# Modelling and Automatically Analysing Privacy Properties for Honest-but-Curious Adversaries

Andrew Paverd Department of Computer Science University of Oxford andrew.paverd@cs.ox.ac.uk Andrew Martin Department of Computer Science University of Oxford andrew.martin@cs.ox.ac.uk Ian Brown Oxford Internet Institute University of Oxford ian.brown@oii.ox.ac.uk

Abstract—For certain communication protocols, undetectability and unlinkability of messages or information items are desirable properties, and are used to reason about anonymity and privacy. Previous work has formalized and analysed these properties using the notions of indistinguishability and observational equivalence. However, it is also possible to perform this analysis using a constructive definition of the adversary model - an approach that has received less attention. The semihonest or honest-but-curious (HBC) adversary is commonly used in the analysis of these privacy properties. In this work, we develop a formal model of the capabilities of an HBC adversary with respect to undetectability and unlinkability. Our HBC model is defined as a deductive system consisting of a set of inference rules. We show that each rule is based on commonly accepted definitions and therefore claim that our overall model is a faithful representation of these definitions. The advantage of this constructive approach is that our HBC model can be directly integrated with methodologies for analysing security properties. We demonstrate this by integrating our HBC model with Casper/FDR, an established protocol analysis method based on the process algebra of CSP. We extend the Casper tool to also analyse undetectability and unlinkability properties for multiple adversaries based on a single description of the protocol. We demonstrate the effectiveness of our HBC model and Casper extension by analysing several protocols in terms of their security and privacy properties. In our case studies, we find new attacks as well as rediscover known attacks.

# I. INTRODUCTION

In addition to providing well-established security properties, it is sometimes desirable for communication protocols to exhibit privacy properties. This can be seen in the development of privacy-enhancing communication technologies such as anonymity networks [1] and anonymous authentication protocols [2][3][4][2]. Two important properties in communication privacy are *undetectability* and *unlinkability* because these can be used to reason about the more complex concepts of anonymity and privacy [5][6][7]. For example, anonymous communication systems aim to prevent the adversary from detecting any identifying information or linking items of interest to specific participants [8]. Recently, these privacy properties have become particularly important in application domains such as Radio Frequency Identification (RFID) communication [9][10] and the smart energy grid [11][12].

As communication protocols become more complex, it is becoming increasingly difficult to ensure that these protocols provide the required privacy properties. This has certainly been the case for security properties such as secrecy and authentication. As with these security properties, there is a need for automated formal analysis of privacy properties [13]. The use of formal methods is becoming increasingly common in the analysis of privacy properties and there has been a significant amount of research on the formalization and analysis of these properties using both the computational and symbolic paradigms. Previous work in the computational paradigm has used the notion of computational indistinguishability to reason about properties such as unlinkability and anonymity [14][15][16][17][18] [19][20]. In the symbolic paradigm, the notion of observational equivalence has been used to model similar properties [13][21][9][22].

In the symbolic paradigm, it is also possible to formalize and analyse these privacy properties as reachability problems - an approach which has received less attention in previous work. Deaune et al. [21] have suggested that it more difficult to express these properties using traditional reachability techniques in comparison to equivalence techniques. However, in this work we show that this approach is possible and indeed can provide benefits for the analysis of privacy properties. The fundamental insight that makes this possible is the use of a constructive formal definition of the capabilities of the adversary as presented in this work.

We focus on the so-called semi-honest or honest-but-curious (HBC) adversary as this type of adversary is commonly used in the analysis of these privacy properties. In this work we develop a formal model of the adversary's capabilities with respect to undetectability and unlinkability. We present this model in Section III. As we show in Section IV, one of the advantages of this constructive approach is that our HBC model can be directly integrated with existing methodologies for analysing security properties. We demonstrate this by integrating our model with Casper/FDR, an established protocol analysis method based on the process algebra of CSP. We extend the Casper/FDR tool to also analyse undetectability and unlinkability properties for multiple adversaries based on a single description of the protocol. In Section V we demonstrate the effectiveness of our HBC model and Casper/FDR extension by analysing several protocols in terms of their security and privacy properties.

### A. Motivating Example

To illustrate the need for automated analysis of both privacy and security properties, we use an example from the smart energy grid - an application domain currently undergoing a period of rapid technological development. The smart energy grid, or smart grid, refers to the next generation architecture for public energy distribution infrastructure in which networked computer systems measure and control the flow of energy. The Advanced Metering Infrastructure (AMI) is arguably the most visible aspect of the smart grid from the consumer's perspective. In the AMI, existing electricity and gas meters will be replaced by smart meters that record frequent energy measurements (in 15 or 30 minute intervals) and send these to the energy supplier or distribution network operator (DNO). These frequent measurements can be used to enable new functionality such as time-of-use billing and load prediction [23][24]. However, it has been shown that this fine-grained measurement of energy consumption could reveal private information about the behaviour of the residents, thus raising a number of privacy concerns [25][26][27][28][29][30]. The challenge is for the smart meters to communicate with the relevant entities, without compromising user privacy, whilst maintaining a sufficient level of authentication. In particular, certain external entities such as the energy supplier might not be trusted by the user. There are also strong security requirements in this domain because the information sent by the smart meter is used for customer billing and managing the flow of energy. A security compromise could therefore lead to financial loss or more serious cyber-physical attacks against the energy grid [31][32]. Therefore, we need to be sure that smart grid communication protocols will provide both security and privacy properties. Various protocols have been proposed, four of which are analysed in Section V.

# B. Contributions

In this work we present three main contributions: Firstly, we develop a formal model of the HBC adversary which differs from the commonly-used approach in that it is based on a constructive definition of the adversary's capabilities. Secondly, we prove the feasibility of this approach by integrating our model with an established methodology for the analysis of security properties. We extend the Casper/FDR tool to analyse undetectability and unlinkability as well as security properties based on a single description of the protocol and we have made our enhanced tool publicly available along with our models of the protocols analysed in this work<sup>1</sup>. Thirdly, we present analyses of several protocols in terms of both security and privacy properties and in doing so we find new attacks as well as rediscovering known attacks.

### **II. DEFINITIONS**

In this section we provide formal definitions of the core concepts used in this work, based on recent literature.

#### A. Honest-But-Curious Adversary

For protocol analysis in general, the most well-known adversary model is the so-called Dolev-Yao (DY) model [33]. The DY model is also one of the strongest possible adversaries in terms of capabilities. In the ideal case, security and privacy properties would be maintained even against a DY adversary. However, in some cases, the DY adversary is too strong to be used in a realistic model of the system. In the motivating example of the smart grid, the system should be secure against an external DY adversary. However, a legitimate participant in the protocol such as the energy supplier could not realistically be modelled as a DY adversary. In reality, various factors limit the capabilities of the energy supplier including regulations, audits, oversight and desire to maintain reputation. However, although a DY model is not appropriate in this case, it does not necessarily mean that the energy supplier is not adversarial. We therefore propose to model this agent as a semi-honest [34] or honest-but-curious (HBC) which we define as follows:

**Definition 1** (Honest-But-Curious Adversary). The honestbut-curious (HBC) adversary is a legitimate participant in a communication protocol who will not deviate from the defined protocol but will attempt to learn all possible information from legitimately received messages.

In comparison to the DY model, the HBC model is more limited in that it will not deviate from the protocol and cannot send any falsified messages. Even in comparison to a passive DY adversary, the HBC adversary is still more limited in that it cannot eavesdrop on arbitrary communication channels and can only receive messages of which it is the intended recipient.

For any protocol, the HBC adversary  $\mathcal{A}$  will have a particular view of the protocol which we denote  $\mathcal{V}_{\mathcal{A}}$ . This view is the set of all information items known to the adversary as well as the properties of these items (e.g. detection of an item) and the relationships between them (e.g. links between items). We formally define the adversary's view of the protocol as follows:

**Definition 2** (View of the system). For an HBC adversary A:

$$\mathcal{V}_{\mathcal{A}} = \{ Init_{\mathcal{A}} \cup Recv_{\mathcal{A}} \cup Ded_{\mathcal{A}} \}$$

Where  $Init_{\mathcal{A}}$  is a set of information items initially known to  $\mathcal{A}$ ;  $Recv_{\mathcal{A}}$  is the set of messages received by  $\mathcal{A}$ ,  $Ded_{\mathcal{A}}$  is the set of deductions made by  $\mathcal{A}$  based on the inference rules.

Therefore,  $\mathcal{V}_{\mathcal{A}}$  is dependant on the current state of the system. As the HBC adversary receives more messages or completes more runs of the protocol, more information items and inferences are included in  $\mathcal{V}_{\mathcal{A}}$ .

#### B. Undetectability

Pfitzmann and Hansen [5] have proposed a widely-cited terminology for privacy. They define undetectability as:

"Undetectability of an item of interest from an attacker's perspective means that the attacker cannot sufficiently distinguish whether it exists or not."

As expected, undetectability is the logical negation of detectability [5]. The *undetectability delta* of an item from an at-

<sup>&</sup>lt;sup>1</sup>Available at: https://www.cs.ox.ac.uk/people/andrew.paverd/casper/

tacker's perspective is the difference between the undetectability of the item based on the attacker's a-priori knowledge and its undetectability given the attacker's a-posteriori knowledge [5]. In the computational paradigm, the undetectability delta could take on a range of values but in the symbolic paradigm, it is assumed to be either zero (perfect undetectability) or maximum (guaranteed detectability).

