Assurance of anonymity for respondents in sensitive online surveys

Halvor Bjørn

Thesis submitted for the degree of Master in Informatics: Programming and Networks 60 credits

Department of Informatics
Faculty of mathematics and natural sciences

UNIVERSITY OF OSLO

Autumn 2017
Assurance of anonymity for respondents in sensitive online surveys

Halvor Bjørn
Abstract

Online survey applications enable researchers to create and distribute surveys in a cost-effective way. However, respondents may feel reluctant to participate in surveys collecting sensitive personal information if they believe that they can be re-identified and related to the collected data set. Giving respondents a strong assurance of anonymity may improve the response rate and quality of data collected. In some instances, it may be beneficial for the researcher or survey administrator to place restrictions on the demographic group or domain where respondents are invited to participate. We approach assurance of anonymity in two contexts. One where the survey is open for everyone and one where the participants are invited or restricted to a specific group.

Conducted interviews have shown a tendency for respondents to feel reluctant to post a submission, or not answer truthfully on sensitive topics if their identity is known. To increase the response rate and data quality in online surveys, we have proposed a solution providing strong assurance of anonymity in open surveys. The proposed solution utilizes client-side encryption, randomization and caching of submissions to decrease the probability of re-identification, at the cost of potentially reduced reliability in the data collection process. Further, we have developed a framework for determining the assurance of anonymity provided by a particular solution, which we use to evaluate the proposed solutions for anonymous authentication of respondents. Our evaluation of the proposed solutions enabling anonymous authentication have shown it is possible to allow respondents to post a submission while given a strong assurance of anonymity. However, the strong assurance of anonymity comes at the cost of requirements for one or several additional elements such as a pre-registration process by the respondent, management of cryptographic keys or secrets by the respondent, or trust in a third-party.

Our results show that a system providing strong assurance of anonymity to respondents is possible to implement, and it may increase the response rate and data quality of sensitive online surveys, with the only disadvantage that it could potentially reduce the reliability of the data collection process. However, the latter is not seen as a practical problem. Truly anonymous authentication of respondents is possible, but at the added cost of increased overhead, decreased usability or by requiring trust in third parties by the respondent.

Future research is recommended to assess the value of a high assurance of anonymity related to response rates and data quality, compared to the trade-off with reliability, overhead, usability and trust identified above.
Acknowledgements

I would like to thank my supervisor, Prof. Audun Jøsang of the Department of Informatics at the University of Oslo. Prof. Jøsang have always been encouraging, supportive and patient with my work. Thank you.

I would also like to express my gratitude to my team members and fellow employees at USIT for their valuable input, feedback and interesting discussions. Without them, the writing of this thesis would not have been possible.

Last but not least, I would like to thank my family, partner and friends for their encouragement and keen interest in my study and work over the recent years. Thank you.

Author
Halvor Bjørn
2.5 Authentication Assurance Levels

2.5.1 Authentication assurance levels in ID-porten

2.6 Technical Anonymity Assurance Levels

2.6.1 Scope of the framework

2.6.2 Requirements for assurance level 1

2.6.3 Requirements for assurance level 2

2.6.4 Requirements for assurance level 3

2.6.5 Requirements for assurance level 4

2.7 Cryptographic primitives

2.7.1 Cryptographic hash functions

2.7.2 Zero Knowledge Proof

2.7.3 General description of ZKP

2.7.4 Requirements for ZKP

2.7.5 Zero Knowledge Authentication

2.7.6 Blind Signature Protocol

2.8 Existing technology for anonymous surveys and Identity Management

2.8.1 Anonize

2.8.2 Attribute-based credentials

3 Nettskjema as a use case

3.1 About Nettskjema

3.2 Risk analysis of Nettskjema

3.3 About TSD

3.4 Respondents

3.4.1 Signed in

3.4.2 Invited

3.4.3 Open surveys

3.4.4 Authentication Assurance Levels in Nettskjema

3.5 Question types

3.5.1 Surveys that contain personal identifiable information

4 Anonymous responses in Nettskjema

4.0.1 Current situation

4.0.2 Benefits

4.1 Requirements for a solution

4.1.1 Removal of identifying attributes

4.1.2 Identifying question fields

4.1.3 Attacks on solutions

4.2 External proxy

4.2.1 Integration with Nettskjema

4.2.2 Benefits and disadvantages

4.3 Anonymous frontend

4.3.1 Nettskjema API
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>A capture of the sample survey on alcohol consumption</td>
<td>16</td>
</tr>
<tr>
<td>1.2</td>
<td>A capture of the sample survey on drug use</td>
<td>17</td>
</tr>
<tr>
<td>2.1</td>
<td>The concept of identity</td>
<td>25</td>
</tr>
<tr>
<td>2.2</td>
<td>The secret cave</td>
<td>33</td>
</tr>
<tr>
<td>4.1</td>
<td>Banner telling respondents that no personal identifiable information is recorded</td>
<td>45</td>
</tr>
<tr>
<td>4.2</td>
<td>System architecture of anonymous front-end</td>
<td>51</td>
</tr>
<tr>
<td>5.1</td>
<td>Registration and submission phase with IBM IdentityMixer</td>
<td>63</td>
</tr>
<tr>
<td>5.2</td>
<td>Registration and submission phase with blind signatures</td>
<td>68</td>
</tr>
</tbody>
</table>
List of Tables

2.1 Level of assurance ............................................ 26
2.2 Level of anonymity assurance ............................... 27
3.1 Potential impact of authentication failure ........... 42
3.2 Mapping between impact of authentication failure and Authentication Assurance Levels ............................... 42
5.1 Properties of the identifying attributes .............. 66
5.2 A comparison between the various authentication mechanisms .................................................. 69
5.3 Assurance levels of anonymity provided ............ 70
6.1 Explanation of variables ...................................... 73
Acronyms

AAL  Authentication Assurance Level. 25  26  41
API  Application Programming Interface. 40
CSP  Content Security Policy. 55
DRM  Digital rights management. 77
FAQ  Frequently Asked Questions. 44
GDPR  General Data Protection Regulation. 21  22  47
HIV  human immunodeficiency virus. 11
IdP  Identity provider. 40  67
MAC  Message Authentication Code. 31
MD5  Message-Digest algorithm 5. 32
NSD  Norwegian Centre for Research Data. 10  44
PGP  Pretty Good Privacy. 40
PrEP  Pre-Exposure Prophylaxis. 11
PRNG  Pseudo-Random Number Generator. 31
RA  registration authority. 60  62  70
REST  Representational state transfer. 40  50  51
SA  survey authorities. 60  62  70
SAML  Security Assertion Markup Language. 40
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGX</td>
<td>Software Guard Extensions.</td>
<td>77</td>
</tr>
<tr>
<td>SHA-1</td>
<td>Secure Hash Algorithm 1.</td>
<td>32</td>
</tr>
<tr>
<td>SSO</td>
<td>Single Sign-On.</td>
<td>40</td>
</tr>
<tr>
<td>TAAL</td>
<td>Technical Anonymity Assurance Level.</td>
<td>27, 28, 54, 64, 65, 70, 72</td>
</tr>
<tr>
<td>TOR</td>
<td>The Onion Router.</td>
<td>14, 54, 61, 62</td>
</tr>
<tr>
<td>TSD</td>
<td>Services for sensitive data.</td>
<td>3, 38, 40</td>
</tr>
<tr>
<td>UiO</td>
<td>University of Oslo.</td>
<td>38, 40, 43, 45, 47, 50, 52, 78</td>
</tr>
<tr>
<td>VPN</td>
<td>Virtual Private Network.</td>
<td>14, 54, 55</td>
</tr>
<tr>
<td>WAPP</td>
<td>Web Application Development Group.</td>
<td>38, 44, 50, 52</td>
</tr>
<tr>
<td>ZKP</td>
<td>Zero Knowledge Proof.</td>
<td>3, 32, 34</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Problem statement and motivation

Online survey applications enable researchers to easily create surveys and distribute them in a cost-effective and easy way to respondents. However, online surveys allow for easier tracking of respondents both by the researcher and third-parties.

While tracking of respondents may be beneficial in many cases, it is the author’s hypothesis that some respondents may alter their answers or feel reluctant to submit sensitive data through an online survey application if their identity is known. This data may include personal sensitive information, such as information about their physical and mental health.

This thesis seeks to answer three questions. Does the assurance of being anonymous have an impact on response rates and the data quality of surveys collecting sensitive information? Can a desirable level of assurance for anonymity be achieved in a generic survey application for unauthenticated respondents? And can a desirable level of assurance for anonymity be achieved in a generic survey application for authenticated users?

Being authenticated in this context can be reduced to a verification process, where it is only desirable to verify that the respondent belongs to a group that is allowed to answer, as it is not a requirement to know the actual identity of the respondent.

In addition to the questions asked above, it is the author’s hypothesis that anonymous survey application, e.g. applications that do not collect or process personal identifiable information, may enable researchers and survey administrators to easier collect and process data from respondents. As discussed in section 2.2 on page 22, anonymous data sets are not considered to be personal information. As such, an approval from NSD, REK or the Norwegian Data Protection Authority is not needed before data collection can start.
1.2 Structured overview

Chapter 1 and 2 introduce the background for the project, an overview of the research done, and an introduction to the theoretical aspects of the thesis.

Chapter 3 introduces the survey application Nettskjema as a use-case for anonymous surveys. The chapter gives an introduction to what Nettskjema is, how it works, and why it was chosen as a use-case.

Chapter 4 discusses requirements for an anonymous solution for open surveys, and two different approaches to enabling respondents to post their submissions, while given a strong guarantee that there will be no connection between their submission and their identity. This chapter will also lay the foundation for some of the requirements needed to implement a solution for anonymous authenticated surveys.

Requirements and possible solutions for an implementation of anonymous authenticated surveys will be discussed in chapter 5, followed by a discussion of the proposed solutions in chapter 6.

1.3 Methodology

1.3.1 Interviews with respondents

To better understand what respondents think about anonymity, and how being anonymous may affect their behaviour when answering online surveys, interviews with a total of 9 respondents were conducted in the period between Friday 7th of April and Monday 10th of April 2017.

The distribution between respondents was equally distributed between respondents identifying themselves as either male or female. The respondents were selected because of their varying interests, job-titles and expertise.

During the interviews, respondents were not asked questions that could directly identify them, and information that could possibly identify a respondent were either left out or rewritten during transcription of the interviews to protect the respondent’s privacy. Permission to conduct the interviews were granted by the Norwegian Centre for Research Data (NSD) ahead of the start of the interview period.

During the interviews, respondents were asked about their thoughts on what anonymity means for them, and if they feel being that anonymous is important. It is assumed that a better understanding of what respondents associate with being anonymous can help application developers and survey administrators presenting anonymous surveys in a better way.

In addition to questions on anonymity, respondents were asked about their thoughts about what kind of metadata they thought could be collected together with their submission, and questions related to their general trust and confidence in how researchers, survey administrators and application developers handle their data.
Metadata in this context, is any additional information that can be included in a submission, which is not directly asked as questions in a survey. This can include the time of delivery, information about the respondents operating system, device or browser, directly identifiable information such as name or e-mail if already known to the survey administrators or indirectly identifiable information such as the respondents ip-address.

Knowledge on respondent’s relation to metadata, and their trust in the IT-systems and people that are responsible for handling their data can be valuable in the development of online survey applications, and for researchers conducting a survey.

The respondents were also presented with two samples of surveys, covering three topics generally considered to be of a sensitive nature. The respondents were not asked what they would answer in those surveys. However, they were asked about their thoughts on the questions asked, and if or how truthfully they would answer, depending on the guarantees of anonymity they were given.

The first survey is a survey conducted by Ragnhild Bø, a PhD candidate at the Department of Psychology, on alcohol consumption.

The second survey is a sample of a survey conducted at the Oslo University Hospital. The survey is a part of an evaluation programme related to the introduction of [Pre-Exposure Prophylaxis (PrEP) medication in Norway, an antiviral drug used to prevent [Human Immunodeficiency Virus (HIV)] infection. The sample questions cover topics on drug use and mental health related to sexual activities.

The surveys were used in the interviews because they cover three topics where it is believed that some of the respondents might consider them sensitive. Questions related to health and criminal records are also information that is considered to be sensitive personal information according to the Norwegian Privacy law (Lov om behandling av personopplysninger) §2-8[1], if a person can be identified from the dataset.

As the topics are sensitive, the initial assumption is that some respondents might not answer truthfully, or possibly not answer the survey at all if the information they provide can be connected to their identity.

The interviews conducted are qualititative by nature and the sample of respondents is relatively small. As such, the findings should not be seen as results stating how and what respondents in general think about anonymity and sensitive data in online surveys. However, the findings can identify potential problems with existing online survey applications, and identify trends among respondents on how they feel about entering sensitive information in online surveys.
1.3.2 Interviews with Dagfinn Bergsager

Dagfinn Bergsager has assisted with questions related to privacy mechanisms in the use-case application Nettskjema. Bergsager is a participant in a project to draft new guidelines for privacy by design, directed by the Norwegian Data Protection Authority. He also advises on questions related to privacy and security for data collection with Nettskjema for research projects at the University of Oslo, Oslo University Hospital, Norwegian Institute of Public Health, Oslo and Akershus University College of Applied Sciences and the University of Tromsø.

1.3.3 Existing technology

In recent years, there has been an increased focus on privacy enhancing technologies. Microsoft U-Prove[2] and IBM Identity Mixer[3], both evaluated and tested in the European Union funded research project ABC4Trust[4], aim to provide privacy enhancing solutions for credentials and identity management.

Anonize[5], a platform for creating ad-hoc online surveys, aim to provide anonymous authentication of invited respondents.

Microsoft U-Prove, IBM Identity Mixer and Anonize will be discussed in detail below.

1.3.4 Applications and source code

In-depth knowledge of the survey application Nettskjema, used in the use-case, has been gathered from reading and working with the source code which is publicly available at [https://bitbucket.usit.uio.no/projects/WAPP/repos/nettskjema](https://bitbucket.usit.uio.no/projects/WAPP/repos/nettskjema). The author is currently employed at the University Center of Information Technology, University of Oslo, which develops Nettskjema.

In addition to the source code, documentation and public information about Nettskjema have been used where applicable. Informal talks with other developers of Nettskjema have also been conducted over the last year related to specific details on how the application works and how it handles data.

1.4 Why Nettskjema as use-case application

The use-case application Nettskjema is developed and maintained at the University Center of Information Technology at the University of Oslo. One of the main priorities of Nettskjema is to provide a secure platform for data gathering in the University sector, which can be seen from the annual plan for strategic direction and priorities at USIT, UiO[6] for 2016-2018.
annual plan lists research and education as the top priority for the given time period.

The source code, documentation and risk analysis of Nettskjema is publicly available, which allows researchers and outsiders to gain a better understanding on how the application works.

Other survey applications, such as SurveyMonkey and Questback were considered. However, Nettskjema was chosen since the application has a primary focus on the University sector, and the source code and documentation are publicly available.

1.5 Interviews

While conducting the interviews, some assumptions about the surveys discussed were made. It was assumed that all the surveys were done in a research context and that the data collected through the surveys only would be used for research, and not for commercial or other purposes. Those assumptions were made to limit the scope of the interviews to data collection in a research context, and to avoid broad discussions on data collection, including but not limited to, commercial use of sensitive personal data.

To further limit the scope of the interviews to focus on the questions in the sampled surveys, only a sample of questions from each survey was included in the interviews. Information about the study, how the researchers would process the data and questions that could either directly or indirectly identify respondents were left out.

The respondents were also told to think of the questions in two different contexts. One where they would answer the surveys, knowing there would be a strong connection with their identity and their submission, and one where they were given strong guarantees that it was no way to connect their identity with their submission.

The assumption of strong guarantees for being anonymous can work fine in theory. However, one respondent said that they are not sure if those guarantees can be given on the Internet today. As such, being guaranteed to be anonymous would probably not affect the way they would answer on a real survey.

