What users want: adapting qualitative research methods to security policy elicitation

Vivien M. Rooney and Simon N. Foley✉

IMT Atlantique, LabSTICC,
Université Bretagne Loire, Rennes, France
[vivien.rooney,simon.foley]@imt-atlantique.fr

Abstract. Recognising that the codes uncovered during a Grounded Theory analysis of semi-structured interview data can be interpreted as policy attributes, this paper describes how a Qualitative Research-based methodology can be extended to elicit Attribute Based Access Control style policies. In this methodology, user-participants are interviewed, and machine-learning is used to build a Bayesian Network based policy from the subsequent (Grounded Theory) analysis of the interview data.

1 Introduction

A challenge in eliciting security requirements is properly understanding the needs of the user. The requirements for security mechanisms are all too often determined by technical experts who interpret user needs through the lens of their own technical experience, and which may be at variance with the requirements of the end-user. While user-centered security [2, 33] can help to provide usable security mechanisms, the tendency is to focus more on ensuring that the mechanism provides a good user-experience, and not so much on discovering what the user, as a person, actually needs from security [12]. When a security mechanism does not match the user needs, the user either figures out some circuitous way to bypass the security control or else avoids using the system properly [3, 8].

The goal of security experts is ensuring security, and one of the many challenges is being able to encompass users' important personal values [12]. Furthermore, the elicitation of user needs remains a stumbling block in the security community. Capturing the subtleties of user requirements is challenging [18]. Eliciting needs and requirements from people is not a new problem. Sociologists and applied psychologists have spent decades attempting to understand people's experience and, more recently, their experience of technology.

Qualitative research methods have evolved as a means to systematically elicit the needs of users. Understanding an experience such as chronic illness requires a method that can delve into the subtle minutae of living with such a condition, and explore the meaning of that experience, for example, the significance of disclosure in the workplace [10]. The methods can be applied to technology, to find out, for example, about the perspective that mothers' have on mobile communication technology [26]. This knowledge provides us with new ways of

understanding, for instance, what the disclosure of a chronic illness means to people, whether that is about control of their body, or the consequences for their social identity. We can encompass the subtle meanings of events or artefacts that might otherwise remain unremarked.

We are interested in understanding how Qualitative Research techniques might be used in a systematic way to elicit security requirements from users. However, the challenge of understanding the human experience of technology is such that there can be no 'silver bullet' for eliciting requirements for all possible scenarios. We therefore limit our study to the elicitation of attribute based policies. Our motivation for focusing on these policies in this paper comes from previous research [16] which suggested that Grounded Theory, a qualitative data analysis technique, might be used to help uncover attributes of a Qualitative Bayesian Network that in turn describes the elicited policy. In this paper we build on this by considering the challenges of integrating it as part of a broader qualitative methodology for eliciting requirements.

This paper is organized as follows. Section 2 considers existing literature on eliciting security needs and discusses the challenges to carrying out qualitative research in practice. Section 3, using a simple use-case, describes how a Qualitative Researcher would conventionally approach understanding a user's needs, in this instance, by using semi-structured interviewing and Grounded Theory analysis. Grounded theory as a research approach is a systematic and in-depth process. While qualitative research is demanding and time-consuming, it is widely accepted as a valid means of gaining an insightful understanding of the user's experience and needs around technology [1]. Section 4 describes how machine learning of the interview transcripts marked up during Grounded Theory analysis can be used to generate a Bayesian network that provides a model security policy of the requirements. Encoding the policy as a probabilistic network recognizes that the policy needs of the user gathered during elicitation may be approximate and can be further learned based on subsequent user behavior. Having illustrated how a qualitative researcher would approach an elicitation exercise, Section 5 considers the lessons learned and considers how a technical person, who is not a Qualitative Researcher, might approach the methodology.

2 Related research

Research on security requirements engineering has tended to focus on the engineering lifecycle of capturing/analyzing security requirmements, through to system development, and with respect to functional requirements. Approaches such as SecureUML [4] can help to model and analyze aspects of understood requirements, while threat driven approaches [14] use threat and countermeasure use-cases to help understand and drive requirements. Methodologies such as Secure Tropos [22, 23] provide a goal-oriented modeling approach to capture and analyze socio-technical security requirements, providing engineering support from early requirements through to detailed design. With respect to security requirements, their emphasis is more about requirements capture through the use

of expressive modeling languages, and less about *how* to draw out the needs of the individual user. In this paper we focus on the elicitation of security needs from the user as a human being. We consider how the requirements engineer can come to learn about these needs by interacting with the user. Intuitively, elicitation occurs before, and provides input to, requirements capture, for instance, before the early requirements phase in Secure Tropos [22, 23]. We focus on requirements for security policies constructed in terms of attributes, as might be used in an attribute based policy. Whether this elicitation technique can be applied to more general requirements capture is a topic for future research.

Our position is that Grounded Theory analysis [9] of semi-structured interviews [20] can provide a methodological basis for eliciting these policy attributes. Grounded Theory is a qualitative research method commonly used in social sciences for generating theories demonstrably grounded in data, hence avoiding the imposition of a priori assumptions about the problem domain. Data can range from transcripts of interviews to recordings of interaction. Existing computer-assisted qualitative data analysis software/Grounded Theory analysis tools, such as NVIVO, provide editing and syntactic analysis, but do not provide semantic modeling. Qualitative methods are often used in usability security studies to better understand the user experience of security. For example, [32] examines the experience of regret in social media use.

