Allowing Non-identifying Information Disclosure in Citizen Opinion Evaluation

Francesco Buccafurri, Lidia Fotia, and Gianluca Lax

DIIES, University Mediterranea of Reggio Calabria
Via Graziella, Località Feo di Vito
89122 Reggio Calabria, Italy
{bucca,lidia.fotia,lax}@unirc.it

Abstract. The continuous participation of citizens in the decisional processes of the community through the submission of their opinions is a key factor of e-democracy. To do this, it appears very promising the use of *lightweight* e-voting systems relying on existing social networks, as a good way to solve the trade-off among security, usability and scalability requirements. Among the other security features, anonymity of citizens (i.e., secreteness) should be guaranteed, at least to be sure that the action of people is actually free from conditioning. However, the decisional process would be better driven if the opinions of citizens were mapped to social, economic, working, personal, non-identifying attributes. In this paper, by extending a previous solution working on existing social networks, we overcome the above limit by re-interpreting the classical concept of secreteness in such a way that a preference expressed by a citizen can be related to a number of (certified) attributes chosen by the citizen herself, yet keeping her anonymity.

1 Introduction

The model of e-democracy is one of the most challenging innovations towards which any community which is a candidate to become a *smart city* should tend. Indeed, the continuous participation of citizens to the decisional processes of the community is actually one of the most important aspects to deal with, whenever the smart-city model is implemented. Recall that the concept of smart city has to be intended in an extended way, thus not necessarily limiting the scope of e-services and the dynamics of the involved processes just to a city, but to an entire community which could be sometime really a city, sometimes a region, sometimes an entire country. It is well known that e-democracy declines in many different forms, all sharing the presence of ICT-based processes allowing citizens to become actors of the government of the community [47,45]. Among these, all the processes aimed at collecting opinions, preferences, evaluations of citizens [11], assume a very important role in the e-democracy model, since represent the concrete way to adapt government decisions to the real expectations of citizens [40,50,48].

Consider for example the preliminary evaluation of a law or a reform, a political parties poll, a satisfaction survey, a primary election, just to mention a

A. Kő et al. (Eds.): EGOVIS/EDEM 2013, LNCS 8061, pp. 241–254, 2013.

few. In these cases, secretness (i.e., anonymity of citizens) should be guaranteed, at least to be sure that the action of people is actually free from conditioning. Moreover, all the remaining basic properties of e-voting systems [13,46], namely uniqueness, verifiability, uncloneability, robustness and scalability, are essential requirements. In [10], it is shown that a suitable use of cryptographic protocols and social networks can be a good way to implement this *light* form of public elections, supporting all the above features. But among such features, secreteness inhibits the possibility to relate the preferences expressed by citizens even to non-identifying attributes [12]. By contrast, this would be a feature very desirable in the considered setting, differently from the one of elections. Indeed, the decisional process would be better driven if the opinions of citizens were mapped to social, economic, working, personal, non-identifying attributes. In this paper, we overcome the above limit by re-interpreting the classical concept of secreteness in such a way that a preference expressed by a citizen can be related to a number of (certified) attributes chosen by the same citizen, yet keeping her anonymity. Besides the possibility to analyze citizens' preferences and to extract useful knowledge from them, it will be possible to enable filtering mechanisms aimed at collecting only preferences of a certain segment of the population, like all people with a certain age range, a certain job, a given region and so on. Observe that the above requirements evokes what is provided by selective disclosure and bit commitment approaches [8,44,37,52], but a direct application of such approaches to our case is not resolutive since the secret used by a citizen to enable the disclosure of the chosen attribute would allow third parties to trace the citizen herself, thus breaking anonymity. The problem is thus non trivial.

We propose a solution that extends the model presented in [10]. It is based on pre-existent social networks, allowing citizens to vote through their own profile and does not require complex overhead besides an electronic card to identify a citizen or any identity management system able to identify people (plausibly, we can consider this is for free in an e-government context), and the owing a profile by each voter in one of the existing social networks.

The paper is organized as follows. In the next section, we recall some background notions. An overview of our proposal is given in Section 3, where the differences of this proposal with the model presented in [10] are discussed. The protocol allowing the selective disclosure of some attribute in an e-voting session is defined in Section 4. In Section 5, we analyze the security of this protocol. Section 6 is devoted to the related literature. Finally, in Section 7, we draw our conclusions and sketch possible future work.

2 Background

In this section, we present the background necessary to the reader to understand the technical aspects of the paper. Such notions are *discrete logarithm problem*, *digital signature* and *partially blind signature*.

The difficulty of solving the discrete logarithm is exploited to guarantee the security of numerous cryptosystems [3]. The discrete logarithm problem can be

formalized as follows. Let G be a multiplicative group and let $\langle g \rangle$ be the cyclic subgroup generated by $g \in G$. Given $g \in G$ and $a \in \langle g \rangle$, the problem consists in finding an integer x such that $g^x = a$. Such an integer x is the discrete logarithm of a to the base g (i.e., $x = log_g a$). Note that $log_g a$ is only determined modulo the order of g.

