Plausible Deniability for Anonymous Communication

Christiane Kuhn*
KASTEL, Karlsruhe Institute of Technology
Germany
christiane.kuhn@kit.edu

Maximilian Noppel*
KASTEL, Karlsruhe Institute of Technology
Germany
maximilian.noppel@kit.edu

Christian Wressnegger
KASTEL, Karlsruhe Institute of Technology
Germany
christian.wressnegger@kit.edu

Thorsten Strufe
KASTEL, Karlsruhe Institute of Technology & TU Dresden
Germany
thorsten.strufe@kit.edu

ABSTRACT

The rigorous analysis of anonymous communication protocols and formal privacy goals have proven to be difficult to get right. Formal privacy notions as in the current state of the art based on indistinguishability games simplify analysis. Achieving them, however can incur prohibitively high overhead in terms of latency. Definitions based on function views, albeit less investigated, might imply less overhead but aren't directly comparable to state of the art notions, due to differences in the model.

In this paper, we bridge the worlds of indistinguishability game and function view based notions by introducing a new game: the "Exists INDistinguishability" (E·IND), a weak notion that corresponds to what is informally sometimes termed *Plausible Deniability*. By intuition, for every action in a system achieving plausible deniability there exists an equally plausible, alternative that results in observations that an adversary cannot tell apart. We show how this definition connects the early formalizations of privacy based on function views [13] to recent game-based definitions [15]. This enables us to link, analyze, and compare existing efforts in the field.

CCS CONCEPTS

• **Security and privacy → Security protocols**; **Formal security models**;

KEYWORDS

anonymity, peer-to-peer, privacy notion, plausible deniability

ACM Reference Format:
Christiane Kuhn, Maximilian Noppel, Christian Wressnegger, and Thorsten Strufe. 2021. Plausible Deniability for Anonymous Communication. In *Proceedings of the 20th Workshop on Privacy in the Electronic Society (WPES '21), November 15, 2021, Virtual Event, Republic of Korea.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3463676.3485605

*Both authors contributed equally to this research.

WPES '21, November 15, 2021, Virtual Event, Republic of Korea

ACM ISBN 978-1-4503-8527-5/21/11…$15.00
https://doi.org/10.1145/3463676.3485605

1 INTRODUCTION

Currently deployed Anonymous Communication Networks (ACNs), like Tor [7] or Freenet [6], address a variety of anonymous communication concepts. *Plausibe deniability* in these cases, for instance, requires that, given the observation of an adversary, for any action of a user (e.g., downloading a file) there exists at least one alternative user that is equally plausible to have performed the action. Hence, users can *plausibly deny* being the involved in any action.

While precise and comparable protection goals are important for anonymous communication, informal definitions are common. Goals are, if at all, defined independently of each other, rarely compared, and their relationship hardly understood. Understanding the complete problem space and the effective protection protocols provide from attacks is thus challenging, especially for fundamental goals such as plausible deniability. One example is a recent attack against Freenet's Opennet mode, that has been demonstrated [16]: In this mode, a node establishes untrusted connections, and an adversary that merely controls a single node can determine when a directly-connected victim is downloading a certain file. It hence shows that Freenet does not achieve plausible deniability in this setting.

Rigorous analysis necessitates a formal definition of the chosen security objectives, and it is an inevitable requirement for the corresponding security proof. The protection goals in most implemented systems, however, are defined ad-hoc or on an informal basis. In a recent effort to improve this situation by providing unambiguous definitions and comparability, Kuhn et al. [15] have developed an extensive hierarchy of formalized protection goals, called *privacy notions*. They are based on indistinguishability games, a fundamental definition structure that can be instantiated with different communication properties, defining, which information is to be protected by the protocol under scrutiny. Thereby they focus on a protection in the worst case scenario and ensure that the adversary cannot distinguish even two of the selected users. Their extensive hierarchy of notions includes all known indistinguishability game-based approaches [1, 10, 12].

Other formal tools were proposed to formalize privacy as well [2, 11, 13, 20–22]. For this work, we are interested in the approach by Hughes and Shmatikov [13] based on function views. They express privacy as the uncertainty of the adversary about a hidden function, e.g. the function that maps communications to senders. Close to our intuition for plausible deniability, they require the adversary to remain uncertain, and for each action there has to be at least one

other option in which it has not been performed by the suspected user. They define opaqueness to this end, which implicitly represents a best case approach to privacy — an interesting counterpart to the known worst case indistinguishability notions.

While indistinguishability games are a predominant way to express privacy due to their well understood relation to cryptography and the corresponding proofs, to understand the relations to and make use of results for the function view definitions, we need to find a bridge between these worlds.

In this work, we investigate how definitions based on function views can be expressed by means of indistinguishability games to make them comparable. We then go on to analyze the Peer-to-Peer (P2P) scenario, in which the anonymization service is realized by cooperation of all participants, without relying on any external services. Deriving a new, corresponding performance bound for the existing worst-case notions, we demonstrate the necessity of a plausible deniability notion in this scenario, and define its game.

We specify the new E·IND game as an indistinguishability based notion that can express several variants of plausible deniability. In doing so, we are creating a framework to easily express deniability for different actions, built on the conventional formal foundations from cryptography (e.g., IND-CPA) and digital signatures (e.g., EUF-CMA). We discuss how our new game-based definition connects to the function view based definitions of Hughes and Shmatikov [13]. Finally we exploit our consolidation of models to formally compare our new definitions to the existing hierarchy of worst-case privacy notions [15]. We show that E·IND expresses weaker guarantees and thus allows to rigorously prove the weak protection of plausible deniability in anonymous communication.
In summary, our main contributions are:

- **P2P network bound on known indistinguishability notions.** We analyze the limits of existing privacy notions and prove a highly restrictive performance bound for P2P networks under these notions. Thus we are demonstrating the importance of weaker notions of privacy.
- **Formal definition for plausible deniability.** We introduce a general game-based formalization of plausible deniability in Anonymous Communication Networks (ACNs). The versatility of indistinguishability games allows us to generalize to arbitrary communication properties.
- **Relation to existing notions.** We link and relate our new notion to other indistinguishability games and privacy notions [15] as well as *function views* [13], enabling a comprehensive comparison.