Veeningen et al. [6][35] use this definition to develop a formal model of undetectability. They divide all information items in the system into three disjoint sets: the set of participating entities (e.g. natural persons), the set of identifiers (e.g. usernames or IP addresses) and the set of data items included in the exchanged messages. A data item is detectable by an actor if it forms part of that actor's view of the system [6].

Our model is based on the definition by Pfitzmann and Hansen [5] and we use a similar approach to Veeningen et al. [6][35] to reach a concrete instantiation of this definition. However, our model does not distinguish between identifiers and data items as these distinctions are specific to the protocol and the context in which it is used. Therefore, for the purpose of this work, any distinct piece of information in the messages that make up the communication protocol is simply referred to as an *information item*. In addition, we note that the definition from Pfitzmann and Hansen [5] does not necessarily require that an information item must be known by the adversary in order to be detected. We interpret their use of the term exists to mean that the item is known by at least one of the participants in the protocol. Items in the adversary's view of the protocol  $\mathcal{V}_{\mathcal{A}}$  are trivially detected by direct observation. Items not in  $\mathcal{V}_{\mathcal{A}}$  can be detected if the adversary becomes certain that the item is known by one or more of the other participants. For example, if the adversary receives an asymmetrically message that can be decrypted using a key in  $\mathcal{V}_{\mathcal{A}}$ , the adversary can be sure that the corresponding encryption key is known to the participant who encrypted the message. Since the adversary does not know the encryption key, we say the adversary has deduced the existence of this key. Therefore, in our model we define undetectability based on the result of a deductive system as follows:

**Definition 3** (Undetectability). From the perspective of an *HBC* adversary A, an information item  $i_1$  is undetectable iff:

$$detect[i_1] \notin Ded_{\mathcal{A}}$$

Where  $detect[i_1]$  is a deduction representing the existence of  $i_1$  and  $Ded_{\mathcal{A}}$  is the set of all deductions made by  $\mathcal{A}$ .

# C. Unlinkability

Pfitzmann and Hansen [5] define unlinkability as:

"Unlinkability of two or more items of interest from an attacker's perspective means that within the system, the attacker cannot sufficiently distinguish whether these items are related or not."

Linkability is the logical negation of unlinkability and the unlinkability delta is the difference in unlinkability based on the attacker's a-priori and a-posteriori knowledge [5]. Sender *anonymity* and *receiver anonymity* are described in terms of unlinkability between messages and the identifiers of the sender or receiver [5].

Veeningen et al. [6][35] again use this definition to present a formal model of unlinkability. In their model, each identifier is associated with a single entity and data items can be associated (linked) with a particular identifier. Therefore, if a particular agent can link two data items to the same identifier, these data items can be linked with each other. By extension, if two data items are linked to each other and one of these can be linked to a specific identifier, then the other can also be linked to that identifier. We use a similar approach to reach a concrete instantiation of the definition by Pfitzmann and Hansen [5]. Again, our model does not distinguish between data items and identifiers.

In most anonymous communication protocols, the most desirable property is that of sender anonymity. For example sender anonymity could be desirable when using locationbased services. If the protocol provides sender anonymity, the user could send his or her precise location to an untrusted service provider without revealing his or her identity. In this case, sender anonymity would be compromised if the service provider could link the user's identity to the anonymous location information. In smart metering, the concern is that fine-grained energy usage measurements from smart meters could be linked to specific individuals, thus compromising their privacy [25][26][27][28][29][30]. Various smart grid communication protocols have been proposed in order to achieve sender anonymity [36][37][38]. In our model, we focus specifically on sender unlinkability as the main requirement for sender anonymity. Unless otherwise stated, we use the term unlinkability to refer to sender unlinkability.

In order to model unlinkability in protocols, it is necessary to assume that all information is explicitly represented in the description of the protocol. For example, if a communication protocol is implemented in a real system, the implementation might introduce additional identifying information, such as an IP address or device identifier, which is not specified in the protocol and thus cannot be evaluated in the analysis facilitated by our model. In our model, we define a *link* ( $\leftrightarrow$ ) as follows:

**Definition 4** (Link). A link ( $\leftrightarrow$ ) is a reflexive, symmetric and transitive binary relation between elements of the set of information items ( $\mathcal{I}$ ), indicating that the items are related such that:

$$\forall a, b : \mathcal{I} \bullet (a \leftrightarrow b) \Rightarrow (b \leftrightarrow a)$$
$$\forall x, y, z : \mathcal{I} \bullet (x \leftrightarrow y) \land (y \leftrightarrow z) \Rightarrow (x \leftrightarrow z)$$

As with undetectability, a link is therefore the result of a deductive system. Given a set of information items, we aim to determine which of these items the agent can associate with the same sender. We consider two or more information items to be linkable if they can be definitively associated with the same sender. For example, we associate messages with the same sender if they share information items that are only known to a single participant other than the HBC adversary by the following deductive reasoning: If the legitimate messages  $m_1$  and  $m_2$  both contain a specific information item that is only known by a subset of entities  $(\mathcal{E}' \subseteq \mathcal{E})$ , then both  $m_1$  and  $m_2$  must have originated from members of  $\mathcal{E}'$ . If  $\mathcal{E}'$  contains only a single entity, all the information items in  $m_1$  and  $m_2$  can be definitively linked to each other and to this entity. If  $\mathcal{E}'$  contains two entities, one of which is the HBC adversary, the adversary can exclude its own messages and link all remaining messages to the other entity in  $\mathcal{E}'$ . Similar relationships can exist for larger cardinalities of  $\mathcal{E}'$ if there are multiple colluding adversaries. This interpretation of unlinkability is similar to that used by Berthold and Clauss [39] and Veeningen et al. [6]. In our model, the user can also specify additional links between information items if these exist because of the context in which the protocol is used. Therefore, for our model we define unlinkability as follows:

**Definition 5** (Unlinkability). From the perspective of an HBC adversary A, information items  $i_1$  and  $i_2$  are unlinkable iff:

$$i_1 \leftrightarrow i_2 \notin Ded_{\mathcal{A}}$$

Where  $i_1 \leftrightarrow i_2$  is a deduction representing that  $i_1$  and  $i_2$  are linked to the same sender and  $Ded_A$  is the set of all deductions made by A.

#### **III. ADVERSARY MODEL**

In this section we present our formal model of the capabilities of the HBC adversary with respect to undetectability and unlinkability. Our model is defined as a deductive system consisting of a set of inference rules. We first define the components and notation used in our model and then proceed to present our deductive system in Section III-B. In Section III-C we describe the set of inference rules that form the core of our deductive system.

# A. Components and Notation

Since we model undetectability and unlinkability as reachability problems, the fundamental components of our model are similar to those used in other reachability-based methodologies such as the analysis of the security security properties of secrecy and authentication. As is usually the case when analysing security properties, we assume ideal representations of the cryptographic primitives. The components and notation used in our model are listed in Table I. Two important components of our model that are not widely used in similar models are the anonymous and probabilistic encryption primitives as explained below:

1) Probabilistic Encryption: Probabilistic encryption, as introduced by Goldwasser and Micali [40] describes an encryption scheme with the property that: "Whatever is efficiently computable about the cleartext given the ciphertext, is also efficiently computable without the ciphertext." Probabilistic encryption introduces a degree of randomness into the encryption scheme so that multiple encryptions of the same message with the same key will result in different ciphertexts. In order to be considered semantically secure [40][41], an encryption scheme

TABLE I	
COMPONENTS AND NOTATION USED IN THE M	AODEL

Notation	Meaning
$\mathcal{A}$	<b>HBC</b> Adversary: This represents the identity of the agent taking the role of the HBC adversary.
$i_1 \ \dots \ i_n$	<b>Information items:</b> The atomic components used in the communication protocol. They can represent any piece of information including agent identities, cryptographic keys, plaintext data, ciphertext data or cryptographic hashes.
$m_1 \ \dots \ m_n$	<b>Message:</b> A sequence of one or more information items sent as a single unit in the protocol. We assume that a message is sent by a single sender.
$E(k, i_1)$	<b>Encryption (symmetric &amp; asymmetric):</b> The encryption of information item $i_1$ with key $k$ using either symmetric or asymmetric encryption. In the case of asymmetric encryption, the corresponding decryption key will be specified.
$E_P(k, i_1)$	<b>Probabilistic Encryption (symmetric &amp; asymmetric):</b> The probabilistic encryption (as explained in Section III-A1) of information item $i_1$ with key k using either symmetric or asymmetric encryption.
$E_A(k,i_1)$	Anonymous Encryption (asymmetric): The anonymous asymmetric encryption (as explained in Section III-A2) of information item $i_1$ with key k.
$E_{AP}(k,i_1)$	Anonymous Probabilistic Encryption (asymmetric): The anonymous probabilistic asymmetric encryption of information item $i_1$ with key k.
$H(i_1)$	<b>Cryptographic Hash:</b> A one-way cryptographic hash of information item $i_1$ . The hash function is fully deterministic in that the hash of the same information item will always result in the same value.
$detect[i_1]$	<b>Detection:</b> A deduction that can be made by the HBC adversary based on the inference rules. It represents the detection of information item $i_1$ .
$linkable[i_1]$	<b>Linkable:</b> A deduction that can be made by the HBC adversary based on the inference rules. It represents that the information item $i_1$ can be linked to other items but has not yet necessarily been linked.
$i_1 \leftrightarrow i_2$	<b>Link:</b> A deduction that can be made by the HBC adversary based on the inference rules. It represents a link between information items $i_1$ and $i_2$ .

must be probabilistic. This concept can be applied to both symmetric and asymmetric encryption. Probabilistic encryption is important when considering undetectability and unlinkability. If a deterministic encryption scheme is used, an adversary who observes the same ciphertext multiple times could deduce that these represented the same message encrypted under the same key even though the messages cannot be decrypted. If a probabilistic scheme were used in this scenario, the adversary would observe multiple different ciphertexts. Some encryption schemes, such as ElGamal and Paillier, are probabilistic by default whereas others, such as RSA, can be made probabilistic by adding randomized padding, such as the Optimal Asymmetric Encryption. We denote probabilistic encryption using the  $E_P(...)$  notation.