1.5.1 Thoughts on anonymity

The respondents were first asked about their general thoughts on anonymity and what being anonymous meant to them. There were significant variations in the responses, ranging from respondents saying that anonymity could be achieved by using a different name to anonymity being not able to tell who you are or that there is no connection to your identity.

Most of the respondents associated anonymity with either privacy-related actions online, or actions related to bullying or other mischievous activities.
Privacy related actions could for instance be chat or disclosing sensitive personal information such as health information in online forums. Bullying and other mischievous activities were mostly related to cyber bullying or leaving negative comments on, for instance, a discussion board.

Some of the respondents said during the interviews that it was sometimes hard for them to distinguish between anonymity and privacy online. This was mostly because anonymity could be used to enhance their privacy, or that anonymity was a result of enhancing their privacy online in the first place.

One example given by some of the respondents was the use of software to block scripts tracking their behaviour, which would hide their usage pattern from advertisement networks and companies specialicing in online tracking. The goal was to enhance their privacy, with anonymity being a side effect as there would be one less connection with their identity to their browsing habits.

Another example mentioned by several of the respondents was the usage of privacy enhancing technologies such as The Onion Router (TOR) and Virtual Private Network (VPN). Reasons for using this kind of software was broad, and varied from wanting to check it out, reading sensitive information online and accessing content that is otherwise blocked.

1.5.2 Assumptions made about the surveys

As the respondents’ definition of anonymity varied considerably, they were told to discuss the surveys in two different context. In the first context, it was assumed that it would be a strong connection between their identity and their response. For instance that they would fill in their name or e-mail adress as a part of the survey. In the second context, it was assumed that there was no way of connecting their identity to their submission. Meaning that no one would be able to re-identify them based on their answers or any other information collected together with the survey. The surveys themselves did not ask questions that could directly or indirectly identify a respondent. Technical details on how the respondents anonymity would be preserved were not discussed during the interviews.

It was also assumed that the surveys were done in a research context, and that the data would not be used for commercial or medical purposes, or be legally obtained by law enforcement to be used an investigation or in a trial. Some of the respondents mentioned those possible usages of their submitted data as reasons to why they would not want to submit an answer, or not answer truthfully to surveys. As such, the assumption was made to make more meaningful comparisions of the data afterwards. However, those reasons for not wanting to answer a survey were noted as well.

To protect the privacy of the respondents, they were never asked what they would answer on a specific question in a survey. In addition, asking
respondents if they felt a specific question was sensitive, could be related to asking the respondent for sensitive information. As such, the respondents were asked on their general thoughts on the surveys.

In general, the respondents said that the sensitivity of many of the questions depended on what they answered. Care was taken to make sure the respondents never had to say what they would have answered. Instead, they were given an assumption on what the answer would be, before discussing the sensitivity. A typical example was questions related to drug use, where the respondents were told that assuming that you have taken any of those drugs, would you answer this question truthfully, or answer it at all?

1.5.3 Survey on alcohol consumption

Questions related to alcohol consumption were generally ranked as the least sensitive among the surveys by the respondents. Most of the respondents said the questions were to some degree sensitive, but that the degree of sensitivity depends a lot on how or what you would answer. Some of the respondents also said the survey asked for more than just your habits on alcohol consumption, meaning that it would be possible to deduce information about physical and mental health problems from the answers.

Most of the respondents said that there was a certain social stigma related to a high consumption of alcohol, or that consumption of alcohol was generally considered taboo to talk about.

In the context of being identified in their submission, most of the respondent said they would answer truthfully. However, some respondents said they might adjust their answers to be more correct or the right answer. Eventually, they would not answer the survey at all. The reasons mentioned for this adjustment was to be seen as a better person by the researcher and that it could be hard to be truthful with yourself about your consumption. One of the reasons for not answering the survey, was that the respondents felt that not answering truthfully would reduce the quality of the study, and it would be better to not answer if you’re not comfortable with the questions asked.

If the respondents were given an assurance that they were anonymous, most said they would complete the survey and answer truthfully. Most respondents said it felt easier to be truthful, both to the researcher and to themselves, if they knew there was no way they could be identified after submitting their response.

1.5.4 Survey on mental health

Questions related to mental health were generally ranked as quite sensitive by the respondents. Some respondents said that they only would discuss those questions with health care workers. However, one respondent said
that while the questions could be quite sensitive, they felt it was generally acceptable to talk about mental health, and that it was easy to be open about the topic. As with the questions related to alcohol consumption, most respondents said that the degree of sensitivity depended on how one would answer.

Several respondents said they would not answer the survey if their identity was known to the researcher, and that they generally needed to trust the researcher before answering. Some respondents pointed out that they would prefer to not answer the entire survey instead of not answering some questions, since leaving questions unanswered could imply that you would have given a response that would have been sensitive. As with the survey on alcohol consumption, some respondents said they might adjust their answers.

In the context of being anonymous, most of the respondents said it would be easier for them to complete the survey. However, several respondents said that the reason for why the survey was conducted was important to them,
and that it would be a factor in deciding if they would want to complete the survey at all.

1.5.5 Survey on drug use

Generally ranked as the most sensitive topic among the respondents. As with the previous surveys, the respondents said that the degree of sensitivity depended on how one would answer. Some of the respondents said that several of the drugs discussed in the survey is illegal, and were concerned about the potential consequences, assuming they had taken those kinds of drugs.

![Figure 1.2: A capture of the sample survey on drug use](image)

Most respondents said, that assuming they had not taken any illegal drugs, it would be acceptable for them to answer the survey. Having their identity connected to their submission was mostly an issue if they had taken illegal drugs. Some respondents were concerned about how this data could be used, and said that it was important for them that the data was handled correctly by a researcher. As with the previous surveys, some respondents
said they would adjust their answers, or not complete the survey. Given a strong assurance of being anonymous, most respondents said it would be easier to complete the survey. Some respondents said it would also be easier to be truthful and honest, both to the survey administrator or the researcher, and to themselves if they were anonymous.

1.5.6 Thoughts on trust

In the context of the interviews, trust refers to trust in the survey administrators, researchers collecting data, developers and maintainers of the survey application and other entities that might have access to the submitted data in a survey.

The degree of trust in entities handling or having access to the respondents data varied a lot between the respondents interviewed. Some of the respondents would describe their trust as blind faith, saying that as long as it is a research project, they trust the researchers and others to handle the gathered data appropriately. However, most of the respondents were a bit more reluctant, saying that trust was a sum of some or all of the following categories:

- Purpose
- Reputation
- Transparency

Several respondents mentions the purpose of the study to have some impact on whether or not they chose to trust the researchers with their data. The purpose could be related to what the researchers wanted to study, how the data would be used, and if answering the survey would have any impact in a positive or negative way on the respondent.

Most of the respondents also mention reputation of the researcher, institution or company conducting the survey to have some impact on their trust. Reputation could be related to having heard about an institution or company before, or thinking that the institution or company itself acts in a trustful way, implying that the survey data will be handled properly as well. One respondent also mentions that it helps on their trust if the survey application looks credible.

Transparency in this context refers to the entities involved with the data gathering being open and transparent about how the data is collected, who will have access to it, and how the data will be treated. Many of the respondents say that they feel their trust in the involved entities’ treatment of data is partially based on faith, or that they simply have to assume the entities do as they say. Entities being open and transparent on their handling of data can significantly improve this trust.
1.5.7 Findings from interviews

The main purpose of the interviews was to see if the possibility for respondents to be anonymous would have an impact on the response rate and quality of submissions for online surveys.

The respondents' definition of being anonymous varied from using a different name to not be able to tell who you are. Some respondents also mixed anonymity with privacy. In the context of online surveys, it might be beneficial for the survey application and the survey administrators to be open and clear about what kind of data is collected, and what is done to ensure there is no way to identify the respondents.

During the interviews, most respondents said it would be easier for them to both answer truthfully, and to complete the surveys if they were given strong guarantees of being anonymous. A strong guarantee in this sense was a theoretical guarantee that there would not be any connection between their identity and their submission, and that it would not be possible to re-identify the respondents at a later point of time. If the same kind of guarantees can be given to respondents in a real online survey application, it is assumed that the possibility to answer anonymously can improve the response rate and quality of the data submitted.

Trust in survey administrators, researchers and application developers was also identified as a factor that could impact the response rate on surveys. Most respondents said that the purpose of a study, along with the reputation of researchers, survey application and organizations conducting a survey, and the transparency on how data would be treated were important factors to them.
Chapter 2

Theory

2.1 What is privacy

Privacy, in its simplest form, is to keep some information to yourself. It can be illustrated by the act of writing a secret diary. The diary is kept secret, not because the content is illegal, but because the content may portray your private thoughts, feelings and experiences that you don’t necessarily want to share with others.

The need for privacy however, is not only related to personal diaries. Most citizens have a legitimate reason to keep some of their actions or thoughts private. And both national and international legislation regulate how governments and organizations can collect and handle private data of their citizens or customers, see the current Norwegian Privacy Law, *Lov om behandling av personopplysninger*[^1] and the to-be-implemented *General Data Protection Regulation*[^7].

2.1.1 Norwegian and European privacy legislation

The current Norwegian Privacy Law, *Lov om behandling av personopplysninger*[^1], in addition to specific legislation, regulates how the government and organizations can collect and handle private data about Norwegian citizens. Private data may include everything that is related to a physical person, and the legislation enforce strict regulation on private data which is considered to be sensitive.

Sensitive data includes information about:

- Ethnic background, philosophical, political and religious believes
- If a person has been suspected, sentenced, charged or convicted of a criminal offense
- Mental and physical health
- Sexual orientation or relationship
- Membership in labour unions

In May 2018, the General Data Protection Regulation \[7\] will be implemented into the Norwegian legislation. The GDPR, unlike the previous Data Protection Directive, must be implemented directly into national law and will provide a common legislation among EU countries.

The GDPR seeks to provide a common legislation for all EU countries. The new regulation will provide new and expanded rights to EU citizens, stricter requirements on data processors and higher penalties for data processors and data controllers in cases of breaches of the regulation.

Rights to data subjects

Some of the new or expanded rights given to EU citizens under the GDPR includes the right to access, right to be forgotten and rights to data portability.

The right to access is to some extent included in the current legislation in Norway. An entity may ask, free of charge, for a copy of all the personal information an organization is collecting or processing regarding the individual. The GDPR specifies that a copy of this data should be provided in an electronic format, within 30 days after the request.

The right to be forgotten, gives individuals the right to have data controllers or processors to remove their personal data when the purpose for processing of the data is no longer relevant or a consent have been withdrawn. The right to be forgotten must be compared with other subjects rights to the public interest in the availability of the data, when data controllers or processors handles such requests.

The right to data portability includes the right to receive and transfer personal data concerning a data subject, in a commonly used and machine readable format.

Requirements to data processors and controllers

The GDPR places stricter requirements on data processors and controllers, including an increased territorial scope, higher penalties, stricter requirements to consent, stricter requirements to breach notification and a requirement of privacy by design.

The increased territorial scope applies the GDPR to all data processors and controllers processing data of data subjects within the European Union, regardless of where the data is processed. The GDPR will also apply to data controllers and processors not established inside the European Union, if their activities relate to either offering goods or services (both paid and free), and monitoring of behaviour of data subjects within the European Union.
The GDPR increases the maximum penalties a data processor or controller may be given in cases of breaching the GDPR. The maximum fines can be up to 4% of the annual global turnover or 20 million euro, whichever is greater.

Under GDPR, there are stricter conditions for consent from data subjects. The request for consent must be given in an intelligible and easily accessible form, and it must be easy for data subjects to withdraw their consent.

In case of a data breach, a breach notification is mandatory and must be given within 72 hours after becoming aware of the breach.

Privacy by design refers to the inclusion of data protection in the design of a system, instead of an addition or afterthought. It also forces data controllers and processors to limit the personal data collected to what is absolutely necessary and limiting such access to only those who need it for processing of the data.

2.2 What is anonymity

In contradiction to privacy, anonymity can be seen as a means to share information with the public, without revealing your identity. The reasons for why a person might want to be anonymous can be many. People might want to express controversial opinions, want to report abuse of power but fear reprisals for doing so, or want to discuss private health information without such information being publicly related to their identity.

2.2.1 Personal information and anonymity

Anonymity can also be related to the collection or processing of personal information. The definition of personal information in the Norwegian Privacy Law defines personal information to be any kind of information that can be related to a person. This information may be objective information, such as where people work or what their income is. It may also be subjective information, like a statement or assessment of a person, such as Eve is always listening. The legislation does not specify that a statement or information must be true to be considered personal information. The statement Alice is cheating on Bob will be considered personal information, even though the statement turns out to be a lie.

For information to be considered personal information, it must also be possible to associate the information with a physical person. Such an association may be either directly or indirectly related to a person. The person associated with the information should also be identified.

The current privacy legislation does not apply to anonymous data sets. As such, it allows data processors to collect and process such data in a way that will not conflict with a physical person’s right to privacy. For
the dataset to be considered anonymous, it should not be associated with a physical person directly or indirectly, or the associated person should not be identified.

The Norwegian Data Protection Authority has published guidelines for anonymization of personal information. The guidelines seek to help organizations anonymize their data, either because they have to publish anonymized data sets, publish information to third parties, be transparent about their own organization and data collection, use collected personal information for new purposes or publish data for statistical purposes.

2.2.2 Anonymity based on policy

A simple way to preserve anonymity is through policy. The policy may describe who should have access to identifying attributes of an entity, how they may be accessed and how they should be presented. When discussing anonymity based on policy, there is a distinction between the services or entities responsible for collecting the data, and the services or entities responsible for processing the data. It should also be noted that the data being processed is not truly anonymous, however, the access control mechanisms and policies in place restrict the services or entities responsible for data processing to re-identify entities.

An example of how policy based anonymity works is online analytics software. Such software may collect a lot of information on visitors to a web-site, such as their IP-address, information about their client and demographic information. In addition, some tools may provide in-depth information about visitors interests or political views, based on which other web-sites they have previously visited. While the analytics software collects information about individual visitors, some information may be aggregated or presented in a way that prevents the web-site owner to re-identify individual visitors.

Anonymity based on policy requires explicit trust in the service or entity responsible for the data collection, as they have full control over the data set being processed. In addition, there is a risk of policies being broken, either deliberately or by accident. As such, the data can not be seen as truly anonymous, even though the identifying attributes are hidden from entities or services processing the data.

2.2.3 Enforced anonymity

As an alternative to anonymity based on policy, I propose the terminology of enforced anonymity. The goal is to remove identifying attributes and de-identify the dataset in an automated process before the data is collected.
2.2.4 Anonymity and pseudonymity

The idea of being anonymous may sometimes be confused with pseudonymity. An anonymous person, or an anonymous dataset will have no identifying attributes related to the person, while a person using a pseudonym, or a pseudonymized dataset will use a replaced set of attributes that can uniquely distinguish an entity without revealing their true identity.

An example of the use of pseudonyms in a dataset may be to replace identifying attributes of a person with other unique attributes, such as their name, address or id number. This allows the owners of the dataset to track persons over time, without revealing their true identity. However, by using pseudonyms, there is an increased risk of re-identification at a later point in time. As such, the use of pseudonymized data are still subject to the current legislation.

While anonymity and pseudonymity both may be used to achieve some of the same goals, e.g. not revealing your true identity, they may have various usages. True anonymity will be the main focus for the rest of this thesis, as the solutions to be discussed must guarantee there is no way to identify or track respondents based on the data collected through the use-case application. In cases where a degree of tracking of respondents over time is desirable, the usage of pseudonyms will be discussed.

2.3 What is identity

An identity consists of one or more attributes. Such attributes may be a name, age or other attributes that can identify a unique identity or a group of entities.

