Ajuntament de Barcelona

User Anonymization for Decidim Barcelona

Direcció de Recerca, Desenvolupament i Innovació, Regidoria de Participació

v. 1.2

Publication date: 05/31/17

Delivery date: 12/22/16

Summary

Metadata

Title	User Anonymization for Decidim Barcelona
Version	1.2
Date	Publication date: 5/31/17 Delivery date: 12/22/16
Editor	David Ruescas (nVotes)
Authors	David Ruescas, Eduardo Robles (nVotes)
Summary	Information technology and the Internet have opened up many possibilities for citizen participation. One of the most sensitive issues that must be addressed is privacy. Decidim, driven by the city council of Barcelona, is one such example of experiments in citizen participation. Because Decidim is an instrument with the potential for political decision making, it is important to attain the right privacy, trust and transparency properties. This document aims to propose technical solutions that offer the city council reasonable choices in these areas. The nature of Decidim as a tool with an already existing feature set narrows down the spectrum of technical methods that can be applied. Specifically, these constraints lead to solutions centered around the use of pseudonyms and the anonymization of existing users. These solutions must be made compatible with integrity requirements (leading to authentication). In order to achieve these two requirements simultaneously it is necessary to employ cryptographic techniques. The result are schemes offering strong privacy guarantees, where even the operators of the system cannot access user's real identities, while ensuring that only authenticated users can participate.
Keywords	Privacy, Anonymity, Verifiability, Citizen Participation, Cryptography
How to cite	Ruescas, D. &, Robles, E. (2017) User Anonymization for Decidim Barcelona. Barcelona, Spain.

Summary and recommendations

This section reviews the main conclusions and then presents some recommendations.

Summary

- Decidim is a citizen participation tool that functions as a social filtering system with proposal, debate and upvoting mechanics. It can be classified as a deliberation tool with voting elements, and additionally as consultation and ideation co-production system (Linders, 2012) with reputation elements (E-Participation – Wikipedia, 2016).
- As a tool with the potential for political decision making, issues of privacy, trust and transparency must be addressed carefully.
- In contrast to other forms of participation, such as voting, there is no clear consensus as to what the best practices for privacy are in the case of Decidim, neither in terms of requirements, nor in terms of methods. The literature is ambivalent about what requirements are appropriate citing arguments for and against anonymity. Moreover, it is unlikely that any existing research will be specific to Decidim's intent and features.
- A flexible approach which supports different anonymity levels is a reasonable response to the uncertainty and lack of consensus on this matter. The system will support three types of users, one of which must be anonymous and is the subject of technical solutions in this document.
- Decidim's characteristics necessarily imply pseudonym based anonymity solutions if one is to avoid adversely impacting functionality or usability. Solutions must implement user anonymization as a way to generate anonymous pseudonyms. Additionally, anonymous channels (such as TOR) are required to maintain privacy in the face of communication and traffic analysis.
- Anonymization must meet the following criteria (captured in objectives O1-O5)
  1) Anonymization solutions must grant users reasonable levels of privacy while maintaining authentication guarantees.
  2) Anonymization must be as transparent as possible to the Decidim software, and must not adversely affect its features and behaviours.
  3) Anonymization must be verifiable to the public, in contrast to approaches based on destroying existing correlations.
- Due to the nature of Decidim, anonymization cannot prevent inference attacks based on data mining user's contributions.
- Satisfying anonymity and authentication requirements simultaneously requires the use of cryptographic techniques, many of which originate in the literature on secure electronic voting. These techniques can be used as building blocks to compose protocols that implement anonymization.
- Several use cases involving anonymization are identified.
- Several protocols are defined (UML sequence diagrams and step-by-step text descriptions) which act as use case realizations. These protocols employ different building blocks and have different characteristics. Protocols compose into schemes.
- Given some supporting assumptions (cryptographic and otherwise), all the schemes satisfy the stated objectives O1-O5 reasonably well. This is stated without formal proof. Assumptions

are listed in relation to objectives.

- Besides complying with the stated objectives, additional evaluation criteria are defined which characterize the protocols and allow distinctions between them.
- The protocols are evaluated according to these additional criteria.

Recommendations

- A flexible approach towards privacy which allows users to autonomously disclose their identity is recommended. This can be accomplished with different user types, where anonymous users are implemented with one of the proposed solutions.
  - The possibility of anonymity, on its own, may have positive consequences in terms of trust, as citizens may recognize this feature as indicative of the city council's attitude towards privacy.
  - The existence of anonymity may, despite the uncertainty in the literature, act as a counterbalance to possible concerns about Decidim operating as a partisan tool.
  - Empirical evidence may be collected as to the effects of anonymity that are specific to Decidim from its use. Experiments can be conducted (further research is also possible).
- The choice of scheme depends on the desired properties listed as evaluation criteria.
  - Reversible anonymity is more practical for situations like password recovery, but may be regarded as a compromise to privacy. This compromise can be made small enough to err in favour of practicality with the participation of trustees.
  - If variable privileges are required, or staged re-anonymization is required, one must use one of the P or M schemes.
  - The M schemes are the most flexible and are generally recommended.
- We recommend against the use of a system based solely on user anonymization as a voting tool for binding elections or referenda. A tool with user anonymization is far from meeting the strong security requirements required for electronic voting. Please consult the literature on end-to-end verifiable voting systems (Chaum, D., 2004; Jonker, Mauw & Pang, 2013).

1 Introduction

Information technology and the Internet have opened up many possibilities for citizen participation. These possibilities are still being explored gathering experience as to what works and what doesn't. One of the most sensitive issues that must be addressed is privacy. In traditional citizen participation, such as voting, there is consensus as to what requirements and best practices must be followed with respect to privacy. Unfortunately this knowledge is not yet available for the new aforementioned modes of participation, neither objectives nor methods.

Decidim, driven by the city council of Barcelona, is one such example of experiments in citizen participation. This internet platform has elements of deliberation, filtering and voting. Because Decidim is an instrument with the potential for political decision making, it is important to attain the right privacy, trust and transparency properties. This document aims to propose technical solutions that offer the city council reasonable choices in these areas. The nature of Decidim as a tool with an already existing feature set narrows down the spectrum of technical methods that can be applied. Specifically, these constraints lead to solutions centered around the use of pseudonyms and the anonymization of existing users. These solutions must be made compatible with integrity requirements (leading to authentication). In order to achieve these two requirements simultaneously it is necessary to employ cryptographic techniques.

Section 2 contains an overview of Decidim and related initiatives that have been launched recently in several city councils in Spain. Section 3 presents the arguments that motivate the search for theoretical and technical solutions to possible privacy concerns, listing several desirable objectives. Section 4 explores a baseline solution used to compare later proposals and illustrates how naive approaches can fail to provide the transparency necessary for the task of guaranteeing anonymity in a political setting. Section 5 defines a simple, abstract scheme which serves as the core with which to build later specific proposals. Section 6 lists the main cryptographic techniques used to instantiate concrete schemes and protocols, which are defined in section 7. Section 8 evaluates these proposed solutions in terms of previously defined objectives and other criteria relevant to a practical implementation. Section 9 concludes with a summary and recommendations.

2 Context

Several city councils in Spain have undertaken pilots and initiatives (Decidim.barcelona, 2016; Home – Oviedoparticipa.es, 2016; Home – Oviedoparticipa.es, 2016; A Porta Aberta, O Portal de Procesos Participativos Do Concello Da Coruña 2016; Decide Madrid, 2016) in the area of citizen engagement and e-participation. Among these projects is Decidim Barcelona, an e-participation platform based on the Consul (Consul, 2016) software system first developed at the Madrid city council. Consul and therefore Decidim are systems aimed at citizen participation through deliberation and consultation, inspired by social information filtering (Lerman, 2007) tools such as Reddit (Reddit: The Front Page of the Internet, 2016) and Digg.

The core mechanics of Decidim and related instantiations are the submission of proposals, t, 2016he deliberation over said proposals and related debates, and the prioritization or filtering of said proposals for further analysis or formal approval. The choice of a social information filtering design aims to overcome the cognitive limits that arise in an open style of participation (Ruescas, D., 2016):

> In more open scenarios of citizen participation, people are asked not only to vote for predefined options, but to contribute their own ideas before the final vote. This introduces another activity to the democratic process, we can call it filtering.
>
> The aim of filtering is to select, out of a very large number of ideas, the very few best ones either to directly implement them or to put them to a formal vote. Filtering has to address the problem that voting cannot: How do we select out of potentially hundreds of ideas without having each voter evaluate them all? The solution provided by filtering is a common theme in many proposals to augment democracy: division of cognitive labour.

A single voter cannot rank hundreds of ideas, but the cognitive load of selecting the best ones can be spread across the many citizens that are contributing them. Filtering can scale because as the number of ideas grows with the number of people suggesting them, so too does the available cognitive effort available to evaluate them.

Although the social information filtering approach enjoys the benefits of cognitive distribution of labour and collective intelligence as originally applied to news aggregation, the overlap is not complete, adjustments must be made for the specific cases of e-participation that Decidim and related projects aim to address.

3 Motivation

Any initiative pertaining to citizen participation, e-governance or e-democracy must be handled with care to respect the principles of democracy. Matters of transparency, trust, and privacy[1] (O'Hara, 2012) emerge and must be considered as core values guiding projects in the aforementioned areas. This is especially true in light of recent developments which point to state surveillance as a practice with important consequences for privacy and citizen freedom (Hopkins, 2013).

Insofar as platforms such as Decidim are as much about citizen engagement as about fostering transparency regarding the city council's decision making it is relevant to remark (O'Hara, 2016):

> Privacy is extremely important to transparency. The political legitimacy of a transparency programme will depend crucially on its ability to retain public confidence. Privacy protection should therefore be embedded in any transparency programme, rather than bolted on as an afterthought. ...
>
> Transparency requires public confidence, and one way to ensure that is to reassure the public that its privacy is a central concern whose protection is embedded in decision-making processes.

Brazil's president Dilma Rouseff echoed these ideas more generally in an address to the UN general assembly:

> In the absence of the right to privacy, there can be no true freedom of expression and opinion, and therefore no effective democracy.

It seems clear from the above that privacy concerns should be carefully factored into the design of e-participation projects and tools. What is not as clear is what form these concerns take (if any) for the specific case of Decidim, and therefore what technical measures must be put in place to address them.

As a comparison, consider the practices applied to a different form of citizen participation central to democratic societies: voting. Privacy concerns in this case result in a clear recommendation, that of employing a secret ballot. The importance and widespread adoption of this practice is such that it appears in the Universal declaration of human rights (UN, 2014).

21.3 The will of the people shall be the basis of the authority of government; this will shall be expressed in periodic and genuine elections which shall be by universal and equal suffrage and shall be held by secret vote or by equivalent free voting procedures.

If Decidim were an electronic voting tool, privacy concerns would have a clear response, the tool would have to support a secret ballot according to the extensive literature on secure electronic voting. In this context a system supporting privacy is defined as (Sampigethaya & Poovendran, 2006).

In a secret ballot, a vote must not identify a voter and any traceability between the voter and its vote must be removed.

But Decidim is not an electronic voting system, but rather a social information filtering tool. Additional classifications are possible as a Consultation and Ideation co-production system (Linders,

1 Openness and Transparency - Pillars for Democracy, Trust and Progress - OCDE 2016.

2012) or a Reputation mechanism[2]. As such, privacy concerns must be addressed in way that is specific to Decidim's intent and features.

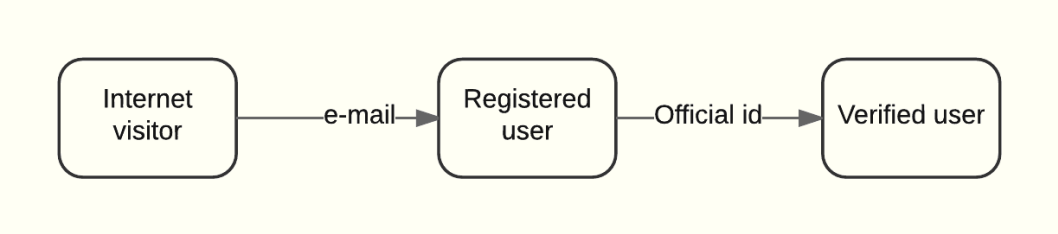
3.1 Decidim: Users and Citizens

As of this writing, the core interactions of Decidim are:

1) Submission of proposals and debate items
2) Commenting on proposals and debate items
3) Upvoting of proposals and comments

The mechanics described in the Context section are built out of these interactions. Because these operations mediate political decision making they require authentication and authorization to preserve the integrity of the process. In order to participate in Decidim, citizens must first register to obtain a user. The user as first registered is authorized to operate in the three ways listed above, except for the case of upvoting proposals. This is because this last capability, upvoting proposals, wields the highest amount of political power. To maintain integrity for results such as proposal prioritization or selection, users must have verified their real-world identity via an official document before they can upvote proposals. This instantiates a central theme found in electronic voting, the tension between integrity and privacy. The greater the political consequences a certain action has, the greater integrity concerns, and therefore the greater need for authorization and authentication.

Authorization and authentication are needed to guarantee that the action is carried out by an eligible citizen, and that said citizen does not exercise this right more times than are allowed (for example, voting twice). But this greater need for authorization brings with it correspondingly greater privacy concerns. In this case, verified users require attaching a real world identity that the administrators of the system and therefore the city council have access to.



We can explicitly list privacy concerns case by case; for each, we state what information is leaked by the use of the platform, and what information pertaining to the citizen is available. Given the three operations above we have

2 E-Participation – Wikipedia, 2016.

	User information	Citizen information
Registered	Authored debates/proposals Authored comments Upvoted/downvoted comments	Citizen's email Inferred identity
Verified	Authored debate/proposals Authored comments Upvoted/downvoted comments **Upvoted proposals**	Citizen's official ID

Privacy concern severity depends on how well citizens can be identified from available data. In case of registered users the citizen identity is not immediately available as an email may not be sufficient to identify the person. However, a verified user is linked unambiguously to a citizen identity via the citizen's official ID.

Note also how we have included "Inferred identity" in the top right box. This corresponds to the possibility that identity can be extracted from user information (debates, comments, upvotes/downvotes) through data mining (Clifton & Marks, 2016) (also known as inference attacks) (Krumm, n.d.):

> Data mining enables us to discover information we do not expect to find in databases. This can be a security/privacy issue: If we make information available, are we perhaps giving out more than we bargained for?

and (Yang & Wu, 2006)

> Several researchers considered privacy protection in data mining as an important topic. That is, how to ensure the users' privacy while their data are being mined.