In Computer Science, qualitative methods have been used to help elicit requirements in Software Engineering [28] and compliance [6]. In the security domain, for example [2, 13, 15, 25], Grounded Theory has been been used to help understand user behaviour as part of security system design. Personal privacy/security assistants such as [7, 19] help users decide policy, however by taking a structured/questionnaire based approach to eliciting policy they pre-judge the attributes that are important, in contrast to the approach taken in this paper.

Qualitative Methods are coming to the fore as a systematic approach for uncovering emergent security requirements. In [27] Grounded Theory is used in conjunction with fault-trees as a methodological means to identify emergent threats and thereby discover unknown knowns. In [18], interviews and focus groups were carried out in order to understand Grid access control needs, and an access control language was developed to support these requirements. Studies have used an enthnomethodological approach to elicit privacy requirements for mobile applications [5,30]. Although these studies help uncover security and privacy needs, the difficulty lies in taking these needs, elicited in qualitative data, and rendering them into requirements that can be directly implemented by developers. The next step is transforming these needs into actionable requirements. The methodology proposed in the following sections addresses this challenge.

3 A Qualitative Research perspective on elicitation

3.1 A use-case

One of the authors, an Applied Psychologist, conducted a qualitative study to research how people make sense of photograph sharing using their camera phone.

In this context, privacy and identity are closely related, and the control of personal identifying information is fundamental [29]. The use of mobile phones has created social norms in relation to acceptable behaviour and habits, and normative behaviour continues to evolve [17]. Understanding social norms around photograph sharing means developing an understanding of how people make sense of this activity in their evolving social world. Conventional photo-sharing security controls can be too coarse grained for user needs and users work around them in order to achieve their shared goals [3]. The aim of this use-case is to go beyond these existing security requirements, exploring individual values, beliefs and experiences. This will facilitate developing an understanding of their interplay and thus why, and how, the nuances of personal preferences are formed.

While the outcome of such a study builds an understanding of people's relationship with technology, from the perspective of user-centered design, it is also relevant to eliciting their needs/requirements for security. In this section we sketch this elicitation process as a form of qualitative research. Section 4 will describe how security policies can in turn be generated from this process.

The focus of the current paper is methodological, hence the original qualitative study will be described briefly. This is detailed elsewhere [16]. The background, in brief, is that seven interviews were conducted, with a duration ranging from 17 to 54 minutes, generating a total of 226 minutes of data. Interview number 7, on which the use case is based, generated a total of 51 codes during the analytic process. The use case is based on a subset of those codes.

3.2 Qualitative Research

The Qualitative Research approach to finding out what people need and want in a particular set of circumstances is to understand how they make sense of that situation. How people make sense of a situation, their unique perspective, is based on the whole of their experience. Experience itself is a complex concept. A person's past, present and anticipated future, are part of their experience. Experience is comprised of components that are physical, social, intellectual and emotional. Given that one person's experience can include these intersecting and non-linear aspects, then the breath and depth of that experience presents a challenge for understanding how they make sense of a particular situation.

Understanding experience demands methodological resources in terms of skills and time. Resources are finite, and researchers adapt accordingly. Thus, at the outset of a research project, a simple yet important question is asked: what is it that you want to find out. For example, from a social psychological perspective, a question might be: how is a person sharing photographs. If I decide to approach the research by compiling a questionnaire, then I will have answers that reflect a particular set of questions. This set of questions will be the same for all participants, and as such, curtails the scope of possible answers. However, with this method, the advantage is in the large number of possible participants. If I decide to approach the research in a different way, I have scope to answer different questions. For example, I can find out, not alone how people share photographs, I can also find out how they make sense of their sharing practices,

about the values that underpin their reasoning, and how those values might have developed. With the latter approach to research, a qualitative approach, it is possible to understand the breath and depth of a person's experience, relevant to a particular set of circumstances.

Qualitative Research is characterised by a diverse and evolving methodology, including ethnography, participant observation, and semi structured interviewing. Analytic methods include Grounded Theory and Discourse Analysis. Methodological decisions are underpinned by particular epistemologies [31]. One position is Social Constructionism, taking the view that humans actively create their world in social interaction. Researchers working from this perspective would argue that research is not objective, rather what emerges is interpretative. Approached in this way, semi-structured interviewing is considered to be a creative engagement where the unexpected can emerge in the dialogue between interviewer and participant. In contrast, a structured interview utilises the same schedule for all participants, and is similar to a questionnaire (Smith, 1995) curtailing what is discussed. Thus, with semi-structured interviewing, rather than structuring the dialogue to fit a preconceived format, it is possible to explore what is unique for each participant. The data that emerges is analysed to understand the process of making meaning.

3.3 Eliciting via semi structured interviewing

The skill of conducting semi structured interviews is in itself the subject of scholarly study [20]. In the current research, the example being used for methodological purposes is interviewing in order to find out about photograph sharing practices. This means, as discussed above, delving into the personal values and beliefs that inform choices. With the aim of developing an in-depth understanding of the unique perspectives that individuals bring to making sense of photograph sharing, semi structured interviews are an ideal way to proceed.

Temporal structure of an interview The interview schedule is a guide for the researcher. The interview follows a temporal structure: the present, the future and the past. This facilitates the participant talking about and reflecting on the subject matter. Discussing the present can be an easy way to begin a dialogue to ensure that, as far as possible, the participant is comfortable with the process and subject matter. During interview, the participants decide on, and choose, the particular matters discussed and the extent that they wish to go into detail. This may include decisions made in the past, whether they now have a different perspective on what they did, and why that may have changed over time. Hypothetical questions related to the area of interest facilitate the participant in considering their approach to the subject matter, and may prompt recall of particular incidents, and conversation about what is, and was, relevant to their decision making. Meaningful past events may have prompted participants to reflect on the subject matter in depth. Such events and reflections can provide valuable insights into the participant experience, and their psychological world.