As seen in figure 2.1, an entity, for instance a person, can have multiple identities. The person Alice can have one identity related to her workplace, and a different identity related to her social media platforms. Both of those identities consist of a set of attributes, some which may be unique to one identity, and some which may be shared across several identities. For instance, Alice’s identity on the workplace may consist of an attribute for an ID number, employee ID and a workplace e-mail address. While she may use a private username and e-mail address on her social media platforms. Some attributes however, can be used across various identities. Alice might prefer to use her real name both at work and on her social media accounts.

Identification is the process of providing attributes to claim your identity, such as stating your name, providing a username or a national ID number.
2.4 What is authentication

While the process of identification is concerned claiming an identity, authentication is the process of providing proof, and verification of the claimed identity.

Physical entities may typically authenticate by proving either one of the three categories:

- Something you know
- Something you are
- Something you have

Proving an identity by something you know, may include providing a secret such as a password. The proof of something you are typically includes providing a unique biometric identifier, such as a fingerprint or iris scan. Proving your identity by something you have includes the possession or access to something that belongs to you. Examples of such belongings may be a token device, a cellphone or an access card.

If the authentication process requires two different factors to authenticate, the process is typically called two-factor authentication. Such a process may increase the assurance in the claimed identity, and reduce the risk of successful authentications by a malicious entity.

2.5 Authentication Assurance Levels

ISO/IEC 29115 provides four Authentication Assurance Level (AAL), which is defined in table 2.1. The standard specifies three phases, namely
Table 2.1: Level of assurance

<table>
<thead>
<tr>
<th>Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - Low</td>
<td>Little or no confidence in the claimed or asserted identity</td>
</tr>
<tr>
<td>2 - Medium</td>
<td>Some confidence in the claimed or asserted identity</td>
</tr>
<tr>
<td>3 - High</td>
<td>High confidence in the claimed or asserted identity</td>
</tr>
<tr>
<td>4 - Very high</td>
<td>Very high confidence in the claimed or asserted identity</td>
</tr>
</tbody>
</table>

the enrolment phase, the credential management phase and the entity authentication phase.

During the enrolment phase, an entity may register with a service or organization. The process may differ depending on the level of assurance required. As specified in the standard, an application seeking to achieve AAL 1 only needs to provide a registration form where the entity can fill out a desired username and password, where AAL 4 requires the entity to meet in person with the registration authority.

The credential management phase involves processes related to creation, issuance, activation, storage, revocation, renewal and record-keeping of credentials.

The entity authentication phase involves processes that can establish a confidence in the claim or assertion of an entity. The protocols used may vary depending on the AAL. As such, AAL 1 may only require the entity to provide a valid username and password, while higher AALs may require the use of two-factor authentication and the use of a cryptographic challenge-response protocol.

With authentication assurance level 1, there is a minimal confidence in the claimed or asserted identity provided by an entity. This level of assurance may be applicable in situations where an entity can register themselves to get access to resources, e.g. with a username and a password. There is some confidence that an entity is the same over consecutive uses, however, there is little confidence to exclude the possibility of another entity being able to claim the same identity. The use of cryptographic authentication methods are not required for assurance level 1.

With authentication assurance level 2, there is some confidence in the claimed or asserted identity provided by an entity. Assurance level 2 requires that authentication is done through a secure protocol, to prove that the entity is in control of the credentials. There should also be methods in place to limit or reduce the effectiveness of eavesdroppers or online guessing attacks, as well as methods to reduce the effectiveness of attacks against stored credentials.

Authentication assurance level 3 requires two factor authentication of an entity to prevent erroneous authentication attempts. Secret information related to the authentication protocols must be cryptographically protected both in transit and at rest. Assurance level 3 provides a high level of
Table 2.2: Level of anonymity assurance

<table>
<thead>
<tr>
<th>Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - Low</td>
<td>Little confidence in the assurance of anonymity</td>
</tr>
<tr>
<td>2 - Medium</td>
<td>Some confidence in the assurance of anonymity</td>
</tr>
<tr>
<td>3 - High</td>
<td>High confidence in the assurance of anonymity</td>
</tr>
<tr>
<td>4 - Very high</td>
<td>Very high confidence in the assurance of anonymity</td>
</tr>
</tbody>
</table>

...confidence in the claimed or asserted identity of an entity.

Authentication assurance level 4 provides the highest level of confidence in a claimed or asserted identity of an entity. This assurance level is quite similar to assurance level 3, but adds the requirements of identities of human entities being proved in person and the use of tamper-resistant hardware to store secrets and private cryptographic keys. Examples of usage for assurance level 4 include cases where there is a high risk for harm in the case of an authentication failure.

2.5.1 Authentication assurance levels in ID-porten

The Agency for Public Management and eGovernment (Difi) provides a common log-in solution to public services through ID-porten. ID-porten provides authentication assurance level 3 through MinID and level 4 through BankID, Buypass and Commfides.

Entities authenticating through ID-porten must be registered in the National Registry (Folkeregisteret). The National Registry is operated by The Norwegian Tax Administration and acts as an authoritative source to ID-porten.

2.6 Technical Anonymity Assurance Levels

To properly evaluate technical solutions that provide anonymity for respondents of the use-case application Nettskjema, we propose a generic framework for Technical Anonymity Assurance Level (TAAL). With the TAAL model the solutions we present in the following chapter can be evaluated.

The framework provides four levels of assurance, ranging from 1, with little confidence in the assurance of anonymity, to 4, which provides a very high confidence in the assurance of anonymity.

The framework is meant to assist in evaluating and determining the assurance of anonymity provided by a system, and to provide a common set of requirements for each of the assurance levels. The framework can further be used to determine the minimum required level of technical anonymity provided by an application or system. While the use-case application and proposed implementations to be evaluated in later chapters are web application that interact with web clients, the evaluation framework may apply to
all applications and systems collecting and processing anonymous data sets.

The higher levels of the framework is concerned with how the assurance of anonymity can be increased and enforced on a technical level. It is assumed that trust through contract or policy to some extent may be violated, either by accident or deliberately. As such, allowing third parties to process data on behalf of the system owner is discouraged in the highest levels of the TAAL framework.

2.6.1 Scope of the framework

The technical anonymity assurance levels are concerned with the entire system surrounding the application that is evaluated. This may include the network the application is connected to, or other systems the application interacts with. The higher levels of the TAAL framework will also identify and evaluate risks concerned with cross referencing of data from other third parties, including Internet Service Providers and other organizations which may maintain logs of an entities requests to a system.

All the assurance levels assume that no identifiable information is to be collected through the system or application, such as names, ID numbers, or a combination of attributes that can be used to uniquely identify an entity. Rather, the framework is concerned with metadata that may be collected and processed as part of a data collection process. Such metadata may be the time and date when an entity access a system, their IP address, or the user-agent string provided by the entitlys web browser, which in many cases may be unique.

Some identifying attributes are needed to process and provide efficient communication and a good user experience for an entity accessing a system. The various technical anonymity assurance levels are concerned with when and how this data is processed and how quick such attributes can be discarded from the system after the processing is over.

The assurance levels are not concerned with how third parties integrate with a system, or if an entity deliberately submits identifying information to the system.

**Generic end-user systems and applications as clients**

Generic end-user clients, such as a web-browser, are partially out of scope when evaluating and determining a systems technical anonymity assurance levels. The higher levels of the evaluation framework place strict requirements on the communication between the system and a client.

2.6.2 Requirements for assurance level 1

The lowest level of technical anonymity assurance level provides little confidence in the assurance of technical anonymity provided by the system. Identifi-
fying attributes in the metadata can be collected, processed and persisted by the application or system. However, access to the identifying attributes are restricted, and care should be taken to separate identifying metadata from an otherwise anonymous dataset.

Access to identifying attributes in the metadata, such as IP addresses, should be restricted either by policy or by access control mechanisms. End-users who process the anonymous data sets should not be given access to the metadata.

Examples of access control mechanisms or policies restricting access to identifying attributes may be a requirement that only developers have access to the database, or that only operations have access to logs of IP addresses connecting to the network. As such, the identifying attributes may be hidden from the entity collecting data through the system or application.

Where applicable, the system or application should make use of cryptographic protocols when communicating with a client, e.g. a web browser, to prevent leakage of data while in transit.

As the identifying attributes are still collected and processed, there is an increased risk of accidental or deliberate exposure of data. For instance, access control mechanisms may be misconfigured, and policies may be broken due to lack of knowledge or information about where the policy applies.

### 2.6.3 Requirements for assurance level 2

Technical anonymity assurance level 2 provides stricter requirements regarding to the collection and processing of identifying attributes. The amount of collected attributes should be kept as minimal as possible. For instance, identifying attributes should not be logged in the application.

Systems and applications compliant with the technical anonymity assurance level 2 should identify and act on risks related to cross referencing of identifying attributes with third party sources. A third party may be an Internet Service Provider or another system that also collects or processes identifying attributes related to the data gathering process. In cases where it is necessary to collect and process identifying attributes, they should be encoded in a way that prevents cross referencing in the case of an accidental or deliberate leakage of data from the system. Identifying attributes must not be stored together with other information that can result in a cross referencing of attributes at a later point.

The use of timestamps or other information that may be used to cross reference an anonymous dataset with third parties should be kept to a minimum, since it may lead to re-identification of the entities involved.
2.6.4 Requirements for assurance level 3

The assurance level 3 requires a complete removal of identifying attributes from metadata before further processing in the system or application. Metadata that can be used to cross reference data from other systems or a third party should not be processed or persisted by the application. This requires a level 3 compliant application or system to never process or persist a timestamp together with an anonymous data set.

Identifying attributes may be processed by a trusted third party on behalf of the system if the attributes being processed can not be correlated with the data being collected. The processing av identifying attributes must be defined through contract, and the processing third party may not be part of the same organization or otherwise under the same authority as the organization collecting and processing the original data.

2.6.5 Requirements for assurance level 4

The highest level of assurance provides strict requirements on the processing of anonymous data sets. The application or system should enforce the usage of technology to obfuscate or remove identifying attributes in the metadata before it is processed by the system. Examples of such technologies may include the usage of external proxying and encryption of the anonymous data set in the client before it is sent to the application for processing.

Where applicable, the application should obfuscate the metadata collected to further reduce the risk of re-identification. The application should cache and shuffle encrypted data, and process the data in batches to reduce the risk of re-identification based on order of the data or the time the data was submitted.

Data being processed, and metadata being collected, should never be available outside of a pre-defined trusted domain. The trusted domain requires all applications and systems collecting or processing data to be in direct control or owned by the system owner. This requirement discourages system owners from processing data in the cloud or allowing third parties to process data on their behalf.

2.7 Cryptographic primitives

2.7.1 Cryptographic hash functions

A cryptographic hash function is a one-way algorithm that takes an input of arbitrary length and produces an output of a fixed length. Being a one-way algorithm implies that it should be easy to generate a hash value from an input, but hard or computationally infeasible to get the corresponding input based on the hashed value. A cryptographic hash function should also be
deterministic in the sense that it should always produce the same hash value for the same input.

The use-cases of cryptographic hash functions varies from verifying the integrity of files, to secure storage of passwords or secrets. Hash functions are also used in other cryptographic primitives, for example to generate a [Message Authentication Code (MAC)] or as a [Pseudo-Random Number Generator (PRNG)] to generate random numbers.

A strong cryptographic hash function should satisfy the following three properties:

- Pre-image resistant
- Second-image resistant
- Collision resistant

Where pre-image resistance refers to the one-way functionality of a hash function. Meaning that for a given hash value $h$ it should be difficult to find any other input that corresponds to $h = \text{hash}(\text{input})$. Second-image resistance refers to, given a known input $1$, it should be difficult to find a different input $2$, that produces $\text{hash}(\text{input}1) = \text{hash}(\text{input}2)$. Collision resistance refers to the difficulty of finding any input $1$ and input $2$ that produces $\text{hash}(\text{input}1) = \text{hash}(\text{input}2)$.

**Birthday attack on collision resistance**

The birthday attack is based on a problem from probability theory called the birthday problem. The attack increases the probability of finding any two inputs to a cryptographic hash algorithm that produces the same output.

A typical example of the birthday problem is illustrated by a classroom of 30 students, where the teachers asks for everybody's birthday to determine if anyone are born on the same day.

Given a student A, who is born on 1st of February, the probability will be quite low in finding another student in the class who is born on the same day, which can be illustrated as $1 - (364/365)^2$, or 7.9%. However, the birthday problem focuses on finding any two students born on the same day, which can be seen as the problem of finding any two inputs that produces the same hash value output. In the case illustrated above, the probability of finding any two students with the same birthday is $1 - 365!/(365 - 30)! * 365^{30}$, which yields a roughly 70% probability.

The birthday problem shows that for a hash of $n$ bits, where all outputs are equally probable, a collision can be computed in $2^{n/2}$ time.
Weak hash functions

In recent years, some common hashing algorithms have been deprecated for cryptographic use due to flaws in the algorithms and low collision resistance. The MD5 algorithm has been deprecated for several years, and security researchers have the last years advised against using the SHA-1 algorithm. SHA-1 was recently demonstrated to be broken, as researchers showed they are able to produce different PDF documents which resulted in the same hash value.  

Weak or broken hash algorithms are not safe to be used for cryptographic purposes, such as a hashing of passwords in a database or signing of documents. However, they might still be usefull in other use-cases.

2.7.2 Zero Knowledge Proof

Shafi Goldwasser, Silvio Micali and Charles Rackoff describes ZKP in their paper The knowledge complexity of interactive proof systems. Zero knowledge proof consists of two entities, a prover and a verifier, and can be either interactive or non-interactive. An interactive proof requires the verifier to take part in the verification process. In a non-interactive proof, the prover is able to send a proof directly to the verifier.

An interactive zero knowledge proof must satisfy the following properties:

- Completeness - Given that a statement is true, an honest verifier will be convinced of the statement given from an honest prover
- Soundness - Given a false statement, there should be a very low probability a cheating prover can convince an honest verifier that the statement is true
- Zero knowledge - Given a true statement, a cheating verifier can not learn anything else than the statement provided

Completeness and soundness are general properties for all interactive proofs. The low probability of a cheating prover being successful is because the prover can guess the correct answer to a challenge issued by the verifier. However, this can be counteracted by doing several rounds of verification, resulting in the probability of cheating being close to zero.

The property of zero knowledge is what makes zero knowledge proofs different from other interactive proofs, as the verifier should not be able to learn anything else than the statement itself. This can be illustrated by the use of authentication using passwords. Through the use of zero knowledge proof, the verifier should only be able to convince himself that the prover knows the password. The verifier should however, not be able to obtain the password itself. The properties of zero knowledge proofs is further discussed in section 2.7.4.
Several protocols have been built around Zero Knowledge Proof, and includes Fiat-Shamir\cite{12} and Feige-Fiat-Shamir Proof of Identity\cite{13}.

### 2.7.3 General description of ZKP

Quisquater and Guillou explains the common concepts of Zero Knowledge Proof in their paper *How to explain zero-knowledge protocols to your children*\cite{14}. The paper presents a story of a mysterious forked cave with two passages. Between the two passages lies a magic door that opens on a secret word. The cave is illustrated in Figure 2.2.

As an example, Alice knows the secret word to open the door, and she wants to convince Bob that she knows the secret. However, Alice does not want to reveal the secret itself or anything else than the statement itself.

To prove that Alice knows the secret, she can enter the cave and choose a passage at random. When Alice has chosen her path, Bob goes all the way in to where the cave path forks. He will then, at random, tell Alice which path to come back from. If Alice is in possession of the secret word, she would be able to come back through either passage. The proof can be run multiple times, each time reducing the chance that Alice would be *lucky* in guessing the correct path.

### 2.7.4 Requirements for ZKP

Although the goal of Zero Knowledge Protocols is to convince a verifier that a prover’s statement is true, without revealing any other information, such protocols should also be resistant to cheating parties and malicious
third party entities. Those entities include dishonest provers and verifiers, eavesdroppers and malicious third parties.