The rest of the paper is organized as follows: In Section 2, we provide background information on privacy notions in Anonymous Communication Networks, before we show the necessity of weaker notions with our performance bound in Section 3 and then introduce our new game-based notion, E·IND, for plausible deniability in Section 4. In Section 5, we show the equivalence of our game-based notion and privacy notions using function views. Section 6 additionally details the relationship between "Exists INDistinguishability" (E·IND) and the recently proposed notions by Kuhn et al. [15], that we refer to as "All INDistinguishability" (A·IND). We discuss our findings in Section 7 and conclude the paper in Section 8.

2 PRIVACY NOTIONS FOR ACNs

Anonymous Communication Networks (ACNs) do not only provide confidentiality of the communicated message, but conceal different types of metadata of the communications. Their specific protection goals, however, differ widely, as well as their assumptions made regarding considered adversaries and their usage scenarios. This has led to various different implementations, such as Tor [7], Mix-Nets [5], Crowds [19], Freenet [6], or DC-Nets [4] to protect the user's privacy in online communications. All of them protect the communication metadata to some extent. Some, however, only aim to unlink the sender from the message and receiver [7], while others hide even more user behavior, like the sending frequencies [4]. Next to the protection goals, also the assumed adversary's capabilities of various ACNs differ from each other. Some attackers observe a small portion of the network only [7], others can act globally [4]. Moreover, the protocols are built upon different network models: Some ACNs protect users as a service and allow to reach receivers that are unaware of the ACN (e.g., web servers on the Internet through TOR) [7], while others protect the communication between users of the ACN as an integrated system [6]. In this work, we focus on ACNs that exchange unicast and unidirectional messages between their users, but do not restrict the network model otherwise.

2.1 Informal Goal Definitions

Informal descriptions of privacy goals in these networks as provided by Pfitzmann and Köhntopp [18] unfortunately are prone to misinterpretations [15]. Formal privacy notions are thus essential to unambiguously express privacy goals and protection guarantees. While different approaches have been proposed in the past [2, 9, 11, 15, 20–22], formalizing security as indistinguishability games has become the state of the art [1, 3, 15]. Subsequently in Section 2.2, we elaborate the currently most extensive framework for such privacy definitions by Kuhn et al. [15]. We will refer to it as "All INDistinguishability" (A·IND) in the remainder of the paper, as it implies that all eligible scenarios are indistinguishable and the adversary can freely choose any two within the game. Additionally, in Section 2.3, we detail the underlying concept of using function views to define anonymity as proposed by Hughes and Shmatikov [13]. Our new notion, defined in Section 4, allows to compare these fundamentally different approaches.

2.2 The "All INDistinguishability" Notions

The A·IND notions assume a setting of multiple unicast and unidirectional *communications*, each transferring a *message* from the *sender* to the *receiver*. We further denote the sequence of all communications, which have happened and are happening over the ACN, as a *scenario*.

We focus on sender unobservability, one of the easiest to grasp notions defined in this model. It requires that it is impossible for an adversary to identify a certain agent to be the sender [18] of any message. It is formalized by challenging the adversary to make this differentiation in a game: Over multiple repetitions, the challenger randomly decides whether the real sender or somebody else was sending and asks the adversary to determine who was sending based on her observations. If the adversary has a strategy that allows for

a non-negligible advantage in correctly identifying the sender over mere guessing, the ACN does not provide sender unobservability.

In a similar manner, Kuhn et al. [15] further define a comprehensive set of other protection goals based on the protected communication properties, such as who sent what or who has been sending how often. Their hierarchy of these protection goals provides a useful groundwork for our comparison.

After this first intuitive description of the A·IND concept, we provide a formalization in Section 2.2.1, as the foundation of our plausible deniability definition of ACNs. Later on in Section 2.2.2, we then detail the properties considered in the rest of the paper.

2.2.1 Formalization. The game compares sequences of communications that we refer to as scenarios $s \in S$. Every communication is a tuple (a, b, m, aux) of the sending and receiving agents a, b (of the agents set A), the exchanged message $m \in M$ and some auxiliary information $aux \in AUX$. The sets of agents A, messages M, and auxiliary information AUX are finite.[1] The ACN itself is modeled by a Probabilistic Polynomial Time (PPT) algorithm $\Pi : S \rightarrow T$ that maps scenarios S to transcripts T. Transcripts include the observations that the adversary makes in the real network. They can e.g. be lists of the observed packets and timings, but their precise format is only of importance for the ACN analysis, not for the definition of notions. The adversary model hence is inherent in the transcripts and thus in the ACN algorithm Π.

The A·IND game inspects whether an adversary can distinguish one pair of self-chosen scenarios, as visualized in Fig. 1. If the adversary cannot even distinguish one self-chosen pair, i.e. the worst case for the analyzed protocol, *all* eligible scenarios are *indistinguishable*.

In the first step, the adversary $\mathcal{A}$ chooses two scenarios $s_0, s_1 \in S$ such that the analyzed property p holds: $p(s_0, s_1)$. Both scenarios are submitted to the challenger C, who verifies the property p and chooses a challenge bit b. Subsequently, she executes scenario s_b in the ACN Π and returns the corresponding *transcript*, which contains the adversary's observations of the execution. Finally, the adversary submits her guess of the challenge bit b' and wins the game if $b = b'$. Note that a random guess yields a winning probability of $\frac{1}{2}$. We hence denote Π as p–A·IND private iff the adversary's winning probability is at most negligibly higher than $\frac{1}{2}$.

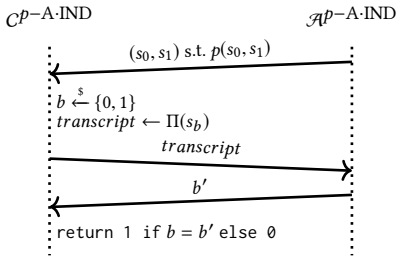

Figure 1: The experiment $\mathbf{Exp}^{p\text{–A·IND}}_{\Pi, \mathcal{A}}$: Vertical lines represent the life lines of all actors , arrows the passing of messages between them.

[1] We assume a limited packet size on the communication medium.

Definition 2.1 (p–A·IND Anonymity). Given a property p, an ACN Π, and the PPT adversary $\mathcal{A}$, we define an experiment $\mathbf{Exp}^{p\text{–A·IND}}_{\Pi, \mathcal{A}}$ as Algorithm 1.