2) Anonymous Encryption: Anonymous encryption refers to an encryption scheme which, in addition to the usual security properties, also provides anonymity or key-privacy as described by Bellare et al. [43]. In the context of a publickey encryption scheme, the adversary will usually have access to the set of public keys with which a message could have been encrypted. In an anonymous encryption scheme, the adversary is unable to determine which of these keys was used to encrypt the message. Symmetric encryption schemes satisfy this property by default because the adversary should not have access to the secret key. As described by Kohlweiss et al. [44], this property is highly relevant for receiver unlinkability and receiver anonymity since the adversary is unable to deduce the intended recipient of an encrypted message. However, we argue that this is also relevant to sender unlinkability in a more general asymmetric encryption scheme since it ensures that an adversary is unable to make linkability deductions about two messages that may have been encrypted under the same key. We denote anonymous encryption using the  $E_A(...)$ notation and the combination of anonymous and probabilistic encryption using the  $E_{AP}(...)$  notation.

#### B. Deductive System

In this context, we use the term *deductive system* to refer to a set of axioms and a set of inference rules that are together used to produce logical deductions. We define our HBC adversary model as a deductive system of this type. At any point in time, our set of axioms consists of the initial knowledge of the adversary as well as any knowledge the adversary has gained be receiving messages in the protocol. These axioms are contained within the adversary's view of the protocol ( $\mathcal{V}_{\mathcal{A}}$ ) as defined in Section II-A. Based on the content of  $\mathcal{V}_{\mathcal{A}}$ , the adversary applies the inference rules in an attempt produce new deductions. A particular inference rule can be applied if all the premises of the rule are satisfied by the current contents of  $\mathcal{V}_{\mathcal{A}}$ . The resulting deductions from a successful application of an inference rule are also added to  $\mathcal{V}_{\mathcal{A}}$  and can therefore be used to satisfy the premises of other inference rules. For example, due to the transitive nature of the link relationship, the adversary can create a graph of links based on multiple deductions in order to test for the existence of a path between items that are supposedly unlinkable. The inference rules in this deductive system are formal representations of the definitions of undetectability and unlinkability and are described in detail in the next section. The aim of our deductive system is to be able to determine whether the property of undetectability (as defined in Section II-B) holds for a particular information item and/or whether the property of unlinkability (as defined in Section II-C) holds for a pair of information items.

#### C. Inference Rules

We have developed a set of inference rules to represent the capabilities of the HBC adversary in terms of detectability and linkability. In this section we present these rules and explain how each rule is derived from the commonly accepted definitions discussed in Section II. Since each rule is a faithful representation of these definitions and since these rules can only be combined in a valid manner in our deductive system, we therefore claim that our overall model is a faithful representation of these definitions. In cases where multiple rules could be applicable, the most specific rule takes precedence.

**Inference Rule 1** (Message Sub-terms). If the adversary A receives a message  $m_1$  consisting of information items  $i_1$  to  $i_n$  then:

$$\frac{m_1 = \langle i_1 \dots i_n \rangle}{detect[i_1] \dots detect[i_n]; \ i_1 \leftrightarrow \dots \leftrightarrow i_n \leftrightarrow m_1}$$

This first rule captures the inferences that the HBC adversary can make from a single message. As defined in Table I, a message is a sequence of one or more information items originating from a single sender. The adversary can detect each of the sub-terms in the message and can link these sub-terms to each other and to the message itself. In terms of the commonly accepted definitions from Pfitzmann and Hansen [5], these information items are detectable because the adversary directly observes their existence and linkable because they are related to each other by a common sender.

**Inference Rule 2** (Symmetric Decryptable). If the adversary A receives a message  $m_1$  encrypted with symmetric key k which is known to A then:

$$\frac{E(k, m_1); \ k}{detect[m_1]; \ detect[k]; \ m_1 \leftrightarrow k}$$

If the adversary receives a symmetrically encrypted message that it can decrypt, the decrypted message is detected and all the sub-terms of this message are also detected according to inference rule 1. The correct decryption of this message confirms that the message must have been encrypted with key k and thus  $\mathcal{A}$  can detect k. The decrypted message  $m_1$  is linked to the symmetric key because both items must have been known by the agent who created the encryption. This rule complies with the definitions [5] in that the detected items can be directly observed and the linked items are related by a common sender.

**Inference Rule 3** (Symmetric Undecryptable). If the adversary A receives a message  $m_1$  encrypted with symmetric key k which is not known to A then:

$$\frac{E(k,m_1); \quad \neg k}{detect[E(k,m_1)]; \quad linkable[E(k,m_1)]}$$

If the adversary receives a symmetrically encrypted message that it cannot decrypt, only the ciphertext term is detected since this is the only item that can be observed. If the  $E(k, m_1)$  term were a sub-term of a larger message, it could be linked to the other parts of the message according to inference rule 1.

**Inference Rule 4** (Probabilistic Symmetric Undecryptable). If the adversary A receives a message  $m_1$  encrypted using probabilistic symmetric encryption with key k which is not known to A then:

$$\frac{E_P(k,m_1); \neg k}{detect[E_P(k,m_1)]}$$

If the adversary receives a message encrypted using symmetric probabilistic encryption that it cannot decrypt, only the ciphertext term is detected. Unlike inference rule 3, the adversary cannot use this term in any links because of the probabilistic encryption. Even if  $\mathcal{A}$  were to receive the same encrypted term again,  $\mathcal{A}$  would not be able to tell that it was the same term without decrypting the message.

**Inference Rule 5** (Asymmetric Decryptable). If the adversary  $\mathcal{A}$  receives a message  $m_1$  encrypted with asymmetric key  $k_e$  for which the decryption key  $k_d$  is known to  $\mathcal{A}$  then:

$$\frac{E(k_e, m_1); \ k_d}{detect[m_1]; \ detect[k_e]; \ m_1 \leftrightarrow k_d}$$

If the adversary receives an asymmetrically encrypted message that it can decrypt, the decrypted message  $m_1$  is detected and all its sub-terms are also detected according to inference rule 1. These terms are detected because they can be directly observed [5]. Since  $\mathcal{A}$  can correctly decrypt the  $m_1$  using the decryption key  $k_d$ ,  $\mathcal{A}$  can be sure that the message was encrypted using the corresponding encryption key  $k_e$ . Therefore,  $\mathcal{A}$  can be certain that  $k_e$  exists and thus  $k_e$  is detected (although  $k_e$  is not known to  $\mathcal{A}$  and cannot be directly observed by  $\mathcal{A}$ ). The decrypted message  $m_1$  is linked to  $k_e$ because both items must have been known to the agent who created the encryption and are therefore related.

**Inference Rule 6** (Asymmetric Undecryptable). If the adversary A receives a message  $m_1$  encrypted with asymmetric key  $k_e$  for which the decryption key  $k_d$  is not known to A then:

$$\frac{E(k_e, m_1); \quad k_{\mathcal{A}}; \quad k_{\mathcal{A}} = k_e; \quad \neg k_d}{detect[E(k_e, m_1)]; \quad detect[k_e]; \quad E(k_e, m_1) \leftrightarrow k_e}$$

If the adversary receives an asymmetrically encrypted message that it cannot decrypt, the ciphertext term is detected since this can be directly observed. Since anonymous encryption has not been used, it is possible for the adversary to determine which key out of a known set of keys was used as explained in Section III-A2. As shown above, if  $\mathcal{A}$  knows the key  $k_{\mathcal{A}}$ which is the same key used for encryption ( $k_{\mathcal{A}} = k_e$ ) then  $\mathcal{A}$  can detect that  $k_e$  exists and can link  $k_e$  to the ciphertext term.

**Inference Rule 7** (Anonymous Asymmetric Undecryptable). If the adversary A receives a message  $m_1$  encrypted using anonymous asymmetric encryption with key  $k_e$  for which the decryption key  $k_d$  is not known to A then:

$$\frac{E_A(k_e, m_1); \quad k_{\mathcal{A}}; \quad k_{\mathcal{A}} = k_e; \quad \neg k_d}{detect[E_A(k_e, m_1)]; \quad linkable[E_A(k_e, m_1)]}$$

If the adversary receives a message encrypted using anonymous asymmetric encryption that it cannot decrypt, only the ciphertext term is detected since this can be directly observed. Unlike inference rule 6,  $\mathcal{A}$  can neither detect the existence of the encryption key  $k_e$  nor use this key as the basis for links as explained in Section III-A2. The adversary can still use the ciphertext term  $E_A(k_e, m_1)$  as the basis for further links because the encryption is not probabilistic.