A dishonest prover tries to convince the verifier that she knows a secrets, even though she does not, while a dishonest verifier tries to get more information from the verifier than the statement itself. Those issues are addressed by the properties of soundness and zero knowledge, and is also discussed in detail in Identification with zero knowledge protocols by Annarita Giani \[15\].

- **If the prover does not know the secret, she should not be able to pretend to know the secret** - If Alice presents a false claim to Bob, he should be able to verify that the claim is indeed false through multiple rounds of verification.

- **The verifier should not be able to learn more than the statement itself** - Bob should not be able to learn more than the fact that Alice possesses a secret, which means that even though Bob is convinced that Alice’s statement is true, he should not be able to convince other entities that he knows the secret as well.

Going back to the example of the mysterious cave in section \[2.7.3\]. If Alice presented a false claim that she knew the secret word to open the door, this could be verified by Bob as false if Alice were unable to come back through the path he chose. As there are two paths, and a path is selected randomly, Alice’s chance of being lucky will be halved for every round. This could also be expressed as $0.5^n$ where $n$ is the number of rounds.

In the same example, there can be assumed that Bob videotapes the verification process. Although Bob is convinced that Alice knows the secret word, the videotape would only prove that Alice returns from the path that Bob chose. Other entities might likely assume that Alice and Bob agreed on the paths before the verification process. It would also be impossible for Bob to prove to any other entity that he is in possession of the same secret.

### 2.7.5 Zero Knowledge Authentication

Building on the basic principles of Zero Knowledge Proof comes Zero Knowledge Authentication protocols. Those protocols allow an entity to authenticate to another service in a secure fashion, without revealing it’s identity or any other information than the fact that they are a valid entity with access to the service. One protocol providing Zero Knowledge Authentication is the Fiat-Shamir protocol, which will be discussed below.

**Requirements for Zero Knowledge Authentication**

Zero knowledge protocols aims at proving knowledge of a secret without revealing the secret itself to the verifier or anyone else. As the secret itself
is never stored by the verifier or transferred over the wire, such protocols can reduce possible attacks from cheating verifiers or malicious third parties. The protocol should also protect against a cheating prover, e.g. an imposer that tries to authenticate as someone else.

The Fiat-Shamir Protocol

In the paper *How to prove yourself: Practical solutions to identification and signature problems* [12], Amos Fiat and Adi Shamir describes the Fiat-Shamir protocol. Fiat-Shamir is a protocol used to authenticate users’ through the use of zero knowledge. The protocol is probabilistic and have a $2^{-n}$ chance of a cheating prover to fool the verifier [16]. Fiat-Shamir requires several rounds to ensure that a prover is honest and it relies on the hard problem of factoring.

The protocol requires a trusted third party, and consists of three steps. **Set-up, the protocol and verification.**

During the set-up, the trusted third party generates a number $n = pq$ where $n$ is public and $p$ and $q$ are kept private. The prover selects a secret $s$ which is greater than 1 and smaller than $n - 1$. The prover then generates a public $v = s^2 \mod n$ that she registers with the trusted third party.

While running the protocol, the prover choses a random number $r$ which is greater than 1 and smaller than $n - 1$. The prover then sends $x = r^2 \mod n$. The verifier sends a random $e$ which can be either 0 or 1. The prover returns $y = rs^e \mod n$.

During verification, the verifier accepts the answer if $y^2 = xv^e \mod n$. The protocol and verification is run until either the prover fails, or the verifier is satisfied that the provers identity is real.

To reduce the probability that the prover can cheat, the value of $e$ should be completely random. The randomness of this value can be related to the cave example were the verifier choses which path the prover should come back from. The random value $r$ must also be generated again for each round, or else it may be possible for a cheating verifier to determine the provers secret $s$.

Possible drawbacks

Although the Fiat-Shamir protocol allows authentication between users without sharing secrets like passwords or private keys, there are some possible drawbacks.

Fiat-Shamir relies on multiple rounds of verification. Although the computational load is relatively low and might be suitable on for example RFID access cards or passports [15], network delays and high load of users might degrade the performance on web applications.
Password based authentication relies on the password being kept secret. Although zero knowledge protocols aims at proving knowledge of a secret without revealing the secret itself, such protocols don’t protect against a malicious user who have already obtained the secret. As with password based authentication, two factor authentication can be used to reduce successful attacks against user accounts on web applications[17].

2.7.6 Blind Signature Protocol

David Chaum describes blind signatures in his 1982 paper *Blind Signatures for Untraceable Payments* [18].

While modern electronic payment schemes, such as the use of a credit card to pay for goods online, may be efficient and simple to use, they may have serious implications on individuals privacy. Chaum describes how tracking of individuals payments may reveal a lot about their *whereabouts, associations and lifestyle*. For instance, the tracking of payments for transport and hotels may reveal where an individual may have been on a given time.

Traditional untraceable payments, such as physical cash, allows an individual to pay for goods and services without revealing their identity or being tracked. However, such payment schemes may make it harder for individuals to provide a proof of payment, and allows for *theft of payment media, black payments for bribes, tax evasion and black markets* [18].

The proposed payment scheme by Chaum seeks to improve both on the issue of privacy for individuals, and to achieve better control over the payment media. As such, the scheme for untraceable payments using blind signatures should have the following properties:

- Inability of third parties to determine payee, time or amount of payments made by an individual
- Ability of individuals to provide proof of payment, or to determine the identity of the payee under exceptional circumstances
- Ability to stop use of stolen payments media

**Analoge example of Blind Signatures**

Chaum illustrates the use of blind signatures through the blind signing of paper envelopes. In the example, a trustee wishes to hold an election with a secret ballot. However, the electors are unable to meet in person, and can not drop their ballott in a single hat during the election day.

To enable the electors to cast their vote, while still keeping their vote secret, the electors can put their vote in a carbon paper lined envelope, which is again put into another envelope. The envelope is addressed to the
trustee, with the electors return address written on the back. The envelope is then sent to the trustee.

When the trustee receives an envelope from an elector, she removes the outer envelope and signs the carbon lined envelope. She then returns the signed envelope to the elector.

On election day, the elector removes his vote from the carbon lined envelope and puts it into a new envelope addressed to the trustee, without his return address written on the envelope. When the trustee receives the ballot, she knows it is a valid vote, since she has previously signed it, but she will not be able to determine which elector cast the vote.

As there are no identifying attributes on the vote, the trustee is able to publish the votes. This allows anyone to count the votes, and an observant elector who remembers identifying aspects of his own vote can verify that his vote has actually been counted. The trustee, who have never seen the actual ballot slips while signing them, and assuming all of her signatures to be identical, will be unable to correlate a vote with an elector.

2.8 Existing technology for anonymous surveys and Identity Management

2.8.1 Anonize

Anonize is a system for creating ad-hoc anonymous surveys for invited participants, providing a strong assurance of anonymity, while ensuring that each respondent is only allowed to submit at most one response\textsuperscript{[19]}. The Anonize system, including the registration, submission and verification scheme, and how it can be used as a protocol enabling anonymous authentication of respondents in the use-case application Nettskjema will be further discussed in chapter\textsuperscript{[5]}

2.8.2 Attribute-based credentials

Attribute-based credentials allows entities to authenticate and verify their credentials in a privacy preserving manner. Attribute-based credentials is discussed in the technical report \textit{D2.1 Architecture for Attribute-based Credential Technologies - Version 1}\textsuperscript{[4]}.

The project \textit{Privacy-Preserving Attribute-Based Credential Engine (p2abcengine)}, available at \url{https://github.com/p2abcengine} provides a framework for attribute-based credentials. The underlying cryptography is provided by either \textit{IBM Identity Mixer} or \textit{Microsoft U-Prove}.

As both Identity Mixer and U-Prove is quite similar in terms of the issuance- and verification process, Identity Mixer will be used as the example when discussing attribute-based credentials in chapter\textsuperscript{[5]}.
Chapter 3

Nettskjema as a use case

3.1 About Nettskjema

Nettskjema is an application for creating online surveys and collecting answers from respondents. Its main target group are students, researchers and administrative employees in the university sector. The application is developed by the Web Application Development Group (WAPP) at the University Center for Information Technology at the University of Oslo (UiO). The source code is open and available from https://bitbucket.usit.uio.no/projects/WAPP/repos/nettskjema/.

Nettskjema currently supports three types of respondent groups on a survey or online form. They can either be signed in, invited, or the survey can be open for anyone. Signed in respondents can authenticate with their credentials at the University of Oslo, through Feide or through ID Porten[20]. Invited respondents get a personalised link by email containing a token to authenticate with Nettskjema. Surveys and forms open for everyone is accessible for all respondents with access to the unique URL for the survey and no authentication is required.

Answers in Nettskjema can either be stored in Nettskjema, asynchronously encrypted and stored on a secure server in Services for sensitive data (TSD) or sent to the survey administrator by email. TSD is a service for secure storage of personal sensitive information hosted and developed by the University of Oslo. It is also possible to specify if detailed information about the respondent should be saved together with the answer where such information is available. This information may include the date and time the answer were submitted and the e-mail or username of the respondent.

One of the main target groups in Nettskjema are students and researchers at the university faculties, and one of the objectives is to enable those users to create online forms used in data gathering for research projects and theses. Surveys and forms collecting personal identifiable information are subject to legislation under the Norwegian privacy laws, and projects col-
lecting personal identifiable information must be approved by a government instance such as NSD before the data gathering can start. According to NSD, projects collecting any of the information below needs approval before the data collection can start [21].

- Directly identifiable information, e.g. name or social security number
- Background information that may identify individuals, e.g. a combination of workplace, age and gender
- Registration of personal data using online surveys, including IP and e-mail addresses
- Recordings and capturing of audio, video etc.

Nettskjema does not log the respondents IP address by default, but such entries may exist in server and access logs available for operations and security staff. As such, all surveys in Nettskjema used for a research project should be reported. At the time of writing, there is no solution for online surveys in Norway known to the author that satisfies NSD or REKs requirements for anonymous collection of data.

Further development of Nettskjema to conform with NSDs requirements related to anonymous surveys may be beneficial for both survey administrators and respondents. Results from interviews, see chapter 1.5 on page 13 have shown that anonymous surveys can improve the response rate and data quality of submissions. Anonymous surveys may also make the process of data gathering easier for survey administrators, as they can start the data gathering without a request for approval.

There is ongoing work in the University and College sector to streamline development and provide common services to all Universities and Colleges in Norway. In this context, it might be beneficial to provide Nettskjema as a service to other institutions outside of the UiO organization. Providing Nettskjema to other institutions and organizations may provide new use-cases for the development of Nettskjema, and the need for anonymous surveys is expected to increase in the future.

### 3.2 Risk analysis of Nettskjema

A risk and threat analysis is conducted by the Webapplication development group, in collaboration with IT-Security staff and UiO lawyers at least every second year. The risk and threat analysis is extended in cases where new threats in the application is discovered.

The risk and threat analysis is available to partners and customers of Nettskjema. Access can in special cases be given by contacting the Webap- plication Development Group.
3.3 About TSD

Services for sensitive data (TSD) is a service developed and maintained at UiO. The service is described in detail in the whitepaper *Services for Sensitive Data (TSD) - Whitepaper v4.2* by Gard Thomassen and Maria F. Iozzi, and a brief summary is given below.

The service provides secure storage of sensitive data for research and analysis. TSD consists of several projects running in separate virtual networks. Hosts in one project is only accessible from other hosts in the same project, and will not be accessible from the outside. Researchers can be granted access to a project, and will sign in using two-factor authentication to get access to work on the data stored in the TSD project.

Nettskjema integrates well with TSD. A survey in Nettskjema can be set up to deliver data to TSD. The researcher will then have to generate a Pretty Good Privacy (PGP) key, and insert the public key to the settings page of the survey in Nettskjema. Submissions in Nettskjema will then be encrypted using the researchers public key, and then be delivered to TSD.

At the time of writing, TSD supports two ways of delivering data to a project. In the first method, data will be encrypted and stored on a common file share accessible by both Nettskjema and a TSD client. When data is dumped from Nettskjema to the temporary storage, the TSD client will fetch the files and import them to the corresponding project. The files will be deleted from the temporary space after successful transmission. In the newly implemented method, Nettskjema is able to deliver files directly to a TSD project over a RESTful API.

3.4 Respondents

Nettskjema supports three groups of respondents for online surveys. A respondent may be either signed in, invited or the form may be open for anyone with access to the forms unique URL. Open forms may also be embedded on a webpage. For all respondent groups, it is possible to choose if Nettskjema should append personal information to the answer if this is available. This information includes a username or e-mail where this applies, and the date and time the answer were submitted. Open surveys also support saving the HTTP-referer-url, i.e. the website that linked to the survey.

3.4.1 Signed in

Nettskjema supports multiple types of signed in respondents. As Nettskjema is developed and commonly used at the University of Oslo, authentication through Weblogin with UiO credentials is the most common. Weblogin provides a Single Sign-On (SSO) to services at UiO over the Security Assertion Markup Language (SAML) 2.0 protocol, where UiO acts as the Identity Provider.
Through Weblogin, Nettskjema also supports authentication through Feide. Feide is developed and maintained by UNINET, and is a national service providing a single electronic identity for everyone in the education sector.

Nettskjema also supports authentication through ID-porten, which is a developed and maintained by DIFI. ID-porten is a service to log in to most public agencies such as https://altinn.no. To use ID-porten, the users must have an electronic ID from either MinID, BankID, BankdID on mobile, Buypass or Commfides. ID-porten also supports multiple levels of security assurance, where MinID provides Authentication Assurance Level (AAL) 3 and the last four provides AAL4. All the electronic IDs use two-factor authentication to authenticate the user.

3.4.2 Invited

Invited respondents in Nettskjema are invited by e-mail. The respondent will get an e-mail with a personalised link containing a token that will authenticate the respondent to Nettskjema. The token is only valid for one single use, and the respondent can request a new token to be sent on e-mail if the previous token have either been used or expired. A token in Nettskjema is linked to the respondent, so administrators of a survey can verify that a respondent have answered the survey and send out reminders to respondents who have not yet answered.

Survey administrators can gather e-mails from respondents in various ways, including asking the respondents in person, or collecting e-mails from authoritative sources. Nettskjema can to some extent verify the validity of an e-mail.

3.4.3 Open surveys

Open forms and surveys are open for everyone with access to the surveys unique URL. The URL can for instance be shared by e-mail, published on a webpage or be distributed through other communication channels. Open surveys can also be embedded on a webpage, either through the University of Oslos publishing system Vortex, or with an iframe code snippet that can be inserted into a HTML page.

3.4.4 Authentication Assurance Levels in Nettskjema

It has been proposed in the development group of Nettskjema to enhance the user experience for survey administrators when they create new surveys. One of the proposed solutions is to make it easier for survey administrators to choose the authentication method of respondents that fit their requirements. By identifying the consequences of authentication failure, and the potential
Table 3.1: Potential impact of authentication failure

<table>
<thead>
<tr>
<th>Incentive</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>low</td>
<td></td>
<td>medium</td>
<td>high</td>
</tr>
<tr>
<td>2</td>
<td>low</td>
<td>medium</td>
<td>high</td>
<td>high</td>
</tr>
<tr>
<td>3</td>
<td>medium</td>
<td>medium</td>
<td>high</td>
<td>very high</td>
</tr>
<tr>
<td>4</td>
<td>medium</td>
<td>high</td>
<td>very high</td>
<td>very high</td>
</tr>
</tbody>
</table>

Table 3.2: Mapping between impact of authentication failure and Authentication Assurance Levels

<table>
<thead>
<tr>
<th>Impact</th>
<th>Authentication Assurance Level</th>
</tr>
</thead>
<tbody>
<tr>
<td>low</td>
<td>1</td>
</tr>
<tr>
<td>medium</td>
<td>2</td>
</tr>
<tr>
<td>high</td>
<td>3</td>
</tr>
<tr>
<td>very high</td>
<td>4</td>
</tr>
</tbody>
</table>

The impact of an authentication failure can be directly identified, as described in Table 3.1.