Develop the interview schedule This directs the dialogue with the participant. A sample is provided in Figure 1. The schedule is not referred to during the interview, to avoid disrupting the flow of the conversation. This process also reflects the semi-structured nature of the interview, as unexpected avenues of discussion raised by participants can become an alternative focus, and this is facilitated by the use of prompts. A sample is provided in Figure 2.

VR: Have you used your camera phone to take photos of family and friends?
VR: Have you kept any of those photos? Can I see some?
VR: Would you give copies of your photographs to someone else? Say that you had a photo of a friend in your camera and another person saw it, but didn't know them, and wanted a copy. Would you give them a copy?
VR: Would you give me a copy? Would you give anyone a copy? How am I different?
VR: Do you think it would be the same if the photograph was of people that neither of you knew, say you took a photo of some famous landmark, like the leaning tower of Pisa, and the photo turned out to have people in it, so you could see them clearly. Would you give those photos to anyone who wanted them?
VR: Would you take a photo of someone you knew if they didn't know what you were doing? Say you saw a friend on the street but they didn't see you, would you think it is ok to take their photo?
VR: What about if someone, maybe a friend or a family member, was at your house and say, fell asleep, would you take their photo?
VR: Would you feel you should show it to them or not? Give them a copy if they wanted it?
VR: Has anyone ever taken a photo of you when you didn't know about it? What do you think about that? Would you mind what they did with the photo? Say if they used it in an advertising campaign? Why?
VR: Have you taken photos that you felt were intrusive, maybe the people didn't know what you were doing, maybe they were asleep, maybe they couldn't stop you?
VR: Would you take a photo of someone you didn't know on the street, say you liked their haircut or shoes and wanted to copy the hair or something, would you do that?
VR: Why not? Would it be different if they couldn't see you? Would you mind if they saw you?
VR: I am wondering about taking photos of people who don't have cameras or access to cameras. Say you were out in a place, maybe a city, and saw people living on the street, would you take their photos?
VR: Is there anything else you would like to add?

Fig. 1. Interview schedule excerpt

Address informed consent Provide information on the research project and methodology, including its qualitative nature using semi structured interviewing and Grounded Theory. Explain anonymity and deidentification, that anything said during interview can be excluded if the participant wishes, and the freedom

to end participation at any time, without consequence. Provide contact details of the researcher, offer a copy of the transcript, explain that verbatim quotations might be used in academic publications. Explain transcription and analysis, including that only the researcher may access the audio recording, which would be deleted following analysis, and that the interview would cease at any time if they chose to do so. Informed consent is regarded as a process in Qualitative Research, and the preceding summary is for illustrative purposes.

VR: any idea why?
VR: that's interesting
VR: hadn't thought of it like that
VR: I wonder why?
VR: could you tell me more about that?
VR: could we talk about that?
VR: I'd like to know more about that

Fig. 2. Interview prompts

3.4 Data Analysis

Transcribe the audio recordings of interviews The goal of transcription is the production of an account of an interview that is manageable, readable, and amenable to the analytic method [20, 24].

Code Data Grounded Theory techniques for data analysis include the coding data at various levels, the use of constant comparison during coding, the generation of categories, and Memo writing. Initial coding can be a line by line process, whereby labels are applied to text, to assign and encapsulate meaning. A coded piece of text could be a phrase, a sentence, or a few sentences [9].

The process of coding is described in Memos, as are ideas on the direction of the analysis. Generating categories means grouping codes together to reflect the emerging analysis.

Initial categories The following categories were generated by the analysis.

- Control of the images
- Entitlement to photographs for the person in them
- Images of self
- Privacy
- Sharing of images
- Photographs of strangers contrasted with friends
- Trust
- Treating others as I wish to be treated (empathy)

Analysis is a flexible process, and categories may be merged or refined, reflecting ideas for theoretical development. Links between categories, such as similarity and difference, are explored and described. All of the foregoing steps are recorded in Memos, which are rewritten and expanded iteratively during the research process. Qualitative research using Grounded Theory methods is time consuming, requiring an intensive engagement with the data. The preceding summarises this for illustrative purposes.

4 Moving from Grounded Theory to attribute policies

Grounded Theory analysis, through coding, identifies a range of code phenomena that can be considered to provide an interpretation, or semantics, of the syntax of the interview text. It is argued [16] that these codes might be treated as discrete probabilistic variables, representing the probability of the occurrence of the phenomena of interest; however, [16] does not consider how it can be generated in practice. In this paper we propose a LaTeX based notation used by the Analyst to mark up interview text[1] with details about the codes and their dependencies. The marked-up script provides a meaning for its content, and machine learning is used on this marked-up data to build a Bayesian Network based policy that represents the relationships between the identified phenomena. Coding identifies a variety of attributes, as probabilistic variables, including those representing the characteristics of entities of interest, and actions and decisions. For example, code (attribute) child means that a photograph contains a child, while code share corresponds to the action of sharing a photograph.

