but data gathered from citizens needs to be anonymized without decreasing the data’s utility or disclosing citizens’ identities. The protocol can also handle sensitive time-series data.

One of the main goals of our proposed scheme is openness, which provides the necessary transparency to stakeholders to justify why certain decisions were made and motivates citizens to participate more actively in city management. It also allows interoperability, as private companies can use open standards to develop their solutions; even if the underlying technology remains secret, other stakeholders can easily interoperate with the data. If a scheme is also privacy-aware, it allows the publication of open government data, leading to further service development and exploitation by organizations, companies, and individuals. Thus, it could be said that data should
be open—accessible to everyone—and at the same time private in the sense that citizens’ privacy is preserved.

**MOTIVATING SCENARIO**

For context, consider a city with several commercial areas containing numerous shops and restaurants that provide goods and services. The municipality would like to know how many people are in each area to better distribute its resources, such as police and street cleaners, and serve its citizens in real time.

In this scenario, businesses might need to inform the municipality about their current number of customers. However, doing so might provide their competitors with vital information to plan campaigns that could direct customers elsewhere, so some businesses might be reluctant to provide this information.

With our proposal, businesses could privately share the number of their clients. The system could then aggregate this data and reveal the total number of customers in a given area of the city in real time to the proper municipal authorities. In this way, the city could better allocate its resources depending on citizens’ activities while keeping private individual business’s customer numbers.

**CRYPTOGRAPHIC PROTOCOL**

Although there are many variants of cryptographic primitives, certain structures have proved very efficient or secure and are widely used by many algorithms. One of these is the elliptic curve, a well-known algebraic structure that generates an abelian group. The main advantage of an elliptic curve is that it introduces—for example, when inverting scalar multiplication—is so high that an adversary cannot solve it without background knowledge. Therefore, cryptosystems based on elliptic curves are not only efficient but also very secure. Moreover, their algebraic structure enables the exploitation of pairings, which allow the transition of properties from one group to another through a homomorphism of the two groups.

In earlier work, we proposed a protocol, here referred to as PCL, that lets individual users send their data to another entity—the aggregator—that performs from the same public-key structure enables the exploitation of pairings, which allow the transition of properties from one group to another through a homomorphism of the two groups.

An advantage of PCL is that it removes pairings from the multi-round variant of the KDK protocol. PCL allows a bounded number of rounds \( t \) to be performed from the same public-key information that each user shares. However, \( t \) depends on the acceptable collusion tolerance \( t \leq n \), or how many of the \( n \) users should collude to recover another user’s value. PCL is \( t \)-private for at most

\[
\ell = \left\lfloor \frac{n-t}{2} \right\rfloor
\]

rounds, which means that to recover the value that an individual submitted, \( t-1 \) users must collude. For example, for \( t = n/3 \) (Byzantine tolerance, a very well-known bound) and \( n = 100 \), 33 rounds can be executed before the users have to publish a new key.

Another advantage of PCL is that it only relies on the Decisional Diffie-Hellman (DDH) assumption in some cyclic group \( G \) of prime order \( p \). DDH is one of the strongest cryptographic assumptions, like single-round KDK. A cyclic group is one generated by a single element, called a generator, which is denoted by \( g \) for the group \( G \).

The group \( G \) may be instantiated by different types of groups such as an elliptic curve over a finite field. An elliptic curve offers more efficiency for the same levels of security compared to other groups. In fact, multi-round KDK requires elliptic curves due to its heavy reliance on pairings, which makes it far less efficient in practical applications than PCL.