The impact of authentication failure can be directly mapped to the Authentication Assurance Levels described in Chapter 2.5 on page 25, as seen in Table 3.2.

Authentication of respondents in Nettskjema maps partially to the assurance levels specified in ISO/IEC 29115. Forms open for everyone is associated with a low consequence related to authentication failure, and can be considered an assurance level 1.

Respondents invited by e-mail get a one-time token that maps their claimed identity with a survey or form in Nettskjema. Invitations do not map directly to an assurance level, as the assurance partially depends on how the e-mails are gathered, how the content of the e-mail is sent to the invited respondent, and how the authentication tokens are treated. All HTTP traffic in Nettskjema is protected by Transport Layer Security, commonly known as HTTPS. As the communication with Nettskjema is encrypted between the client and the Nettskjema server from the moment a respondent clicks the invitation link in their e-mail client, there is a high confidence that the authentication token or the submission has not been altered by malicious third parties. The length of a token, its single use property and the mapping between a token and a survey further reduces the chance that a malicious third party can authenticate as a valid invited user. Depending on the confidence in the claimed identity of the e-mail owner, the assurance level of invitations can be placed between assurance level 1 and 2.

Nettskjema supports various methods of authenticating signed in re-
spondents, through multiple identity providers. UiO and Feide users can sign in with their regular credentials. As neither UiO nor Feide provides two-factor authentication, this places the authentication in assurance level 2. In addition, Nettskjema supports authentication through ID-Porten, which offers authentication assurance levels 3 and 4.

### 3.5 Question types

Nettskjema supports a wide range of various question types, including open text questions, radio buttons and checkboxes and matrix questions. In addition to those standard question types, Nettskjema provide special fields for ID numbers, e-mail, file uploads, numbers and date fields. Some of these question types may directly collect personal identifiable information, such as the ID number and e-mail fields.

#### 3.5.1 Surveys that contain personal identifiable information

According to Norwegian Privacy laws, it is prohibited to collect and store personal identifiable information when the reason for storing the information is no longer valid. To comply with this, Nettskjema regularly generates a report containing forms and surveys suspected of collecting personal identifiable information, which have not received new responses or otherwise been active the last 6 months. Survey owners may then be notified and asked to export their data if they are still needed, and personal information will be deleted from Nettskjema.

Surveys that may contain personal identifiable information includes all surveys which requires authentication by the respondent, as long as the survey is set up to include personal information about the logged in user and the time of submission, and all surveys that have special question fields that may directly identify the respondent, such as the e-mail and ID number field. Question fields that include one or more strings from a pre-defined list of typical questions that ask for personal identifiable information will also be marked for deletion. Such strings may include last name, username and study id [24].
Chapter 4

Anonymous responses in Nettskjema

In 2016, the Web Application Development Group started work on improving anonymous responses in Nettskjema. Nettskjema already supports a simple solution that will hide the link between a submission and an authenticated user from the survey administrator. Collection of additional metadata such as the time of delivery will also be disabled for surveys set up to not collect personal identifiable information. However, the current implementation does not comply with NSD’s definition of anonymous online surveys and forms. As such, all surveys in Nettskjema used to collect data for scientific research must be reported to, and approved by NSD before the data collection can begin.

The FAQ section on NSD’s website says the following on online forms:

For online surveys not to be subject to notification, one has to make sure that the IT solution is completely anonymous (among other things, the respondent’s email or IP address cannot at any moment be connected to the survey), and that the survey does not contain questions about identifiable information.

In addition to e-mail and IP-addresses, the survey should not ask for directly or indirectly identifiable information. Directly identifiable information may include the name or ID number of the respondent. Indirectly identifiable information can be any kind of information that, when grouped together, can identify a respondent. An example of such a combination is the combination of age, gender and workplace.

The requirement that e-mail addresses, or other unique attributes that can identify a respondent, can not be stored in an anonymous solution implies that only surveys open for everyone can be truly anonymous. As there
Figure 4.1: Banner telling respondents that no personal identifiable information is recorded

Source: Nettskjema

are no unique identifiers, the proposed solutions can also not limit the number of submissions from a respondent, or otherwise enforce who can post a submission and not.

Personal identifiable information in Nettskjema surveys can be divided into two groups. The first group contains metadata that is handled or collected by Nettskjema and the infrastructure on the UiO network. The second group contains information from question fields in a survey.

Metadata and other personal identifiable information collected by Nettskjema and the surrounding infrastructure will be the main focus when discussing improvements to anonymous submissions in Nettskjema. Personal identifiable information collected through question fields is the responsibility of the survey administrator. While Nettskjema can disable the usage of some special question fields, and provide guidelines on what kind of information that should not be collected through an anonymous survey, the decision on which questions to ask is ultimately up to the survey administrator.

4.0.1 Current situation

In the current implementation of Nettskjema, it is possible to set up a survey to not collect personal identifiable metadata about a respondent. This metadata includes the time and date of the delivery of a submission. Surveys that require authentication by the respondent, such as a Feide login or an e-mail invitation, can be set up to not collect identifiable metadata. In those cases, the link between a respondent and a submission will be hidden from the survey administrator, but some information about the respondent will be stored in Nettskjema.

Since authenticated surveys use unique attributes to identify a respondent, such as an e-mail address or a Feide username, the surveys will not be anonymous. Ways to improve anonymity for authenticated surveys will be further discussed in part 5.

Open surveys configured to not collect any personal identifiable information will not store any information about the respondent in Nettskjema. However, it is possible for the survey administrator to ask for such information. For instance, it is possible for a survey administrator to add fields
for e-mail addresses, ID numbers or ask questions that can either directly or indirectly identify a respondent.

It is also possible for respondents to request that a receipt of their submission should be sent to them by e-mail. In those cases, a copy of the submission will be processed by Nettskjema while sending the e-mail receipt. To prevent accidental loss of e-mails, Nettskjema will temporarily persist and queue e-mails in the database. The receipt will be deleted from the database once it has been sent successfully.

Nettskjema does not store the respondents IP-address together with a submission. However, the respondents IP-address may be stored in various logs at UiO. Those logs may include application logs in Nettskjema, access logs from front-end proxies and load balancers and NetFlow logs, a log over which IP-addresses that connect to each other, at the UiO network. Some of those logs are required to monitor, react to and prevent security incidents at the UiO network. Application logs can also be useful to application developers for debugging errors and bugs in the production environment, and to improve features in Nettskjema.

Access to the various logs are restricted, and survey administrators will never have access to any of the logs. However, it might be possible for a malicious entity with access, or an attacker using a compromised user account or machine, to correlate those logs with submissions stored in the Nettskjema database. As such, for a solution to be truly anonymous, it should not be possible to correlate any submission with an IP-address or a request done to a service on the UiO network.

De-identification of authenticated submissions

As with open surveys, it is possible to configure an authenticated survey to not collect personal identifiable data about the respondents. An additional requirement may be that each respondent should only be allowed to post at most one submission. This is useful in cases where the survey administrator wants to be able to control who can post a submission, but do not want to be able to correlate a person with a submission.

Authenticated surveys that are configured to not collect personal identifiable data, will not store any information about the respondent together with their submission. However, some metadata about the respondent will be stored in a separate table only accessible to the Nettskjema application. This information includes who are invited or allowed to submit a response to the survey, and if, or how many submissions they have posted.

This information is needed to allow survey administrators to send out reminders to users who have not yet posted a submission, and to limit the amount of submissions a user should be able to submit.

As the information about a user or invited respondent having posted a submission or not is stored in a separate table from the submissions itself,
it will not be possible for survey administrators to link a submission to a specific respondent. However, it may be possible for technical staff with access to the database to deduct such knowledge, due to the way new rows are appended to the table.

4.0.2 Benefits

As discussed in section 1.5 on page 13 and 3.1 on page 38, providing a solution for anonymous surveys can improve the response rate and the quality of data gathered. A robust solution that complies with NSD’s requirements for anonymous surveys can also make the process of data gathering for survey administrators and researchers easier, as they might not need an approval before the data collection can begin.

A solution that can provide truly anonymous submissions may also protect the respondents in the case of an intrusion, data breach or other security incident on the UiO network. The to-be-implemented General Data Protection Regulation places strict requirements on the Data Controller and Data Processor in cases of data breaches. Breaches must be reported to the country’s Supervisory Authority within 72 hours of the breach. However, no notification is needed if the data set is anonymous or pseudonymisation techniques have been implemented by the Data Processor.

4.1 Requirements for a solution

Before drafting and evaluating the possible solutions, some key requirements must be identified. The requirements are related to how the solutions can preserve anonymity for the respondent, and how the solutions can withstand various attacks from both a malicious insider and third parties.

Solutions should at least comply with NSD’s requirements for anonymous surveys, as discussed in chapter 4, and provide strong assurance of anonymity to the respondent.

4.1.1 Removal of identifying attributes

A solution that provides anonymous submissions on a survey should not collect any attributes that can be used to uniquely identify a respondent. Those attributes may include various kinds of metadata that can be collected together with a submission by Nettskjema, and includes the IP-address of the respondent, various HTTP-headers and the time of delivery.

To reduce the possibility that those attributes can be used to identify a respondent, they should be filtered out or excluded as early as possible in the delivery process. For instance, IP-addresses should not be logged by the network or the application. Logging in the application should be kept to a minimum, and should not include any information related to a respondent.
or a submission. Where possible, attributes that can identify a respondent should not be sent from the client to Nettskjema.

Anonymous surveys should also not collect the e-mail address of respondents at any time before, during or after the process of delivering a submission, since it may be possible to correlate an e-mail with a respondent or their submission. Hence, it should not be possible to send out a link or invitation to the survey on e-mail from Nettskjema, or to send receipts of delivered submission to respondents.

4.1.2 Identifying question fields

Special feields that are used to collect directly identifiable information about a respondent, such as the ID-number- and e-mail address fields should not be allowed in an anonymous solution for Nettskjema, as they defeat the purpose of an anonymous survey. Removing those fields from an anonymous solution also lowers the risk of directly identifiable information being accidentally collected by the survey administrators.

It may still be possible for survey administrators to use other fields to collect personal identifiable information, such as using the free text fields to ask for the respondents age, name or similar. While using question fields to collect personal identifiable information in an anonymous survey defeats its purpose, this issue should be identified and acted upon when it happens. It should be possible for respondents to report surveys asking for identifiable information in anonymous surveys, and survey administrators should be made aware that some questions can either directly or indirectly identify a respondent. There should also be done improvements to the implementation of Nettskjema in identifying surveys that may contain personal identifiable information.

4.1.3 Attacks on solutions

The proposed solutions should identify and protect against typical attacks that can breach the confidentiality of the respondent and data, the integrity of the data, and the availability of the service. It should be assumed that both malicious insiders and malicious third parties may try to attack the respondents, data and the service. And that such attacks may be targeted at specific respondents or targeting a broader audience, such as all incoming traffic to the service.

Combination of logs and metadata

While logging in an anonymous solution should be kept to a minimum, there might exist logs on the UiO network, on client machines and with third party ISPs indicating that a client machine have been sending requests to
an anonymous solution for Nettskjema. Access to such logs, in the combination with metadata from Nettskjema can possibly link a submission to a respondent.

4.2 External proxy

One early proposal for anonymous responses was to set up an external proxy for anonymous surveys. The proxy would be hosted by a trusted third party, and traffic from respondents would be routed through the proxy to Nettskjema.

A trusted third party in this context could be any actor that could provide sufficiently strong guarantees regarding the privacy of the respondents. Ideally, no requests to or from the proxy should be logged by the service provider. In cases where this is not possible, employees at UiO should be restricted from accessing the logs. In the case of a security incident on the UiO network related to the proxy solution, IT security staff could be granted access to access logs for digital forensic purposes.

Through the proxy, IP addresses of the respondents could be rewritten, and HTTP-headers containing information about the respondents User-Agent or other identifiable information could be removed. It does however, not prevent the proxy owner to log the IP addresses or other metadata about respondents.

4.2.1 Integration with Nettskjema

Integration with a proxy solution in Nettskjema would require a minimal amount of effort. A simple implementation could require that anonymous surveys would get a different url that would point to the proxy. Nettskjema could also restrict respondents from accessing the anonymous survey directly in Nettskjema.

However, care should be taken to ensure that all traffic on anonymous surveys would be served through the proxy. For instance, all assets such as style sheets, images and javascript should be loaded through the proxy. Loading assets directly from the Nettskjema servers could compromise the anonymity of the respondent.

4.2.2 Benefits and disadvantages

Implementation of a proxy solution would remove the IP address of the respondent from Nettskjema logs. Given that the time of delivery for a submission would not be stored in Nettskjema, it would be harder for a malicious insider to correlate logs with a submission to identify the respondent.

However, it would still be possible for the third party to log requests to and from the proxy. For the proxy to be able to rewrite the respondents IP
address and request headers, it would also have to terminate the TLS encryption before the request were to be forwarded to Nettskjema. A malicious entity with access to the proxy server could therefore be able to access the submissions in plaintext, and easily correlate those with the respondents IP address. While it may be reasonable to assume that security mechanisms in place at the proxy server are equal to or stronger than security mechanisms at the UiO network, this solution would provide little to no extra protection and a single point of attack surface that could compromise all anonymous submissions to Nettskjema.

The proxy solution also only solves the initial problem of logging of IP-addresses in Nettskjema. However, such logs may still exist in the proxy, and may be accessible to technical staff if a security incident on the UiO network occurs.

A simple proxy will also allow identifying question fields to be added to a survey, either accidentally or deliberately by a survey administrator, as the proxy is not concerned with the structure of the survey. More adjustments to the Nettskjema application should be done to account for this, e.g. by not allowing Personal ID or email fields to be insterted into an anonymous survey.

4.3 Anonymous frontend

Another proposal made in the Web Application Development Group (WAPP), is to develop a separate front-end service for anonymous submissions. This solution requires more effort than using a simple proxy, but it may also be more robust and provide stronger guarantees of anonymity to the respondent.

There has been done extensive work on providing an open RESTful API for posting submissions in Nettskjema. The API enables third party applications to easily integrate with Nettskjema. The use of an open API accessible over the Internet would make it possible to develop a standalone front-end that can be placed in a trusted network or with a hosting provider that can guarantee a sufficient level of anonymity.

The proposed front-end would be a restricted version of the current answer-module in Nettskjema, and support for special fields used to collect personal identifiable information can easily be disabled.

To protect the respondents from accidental exposure of their responses, submissions should be encrypted using a public key from Nettskjema in the client browser. The encrypted submissions can then be queued by the front-end server and delivered to Nettskjema in batches to obfuscate the time of delivery.
4.3.1 Nettskjema API

Nettskjema exposes some simple RESTful endpoints to retrieve the survey structure and deliver submissions. The endpoints produce and consume JSON objects, which are used by the JavaScript code in the regular front-end of Nettskjema. The same endpoints can be used to integrate Nettskjema with mobile applications, and a standalone anonymous front-end server.

A GET request to the `/answer/answer.json?formId=formId` endpoint would return the survey structure, including all questions fields, information about how the survey should be rendered, e.g. page-breaks, and metadata about the survey. The same JSON structure can be used by the anonymous front-end application to render the anonymous survey.

During the spring 2017, work began on enhancing the API provided by Nettskjema, and new endpoints for administration of surveys, and posting encrypted submissions are under development.

4.3.2 Implementation with Nettskjema

As with the proposed proxy solution, some adjustments should be made to the Nettskjema application to enhance the privacy and security of the respondents.

The Nettskjema application should mark the survey as an anonymous survey, and block attempts to answer the survey directly from Nettskjema.
to prevent respondents accidentally exposing their identity when posting a submission.