This possibility exists for both Registered and Verified users, but it has not been included in the bottom right box because an exact identity is already available. A distinction could be made as to who is able to obtain this identity, in which case its inclusion would not be redundant, but we leave this level of detail out here; the aim in this document is to provide suggestions to mitigate worst case scenario privacy breaches, in which case system administrators would not benefit from extracting identity information when it is already available.

3.2 Objectives

Given the short overview of Decidim's specifics as to intent and features, we return to the question posed earlier: what privacy concerns and therefore mitigating strategies apply here?

As a tool that features upvoting proposals, Decidim has some resemblance with an electronic voting tool. However, the strong security requirements associated with such a purpose are far above and beyond what Decidim can currently provide. It seems more appropriate to consider Decidim as a deliberation tool with some additional filtering functionality which may serve during phases prior to official approval that do not require all the assurances of binding decisions.

In this regard, the privacy demands and methods are unclear.

The demands are unclear in the sense that it is difficult to determine what privacy characteristics one would want in a deliberative setting, or whether the notion of privacy is entirely incompatible with deliberation. The methods are unclear in the sense that privacy technologies one could apply to the case of deliberation are not as well defined as they are for other purposes such as the aforementioned secure electronic voting case with more than 30 years of academic research behind it.

Fortunately it may be the case that these technologies can be repurposed for slightly different cases than voting, as we will analyze later.

Let's briefly look at prior research on the first aspect. For example (De Cindio, 2012)

> Our long community-network experience suggests that this weak form of identification is inadequate, if a trustworthy social environment that encourages public dialogue and deliberation is to be created. Online identity should, insofar as possible, reflect offline identity: if citizens wish to get a public answer from someone who plays a public role and appears online with her/his actual identity, they must do the same. They have to 'show their face' and take responsibility for participating under their actual identity (Casapulla et al., 1998). This serves also to root the online community in the "proximate community" served by the network (Carroll & Rosson, 2003).

The authors consider that deliberation is indeed incompatible with anonymity and therefore privacy, as we mentioned above. Citizens participating anonymously would not be accountable or responsible for their contributions, and these are important requirements for deliberation to take place. On the other hand, the authors also observe that

> Nevertheless, even in online deliberative contexts, there are cases in which it is worth protecting participants' privacy. This might occur during public consultations and discussion on sensitive issues or public assessments of an official that could bounce back on the participants, as in the case of the assessment of a teacher by his/her students as well as in the case of doctors rated by patients (e.g. http://www.patientopinion.org.uk/). In all these cases, there is the need to integrate a strong authentication policy (so that, e.g., only the students who have actually taken a class can rate the teacher) with secrecy techniques for protecting participants' identity. Software can help achieve this by obscuring the identity of the sender of a message in such critical discussion areas.

Although not specific to Decidim's features, literature on related topics (for example, anonymity applied to online commenting on news media) reflects this ambivalence.

- Greater anonymity may increase uncivil behaviour and the use of offensive words (Fredheim, Moore & Naughton, n.d.) (Cho & Acquisti, 2013).
- Greater anonymity may reduce comment quality (Diakopoulos & Naaman, 2011)
- Greater anonymity may reduce trust, cooperation and accountability (Cho & Acquisti, 2013)

Conversely

- Greater anonymity may increase participation (Fredheim, Moore & Naughton, n.d.) and engagement (Davies, 2009)
- Greater anonymity may yield more information (Diakopoulos & Naaman, 2011) and produce more honest (Davies, 2009) and original ideas (Connolly, Jessup & Valacich, 1990)
- Greater anonymity may produce more equal (Flanagin et al., 2002) (Klenk & Hickey, 2011) interactions leading to free discussion of controversial issues.

It is hard to arrive at conclusions from example like these for mainly one reason. Prior research is not specific any enough to warrant practical recommendations for Decidim, only general trends to bear in mind. Some of the drawbacks and benefits mentioned above may not appear when using anonymized pseudonyms, since that technique exists at a midpoint in the anonymity spectrum (Identity and Anonymity, 2016).

What we can take away from these comments, for and against the need for anonymity in deliberation, is that both possibilities have supporting arguments and neither would be entirely out of place in Decidim. It is here that the city council's decision becomes a motivating assumption/objective that we rely on for the rest of the document. Hints of this objective can be seen at the registration page:

The text surrounded in blue reads:

> Public name that will appear in your posts. In order to ensure anonymity can be any name.

which is a reflection of the city council's objective to ensure some degree of privacy. In the end, the city council has decided to allow for three types of user privacy profiles, allowing flexibility since as we have remarked above there seems to be no clear best solution. This also fits well with the remark (Cho & Acquisti, 2013).

> anonymous speech helps construct free discussion environment through the autonomous disclosure of personal identity (Zarsky, 2004)

The three profiles are

1. Public

   These are users whose identity is known to the system administrators (and therefore city council) and whose identity appears publically on the Decidim system for anyone to see.

2. Semi-public

   These are users whose identity is known to the system administrators (and therefore city council) but whose identity is hidden from the general public and does not appear publically on the Decidim system.

3. Anonymous

   These are users whose identity is unknown to both the system administrators (and therefore city council) and the general public. Allowing for this type of user is the objective of this document.

From now on we will refer to Users and Anonymous Users as belonging to the third category. Implementation strategies or details of the first two cases will no longer be discussed.

Thus, privacy objectives can be formulated as:

O1) Users must be dissociated from their real identity (henceforth, anonymized) such that User contributions are difficult to trace to their real world author.

O2) User's contributions must be subject to the same authorization and authentication restrictions as before any anonymization. In particular, users need to have been validated at some point in order to upvote proposals.

The phrase "difficult to trace" in O1 has a technical meaning:

» Users are anonymous in the technical sense that they belong to an anonymity set (Danezis & Diaz, 2008) composed of users in their same class[3].

> To enable the anonymity of a subject, there always has to be an appropriate set of subjects with potentially the same attributes. Anonymity is thus defined as the state of being not identifiable within a set of subjects, the anonymity set. The anonymity set is the set of all possible subjects. With respect to acting entities, the anonymity set consists of the subjects who might cause an action.

The degree of anonymity is given by the size of this set (disregarding inference and information-theoretic metrics). Users function as pseudonyms (Danezis & Diaz, 2008).

» The anonymity of users is protected by the conjunction of computational hardness assumptions with auxiliary security assumptions[4].

As mentioned earlier, it is the combination of O1 and O2 that instantiates the tension between privacy and integrity and makes technical solution difficult and requires cryptography. Dropping either of the two objectives makes everything much simpler; without privacy it is easy to ensure integrity, without integrity it is easier to ensure privacy.

Note that these two objectives imply the existence of User tuples as pseudonyms, a feature that could itself be questioned since it gives rise to privacy concerns we referenced above when talking of inferred identity. One could theoretically address this by unlinking data records that are currently grouped as shown on the left column of the table above. In other words (Danezis & Diaz, 2008).

> [Unlinkability] ensures that a user may make multiple uses of resources or services without others being able to link these uses together. [...] Unlinkability requires that users and/or subjects are unable to determine whether the same user caused certain specific operations in the system.

However, we will not go into this further as it would be infeasible[5] to fit in with Decidim's features.

For some more detail along technical lines, we add two extra objectives

O3) Decidim's features and behaviours remain unchanged given user anonymization

O4) Anonymization should be as transparent as possible for the Decidim software

We can begin by formalizing the first objective in terms of the data that defines a User[6].

» A Registered User is a tuple R = {Debates, Proposals, Comments, Upvotes, Email}

» A Verified User is tuple V = {Debates, Proposals, Comments, Upvotes, Email, CitizenID}

We wish to unlink part of these records such that

» An Anonymized User is a tuple A = {Debates, Proposals, Comments, Upvotes}

Privacy leaks for Anonymized Users now become the following

3 This class may be all users, anonymizing groups for staged re-anonymization or privilege groups.

4 See Appendix A.

5 Recall that one of the possible classifications of Decidim is as a reputation system (see Section 3). It is no coincidence that one of the benefits of pseudonyms is that they allow building reputation (as well as dealing with misbehaviour).

6 We are including both upvotes and downvotes in "Upvotes". Also, this is a simplification of the data actually stored in the Consul database.

	User information	Citizen information
Registered	Authored debates/proposals Authored comments Upvoted/downvoted comments	Inferred identity
Verified	Authored debates/proposals Authored comments Upvoted/downvoted comments **Upvoted proposals**	Inferred identity

In the top right, the Citizen's email has disappeared, leaving only Inferred identity mined from user information. In the bottom right, the CitizenID has disappeared, and in this case Inferred identity is no longer redundant. We stress that meeting objective O1, the unlinking of CitizenID, cannot remove leaks obtained through inference.

Moving on to O2, this objective ensures the integrity of decisions made through Decidim despite the fact that Users have been anonymized. In other words, Anonymized Users must correspond to Validated Users. This can be formalized as

» There is an injection[7] Anon: (**R** *U* **V**) → ***A*** from the set of non-anonymized users to the set of Anonymized Users that preserves user access and privileges.

We won't go into more details about O3 and O4, they are more about how anonymization should be implemented according to sound architectural principles than about requirements for privacy. Nonetheless, these objectives are included as it is important that Decidim's software, infrastructure and usability is not too negatively impacted by the protocols and implementations required by any anonymization proposals made in this document.

In other words, Decidim's software and operations should be as insulated as possible from the technical details of anonymization, and its mechanics and interactions should function almost identically with anonymization as a "given". Following this principle should also protect user experience with anonymization again functioning as a given[8].

7 Function (mathematics) - Wikipedia 2016.

8 Of course, anonymization will have some impact on users as they will have to participate in whatever procedure is necessary, but ideally this should be a one-off cost that once incurred becomes transparent.

4 A baseline solution

Here we present a trivial, "null" method for attaining the goals of anonymization. This method will to serve as a baseline with which to compare subsequent proposals and is not meant to be applied in practice. We will see why.

Recall that we defined a Verified User as a tuple

V = {Debates, Proposals, Comments, Upvotes, Email, CitizenID}

that when anonymized yields an An Anonymized User

A = {Debates, Proposals, Comments, Upvotes}

This is an abstract representation that is implemented by some concrete data storage implementation within Decidim's software, hardware and administrative infrastructure. This implementation will have some feature that allows the linking together of each of the elements of the tuple. The baseline solution is simply the removal of the concrete linking mechanism corresponding to the composition of elements into a tuple (at the abstract level).

Recall our definition of anonymization as an injection between Validated Users and Anonymized Users

» There is an injection Anon: ***(R U V)*** → ***A*** from the set of non-anonymized users to the set of Anonymized Users which preserves user access and privileges.

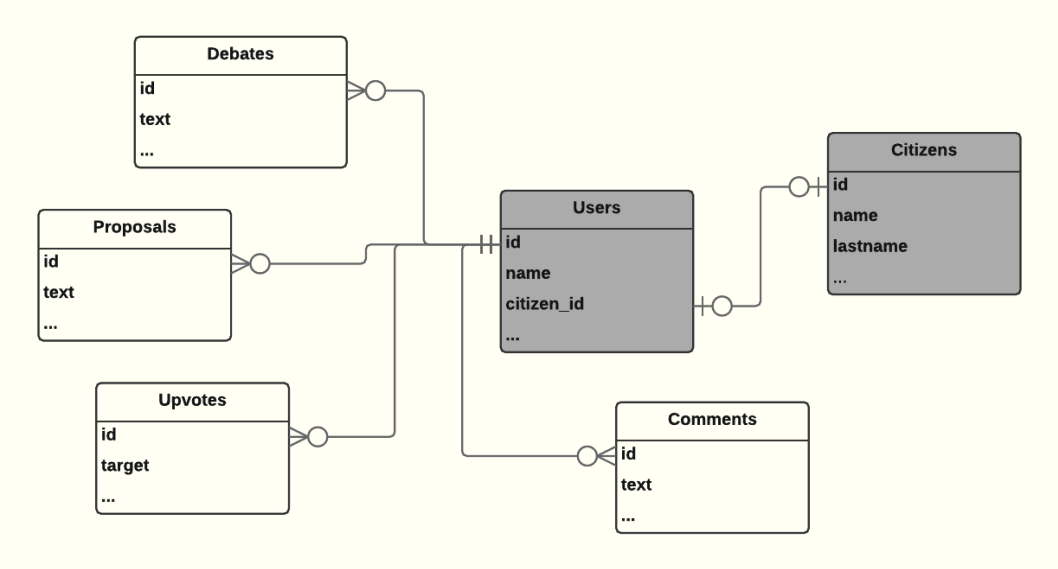
In the base solution the injection is implemented by some process that carries out mutation ("Pure Function – Wikipedia", 2016) on the data storage for users. For clarity, we add some detail capturing the notion of preserving user privileges by adding a reference to the authentication token, usually a password, and some abstract placeholder for privileges.

Anon({D, P, C, U, Email, CitizenID, auth, privileges}) → *{D, P, C, U, auth, privileges}*

Because the process performs mutation, the original data is replaced by the anonymized data and the correspondence information is lost; the linkage is destroyed. The citizen can still log in and operate normally as privileges and auth are unchanged. In the baseline solution we can consider Anon to be a partial function[9] between user records before and after they are modified.

Consider an example where the data storage implementation is a set of tables with relationships mediated by foreign keys. These foreign keys are the concrete linkage mechanism, and the removal is, in database terminology, to DROP them (including the columns). Here's a simple visualization

9 We are talking of a function in the mathematical sense, a type of relation.



What we are interested in above is the linking foreign key between Users and Citizens which is shown as an arrow in Crow's foot notation (Barker, 1990). Anonymizing Users in this case is the dropping of said foreign key and column such that is no longer possible to establish the correspondence between both pieces of information in the database.

However, even in this simplified case there are flaws. If we performed the two mentioned database operations to carry out the anonymization and left it at that we would problems. What if the database had been backed up? Even if the live database contained no correspondence it would still be present in any backups[10] performed earlier. So we'd have to have kept track of every backup every made and their location and either remove the correspondence from them or destroy them. As long as one backup remained present somewhere on some piece of hardware infrastructure the anonymization would have failed. This is without even considering malicious behaviour from some system administrator or operator that had secretly performed some backup of copy of said sensitive data.