**Inference Rule 8** (Probabilistic Asymmetric Undecryptable). If the adversary  $\mathcal{A}$  receives a message  $m_1$  encrypted using probabilistic asymmetric encryption with key  $k_e$  for which the decryption key  $k_d$  is not known to  $\mathcal{A}$  then:

$$\frac{E_P(k_e, m_1); \ k_{\mathcal{A}}; \ k_{\mathcal{A}} = k_e; \ \neg k_d}{\det etect[E_P(k_e, m_1)]; \ detect[k_e]; \ linkable[k_e]}$$

If the adversary receives a message encrypted using probabilistic asymmetric encryption that it cannot decrypt, the ciphertext term is detected since this can be directly observed. As in inference rule 6, the adversary can detect that the encryption key  $k_e$  exists by matching it to a key  $k_A$  which is already known by the adversary. Unlike inference rules 6 & 7, the adversary cannot use the undecryptable ciphertext as the basis for any links due to the use of probabilistic encryption as explained in Section III-A1. However, the adversary can still use the detected encryption key  $k_e$  as the basis for links as this is not covered by the probabilistic encryption.

**Inference Rule 9** (Anon. Prob. Asymmetric Undecryptable). If the adversary  $\mathcal{A}$  receives a message  $m_1$  encrypted using anonymous probabilistic asymmetric encryption with key  $k_e$ for which the decryption key  $k_d$  is not known to  $\mathcal{A}$  then:

$$\frac{E_{AP}(k_e, m_1); \ k_{\mathcal{A}}; \ k_{\mathcal{A}} = k_e; \ \neg k_d}{detect[E_{AP}(k_e, m_1)]}$$

If the adversary receives a message encrypted using anonymous probabilistic asymmetric encryption that it cannot decrypt, only the ciphertext term is detected since this is the only item that can be directly observed. Due to the use of anonymous encryption, the adversary cannot detect the encryption key  $k_e$  nor use it as the basis for links. The use of probabilistic encryption makes it impossible for the adversary to use the ciphertext as the basis for links.

**Inference Rule 10** (Known Hash). If the adversary A receives a cryptographic hash of an information item  $i_1$  and the value of  $i_1$  is known to A then:

$$\frac{H(i_1); i_{\mathcal{A}}; i_{\mathcal{A}} = i_1}{detect[H(i_1)]; detect[i_1]; H(i_1) \leftrightarrow i_1}$$

If the adversary receives a cryptographic hash of an information item, the hash value is detected since it can be directly observed. If the adversary already knows an information item  $i_A$  which is the same as  $i_1$ , the adversary can conclude that this is a hash of  $i_1$  and so link  $i_1$  to the hash value. This also allows the adversary to detect that  $i_1$  exists. For this rule, the assumption of ideal cryptographic primitives precludes hash collisions. The hash value can be used as the basis of links because the hash function is defined as a deterministic function. For example, provided the same hash algorithm is used, the hash of a particular information item will always result in the same value and this can be used to establish links between different messages if the hash of a particular information item is used more than once.

**Inference Rule 11** (Unknown Hash). If the adversary A receives a cryptographic hash of an information item  $i_1$  and the value of  $i_1$  is not known to A then:

$$\frac{H(i_1); \ \neg i_{\mathcal{A}}; \ i_{\mathcal{A}} = i_1}{detect[H(i_1)]; \ linkable[H(i_1)]}$$

If the adversary receives a cryptographic hash of an information item and does not already know any information item  $i_{\mathcal{A}}$  such that  $i_{\mathcal{A}} = i_1$  then only the hash value is detected since it can be directly observed. As in inference rule 10, the value of the hash can be used as the basis for links because the hash function is deterministic.

User-Defined Inference Rules Since we are using a constructive definition of the HBC adversary's capabilities, the user is able to define and add new inference rules to our model. These user-defined inference rules are usually specific to the protocol being analysed and are used to capture symbolic information that would not otherwise be included in the model. For example, the user-defined rules can be used to model mathematical equivalences. In one of the protocols for smart grid communication that we analyse in Section V, we use this capability to model the additive property of energy measurements from smart meters. Although the half-hourly energy consumption measurements are represented as distinct information items, the summation of all half-hourly measurements from a single user is equal to the total consumption measurement for that user. Provided that the half-hourly measurements can all be linked to each other, their summation can be linked to the total consumption measurement using a user-defined inference rule. In comparison to analyses based on observational equivalence, we argue that it is significantly easier to capture such behaviour in the form of user-defined inference rules in our constructive reachability-based approach.

## IV. INTEGRATION WITH EXISTING METHODS

In an observational equivalence approach, the aim is to determine if specific traces of events are equivalent from the perspective of the HBC observer. When this is decidable, it proves *whether* a specific information item is detectable and/or whether it can be linked with another information item. In contrast, our constructive reachability-based definition of the HBC adversary's capabilities can be used to determine precisely *how* the undetectability or unlinkability property is violated. This paradigm is broadly similar to the approach used in the analysis of security properties such as secrecy and authentication. Therefore, one of the advantages of this approach is that our model can be integrated with existing methodologies for analysing security properties. To demonstrate this, we have integrated our model with the Casper/FDR

TABLE II New Casper/FDR Specifications

#Privacy

```
– Undetectable (Agent, {Information Items})
Undetectable (A, {X, Y})
```

- Unlinkable (Agent, {Items}, {Exclude}, {Extra}) Unlinkable (A, {X, Y}, {K}, { ({M, N}, {Y}) })

tool developed by Lowe [45]. In the following subsections we describe the original Casper/FDR tool and our enhanced version. Our model is by no means restricted to integration with the Casper/FDR tool or with CSP and we believe that it should be possible to integrate our model into various methods and tools used in protocol analysis.

## A. Casper/FDR Tool

There is an established method developed by Roscoe [46] and Lowe [47] for verifying the security properties of protocols (secrecy and authentication) using the process algebra CSP [48] and its model checker FDR [49]. As a process algebra, CSP provides a formal method for modelling concurrent systems. In CSP, a system is modelled as a set of processes in which each process can perform specific events. Processes can be composed in parallel and synchronized on specific events such that these events can only occur when all synchronized processes are ready to perform them. The possible sequences of events of a process or a parallel combination of processes are referred to as the event traces. In the existing analysis method [46][47], the communication protocol is first modelled in CSP and then the FDR tool is used to perform a trace refinement on the system. This ensures that no possible event trace will violate the specified security properties. If any property is violated, FDR outputs the relevant event trace as a counter-example. The Casper/FDR tool developed by Lowe [45] greatly simplifies this analysis by compiling descriptions of protocols (e.g. in so-called Alice & Bob notation) into CSP models and interpreting the FDR output. Further background about CSP and the Casper/FDR tool is available in recent online references [50][51]. It should be noted that the analysis method used in this tool is limited to bounded analysis of protocols. Whilst this does not affect protocols in which an attack is found, it limits the generality of the claims that can be made when no attack is found. In general, our adversary model can be used in either bounded or unbounded analysis but in the current work, our integration with this particular tool means that the overall analysis will be bounded.

## B. Enhanced Casper Specifications

In our enhanced version of Casper/FDR, we have defined new specifications for undetectability and unlinkability properties These can be added to the input script as a separate section under the heading #Privacy. Each specification is fully described by a single line as shown in Table II. The undetectability specification begins with the keyword Undetectable and takes two parameters, an agent and a set of information items. The first specifies the agent who will take the role of the HBC adversary. The second parameter lists the information items that are supposed to be undetectable by the HBC adversary and can contain any information items defined in the protocol (e.g. identifiers, data items or keys). The undetectability specification will fail if the HBC agent can detect one or more of the specified information items. For example, the specification in Table II will fail if agent A detects either of the information items X or Y.

The unlinkability specification begins with the keyword Unlinkable and takes four parameters. The first parameter is the HBC agent who will attempt to establish a definitive link between all the information items specified in the second parameter. The third parameter is a list of information items that should be excluded from the linking algorithm. This is used to represent information items that could be shared by multiple agents (e.g. shared keys) which should not be used as the basis of links. The fourth parameter allows the user to specify additional inference rules that the HBC adversary can use in that specific protocol. These are specified as tuples containing a left set and a right set. If the HBC adversary can establish a definitive link between all the information items in the left set, then the adversary can deduce a definitive link between all items in the union of both sets. In Table II, if the adversary can link information items M and N  $(M \leftrightarrow N)$ , then links between items M, N, and Y can be inferred  $(M \leftrightarrow N \leftrightarrow Y)$ . These extra links are used to represent higher-level relationships that exist for specific protocols. For example, if M and N are the first and second halves of a password and the adversary is sure that M and N originated from the same sender (i.e. they are linked), then the adversary will have learned the full password Y and can also link Y to that sender. This can be used to model a reply to a message sent via an anonymous channel. For example, if an agent sends a request to the adversary via an anonymous channel that allows the adversary to send a reply, the adversary could generate a unique information item and send it to the agent. If the adversary later received a message containing this unique item, it could be used to establish a link to the agent's original request. To model this example, the user would provide an explicit link between the agent's request and the adversary's unique response using this fourth parameter. The unlinkability specification will fail if the honest-but-curious agent can establish a definitive link between all the information items in the second parameter taking into account all the exclusions and extra links.

In contrast to the existing Casper/FDR secrecy and authentication specifications, our new specifications deal with the actual variables in the system rather than the free variables. The specifications can therefore be used in systems where multiple actual agents take the same role or where multiple repetitions of the protocol take place. The Casper/FDR compilation step has been enhanced to automatically compile these new specifications into our new CSP model and integrate the

TABLE III ENHANCED CASPER SPECIFICATIONS

- Original encryption specification 1. $a \rightarrow b$ : {message}{key}
- Anonymous encryption 1. $a \rightarrow b$ : {message}{key-A}
- Probabilistic encryption 1. $a \rightarrow b$ : {message}{key-P}
– Anonymous probabilistic encryption 1. $a \rightarrow b$ : {message}{key-AP}

result with the existing Casper/FDR output.