The digital signature mechanism relies on public key infrastructure. Each user owns two keys, a private key and a public one. The private key is kept secret and the public one is made public. Guessing a private key is computationally unfeasible for enough large keys. The first step of the signature generation process is the computation of a cryptographic hash function [28,26] of the document to be signed. The result, called digest, can substitute the original document in the signature generation process since the probability of having two distinct documents producing the same digest is negligible. Moreover, the problem of finding a document with digest equal to that of another given document is unfeasible, so that an attacker cannot corrupt a signed document without the signature detects it. The digital signature is produced by encrypting the digest with the private key using an asymmetric cryptographic cipher, typically RSA. The verification of the signature is done by checking that the decryption of the signature done with the public key of the subscriber coincides with the (re-computed) digested of the document.

Partially blind signatures [1] are a particular type of signature allowing the signer to explicitly include in unblinded form some pre-agreed information in the blind signature, like an expiry date, and are mainly used in the context of electronic cash (e-cash).

3 An Overview of the Proposal

In this section, we briefly describe the scenario we have designed in our proposal. The e-voting protocol will be described in the next section. The scenario is close to the one presented in [10]. We assume that citizens may use a smart card embedding a certificate granted by any Certification Authority including only a unique numeric ID and a list of pairs $\langle x, y \rangle$ where x is the attribute name and y is the obscured attribute value. This certificate, not existing in [10], includes information about the citizen in an obscured form, in such a way that the user may decide which information can be disclosed. Attributes encode standard information about users like personal data, but also more general information like job, qualification, marital status, etc. As usual, the certificate is a semi-structured document where the attributes are optionally included. For each attribute, its value is obscured by applying a one-way function using a key. A different key for each attribute is used. The keys are shared between the user and the Certification Authority. Thus, for a given attribute value A and a given key k, we obscure the attribute value by computing $g(A, k)$, where g is a one-way function. This means that it is unfeasible to compute A from the knowledge of just $g(A, k)$. For function g, we adopt the modular power function. The infeasibility of the computation of the discrete logarithm ensures us that the function is one way.

As in [10], the solution is based on the usage of existing social networks. Citizens vote by using their social network profile. The e-voting infrastructure is implemented by exploiting, for the selected social networks several profiles whose `URL` is of the form: `http://www.socialnetwork.org/poll_Y`, where `Y` is a cardinal number. These profiles are managed by possibly different government entities. Each entity replicates its profile over the most common social networks. The only requirement we have for these *super* profiles is the service continuity. These profiles are called *credential providers* and play the role of granting credentials to voters they can spend in order to submit their vote to a Trusted Third Party (TTP), responsible of generating the ballots for each e-voting. The domain of credential providers is built by collecting a large variety of subjects, like public sector offices, postal offices, universities, schools, military subjects and so on.

Recall that, differently from the model presented in [10] where both credentials and votes submitted by citizens do not include any additional information, our goal here is to associate votes with some attribute values chosen by the voter. It is worth noting that the trivial extension of the protocol of [10] consisting in including in the credentials granted by the credential providers also the attributes the voter wants to disclose does not work. Indeed, in this case the credential providers would able to incrementally relate information about the voter, as they know her identity. This way, the protocol would violate the confidentiality of the attribute certificate. By contrast, we want to relate votes to voters' attributes without discovering to anyone whom these attribute refer to.

To do this, we include in the credentials a further obscuration of the attribute values of the certificate by means of a different key per attribute. Each credential provider shares these keys with the user, but, obviously, it does not know the attribute values because operates only on already obscured values taken from the attribute certificate.

The credentials obtained by the voter contain a double obscuration of all the attributes of the voter, each with two keys, namely k and r (different for each attribute), in such a way that the knowledge of the product $k \cdot r$ allows us to obtain the final obscuration starting from the plain value of the attributes. Indeed, for the chosen function g, it holds that $g(g(A, k), r) = g(A, k \cdot r)$.

This way, whenever the voter submits her vote to TTP, she decides which attributes to disclose, simply by including into the vote record the attribute value, say A, and the product of keys $k \cdot r$. Then, TTP for each chosen attribute A, just has to compute the value $g(A, k \cdot r)$ and to verify whether this value is included in the related credential.

The scenario is summarized in Fig. 1. To avoid that the protocol is breakable by just one misbehaving credential provider, we use the common approach of replicating the responsibilities over a number of different independent parties [25,52,30,35]. In fact, the voter selects a suitable number $\overline{t}$ of credential providers on the basis of the value of her ID and asks them for the credentials necessary for the e-voting session. In Fig. 1, we describe the different steps related to an e-voting session. First, the user receives the obscured certificate from a