Algorithm 1 The $\mathbf{Exp}^{p\text{–A·IND}}_{\Pi, \mathcal{A}}$ experiment

```
(s_0, s_1, state_A) ← A(1^κ, getPair)
abort if ¬p(s_0, s_1)
b ←$ {0, 1}
transcript ← Π(s_b)
b' ← A(1^κ, attack, transcript, state_A)
return 1 if b = b' else 0
```

The advantage of adversary $\mathcal{A}$ is given by

$$\mathbf{Adv}^{p\text{–A·IND}}_{\Pi, \mathcal{A}}(1^\kappa) := \left| Pr\left[\mathbf{Exp}^{p\text{–A·IND}}_{\Pi, \mathcal{A}} \rightarrow 1\right] - \frac{1}{2}\right| \cdot 2.$$

The adversarial algorithm $\mathcal{A}$ is denoted *valid* for p–A·IND iff it runs in PPT and guarantees that $p(s_0, s_1)$ holds. The challenger C is denoted *valid* iff she guarantees that the transcript is either $\Pi(s_0)$ or $\Pi(s_1)$ with equal probability. The ACN Π is p–A·IND private iff for every *valid* PPT-adversary $\mathcal{A}$ the function $\mathbf{Adv}^{p\text{–A·IND}}_{\Pi, \mathcal{A}}(1^\kappa)$ is negligible in the security parameter κ.

Remark 1. *Note that we can compare notions based on properties: If the property is more restrictive (accepts fewer pairs of scenarios), it results in a weaker (easier to achieve/harder to break) notion as the adversary has less freedom when attacking it. Thus, for any two properties p^a and p^b, the following statement holds: If $\forall s_0, s_1 \in S : p^a(s_0, s_1) \Rightarrow p^b(s_0, s_1)$ then p^b–A·IND $\Rightarrow p^a$–A·IND .*

2.2.2 Notions and Properties. From the choice of all notions we focus on one specific sender notion to investigate plausible deniability: *Sender Unobservability*, $\overline{SO}$–A·IND. This notion allows to input any scenarios that only differ in the senders. The receivers, messages and auxiliary information are identical in both scenarios and thus do not help to distinguish the cases. Note that this requires that nothing about the senders can be learned by the adversarial observation (not even which senders are sending messages). However, the number of real communications in both scenarios and information about receivers and messages can be learned. To express this formally, the property *Equal but senders* E_S is defined to accept any scenarios that have the same receivers, messages, and auxiliary information in each communication. In this paper, we use *Sender Unobservability* $\overline{SO}$–A·IND synonymously with E_S–A·IND.

2.3 Function Views

Hughes and Shmatikov [13] represent scenarios with functions. For instance, the sender function $f : C \rightarrow A$ specifies which agent $a \in A$ is the sender of communication $c \in C$. As the sending agent for a communication depends on the scenario, we write f_s for the f function of the scenario $s \in S$.

2.3.1 Function Knowledge. Hughes and Shmatikov formally model privacy in terms of the observer's uncertainty about a hidden function. They consider three different kinds of knowledge about a function.

Graph Knowledge F. The function graph represents the linking of in- and outputs, modeled as tuples $F \subseteq C \times A$. Knowing it thus entails the complete knowledge of the f function. With partial knowledge e.g., multiple entries for the same communication ((c, a) and (c, a')) uncertainties of the adversary (a or a' send in c) are expressed.

Image Knowledge I. The image is the subset of the output values assumed by the function e.g., $\mathrm{Im}\, f \subseteq A$. If the adversary knows the complete image she learns all agents that did the action e.g., for f all active senders. With partial knowledge $I \subset \mathrm{Im}\, f$ the observer's uncertainty is expressed.

Kernel Knowledge K. The kernel describes which inputs induce the same output, modeled as tuples $K \subseteq C \times C$. So if (c_0, c_1) is in the kernel knowledge K: $f(c_0) = f(c_1)$. For f knowing a tuple of the kernel entails knowing that two communications are sent by the same agent, but not knowing by whom.

From Knowledge to Function Views. The tuple $N = \langle F, I, K \rangle$ combining the knowledge is the *function knowledge*. However, the different types of knowledge are not independent. There are thus function knowledge tuples that do not state what the adversary can actually infer. Knowing e.g., $K = C \times C$, i.e. all communications have the same sender, and $I = \{a\}$, i.e. a is sending, allows to conclude the graph as (c, a) for all c, no matter what the graph knowledge actually states. *Function views* are the knowledge tuples that state the maximal adversary's knowledge directly, i.e. no other than the stated knowledge about F, I or K can be derived from the other two kinds of knowledge.

2.3.2 Equivalence and Opaqueness. The *observational equivalence* relation $\sim: S \times S \to \{0, 1\}$ expresses which scenarios can be distinguished based on the adversary's observations. If two scenarios s_0 and s_1 are observational equivalent, then the adversary cannot be sure if the function is f_{s_0} or f_{s_1}. The observational equivalence induces equivalence classes, i.e. groups of scenarios that cannot be distinguished from each other.

Opaqueness. For each knowledge type at least one opaqueness definition is introduced. These opaqueness definitions require to hide their corresponding type of knowledge.

Example: Image Opaqueness. A function view, i.e. the observations for one equivalence class, is *image opaque* if the image knowledge is empty: $I = \emptyset$. Intuitively speaking, the adversary does not know a single agent that is guaranteed to be active. A complete ACN Π is image opaque on the f function if the function view for every equivalence class is image opaque. In other words, in every equivalence class and for every agent, there must exist a scenario in the same class, in which the agent is not active:

$$\forall s_0 \in S \,.\, \forall a \in A \,.\, \exists s_1 \;\sim\; s_0 \,.\, a \notin \mathrm{Im}\, f_{s_1}$$

Note that "$\forall s_0 \in S$" is equivalent to "$\forall$ equivalence classes", as every scenario is part of exactly one equivalence class.

3 LIMITS FOR THE A·IND GAME

Achieving any privacy notion against an adversary naturally requires a certain minimum overhead in practice. In the following, we show that achieving even the *Sender Message Unlinkability* $(SM)\overline{L}$–A·IND notion as defined by Kuhn et al. [15], requires an impracticable high overhead in a P2P network. Sender message unlinkability $(SM)\overline{L}$–A·IND is weaker than sender unobservability and requires that every agent is sending the same number of messages in both scenarios, but which message is sent by whom can differ in both scenarios.

We demonstrate a lower bound on the overhead necessary to achieve $(SM)\overline{L}$–A·IND against an adversary that passively controls a) one forwarding node in a P2P setting and b) can tell which message she forwards (e.g., by recognizing a message identifier)[2]. We further assume that the routes of different communications are chosen independently from each other.