We have also enhanced certain existing Casper/FDR specifications. In particular we have augmented the encryption specification to model anonymous encryption as well as probabilistic encryption and any combination thereof as shown in Table III. For the purpose of this work we therefore consider the original specification to represent encryption that is neither anonymous nor probabilistic. In our enhanced Casper/FDR specifications, the user can model probabilistic or deterministic encryption by adding or omitting the -P qualifier after the key. In the same way, the user can model anonymous encryption using the -A qualifier or combine these to represent anonymous and probabilistic encryption by using the -AP qualifier.

# C. Implementation in CSP

This section describes the implementation of our deductive system and inference rules in CSP and their integration with the Casper/FDR tool. Our implementation is similar to the approach used to model the external intruder in the existing analysis method [46][47]. The main steps in this approach are shown in Figure 1.

In the first phase, the system constructs the overall set of possible deductions.

In CSP we represent a *deduction* (i.e. the result of successfully applying an inference rule) as a tuple of a single fact, the subject, and a set of facts from which the subject can be deduced. In this context, we use the term *fact* to refer to any information item in the protocol or a statement about one or more information items. For example, we define the fact Detect.X to represent the detection of information item X and the fact Link (Y, Z) to represent a link between items Y and Z. For each fact (i.e. an information item or a deduction), the implementation calculates the set of messages from which the fact can be learned and the sets of other facts from which it can be deduced. The implementation also determines the set of deductive rules in which each fact could be used.

In the second phase, the implementation constructs the HBC process, a parallel combination of processes to represent the state of all facts in the system from the perspective of the HBC adversary. Each unknown fact is initially modelled as a process in the UNKNOWN state. These processes can transition to the KNOWN state either through receiving a message containing



Fig. 1. Overview of the deductive system

the information item or through the successful application of an inference rule. Once in the KNOWN state, the process is willing to participate in any relevant deduction event. These processes are synchronized on all deduction events so that the deductions will only succeed if all the required facts are known. Processes in the KNOWN state representing a Detect.X or Link. (X, Y) fact can generate an HBC event if the information items they represent have been specified as undetectable or unlinkable.

The HBC process is then placed in parallel with Casper/FDR's existing SYSTEM process that represents the behaviour of honest agents in the protocol. These two processes are synchronized on events performed by the agent in the role of the HBC adversary. In the final phase of this system, trace refinement is used to determine if any sequence of events violates an undetectability or unlinkability specification.

#### V. ANALYSES

To demonstrate the effectiveness of our adversary model and our enhanced Casper/FDR tool, we have analysed a number of protocols in terms of both security properties (secrecy and authentication) and privacy properties (undetectability and unlinkability). In Tables IV & V we summarize the results of our analyses. Following from our motivating example, the protocols shown in Table V are all related to the smart energy grid. For each protocol we present a overview of the main aims and fundamental features of the approach. We then briefly describe the results of our analysis of the security and privacy properties. The security properties are analysed using the Dolev-Yao adversary from the existing methodology whilst the privacy properties are analysed using our HBC adversary. Based on our analyses, we can begin to define classes of attacks which rely on similar mechanisms and can therefore probably be solved using similar approaches. This type of classification can be viewed as an initial step towards establishing protocol design anti-patterns in terms of security and privacy properties. From our analyses, the following classes can be defined:

## A. Direct information leak

In the most basic class (which we describe as a flaw rather than an attack) a protocol directly provides the HBC adversary with private information. This is the case in the TAS<sup>3</sup> Attribute Aggregation Protocol [52], leading to the undetectability flaw described by Veeningen et al. [6] and the further unlinkability flaw identified by our tool. This is also the case for our analysis of the OpenADR 2.0 Standard [57]. Although this standard was not designed to provide privacy properties, it nevertheless directly reveals private information about the user. In these cases, the most straightforward amelioration is to modify the protocol to avoid this direct leak.

# B. Indirect information leak

A more subtle class of attack is where information is inadvertently or indirectly leaked to the adversary. The attack on the French implementation of the BAC protocol in epassports, as described by Arapinis et al. [9] and confirmed by our analysis, falls into this category. Once the indirect leak has been identified, the solution will likely involve reducing the specificity of that part of the protocol to avoid inadvertently leaking information.

# C. Pseudonyms

A third class of attack is based on the use of pseudonyms to provide anonymity. This is applicable to the first and second smart grid protocols in Table V in which pseudonyms are used to provide authentication whilst supposedly maintaining anonymity. However, due to the possibility of linking information items using these pseudonyms, our analyses show that they are the root cause of the privacy attacks we have discovered. As suggested by Finster and Baumgart [55], a possible solution to this class of attack is to frequently renew the pseudonyms in an unlinkable manner.

 TABLE IV

 Summary of general protocols analysed and results obtained.

Protocol Overview	Security Properties	Privacy Properties
<b>TAS<sup>3</sup></b> Attribute Aggregation Protocol [52]: The purpose of this protocol is to allow a user (U) to supply a service provider $(SP)$ with identity-related attributes from multiple identity providers $(IdPs)$ in a single session without necessarily having to authenticate to each provider during the session. To achieve this, the protocol introduces a linking service (LS). Once U has been authenticated by the first identity provider $IdP_1$ , the SP receives the address of the LS and a token for the LS from $IdP_1$ . The SP contacts the LS which responds with the address of $IdP_2$ and another authorization token. The SP then contacts $IdP_2$ to obtain further attributes about U [52]. Veeningen et al. [6] have analysed this section of the protocol in terms of undetectability and unlinkability.	In this protocol, all communication takes place using Transport Layer Security (TLS) and strong identity guarantees are provided by each of the communicating nodes. We have used our enhanced tool to analyse various security properties including the secrecy of data items and identifiers and the authentication of the communicating parties. Our analysis does not reveal any compromises of these properties in the presence of an external DY adversary.	Veeningen et al. [6] define three unde- tectability properties and one unlinkability property for this protocol. Their analysis shows that two undetectability properties (P2 and P3 in [6]) do not hold, allowing both the $LS$ and $IdP_2$ to detect and ob- serve attributes of $U$ which should only be available to the $SP$ . Our analysis produces the same results for all four properties and in addition, shows that both the $LS$ and $IdP_2$ can link the attributes they have detected to a specific user, thus constituting a serious privacy flaw.
<b>Unlinkability of RFID e-Passports [9]:</b> Arapinis et al. [9] have used the applied pi calculus to analyse one of the RFID communication protocols used in e-passports. The Basic Access Control (BAC) protocol is a four-message protocol designed to established a shared session key between the RFID reader and the tag in the passport. Before the protocol begins, the reader scans the optical data on the passport to obtain the tag's long-term encryption key $k_e$ and message authentication key $k_m$ . Using $k_e$ and $k_m$ , the parties exchange encrypted messages and message authentication codes (MACs). Arapinis et al. [9] have described a linkability flaw in the BAC protocol that could allow an adversary to identify and track the passport over RFID without the consent of the user. It must be noted that the exploitation of this flaw requires an active adversary so for this analysis we have manually extended the capabilities of our adversary model to recreate the analysis in [9].	For this protocol, the primary security re- quirements with respect to an external DY adversary are the secrecy of the exchanged keys and the authentication between the tag and the reader. As in the analysis in [9], we assume that $k_e$ and $k_m$ are shared by the tag and the reader over a secure channel. Our analysis confirms that an external DY adversary is unable to compromise the se- curity properties of this protocol without first learning these long-term secret keys.	As described in [9], the adversary records the third RFID message $(m_3)$ from a le- gitimate run of the protocol for a target passport. This message from the reader to the tag contains the original nonce gener- ated by the tag and a MAC using $k_m$ . The adversary then runs the protocol with any tags in range and replays $m_3$ . The tags first check the MAC and then the nonce in $m_3$ and will send an error message if either check fails. Critically, in the French implementation of this protocol, the tag produces different error messages for each check. As confirmed by our analysis, the nonce check will always fail but if the MAC check passes the adversary learns that this is the target passport.
<b>Protecting location privacy with k-anonymity [53]:</b> Gedik and Liu [53] have proposed a protocol to enhance privacy in location-based services (LBS). In an LBS, a user sends his or her current location to a service provider $(SP)$ in order to receive some information or service relevant to that particular location. However, users do not always trust the $SP$ and so this agent should be represented as a semi-honest adversary. The aim of this protocol is to prevent the $SP$ from linking the submitted location information to a specific user. In this protocol, the users send their requests containing their real identities and precise locations to a trusted anonymity server. This server implements a spacial cloaking algorithm before forwarding parts of the requests to the $SP$ . Even though it is assumed that no user identities are required by the $SP$ , there is still a risk that the $SP$ could link multiple requests together to create location patterns of specific users. Auxiliary information could then be used to link these location patterns to named users.	Gedik and Liu [53] explain that in this protocol, the users are assumed to have a secure connection to the trusted anonymity server. The users identify themselves to this server using their real identities $(u_{id})$ and it is assumed that the communication could take place over TLS connections. It is also assumed that communication between the anonymity server and the LBS <i>SP</i> could take place over TLS because the privacy properties are not affected by the untrusted <i>SP</i> learning the identity of the anonymity server. Based on these reasonable assump- tions, our analysis does not reveal any com- promises of the security properties with respect to an external DY adversary.	The two main privacy requirements in this protocol are that the untrusted $SP$ should not be able to detect identifying information for individual users and that $SP$ should not be able to link multiple requests together. Our analysis confirms that this protocol achieves these objectives with respect to a semi-honest $SP$ . In particular, the step taken by the trusted anonymity server of replacing the user identity $(u_{id})$ and the request number $(r_{no})$ with a random string before sending the request to $SP$ is critical to the privacy properties. However, further analysis shows that the anonymity server could compromise both these privacy properties. It is therefore critical that this service is provided by a trustworthy entity.