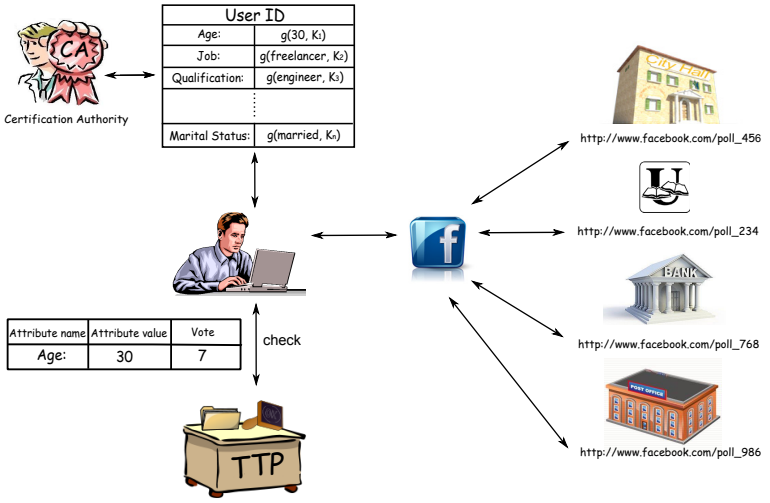

Fig. 1. The e-voting scenario

certification authority. Then, on the basis of her ID, the voter (we assume she has joined Facebook) computes four values (i.e., $Y_1 = 456$, $Y_2 = 234$, $Y_3 = 768$, and $Y_4 = 986$) identifying the respective credential users (in this example, $\overline{t} = 4$). The Trusted Third Party collects votes, verifies that they are admissible, and generates the ballots for each e-voting. The protocol ensures that the credential providers, even though may identify voters cannot link them to their vote, while TTP cannot identify voters but can only be aware about the attributes voluntarily disclosed by the voter.

As already done [10], the only assumption is that no more than t credential providers collude, where t is a parameter of the system. The number $\overline{t}$ of contacted credential providers per voter is directly related to t. The detail of the protocol is shown in the next section.

4 The E-Voting Protocol

In this section, we describe how the e-voting protocol works. Consider an e-voting session identified by ID_{vs}. For the sake of presentation, we assume that a preference is expressed by reporting the number i identifying the choice of the voter. For example, if the voting session regards the choice of one among 8 candidates in a primary election, then the choice of the voter could be represented by a number from 1 to 8. However, extending our technique to the cases in which preferences are given in a difference way (for example, in the case of a primary election, by indicating the name of the candidate) is possible with no impact on the model.

The e-voting process involves the following four basic entities:

1. The *Voter* V. We describe how the protocol run for the voting done by one user. Clearly, the overall e-voting session involves many user, each running these steps.
2. A *Certification Authority* CA granting attribute certificates to voters.
3. A set $\langle CP_1, \ldots, CP_c \rangle$ of c special users, named *credential providers*, issuing the credential exploited by the voter to prove her authorization to vote.
4. A *Trusted Third Party*, say TTP, responsible of generating the *certified ballots* for each e-voting.

Our technique is parametric with respect to a value t. It is chosen in such a way that the likelihood that t randomly selected users misbehave is negligible. This is a common assumption in this context [52,30,35,25].

Now, we describe how the e-voting process proceeds. It consists of the following steps:

1. *Certificate Issue.* In this first step, CA generates the attribute certificate for the voter V which contains ID_V (i.e., a value that uniquely identifies each voter) and a list of n associated attributes. All the attributes but ID_V are obscured, in such a way that a third party cannot know the values of such attributes by accessing the certificate. In particular, for each attribute, its value is obscured by applying a one-way function using a key. A different key for each attribute is used. The keys are shared between the voter and CA. In detail, for a given attribute value A and a given key k, we obscure the attribute value by computing $g(A, k)$, where g is a one-way function. This means that it is unfeasible to compute A from the knowledge of just $g(A, k)$. For function g, we adopt the modular power function $A^k \bmod m$, where m is a prime number greater than any possible A. In practise, m can be set by assuming a realistic upper bound for the values of attributes. If the above assumption is not applicable, we can use for each attribute a different module, which depends on the actual value of the attribute. In this case, the value of the module has to be saved in the certificate.

   Thus, CA selects a random vector of keys $(k_1, \ldots, k_n)$. Each attribute included in the certificate is a pair $(AN, g(AV, k_i))$, where AN is the attribute name and AV is the attribute value. Therefore, in the certificate, instead of the plain value AV, only the obscured value $g(AV, k_i) = AV_i^{k_i} \bmod m$ is inserted. At the end of this operation, CA signs the certificate and sends it to V together with the vectors $(k_1, \ldots, k_n)$ and $(AV_1, \ldots, AV_n)$. We denote by C the so obtained certificate.
2. *CPs Identification.* In the first step, V has to select $\overline{t} = 2 \cdot t + 1$ of the c credential providers that will generate the credentials. The p-th credential provider chosen by V, say CP_p^V, with $1 \leq i \leq \overline{t}$, is CP_j, with $j =$SHA-$1(ID_V||i) \bmod c$. Specifically, the first credential provider is obtained by applying the hash function SHA-1 to the concatenation between the voter identifier ID_V and the number 1 (i.e., $i = 1$), and then by mapping the result to one of the c credential providers through the `mod` operation. Note that the value j computed by SHA-1 corresponds to the number `Y` completing the

URL identifying the credential provider (recall the discussion done concerning the scenario described in Fig. 1).