For our estimate, we use the *Predecessor Attack* [16, 23]: considering an adversarial node to forward a message from a neighbor, it can easily deduce that this message has more likely been sent by said neighbor, than a random, different user. In an indistinguishability game, an adversary thus decides on her neighbor as the first potential sender and another random user as the second. If she gets to forward exactly one of the two relevant messages from her neighbor, she may guess that this is the first forwarding of the message and can blame her neighbor as the sender of this message. In all other cases, the adversary guesses randomly.

Given that the routes of messages are chosen independently, an adversary following this strategy has a non-negligible advantage of winning. To estimate the advantage, we calculate her probabilities to observe the relevant message either when it has been sent by her neighbor, or when it is sent by the alternative candidate.

We thereby derive that to achieve $(SM)\overline{L}$–A·IND without allowing any adversarial advantage at all, we need to relay our message (or a copy of it) as often as the number of users of the P2P network and "not much less often" for our requirement of a negligible advantage. This of course are very high costs. We hence are convinced of the demand for a weaker, yet still meaningful privacy definition, and we will define such a notion in Section 4.

Theorem 3.1. *For $(SM)\overline{L}$–A·IND privacy with*

$$\boldsymbol{Adv}^{(SM)\overline{L}\text{–}A\cdot IND}_{\Pi,\mathcal{A}} = 0$$

a passively observing adversary that can recognize the message, and any efficient protocol [14] where the routes of different communications are chosen independently the following equation must hold

$$n \leq h \,,$$

where n refers to the number of user nodes and h denotes the maximum number of hops the message or a copy of it is forwarded (excluding the intended receiver).

[2] This requirement is e.g. given in networks that handle published content based on content identifiers, like Freenet [6].

For $(SM)\overline{L}$–A·IND privacy with negligible $\boldsymbol{Adv}^{(SM)\overline{L}-A\cdot IND}_{\Pi,\mathcal{A}}$ (according to Definition 2.1) in the otherwise identical setting the following equation must hold

$$n - \frac{n \cdot \eta}{poly(\kappa)} \leq h \,,$$

where h is defined as above, κ denotes the security parameter, poly is a polynomial function, and η refers to a user's maximum number of neighbors.

Proof. We start with describing the *attack*:

(1) The adversary decides on two challenge messages m, m', receiver B and two senders A, C with A being a randomly chosen neighbor of her own node and C being a random other agent. She builds the scenarios such that A sends m to B in the first one and m' to B in the second. The other message is sent by C to B in each scenario.
(2) The adversary waits the time that forwarding a message for the maximum number of hops h allowed for the original message can take (e.g. h rounds if a message is forwarded once a round).
(3) If the adversary only forwarded the message m from A, she guesses 0. If she only forwarded m' from A, she guesses 1. Otherwise she guesses randomly.

Advantage. From the advantage definition:

$$\mathbf{Adv} = \left| Pr(\mathbf{Exp} \rightarrow 1) - \frac{1}{2} \right| \cdot 2$$

which is equivalent to:

$$\mathbf{Adv} = Pr(\mathcal{A}\ guesses\ 0 \mid b = 0) - Pr(\mathcal{A}\ guesses\ 0 \mid b = 1)$$

Equivalence:

$$\begin{aligned}
\mathbf{Adv} &= \left(Pr(\mathbf{Exp} \rightarrow 1) - \frac{1}{2} \right) \cdot 2 \\
&= (0.5Pr(\mathcal{A}\ guesses\ 0 \mid b = 0) + 0.5Pr(\mathcal{A}\ guesses\ 1 \mid b = 1) - 0.5)2 \\
&= Pr(\mathcal{A}\ guesses\ 0 \mid b = 0) + Pr(\mathcal{A}\ guesses\ 1 \mid b = 1) - 1 \\
&= Pr(\mathcal{A}\ guesses\ 0 \mid b = 0) + (1 - Pr(\mathcal{A}\ guesses\ 0 \mid b = 1)) - 1 \\
&= Pr(\mathcal{A}\ guesses\ 0 \mid b = 0) - Pr(\mathcal{A}\ guesses\ 0 \mid b = 1)
\end{aligned}$$

Note that in the case that both messages or none are sent over the observed link, the probability of guessing each way is $\frac{1}{2}$ and thus this cancels out in the above equation. Hence, the following probabilities remain:

$$\begin{aligned}
\mathbf{Adv} = &Pr(only\ m\ is\ sent\ over\ the\ link \mid A\ is\ the\ sender\ of\ m) \\
&- Pr(only\ m\ is\ sent\ over\ the\ link \mid C\ is\ the\ sender\ of\ m)
\end{aligned}$$

Intuitively, the probability for the message to be transmitted over the observed link is much higher in the case that A, the adversary's neighbor, is the sender, than if C, a remote node has sent it. Thus, we want the probability that A sends to the malicious neighbor as small as possible, and the probability that a message sent by C is forwarded over the adversarial node and A to be as large as possible (or at least similarly high as the former). As A has to send over some neighbor and cannot know which neighbor is malicious, for any efficient protocol $Pr(m\ is\ sent\ over\ the\ link \mid A\ is\ sender\ of\ m) \geq \frac{1}{|N(A)|}$, where $N(A)$ is the set of neighbors of A. Also, no efficient protocol can forward C's message over more than h links and as the protocol cannot know which link of the total $n \cdot \eta$ directed links is observed, $Pr(m\ is\ sent\ over\ the\ link \mid C\ is\ sender\ of\ m) \leq \frac{h}{n \cdot \eta}$ holds for any efficient protocol. Note, that both probabilities do not depend on which message is sent. Using the assumptions that the routes are chosen independent of each other, we can now calculate **Adv**:

$$\begin{aligned}
\mathbf{Adv} = &Pr(m\ is\ sent\ over\ the\ link \mid A\ is\ sender\ of\ m) \cdot \\
&Pr(m'\ is\ not\ sent\ over\ the\ link \mid C\ is\ sender\ of\ m') \\
&- Pr(m\ is\ sent\ over\ the\ link \mid C\ is\ sender\ of\ m) \cdot \\
&Pr(m'\ is\ not\ sent\ over\ the\ link \mid A\ is\ sender\ of\ m') \\
\mathbf{Adv} \geq\ &\frac{1}{|N(A)|} \cdot \left(1 - \frac{h}{n \cdot \eta}\right) - \frac{h}{n \cdot \eta} \cdot \left(1 - \frac{1}{|N(A)|}\right) \\
\mathbf{Adv} \geq\ &\frac{1}{|N(A)|} - \frac{h}{n \cdot \eta} \\
\mathbf{Adv} \geq\ &\frac{1}{\eta} - \frac{h}{n \cdot \eta}
\end{aligned}$$