But it gets worse. Not only would we have to keep track of backups, but User-Citizen correspondences could also likely be extracted from database transaction logs with some processing. So we'd either have to manipulate these logs or destroy them. And of course, all of these operations for both backups or logs would have to be persisted effectively to disks which can contain information even after delete or modifications take place (Data Recovery – Wikipedia, 2016).

This shows that even in a simple example of database storage technology as the concrete realization of User tuples, it's very hard to ensure that anonymization has been carried out exhaustively, and that's without factoring in malicious behaviour by any one of the actors with access to the information in any of its possible forms.

At this point it seems clear that this solution technically unsound.

But once again, this is not the worst drawback of the baseline solution. The more serious problem is conceptual rather than technical. Recall that the objective of anonymization is to maintain user privacy with respect to not only the general public but also the city council itself. If the point is to protect this privacy even from the city council, it makes no sense employ a solution in which the anonymizing process carried out by the city council cannot be verified or audited. In other words, an anonymizing procedure that considers the city council as a possible adversary must be public-

10 Even though some records may not be present.

ally verifiable without having to trust the administrators. Any other process requires an element of trust which is precisely what one is trying to avoid in the first place.

In summary, the baseline solution suffers from two serious problems, the second of which is an outright showstopper.

- Removing correlation data may be complex and brittle, prone to omissions and even malicious behaviour by dishonest administrators.
- There is no way for the general public to verify that these internal operations have been carried out correctly and honestly. If one wishes to attain privacy with respect to the city council itself, this approach is conceptually incoherent.

Instead we would like a solution one can be confident of at a technical level, and more importantly whose privacy guarantees can be publicly verified by anyone. Both these desiderata can be embodied in an extra objective

O5) Anonymization should be technically and publicly verifiable by third parties

5 A general, anonymous-by-construction scheme

This section describes a general anonymization scheme that attempts to satisfy the objectives collected so far. These are

O1) Users must be dissociated from their real identity (henceforth, anonymized) such that User contributions are difficult to trace to their real world author.

O2) User's contribution must be subject to the same authorization and authentication restrictions as before any anonymization. In particular, Users need to have been validated at some point in order to upvote proposals.

O3) Decidim's features and behaviours remain unchanged given user anonymization.

O4) Anonymization should be as transparent as possible for the Decidim software.

O5) Anonymization should be technically and publicly verifiable by third parties.

We begin by addressing one of the shortcomings of the baseline solution described in the last section, which gave rise to objective O5. The problem in question stems from the fact that when approaching anonymization by removing information it is difficult to ensure that all correspondences have been effectively removed; the solution is inherently unsafe.

Instead of starting from identified users and removing correspondence information, we can follow the inverse approach. We can start from empty users and only incrementally add information necessary for the system to work as before. These users are anonymous by default, or by *construction*. This makes technical verification simpler since we do not need to go through all previous existing data, only to make sure no privacy compromising data is added to our new anonymous users. Additionally, anonymization is publicly verifiable since it does not depend on operators deleting information; instead the operators never have access to that information in the first place.

With this approach Anon is a function between a set of existing user data and a set of newly created user data; instead of mutating existing user information the implementing process creates new user data. Let's start with a version of Anon that maps to empty records

$$Anon(\{D, P, C, U, Email, CitizenID, auth, privileges\}) \rightarrow \{\varnothing, \varnothing, \varnothing, \varnothing\}$$

The result contains only empty sets ($\varnothing$) as placeholders for user contributions and no reference to any citizen information. Because it is empty the data is clearly anonymous. Unfortunately it is also useless since the citizen in question cannot authenticate against it and operate in the system. We need to add to this data the necessary authentication token such that the citizen can use it. Assuming for simplicity that privileges are the same for all users, we could try something like

$$Anon(\{D, P, C, U, Email, CitizenID, auth\}) \rightarrow \{\varnothing, \varnothing, \varnothing, \varnothing, auth\}$$

Now the new user is accessible by the original citizen, because the authentication token auth matches the token the citizen already had for the old user. Unfortunately the new user is no longer anonymous for the same reason: the auth token can be used to match against the old user which is not anonymous. What we need is to map to new user data with an authentication token that

1) Is difficult to trace back to the original token
2) Is known only by the citizen who knows the old authentication token, and no other citizen

We model this as a one-way partial function (One-Way Function – Wikipedia, 2016) that maps the old token to the new token, and which, by assumption, can only be evaluated by the citizen who knows the old token

$f : \{0,1\}^* \to \{0,1\}^*$

which when combined with Anon results in

$\text{Anon}(\{D, P, C, U, \text{Email}, \text{CitizenID}, \text{auth}\}) \quad \rightarrow \quad \{ \quad \oslash , \oslash , \oslash , \oslash \quad , f(\text{auth})\}$

With the addition of the new authorization token the new user remains anonymous but can now be accessed by citizens to operate in the system. Because the scheme is anonymous by construction it's easier to verify. In particular, the operators never have access to the correspondence between users and citizens (given the above assumptions). Hence, we can say, without a formal proof, that this scheme satisfies O1, O2[11] and O5. Note that the authorization token f(*auth*) plays the part of an (simple) anonymous credential (David Chaum, 1983).

Of course, this is just an abstract specification where the heavy lifting[12] is done by assumptions. What remains is the difficult technical challenge of finding some concrete implementation of this scheme that can be made operational and also meet the rest of the objectives, O3-O4. This will require the use of several cryptographic building blocks.

11 We are considering for the moment that all users have the same privileges. Dealing with different privileges is discussed later.

12 See Appendix A.

6 Building blocks

In this section we list several techniques to use as building blocks when constructing protocols to implement the anonymization scheme defined previously. For each we provide a short hint of its role in protocols as well definitions from the literature and corresponding references.

6.1 Cryptographically secure pseudorandom number generator

6.1.1 Intent

To generate authentication tokens that are unpredictable and therefore anonymous and secret (secure with respect to impersonation).

6.1.2 Definition

We have (Menezes, van Oorschot & Vanstone, 1996)

A pseudorandom bit generator (PRBG) is a deterministic algorithm which, given a truly random binary sequence of length k, outputs a binary sequence of length l k which "appears" to be random. The input to the PRBG is called the seed, while the output of the PRBG is called a pseudorandom bit sequence.

A PRBG that passes the next-bit test (possibly under some plausible but unproved mathematical assumption such as the intractability of factoring integers) is called a cryptographically secure pseudorandom bit generator (CSPRBG).

We do not specify this building block further; the available implementations will depend on the technological environment (eg browser). However, the implementation must be chosen to be cryptographically secure as defined above.

6.2 Cryptographic hash function

6.2.1 Intent

To generate authentication tokens that are unique (collision resistance) and therefore secret.

6.2.2 Definition

We have (Menezes, van Oorschot & Vanstone, 1996)

A hash function (in the unrestricted sense) is a function h which has, as a minimum, the following two properties:

1. *compression — h maps an input x of arbitrary finite bitlength, to an output h(x) of fixed bitlength n.*
2. *ease of computation — given h and an input x, h(x) is easy to compute.*

To facilitate further definitions, three potential properties are listed (in addition to ease of computation and compression as per Definition 9.1), for an unkeyed hash function h with inputs x, x and outputs y, y.

1) *preimage resistance — for essentially all pre-specified outputs, it is computationally infeasible to find any input which hashes to that output, i.e., to find any preimage x such that h(x) = y when given any y for which a corresponding input is not known.1*
2) *2nd-preimage resistance — it is computationally infeasible to find any second input which has the same output as any specified input, i.e., given x, to find a 2nd-preimage x = x such that h(x) = h(x).*
3) *collision resistance — it is computationally infeasible to find any two distinct inputs x, x which hash to the same output, i.e., such that h(x) = h(x). (Note that here there is free choice of both inputs.)*

As in the previous section, we do not specify exactly which hash function to use. This choice depends on which hashes are considered cryptographically secure at the time of the choice(Hash Function Security Summary – Wikipedia, 2016). At the time of writing sha-2 (SHA-2 – Wikipedia, 2016) is a safe choice.

6.3 ElGamal public-key encryption scheme[13]

6.3.1 Intent

To keep authentication tokens secret with respect to system administrators.

6.3.2 Definition

We have (Tsiounis & Yung, 1998)

The ElGamal public key encryption scheme is defined by a triplet (G, E, D) of probabilistic polynomial time algorithms, with the following properties:

- *The system setup algorithm, S, on input 1^n, where n is the security parameter, outputs the system parameters (P, Q, g), where (P, Q, g) is an instance of the DLP collection, i.e., P is a uniformly chosen prime of length $|P| = n+\delta$ for a specified constant δ, and g is a uniformly chosen generator of the subgroup GQ of prime order Q of Z_P, where $Q = (P - 1)/\gamma$ is prime and γ is a specified small integer.*
- *The key generating algorithm, G, on input (P, Q, g), outputs a public key, e = (P, Q, g, y), and a private key, d = (P, Q, g, x), where*
  - *x is a uniformly chosen element of Z_Q, and*
  - *$y \equiv g^x \bmod P$*
- *The encryption algorithm, E, on input (P, Q, g, y) and a message $m \in G_Q$, uniformly selects an element k in Z_Q and outputs $E((P, Q, g, y), m)=(g^k \bmod P), my^k \bmod P))$.*
- *The decryption algorithm, D, on input (P, Q, g, x) and a ciphertext (y1, y2), outputs $D((P, g, x), (y1, y2)) = y2(y1x)^{-1} \bmod P$.*

ElGamal has been proven (Tsiounis & Yung, 1998) semantically secure under the Decisional Diffie/Hellman assumption:

Theorem 1. If the ElGamal encryption scheme is not secure in the sense of indistinguishability, then there exists a p.p.t. TM that solves the decision Diffie-Hellman problem with overwhelming probability.

A typical choice for ElGamal modulus is a safe prime p = 2q + 1, generating a message space of quadratic residues.

13 (ElGamal, n.d.)

6.4 Signed ElGamal

6.4.1 Intent

To make ElGamal non-malleable under chosen plaintext attack (in addition to standard semantic security)

6.4.2 Definition

We have (Schnorr & Jakobsson, 2000)

The private/public key pair for encryption is x, $h = g^x$ where x is random in Z_q. The basic encryption scheme is for messages in M = G, ElGamal ciphertexts are in G × M, the added Schnorr signature signs pairs in G × M and uses a random hash function $H : G^2 \times M \to Z_q$.

In order to encipher a message $m \in G$, we pick random $r, s \in_R Z_q$, compute $g^r , mh^r , c := H(g^s , g^r , mh^r)$ and $z := s + cr$ and output the ciphertext $(g^r , mh^r , c, z) \in G^2 \times Z^2_q$.

A signed ciphertext (g^r , mh^r , c, z) consists of an ElGamal ciphertext (g^r , mh^r) and a Schnorr signature (c, z) of the "message" (g^r , mh^r) for the public signature key g^r . The signature (c, z) does not contain any information about m as (c, z) depends on m exclusively via some hash value that is statistically independent of m.

Signed ElGamal makes the cryptosystem non-malleable under chosen plaintext attack[14].

6.5 RSA blind signature [15]

6.5.1 Intent

To allow signing of tokens (and therefore authenticating users) while preserving anonymity.

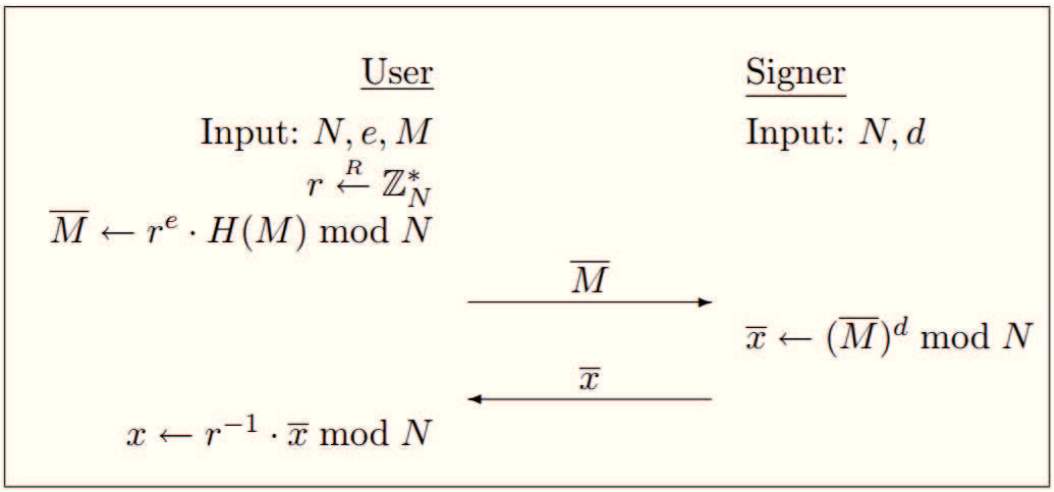
6.5.2 Definition

We have (M. Bellare et al., 2003)

The RSA blind signature scheme [12] consists of three components: the key generation algorithm [...] the signing protocol [...]; and the verification algorithm. The signer has public key N, e and secret key N, d. Here $H: \{0, 1\}^* \to Z^*_N$ is a public hash function which in our security analysis will be modeled as a random oracle. In that case, the signature schemes is the FDH-RSA scheme of (Goldwasser, Micali, and Rivest 1988). A message-tag pair (M, x) is said to be valid if x e mod N is equal to H(M). The verific - ation algorithm is the same as that of FDH-RSA: to verify the message-tag pair (M, x) using a public key (N, e), one simply checks if the message-tag pair is valid.

14 See also Bernhard, Pereira & Warinschi 2012.

15 (Chaum, 1983)



RSA blind signatures have been proven (M. Bellare et al., 2003) to be blind and unforgeable (Pointcheval & Stern, 1996).