3. *Credential Issue.* In this step, the voter starts a connection with each CP_p^V (among the $\overline{t}$ ones). CP_p^V verifies that it has been correctly contacted by recomputing the function SHA-1 as done by V at the previous step. If this is the case, then CP_p^V generates the credential C_p^V allowing V to participate to the e-voting session. Otherwise, the connection is terminated. Before the generation of the credential, V sends the certificate C issued in Step 1 to CP_p^V, together with a random vector $\langle r_1, \ldots, r_n \rangle$, where, we recall, n is the number of the attributes in C. Then, CP_p^V generates a n-tuple of pairs $AT = \langle (AN_1, g(AV_1, k_1 \cdot r_1)), \ldots, (AN_n, g(AV_n, k_n \cdot r_n)) \rangle$. Observe that the second element of the i-th pair is the further obscuration of the i-th attribute value by means of the random value r_i, i.e., $g(g(AV_i, k_i), r_i) = AV_i^{k_i \cdot r_i}$. We denote the attribute name AN_i by $AT(i).name$ and the attribute value AV_i by $AT(i).value$.

   At this point CP_p^V is ready to construct the credential C_p^V. It consists in the signature of the pair $\langle ID_{vs}, AT \rangle$, where ID_{vs} is the identifier of the voting session.

4. *Voting.* After the voter has collected the credential from each of the $\overline{t}$ credential providers, these credentials are presented to TTP in order to obtain the possibility to vote.

   In particular, TTP performs the following tests on the received credentials:

   (a) It checks authenticity and integrity of each credential and that the voting reference (i.e., ID_{vs}) in each credential coincide.
   (b) It verifies that in the past, no user has presented credentials issued from the same credential providers as the current voter for the same voting session (otherwise, it means that the voter is trying to repeat her participation to the same voting).

   If both the tests succeed, then the voter is authorized to vote possibly disclosing some attributes.

   Suppose now that V decides to disclose h attributes, with $h \leq n$. In this case, she must send to TTP the h-tuple of pairs $T = \langle (B_1, e_1), \ldots, (B_h, e_h) \rangle$, where B_i is the value of a chosen attribute, say it the attribute A_x, and e_i is the i-th product $k_x \cdot r_x$, for $1 \leq i \leq h$. To verify that the voter choice is valid, it is necessary that TTP checks the consistence of T with AT. In particular, given the function $f : \{1, \cdots, h\} \rightarrow \{1, \cdots, n\}$, such that:

$$f(i) = \begin{cases} j & \text{if } \exists\ j \in [1, n] \mid AT(j).value = B_i^{e_i}, \\ undefined & \text{otherwise.} \end{cases}$$

   TTP has to verify that f is total, i.e., is defined over all the domain $\{1, \cdots, h\}$. If this check fails, the vote is invalidated. Otherwise, TTP generates the ballot. The ballot consists in the partially blind signature of the quadruple $\langle ID_{vs}, \widetilde{r}, \widetilde{pr}, (AT(f(1)).name, B_{f(1)}), \cdots, (AT(f(h)).name, B_{f(h)}) \rangle$, where $\widetilde{r}$ is a fresh 128-bit random sequence and $\widetilde{pr}$ represents the preference specified by the voter.

The values ID_{vs} and $(AT(f(1)).name, B_{f(1)}), \cdots, (AT(f(h)).name, B_{f(h)})$ are unblindly signed, whereas $\widetilde{r}$ and $\widetilde{pr}$ are blindly signed. Finally, TTP stores the received credentials in order to detect a possible re-submission of the same credentials.

5. *Ballot Publication.* After the voter obtains the signed ballot, she unblinds it in order to obtain a new ballot still correctly signed by TTP but not linkable anymore to the voter. As usual, timing attacks are prevented by introducing an unpredictable delay before sending the new ballot back to TTP. The final ballot is thus $\langle ID_{vs}, r, pr, (AT(f(1)).name, B_{f(1)}), \cdots, (AT(f(h)).name, B_{f(h)})\rangle$.

   Observe that, due to the presence of the tuples $(AT(f(1)).name, B_{f(1)}), \cdots, (AT(f(h)).name, B_{f(h)})$, the list of attribute names and values that V has chosen to disclose is shown in the ballot.

   At the end of the e-voting session, TTP verifies the signature of all received ballots and publishes valid ones. The presence of non identifying information about the voter enables the possibility to analyze citizens' preferences in order to extract useful knowledge from them.

5 Security Analysis

This section is devoted to the analysis of the robustness of our protocol against a large number of realist attack model. We extend the security analysis done in [10] taking into account the improvements introduced by our proposal. Also in this case, the basic assumption is that at most t users misbehave during the whole evaluation process.