Bound. Thus, for $\mathbf{Adv} = 0$: $\frac{h}{n \cdot \eta} \geq \frac{1}{\eta} \iff h \geq n$
and for negl. **Adv**, for any positive polynomial *poly*:
$\frac{1}{poly(\kappa)} > \mathbf{Adv} \geq \frac{1}{\eta} - \frac{h}{n \cdot \eta} \iff h > n - \frac{n \cdot \eta}{poly(\kappa)}$

□

Remark 2. *As $S\overline{O}$–A·IND is strictly stronger than $(SM)\overline{L}$–A·IND, this bound is also a lower bound for $S\overline{O}$–A·IND.*

4 THE "Exists INDistinguishability" GAME

Recall, that "All INDistinguishability" requires that any two comparable scenarios (i.e., scenarios that fulfill the properties in question) are indistinguishable to an adversary. Plausible deniability, however, requires that at least *one* other option is an equally *plausible* explanation for the observations. For a user sending a specific message, it hence must be equally plausible that she did not send it, that is, either another user or nobody might have sent the respective message.

We hence introduce the "Exists INDistinguishability" (E·IND) game to express that at least one other, *indistinguishable* scenario *exists* that fulfills a specific property. There does not necessarily have to be *only one* indistinguishable alternative, but the existence of at least one is sufficient for this goal.

Game Formulation. Formulating a game for plausible deniability requires a few adaptations to the existing notions: We require *one other* option for deniable actions in the real world. In the game, we thus allow the adversary to decide the action of the first scenario s_0 as a representation of the real world. However, a *single* indistinguishable scenario fulfilling the properties is sufficient. The adversary can thus no longer pick the second scenario s_1. If we would allow her to choose, she would decide on the easiest distinguishable scenario. Instead, we present the adversary with the scenario that is most difficult to distinguish, i.e. the closest match to her chosen scenario, that fulfills the properties.

Based on this decision, we can then use the same structure as described in Section 2.2 to decide whether the chosen scenarios can be distinguished by the adversary, or not.

Outline. In Section 4.1, we begin with a discussion on how to find an indistinguishable scenario for our game-based formulation, before we elaborate on why identical scenarios need to be prohibited in Section 4.2. Next, we introduce the advanced set of parameterized properties in Section 4.3. The fundamental change from A·IND to E·IND further requires us to adapt two details in the A·IND game that make no difference for the A·IND game, but are important for E·IND: the protocol state and randomness in Sections 4.4 and 4.5. Finally, we provide the formal definition of the E·IND notions in Section 4.6.

4.1 Finding the Alternative

The most difficult scenario for the game can only be identified by a trusted challenger and comprises two sub-problems: 1) What should happen if there is *no* other scenario fulfilling the properties in question and 2) what if there are *multiple* such other scenarios?

If for the analyzed property p and the chosen scenario s_0, there is *no other matching scenario* s_1 that fulfills the properties, there is no alternative to plausibly deny s_0 being the *real* scenario. In such a case the adversary wins the game immediately, as the requirement is not satisfiable. Formally, we define satisfiability as follows:

Definition 4.1 (p-Satisfiability). For a property p and a scenario $s_0 \in S$ we say "s_0 is *p-satisfiable*", if there exists a $s_1 \in S$ with $p(s_0, s_1)$.

If for the analyzed property p and chosen scenario s_0, there are *multiple matching scenarios* s_1 that fulfill the properties, the *closest* match has to be considered in the game. The *closest* match is the scenario that is the hardest to distinguish, hence, where the adversarial observations are as identical as possible. While some protocols pick the alternative explicitly during operation (e.g., QuisQuis [8] or Monero [17]) for other protocols, e.g., P2P networks, choosing a neighbor of the victim is an obvious heuristic. However, even if such an indistinguishable scenario s_1 exist, the challenger may not be able to find it in polynomial time. We thus require the adversary to have a non-negligible advantage against *every other challenger* that decides on a scenario s_1 fulfilling the properties of scenario s_0. Iterating all challengers includes the one that picks the closest match. In contrast to A·IND, where we consider the maximal advantage, in the E·IND game, we are instead interested in the adversary's minimal advantage for all scenarios fulfilling the properties, as we only require that *one* alternative is indistinguishable.

4.2 Prohibiting Identical Scenarios

Most properties from the A·IND notions, such as $S\overline{O}$–A·IND, are reflexive and allow to pick two identical scenarios: $p(s_0, s_0)$. In the A·IND game the adversary decides on the easiest to distinguish scenarios. She hence never picks these and allowing this choice has no ramifications. For E·IND, however, this renders the notions useless, as the adversary cannot win against the challenger that picks $s_0 = s_1$ and, thus, any such notion is trivially achieved, while really no deniability is attained. We thus adapt $S\overline{O}$ as follows and introduce a new property:

Forced Change Property. In addition to the *Equal but senders* property, E_S, that restricts to only change the senders, we require the inequality property $\neq$ to hold, which states that s_0 and s_1 cannot be identical. This property still allows to attack the empty scenario $s_0 = \varphi$, that is, a scenario in which no communication happens at all,[3] which is unsatisfiable under $(E_S, \neq)$–E·IND. Hence, we further require at least one communication in s_0 by defining the new property $len^{>0}$ and define sender unobservability for the E·IND game as $S\overline{O}$–E·IND = $(E_S, \neq, len^{>0})$–E·IND.

Remark 3. *Note that neither $\neq$, nor $len^{>0}$ constrain $S\overline{O}$–A·IND. The adversary has no advantage by attacking $s_0 = s_1$ and $s_0 = \varphi$ induces $s_1 = \varphi$ as the receivers and messages have to be equal in both scenarios.*

4.3 Parameterized Properties

We need to ensure that a single user is sending something in one, but not the other scenario, that is, the user can deny the action of sending *anything at all*. For this, the adversary has to additionally choose a sender in the first scenario, such that the challenger knows which user must not send a message in the second. Thus, the second scenario does no longer only depend on the first, but also on the chosen sender. To handle such additional parameters, we define a new type of property:

Definition 4.2 (Parameterized Property). A *parameterized property* $p_\tau : S \times S \to \{0, 1\}$ is a $\mathcal{T}$-indexed family of relations $(p_\tau)_{\tau \in \mathcal{T}}$ with the *parameter tuple* τ from the *set of possible parameters* $\mathcal{T}$.