6.6 Partially blind WI-Schnorr signature[16]

6.6.1 Intent

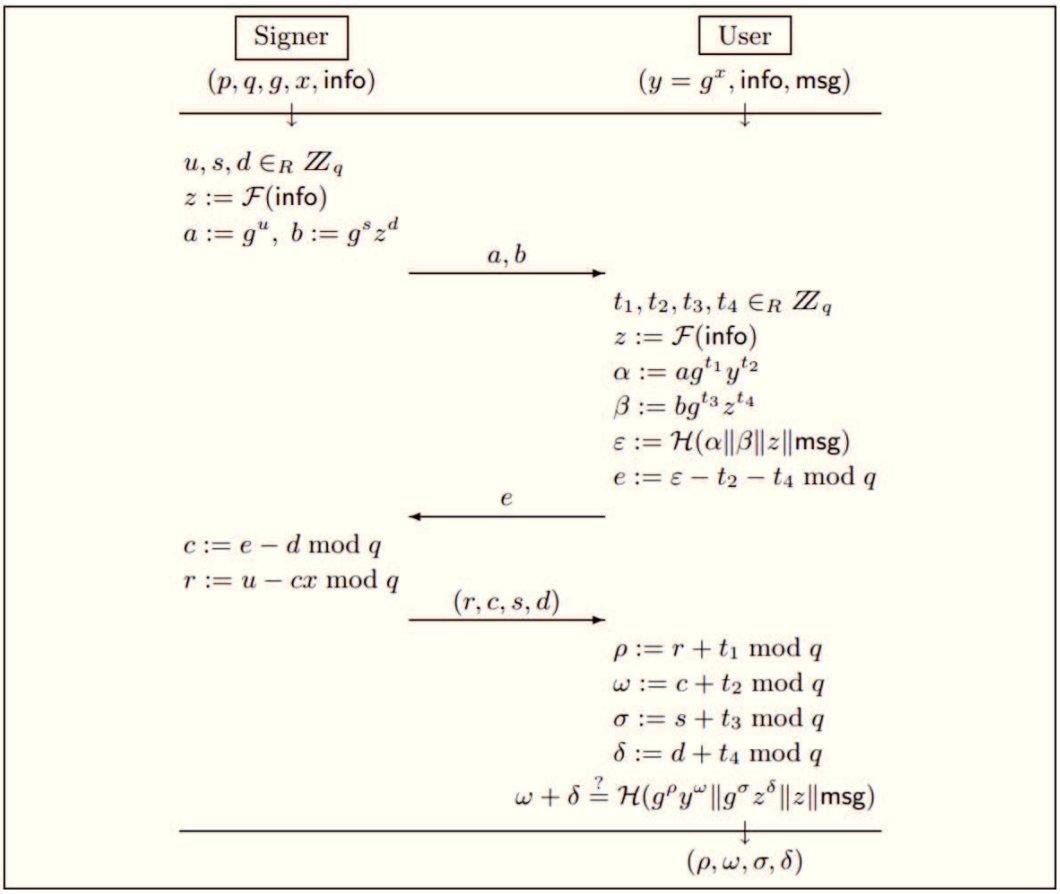
To allow signing of tokens (and therefore authenticating users) while preserving anonymity. Common info allows encoding variable and ad-hoc anonymity sets.

6.6.2 Definition

We have (Abe & Okamoto, 2000)

Let GDL be a discrete logarithm instance generator that takes security parameter n and outputs a triple (p, q, g) where p, q are large primes that satisfy q|p − 1, and g is an element in Z_p whose order is q. Let <g> denote a subgroup in Z_p generated by g.

16 (Abe & Fujisaki, 1996)



Let H : {0, 1} → Z_q and F : {0, 1} → <g> be public hash functions. Let x ∈ Z_q be a secret key and y := g^x be a corresponding public key. Signer S and user U first agree on common information info in an predetermined way. They then execute the signature issuing protocol illustrated [above]. The resulting signature for message msg and common information info is a four-tuple (ρ, ω, σ, δ). A signature is valid if it satisfies

$\omega + \delta \equiv H(g^\rho y^\omega \| g^\sigma F(info)^\delta \| F(info) \| msg) \pmod q$.

Partially blind WI-Schnorr signatures have been proven to be blind and unforgeable[17].

6.7 Distributed ElGamal

6.7.1 Intent

To allow distribution of trust pertaining to privacy of authentication tokens.

6.7.2 Definition

We have (Haenni, 2016)

A threshold cryptosystem, which is limited to the particular case of t = n, is called distributed cryptosystem. A simple distributed version of the ElGamal cryptosystem results from setting x = $\Sigma_i x_i$ To avoid that x gets publicly known, each of the n parties secretly selects its own key share $x_i \in_R Z_q$

17 For specific meanings of blind and unforgeable given in Abe & Okamoto, 2000.

and publishes $y_i = g^{x_i}$ as a commitment of x_i. The product $y = \Pi_i y_i = g^{\Sigma x_i} = g^x = g^x$ is then the common public encryption key.

If $E = (a, b) = Enc_y(m, r)$ is a given encryption, then m can be jointly recovered if each of the n parties computes $a_i = a^{-x_i}$ using its own key share x_i. The resulting product $a^{-x} = \Pi_i a_i$ can then be used to derive $m = Dec_x(E) = a^{-x} \cdot b$ from b. Instead of performing this simple operation in parallel, it is also possible to perform essentially the same operation sequentially in form of a partial decryption function $Dec_{x_i}(E) = (a, a^{x_i} \cdot b)$. Applying Dec'_{x_i} "removes" from E the public key share y_i by transforming it into a new encryption $E' = Dec'_{x_i}(E)$ for a new public key $y \cdot y_i^{-1}$. If all public key shares are removed in this way (in an arbitrary order), we obtain a trivial encryption (a, m) from which m can be extracted.

Details of the zero knowledge proofs of key generation and decryption can be found in the reference. The same method of multiplying private shares to obtain the public key is used in Glondu, 2016.

6.8 Distributed threshold ElGamal[18][19]

6.8.1 Intent

To allow fault tolerant distribution of trust pertaining to privacy of authentication tokens.

6.8.2 Definition

We have (Cortier et al., 2013)

Let D = (DistKG, Enc, ShareDec, Rec) be then the threshold cryptosystem:

- DistKG(1^λ, t, l)

1. *Each party P_i chooses a random t-degree polynomial $f_i(x) = a_{i0} + a_{i1}x + \ldots + a_{it}x^t \in Z[x]$ and broadcasts $A_{ik} = g^{a_{ik}}$ for k = 0, . . . , t. Denote the secret held by P_i as $s_i = f_i(0)$ and let $Y_i = g^{fi(0)}$ g. Each party P_i computes shares $s_{ij} = f_i(j)$ mod q of its own secret si for j = 1, . . . , l and sends $s_{ij} \in Z_q$ secretly to party P_j.*
2. *Each party P_j verifies the shares he received by checking for i = 1, . . . , l:*

   $$g^{s_{ik}} = \prod_{k=0}^{t} (A_{ik})^{jk} \quad (1)$$

   If a check fails for an index i then P_j broadcasts a complaint against P_i.
3. *Party P_i reveals share $s_{ij} \in Z_q$ if it receives a complaint against him by party P_j . If any of the revealed shares s_{ij} fails to satisfy Equation 1, then P_i is disqualified. Let us define the set QUAL $\neq \emptyset$ as the set of qualified players.*
4. *The public key is computed as $pk = \Pi_{i \in QUAL} Y_i$. Each P_j sets his share of the secret key as $x_j = \Sigma_{i \in QUAL} s_{ij}$ mod q. The virtual decryption key $x = \Sigma_{i \in QUAL} s_i$ mod q is not needed to be known to be able to decrypt. The public verification keys are computed as $vk_j = \Pi_{i \in QUAL} g^{sij}$ for j = 1, . . . , l.*

- Enc(pk, m) outputs $C = (R, S) = (g^r, Y^r \cdot m)$ for a plaintext $m \in G$ and randomness $r \leftarrow_R Z_q$.

- ShareDec(sk_i, vk_i, C) outputs $(i, c_i = R^{x_i})$.

*- Rec(pk, vk, C, **C**) parses $C = (R, S)$, $\mathbf{C} = \{ c_{i1}, \ldots, c_{i\,t+1} \}$ and outputs $m = S \cdot \left(\prod_{j \in X} c_j^{\lambda_j^X} \right)^{-1}$*

with $X = \{ i_1, \ldots, i_{t+1} \}$, where the λ_j^X's are the Lagrange coefficients, $\lambda_j^X = \prod_{k \in X \setminus \{j\}} \frac{k}{k-j} \in Z_q^*$

18 (Cramer, Gennaro & Schoenmakers 1997)

19 (Gennaro et al., 1999)

We thus have that $\sum_{j \in X} f(j)\lambda_j^X = f(0)$ *for any polynomial f of degree at most t.*

6.9 ElGamal re-encryption mixnet[2021] with Terelius-Wikstrom proofs of shuffle[22]

6.9.1 Intent

To anonymize encrypted tokens prior to (joint) decryption.

6.9.2 Definition

Note that the below description refers to ballots and voting. This is not important for the fundamentals of the mixnet. We have (Rivest, 2004)

Recall that in the El-Gamal encryption scheme, an encryption of a message m, with respect to a public key (p, g, y), consists of a pair (gr , myr), where all the operations are done modulo p, and $r \in_R Z_q$ *where q is a large prime dividing p − 1, where g is a generator of the subgroup of elements whose order divides q, and m is in this subgroup. The secret key corresponding to (p, g, y) is x such that* $g^x = y \pmod p$.

The El-Gamal encryption scheme has the following nice re-encrypting property: any encrypted message (a, b) = (g^r , my^r) *can be re-encrypted by choosing a random* $s \in_R Z_q$ *and computing* $(ag^s , by^s) = (g^{r+s} , my^{r+s})$. *Note that this re-encrypting operation results with a random ciphertext for the same message m.*

We are now ready to define the El-Gamal based re-encryption mix net:

1. *An El-Gamal public-key (p, g, y) is generated (in some distributed manner).*
2. *The initial encryption phase E simply encrypts all the ballots* $B_1, \ldots, B_n$ *by applying the El-Gamal encryption algorithm with the public-key (p, g, y). It then posts all the resulting ciphertexts* $(C_{1,0}, \ldots, C_{n,0})$ *on a bulletin board.*
3. *The i'th mix phase, on input a set of ciphertexts* $(C_{1,i-1}, \ldots, C_{n,i-1})$, *re-encrypts each ciphertext and permutes the resulting ciphertexts using a secretly chosen random permutation.*
4. *The final decryption phase D, given a set of ciphertexts* $(C_{1,k}, \ldots, C_{n,k})$, *simply decrypts all the ciphertexts in some distributed manner (in order to achieve robustness).*

Refer to (Terelius & Wikström, 2010) for shuffle proof details. These shuffles are used in the (Wikstrom, 2016) and (UniVote, 2016) systems among others.

6.10 Anonymous communication channel

Note that, unlike the previous cases, an anonymous communication channel should be seen more as a dependency (or even a limitation) rather than a building block in the schemes presented in section 7. This is because the dependency is an unavoidable consequence of the features of Decidim (as mentioned in O3), and not so much a freely chosen component of the solutions presented here.

Nonetheless we list it as a building block to make the dependency explicit in the scheme and protocol descriptions.

20 (D. L. Chaum 1981)

21 (Sako and Kilian 1995)

22 (Terelius and Wikström 2010)

6.10.1 Intent

To allow interactions with the system that retain anonymity with respect to network/ip identification.

6.10.2 Definition

We have (Danezis & Diaz, 2008)

Data communication networks use addresses to perform routing which are, as a rule, visible to anyone observing the network. Often addresses (such as IP addresses, or Ethernet MACs) are a unique identifier which appear in all communication of a user, linking of all the user's transactions. Furthermore these persistent addresses can be linked to physical persons, seriously compromising their privacy.

Anonymizing the communication layer is thus a necessary measure to protect the privacy of users, and protect computer systems against traffic analysis

The choice of anonymous communication channel is out of the scope of this document. Surveys can be found in (Danezis & Diaz, 2008) and (Edman & Yener, 2009).

6.11 Other relevant cryptographic techniques

For completeness we list pointers to the techniques in the literature that have not been used as building blocks but are still relevant to the problem. Many of these techniques do not fit well with the specific objectives for Decidim but could be explored for stronger anonymity properties (eg unlinkable contributions) (Danezis & Diaz, 2008).

- Algebraic MACs and Keyed-Verification Anonymous Credential - (Chase, Meiklejohn & Zaverucha, 2014)
- Anonymous Credentials Light - (Baldimtsi & Lysyanskaya, 2013)
- An efficient system for non-transferable anonymous credentials with optional anonymity revocation - (Baldimtsi & Lysyanskaya, 2013; Camenisch & Lysyanskaya, 2001)
- Fair Blind Signatures - (Stadler, Piveteau & Camenisch 1995)
- Ring Signatures without Random Oracles - (Chow et al., 2006)

7 Schemes

In this section we present schemes and protocols that are concrete specifications of the general anonymous-by-construction scheme in section 5. These specifications are composed of the building blocks of the previous section. The specification will be semi-formal using UML style sequence diagrams together with text descriptions of the main steps.

7.1 Building block abbreviations

For convenience, the building blocks listed in section 6 will be abbreviated according to

RNG	6.1 Cryptographically secure pseudorandom number generator
HASH	6.2 Cryptographic hash function
ELG	6.3 ElGamal public-key encryption scheme
SELG	6.4 Signed ElGamal
BLS	6.5 RSA blind signature
PBLS	6.6 Partially blind WI-Schnorr signature
DELG	6.7 Distributed ElGamal
TELG	6.8 Distributed threshold ElGamal
MIX	6.9 ElGamal re-encryption mixnet
ANC	6.10 Anonymous communication channel

7.2 Use cases

This section lists the user-anonymization related use cases that may or may not be supported by the protocols that make up each scheme. Use case realizations (Use-Case Analysis - Wikipedia 2016) are composed of one or two protocols.

7.2.1 Registration

This is the central use case that performs anonymization. Existing users initiate this use case by requesting a new anonymous user. Once this is complete, citizens will be registered with a new authenticated, anonymous User ready to participate in Decidim.

7.2.2 Recovery

Users may at some point forget or lose their passwords. Because they are anonymized, special procedures are necessary to prevent locking out citizens from participating. These procedures may require revoking the locked out user's anonymity as a lesser evil.