# D. Insufficient authentication

In some cases, an overemphasis on privacy properties could introduce attacks against the security properties of the protocol. This is the case in the third smart grid protocol in Table V. Our analysis has shown how a single compromised smart meter can remain anonymous whilst invalidating all measurements from the group due to the use of group identifiers. A possible solution would be to use some form of individual anonymous authentication. This type of attack demonstrates the need for combined verification of security and privacy properties as facilitated by our enhanced tool.

# VI. RELATED WORK

Formal analysis of the security properties of communication protocols is an established area of research. In addition to the work by Roscoe [46] and Lowe [47][45] using CSP and the Casper/FDR tool [45] there have been other research efforts in this field [61][62][63][64].

There have also been related research efforts to formalize

TABLE V Summary of smart grid protocols analysed and results obtained.

Protocol Overview	Security Properties	Privacy Properties
Smart Meter Anonymization through Pseudonyms [36]: In this protocol, the authors use two unlinkable identifiers for reporting energy measurements from a smart meter to an energy utility. The high-frequency identifier $(HFID)$ is a pseudonym for reporting frequent measurements and the low-frequency identifier $(LFID)$ , which contains the user's personal information, is used for infrequent communication such as reporting the total monthly consumption. The link between an $HFID$ and the corresponding $LFID$ should only be known by a trusted third party. However, in their analysis of this protocol, Jawurek et al. [54] have described multiple ways in which it could be possible to link an $HFID$ to a real user based on correlation with secondary data sources.	In this protocol, the authors propose the use of asymmetric cryptography and digital certificates from a mutually trusted certificate authority (CA) in order to maintain the security properties. In particular, each smart meter has a separate certificate and key pair for the $HFID$ and the $LFID$ . Our analysis does not reveal any compromises in terms of the secrecy of the messages or the authentication of the communicating entities in the presence of an external DY adversary.	We have analysed the unlinkability be- tween a user's $HFID$ and $LFID$ from the perspective of the energy utility and shown that this fundamental claim does not hold. The utility can use the $HFID$ as a pseudonym to link high frequency mea- surements and obtain the total consumption for an $HFID$ as the summation of these values. Since these measurements are suf- ficiently detailed, the utility can uniquely match this total to the values reported us- ing the $LFID$ and thus de-anonymize the user's energy measurements.
<b>Pseudonymous Smart Metering without a Trusted</b> <b>Third Party [55]:</b> Finster and Baumgart [55] use a sim- ilar approach to the protocol above to anonymize energy measurements. In their protocol, each smart meter (S) generates an asymmetric key pair { $S_{Public}, S_{Private}$ }. Initially S authenticates itself to the energy utility and sends a cryptographically blinded version of $S_{Public}$ to be signed. After unblinding the result, S uses this as a pseudonym to report high-frequency measurements. Sim- ilarly to the above protocol, low-frequency measurements are still reported using the user's real identity. Again the fundamental requirement is that the adversary should not be able to link a specific $S_{Public}$ to a particular user. A proposed option in this protocol is that these pseudonyms could be re-issued on a more frequent basis (e.g. daily).	Similarly to the protocol above, the authors use asymmetric cryptography and digital certificates to encrypt and sign the mea- surement data. The high-frequency mea- surements are signed by the smart meter's private key $S_{Private}$ and can be verified using $S_{Public}$ which is in term signed by the energy utility. Our analysis does not reveal any compromises in terms of the secrecy of the messages or the authenti- cation of the communicating entities in the presence of an external DY adversary.	Using our model, we have analysed the un- linkability between $S_{Public}$ and the user's identity $(ID_U)$ from the perspective of an HBC energy utility. As above, the energy utility can establish links between the high- frequency consumption values based on the pseudonym $S_{Public}$ . If all the measure- ments in a billing period can be linked, the total can be linked to $ID_U$ as before thus violating the unlinkability require- ment. However, further analysis shows that if the pseudonyms are changed during the billing period, it would not be possible to link all the required high-frequency mea- surements and thus the desired unlinkabil- ity property would be maintained.
Smart Meter Anonymization through Group Identi- fiers [37]: The authors describe a generalized represen- tation of a privacy-enhancing protocol for smart meter communication based on anonymity networks. Similarly to the protocols above, they differentiate between high- frequency anonymized information and low-frequency identifiable information. A unique customer identifier (IdC) is used for low-frequency messages whilst an anonymous identifier $(IdG)$ is used for high-frequency messages. To avoid the pseudonym attacks above, they use one $IdG$ to represent a group of users thus making IdG a group identifier. The adversary should not be able to connect any high-frequency measurements to a specific IdC.	Through our combined security and privacy analysis, we have discovered an at- tack against the security properties of this protocol. We make the realistic assumption that a Dolev-Yao adversary controls one of the smart meters in the system (e.g. using available open-source tools [56]). Since IdG is shared by all smart meters, this adversary can generate and send multiple falsified high-frequency measurements in each reporting period, thus invalidating the legitimate reporting from the whole group. As confirmed by the unlinkability property, it would not be possible for the utility to identify which meter from the group had been compromised.	In this protocol, our analysis of the privacy properties does not reveal any compro- mises. We have confirmed that measure- ments reported using the group identifier (IdG) cannot be linked to any unique customer identifier $(IdC)$ by our HBC adversary without additional auxiliary in- formation. However, this strong anonymity guarantee exacerbates the consequences of the security flaw discovered in this protocol as described in the previous column.
<b>OpenADR 2.0 Standard [57]:</b> In the smart grid, the term demand response (DR) describes a set of actions to dynamically reduce energy demand at specific times and locations. DR differs from smart metering but will generally use bi-directional communication between smart meters and the energy supplier. Incentive-based DR schemes offer consumers some incentive to voluntarily participate in demand response events. These schemes usually involve a bidding process in which consumers indicate the amount by which they are currently willing to reduce their consumption. These bids are accepted until the required amount of demand reduction has been achieved. OpenADR is a communication data model that can be used for incentive-based DR [57]. This standard introduces the concept of the Demand Response Automation Server (DRAS) which receives bids from the users and forwards them to the energy supplier [58]. In these systems, all communicating nodes provide strong identity guarantees.	In order to maintain the desired security properties, the OpenADR protocol uses strong identity guarantees from all of the communicating nodes and uses TLS con- nections for all communication between the nodes [59]. Our analysis does not reveal any potential compromises in terms of the secrecy of the messages or the authentica- tion of the communicating entities in the presence of an external DY adversary.	Previous work has described potential pri- vacy concerns in OpenADR based on the bi-directional flow of information [59][60]. In modelling this system, we represent the energy supplier as an HBC adver- sary ( $\mathcal{A}$ ). Given that the DRAS forwards the messages directly to $\mathcal{A}$ , the adver- sary can detect the bid amounts and can link these to individual users. Furthermore, given sufficient auxiliary information such as a database of energy signatures, $\mathcal{A}$ can link these bid amount to specific signatures to learn private information such as which appliances are being used. Our analysis shows that this flaw also exists if the DRAS is an HBC adversary as described in [59].

the concepts of privacy and anonymity and their constituent properties including undetectability and unlinkability. A comprehensive review of the formalization of anonymity has been presented by Wright et al. [8]. There are various examples of formalizations and analyses of privacy properties in the computational paradigm [14][15][16][17][18] [20][19].

There are also various examples of the use of the symbolic paradigm: Mauw et al. have developed a formalization of anonymity in onion routing using trace equivalence [13]. Berthold and Clauss [39] use the technique of Formal Concept Analysis (FCA) to reason about unlinkability. Similarly to our model, they construct links between messages if the messages share information items. The applied pi calculus and the ProVerif tool have been used to verify privacy-type properties for various types of protocols including voting protocols [65][21] and the Direct Anonymous Attestation (DAA) protocol [21][66]. There have been efforts to verify privacy properties in the protocols used by RFID tags [9][10].

In terms of modelling undetectability and unlinkability, the most closely related work is that of Veeningen et al. [6][35]. Our work builds on their definitions of undetectability and unlinkability. However, we extend their definitions to include deductions that could be made by the HBC adversary. Furthermore, we have specifically designed our model to facilitate integration with other analysis methodologies such as the analysis of security protocols.

Fournet and Abadi [67] have presented one of the earliest examples of combining the analysis of security properties with that of privacy properties. They used the applied pi calculus [68] to analyse a *private authentication* protocol. They verified the security properties of authentication and secrecy in addition to a specific privacy property: that external observers are unable to learn the identities of the protocol's participants. In our model, this property can be constructed in terms of undetectability of participant identities and unlinkability between participants and messages. Our model extends this idea to allow for verification of undetectability and unlinkability properties for any information item with respect to any agent. This flexibility allows our model to be applied to a significantly larger set of communication protocols.