When an anonymous survey is created, Nettskjema should register the survey with the anonymous front-end. The JSON structure, and other static assets such as images, can then be cached by the anonymous front-end and easily be served to respondents when a request is done to the front-end server.

**Push vs pull model of survey structure from Nettskjema**

In one early proposal of the anonymous front-end server, it was assumed that a request could go directly from the client, through the front-end server, to Nettskjema to fetch the JSON structure of the survey. Such a model would be easy to implement, and create a loose coupling between Nettskjema and the anonymous front-end as the front-end server could use the regular endpoints in Nettskjema to fetch the structure.

However, such a model would increase the chance of an attacker with access to IP-logs, either from the UiO network or from the respondents ISP, to successfully correlate a submission with the respondent.

By allowing the front-end application to request the survey structure from Nettskjema each time a respondent accessed the application to post a submission, a timestamp would be left in the requests logs to Nettskjema.

While such an attack would require access to the request logs in Nettskjema, access to IP or Network flow logs related to the respondent, and access to the decrypted submissions stored in Nettskjema, it can easily be mitigated by forcing Nettskjema to push the structure directly to the front-end, where no logging of requests are done.

Caching the survey structure at the front-end server will also have the benefit of reducing the load on the Nettskjema backend.

**4.3.3 Placement of front-end server**

The placement of the front-end server on a trusted network or with a trusted service provider builds on some ideas from the anonymous proxy solution described above. Access to the instance should be restricted, and logging can be kept to a minimum.

The solution should provide strong assurance for the respondents anonymity, and security mechanisms should be in place to ensure that the confidentiality or integrity of the respondent or submitted data will not be breached. As such, the anonymous front-end server can in theory be placed with any public cloud provider, such as Amazon Web Services, or Microsoft Azure.

It may, however, be desirable to place the front-end server on a dedicated network within the UiO organization. During a meeting between the...
Operations and IT-Security staff, it was proposed to set up such an application on a dedicated UiO network with the Network flow logs disabled. It would also be possible to provide a dedicated DSL line from an external ISP in to the network. As such, request and access logs can be disabled in all parts of the network in control by UiO.

Running the anonymous front-end on in-house hardware may also be cost effective for the organization, as the computing power needed and network setup is already available. However, there is an increasing demand from management and the Norwegian government to move applications to the cloud. Cloud computing is listed among the top three priorities of USIT’s plan for strategic direction and priorities[6]. The front-end server should therefore be developed to be cloud ready, and implement extra security measures where possible to ensure that the anonymity of respondents will not be breached, unrelated to where the front-end server is run.

4.3.4 Encryption in the client browser

Submissions would be encrypted by the client web browser using either a public RSA key or a certificate provided by Nettskjema, before the submission would be sent to the anonymous front-end server. The private key will only be installed in the Nettskjema backend application. As such, it will not be possible for the anonymous front-end server to decrypt a submission once it is delivered.

Most modern web browsers support the WebCryptography API. Internet Explorer 11 supports an older version of the specification, older versions of Internet Explorer does not support the API. The Web Cryptography API supports encryption, decryption, key import and export, and key generation. It is possible to encrypt using both symmetric and asymmetric keys. The specification of the Web Cryptography API does not specify a mandatory set of cryptographic algorithms to be implemented, but provides methods to detect which algorithms that are supported by the underlying client[26]. The Mozilla Developer Network currently lists AES-CBC, AES-CTR, AES-GCM and RSA-OAEP as algorithms that can be used in encryption and decryption of data in the browser.

Support for a modern cryptography API in the client browser is necessary to provide sufficient assurance of anonymity to the respondent. It is possible to use third party libraries to perform cryptographic operations in older browsers that doesn’t support the Web Cryptography API, but this is not desirable. The usage of a third party cryptography library increases the code complexity, and it may contain vulnerabilities that can be hard to identify and patch in a production environment.
4.3.5 Security mechanisms in anonymous frontend server

To further enhance the anonymity of respondents, the encrypted submissions would be given a random ID when received in the frontend server. Submissions would then be sorted by ID and stored in memory until they are scheduled for batch delivery to Nettskjema.

The delayed, scheduled delivery of submissions in batches to the Nettskjema backend is important to ensure that no submission can be directly linked to a timestamp, IP-address or other information that can help a malicious attacker re-identify a respondent once it’s submission have been delivered to Nettskjema. The scheduling can run with a random interval, after at least a certain number of submissions have been delivered to the anonymous front-end server, or both.

Encrypted submissions should be stored in memory while queued for batch delivery. Persistence on disk may reveal the time of delivery in file system logs, and the encrypted submission may be retrievable even after it have been delivered and deleted from the frontend server. However, by never persisting submissions before batch delivery, there is an increased risk of submissions not being delivered in the case of a server reboot or application failure. As such, the solution can only guarantee a best-effort on delivery, and at should be assumed that some submissions might get lost.

4.3.6 Security of the client

It is to some extent assumed that the respondents’ operating system and browser is not compromised. A malicious third party with access to a compromised client, will in the worst case have access to read the submissions before they are encrypted, or log input from the mouse and keyboard.

It might be beneficial to inform the respondent of best practices related to the security of their client through dedicated help pages in the application, which includes to use an up-to-date browser and a recent operating system with the latest patches installed. The help pages may also include information on how to use the [TOR] browser, connect through a [VPN] or how to boot an operating system from a Live-CD to ensure the client is free from malware.

Most operating systems are set up to automatically install new patches and upgrades by default, and most modern browsers are evergreen, meaning they automatically update themselves.

4.3.7 Enforcing anonymous surveys

The solution should ensure that no identifiable information is collected through an anonymous survey. As described above, this information includes both questions in surveys, and metadata collected by Nettskjema.
Special question fields, such as ID number fields or e-mail fields should not be enabled for anonymous surveys. The option to add such fields should be disabled in the survey builder of Nettskjema. In addition, rendering of such elements should not be supported in the anonymous front-end solution to prevent such data from accidentally being collected if misconfiguration in the Nettskjema back-end occurs.

The Nettskjema back-end should also enforce that surveys marked as anonymous are always served through the anonymous front-end. All surveys in Nettskjema have a unique ID, and all open surveys are accessible through the nettskjema.uio.no/answer/{id}.html endpoint. Anonymous surveys should not be directly accessible from Nettskjema, and the back-end should either re-direct to the anonymous front-end, or show a 403 - forbidden message to the respondent, with information on how this survey can be reached.

As the anonymous front-end solution will be developed as a stand-alone web application, all assets required to serve the surveys can be served through the front-end. In addition, a strict Content Security Policy (CSP) should be set, only allowing content served from the front-end, to prevent other host to accidentally or deliberately track and compromise respondents anonymity. Ideally, the CSP should only allow connections to the anonymous front-end, and all static assets such as pictures, sound or video should be uploaded to the front-end by Nettskjema upon creation of the survey.

4.3.8 Problems with implementation

The solution above provides strong guarantees to the respondent that there will be no direct link between their submission and their identity. There are also mechanisms in place to ensure that a malicious insider or third-party at a later point in time will not be able to re-identify a respondent.

The solution builds on some of the same principles as the anonymous proxy solution, and may share some of the same vulnerabilities as described earlier. An attacker with access to the front-end server may in the worst case be able to listen on all requests made to the server, and retrieve metadata about respondents. However, the submissions will be encrypted, and the confidentiality of the respondents submission will be preserved. To mitigate the risk of exposure of the respondents IP-address, the respondents should be encouraged to use privacy preserving technology such as TOR or VPN when posting submissions to sensitive surveys.

However, the solution does not protect a respondent from an attacker who have compromised their client. A compromised Operating System or browser may allow an attacker to read the submission in clear-text before it is encrypted, and render the following anonymization process obsolete.

The described solution can only provide a best-effort on delivery of submissions to Nettskjema. While this may be acceptable for large scale studies and course evaluations, some use-cases may require a guarantee that all sub-
mitted data will be delivered to Nettskjema. The best-effort delivery is a part of the design, and while it can not guarantee that all submissions will be delivered to the Nettskjema backend, the solution gives strong guarantees that the confidentiality of the respondent and their submissions will not be breached.

**Validation of submissions**

In the regular version of Nettskjema, it is possible for survey administrators to make questions mandatory, or require validation on some fields. For instance, a survey administrator may ask the question *How many ice creams do you eat in a month?*, set the question as mandatory, and only allow positive integers to be entered into the numeric question field.

This validation can be done by the client, but Nettskjema will also validate such questions when receiving a submission to the backend to ensure that only valid submissions will be stored in the database. If a submission is invalid, Nettskjema can return an error code and the respondent will be able to correct the submission and send a new delivery.

When submissions are encrypted before delivery to the front-end server, Nettskjema will have to rely on front-end validation of submissions before they are submitted. A validation in the back-end can still be done, but it will be impossible to return an error code or other feedback to the respondent. If a submission is invalid, it should either be dropped by Nettskjema, or persisted in a different table for *invalid submissions* if they are still of interest to the survey administrator.

**Denial of Service**

It may also be possible for a malicious respondent to encrypt and post invalid data to the front-end server, as the server is not able to verify if the content is valid or not. Invalid submissions will be stored in memory on the front-end server until they are delivered in batches to the Nettskjema backend.

Flooding the front-end server with invalid encrypted submissions may result in a Denial of Service attack if enough submissions are stored to drain the free memory, or trigger repeated batch deliveries, resulting in increased load of the Nettskjema backend application.

A Denial of Service attack will not affect the confidentiality of respondents and their submitted responses, as the content will be encrypted. However, it may result in loss of submissions.

Since the front-end server is configured to not log IP-addresses, and monitoring of traffic is turned off at the dedicated network, it will not be possible to keep track of the number of requests originating from each IP and drop or block the requests if they reach a threshold value.
Instead, it might be beneficial to limit the maximum size of encrypted content that will be accepted on each request. A submission usually contains a few kilobytes of JSON data, plus metadata and attachments. By disabling attachments for anonymous surveys, a maximum size limit can be set relatively low.

If the risk of DOS attacks becomes a problem in the future, the front-end server may be configured to maintain a salted hash table of respondents’ IP-addresses upon delivery. The salted hash table requires the IP-address to be salted with a unique salt for each survey, to allow respondents to submit responses to multiple surveys.

The use of a salted hash table of respondents’ IP-addresses will not prevent an attacker from changing their IP-address to submit multiple responses. However, it may reduce the impact of automated massive denial of service attacks.

Further, the salted hash table can be kept in memory and flushed, based on a sliding time window with a relatively narrow time window, to prevent accidental exposure of respondents’ IP-addresses. The value space for IPv4 addresses is quite limited, and if the salt is known to an attacker it may be computationally feasible to compute salted hash values and compare those with values in the hash table to re-identify respondents.

Since the introduction of a salted hash table used to prevent multiple submissions from each IP-address may increase the risk of re-identification of respondents, the benefits of the hash table should be weighted against the cost of possible re-identification.

4.4 Comparison and evaluation of solutions

The proposed anonymous front-end application builds on the idea of using a proxy to remove identifying attributes from the data being collected. By encrypting the submissions in the client before they are posted to the application, shuffling and delivering submissions in batches, the assurance of anonymity is increased, and the risk of re-identification is significantly lowered compared to the initial proxy solution.

The anonymous front-end application addresses the TAAL framework by removing and discarding identifying metadata before it is processed. By disabling logging in the application and on the network, and increasing the timeframe for when a submission were delivered, the anonymous front-end application may be assigned a TAAL 4.

However, some security measures against denial of service attacks, and requirements to be cloud ready may reduce the assurance to level 3.
Chapter 5

Anonymous authentication of respondents in Nettskjema

Anonymous authentication of respondents is based on the idea that only an authenticated respondent should be able to post a submission to a survey, without there being a connection or link between the respondent and their submission. By restricting who is allowed to respond, researchers may get better control over the data submitted and restrict the respondents to specific demographic groups. The authentication process also prevents malicious entities from flooding a survey with random data, which could decrease the data quality.

While anonymous authentication aims to remove the link between a respondent and their submission, it should be noted that the survey administrator will still be handling and collecting personal identifiable information. To send out invitations, the survey administrator will need the e-mail address, or some other way to directly contact the respondents. A survey used for course evaluations will need a list of students attending the course, to make sure only students who have enrolled in the course are allowed to submit their feedback.

Depending on the nature of the survey, being invited may be sensitive personal information in itself. For instance, an invitation to participate in an evaluation survey on a new antiviral drug may imply that the respondent has a chronic medical condition.

As such, a solution to provide anonymous authenticated surveys may only seek to remove the connection between a respondent and their submission. Survey administrators may still have to seek permission to run the study from NSD, REK or the Norwegian Data Protection Authority, since they collect and process personal information, such as the respondents e-mails. However, based on the findings from the interviews, it is believed that providing an anonymous solution for authenticated surveys may improve the response rate and quality of submissions.
5.0.1 Use cases

Two use-cases will be used when discussing requirements and implementations of proposed solutions.

The first use-case is course evaluation surveys. Course evaluations may provide useful feedback to lecturers and course administrators, and provide an easy way for students to share their thoughts on the course. Course evaluations at the Department of Informatics, at the University of Oslo have been carried out using Nettskjema in recent years. The respondents are invited by their UiO username or e-mail. The surveys are set up to not store personal information about respondents, however, as discussed in earlier chapters, the current implementation of Nettskjema still persists and processes some information about respondents, such as whether they have posted a submission or not. While this information is not available to the survey administrator, the data exists in the database.

The second use-case is a traditional survey, where participants are invited by e-mail. Most or all of the invited respondents are external participants outside of the UiO domain, as such their e-mail is the main identifier.

5.1 Requirements

A solution for anonymously authenticated surveys should build on the work done in chapter 4 which proposes a solution for open anonymous surveys. By extending or building upon an existing solution for anonymous open surveys, a solution for anonymous authenticated surveys may draw from the benefits of submissions already being encrypted in the browser, and the knowledge that no personal identifiable information will be collected by the application.

The survey application should ensure that only valid respondents are allowed to post a submission. However, the respondents’ real identity should not be processed or persisted as part of the data collection.

5.1.1 Optional features

Some features are common in online survey applications. While they may help survey administrators increase their response rate, or improve the quality of the data gathered, they might provide an increased risk of re-identification for respondents. Such features include the ability for survey administrators to restrict the number of submissions for each respondent, and the ability to remind respondents who have yet not posted a submission that their contribution is highly valued.

Depending on the implementation, those features may pose a risk of re-identification of the respondents. As such, they are treated as optional features, which may be implemented if possible.
Limit number of submissions

Some use-cases may require that a respondent is only allowed to post a limited number of submissions to a survey. The ability to restrict the number of submissions from a respondent may give survey administrators and researchers better control over the data collected, and prevents authenticated malicious entities to flood the survey with random data.

A variant of this feature is to allow at most one submission for each respondent. The survey application can be configured to only accept the first submission, or to always accept the latest submission and discard the previously submitted by a respondent. Such a scheme may be seen in electronic voting, where each entities vote should be counted only once.

Reminders to respondents

It is often desirable for survey administrators to send out reminders to respondents who have yet not posted a submission to a survey, to increase the response rate of the survey. To send out individual reminders, it is necessary for the survey application to know which respondents have posted a submission and not.

Keeping a record on whether a respondent has posted a submission or not, may increase the risk of re-identification, since metadata about the respondent must be updated upon successful delivery of a submission. Entities with direct access to the persisted metadata may be able to monitor changes, and compare the time of change in delivery status for a respondent with the time of delivery for a anonymous submission to the survey.

An alternative approach is to allow survey administrators to send out a reminder to all invited respondents if the response rate is low, and optionally provide a way for individuals to unsubscribe from further notifications on a given survey. While unsubscribing to reminders may imply that a respondent have already posted a submission, it may also mean that the respondent do not want to participate in the study.