We start by analyzing the possibility for a credential provider to be aware of information about the voter. Any selected credential provider, say CP, cannot guess the value of the attributes in the certificate. Indeed, let us assume that CP wants to know whether the real value of the obscured attribute $A' = A^k \bmod m$ is equal to F. Then, it has to find a value k' such that $F^{k'} \bmod m = A'$ which corresponds to find the discrete logarithm of A', which is unfeasible. With stronger reason, any other entity which is aware of the attribute certificate or credentials of the voter can guess the value of the non-disclosed attributes.

There is no link between the certificate and the credentials issued to the same voter. Indeed, the voter ID is not included in the credential and any attribute $g(AV_i, k_i)$ in the certificate is transformed to $g(AV_i, k_i \cdot r_i)$. Thanks to the further obscuration performed by r_i, there is no way, without the knowledge of this random value, to link the credential to the attribute certificate (and then to the voter). Clearly, ID_{vs} is the identifier of the voting session and is not included in the certificate. The only information known by TTP is the e-voting session and the disclosed attributes and cannot link the voter and the preference rate of her ballot thanks to the use of the partially blind signature (at Step 4). Observe that the collusion between TTP and a credential provider allows them to link the voter identifier to the disclosed attributes also in different voting sessions. However, they cannot know also the preference score which is indistinguishable

among all the votes of the e-voting session (the partially blind signature of TTP on the ballot hides the preference score).

Our protocol allows each user to express only one preference. In case the voter tries to use the same credentials for a second time, the double vote is detected by TTP in Step 4.(b). Again, if the attacker requires to the certification authority a new certificate, the user ID is the same, thus resulting in the failure of the attack. Moreover, observe that in principle it could occur that two different voters V_1 and V_2 in the same voting session are considered the same by TTP in the case that the two voters share the set of credential providers due to the collision of the hash function SHA-1. This would result in improperly rejecting the latest vote erroneously detected as duplicated vote. However, this event can be considered impossible since its probability is negligible in a realistic scenario. For example, since the number of possible different sequences of credential providers is $c\,!/(c-\overline{t})!$ (we recall c is the number of credential providers and $\overline{t} = 2 \cdot t + 1$), for the realistic values $\overline{t} = 21$, $c = 200$, and even hypothesizing an unrealistically high number of users 10^{12} voters, we obtain that the probability of collisions is less than 10^{-20}.

The vote verifiability continues to be guaranteed. Each user can find its vote identified by r on the published ballot list and verify its correctness. The probability that two voters generate the same 128-bit sequence r is $p(u; D) \approx 1 - e^{-u^2/(2 \cdot D)}$ (birthday attack) where u is the number of users expressing her preference for a candidate and D is the domain of r. Assuming again a number of users u equals to 10^{12} (in the worst case), such a probability is negligible (in numbers, this probability is less than 10^{-15}).

Also uncloneability holds. This property ensures that generating a bogus ballot starting from a legal one must be detected. We observe that a valid ballot has been signed by TTP and thus it cannot be modified. Obviously, it cannot be duplicate thanks to the presence of the bit-sequence r identifying the ballot, according to the previous probability consideration.

Concerning the possibility that two obscured values $g(AV_1, k_1 \cdot r_1)$ and $g(AV_2, k_2 \cdot r_2)$ in AT collide (recall TTP verifies that the function f is total at Step 4), the probability of this event is negligible thanks to the randomness of r_1 and r_2 assuming that the number of bits of such random values is sufficiently large. According to this observation, even though from a formal point of view the definition of the function f does not allow us to guarantee that f is deterministic, from a practical point of view f returns always a unique value when it is defined.

It is worth noting that the application of the hash function SHA-1 at Step 2 returns a pseudo-random value depending on the voter (through her identifier) which allows us to assume that the credential providers selected by the voter can be considered randomly chosen. Thanks to this assumption, we can reach another important result. The unfair behaviour of at most t credential providers (according to our initial assumption in Section 4 about the possible misbehaving of users) is detected. Indeed, among the $\overline{t} = 2 \cdot t + 1$ credentials provided by the voter, at least $t + 1$ of them must be correct. As a consequence, fake credentials are detected since they are in the minority.

6 Related Work

E-government is a topic widely investigated in the last years by researchers [51,24,27]. In this section, we briefly survey the literature related to the topics of e-voting, which our proposal is clearly related to, and focus on selective disclosure, which represents a key issue in our approach.

Let us start with e-voting. Guaranteeing anonymity of the voter is an important requirement. For this purpose, Chaum [17] introduced the notion of *mix-net*, which exploits a sequence of servers. Each server receives a batch of input messages and produces as output the batch in permuted (mixed) order. An observer should not be able to tell how the inputs correspond to the outputs. Mix-nets are used to ensure voter privacy by providing the ballots of the voters as input to them. In Chaum's original proposal, before a message is sent through the mix-net, it is encrypted with the public keys of the mixes it will traverse in reverse order. Each mix then decrypts a message before sending it on to the next mix. A modified version of the protocol was published later by Chaum [20]. Here, a new kind of receipt improves security by letting voters verify correctness of the election outcome, even though all election computers and records were to be compromised. The system preserves ballot secrecy, while improving access for voters, robustness, and adjudication, all at lower cost.