7.2.3 Re-anonymization

In section 3.2 we remarked that the privacy protecting approach based on the use of pseudonyms obtained through user anonymization cannot remove privacy leaks due to inference and data mining (inference attacks). The degree to which user contributions leak identity depends directly on how much information the adversary has available. As more information accumulates the space of possible identities is theoretically narrowed down. A loose analogy can be made with privacy budgets in the field of differential privacy (Dwork, 2006), where (Haeberlen, Pierce & Narayan 2011)

In this model, a third party is permitted to submit arbitrary queries over the database, but the data owner imposes a "privacy budget" that limits the amount of information the third party can obtain about any individual whose data is in the database. The system analyzes each new query to determine its potential "privacy cost" and allows it to run only if the remaining balance on the privacy budget is sufficiently high.

The purpose of the re-anonymization use case is to limit the amount of information that can be linked to any particular pseudonym. To do this, pseudonyms can be periodically "refreshed", such that contributions after a re-anonymization are only associated to the last pseudonym, and therefore unlinked to previous pseudonyms. This reduces the risk that an adversary has enough information to conduct an inference attack.

Of course, the theoretical benefits of executing this procedure have to be balanced with its negative impacts on user experience and usability. In particular, some of the benefits of using a pseudonym (for example, accumulating reputation or applying gamification) could suffer.

7.2.4 Privilege modification

In cases where variable privileges exist[23] it may be desirable to update them. One example could be a citizen changing location which could entail the right to participate in location sensitive processes (and loss of rights for previous ones). Because users are anonymized it is difficult to support this use case. Furthermore, special care must be taken to not compromise privacy as a result of intersecting anonymity sets[24]. For this reason privilege modification is included as a use case but with the proviso that it must only be carried out after careful analysis.

7.3 Schemes

7.3.1 B1

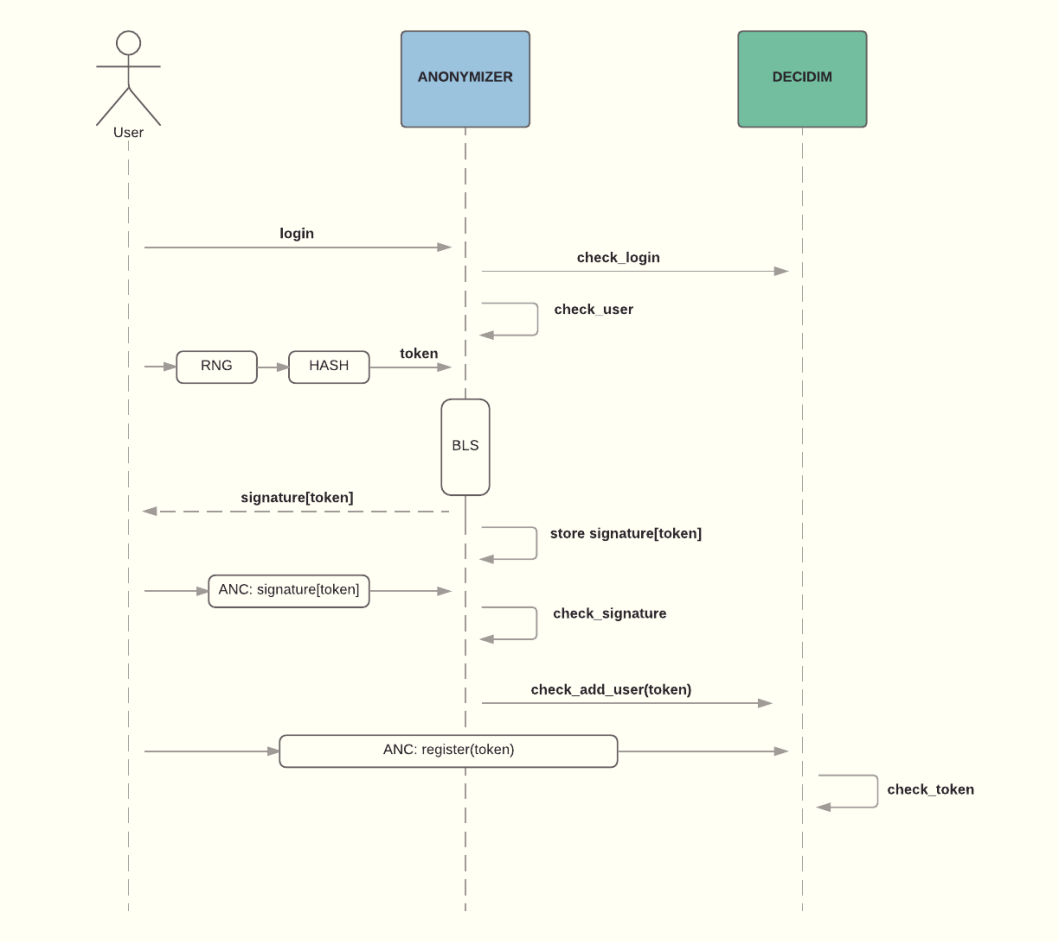
Anonymization is implemented with a blind signature. The token is blinded and then submitted for signing. The unblinded authentication token is then submitted through an anonymous channel for registration. This scheme is irreversibly anonymous and features no trustees. Deactivation is global.

23 See section 8.3

24 Intersection attacks (Ganta, Kasiviswanathan & Smith 2008).

Protocols

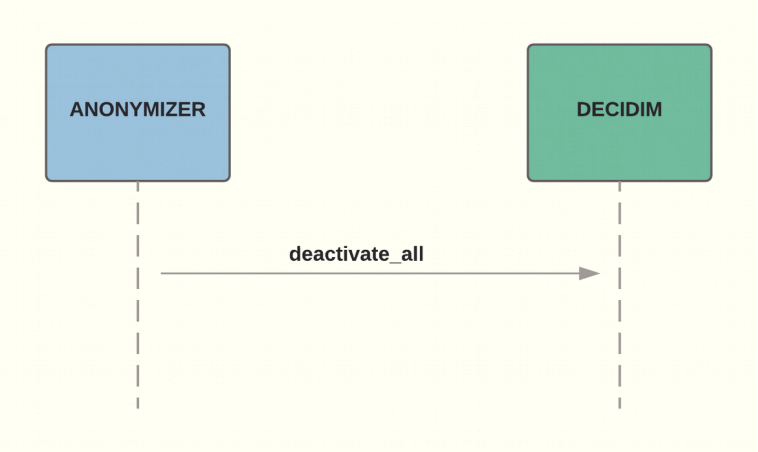
Registration



Registration

1. The user logs in with existing credentials (password)
2. The user is checked for an existing signature, if so that signature is returned.
3. The user's browser generates a random number which is concatenated with user id and then hashed producing a token
4. The user and anonymizer execute the blind signature protocol
5. The blind signature is returned to the user and also stored at the anonymizer
6. The user anonymously submits their unblinded token and signature
7. If correct and a corresponding user does not exist, a user is created for that token
8. The user completes registry by anonymously submitting the token

Deactivation



Global deactivation

1. Anonymizer requests deactivation of all users

Use case mappings

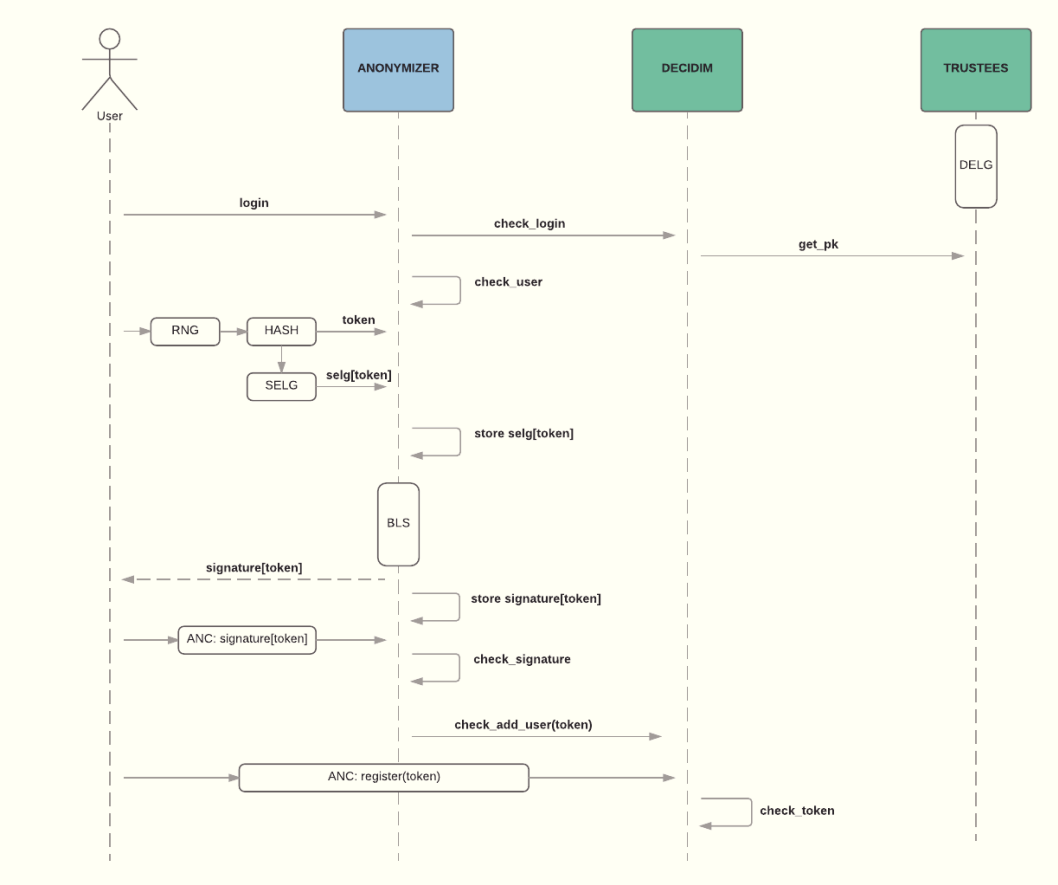
Registration	Registration
Recovery	Not supported
Re-anonymization	Deactivation + Registration
Privilege modification	Not supported

7.3.2 B2

Anonymization is implemented with a blind signature. This scheme supports individual deactivation through joint decryption of distributed ElGamal encrypted tokens sent by users. Because it is possible for users to modify the client to send a garbage token in the encrypted data, deactivation can be said to be voluntary. In other words, for normal operation the scheme is reversibly anonymous, but can theoretically be made irreversible.

Protocols

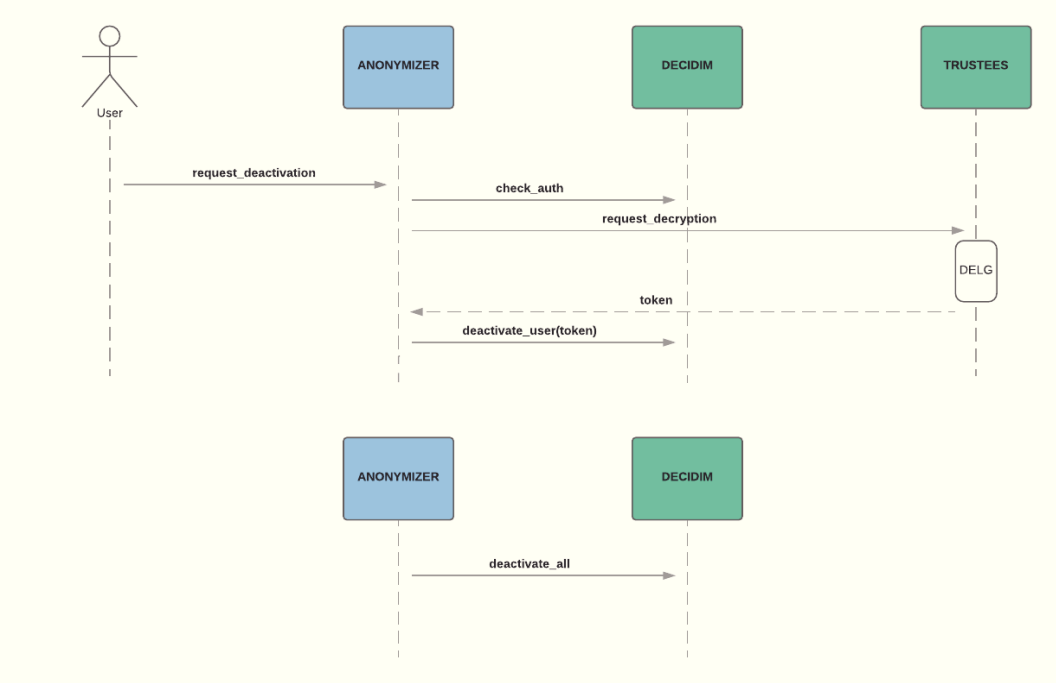
Registration



Registration

1. The trustees jointly generate a deactivation public key
2. The user logs in with existing credentials (password)
3. The user is checked for an existing signature, if so that signature is returned.
4. The user's browser generates a random number which is concatenated with user id and then hashed producing a token
5. The user and anonymizer execute the blind signature protocol
6. The user encrypts the token with the deactivation public key
7. The encrypted token is stored at the anonymizer
8. The blind signature is returned to the user and also stored at the anonymizer
9. The user anonymously submits their unblinded token and signature
10. If correct and a corresponding user does not exist, a user is created for that token
11. The user completes registry by anonymously submitting the token

Deactivation



Individual deactivation

1. User requests deactivation
2. User request is validated (eg email or physical id)
3. Trustees jointly decrypt token stored during registration
4. User is deactivated

Global deactivation

1. Anonymizer requests deactivation of all users

Use case mappings

Registration	Registration
Recovery	Deactivation + Registration
Re-anonymization	Deactivation + Registration
Privilege modification	Not supported

7.3.3 B2T

This scheme is a threshold version of B2 with Distributed ElGamal replaced by Threshold ElGamal.

Protocols

This protocols are the same as B2 with Distributed ElGamal replaced by Threshold ElGamal.