5.2 Anonize

The Anonize system is presented in the paper An Overview of ANONIZE: A Large-Scale Anonymous Survey System written by Hohenberger, Myers, Pass and shelat[19]. The paper presents a system for creating anonymously authenticated ad-hoc surveys.

The system consists of three parties, a single registration authority (RA), one or more survey authorities (SA) and users who submit data to the survey application. The RA is responsible for user registration and issuance of master tokens, while the SA creates surveys and survey identities. The master token and survey id is both required to produce a unique token that
allows a participant to submit a response. The uniquely generated token satisfies two properties. It is unlinkable, e.g. anonymous, and it allows each participant to post at most one submission.

The paper presents three use-cases where Anonize may be utilized, which is course evaluation, product reviews and whistleblowing within an organization. As the three use-cases are similar in terms of registration, authentication and submission of survey data, only the use-case of course evaluation will be discussed below.

The full cryptographic scheme and proofs is presented in the paper ANONIZE: A Large-Scale Anonymous Survey System[5], and will only be discussed briefly.

5.2.1 Registration
Anonize requires participants to register with the RA before they can participate in a survey. The registration process only needs to be done once, and should ideally be done ahead of participation in a survey. During the registration process, a master token is issued, which may be used in the authentication process of all surveys created by any survey authority.

In the use-case of course evaluation, students may be registered with the RA when they get admitted to a university.

5.2.2 Survey submission
During the submission process, the respondent presents a non-interactive zero knowledge (NIZK) argument providing proof of knowledge of a master token with the RA and the survey id. The survey data is submitted as a part of the NIZK argument.

Once submitted, the responses must be verified. The experiments conducted in An Overview of ANONIZE: A Large-Scale Anonymous Survey System[19] shows that the verification process may be computationally intensive, as a single machine would be able to verify 1 million submissions in approximately 33 hours. However, the verification process can be run in parallel on several computing nodes, which can increase the overall capacity.

5.2.3 Metadata and monitoring of participants
While the Anonize scheme allows participants to anonymously authenticate when posting a submission, the paper describing Anonize does not discuss how metadata such as IP-addresses and timestamps are handled by the registration authority and survey authorities. However, the papers suggest that respondents should use privacy enhancing technologies such as TOR to preserve their anonymity.

The benefits of privacy enhancing technologies to masquerade participants’ IP-address may be lower in situations where a malicious entity can
gain access to network logs for both the participants’ client, the registration authority and survey authority. Larger organizations, such as universities, may wish to monitor the network traffic on both clients and servers connected to the network. While traffic routed through TOR is encrypted, timestamps related to TOR traffic on client machines and survey authorities may be correlated to identify the machine used to post a submission.

It may be beneficial to deploy the RA and SA in the same environment as the previously discussed anonymous front-end server to prevent accidental logging or monitoring of IP-addresses.

5.2.4 Applicability to traditional surveys

The Anonize scheme requires participants to register with the RA before submitting a survey. Ideally, participants should register ahead of time before the survey is opened for submissions.

The process of registration may add a significant overhead for respondents compared to authentication based on tokens or other credentials generated by the survey application. Depending on the nature of the survey, it is the author’s belief that some respondents may consider the registration effort to be too much work and drop out of the survey.

5.3 Identity Mixer

The IBM Identity Mixer uses attribute-based credentials to authenticate an entity. Through Identity Mixer, an entity may prove to verifier that they are in possession of a set of attributes, without revealing the attributes themselves.

The simplest scheme of attribute-based credentials involves three parties, namely the user who collects credentials from issuers and controls which credentials to present to a verifier, the issuer who issues credentials and the verifier who protects access to a resource or service.

One of the use-cases presented by IBM Research is the requirement of a valid subscription and a minimum age to access a specific website. If a person, Alice, were to access the website, she could provide a valid subscription card and a copy of her ID card stating her date of birth to prove she is allowed to enter the website. By presenting her ID card, Alice could possibly reveal more information about herself, such as when she was born, her real name and national ID number. With Identity Mixer, Alice would be able to prove possession of two attributes, namely that she has a valid subscription, and is above the legal age to enter, without revealing other attributes such as her name or national ID number.

To authenticate to a service, the user will first obtain a presentation policy describing which credentials are required, and which attributes she must reveal. Based on the required credentials, the user may then derive a
presentation token. The presentation token satisfies the properties of being unlinkable to a user, and untraceable. In addition, the scheme allows to request and reveal predicates over attributes, such as being born before a given date, without revealing the persons date of birth. The token is then presented to the verifier, who is able to grant or deny access to a resource based on the information provided in the token.

5.3.1 Integration with Nettskjema

An attribute-based credential scheme, such as Identity Mixer, can be integrated with the use-case application Nettskjema, and the anonymous front-end application discussed in chapter 4.

By considering Nettskjema to be a trusted issuer, the Nettskjema application can issue credentials to invited respondents containing an attribute stating the respondent is invited to submit a response to a specific survey. The anonymous front-end application may then act as the verifier, requesting respondents to provide a token containing the attribute of being invited to the survey. A simple overview can be seen in figure 5.1.

![Figure 5.1: Registration and submission phase with IBM IdentityMixer](image)

The attribute-based credential scheme can either be implemented through the IBM Bluemix cloud available at [https://console.bluemix.net](https://console.bluemix.net) or through the p2abcengine available at [https://github.com/p2abcengine/p2abcengine](https://github.com/p2abcengine/p2abcengine) using Identity Mixer to generate the cryptographic values used in authentication tokens.

Integrating Nettskjema with the attribute-based credential scheme requires integration with, or development of a trusted application used to manage credentials and creating presentation tokens on the respondents’ behalf.
5.4 Ad-hoc invitation through an anonymous authentication server

Authentication of invited respondents may be done through a pseudo-anonymous authentication server. The server may not be truly anonymous, as it will have to process and persist some attributes that can be used to directly or indirectly identify a respondent. Such attributes may be the respondent’s username, e-mail or a unique invitation token, and whether the person has already submitted a response or not. As some identifying attributes are processed and persisted by the authentication server, there are restrictions to the maximum TAAL which may be achieved.

The example solution discussed below will use OAuth 2 for the underlying access protocol, but other protocols, such as SAML may be used as well. The goal is to provide the survey application with an assurance that the respondent is allowed to post a submission to the given survey.

5.4.1 General concept

The authentication server allows respondents to be registered beforehand by a survey administrator through the use-case application Nettskjema. The registration may include a unique identifier, such as a token, the formId and optionally, how many times the respondent may post a submission. The token is used to identify the respondent in the authentication process. The formId acts as the scope, indicating which resources the respondent may access.

If the registration of respondents is successful, invitations may be sent out to the respondents in batches, either by SMS or E-mail. The invitation will include a link to the anonymous survey, and the previously generated token.

As a pre-requisite, the survey has been registered in the anonymous survey application, with a flag indicating that the survey is a protected resource. When the survey is accessed, respondents will be redirected to the authentication server, where they are asked to provide their unique invitation token.

If the invitation token exists in the authentication servers database, matches the scope required by the survey application, and optionally, the number of posted submissions is less than the maximum allowed, a new access token is generated and signed by the authentication server. The respondent will then be redirected to the anonymous survey application, which checks if the access token is valid and grants the respondent access to post a submission.

OAuth requires a client identifier to be set by the authentication server, and various metadata may be provided by the scope selected. As the client identifier is not needed by the anonymous survey application, it may be set to a static value. Additionally, no metadata should be provided by the authentication server.
5.4.2 Location and configuration of authentication server

Where the authentication server runs, and the configuration of the server depends on the desired TAAL to be achieved. The server itself should not export any metadata about respondents to third parties.

The TAAL 3 allows identifying attributes to be processed by trusted third parties, given that the system originally collecting and processing the data otherwise will be TAAL 3 compliant.

The anonymous front-end server discussed in the previous chapter will not collect or persist identifiable information about respondents. If the authentication server can be deployed by a trusted third party, and integrated with the anonymous front-end, it may be possible to achieve a higher TAAL.

5.4.3 Registration of subjects

Invited respondents must be registered with the authentication server before the survey starts, and there should be set a minimum limit on number of invited participants before the survey can be opened.

Allowing respondents to be registered after a survey is opened for submissions, or inviting a small number of participants may enable a malicious survey administrator to target specific participants and re-identify their submissions at a later time.

Restriction on registration of new subjects

Registering new respondents as participants in a survey after the survey has started may leak information on which submission that belongs to a respondent. For instance, given that a survey administrator invites 10 respondents to participate in a survey and then invites another participant two weeks later, she might be able to guess, with a certain degree of confidence, which submission that belongs to the last invited participant by looking at the time of delivery.

The underlying infrastructure providing anonymous submissions to Nett-skjema addresses some of those concerns, by caching submissions for a certain amount of time. By caching encrypted submissions from invited respondents for a given period of time, before they are dumped in batches to Nettskjema, the chance of respondents being re-identified is lowered. However, there is a trade-off between the amount of time to cache, and the time it is acceptable for a survey administrator to wait until all submissions have been delivered to Nettskjema.

In addition to relying on the underlying infrastructure, an implementation of the authentication server should put restrictions on registration of participants. The restrictions may be based on the number of participants to be registered and the last time a participant were registered to participate in the survey. Eventually, the authentication server may be configured to
Table 5.1: Properties of the identifying attributes

<table>
<thead>
<tr>
<th></th>
<th>Directly identifiable</th>
<th>Should use secret credentials to authenticate</th>
</tr>
</thead>
<tbody>
<tr>
<td>E-mail</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Token</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

not accept new registrations to a survey once the survey have opened and accepted submissions.

Usage of identifying attributes during registration

The responsibility of the authentication server is to identify and authenticate respondents, and verify that they have access to post a submission to a survey. To be able to identify a respondent, the authentication server will need to persist and process a unique identifying attribute of the respondent. Such an attribute may be the respondents e-mail, username, or a unique token. Additionally, the respondent may be asked to provide some sort of pin-code or password to prove their identity.

E-mails and usernames may uniquely identify a respondent and their relation to a survey. As discussed earlier, the fact that a person is invited to participate in a study may be sensitive information in itself. However, this may be acceptable in many cases, such as surveys for course evaluation where the list of participants may already be published or easily accessible.

If the respondents e-mail address is used as an identifying attribute, a password or PIN-code should be generated and persisted together with the identifying attribute to ensure only invited respondents with knowledge of their credentials can post a submission. Credentials can be created automatically and sent to the respondent together with the invitation to participate in the survey.

Unique tokens may be used as a substitute to e-mails or usernames. A token will not provide a way to directly identify a respondent, as long as the token is not persisted together with other identifying attributes. The use of a token may also to some extent reduce the need for a password or pin-code to authenticate a respondent. Random, unique tokens also have the property of not being a directly identifiable attribute, which may reduce the probability of re-identification, should a malicious entity gain access to the authentication server.

Persistance of respondent metadata

Persistance of metadata in the authentication server may reveal sensitive information about a respondent. By default, the authentication server should not log or persist information about when a respondent last authenticated,
or failed to provide valid credentials.

As with the submissions posted to the anonymous front-end, there is a risk of re-identification if those records are persisted to disk. To prevent accidental leakage of timestamps that can be used to re-identify respondents and cross-reference authentication with a submission, such a record may only be kept in memory.

Records kept in memory will be deleted if the authentication server is restarted or crashes. As such, the authentication server will not be able to provide any assurance that respondents is only allowed to submit a maximum number of invitations. However, it will prevent most respondents from accidentally submitting twice, and large scale automated attacks against an authenticated survey where attackers have gained valid credentials.

### 5.4.4 Limiting the number of submissions for each respondent

A limit to the number of submissions a respondent is allowed to submit can be implemented by setting the number upon registration with the authentication server. When a respondent posts a submission, the authentication server is notified and it increases a counter for the respondent.

### 5.4.5 Weaknesses of implementation

The security and ability to keep respondents anonymous largely depends on the treatment of credentials and metadata, both by the authentication server and by applications interacting with it. Neither the authentication server or Nettskjema should keep records on the link between a token and a respondent.

While the IdP may be deployed in a secure network, and be configured to not log any data about respondents, it will still be considered a trusted third party.

### 5.5 Blind Signatures

Blind signatures may be used to authenticate respondents in the use-case application Nettskjema. When an invitation is sent out, the respondent blinds their invitation and send it back to the Nettskjema application for signing. The respondent will then keep the blinding factor and their signed invitation locally until they post a submission.

When the respondent is ready to post a submission, they remove the blinding factor from the signed invitation and post it together with the submission. The Nettskjema application can then verify that the invitation
is valid and further process the submission. A graphic representation of the registration and submission phases can be seen in figure 5.2.

Figure 5.2: Registration and submission phase with blind signatures

5.5.1 Registration phase

During the registration phase, the respondent will get an invitation to participate in a survey. The respondent then applies a blinding factor to the invitation and submits it to the Nettskjema application for signing. After the signing process, the respondent needs to keep both the blinding factor and the signed invitation until the survey opens and they can post a submission.

How the respondents store and manage the blinding factor and signed invitation after the registration phase is not relevant for the blind signature scheme to work, as long as the blinding factor and signed invitation is kept secret and not accessible to third parties. For instance, the registration phase may be processed with JavaScript in a webbrowser and downloaded as a text file once the registration is completed.

However, forcing respondents to manually manage secrets may add complexity and overhead to the registration process, as well as greatly reduce the usability of the survey application. To enhance usability and reduce the overhead, secrets may be processed and persisted in the webbrowsers local storage, or with a special purpose mobile application.

While the blinding factor provides untraceability to an invitation, there is a risk of correlation between submissions and invitations if the respondents are allowed to register and sign their invitations right before they post a submission. As such, the registration phase and submission phase should be separate. Ideally, respondents should be invited before the survey opens,
Table 5.2: A comparison between the various authentication mechanisms

<table>
<thead>
<tr>
<th></th>
<th>Registration</th>
<th>Key management</th>
<th>3rd party trust</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anonize</td>
<td>Yes</td>
<td>Yes</td>
<td>Partially</td>
</tr>
<tr>
<td>IdentityMixer</td>
<td>Yes</td>
<td>Yes</td>
<td>Partially</td>
</tr>
<tr>
<td>Auth Server</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Blind Signatures</td>
<td>Yes</td>
<td>Yes</td>
<td>Partially</td>
</tr>
</tbody>
</table>

and no more respondents should be invited once the survey is opened for submissions.

5.5.2 Submission phase

When the survey is opened for submissions, the respondent may fill out the survey as usual. To assure the respondent of their anonymity, submissions can be delivered to the anonymous front-end proposed in chapter 4. When the submission is completed and ready for delivery, the respondent removes the blinding factor from the signed invitation and attaches it to the submission.

The application processing the submissions may then verify the attached signature and check if the respondent is indeed invited. If the verification process is successful, the submission is persisted in the Nettskjema database, if not, it may simply be discarded.

5.6 Comparision between solutions

The proposed solutions discussed above provides various ways of enforcing and maintaining the anonymity of respondents. An overview of the properties of the proposed solutions can be seen in table 5.2.

Anonize, IdentityMixer and the use of blind signatures all require the respondent to register or interact with Nettskjema before they are able to post a submission. IdentityMixer requires an issuer that is trusted to interact with Nettskjema and verify that the respondent is invited to a survey during the registration phase. Anonize only requires interaction with a trusted third party once to obtain a master token, while the use of blind signatures require the respondent to interact directly with Nettskjema to get a signed, blinded token that verifies that they are invited to a survey.

Anonize only requires the registration process to be completed before they submit a response, and allows registrations before the survey is created. IdentityMixer or blind signatures requires that the respondent registers after the survey is created and before the survey is opened for submissions.