Fig. 3. Sample Bayesian Network Policy

Figure 3 gives a fragment of a policy for the photograph-sharing use-case. Variables suffering and child are observed; for example, indicating the presence of corresponding image tags in a photograph, where child means the scene contains a child, and so forth. Latent attributes vulnerable and share are inferred and share represents the decision to share the photograph in question, potentially with a family member, or otherwise A policy query is interpreted as: *given a collection of observations that describe attributes related to some proposed action, then what is the probability of a given decision?* For example, what is the probability that it is OK to share a photograph depicting a child (tag) in a public place (tag)

[1] LaTeX was chosen for expediency.

with a family member? In eliciting a policy, one identifies these variables along with associated probability distributions as follows.

4.1 Line by Line Coding

The observation of a phenomena, denoted by the code v:c, uncovered during a Grounded Theory analysis, is represented by a state c of a discrete random variable v. This observation is identified by marking-up, using a line by line code, in the dialogue transcript in which the phenomena is observed. We assume that each code uncovered during a Grounded Theory analysis corresponds to a state (identified as c) of some random variable (identified as v) and a coding markup (in LaTeX) \qaCode{v:c}{text} specifies the observation of a phenomena, as state c of a random variable v, in the given transcript text. For example, observations are noted about the public and private sharing of photographs:

```
\qaCode{share:private}{protecting people's dignity and privacy and
things like that} [...] Its probably more straightforward when its
family and friends but if you're using it in a
\qaCode{share:public}{public context to serve a different purpose, I
think then maybe its a bit harder to weigh it up,} [...]
```

and this is typeset in LaTeX as:

Answer: protecting people's dignity and privacy and things like that [...] its probably more straightforward when its family and friends but if you're using it in a public context to serve a different purpose, I think then maybe its a bit harder to weigh it up,	share:private share:public

A predefined taxonomy of policy variables/states is not required: it is the coding during Grounded Theory that surfaces the variables/states relevant to the policy.

During coding, a phenomena v:c is routinely characterized as a tautology, and therefore, for simplicity, we assume that its complementary state is identified syntactically as !v. Furthermore, many phenomena are binary in nature, and when no ambiguity arises, we assume that the state and its corresponding variable use the same identifier and drop reference to the random variable in marked-up text. For example, code child denotes a state of variable child, reflecting the observation of phenomena of the presence of a child in a photograph:

```
\qaCode{!child}{I dunno, pictures of kids, they're bordering on
something else, they're, I don't know, I don't know if I'd take
pictures of [...]}
```

and code !child denotes an observation that a child is not present in a photograph. Thus, probabilistic variable child has literal states child and !child, while probabilistic variable share has states sharepublic, shareprivate and !share.

4.2 Observing collections of phenomena

During the coding process, the analyst may group any number of codes together using code conjunction (+) and disjunction (,) operators in order to specify a simultaneous observation of phenomena in the transcript. For example,

```
\qaCode{child+vulnerable }{[...] kids don't tend to have that ability
to be able to tell if it was a right or wrong thing or what they
think or what they feel , no I don't think I'd take pictures of
anyone's children , of any children . }
```

is the observation that the participant is relating a child as a vulnerable person in the photograph. In coding this simultaneous observation of the phenomena, no assumption is made about the nature of any statistical dependency between the variables, other than the observation of a simultaneous occurrence of states child and vulnerable.

The analyst may also assert there is no known relationship between phenomena, but for convenience, wishes to group the codes together using disjunction. For example,

```
\qaCode{share:public,share: private }{it could be public or private }
```

Inclusion of one code within the scope of the text of another is considered to define a simultaneous observation of phenomena. For example, during a discussion about sharing photographs privately, the participant remarks

```
[...] \qaCode{share:private }{[...] because there's a line that you
don't cross when it comes to \qaCode{vulnerable}{protecting people's
dignity and privacy} and things like that and I think the difficulty
sometimes is trying to weigh that up.}
```

reflects that, in addition to the observation of the phenomena of sharing photographs, there is a simultaneous observation of privately sharing photographs of vulnerable individuals.

Axial-coding, which is used in Grounded Theory to relate codes (categories and concepts) to each other via a combination of inductive and deductive thinking, organizes codes according to categories, for example,

```
\qaCode{place>developingcountry}{for instance pictures taken in
developing countries of starving children and [...]}
```

the axial code place>developingcountry specifies an observation about the code phenomena developingcountry related to the code category place. While categories may be associated with multiple codes, a category may also define a code in itself. For example,

```
\qaCode{vulnerable>suffering }{[...]  I think to me it would be
 exploiting that person really and considering their circumstances, its
almost like you're taking, sort of dehumanizing that person, almost
 objectifying them sort of, so in a sense you're homeless, you're on
the street , }
```

In this case, markup with the code vulnerable > suffering specifies the simultaneous observation of codes vulnerable and suffering in the text, and this is interpreted as the code expression vulnerable ,(vulnerable + suffering).

The analyst uses the codes to reflect the observations concerning phenomena. When there is ambiguity then we assume that the analyst will either use the codes to reflect this ambiguity or will seek to provide additional context in order to eliminate ambiguity. For example, a participant is unsure whether or not he would be comfortable in privately sharing a photograph of a child,

```
\qaCode{share:private, share: public ,! share}{ \qaCode{child,! child }{
sometimes I'll share a photograph of a child and sometimes not } }
```

However, the goal of the semi-structured interview and subsequent Grounded Theory analysis is, in this case, to uncover the phenomena that characterize sharing of photographs, and, therefore, it is anticipated that such ambiguity should be avoided.