One can assume that KDK is based on a fixed matrix \( A \) with its entries being \(-1, 0, 1\), determining the exponents used to compute a value \( w \). More precisely, KDK defines a matrix \( A \) of the following form:

\[
A = \begin{bmatrix}
0 & -1 & -1 & -1 & -1 \\
1 & 0 & -1 & -1 & -1 \\
1 & 1 & 0 & \cdots & \cdots \\
\vdots & \vdots & \ddots & \ddots & \ddots \\
1 & 1 & 1 & \cdots & 0
\end{bmatrix}
\]

In the first step of the protocol, each user \( U_i \) publishes a value \( u_i = g^w \), keeping the value \( x_i \) secret. Then, each user has to compute a value \( w \). To compute this value, each user \( U_i \) raises user \( U_j \)’s public key \( u_j \) to the power that cell \( A_{i,j} \) indicates. Then, \( w_i \) is defined as a product of these values. The user then publishes the following value:

\[
v^r_i = w^r_i \cdot g^{m_i}
\]

where \( m_i \) is the value that user \( U_i \) wants to send for aggregation in round \( r \). If the aggregator multiplies all the \( v_i \) values, he or she will compute \( g^{\sum m_i} \). Therefore, if the
value $\sum m_i$ is well bounded and not very large, it can be easily extracted using an exhaustive search.

It is clear that $A$ is a skew-symmetric matrix: $-A = A^T$. PCL’s main goal is to generate a new skew-symmetric matrix $A^{(k)}$ in a deterministic manner for each round $k$ so that users do not have to republish information often. Furthermore, $A^{(k)}$ has coefficients in $Z_p$ instead of $\{1, 0, 1\}$; depending on his or her collusion tolerance, the user may select how many zeros each row will have. The more zeros in row $i$, the fewer computations user $U_i$ has to make and the bigger tolerance he or she has. Here we assume a function

$$x: Z_p \times Z_p \rightarrow Z_p^{\times n}$$

that takes a random seed and a round number, and then outputs a pseudorandom skew-symmetric matrix over $Z_p$. Note that the seed can be predetermined or derived from users’ public keys.

The main steps of PCL are as follows:

1. Party $P_i$ generates a secret key $x_i \in Z_p$ and computes his or her public key $U_i = g^{x_i} \in G$. He or she broadcasts $U_i$.
2. For every round $r \in \{1, \ldots, \ell\}$:
   - Party $P_i$ chooses his or her input $m_i \in \{0, \ldots, \beta\}$.
   - Compute $A^{(r)} = x_i S_i$.
   - Compute $w = \prod_{j=1}^n U_j^{m_j} \in G$.
   - Compute $v^{(r)} = w^{x_i} \cdot g^{m_i} \in G$.
   - Broadcast $v^{(r)}$.
3. The protocol produces an output of $\ell$ elements, namely the sum of the inputs in each round. To compute the sum $\sigma_i$ for round $r$:
   - Compute $z = \prod_{j=1}^n v^{(r)}$.
   - Use Pollard’s lambda algorithm to compute the discrete log $\sigma_i \in \{0, \ldots, n\}$ of $z$ with respect to $g$ in $G$. The time complexity of Pollard’s lambda algorithm is $\sqrt{n\beta}$.
   - The final output is $(\sigma_1, \ldots, \sigma_\ell)$.

It can be easily observed that for any $1 \leq r \leq \ell$,

$$\prod_{j=1}^n v_j^{r} = g^{\sum_{r=1}^\ell m_j^{(r)}}.$$

PCL can be extended to support blind aggregation. This means that users submit their data in a more obfuscated form so that the aggregator computes an obfuscated form of the summary, which can only be recovered by another entity. This provides a differentiation between the entities. Users submit their data to the aggregator, which forwards the obfuscated result to the data consumer. The information cannot be intercepted throughout the procedure. Moreover, due to the nature of the protocol, the aggregator need not be fully trusted, which significantly decreases data exposure. An overview of the protocol is illustrated in Figure 2.

**FIGURE 2.** The proposed PCL protocol. In the initialization phase, users publish their public shares, which are used in each of the $r$ rounds to produce obfuscated values. These values are then multiplied by the aggregator to calculate the sum of the values.