In addition, Anonize and the use of blind signatures requires the respondent to maintain cryptographic keys or secrets on their own devices,
Table 5.3: Assurance levels of anonymity provided

<table>
<thead>
<tr>
<th>Method</th>
<th>TAAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anonize</td>
<td>4</td>
</tr>
<tr>
<td>Identity Mixer</td>
<td>3-4</td>
</tr>
<tr>
<td>Auth server</td>
<td>3</td>
</tr>
<tr>
<td>Blind signatures</td>
<td>4</td>
</tr>
</tbody>
</table>

e.g. in a mobile application, which can be used in the verification process.

The anonymous authentication server is the only proposed solution that does not require respondents to register beforehand, or maintain keys or secrets locally on their own devices. However, the authentication server is considered as a trusted third party, which processes directly or indirectly identifying attributes on behalf of Nettskjema. Respondents need to trust that the authentication server is deployed and configured in a way that prevents accidental leakage of metadata, such as time of authentication or time of last posted submission. A malicious entity with access to the authentication server may be able to capture information which may be used to re-identify respondents at a later point of time.

5.6.1 Maximum achievable TAAL

The proposed solutions may be evaluated against the previously discussed Technical Anonymity Assurance Level (TAAL) framework.

The higher levels of the TAAL framework has strict requirements related to processing and collection of identifying attributes. The TAAL 2 requires the system being evaluated to ensure the collected data can not be cross-referenced with other third parties to re-identify entities in the collected data set. In addition, the TAAL 4 requires that all data processing is done inside a trusted, local domain. Hence, the system being evaluated should not be allowed to interact with third parties, even though the third party is trusted through a data processing agreement or policy.

An overview of the comparison between the various proposed solutions and their associated TAAL can be seen in table 5.3. A brief discussion on each proposed solution follows below.

Anonize

The protocol behind Anonize requires an RA and SA to register respondents and managing surveys. While the RA and SA can be deployed anywhere, it is assumed they will be deployed inside the trusted local domain of Nettskjema.

The RA will collect and process directly identifiable information, such as respondents e-mail address. However, the link between an e-mail and a respondent is obfuscated and protected through a non-interactive proof.
of knowledge, where the respondent proves being invited without revealing their master token or e-mail.

Assuming the Anonize protocol is deployed and integrated in the trusted local domain, and utilizing the benefits of the anonymous front-end application as discussed earlier, the maximum achievable TAAL is level 4.

Identity Mixer

The framework for attribute-based credentials using Identity Mixer may be set up and configured in two different ways. Integration with Identity Mixer may be run in the cloud, using IBM Bluemix cloud where Identity Mixer is provided as a service. Identity Mixer may also be integrated in a standalone application, through the \textit{p2abcengine} as discussed above.

If attribute-based credentials are integrated with IBM Bluemix cloud, there will be a reliance of a trusted third party. However, the application itself should not be able to obtain information that can re-identify the respondent. It is further assumed that the further processing of the collected data will take place in an environment where data is shuffled and cached to further decrease the risk of re-identification. As such, the maximum achievable TAAL would be level 3.

If attribute-based credentials are integrated using the \textit{p2abcengine}, it may be possible to process all data within the same trusted environment. As such, the maximum achievable TAAL would be level 4.

Authentication server

The anonymous authentication server does not place any restrictions on what kind of identifying attributes that may be used to identify a respondent. For instance, it may be possible to allow respondents to authenticate both with their e-mail and a pin-code, and with a unique invitation token.

The authentication server also needs to be configured in a way that prevents it from logging or persisting additional metadata about entities authenticating with the server.

It is assumed that the authentication server will be deployed by a trusted third party, as defined in TAAL 3. By outsourcing the process of authenticating, systems who are otherwise TAAL 3 compliant can integrate with the authentication server while maintaining their TAAL.

Blind signatures

The blind signature scheme does not require any identifiable attributes to be processed or stored by the use-case application Nettskjema. While the scheme requires a respondent to interact directly with Nettskjema in the signing process, it only shows an intent to deliver a submission at a later
time. During the submission process, an unblinded signature will be de-
levered, which proves the respondent is indeed invited. However, the signa-
ture is untraceable to the respondent.

Since no information about the respondent is collected through the sub-
mission process, and all interaction is done with services in a trusted local
domain, the blind signature scheme can be assigned a TAAL 4.
Chapter 6

Discussion

6.1 In-memory caching and risk of data loss

Many of the techniques used to preserve the anonymity of the respondents relies on in-memory caching. The in-memory caching is used to prevent timestamps and ordering in relational databases to leak information about when a submission was posted and who the respondents are. However, the caching mechanisms relies on enough data being submitted before the cache is persisted to disk or batch processed. By only caching data in memory, the applications discussed are also at risk of losing data should the application crash or be unexpectedly terminated or restarted.

The risk of losing submissions is a trade-off between the enhanced privacy and preservation of anonymity provided by the discussed solutions above.

The advantage, or benefit, of caching can be seen as a function of the number of submissions per time-unit multiplied with the number of time-units the submissions are cached. Given that an average of 100 submissions is posted to the anonymous server every hour, the probability of re-identifying a respondent would decrease if the period of caching were extended from 24 hours to 48 hours.

Table 6.1: Explanation of variables

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AS</td>
<td>Average submissions each time unit</td>
</tr>
<tr>
<td>h</td>
<td>Time-units of caching</td>
</tr>
<tr>
<td>PI</td>
<td>Probability of re-identification</td>
</tr>
<tr>
<td>PD</td>
<td>Probability of data loss each time unit</td>
</tr>
<tr>
<td>t</td>
<td>Total probability of data loss</td>
</tr>
<tr>
<td>AD</td>
<td>Average data loss</td>
</tr>
</tbody>
</table>
\[ PI = \frac{1}{(AS \times h)} \] (6.1)

The function above assumes a malicious entity with access to external logs, such as logs of the IP addresses connected to the server in a given time-frame, who tries to guess which submission belongs to a specific respondent. Given a wider timeframe or a higher number of average submissions each hour, it will be harder for the malicious entity to guess which submission belongs to a specific respondent.

In the same manner, the probability of data loss can be seen as a function where the chance of unexpected termination per time-unit is multiplied with the number of time-units the submissions are cached. Given that the application would unexpectedly restart or terminate once a month, and the hours of caching is set to 24 hours, the total probability of data loss would be 0.03, or 3%. By increasing the hours of caching to 48 hours, the total probability of data loss would increase to 0.06, or 6%.

\[ t = PD \times h \] (6.2)

Since the number of submissions increases for each time-unit, the average loss for each failure will be higher if the length of caching is increased. Assuming a probability of data loss of 0.3%, an average of 100 submissions an hour, and a caching period of 24 hours, the average number of submissions lost will be \(0.03 \times \frac{(100 \times 24)}{2} = 36\).

\[ AD = t \times \left(\frac{AS \times h}{2}\right) \] (6.3)

Determination of the number of time-units, e.g. hours, to cache, requires knowledge of the average number of submissions that are submitted within a time-unit and the probability of data loss related to unexpected application termination or application crashes.

By comparing the graph provided by the function for probability of re-identification, and the graph from the function of average data loss for each failure, it is possible to determine an acceptable value for the number of time-units the application should cache data before it is submitted for batch processing.

Determining an acceptable value for probability of re-identification or probability of data loss is out of scope for this thesis. The stability of the application server may vary depending on where it is deployed, and requirements to what may be considered an acceptable loss of submissions may depend on the organization responsible for the data collection.
6.2 Security of browsers / clients

The respondent’s client or web browser interacting with the survey application discussed in previous chapter has been partially out of scope for this thesis. While endpoints interacting with the application are a critical component in the process of anonymous data gathering, the focus has been on improving and extending the survey application in areas the developers and system administrators can control.

The confidentiality of the data submitted by a respondent is protected by encryption of the data while in transit and during caching on the web-servers handling submissions. However, the respondent is not protected against an attacker with access to their device. An attacker with physical or remote access may be able to circumvent protection mechanisms in the browser, or read data directly from the keyboard.

General security advice applies equally well to protection of the client. By keeping their device and installed software up-to-date, and not install software from untrusted sources, respondents may reduce the attack surface on their client.

The author believes that securing the end-points interacting with the survey application is important, and recognizes that a vulnerable client may leave the respondent exposed to targeted attacks.

6.3 Communication channels for invitations

Some of the proposed solutions for anonymous authentication requires invitation tokens or other information to be sent to respondents. We believe that e-mail or SMS is the most practical communication channels for such information. However, neither e-mail or SMS is encrypted by default. As such, it may be possible for a malacious entity to obtain invitations or credentials for a survey.

Exposing the name or purpose of a survey in an e-mail title or body may reveal sensitive information about the invited respondent. For instance, the title *Follow-up study on kidney transplant patients* clearly indicate that the invited respondent might have a medical condition. Instead, we suggest that invitation title and message is as generic as possible, such as the title *Invitation to participate in study*.

In cases where invitations are sent over un-encrypted e-mail or SMS, there is a chance of an attacker being able to impersonate the respondent by stealing their invitation token or credentials. While such an attack may compromise the integrity of the study, since a malicious entity, Malory, is able to post a submission when she is not invited, we do not believe it can compromise the confidentiality or anonymity of the originally invited respondent.
6.4 Response rate and data quality of anonymous responses

The interviews conducted as part of this thesis suggest that the ability to post submissions anonymously can improve the data quality and response rates of surveys and questionnaires covering sensitive topics. Such topics may include the use of alcohol and drugs, physical and mental health, and sexual activities. In addition to being anonymous, several respondents mentioned other factors which could impact their willingness complete the surveys. Such factors would be the purpose of the study, who would have access to the data, and their general trust in the organization or individuals responsible for the study.

The paper *Response Rates to Mail Surveys Published in Medical Journals* [27] published by Asch, Jedrziewski and Christakis in 1997, evaluates factors that influence the response rate of traditional mail surveys. The paper suggests that the anonymity factor has no significant impact on the response rate of traditional mail surveys. We question this finding, as our own study indicates that respondents may feel reluctant to participate in surveys covering sensitive topics if their identity is known.

It is the authors belief that the increased use of online surveys compared to traditional mail surveys might influence how respondents feel about entering sensitive data. However, further research might be needed on how anonymity influences the response rate of online surveys.

Authentication was not discussed during the interviews. Participants were simply asked how they would answer various questions from the example surveys given that their name were either hidden or known to the survey administrators. As such, it is possible the participants would be reluctant to post a submission if they were asked to log in with a username or token, even though the application told them they were anonymous.

As discussed in section 6.1, many of the techniques used to preserve respondents anonymity increases the risk of loss of submissions. While some loss may be considered acceptable, loss of submissions will impact the overall response rates to a survey. As such, the response rate should also be a deciding factor when considering an anonymous survey as an alternative to a traditional survey, which provides much higher assurance of data delivery.

6.5 Trust in the application code and configuration

For the anonymity preserving techniques discussed in previous chapter to be useful, respondents need to trust that the proposed techniques are actually
implemented in the running software. The author suggests that a proper solution, if implemented, will be open sourced and publically available for review.

In addition, respondents need to trust that the running software is the same as the published source code. Developers, system owners, malicious third parties and government agencies may have incentives to add or inject malicious code to the production environment.

Open Whisper Systems, the organization behind the encrypted messaging application Signal, recently proposed a solution for remote attestation of the server application through the use of the Intel [Software Guard Extentions](https://en.wikipedia.org/wiki/Intel_Software_Guard_Extentions) module in modern Intel chips. While the SGX module were initially developed for Digital rights management (DRM) enforcement on client machines, it may allow clients to verify that the software running on the server is not modified in an unauthorized way. If proved useful in the Signal application, it may be worthwhile to look into the possibility of implementing the same feature in an anonymous survey application.

It should also be possible for clients and external entities to do a sanity check of the system configuration, and verify that the system, including configuration of servers and network, is properly set. Verification of the integrity of the entire system configuration is out of scope for this thesis, but may prove useful to further improve participants trust in the system.

### 6.6 Security and complexity

All the proposed solutions and techniques used to enhance the assurance of anonymity of respondents may add complexity to the overall system architecture. By adding components and dependencies to a system, there is an increased need for maintenance of the total codebase.

Integration with new components may introduce errors or vulnerabilities, and reliance on third party dependencies may leave the system vulnerable for attack, should a vulnerability or exploit be discovered in the third party dependency. This is especially important for the ABC4Trust project, including IBM Identity Mixer, and Anonize, which haven’t been updated in several years.

While the proposed solution may be proven to provide a sufficient assurance of anonymity in their own context, we believe care should be taken while implementing or integrating with new components or dependencies used to enhance the assurance of anonymity, privacy or security of an application.
6.7 Usage of anonymous authentication

Several proposed solutions for anonymous authentication of respondents have been discussed. However, they either require the respondent to register beforehand, maintain and store keys or secrets on their own device, or trust in a third-party application to ensure their anonymity.

The requirement of registration beforehand may be reasonable in many use-cases, such as course evaluations where students already have valid credentials at their university. The registration process may then easily be integrated as part of account creation when students get admitted to the university, or when they enroll in a course.

The requirement of key management and other secrets, such as master tokens may also be reasonable in some use-cases. Keys and secrets can be managed through a mobile- or desktop application.

However, the use-case of traditional surveys may involve participants that apart from being participants in a study, may have no connection to the UiO domain or Nettskjema. As such, the requirement of pre-registration before they can post a submission in a survey may add significant overhead to the submission process and lower the usability experience. In the worst case, this may discourage invited participants to take part in the study, which may reduce the response rate.

The proposed anonymous authentication server does not require respondents to register or manage secrets locally. However, it requires respondents to trust that Nettskjema and the authentication server manages tokens and metadata in a way that will not compromise their anonymity.

6.8 Conclusion

In this thesis, we have attempted to research three questions. The first question was whether the factor of being anonymous have an impact on response rate and data quality in surveys collecting sensitive data. The second question was to investigate if a desirable level of assurance of anonymity can be achieved in a survey for unauthenticated respondents. The last question were to investigate if a desirable level of assurance of anonymity could be achieved for authenticated respondents.

Results from the interviews conducted as part of this thesis suggests that the factor of being anonymous may affect how respondents answer to sensitive questions in a survey, and if they will post a submission at all. Technical details on how a survey application would assure respondents of their anonymity were not discussed in the interviews. Instead, the respondents were asked to think about anonymity simply as there being no connection between
their identity and their submission.

In addition, several respondents said that the factor of trust could affect their willingness to participate in a study. Factors affecting the respondents’ trust included the purpose of a study, reputation and the level of transparency.

We believe that being able to communicate the level of assurance of anonymity provided by a survey application to respondents is important. Being open and transparent on how data is collected, and what data that will be processed by the survey application may improve the general trust. However, we believe that further research is needed to find out how to efficiently communicate this in a way that is easy to understand for respondents.

A proposed solution for anonymous unauthenticated submissions were discussed in chapter [3]. The proposed solution utilizes encryption, caching and shuffling of submissions in memory to remove identifying attributes and reduce the probability of re-identification of respondents.

We believe that the proposed solution can offer an acceptable assurance of anonymity, with a low probability of re-identification. However, the assurance of anonymity is a trade-off between low probability of re-identification and the risk of losing submissions. Submissions will only be kept in memory while cached, as such, they may be lost if the server crashes or restarts. We believe that some loss of submissions may be acceptable in most use-cases, given that the assurance of anonymity increases the total response rate more than the average number of submissions lost.

Several methods providing anonymous authenticated submissions were discussed in chapter [5]. Except for the anonymous ad-hoc authentication server, the discussed methods were able to provide a high level of assurance of anonymity. However, the high level of assurance came at a cost of requiring respondents to either register before the start of a survey, manage keys or secrets or relying on a trusted third party to manage their credentials.

We believe that while it is possible to achieve a high level of assurance of anonymity to authenticated respondents, the requirements of pre-registration and management of keys and secrets may reduce the incentive to participate in a survey. In addition, the reliance of third-parties to maintain credentials and other attributes related to a survey may further reduce respondents’ trust in the survey application.
Bibliography