Sako et al. [49] propose another approach to e-voting based on *re-encryption mix-nets* [43] and on *proofs*, used by voters to prove the correctness of the votes they sent. Zwierko et al. [52] propose an agent-based scheme for secure electronic voting. The protocol, presented in [32], is designed for large scale elections.

Chaum pioneered privacy-enhancing cryptographic protocols that minimize the amount of personal data disclosed. Chaum et al. defined the principles of anonymous credentials [18,19,22], group signatures [23], and electronic cash [18]. In all these papers, some party issues a digital signatures where the signed message includes information about the user (i.e., attributes). Subsequently, more efficient implementation of these concepts were proposed concerning group signatures [2,6,39], e-cash [7,14,31], and anonymous credentials [8,15,16]. Moreover, a number of new concepts were introduced, like traceable signatures [38], anonymous auctions [42], and electronic voting based on blind-signatures [32]. Many of these schemes use as building blocks signed attributes and protocols that selectively reveal these attributes or prove properties about them. Their implementations typically encode attributes as a discrete logarithm or, more generally, as an element (exponent) of a representation of a group element, resulting in protocols where the number of transmitted group elements and the performed commutations are linear in the number of encoded attributes.

An interesting approach for maximizing privacy protection is to selectively disclose attributes within a credential, so that only the needed subset of properties is made available to the recipient of the credential. A system to partially disclose credentials relies on the use of the bit commitment technique, which enables users to commit a value without revealing it. Bit commitment has been used for zero-knowledge protocols [33], [9], identification schemes [29], and multiparty protocols [34,21], and it can implement Blum's coin flipping over the phone

[5]. The idea of selectively disclosing credential attributes is not new [8,44]. [37] focuses on selective disclosure of credentials during negotiations and provides a prototype implementation. The focus of Bertino et al. [4] is to deeply analyze the impact of protected attribute credentials on trust negotiations, and to devise new strategies allowing interoperability between users adopting various credential formats. Further, instead of using the bit-commitment technique, the authors adopt a multi-bit hash commitment technique for attribute encoding, as the length of attributes will likely be longer than one bit.

Naor [41] shows how a pseudorandom generator can provide a bit-commitment protocol and also analyzes the number of bits communicated when parties commit to many bits simultaneously. Let $m(n)$ be some function such that $m(n) > n$. $G : \{0,1\}^n \rightarrow \{0,1\}^{m(n)}$ is a pseudorandom generator. $G_l(s)$ is used to denote the first l bits of the pseudorandom sequence on seed $s \in \{0,1\}^n$. $B_i(s)$ is used to denote the i-th bit of the pseudorandom sequence on seed s. The user selects seed $s \in \{0,1\}^n$ and sends $G_m(s)$ and $B_{m+l}(s) \bigoplus b$. In the reveal stage, the user sends s and the verifier checks that $G_m(s)$ is equal to the previously received value and computes $b = B_{m+l}(s) \bigoplus (B_{m+l}(s) \bigoplus b)$.

The system of Holt et al. [36] uses bit commitments to create selective disclosure credentials with a limited amount of data the holder must reveal. A selective disclosure credential has several attributes. When the user shows the credential to a verifier, she can choose to reveal only some of them. Credential sets accomplish this with the help of bit commitment that allows the user to commit to a value without revealing it. The user's commitment is the output of a one-way function *oneway*() operating on the concatenation of her secret value s and a random string r. The user first sends it to the verifier. If she chooses not to reveal the value, the verifier can't determine what the value was. To reveal her secret, she sends s and r to the verifier who computes the one-way function and checks that the result equals the value sent previously by the user.

We observe that the approaches based on selective disclosure and bit commitment do not solve the problem investigated in our paper. Indeed, the secret used by a citizen to enable the disclosure of the chosen attribute would allow third parties to trace the citizen preferences in the different voting session, thus breaking unlinkability.

7 Conclusion

In this paper, we have proposed a lightweight e-voting system relying on the use of social networks and allowing the voter to graduate the privacy level of the vote. In particular, the citizen may decide, whenever she submits a vote, to reveal some non-identifying personal certified attribute to link to the vote. The e-voting system is oriented to all those processes aimed at collecting opinions, preferences, evaluations of citizens, which assume a very important role in the e-democracy model, since represent the concrete way to adapt government decisions to the real expectations of citizens.

The result we have obtained is a fair compromise between the secreteness of the vote and the necessity of government parties to conduct analyses on the

collected opinions, in order to relate them to various types of information describing the inquired population. The solution shows also good features of feasibility since it does not require complex ad-hoc infrastructures by exploiting pervasive and user-accepted media (i.e., social networks). The security analysis also shows that all the basic properties of an e-voting system are satisfied and that a correct utilization of our extended notion of secreteness does not invalidate the anonymity of the voters. As a future work we plan to investigate the implementation issues with the goal of implementing a system prototype useful to perform real-life experiences also in limited (specific) domains.