Similarly, rather than using weighting schemes, it is assumed that sufficient context is provided during coding in order to clarify the significance of a phenomenon. For example,

```
\qaCode{share:public+adult}{I'd nearly always share a photograph
containing an adult }
```

During the interview, a context is elicited

```
\qaCode{!share+adult+suffering }{[...]  but not if the person was
 suffering },  \qaCode{share:private+adult+suffering}{but perhaps it
would be OK if done privately }
```

4.3 Describing dependencies between phenomena

During the process of analysis, dependencies between the codes/random variables are inferred by the analyst. For example, the analyst identifies a dependency between codes vulnerable and child and decides that the value of child influences vulnerable in some way. Similarly, the analyst decides that the decision on sharing is influenced by vulnerable and whether it is with a family member.

The text of the dialogue is marked up to reflect these phenomena, though it need not be tied to any particular part of the dialogue, as it is a theory

that emerges over the entire transcript. These code dependencies, in turn, are used to define the variable dependencies in the generated Bayesian Network policy. For example, the markup \qaDep{vulnerable}{share} specifies dependency vulnerable→share. During the course of the Grounded Theory analysis of the photograph sharing interviews, dependencies were identified and marked up in the LaTeX source, and, for the example in this paper, these included the following.

```
\qaDep{child}{vulnerable}
\qaDep{adult}{vulnerable}
\qaDep{suffering}{vulnerable}
\qaDep{vulnerable}{share}
\qaDep{family}{share}
```

4.4 Learning policies

The marked-up transcript identifies (multiple) occurrences of phenomena (variables) and their dependencies and these are used to construct the structure for the Bayesian Network policy given in Figure 3. Consider the typeset fragments of marked up interview text in Figure 4; the observed codes and code-dependencies marked-up in the LaTeXsource during analysis are typeset in the margin.

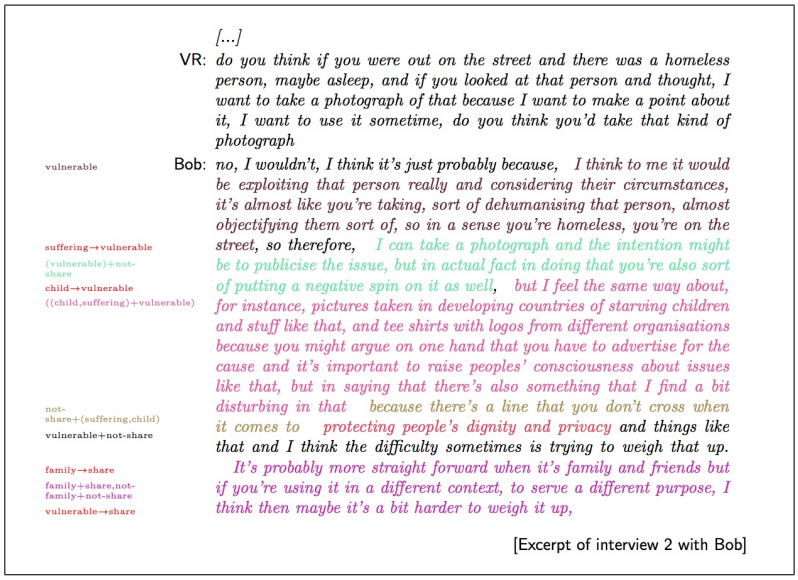

Fig. 4. Typeset interview

Each code markup in the transcript is a code-expression corresponding to the observation of a collection of phenomena, and we store each markup as a

disjunctive normal form dataset of observed codes. Figure 3 provides an example of the phenomena observed in the above, along with the generated dataset. Note

	child	family	share	suffering	vulnerable
vulnerable	-	-	-	-	vulnerable
(vulnerable)+!share	-	-	!share	-	vulnerable
((child,suffering)+vulnerable)	child	-	-	-	vulnerable
!share+(suffering,child)	-	-	-	suffering	vulnerable
vulnerable+!share	-	-	!share	suffering	-
family+share, !family+!share	child	-	!share	-	-
	-	-	!share	-	vulnerable
	-	family	share	-	-
	-	!family	!share	-	-

Fig. 5. Selected observations and generated DNF dataset

that absence ("-") of a phenomenon observation means that nothing is known about that code and it does not necessarily mean the complement of the state. Recording (child+vulnerable) indicates the phenomena of a vulnerable child has been observed, but no observation is made about sharing at that point in the transcript. Equally, recording (share+family) indicates the willingness to share a photo with a family member, but with no statement made about other variables at that point.

Intuitively, each line of the dataset in Figure 5 represents an access control rule about sharing, based on attributes (codes) discovered during analysis. However, we cannot treat this dataset as exhaustive: no matter how expressive a phenomena-coding markup language might be, every possible sharing combination cannot be explicitly discussed during an interview. It is therefore necessary to estimate the gaps in the policy rules by inferring the probabilities for the variables, including the transitional probabilities. For ease of exposition in this paper, we took a somewhat promiscuous view of access control, whereby the Expectation Maximum (EM) learning algorithm [21] is used to maximize the variable probabilities. This means that a user is assumed likely to share photographs so long as it not inconsistent with their policy (a probabilistic default share). Learning, based on a probabilistic default not-share is a topic for future work. The appendix gives the details of the generated Bayesian Network.