SCALABILITY AND PERFORMANCE

As indicated earlier, our proposed protocol offers a special feature where-in the result can be obfuscated from the aggregator, meaning many local aggregators could be installed to perform aggregation on a local scale. Depending on the type of data to be aggregated, a local aggregator can
host up to hundreds or thousands of users without significant processing requirements. Moreover, due to the nature of the protocol, the local aggregator cannot find the values of individuals or access the resulting aggregated value. The local aggregators thus act as “anonymous proxies” between users and the system, resubmitting the aggregated information without disclosing users’ identities. This means that PCL can be used to provide scalability, as Figure 3 shows.

Several of the protocol’s features help boost its performance, aside from user-customizable privacy tolerance. Elliptic curves allow the scheme to use small groups and publish minimal information each time it is needed. Moreover, the use of elliptic curves has been widely accepted as a cryptographic standard for devices with low processing resources, as the computations, such as point addition and scalar multiplication, can be performed very efficiently without much effort. The lack of pairings, used widely in the KDK protocol, provides an additional performance boost, as these are very slow when used on devices such as sensors. In addition, PCL enables the reuse of published information for several rounds, diminishing bandwidth cost.

To better understand the protocol’s efficiency, note that time complexity is linear to the number of users. The reason is that at any step we have to make $n$ multiplications and one exponentiation. Running one round of the protocol for 1,000 users with full privacy on a common laptop takes around 45 milliseconds, while a non-optimized version on a mobile phone (without elliptic curves) takes around 330 ms.

We developed an Android application to further test PCL efficiency. The tests were performed on a Samsung GT-I9001 with Android 2.3.3. In these tests, a user submits a value sampled in the range of [0–1000]. The experimental results, summarized in Table 1, clearly indicate that computational cost is minimal, enabling submission of data in real time depending on application needs. Moreover, our implementation used the less efficient approach of the multiplicative group mod $q$ without multithreading, which means that these measurements can be drastically improved.

**INTEROPERABILITY**

Interoperability can be understood in many ways. The European Parliament defined it in 2009 as “the ability of disparate and diverse organizations to interact towards mutually beneficial and agreed common goals, involving the sharing of information and knowledge between the organizations, through the business processes they support, by means of the exchange of data between their respective ICT systems.”

E-government protocol interoperability allows manufacturers to easily deploy their solutions and thus encourages e-participation. More citizens will embrace a solution that is platform- and device-independent. Our protocol can readily be implemented in many platforms. But it can also be integrated in more efficient and interoperable ways, especially in the case of devices with low processing capabilities, such as sensors.

Currently, sensors are widely deployed in urban areas to facilitate the needs of smart cities and e-governments. Numerous manufacturers and devices support many protocols, creating a heterogeneous mixture that makes sensor discovery and access a nontrivial task. Many proposed interoperability solutions follow a bottom-up approach, such as Device Description Language (DDL). One widely adopted approach is SensorML (www.opengeospatial.org/standards/sensorml), which was created in 1998 and quickly standardized by the Open Geospatial Consortium (OGC). Initially, SensorML was used to describe the geometric, dynamic, and radiometric properties of dynamic remote sensors. OGC generalized it even more, targeting the development of standard sensor models and standardized descriptions of sensors and their data. Currently, SensorML is one of the most commonly used standards for sensor description. The concept behind SensorML was to generate an XML scheme that encodes metadata for describing sensors, sensor platforms, sensor tasking interfaces, and sensor-derived data. XML was adopted for publishing sensor descriptions to support interoperability, as the vast majority of programming languages can support XML with their parsers. In this sense, SensorML arguably provides an XML wrapper for publishing...
formal descriptions of sensors’ capabilities, locations, and interfaces.

We propose the use of SensorML, which exposes a RESTful interface. Each sensor exposes two additional methods, GetPublicShare and SendPublicShare. The aggregator discovers and registers the devices using SensorML, and, whenever data are required, it calls the SendPublicShare method to each sensor. Upon receiving this call, the sensor will generate its public share for the first round and reply with that value. Whenever the aggregator receives all the public shares, it creates a vector of all the values and sends them, as parameters, to the GetPublicShare method. Each sensor can then calculate the new share, embed its measurement, and return the result to the aggregator. While SensorML simplifies the procedure of describing and managing sensors, the RESTful interface provides ease of functionality. Given sensors’ low processing capabilities, the combination of SensorML and the RESTful interface is superior to other options like SOAP because the computational and communication overhead is significantly lower.