In the following we simply write p for the whole family $(p_\tau)_{\tau \in \mathcal{T}}$. With this, we can then define a property that allows the user to plausible deny to have sent anything at all:

Definition 4.3 (Property Inactive Sender). Given two scenarios $s_0, s_1 \in S$ and an agent $a \in A$ the following holds:

$$\overline{\text{Sender1}}_a(s_0, s_1) \iff a \text{ does not send anything in } s_1$$

The set of possible parameters is the set of agents: $\mathcal{T} := A$.

4.4 Protocol State

In A·IND the protocol state is only indirectly defined, as the protocol model starts in an initial state and protocol queries are used to modify it [15]. We however make the protocol state explicit and include it directly in the protocol model to $\Pi : \Theta \times S \to T$, where Θ is the set of *protocol states*. This protocol state is generated at random according to some distribution $\theta \xleftarrow{\mathcal{D}} \Theta$ that corresponds to the situation of the protocol which is currently analyzed.

Moreover, we define who has access to the state. In A·IND the adversary gets partial knowledge of the state via the observations she learns from the returned transcripts and the challenger does not make any decisions dependent on the state. In E·IND the challenger however benefits from knowing the state when deciding on the second scenario. Therefore, we introduce a *perspective* on the state for both, the challenger and the adversary. The adversarial knowledge of the state, e.g., the neighbors of the adversarial nodes in a P2P network, are contained in her perspective: $persp_{\mathcal{A}} : \Theta \to \mathcal{P}(\Theta)$, where $\mathcal{P}(\Theta)$ is the product set of protocol states from the random oracle. In the game, the adversary is allowed to make use of her perspective to select the attacked scenarios. For the challenger we assume that she has read access to the complete protocol state.

[3] Note, the difference to an empty communication as defined in Kuhn et al. [15]. An empty communication just expresses that *at this point* no communication happens.

Note, that the protocol state is explicitly not generated or even manipulated by the challenger.

Remark 4. *While an explicit protocol state in A·IND notions does not change their expressiveness, it does provide a handy shortcut.*

4.5 Randomness

For achieving anonymity nearly all protocols rely on some randomness (e.g., the shared secrets between participants in DC-Nets [4] or the randomly chosen delays and keys in Mix-Nets [5]). In A·IND the corresponding randomness is implicit in the protocol model and as the challenger has no decision that depends on it, this is sufficient. For our E·IND game however, knowing the randomness gives the challenger a benefit in picking the second scenario. We hence differentiate two options: Either the challenger knows the randomness or she does not.

In the first case, a sufficient number of random bits may already be included in the protocol state and can be used to execute the scenario's communications. As discussed previously in Section 4.4, the challenger has full read access to the protocol state. She thus is able to predict every future random decision before deciding for the alternative scenario s_1. In very rare cases, she is even capable of picking the single correct scenario that matches this rare situation. For the second case, in which the challenger does not know the introduced randomness, the random bits are generated *after* the challenger decides for an alternative scenario. Here, our notion results in a single alternative scenario for any future random decision. The probability that this scenario is indeed indistinguishable is part of the adversary's advantage.

As an example, consider a protocol that guarantees to randomly pick one of 1000 users as the alternative sender—e.g., only one random user is sending dummy traffic. In the first case, the challenger is able to choose the correct sender for the alternative scenario as she has read access to future random decisions. In the second case her chances are just 0.1 % to choose an indistinguishable sender.

Formalization. In the first case the challenger is stronger and thus the notions offer a weaker protection. We thus prefer and use the second case in the following. We therefore adapt the model of Π again and add a random input: $\Pi : \Theta \times S \times \{0,1\}^* \to T$.

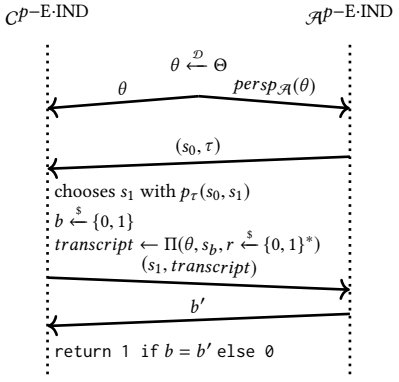

Figure 2: The $\mathbf{Exp}^{p\text{–E·IND}}_{\Pi,\mathcal{A},C}$ experiment.

4.6 The E·IND Experiment

We can now describe our new game between challenger C and an adversary $\mathcal{A}$ formally. Fig. 2 provides a depiction of the E·IND experiment that operates based on the (parameterized) property p:

At first, the adversary learns about the protocol state θ according to her capabilities[4] and then she submits $s_0 \in S$ as well as the parameter tuple τ. If s_0 is not p_τ*-satisfiable*, the adversary wins the game. Otherwise, the challenger chooses s_1 such that $p_\tau(s_0, s_1)$ and generates a random bit $b \xleftarrow{\$} \{0,1\}$. Note that the challenger has no information about the randomness r in the protocol execution and cannot predict future (random) decisions. Scenario s_b is executed in the ACN Π and the challenger returns both the $transcript \leftarrow \Pi(s_b)$ and scenario s_1 to the adversary. Based on the $transcript$, the adversary makes her guess on the executed scenario b' and wins if her guess is correct. With this experiment, we can define achieving p–E·IND similar to A·IND before:

Definition 4.4 (p-E·IND Anonymity). For an ACN Π, an PPT-adversary $\mathcal{A}$, a challenger C, and a property p with the possible parameter set $\mathcal{T}$ and the parameter tuple $\tau \in \mathcal{T}$, we define the $\mathbf{Exp}^{p\text{–E·IND}}_{\Pi,\mathcal{A},C}$ experiment in Algorithm 2. An adversarial algorithm $\mathcal{A}$ is *valid* iff it is PPT. Algorithm C, in turn, is *valid* for p–E·IND iff it guarantees that $C(1^\kappa, s_0, \tau)$ returns a scenario s_1 such that $p_\tau(s_0, s_1)$ holds. If no such scenario exists, there exists no valid challenger either.