In the interview fragment above, the participant predominantly speaks of photographs depicting suffering, children and vulnerability, and therefore, based on the markup/observations in Figure 5, the probabilities of these events in the policy are high. In the full interviews, the participants spoke of other occasions when the photograph subject was not considered vulnerable, leading to a lower likelihood of vulnerability in the policy. Based on the EM-learning of the observations in Figure 5, the conditional probability table calculated for the latent variable share in the above example is:

	family		!family	
	vulnerable	!vulnerable	vulnerable	!vulnerable
share	0.8148	1.0	0.0964	0.0
!share	0.1852	0.0	0.9036	1.0

In our example, the learning of probabilities may be open to question given the small number of observations in the dataset of interview markup provided in the fragment above. However in practice, qualitative analysis identifies not just the existence of a code, but marks up every occurrence of that code throughout the interview and thereby ensuring a larger and more effective learning dataset. Intuitively, the more the participant touches on a phenomenon in the interview, the more the code is observed/marked-up and thus, the more likely it is considered to occur. Whether it is methodologically qualitatively sound to assume that the more a participant touches on a topic during an interview, then the more relevant that topic is, is open to discussion. Designing a more expressive markup language whereby the analyst can override/weight significant phenomena is a topic for further work.

A LaTeX package was implemented that generates a Baysian network as described above. Based on the LaTeX markup, the package generates the observed phenomena dataset and the SamIam [11] EM learning implementation uses this to generate the probabilities for the Bayesian Network defined by the dependencies in the markup.

4.5 Policies in practice

In this paper a policy is represented as a Bayesian Network, reflecting the approximate nature of the data gathered and analyzed during elicitation. In one prototype application for these policies, a 'clippy' style Android photograph sharing assistant was implemented that uses the elicited policy to provide security advice to the user as to whether it is safe to upload and share a given photograph on Google Picasa; it was built using the Netica-J API for Android.

The Android camera phone enables the user to tag camera JPEG images with the attributes identified during the elicitation process. These tags provide observations for variables in a policy query, while the values for other policy variables can be based on the outcome of the EM-learning carried out during policy elicitation. The outcome of a query is the probability of the share variable: during a photograph upload, an alert is generated if the share probability is below a user-defined threshold, and the user is given the option to override. If the user decides to override the advice, or to add previously unseen tags, then this new sharing behavior can be further learned and the policy dynamically updated.

4.6 Methodological Threats

With the aim of developing an in-depth understanding of the subtleties and nuances of participant experience, the research methods of semi-structured interviewing [20], in conjunction with Grounded Theory [9] analysis were selected

as the most suitable to answer the research question. The scope of the qualitative study was subject to practicalities, such as time frame and available resources. The time frame was a period of six months, with a resource of one researcher available on a part-time basis. With the foregoing constraints, the number of possible participants was deemed to be between 5 and 10. In the event, 7 participants were interviewed. The recruitment, data collection and analysis were conducted by a single researcher. Epistemologically, Social Constructionism underpins the original qualitative research, and is coherent with the interpretivist approach taken in the data analysis [31]. Hence, the results make no claim to objectivity, or to be applicable to the general population. Rather the emergent analysis is an in-depth and subjective exploration of the experience of a small group, on which the use-case is derived.

5 Discussion and Conclusion

In this paper we consider how techniques that are used in Applied Psychology to understand a person's feelings and needs might provide a means to elicit their security needs. While it is a truism that Qualitative research techniques could be used to achieve this, the contribution of this paper is to map the activity into something actionable, that is, provide a means to generate a machine-interpretable security policy. Recognising that the codes uncovered during a Grounded Theory analysis of semi-structured interview data can be interpreted as attributes for an attribute-based access control policy, the paper describes how the Qualitative Method proceeds, from interviewing the user-participant, to analysing and uncovering the codes, and to mapping these codes to probabilistic variables in a Bayesian Network that provides the final policy.

One of the key contributions of the paper is the demonstration of how Grounded Theory can be used to identify policy attributes and their relationships. We chose to use a Bayesian Network as the policy model in order to support incompleteness in elicitation. Mapping these attributes to a particular ABAC model, for example XACML, is a topic for future research. We believe that the Qualitative approach in Section 3 could assist in eliciting attributes for more general requirements, such as SI* requirements [22]; however, the extent to which requirements could be learnt using a strategy similar to Section 4 is an open question.

Grounded Theory analysis of semi-structured interview data is effective at uncovering individuals' needs and wants, however, by virtue of what it attempts to uncover, it is a costly and time-consuming activity, and requires a high degree of skill to carry out properly. As a technical person, a security requirements engineer is unlikely to have the requisite skills, nor the luxury of resources, to be able to conduct such an in-depth study in order to obtain the user policies. However, we argue that a more conventional elicitation of user requirements will not necessarily uncover the user's true needs to the same degree.

We are exploring how the Qualitative method described in Section 3 could inform a more lightweight methodology for policy elicitation that could be used

by a Requirements Engineer in a cost-effective manner. For example, rather than conducting semi-structured interviewing with recordings and transcripts, the Engineer might simply make their own hand-written notes during their discussions, which they subsequently mark-up, and from which the Bayesian Network policy is learned. However, as is clear from Section 3, the validity of the requirements that are elicited very much depend on the skills of the interviewer, and one must be careful not to sacrifice completeness for the sake of expediency.

Acknowledgements Thanks to Simon O'Donovan who prototyped the Android photograph sharing assistant for his UCC Bachelor's degree project. This work was supported, in part, by Science Foundation Ireland grant SFI/12/RC/2289 and by the Cyber CNI Chair of Institute Mines-Télécom which is held by IMT Atlantique and supported by Airbus Defence and Space, Amossys, EDF, Orange, La Poste, Nokia, Société Générale and the Regional Council of Brittany; it has been acknowledged by the French Centre of Excellence in Cybersecurity.