Acknowledgment. This work has been partially supported by the TENACE PRIN Project (n. 20103P34XC) funded by the Italian Ministry of Education, University and Research.

References

1. Abe, M., Fujisaki, E.: How to date blind signatures. In: Kim, K.-c., Matsumoto, T. (eds.) ASIACRYPT 1996. LNCS, vol. 1163, pp. 244–251. Springer, Heidelberg (1996)
2. Ateniese, G., Camenisch, J., Joye, M., Tsudik, G.: A practical and provably secure coalition-resistant group signature scheme. In: Bellare, M. (ed.) CRYPTO 2000. LNCS, vol. 1880, pp. 255–270. Springer, Heidelberg (2000)
3. Bach, E.: Discrete logarithms and factoring. Computer Science Division, University of California (1984)
4. Bertino, E., Ferrari, E., Squicciarini, A.C.: Privacy-preserving trust negotiations. In: Martin, D., Serjantov, A. (eds.) PET 2004. LNCS, vol. 3424, pp. 283–301. Springer, Heidelberg (2005)
5. Blum, M.: Coin flipping by telephone a protocol for solving impossible problems. ACM SIGACT News 15(1), 23–27 (1983)
6. Boneh, D., Boyen, X., Shacham, H.: Short group signatures. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 41–55. Springer, Heidelberg (2004)
7. Brands, S.A.: An efficient off-line electronic cash system based on the representation problem (1993)
8. Brands, S.A.: Rethinking public key infrastructures and digital certificates: building in privacy. MIT Press (2000)
9. Brassard, G., Chaum, D., Crépeau, C.: Minimum disclosure proofs of knowledge. Journal of Computer and System Sciences 37(2), 156–189 (1988)
10. Buccafurri, F., Fotia, L., Lax, G.: Allowing continuous evaluation of citizen opinions through social networks. In: Kő, A., Leitner, C., Leitold, H., Prosser, A. (eds.) EDEM 2012 and EGOVIS 2012. LNCS, vol. 7452, pp. 242–253. Springer, Heidelberg (2012)
11. Buccafurri, F., Fotia, L., Lax, G.: Privacy-preserving resource evaluation in social networks. In: Proceedings of the 2012 Tenth Annual International Conference on Privacy, Security and Trust, PST 2012, pp. 51–58. IEEE Computer Society (2012)
12. Buccafurri, F., Lax, G., Nocera, A., Ursino, D.: Discovering links among social networks. In: Flach, P.A., De Bie, T., Cristianini, N. (eds.) ECML PKDD 2012, Part II. LNCS, vol. 7524, pp. 467–482. Springer, Heidelberg (2012)
13. Burmester, M., Magkos, E.: Towards secure and practical e-elections in the new era. Secure Electronic Voting, 63–76 (2003)

14. Camenisch, J., Hohenberger, S., Lysyanskaya, A.: Compact E-cash. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 302–321. Springer, Heidelberg (2005)
15. Camenisch, J., Lysyanskaya, A.: An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol. 2045, pp. 93–118. Springer, Heidelberg (2001)
16. Camenisch, J., Lysyanskaya, A.: Signature schemes and anonymous credentials from bilinear maps. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 56–72. Springer, Heidelberg (2004)
17. Chaum, D.: Untraceable electronic mail, return addresses, and digital pseudonyms. Communications of the ACM 24(2), 84–90 (1981)
18. Chaum, D.: Blind signatures for untraceable payments. In: McCurley, K.S., Ziegler, C.D. (eds.) Advances in Cryptology 1981 - 1997. LNCS, vol. 1440, pp. 199–203. Springer, Heidelberg (1999)
19. Chaum, D.: Security without identification: Transaction systems to make big brother obsolete. Communications of the ACM 28(10), 1030–1044 (1985)
20. Chaum, D.: Elections with unconditionally-secret ballots and disruption equivalent to breaking RSA. In: Günther, C.G. (ed.) EUROCRYPT 1988. LNCS, vol. 330, pp. 177–182. Springer, Heidelberg (1988)
21. Chaum, D., Damgård, I.B., van de Graaf, J.: Multiparty computations ensuring privacy of each party's input and correctness of the result. In: Pomerance, C. (ed.) CRYPTO 1987. LNCS, vol. 293, pp. 87–119. Springer, Heidelberg (1988)
22. Chaum, D., Evertse, J.-H.: A secure and privacy-protecting protocol for transmitting personal information between organizations. In: Odlyzko, A.M. (ed.) CRYPTO 1986. LNCS, vol. 263, pp. 118–167. Springer, Heidelberg (1987)
23. Chaum, D., van Heyst, E.: Group signatures. In: Davies, D.W. (ed.) EUROCRYPT 1991. LNCS, vol. 547, pp. 257–265. Springer, Heidelberg (1991)
24. Cordella, A.: E-government: towards the e-bureaucratic form? Journal of Information Technology 22(3), 265–274 (2007)
25. Cramer, R., Gennaro, R., Schoenmakers, B.: A secure and optimally efficient multiauthority election scheme. European Transactions on Telecommunications 8(5), 481–490 (1997)
26. Dobbertin, H., Bosselaers, A., Preneel, B.: RIPEMD-160: A strengthened version of RIPEMD. In: Gollmann, D. (ed.) FSE 1996. LNCS, vol. 1039, pp. 71–82. Springer, Heidelberg (1996)
27. Dunleavy, P., Margetts, H., Bastow, S., Tinkler, J.: Digital era governance: IT corporations, the state, and e-government. OUP Catalogue (2006)
28. Eastlake, D., Jones, P.: US secure hash algorithm 1 (SHA1). Technical report, RFC 3174 (September 2001)
29. Fiat, A., Shamir, A.: How to prove yourself: practical solutions to identification and signature problems. In: Odlyzko, A.M. (ed.) CRYPTO 1986. LNCS, vol. 263, pp. 186–194. Springer, Heidelberg (1987)
30. Fouque, P.-A., Poupard, G., Stern, J.: Sharing decryption in the context of voting or lotteries. In: Frankel, Y. (ed.) FC 2000. LNCS, vol. 1962, pp. 90–104. Springer, Heidelberg (2001)
31. Frankel, Y., Tsiounis, Y., Yung, M.: Fair off-line e-cash made easy. In: Ohta, K., Pei, D. (eds.) ASIACRYPT 1998. LNCS, vol. 1514, pp. 257–270. Springer, Heidelberg (1998)
32. Fujioka, A., Okamoto, T., Ohta, K.: A practical secret voting scheme for large scale elections. In: Zheng, Y., Seberry, J. (eds.) AUSCRYPT 1992. LNCS, vol. 718, pp. 244–251. Springer, Heidelberg (1993)