With the wide adoption of ICT, many aspects of our daily lives are changing. E-government is a clear example of the way ICT can improve efficiency, interoperability, privacy, and participation. We support the idea that smart cities can contribute to the better deployment of e-government strategies at the municipal level. In this context, we propose the use of a cryptographic protocol and infrastructure that allow the private sharing of information to promote e-participation, which allows municipalities to make better decisions based on fresh data privately contributed by citizens.

Our solution satisfies the four challenges that we identified: it is efficient, it is interoperable thanks to the use of open standards, it protects citizens’ privacy, and it is scalable. Further research will focus on applying our scheme to specific scenarios like smart health, urban sensing, and pollution monitoring.

ACKNOWLEDGMENTS
Constantinos Patsakis and Mélanie Bouroche are supported by Science Foundation Ireland under the Principal Investigator research program 10/IN.1/I2980, “Self-Organizing Architectures for Autonomous Management of Smart Cities.” Agustí Solanas is partly funded by La Caixa Foundation through the RecerCaixa 2012 project “SIMPATIC: Intelligent, Autonomous and Private Monitoring System based on ICT,” and by the Government of Catalonia under grant no. 2014 SGR 537. Michael Clear’s work is funded by the Irish Research Council’s Embark Initiative.

REFERENCES

<p>| TABLE 1. Average time per round and per user for different lengths of primes and sets of users. |
|-----------------|-----------------|-----------------|</p>
<table>
<thead>
<tr>
<th>Number of users</th>
<th>512 bits (length of prime)</th>
<th>1,024 bits (length of prime)</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>168 ms</td>
<td>291 ms</td>
</tr>
<tr>
<td>500</td>
<td>174 ms</td>
<td>339 ms</td>
</tr>
<tr>
<td>1,000</td>
<td>324 ms</td>
<td>719 ms</td>
</tr>
<tr>
<td>2,000</td>
<td>723 ms</td>
<td>932 ms</td>
</tr>
</tbody>
</table>
OUTLOOK

ABOUT THE AUTHORS

CONSTANTINOS PATSAKIS is a lecturer in the Department of Informatics at the University of Piraeus, Greece. His main areas of research include cryptography and security. Patsakis received a PhD in cryptography at the University of Piraeus. He is a member of IEEE, IEEE Computer Society, ACM, and IACR. Contact him at kpatsak@unipi.gr.

PAUL LAIRD is a researcher in the Distributed Systems Group of the School of Computer Science and Statistics at Trinity College Dublin. His research focuses on flexible languages for secure multiparty computation, secure multiparty communication protocols, context-sensitive software, and on-the-fly content-based document encryption. Contact him at lairdp@tcd.ie.

MICHAEL CLEAR is a PhD student in the School of Computer Science and Statistics at Trinity College Dublin. His research is in the field of cryptography, with a focus on homomorphic encryption with cryptographic access control. Contact him at clearm@tcd.ie.

MÉLANIE BOUROCHE is a lecturer and manager of the Future Cities Research program in the School of Computer Science and Statistics at Trinity College Dublin. Her research interests include middleware for future cities, wireless sensor networks, vehicular communication, coordination of mobile autonomous entities, and distributed systems. Bouroche received a PhD in computer science from Trinity College Dublin. Contact her at melanie.bouroche@gmail.com.

AGUSTI SOLANAS is an associate professor in the Department of Computer Engineering and Mathematics and head of the Smart Health Research Group at Rovira i Virgili University, Catalonia, Spain. His research interests are smart health, privacy, security, and clustering. Solanas received a PhD in telematics engineering from Technical University of Catalonia. He is a senior member of the IEEE, a member of the IEEE Computer Society, a member of the ACM, and category editor of ACM Computing Reviews. Contact him at agusti.solanas@urv.cat.