References

1. Adams, A., Lunt, P., Cairns, P.: A qualititative approach to HCI research. In: Cairns, P., Cox, A. (eds.) Research Methods for Human-Computer Interaction. Cambridge University Press (2008)
2. Adams, A., Sasse, M.: Users are not the enemy. CACM 42(12), 40–46 (1999)
3. Ahern, S., Eckles, D., Good, N.S., King, S., Naaman, M.: Over-exposed? privacy patterns and considerations in online and mobile photo sharing. In: SIGCHI conference on human factors in computing systems. pp. 357–366 (2007)
4. Basin, D., Doser, J., Lodderstedt, T.: Model driven security for process-oriented systems. In: Symposium on Access control Models and Technologies (2003)
5. Bellotti, V., Sellen, A.: Design for privacy in ubiquitous computing environments. In: European Conference on Computer Supported Cooperative Work (1993)
6. Breaux, T., Antón, A.: Analyzing regulatory rules for privacy and security requirements 34(1), 5–20 (2008)
7. Cadiz, J., Gupta, A.: Privacy interfaces for collaboration. Microsoft Research Journal (2001)
8. Caputo, D.D., Pfleeger, S.L., Sasse, M.A., Ammann, P., Offutt, J., Deng, L.: Barriers to usable security? Three organizational case studies. IEEE Security and Privacy 14(5), 22–32 (Sep 2016), https://doi.org/10.1109/MSP.2016.95
9. Charmaz, K.: Constructing Grounded Theory. Sage Publications, London (2006)
10. Charmaz, K.: Disclosing illness and disability in the workplace. Journal of International Education in Business 3(1/2), 6–19 (2010)
11. Darwiche, A., et al.: SamIam: Sensitivity Analysis, Modeling, Inference and More. http://reasoning.cs.ucla.edu/samiam/, UCLA Automated Reasoning Group. accessed 07/07/2017
12. Dodier-Lazaro, S., Abu-Salma, R., Becker, I., Sasse, M.A.: From paternalistic to user-centred security: Putting users first with value-sensitive design. In: Proceedings of the 3rd CHI Workshop on Values in Computing (2017)
13. Dourish, P., Grinter, E., de la Flor, J.D., Joseph, M.: Security in the wild: user strategies for managing security as an everyday, practical problem. Personal Ubiquitous Comput. 8(6), 391–401 (2004)

14. Firesmith, D.: Security use cases. The Journal of Object Technology 3(2) (2003)
15. Flechais, I., Mascolo, C., Sasse, M.: Integrating security and usability into the requirements and design process. Int. J. Electron. Secur. Digit. Forensic 1(1), 12–26 (2007)
16. Foley, S.N., Rooney, V.M.: Qualitative analysis for trust management. In: Security Protocols Workshop. pp. 298–307. Springer LNCS 7028 (2009)
17. Hakkila, J., Chatfield, C.: It's like if you opened someone else's letter: user perceived privacy and social practices with sms communication. In: CHI 05: MobileCHI, 7th international conference on Human Computer Interaction with mobile devices and services. pp. 357–366 (2005)
18. Inglesant, P., Sasse, A., Chadwick, D., Shi, L.: Expressions of expertness: The virtuous circle of natural language for access control policy specification. In: Symposium on Usable Privacy and Security (SOUPS) 2008, Pittsburg, PA, USA (2008)
19. Jendricke, U., Gerd tom Markotten, D.: Usability meets security - the identity-manager as your personal security assistant for the internet. In: 16th Annual Computer Security Applications Conference (2000)
20. Kvale, S., Brinkmann, S.: InterViews. Learning the Craft of Qualitative Research Interviewing. Sage Publications, London, 2 edn. (2009)
21. Lauritzen, S.: The EM algorithm for graphical association models with missing data. Computational Statistics & Data Analysis 19, 191–201 (1995)
22. Massacci, F., Mylopoulos, J., Zannone, N.: Security requirements engineering: The SI* modeling language and the secure tropos methodology. In: Advances in Intelligent Information Systems, pp. 147–174 (2010)
23. Mouratidis, H., Giorgini, P.: Secure Tropos: a security-oriented extension of the Tropos methodology. International Journal of Software Engineering and Knowledge Engineering 17(2), 285–309 (2007)
24. O'Connell, D.C., Kowal, S.: Basic principles of transcription. In: Smith, J.A., Harre, R., Van Langenhove, L. (eds.) Rethinking Methods in Psychology. Part II, Discourse as Topic, chap. 7. Sage Publications, London (1995)
25. Onabajo, A., Jahnke, J.: Properties of confidentiality requirements. In: 19th IEEE Symposium on Computer-Based Medical Systems (2006)
26. Parkkola, H., Saariluoma, P., Berki, E.: Action-oriented classification of families' information and communication actions: exploring mothers' viewpoints. Behaviour and Information Technology 28(6), 525–536 (2009)
27. Rashid, A., et al.: Discovering "unknown known" security requirements. In: International Conference on Software Engineering. ACM Press (2016)
28. Seaman, C.: Qualitative methods in empirical studies of software engineering. IEEE Trans. on Software Engineering 25(4), 557–572 (1999)
29. Srivastava, S.: Mobile phones and the evolution of social behaviour. Behaviour and Information Technology 24(2), 111–129 (2005)
30. Thomas, K., Bandara, A., Price, B., Nuseibeh, B.: Distilling privacy requirements for mobile applications. In: 36th International Conference on Software Engineering (ICSE2014), 31 May-7 June, 2014, Hyderabad, India. pp. 871–882 (2014)
31. Twining, P., et al.: Some guidance on conducting and reporting qualitative studies. Computers and Education 106(March), A1–A9 (2017)
32. Wang, Y., et al.: I regretted the minute I pressed share: A qualitative study of regrets on facebook. In: Symposium on Usable Privacy and Security (SOUPS) 2011, Pittsburg, PA, USA (2011)
33. Zurko, M.E., Simon, R.T.: User-centered security. In: 1996 Workshop on New Security Paradigms. pp. 27–33. NSPW '96, ACM