33. Goldreich, O., Micali, S., Wigderson, A.: Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. Journal of the ACM (JACM) 38(3), 690–728 (1991)
34. Goldwasser, S., Micali, S., Wigderson, A.: How to play any mental game, or a completeness theorem for protocols with an honest majority. In: Proc. of the Nienteenth Annual ACM STOC, vol. 87, pp. 218–229 (1987)
35. Hirt, M., Sako, K.: Efficient receipt-free voting based on homomorphic encryption. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 539–556. Springer, Heidelberg (2000)
36. Holt, J.E., Seamons, K.E.: Selective disclosure credential sets (2002), Accessible as `http://citeseer.nj.nec.com/541329.html`
37. Jarvis, R.: Selective disclosure of credential content during trust negotiation. Master of Science Thesis, Brigham Young University, Provo, Utah (2003)
38. Kiayias, A., Tsiounis, Y., Yung, M.: Traceable signatures. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 571–589. Springer, Heidelberg (2004)
39. Kiayias, A., Yung, M.: Secure scalable group signature with dynamic joins and separable authorities. International Journal of Security and Networks 1(1), 24–45 (2006)
40. Medaglia, R.: eParticipation research: Moving characterization forward (2006–2011). Government Information Quarterly (2012)
41. Naor, M.: Bit commitment using pseudorandomness. Journal of Cryptology 4(2), 151–158 (1991)
42. Naor, M., Pinkas, B., Sumner, R.: Privacy preserving auctions and mechanism design. In: Proceedings of the 1st ACM Conference on Electronic Commerce, pp. 129–139. ACM (1999)
43. Park, C., Itoh, K., Kurosawa, K.: Efficient anonymous channel and all/nothing election scheme. In: Helleseth, T. (ed.) EUROCRYPT 1993. LNCS, vol. 765, pp. 248–259. Springer, Heidelberg (1994)
44. Persiano, P., Visconti, I.: User privacy issues regarding certificates and the TLS protocol: the design and implementation of the SPSL protocol. In: Proceedings of the 7th ACM Conference on Computer and Communications Security, pp. 53–62. ACM (2000)
45. Persson, A., Goldkuhl, G.: Government value paradigms-bureaucracy, new public management, and e-government. Communications of the Association for Information Systems 27(1), 4 (2010)
46. Pieprzyk, J., Hardjono, T., Seberry, J.: Fundamentals of computer security. Springer (2003)
47. Rose, J., Sæbø, Ø.: Establishing political deliberation systems: Key problems (2008)
48. Sæbø, Ø., Rose, J., Skiftenes Flak, L.: The shape of eParticipation: Characterizing an emerging research area. Government Information Quarterly 25(3), 400–428 (2008)
49. Sako, K., Kilian, J.: Receipt-free mix-type voting scheme. In: Guillou, L.C., Quisquater, J.-J. (eds.) EUROCRYPT 1995. LNCS, vol. 921, pp. 393–403. Springer, Heidelberg (1995)
50. Susha, I., Grönlund, Å.: eParticipation research: Systematizing the field. Government Information Quarterly (2012)
51. Viscusi, G., Mecella, M.: Information systems for eGovernment: A quality-of-service perspective. Springer (2011)
52. Zwierko, A., Kotulski, Z.: A light-weight e-voting system with distributed trust. Electronic Notes in Theoretical Computer Science 168, 109–126 (2007)