Use case mappings

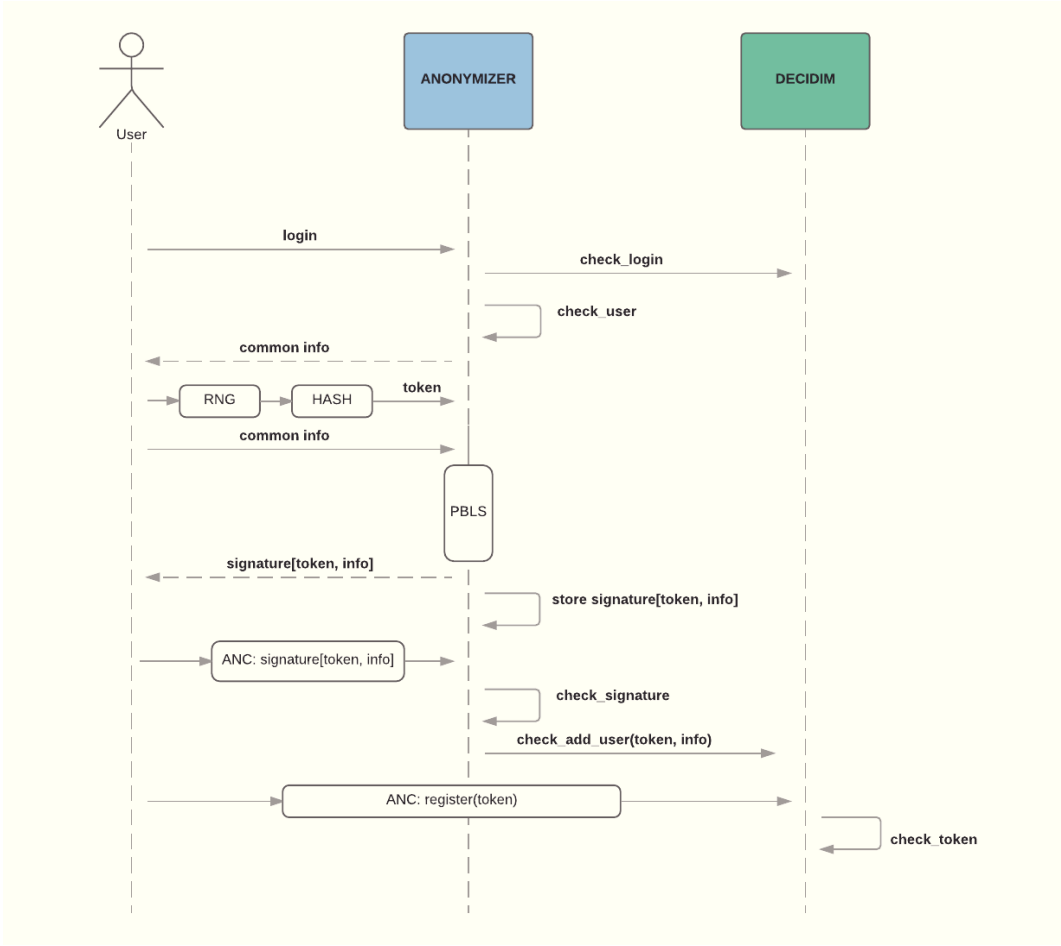
Registration	Registration
Recovery	Deactivation + Registration
Re-anonymization	Deactivation + Registration
Privilege modification	Not supported

7.3.4 P1

Anonymization is implemented with a partially blind signature. The user and anonymizer agree on some common information that is reflected in the public part of the signature. This allows transmitting information through the signature, such as privileges and anonymization group. Thus the schemes using partially blind signatures can support variable privileges and group deactivation. This scheme is irreversibly anonymous and features no trustees.

Protocols

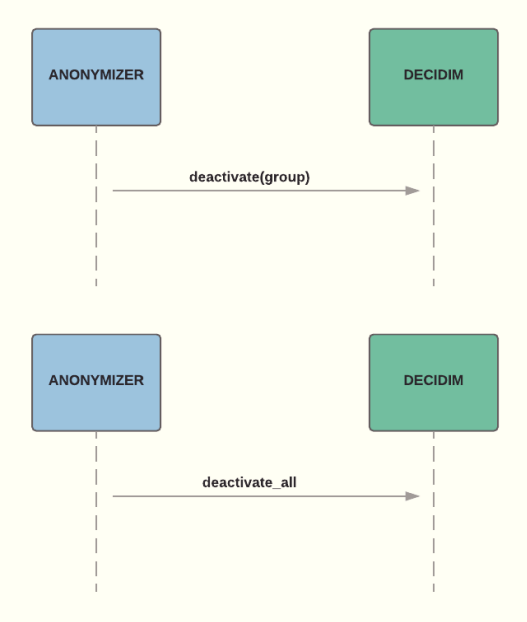
Registration



Registration

1. The user logs in with existing credentials (password)
2. The user is checked for an existing signature, if so that signature is returned.
3. The user's browser generates a random number which is concatenated with user id and then hashed producing a token
4. The user is presented with common signing information to validate
5. The user and anonymizer execute the partially blind signature protocol with token and common info
6. The partially blind signature is returned to the user and also stored at the anonymizer
7. The user anonymously submits their unblinded token and signature
8. If correct and a corresponding user does not exist, a user is created for that token and common info
9. The user completes registry by anonymously submitting the to

Deactivation



Group deactivation

1. Anonymizer requests deactivation of users belonging to an anonymity set as determined by the common information in the partially blind signature.

   This process can be repeated to allow for staged re-anonymization[25].

Global deactivation

1. Anonymizer requests deactivation of all users

25 See section 8.3

Use case mappings

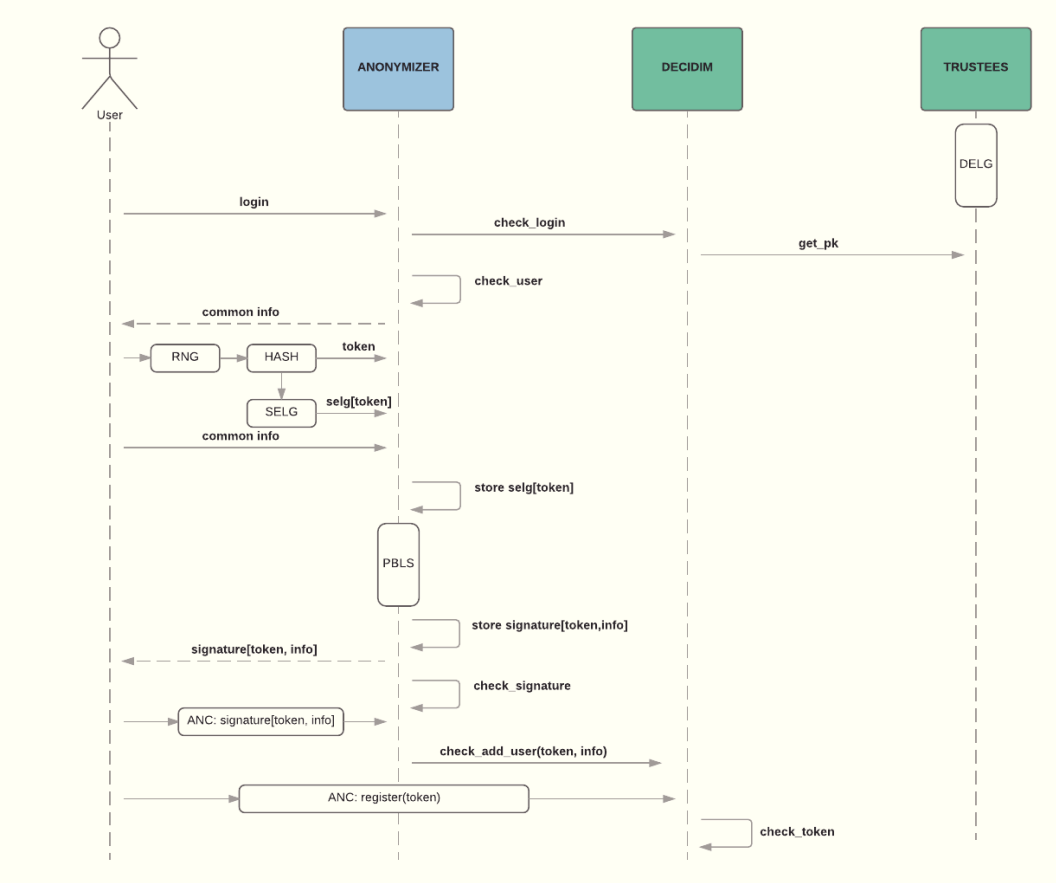
Registration	Registration
Recovery	Not supported
Re-anonymization	Deactivation + Registration
Privilege modification	Not supported

7.3.5 P2

Anonymization is achieved with a partially blind signature. The scheme supports variable privileges and deactivation groups. This scheme supports individual deactivation through joint decryption of distributed ElGamal encrypted tokens sent by users. It is reversibly anonymous, but can theoretically be made irreversible.

Protocols

Registration

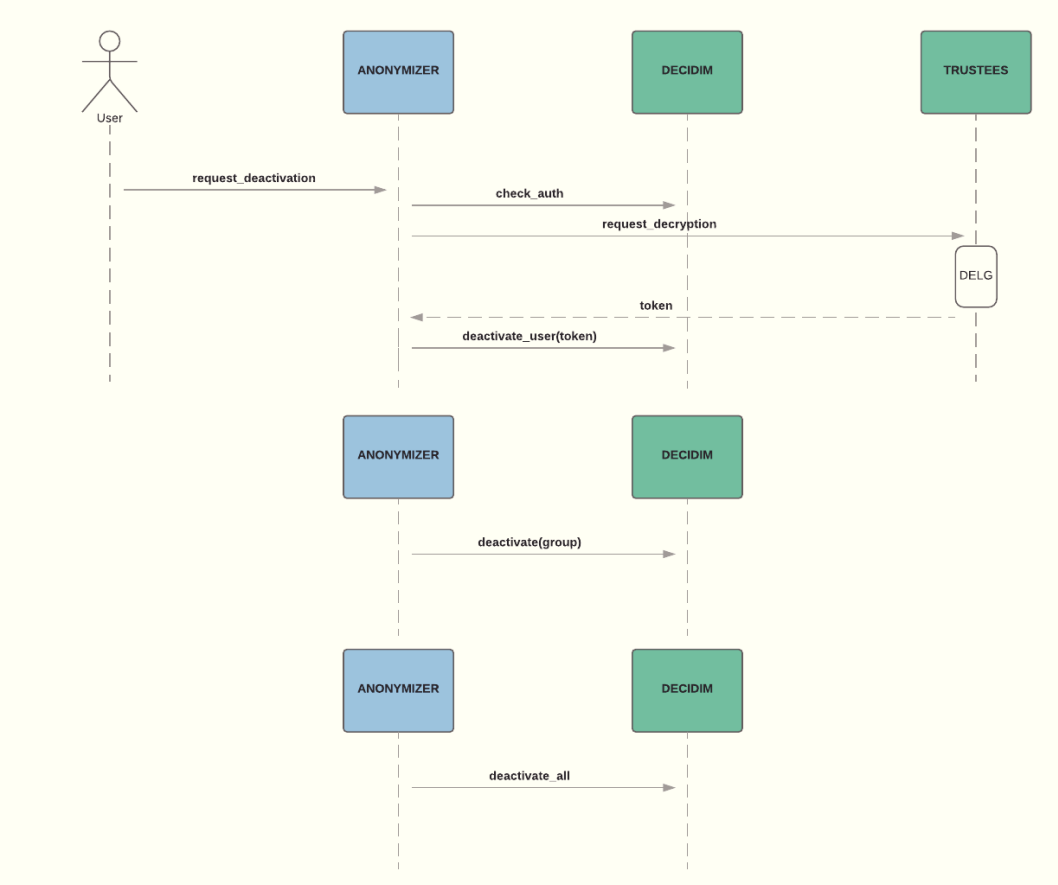


Registration

1. The trustees jointly generate a deactivation public key

2. The user logs in with existing credentials (password)
3. The user is checked for an existing signature, if so that signature is returned.
4. The user's browser generates a random number which is concatenated with user id and then hashed producing a token
5. The user is presented with common signing information to validate
6. The user and anonymizer execute the partially blind signature protocol with token and common info
7. The user encrypts the token with the deactivation public key
8. The encrypted token is stored at the anonymizer
9. The partially blind signature is returned to the user and also stored at the anonymizer
10. The user anonymously submits their unblinded token and signature
11. If correct and a corresponding user does not exist, a user is created for that token and common info
12. The user completes registry by anonymously submitting the token

Deactivation



Individual deactivation

1. User requests deactivation
2. User request is validated (eg email or physical id)

3. Trustees jointly decrypt token stored during registration
4. User is deactivated

Group deactivation

1. Anonymizer requests deactivation of users belonging to an anonymity set as determined by the common information in the partially blind signature.

This process can be repeated to allow for staged re-anonymization[26].

Global deactivation

1. Anonymizer requests deactivation of all users

Use case mappings

Registration	Registration
Recovery	Deactivation + Registration
Re-anonymization	Deactivation + Registration
Privilege modification	Not supported

7.3.6 P2T

This scheme is a threshold version of P2 with Distributed ElGamal replaced by Threshold ElGamal.

Protocols

The protocols are the same as P2 with Distributed ElGamal replaced by Threshold ElGamal.