A Sample policy

A.1 Marked up interview text

```
%A small fragment of the original marked-up interview.
%
\begin{interview}{Bob}{2}
 [...]
\qquestion
do you think if you were out on the street and there was a homeless
person, maybe asleep, and if you looked at that person and thought, I
want to take a photograph of that because I want to make a point about
it, I want to use it sometime, do you think you'd take that kind of
photograph

\answer
no, I wouldn't, I think it's just probably because,
\qaCode{vulnerable}{I think to me it would be exploiting that person
 really and considering their circumstances, it's almost like you're
taking, sort of dehumanising that person, almost objectifying them
sort of, so in a sense you're homeless, you're on the street }, so
 therefore , \qaDep{suffering}{vulnerable}
\qaCode{(vulnerable)+!share}{I can take a photograph and the intention
might be to publicise the issue, but in actual fact in doing that
you're also sort of putting a negative spin on it as well },
\qaDep{child}{vulnerable} \qaCode{((child, suffering )+vulnerable)}{but
I feel the same way about, for instance, pictures taken in developing
 countries of starving children and stuff like that, and tee shirts
with logos from different organisations because you might argue on one
hand that you have to advertise for the cause and it's important to
 raise peoples' consciousness about issues like that, but in saying
that there's also something that I find a bit disturbing in that}
\qaCode{!share+(suffering, child )}{because there's a line that you
don't cross when it comes to} \qaCode{vulnerable+!share}{protecting
people's dignity and privacy} and things like that and I think the
 difficulty sometimes is trying to weigh that up.
\qaDep{family}{share} \qaCode{family+share,!family+!share}{It's
probably more straight forward when it's family and friends but if
you're using it in a different context, to serve a different purpose,
I think then maybe it's a bit harder to weigh it up, } [...]
\qaDep{vulnerable}{share}
\end{interview}
```

A.2 Generated Bayesian Network Policy

```
node child
{
        states = ("child" "XXXNOTchild" );
        diagnosistype = "AUXILIARY";
        DSLxSUBMODEL = "Root Submodel";
        ismapvariable = "false";
        ID = "child";
        label = "child";
        DSLxEXTRA_DEFINITIONxDIAGNOSIS_TYPE = "AUXILIARY";
        excludepolicy = "include whole CPT";
}
node vulnerable
{
        states = ("vulnerable" "XXXNOTvulnerable" );
        diagnosistype = "AUXILIARY";
        DSLxSUBMODEL = "Root Submodel";
        ismapvariable = "false";
        ID = "vulnerable";
        label = "vulnerable";
        DSLxEXTRA_DEFINITIONxDIAGNOSIS_TYPE = "AUXILIARY";
        excludepolicy = "include whole CPT";
}
node suffering
{
        states = ("suffering" "XXXNOTsuffering" );
        diagnosistype = "AUXILIARY";
        DSLxSUBMODEL = "Root Submodel";
        ismapvariable = "false";
        ID = "suffering";
        label = "suffering";
        DSLxEXTRA_DEFINITIONxDIAGNOSIS_TYPE = "AUXILIARY";
        excludepolicy = "include whole CPT";
}
node family
{
        states = ("family" "XXXNOTfamily" );
        diagnosistype = "AUXILIARY";
        DSLxSUBMODEL = "Root Submodel";
        ismapvariable = "false";
        ID = "family";
        label = "family";
        DSLxEXTRA_DEFINITIONxDIAGNOSIS_TYPE = "AUXILIARY";
        excludepolicy = "include whole CPT";}
```

```
node share
{
        states = ("share" "XXXNOTshare" );
        diagnosistype = "AUXILIARY";
        DSLxSUBMODEL = "Root Submodel";
        ismapvariable = "false";
        ID = "share";
        label = "share";
        DSLxEXTRA_DEFINITIONxDIAGNOSIS_TYPE = "AUXILIARY";
        excludepolicy = "include whole CPT";
}
potential ( child | )
{
        data = (        0.8553  0.14467 );
}
potential ( vulnerable | suffering  child )
{
        data = (((      0.8527  0.1473  )
                 (      0.8586  0.1414  ))
                 ((     0.8655  0.1345  )
                 (      0.8789  0.1211  )));
}
potential ( suffering | )
{
        data = (        0.8481  0.1519  );
}
potential ( family | )
{
        data = (        0.2545  0.7455  );
}
potential ( share | family  vulnerable )
{
        data = (((      0.8148  0.1852  )
                 (      1.0     0.0     ))
                 ((     0.0964  0.9036  )
                 (      0.0     1.0     )));
}
```

The above Bayesian network, in Hugin .net format, was generated by SamIam [11] using EM-learning on the dataset given in Figure 5. Note that in this implemented policy, each complementary state $!v$ is encoded as literal XXXNOTv.