More recently, Luu et al. [69] presented *SeVe*, a tool for automatically verifying security properties that has been implemented as a module of the Process Analysis Toolkit (PAT). In their system they model the security properties of secrecy and authentication as well as three privacy properties: anonymity, receipt freeness and coercion resistance. These properties can be verified automatically with respect to an external intruder. Although not explicitly described, it appears that the *SeVe* tool can combine the verification of security and privacy properties for the same protocol and is therefore closely related to our work. However, our model differs in that we consider a different set of privacy properties and that we verify these properties with respect to an internal HBC adversary who is a legitimate participant in the protocol rather than an external intruder.

# VII. CONCLUSION AND FUTURE DIRECTIONS

Automatic analysis of privacy properties such as undetectability and unlinkability is an important challenge, especially as communication protocols increase in complexity. Various approaches have been used to address this challenge including the notions of computational indistinguishability and symbolic observational equivalence. We have demonstrated the feasibility and effectiveness of using a reachability-based approach to model and analyse these properties. Our approach uses similar techniques to those used in the analysis of security protocols such as secrecy and authentication. Our approach is facilitated by the formal model we have developed to represent the capabilities of an HBC adversary with respect to undetectability and unlinkability. We constructively model the adversary's capabilities as a deductive system consisting of a set of inference rules. We show that these rules are based on commonly accepted definitions and we therefore claim that our deductive system is a faithful representation of these definitions. One of the advantages of our approach is that our HBC adversary model can be integrated with existing methodologies for analysing security properties. We have demonstrated this by implementing our model in the process algebra of CSP and integrating it with an established analysis methodology. We have released an enhanced version of the Casper/FDR tool that can model and analyse security and privacy properties based on a single description of the protocol. To demonstrate the effectiveness of our model, we have used this tool to analyse several communication protocols in terms of security and privacy properties. Through our analyses we have found new attacks and rediscovered known attacks.

We propose two main future directions for this work: The first is to extend the set of properties for which the HBC adversary is defined. Although undetectability and sender unlinkability are central to reasoning about concepts such as anonymity and privacy, we can envisage various other properties, such as recipient unlinkability or receipt-freeness, that could also be important in the analysis of specific protocols. The second proposed direction is to use this constructive approach to develop other adversary models. In addition to the HBC adversary and the well-known Dolev-Yao model, we can envisage other adversary models that could be used in the analysis of specific protocols. For example, a slightly stronger version of our HBC adversary might be able to eavesdrop on other communication channels or might be able to compromise long-term keys. The advantage of using this constructive approach is that all these models should theoretically be composable and thus facilitate combined analysis of security and privacy properties with multiple adversary models.

# ACKNOWLEDGEMENTS

This research was conducted as part of the *Future Home Networks and Services* project at the University of Oxford, funded by BT. The authors thank Cas Cremers, Thomas Gibson-Robinson, John Lyle and Joss Wright for the helpful discussions and comments on this work.

#### REFERENCES

- R. Dingledine, N. Mathewson, and P. F. Syverson, "Tor: The Second-Generation Onion Router," in *Proceedings of the 13th USENIX* Security Symposium, 2004, 2004.
- [2] A. Leung and C. Mitchell, "Ninja: Non Identity Based, Privacy Preserving Authentication for Ubiquitous Environments," in *UbiComp* 2007: Ubiquitous Computing, ser. Lecture Notes in Computer Science, J. Krumm, G. Abowd, A. Seneviratne, and T. Strang, Eds. Springer Berlin Heidelberg, 2007, vol. 4717, pp. 73–90.
- [3] E. Cesena, H. Löhr, G. Ramunno, A.-R. Sadeghi, and D. Vernizzi, "Anonymous Authentication with TLS and DAA," in *Trust and Trustworthy Computing*, ser. Lecture Notes in Computer Science, A. Acquisti, S. Smith, and A.-R. Sadeghi, Eds. Springer Berlin Heidelberg, 2010, vol. 6101, pp. 47–62.
- [4] C. Wachsmann, L. Chen, K. Dietrich, H. Löhr, A.-R. Sadeghi, and J. Winter, "Lightweight anonymous authentication with TLS and DAA for embedded mobile devices," in *Proceedings of the 13th international conference on Information security*, ser. ISC'10. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 84–98.
- [5] A. Pfitzmann and M. Hansen, "A terminology for talking about privacy by data minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management," Aug. 2010. [Online]. Available: http://dud.inf.tu-dresden.de/literatur/Anon\_Terminology\_v0.34.pdf
- [6] M. Veeningen, B. Weger, and N. Zannone, "Formal Privacy Analysis of Communication Protocols for Identity Management," in *Information Systems Security*, ser. Lecture Notes in Computer Science, 2011, vol. 7093, pp. 235–249.
- [7] M. Brusó, K. Chatzikokolakis, S. Etalle, and J. Den Hartog, "Linking Unlinkability," in *Proc. International Symposium on Trustworthy Global Computing (TGC)*, Newcastle upon Tyne, Royaume-Uni, 2012.
- [8] J. Wright, S. Stepney, J. A. Clark, and J. Jacob, "Formalizing Anonymity: A Review," University of York, Tech. Rep. June, 2005. [Online]. Available:
- http://eprints.whiterose.ac.uk/72501/1/YCS\_2005\_389.pdf
- [9] M. Arapinis, T. Chothia, E. Ritter, and M. Ryan, "Analysing Unlinkability and Anonymity Using the Applied Pi Calculus," in *Proc. IEEE Computer Security Foundations Symposium (CSF)*, 2010, pp. 107–121.
- [10] M. Bruso, K. Chatzikokolakis, and J. den Hartog, "Formal Verification of Privacy for RFID Systems," in *Proc. IEEE Computer Security Foundations Symposium (CSF)*, 2010, pp. 75–88.
- [11] H. S. Fhom and K. M. Bayarou, "Towards a Holistic Privacy Engineering Approach for Smart Grid Systems," in *Proc. IEEE International Conference on Trust, Security and Privacy in Computing* and Communications, Nov. 2011, pp. 234–241.
- [12] C. Rottondi, G. Verticale, and C. Krauss, "Distributed Privacy-Preserving Aggregation of Metering Data in Smart Grids," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 7, pp. 1342–1354, Jul. 2013.
- [13] S. Mauw, J. H. S. Verschuren, and E. P. Vink, "A Formalization of Anonymity and Onion Routing," in *Computer Security ESORICS* 2004, ser. Lecture Notes in Computer Science, P. Samarati, P. Ryan, D. Gollmann, and R. Molva, Eds. Springer Berlin Heidelberg, 2004, vol. 3193, pp. 109–124.
- [14] P. Syverson, G. Tsudik, M. Reed, and C. Landwehr, "Towards an Analysis of Onion Routing Security," in *Designing Privacy Enhancing Technologies*, ser. Lecture Notes in Computer Science, H. Federrath, Ed. Springer Berlin Heidelberg, 2001, vol. 2009, pp. 96–114.
- [15] S. Steinbrecher and S. Köpsell, "Modelling Unlinkability," in *Privacy Enhancing Technologies*, ser. Lecture Notes in Computer Science, 2003, vol. 2760, pp. 32–47.
- [16] J. Camenisch and A. Lysyanskaya, "A Formal Treatment of Onion Routing," in *Advances in Cryptology CRYPTO 2005*, ser. Lecture Notes in Computer Science, V. Shoup, Ed. Springer Berlin Heidelberg, 2005, vol. 3621, pp. 169–187.
- [17] M. Franz, B. Meyer, and A. Pashalidis, "Attacking Unlinkability: The Importance of Context," in *Privacy Enhancing Technologies*, ser. Lecture Notes in Computer Science, 2007, vol. 4776, pp. 1–16.
- [18] L. Fischer, S. Katzenbeisser, and C. Eckert, "Measuring unlinkability revisited," in *Proc. ACM workshop on Privacy in the electronic society*, ser. WPES '08, 2008, pp. 105–110.