Use case mappings

Registration	Registration
Recovery	Deactivation + Registration
Re-anonymization	Deactivation + Registration
Privilege modification	Not supported

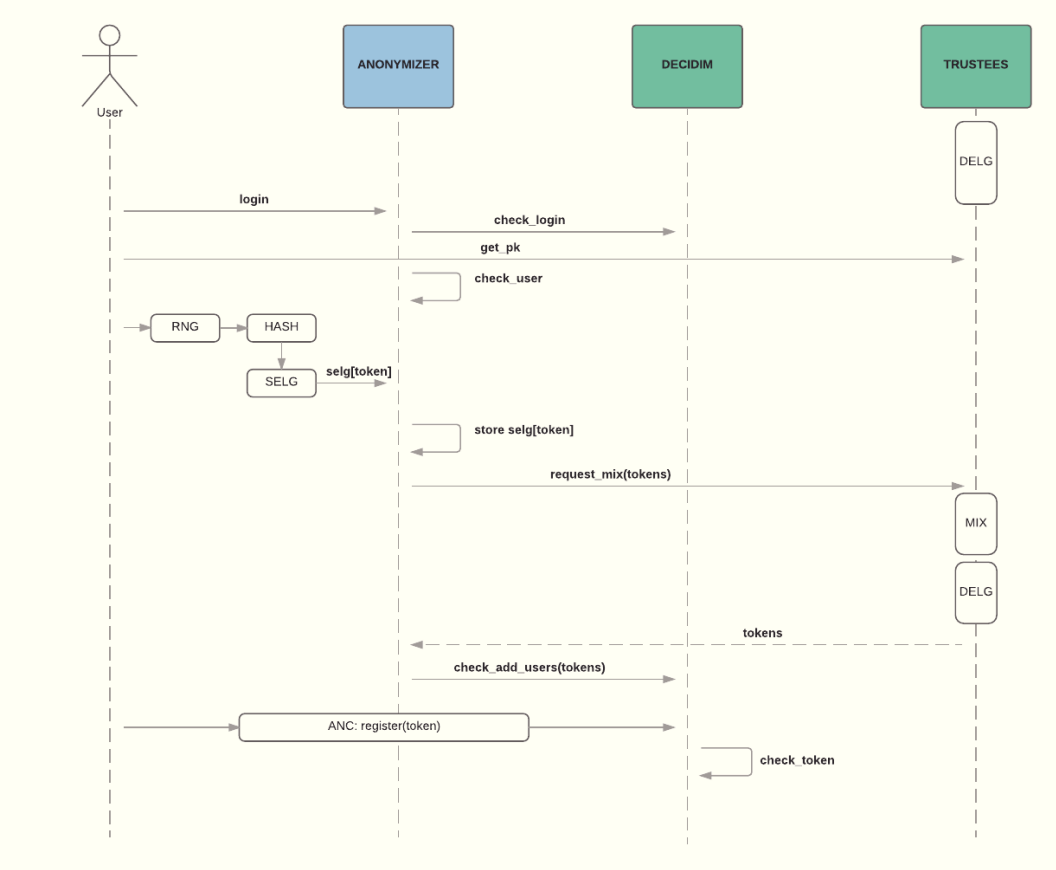
7.3.7 M1

Anonymization is achieved with a re-encryption mixnet. The scheme supports variable privileges and deactivation groups. In contrast to partially blind signatures, these groups can be formed dynamically (in partially blind signature schemes these groups must be determined at signature time) by collecting arbitrary ciphertexts. This scheme supports individual deactivation through joint decryption of distributed ElGamal encrypted tokens sent by users. It is unconditionally reversibly anonymous.

26 See section 8.3

Protocols

Registration

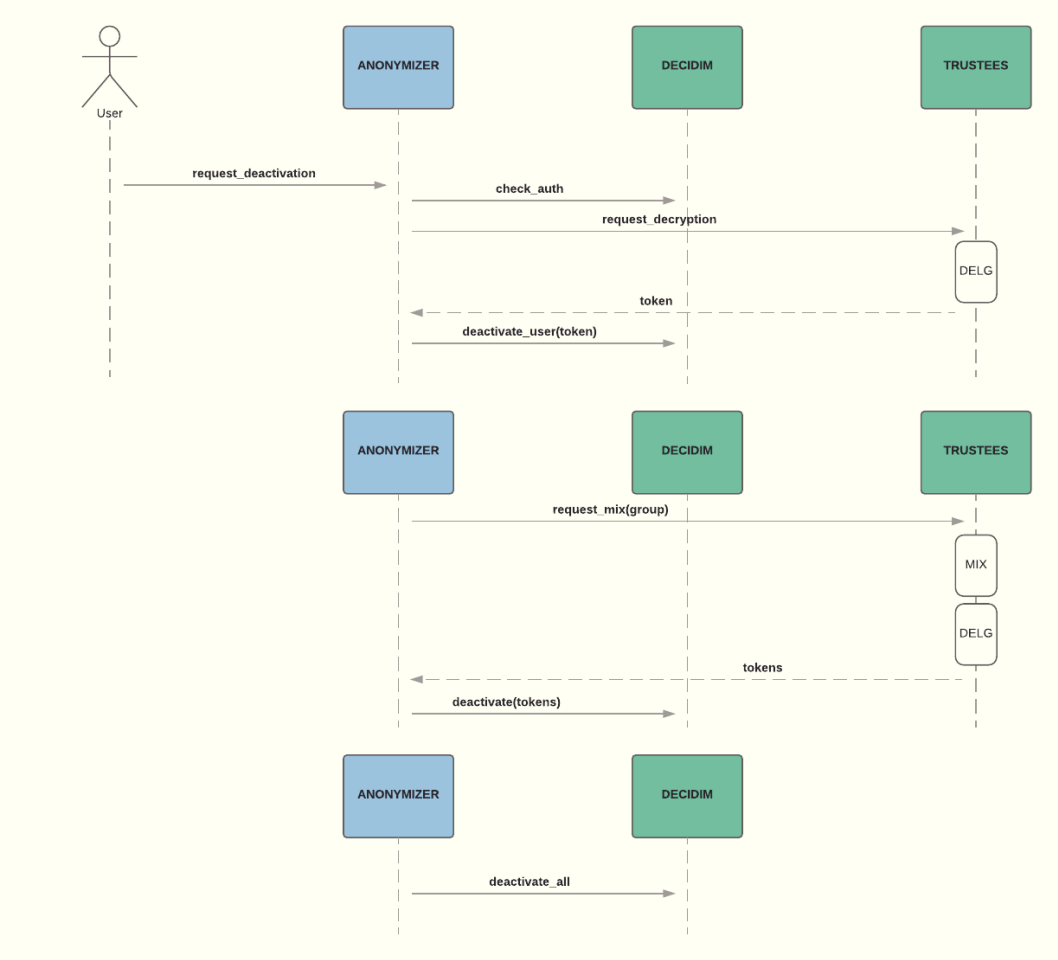


Registration

1. The trustees jointly generate a token encryption public key
2. The user logs in with existing credentials (password)
3. The user is checked for an existing encrypted, if so the process terminates
4. The user's browser generates a random number which is concatenated with user id and then hashed producing a token
5. The user encrypts the token with the public key
6. The encrypted token is stored at the anonymizer
7. When the registration periods ends, the anonymizer requests a mix from the trustees[27]
8. The trustees mix and jointly decrypt the tokens
9. A user is created for each decrypted token
10. The user completes registry by anonymously submitting their token

27 This process can be carried out in groups to match privilege or anonymity groups. Unlike in blind signature schemes, these groups can be determined dynamically. See 8.3.

Deactivation



Individual deactivation

1. User requests deactivation
2. User request is validated (eg email or physical id)
3. Trustees jointly decrypt token stored during registration
4. User is deactivated

Group deactivation

1. Anonymizer requests mix for encrypted tokens corresponding to group
2. Trustees mix and jointly decrypt tokens
3. Anonymizer requests deactivation of users corresponding to decrypted tokens

   This process can be repeated to allow for staged re-anonymization[28].

Global deactivation

1. Anonymizer requests deactivation of all users

28 See section 8.3

Use case mappings

Registration	Registration
Recovery	Deactivation
Re-anonymization	Deactivation + Registration
Privilege modification	Deactivation + Registration

7.3.8 M1T

This scheme is a threshold version of M1 with Distributed ElGamal replaced by Threshold ElGamal.

Protocols

The protocols are the same as M1 with Distributed ElGamal replaced by Threshold ElGamal.

Use case mappings

Registration	Registration
Recovery	Deactivation
Re-anonymization	Deactivation + Registration
Privilege modification	Deactivation + Registration

7.3.9 A note on timing attacks

Although not specified in protocol descriptions it is necessary to address possible attacks stemming from time correlations between anonymization and registration actions. A simple solution is to establish separate anonymization and registration periods that do not overlap, with a reasonably large time interval (eg 1 day) between the two.

7.4 Building-block scheme matrix[29]

	SELG	BLS	PBLS	DELG	TELG	MIX	ANC
B1		X					X
B2	X	X		X			X
B2T	X	X			X		X
P1			X				X
P2	X		X	X			X
P2T	X		X		X		X
M1	X			X		X	X
M1T	X				X	X	X

29 We do not include RNG or HASH in the matrix

8 Evaluation

In this section we evaluate the different schemes according to the stated objectives and additional criteria defined below.

8.1 Objectives 1,2,5

The schemes presented in section 7 were constructed in order to conform to the general scheme of section 5. The conformance of these schemes is founded upon the properties of the cryptographic building blocks defined in section 6, together with additional auxiliary assumptions. In turn, said building blocks employ standard cryptographic assumptions. Please refer to appendix A for details regarding these dependencies and their relations.

Given the root assumptions, and as the schemes presented in section 7 therefore conform to the properties of the general scheme defined in section 5 we say, without formal proof, that those schemes satisfy objectives O1, O2 and O5.

8.2 Objectives 3,4

From section 7 it can be seen that the anonymization schemes have two main contact points with the Decidim system. First, to authenticate existing users against Decidim's records. Second, to create new anonymized blank users linked to secret tokens. This means that changes to the Decidim software are minimal and restricted to the two contact points and any required implementations are simple and easy to isolate. In this way we can say that O4, transparency of anonymization with respect to the Decidim software is satisfied.

Once users have been anonymized the operations within Decidim are unchanged. An anonymized user can operate according to the same features and behaviours as existing users. There is one significant caveat, however. Although they can carry out the same operations within the platform, users must connect to Decidim through an anonymous channel (6.10) in order to preserve the anonymity properties of the proposed schemes. Due to this restriction we cannot claim that objective O3 is entirely satisified, as some usability penalties will be incurred. Hence, objective O3 is partially satisfied.

8.3 Additional criteria

As seen above, the anonymization schemes satisfy the stated objectives reasonably well, and do so in a uniform manner across each one. This section presents additional evaluation criteria that reflect the ways in which the schemes are different, allowing distinctions to be made and pros and cons to be considered.

8.3.1 Group deactivation

Group deactivation refers to the possibility of deactivating users in groups to preserve anonymity without having to resort to global deactivation. Deactivation is a necessary component of several use cases (7.2). For example, in staged re-anonymization, users are deactivated in batches to avoid disabling the platform for everyone at once, and requires therefore group deactivation capability.

» No

Group deactivation is not possible.

» Yes - Static

Group deactivation is possible. Groups must be predefined prior to registration time, so system operators must choose these groups without knowing what subset of users will participate in anonymization. Thus, the groups may not be optimal in the sense of equal sized anonymity sets.

» Yes - Dynamic

Group deactivation is possible. Groups can be defined on the fly at deactivation time. This allows the maximum flexibility for deactivation, and allows optimally sized anonymity sets.

8.3.2 Reversible anonymity

Determines whether or not, under special circumstances, anonymity can be revoked. Revocation of anonymity may be required for the recovery use case (7.2.2). On the other hand, the possibility of revocation requires citizens to place a certain degree (distributed thanks to the distributed ElGamal building blocks) of trust on the system operators. Three different sub-capabilities are offered by the schemes in order to accommodate these types of considerations.

» No

Anonymity cannot be revoked.

» Optional

Revocation is optional in the scheme. Both system designers and users have a choice in whether they want to enable revocation. System designers may altogether leave out the feature when implementing the scheme. User's on the other hand are technically capable of modifying the client to send garbage as the encrypted token. Note also that these schemes could allow users to send other citizens' tokens as deactivation instead of their own. This attack is considered redundant as access to another citizen's token has much more severe consequences than a "deactivation attack", the user would be effectively stolen.

» Yes

Anonymity can be revoked. The user cannot opt out, as the mechanism cannot be separated from registration.

8.3.3 Variable privileges

With variable privileges users may have different authorization profiles to perform different restricted actions. For example, a set of users could be authorized to participate on issues that are relevant to them under a certain geographical criterion. Variable privileges are intrinsically opposed to anonymity as they reduce the anonymity set and therefore make users theoretically more identifiable. This capability should be used with care. Furthermore, support for variable privileges does imply not that full featured capability systems like ACL's are possible, but rather that simple categorizations (as above) can be made to work.

» No

Users all have the same universal privilege profile.

» Yes – Static

Variable privileges are supported, but privilege modification is not possible.

» Yes - Dynamic

Variable privileges are supported, privilege modification is theoretically possible. We use the word "theoretically" because the modification of privileges requires more than one user with the same

privilege to perform the deactivation/registration in order to preserve anonymity. The scheme supports deactivation/registration of these users selectively in this case.

8.3.4 Fault tolerant trustees

Schemes that employ distributed encryption either for anonymization or deactivation require the participation of trustees, these are individuals or institutions which are custodians of privacy, to operate. Several trustees are required to distribute trust. Depending on whether the scheme employs a threshold system or not it is tolerant to a failure of a subset of these trustees without affecting the outcome of the process.

» No

Fault tolerance is not supported, all trustees are necessary for anonymization or deactivation.

» Yes

Fault tolerance is supported, a threshold of trustees is sufficient for anonymization or deactivation.

» N/A Not Applicable

The scheme does not use trustees.

8.3.5 Implementation difficulty

The implementation difficulty of a scheme measures the qualifications and effort necessary to implement said scheme by software engineers. This is a relative measure used to give an indication of how hard implementations of each scheme will be compared to each other.

» 1 - 5 (relative measures)

All schemes require a minimum of qualifications; an engineer with knowledge of cryptography and experience in implementing cryptographic protocols will be needed to carry out or supervise software development.

8.3.6 Scheme evaluation matrix

Below we evaluate the schemes with respect to the additional criteria defined above. See 8.1 and 8.2 for compliance with the objectives O1-O5.

	Group deactivation	Reversible anonymity	Variable privileges	Threshold trustees	Implementation difficulty
B1	No	No	No	N/A	1
B2	No	Optional	No	No	2
B2T	No	Optional	No	Yes	3
P1	Yes - Static	No	Yes - Static	N/A	2
P2	Yes - Static	Optional	Yes - Static	No	3
P2T	Yes - Static	Optional	Yes - Static	Yes	4
M1	Yes - Dynamic	Yes	Yes - Dynamic	No	4
M1T	Yes - Dynamic	Yes	Yes - Dynamic	Yes	5

9 References

Abe, Masayuki & Eiichiro Fujisaki (1996). How to Date Blind Signatures. In *Lecture Notes in Computer Science*, 244–51.

Abe, Masayuki & Tatsuaki Okamoto (2000). Provably Secure Partially Blind Signatures. In *Lecture Notes in Computer Science*, 271–86.

A Porta Aberta, O Portal de Procesos Participativos Do Concello Da Coruña. 2016. *A Porta Aberta*. Accessed November 19. https://aportaaberta.coruna.es/.