Algorithm 2 The $\mathbf{Exp}^{p\text{–E·IND}}_{\Pi,\mathcal{A},C}$ experiment

1 $\theta \xleftarrow{\mathcal{D}} \Theta$
2 $(s_0, \tau, state_{\mathcal{A}}) \leftarrow \mathcal{A}(1^\kappa, \mathrm{getScenario}, persp_{\mathcal{A}}(\theta))$
3 $s_1 \leftarrow C(1^\kappa, s_0, \tau, \theta)$
4 $b \xleftarrow{\$} \{0,1\}$
5 $transcript \leftarrow \Pi(\theta, s_b, r \xleftarrow{\$} \{0,1\}^*)$
6 $b' \leftarrow \mathcal{A}(1^\kappa, \mathrm{attack}, s_1, transcript, state_{\mathcal{A}})$
7 **if** $b = b'$: **return** 1
8 **else**: **return** 0

The advantage of adversary $\mathcal{A}$ against a specific challenger C is defined as

$$\mathbf{Adv}^{p\text{–E·IND}}_{\Pi,\mathcal{A},C}(1^\kappa) := \left| Pr[\mathbf{Exp}^{p\text{–E·IND}}_{\Pi,\mathcal{A},C} \to 1] - \frac{1}{2} \right| \cdot 2 \,,$$

while the general advantage of her is defined as

$$\mathbf{Adv}^{p\text{–E·IND}}_{\Pi,\mathcal{A}}(1^\kappa) := \begin{cases} \min_{\forall\ valid\ C} \left(\mathbf{Adv}^{p\text{–E·IND}}_{\Pi,\mathcal{A},C}(1^\kappa) \right) & \text{if } \exists\ valid\ C \\ 1 & \text{else} \end{cases}$$

The ACN Π is p–E·IND iff $\mathbf{Adv}^{p\text{–E·IND}}_{\Pi,\mathcal{A}}(1^\kappa)$ is negligible in κ for all *valid* adversaries.

Remark 5. *Infinitely many* valid *challengers may exist. Disproving an E·IND privacy notion can therefore be harder than proving it.*

[4] This, for instance, includes her neighbor's IP addresses in a P2P network.

5 COMPARISON TO FUNCTION VIEWS

We compare our "Exists INDistinguishability" (E·IND) game to the notions established by Hughes and Shmatikov [13] that are based on *function views*. We manage to carve out the relations between both, showing that the E·IND notion generalizes upon function views. While our formulation allows for any arbitrary property, Hughes and Shmatikov propose a variety of concrete deniable actions to build their notions of *opaqueness*.

To show the equivalence of E·IND and the opaqueness definitions, proposed by Hughes and Shmatikov, we define our interpretation of observational equivalence and match the opaqueness and the E·IND definition at the example of image opaqueness.

5.1 Indistinguishability

We define observational equivalence of two scenarios as their indistinguishability in the A·IND game. We define the $\mathbf{Exp}^{s_0,s_1}_{\Pi,\mathcal{A}}$ experiment similar to $\mathbf{Exp}^{p\text{–A·IND}}_{\Pi,\mathcal{A}}$, except that the adversary is only allowed to attack the pair (s_0, s_1).[5]

Definition 5.1 (Indistinguishability). Let $s_0, s_1 \in S$ be two scenarios. s_0 and s_1 are *indistinguishable* iff no PPT algorithm $\mathcal{A}$ has a non-negligible advantage $\mathbf{Adv}^{s_0,s_1}_{\Pi,\mathcal{A}}$.

The binary relation $\sim\colon S \times S \rightarrow \{0,1\}$ is 1 if the scenarios are indistinguishable and 0 otherwise.

The only requirement on *observational equivalence* is that it is an equivalence relation. Therefore we show in Appendix A.1 that our interpretation of $\sim$ is indeed an equivalence relation:

Lemma 5.2. *The binary relation $\sim$ is an equivalence relation.*

Remark 6. *An ACN can be described completely by its $\sim$ equivalence classes. We therefore describe systems in upcoming proofs only by their equivalence classes.*

A·IND Indistinguishability. In p–A·IND the adversary can attack every pair of scenarios (s_0, s_1) with $p(s_0, s_1)$. Therefore every such pair must be observational equivalent to achieve p–A·IND:

$$p\text{–A·IND} \Leftrightarrow \forall s_0, s_1 \in S : \; p(s_0, s_1) \Rightarrow s_0 \;\sim\; s_1$$

5.2 Matching Opaqueness and E·IND Definition

Recall Image Opaqueness as

$$\forall s_0 \in S \,.\, \forall a \in A \,.\, \exists s_1 \;\sim\; s_0 \,.\, a \notin \operatorname{Im} f_{s_1}$$

This equation can be generalized with parameterized properties:

$$\forall s_0 \in S \,.\, \forall \tau \in \mathcal{T} \,.\, \exists s_1 \;\sim\; s_0 \,.\, p_\tau(s_0, s_1)$$

Notice that anything with an ∀ quantifier is chosen as worst case for the protocol. This corresponds to the game adversary in an indistinguishability game. Anything with an ∃ quantifier has to be chosen in favor of the protocol. This corresponds to the challenger in the game. Finally, the scenarios that are allowed to be compared correspond to the property in the game.

[5] Formally this is equivalent to $\mathbf{Exp}^{\{(s_0,\,s_1)\}\text{–A·IND}}_{\Pi,\mathcal{A}}$ where $\{(s_0, s_1)\}$ is the binary relation that only holds for s_0 and s_1

$$\overbrace{\forall s_0 \in S.\, \forall \tau \in \mathcal{T}}^{\text{Adversary's choice}} \,.\; \overbrace{\exists s_1 \sim s_0}^{\text{Challenger's task}} \;.\; \overbrace{p_\tau(s_0, s_1)}^{\text{Property}}$$

Notice that this description matches our E·IND game. The adversary is choosing s_0 and the parameters τ. The challenger is responsible to show the existence of an indistinguishable scenario that fulfills the property.

Example: Property Inactive Sender is Equivalent to Image Opaqueness. Property *Inactive Sender* ($\overline{\text{Sender1}}_A$) from Definition 4.3 results in the $\overline{\text{Sender1}}_A$–E·IND notion that is equivalent to image opaqueness, as it requires that the chosen agent a is not sending in the second scenario, i.e. $a \notin \operatorname{Im} f_{s_1}$.