- [19] M. Backes, I. Goldberg, A. Kate, and E. Mohammadi, "Provably Secure and Practical Onion Routing," in 2012 IEEE 25th Computer Security Foundations Symposium. IEEE, Jun. 2012, pp. 369–385.
- [20] M. Backes, A. Kate, P. Manoharan, S. Meiser, and E. Mohammadi, "AnoA: A Framework for Analyzing Anonymous Communication Protocols," in 2013 IEEE 26th Computer Security Foundations Symposium. IEEE, Jun. 2013, pp. 163–178.
- [21] S. Delaune, M. Ryan, and B. Smyth, "Automatic Verification of Privacy Properties in the Applied pi Calculus," in *Trust Management II*, ser. IFIP The International Federation for Information Processing, Y. Karabulut, J. Mitchell, P. Herrmann, and C. Jensen, Eds. Springer US, 2008, vol. 263, pp. 263–278.
- [22] M. Arapinis, V. Cheval, and S. Delaune, "Verifying Privacy-Type Properties in a Modular Way," in 2012 IEEE 25th Computer Security Foundations Symposium. IEEE, Jun. 2012, pp. 95–109.
- [23] IEEE Standards Association, "IEEE 2030-2011: IEEE Guide for Smart Grid Interoperability of Energy Technology and Information Technology Operation with the Electric Power System (EPS), End-Use Applications, and Loads," Tech. Rep. September, 2011.
- [24] National Institute of Standards and Technology (NIST), "NIST Special Publication 1108R2: NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 2.0," Tech. Rep., 2012.
- [25] G. W. Hart, "Residential energy monitoring and computerized surveillance via utility power flows," *Technology and Society Magazine, IEEE*, vol. 8, no. 2, pp. 12–16, Jun. 1989.
- [26] U. Greveler, P. Glösekötter, B. Justus, and D. Loehr, "Multimedia Content Identification Through Smart Meter Power Usage Profiles," 5th International Conference on Computers, Privacy and Data Protection, 2012.
- [27] A. Molina-Markham, P. Shenoy, K. Fu, E. Cecchet, and D. Irwin, "Private memoirs of a smart meter," in *Proceedings of the 2nd ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Building*, ser. BuildSys '10. New York, NY, USA: ACM, 2010, pp. 61–66.
- [28] C. Cuijpers and B.-J. Koops, "Smart Metering and Privacy in Europe: Lessons from the Dutch Case," Feb. 2013.
- [29] E. L. Quinn, "Privacy and the New Energy Infrastructure," SSRN eLibrary, 2009.
- [30] I. Brown, "Britain's Smart Meter Programme: A Case Study in Privacy by Design," *International Review of Law, Computers & Technology*, Feb. 2013.
- [31] NIST Smart Grid Interoperability Panel Cyber Security Working Group, "NISTIR 7628: Guidelines for Smart Grid Cyber Security: Vol. 1, Smart Grid Cyber Security Strategy, Architecture, and High-Level Requirements," Tech. Rep. August, 2010.
- [32] R. Anderson and S. Fuloria, "Who Controls the off Switch?" in Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on, 2010, pp. 96–101.
- [33] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198–208, Mar. 1983.
- [34] O. Goldreich, *Foundations of Cryptography: Volume 2, Basic Applications*. Cambridge University Press, 2009, vol. 2.
- [35] M. Veeningen, B. Weger, and N. Zannone, "Modeling Identity-Related Properties and Their Privacy Strength," in *Formal Aspects of Security* and Trust, ser. Lecture Notes in Computer Science, 2011, vol. 6561, pp. 126–140.
- [36] C. Efthymiou and G. Kalogridis, "Smart Grid Privacy via Anonymization of Smart Metering Data," in *Proc. IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2010, pp. 238–243.
- [37] F. Borges, L. A. Martucci, and M. Muhlhauser, "Analysis of privacy-enhancing protocols based on anonymity networks," in *Proc. IEEE Third International Conference on Smart Grid Communications (SmartGridComm)*, Nov. 2012, pp. 378–383.
  [38] M. Stegelmann and D. Kesdogan, "GridPriv: A Smart Metering
- [38] M. Stegelmann and D. Kesdogan, "GridPriv: A Smart Metering Architecture Offering k-Anonymity," in *Proc. IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, Jun. 2012, pp. 419–426.
- [39] S. Berthold and S. Clauss, "Linkability estimation between subjects and message contents using formal concepts," in *Proc. ACM workshop* on Digital identity management, ser. DIM '07, 2007, pp. 36–45.
- [40] S. Goldwasser and S. Micali, "Probabilistic encryption," Journal of Computer and System Sciences, vol. 28, no. 2, pp. 270–299, 1984.

- [41] —, "Probabilistic encryption & how to play mental poker keeping secret all partial information," in *Proc. ACM symposium on Theory of computing (STOC '82)*, May 1982, pp. 365–377.
- [42] M. Bellare and P. Rogaway, "Optimal asymmetric encryption," in Advances in Cryptology EUROCRYPT'94, ser. Lecture Notes in Computer Science, 1995, vol. 950, pp. 92–111.
- [43] M. Bellare, A. Boldyreva, A. Desai, and D. Pointcheval, "Key-Privacy in Public-Key Encryption," in *Advances in Cryptology (ASIACRYPT)*, ser. Lecture Notes in Computer Science, C. Boyd, Ed. Springer Berlin Heidelberg, 2001, vol. 2248, pp. 566–582.
- [44] M. Kohlweiss, U. Maurer, C. Onete, B. Tackmann, and D. Venturi, "Anonymity-Preserving Public-Key Encryption: A Constructive Approach," in *Privacy Enhancing Technologies*, ser. Lecture Notes in Computer Science, E. Cristofaro and M. Wright, Eds. Springer Berlin Heidelberg, 2013, vol. 7981, pp. 19–39.
- [45] G. Lowe, "Casper: a compiler for the analysis of security protocols," in *Proc. Computer Security Foundations Workshop*. IEEE Comput. Soc. Press, 1997, pp. 18–30.
- [46] A. Roscoe, "Modelling and verifying key-exchange protocols using CSP and FDR," in *Proc. IEEE Computer Security Foundations Workshop.* IEEE Comput. Soc. Press, 1995, pp. 98–107.
- [47] G. Lowe, "Breaking and fixing the Needham-Schroeder Public-Key Protocol using FDR," in *Tools and Algorithms for the Construction* and Analysis of Systems, ser. Lecture Notes in Computer Science, T. Margaria and B. Steffen, Eds. Springer Berlin Heidelberg, 1996, vol. 1055, pp. 147–166.
- [48] C. A. R. Hoare, *Communicating Sequential Processes*. Prentice Hall, 1985.
- [49] A. Roscoe, "Model-checking CSP," in A classical mind: essays in honour of CAR Hoare, 1994.
- [50] C. A. R. Hoare and J. Davies, *Communicating Sequential Processes*, 2004.
- [51] G. Lowe, "Casper : A Compiler for the Analysis of Security Protocols," Oxford University Computing Laboratory, Tech. Rep., 2009. [Online]. Available: http://www.cs.ox.ac.uk/gavin.lowe/Security/Casper/
- [52] D. Chadwick, "Design of Identity Management, Authentication and Authorization Infrastructure," Tech. Rep., 2010. [Online]. Available: http://homes.esat.kuleuven.ac.be/~decockd/tas3/final.deliverables/pm36/ TAS3\_D07p1\_IDM-Authn-Authz\_V3p0.pdf
- [53] B. Gedik and L. Liu, "Protecting Location Privacy with Personalized k-Anonymity: Architecture and Algorithms," *IEEE Transactions on Mobile Computing*, vol. 7, no. 1, pp. 1–18, 2008.
- [54] M. Jawurek, M. Johns, and K. Rieck, "Smart metering de-pseudonymization," in *Proc. Computer Security Applications Conference*, ser. ACSAC '11, 2011, pp. 227–236.
- [55] S. Finster and I. Baumgart, "Pseudonymous Smart Metering without a Trusted Third Party," in 2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications. IEEE, Jul. 2013, pp. 1723–1728.

- [56] SecureState, "Termineter Framework: Open Source Smart Meter Hacking Tool," 2012. [Online]. Available: http://blog.securestate.com/ termineter-framework-open-source-smart-meter-hacking-tool/
- [57] OpenADR Alliance, "OpenADR 2.0b Profile Specification," Tech. Rep., 2013.
- [58] M. A. Piette, G. Ghatikar, S. Kiliccote, E. Koch, D. Hennage, P. Palensky, and C. McParland, "Open Automated Demand Response Communications Specification (Version 1.0)," California Energy Commission, PIER Program, Tech. Rep. April, 2009.
- [59] M. Karwe and J. Strüker, "Maintaining Privacy in Data Rich Demand Response Applications," in *Smart Grid Security*, ser. Lecture Notes in Computer Science, J. Cuellar, Ed. Springer Berlin Heidelberg, 2013, vol. 7823, pp. 85–95.
- [60] A. Paverd, "Security and Privacy in Smart Grid Demand Response Systems," in Second Open EIT ICT Labs Workshop on Smart Grid Security - SmartGridSec14, 2014.
- [61] M. Burrows, M. Abadi, and R. M. Needham, "A Logic of Authentication," *Proc. Royal Society of London -Mathematical and Physical Sciences*, vol. 426, no. 1871, pp. 233–271, 1989.
- [62] M. Abadi and A. D. Gordon, "A calculus for cryptographic protocols: the spi calculus," in *Proc. ACM conference on Computer and communications security*, ser. CCS '97, 1997, pp. 36–47.
- [63] D. X. Song, S. Berezin, and A. Perrig, "Athena: a novel approach to efficient automatic security protocol analysis," *Journal of Computer Security*, vol. 9, no. 1, pp. 47–74, 2001.
- [64] B. Blanchet, "Security Protocol Verification: Symbolic and Computational Models," in *Proceedings of the First International Conference on Principles of Security and Trust*, ser. POST'12. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 3–29.
- [65] S. Kremer and M. Ryan, "Analysis of an Electronic Voting Protocol in the Applied Pi Calculus," in *Programming Languages and Systems*, ser. Lecture Notes in Computer Science, M. Sagiv, Ed. Springer Berlin Heidelberg, 2005, vol. 3444, pp. 186–200.
- [66] E. Brickell, J. Camenisch, and L. Chen, "Direct anonymous attestation," in *Proc. ACM conference on Computer and communications security (CCS '04)*, Oct. 2004, p. 132.
- [67] C. Fournet and M. Abadi, "Hiding Names: Private Authentication in the Applied Pi Calculus," in *Proc. Mext-NSF-JSPS international conference on Software security: theories and systems*, ser. Lecture Notes in Computer Science, M. Okada, B. Pierce, A. Scedrov, H. Tokuda, and A. Yonezawa, Eds., 2003, vol. 2609, pp. 317–338.
- [68] M. Abadi and C. Fournet, "Mobile values, new names, and secure communication," in *Proc. ACM SIGPLAN-SIGACT symposium on Principles of programming languages (POPL '01)*, vol. 36, no. 3, Jan. 2001, pp. 104–115.
- [69] A. Luu, J. Sun, Y. Liu, J. Dong, X. Li, and T. Quan, "SeVe: automatic tool for verification of security protocols," *Frontiers of Computer Science*, vol. 6, no. 1, pp. 57–75, 2012.