Baldimtsi, Foteini & Lysyanskaya (2013). Anonymous Credentials Light. In Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security - CCS '13. doi:10.1145/2508859.2516687.

Barker, R. (1990). *CASE Method: Entity Relationship Modelling*. Addison Wesley Longman.

Bellare, Mihir & Rogaway. 1993. Random Oracles Are Practical. In *Proceedings of the 1st ACM Conference on Computer and Communications Security* - CCS '93. doi:10.1145/168588.168596.

Bellare, M., C. Namprempre, D. Pointcheval & Semanko, M. (2003). The One-More-RSA-Inversion Problems and the Security of Chaum's Blind Signature Scheme. *Journal of Cryptology. The Journal of the International Association for Cryptologic Research* 16 (3): 185–215.

Bernhard, David, Olivier Pereira & Bogdan Warinschi (2012). How Not to Prove Yourself: Pitfalls of the Fiat-Shamir Heuristic and Applications to Helios. In *Lecture Notes in Computer Science*, 626–43.

Boneh, D. (1998). The Decision Diffie-Hellman Problem. In *Lecture Notes in Computer Science*, 48–63.

Camenisch, Jan & Anna Lysyanskaya (2001). An Efficient System for Non-Transferable Anonymous Credentials with Optional Anonymity Revocation. In *Lecture Notes in Computer Science*, 93–118.

Carroll, J. M. & Rosson, M. B. (2003). A Trajectory for Community Networks Special Issue: ICTs and Community Networking." The Information Society 19 (5): 381–93.

Casapulla, Giovanni, Fiorella De Cindio, Oliverio Gentile & Leonardo Sonnante (1998). A Citizen-Driven Civic Network as Stimulating Context for Designing On-Line Public Services." PDC, January, 64–74.

Chase, Melissa, Sarah Meiklejohn & Greg Zaverucha. 2014. Algebraic MACs and Keyed-Verification Anonymous Credentials. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security* - CCS '14. doi:10.1145/2660267.2660328.

Chaum, D. (2004). Secret-Ballot Receipts:True Voter-Verifiable Elections. IEEE Security&Privacy Magazine 2(1): 38–47.

Chaum, D. (1983). Blind Signatures for Untraceable Payments. In *Advances in Cryptology*, 199–203.

Chaum, D. (1981). Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. Communications of the ACM 24 (2): 84–90.

Cho, Daegon & Alessandro Acquisti (2013). The More Social Cues, The Less Trolling? An Empirical Study of Online Commenting Behavior. http://repository.cmu.edu/cgi/viewcontent.cgi?article=1338&context=heinzworks.

Chow, Sherman S. M., Victor K. Wei, Joseph K. Liu & Tsz Hon Yuen (2006). Ring Signatures without Random Oracles. In *Proceedings of the 2006 ACM Symposium on Information, Computer and Communications Security* - ASIACCS '06. doi:10.1145/1128817.1128861.

Clifton, Chris & Don Marks. 2016. Security and Privacy Implications of Data Mining. Accessed November 24. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.28.891&rep=rep1&type=pdf.

Connolly, Terry, Leonard M. Jessup & Joseph S. Valacich (1990). Effects of Anonymity and Evaluative Tone on Idea Generation in Computer-Mediated Groups. *Management Science 36* (6): 689–703.

Consul. 2016. Consul/consul. GitHub. Accessed November 19. https://github.com/consul/consul.

Cortier, Véronique, David Galindo, Stéphane Glondu & Malika Izabachène (2013). Distributed ElGamal à La Pedersen. In Proceedings of the 12th ACM Workshop on Workshop on Privacy in the Electronic Society - WPES '13. doi:10.1145/2517840.2517852.

Cramer, Ronald, Rosario Gennaro & Berry Schoenmakers (1997). A Secure and Optimally Efficient Multi-Authority Election Scheme. In *Lecture Notes in Computer Science*, 103–18.

Danezis, George & Claudia Diaz (2008). A Survey of Anonymous Communication Channels. https://www.esat.kuleuven.be/cosic/publications/article-927.pdf.

Data Recovery - Wikipedia. 2016. Accessed December 3. https://en.wikipedia.org/wiki/Data_recovery.

Davies & Todd (2009). *Online Deliberation: Design, Research, and Practice*. Stanford Univ Center for the Study.

Decide Madrid. 2016. Accessed November 19. https://decide.madrid.es/.

Decidim.barcelona. 2016. Accessed November 19. https://decidim.barcelona/.

De Cindio, F. (2012). Guidelines for Designing Deliberative Digital Habitats: Learning from E-Participation for Open Data Initiatives. *The Journal of Community Informatics* 8 (2). http://www.ci-journal.net/index.php/ciej/article/view/918.

Diakopoulos, Nicholas & Mor Naaman (2011). Towards Quality Discourse in Online News Comments. In *Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work* - CSCW '11. doi:10.1145/1958824.1958844.

Druescas (2016). Reddit-Style Filtering for E-Democracy | Agora Voting Blog. Accessed November 20. https://blog.agoravoting.org/index.php/2016/02/15/reddit-style-filtering-for-e-democracy/.

Dwork, C. (2006). Differential Privacy. In *Lecture Notes in Computer Science*, 1–12.

Edman, Matthew & Bülent Yener (2009). On Anonymity in an Electronic Society: A Survey of Anonymous Communication Systems. http://www.cs.ucf.edu/~dcm/Teaching/COT4810-spring2011/Literature/AnonimityCommunication.pdf.

ElGamal, Taher. (n.d.) A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. In *Lecture Notes in Computer Science*, 10–18.

E-Participation – Wikipedia. 2016. Accessed November 21. https://en.wikipedia.org/wiki/E-participation#Models_and_tools_for_e-participation.

Flanagin, A. J., V. Tiyaamornwong, J. O'Connor & D. R. Seibold (2002). Computer-Mediated Group Work: The Interaction of Sex and Anonymity. *Communication Research* 29 (1): 66–93.

Fredheim, Rolf, Alfred Moore & John Naughton (n.d.) Anonymity and Online Commenting: An Empirical Study. SSRN Electronic Journal. doi:10.2139/ssrn.2591299.

Function (mathematics) - Wikipedia. 2016. Accessed December 5. https://en.wikipedia.org/wiki/Function_(mathematics)#Injective_and_surjective_functions.

Ganta, Srivatsava Ranjit, Shiva Prasad Kasiviswanathan & Adam Smith. 2008. Composition Attacks and Auxiliary Information in Data Privacy. In *Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* - KDD 08. doi:10.1145/1401890.1401926.

Gennaro, Rosario, Stanisław Jarecki, Hugo Krawczyk & Tal Rabin. 1999. Secure Distributed Key Generation for Discrete-Log Based Cryptosystems. In *Lecture Notes in Computer Science*, 295–310.

Glondu, Stéphane. 2016. Belenios Specification. Accessed December 7. http://belenios.gforge.inria.fr/specification.pdf.

Goldwasser, Shafi, Silvio Micali & Ronald L. Rivest. 1988. A Digital Signature Scheme Secure Against Adaptive Chosen-Message Attacks. *SIAM Journal on Computing* 17 (2): 281–308.

Haeberlen, Andreas, Benjamin C. Pierce & Arjun Narayan (2011). Differential Privacy Under Fire.

https://www.usenix.org/legacy/event/sec11/tech/full_papers/Haeberlen.pdf.

Haenni, R . (2016). UniVote System Specification. Accessed December 7.
https://www.univote.ch/documentation/univote/1.9/doc/specification.pdf.

Hash Function Security Summary - Wikipedia. 2016. Accessed December 6.
https://en.wikipedia.org/wiki/Hash_function_security_summary.

Home - Oviedoparticipa.es. 2016. Accessed November 19.
http://www.oviedoparticipa.es/.

Hopkins, N. (2013). Surveillance, Democracy, Transparency – a Global View. The Guardian. October 10.
http://www.theguardian.com/world/2013/oct/10/surveillance-global-view-debate.

Identity and Anonymity. (2016). Accessed December 20.
http://web.mit.edu/gtmarx/www/identity.html.

Jonker, Hugo, Sjouke Mauw & Jun Pang. (2013). Privacy and Verifiability in Voting Systems: Methods, Developments and Trends." Computer Science Review 10: 1–30.

Klenk, Nicole L. & Gordon M. Hickey. (2011). A Virtual and Anonymous, Deliberative and Analytic Participation Process for Planning and Evaluation: The Concept Mapping Policy Delphi. *International Journal of Forecasting* 27 (1): 152–65.

Krumm, J. (n.d.). Inference Attacks on Location Tracks. In *Lecture Notes in Computer Science*, 127–43.

Lerman, K. (2007). Social Information Processing in News Aggregation. IEEE Internet Computing 11 (6): 16–28.

Linders, D. (2012). From E-Government to We-Government: Defining a Typology for Citizen Coproduction in the Age of Social Media. *Government Information Quarterly* 29 (4): 446–54.

Menezes, Alfred, Paul van Oorschot & Scott Vanstone (1996). Handbook of Applied Cryptography.

O'Hara, K. (2012). Transparency, Open Data and Trust in Government. In *Proceedings of the 3rd Annual ACM Web Science Conference on - WebSci '12*. doi:10.1145/2380718.2380747.

O'Hara, K. (2016). Transparent Government, Not Transparent Citizens: A Report on Privacy and Transparency for the Cabinet Office. Accessed November 20.
https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/61280/transparency-and-privacy-review-annex-b.pdf.

One-Way Function - Wikipedia. (2016). Accessed December 5.
https://en.wikipedia.org/wiki/One-way_function#Theoretical_definition.

Openness and Transparency - Pillars for Democracy, Trust and Progress-OCDE. (2016). Accessed November 20.
http://www.oecd.org/fr/etatsunis/opennessandtransparency-pillarsfordemocracytrustandprogress.htm.

Pointcheval, David & Jacques Stern. (1996). Provably Secure Blind Signature Schemes. In *Lecture Notes in Computer Science*, 252–65.

Pure Function - Wikipedia. 2016. Accessed December 4.
https://en.wikipedia.org/wiki/Pure_function.

Reddit: The Front Page of the Internet. (2016). Accessed November 20.
https://www.reddit.com/.

Rivest, Ron. 2004. Lecture 18: Mix-Net Voting Systems.
http://courses.csail.mit.edu/6.897/spring04/L18.pdf.

Sako, Kazue & Joe Kilian (1995). Receipt-Free Mix-Type Voting Scheme. In *Lecture Notes in Computer Science*, 393–403.

Sampigethaya, Krishna & Radha Poovendran (2006). A Framework and Taxonomy for Comparison of Electronic Voting Schemes. *Computers & Security* 25 (2): 137–53.

Schnorr, Claus Peter & Markus Jakobsson (2000). Security of Signed ElGamal Encryption. In *Lecture Notes in Computer Science*, 73–89.

SHA-2 - Wikipedia. 2016. Accessed December 6.
https://en.wikipedia.org/wiki/SHA-2.

Stadler, Markus, Jean-Marc Piveteau & Jan Camenisch (1995). Fair Blind Signatures. In *Lecture Notes in Computer Science*, 209–19.

Terelius, Björn & Douglas Wikström (2010). Proofs of Restricted Shuffles. In *Lecture Notes in Computer Science*, 100–113.

Tsiounis, Y. & Moti Yung (1998). On the Security of ElGamal Based Encryption. In *Lecture Notes in Computer Science*, 117–34.

UN (2014). Universal Declaration of Human Rights. In *The Core International Human Rights Treaties*, 3–10.

UniVote (2016). Accessed December 7.
https://www.univote.ch/voting-client/.

Use-Case Analysis - Wikipedia. (2016). Accessed December 9.
https://en.wikipedia.org/wiki/Use-case_analysis#Realization.

Wikstrom, D. (2016). Verificatum. Accessed December 7.
http://www.verificatum.com/.

Yang, Qiang & Xindong Wu (2006). 10 CHALLENGING PROBLEMS IN DATA MINING RESEARCH. International Journal of Information Technology & Decision Making 05 (04): 597–604.

APPENDIX

A Security properties

A.1 Standard cryptographic assumptions

Building block	Property	Assumption
RSA Blind signature	Blindness, Unforgeability	ROM[30]
Partially blind signature	Blindness, Unforgeability	ROM
Signed ElGamal	Semantic security Correct decryption	DDH[31] ROM
Distributed ElGamal	Semantic security Correct decryption	DDH ROM
Distributed threshold ElGamal	Semantic security Correct decryption	DDH ROM
ElGamal re-encryption mixnet	Correct shuffle	ROM

A.2 Auxiliary assumptions

A.2.1 CLI: The client device used to participate in anonymization protocols is not under control of an adversary.

A.2.2 TRU: At least one of the trustees is honest.

A.2.3 AUT: Authentication is properly implemented for all protocol executions (only users with valid credentials are allowed to participate).

A.2.4 RNG: The probability of guessing an output of a CSPRBG[32] execution for any of the users participating in anonymization protocols is negligible.

A.2.5 H: The probability of a hash[33] collision between any of the users participating in anonymization protocols is negligible.

A.3 General scheme dependencies

Section 5 defined a general scheme with several properties

1) The token f(auth) is difficult to trace back to the original token

30 (Mihir Bellare & Rogaway, 1993)

31 (Boneh, 1998)

32 See 6.1

33 See 6.2

2) The token f(auth) is known only by the citizen who knows the old authentication token, and no other citizen
3) The Anon function is injective

These properties have the following dependencies (across all protocols)

1	CLI, Blindness, Semantic security, TRU
2	CLI, AUT, RNG, HASH, Semantic security, TRU
3	Unforgeability, AUT (only one encrypted token is allowed per user)

A.4 Objective-assumption matrix

The objectives O1 and O2 have the following dependencies (across all protocols)

	B1	B2	B2T	P1	P2	P2T	M1	M1T
O1	CLI ROM	CLI ROM DDH TRU	CLI ROM DDH TRU	CLI ROM	CLI ROM DDH TRU	CLI ROM DDH TRU	CLI DDH TRU	CLI DDH TRU
O2	CLI AUT RNG H ROM	CLI AUT RNG H ROM TRU	CLI AUT RNG H ROM TRU	CLI AUT RNG H ROM	CLI AUT RNG H ROM TRU	CLI AUT RNG H ROM TRU	CLI AUT RNG H ROM TRU	CLI AUT RNG H ROM TRU