6 COMPARISON TO A·IND

An in-depth understanding of the relations between privacy goals allows for an easier analysis and better development of ACNs. We thus compare our new "Exists INDistinguishability" (E·IND) notions with the "All INDistinguishability" (A·IND) notions from the state of the art hierarchy in order to highlight their relations. We have already intuitively stated that A·IND expresses stronger guarantees as E·IND that, however, might not even be needed in all use cases but require an extensive performance overhead (see Section 3). We thus establish the weaker E·IND as provable guarantee for networks aiming at plausible deniability. Formally, we can prove that the following holds for a relevant subset of properties:

$$p^a\text{–A·IND} \Rightarrow p^b\text{–E·IND}\,, \tag{1}$$

where p^a and p^b can be different, but related properties. In subsequent sections, we develop a general toolbox for a hybrid analysis of ACNs, using A·IND and E·IND notions.

6.1 Relevant Subset of Properties

For an easier comparison, we introduce characteristics of properties, which are similar to the earlier introduced satisfiability requirement.

Completely Satisfiable Properties. If a combination (s_0, τ) exists s.t. s_0 is not p_τ*-satisfiable*, the E·IND notion cannot be achieved by any ACN. The adversary simply attacks this combination and wins the game. If no such combination exists, we denote the property as *completely satisfiable* property:

Definition 6.1 (Completely Satisfiable Property). A property p with the possible parameter set $\mathcal{T}$ is denoted as *completely satisfiable* property iff $\forall s_0 \in S. \forall \tau \in \mathcal{T} : p_\tau(s_0, \cdot)$ is *satisfiable*.

Intuitively this means that the adversary cannot win in the E·IND game by submitting a scenario without any matching alternative, as there is no such scenario. To win the E·IND game, she needs to distinguishing the transcripts of two scenarios.

Efficiently Satisfiable Properties. For p–E·IND we iterate over all *valid* challengers. We hence include the challenger who chooses a matching s_1, if such a scenario exists. For the following proofs finding a matching scenario in an efficient way is however useful. We denote any property for which we can efficiently construct a matching s_1, if it exists, as *efficiently satisfiable* property:

Definition 6.2 (Efficiently Satisfiable Property). A property p with the possible parameter set $\mathcal{T}$ and the parameter tuple $\tau \in \mathcal{T}$ is denoted as *efficiently satisfiable* property iff a PPT function $\gamma : S \times \mathcal{T} \rightarrow S$ exists s.t.

$$\forall s_0 \in S. \forall \tau \in \mathcal{T} : \gamma(s_0, \tau) \begin{cases} \in \{s \mid p_\tau(s_0, s)\} & \texttt{if } s_0 \text{ is } p_\tau\text{-satisfiable} \\ = \perp & \texttt{else} \end{cases}$$

For all properties mentioned in [15] γ is an easy construction and in general, most reasonable properties fulfill at least one of the two requirements.

6.2 Comparing p–A·IND with p–E·IND

In A·IND the adversary is allowed to choose both scenarios. In contrast, in E·IND the challenger is choosing the second scenario. Hence, intuitively speaking, E·IND requires a stronger attack that works against any second scenario, i.e., E·IND is easier to achieve and harder to break, and thus a weaker notion. However to translate an E·IND to an A·IND attack, p has to be completely and efficiently satisfiable, as otherwise we cannot know how to choose the second scenario during the attack. In Appendix A.2 we formally show that:

Lemma 6.3. *For every completely and efficiently satisfiable property p with the possible parameter set $\mathcal{T}$ and the parameter tuple $\tau \in \mathcal{T}$ it holds that: p–A·IND $\Rightarrow$ p–E·IND.*

Proof Sketch. A reduction proof: *As the property p is completely and efficiently satisfiable, the PPT adversary on p–A·IND can simply generate the second matching scenario by using the γ function. Fig. 3 visualizes this fact.*

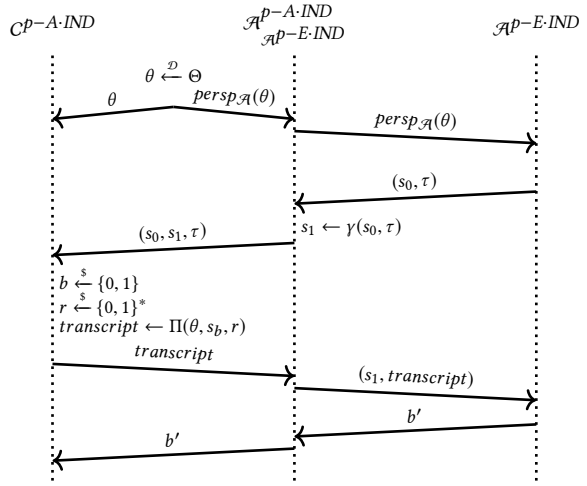

Figure 3: p–A·IND $\Rightarrow$ p–E·IND

6.2.1 Comparing A·IND with E·IND for Related Properties. If we know that some property implies another property, we can show a relation between the corresponding games.

Parameterized Properties in A·IND. In Section 2, we mentioned that for non-parameterized properties p^b–A·IND $\Rightarrow$ p^a–A·IND, if $p^a \Rightarrow p^b$. To show this likewise for parameterized properties, we need to handle the parameter tuple appropriately. Even if the possible parameter sets are equal ($\mathcal{T}_a = \mathcal{T}_b$), they may get interpreted differently for each property. We hence must transform $\tau_a \in \mathcal{T}_a$ into a parameter tuple for p^b s.t. the implication $p^a_{\tau_a}(s_0, s_1) \Rightarrow p^b_{\tau_b}(s_0, s_1)$ holds. We denote this as *parameter transformation*. With this we show the following statement in Appendix A.3:

Lemma 6.4. *Let p^a and p^b be two parameterized properties with the possible parameter sets $\mathcal{T}_a$ and $\mathcal{T}_b$ and the parameter tuples τ_a and τ_b. If an efficient parameter transformation function $\omega_b : S \times S \times \mathcal{T}_a \rightarrow \mathcal{T}_b$ exist s.t.*

$$\forall \tau_a \in \mathcal{T}_a. \forall s_0, s_1 \in S : \quad p^a_{\tau_a}(s_0, s_1) \quad \Rightarrow \quad p^b_{\omega_b(s_0, s_1, \tau_a)}(s_0, s_1)$$

$$\textit{then } p^b\text{–A·IND} \Rightarrow p^a\text{–A·IND}.$$

Proof Sketch. *With the efficient function ω_b we can do a reduction proof similar to Lemma 6.3. We provide a visualization in Fig. 4.*

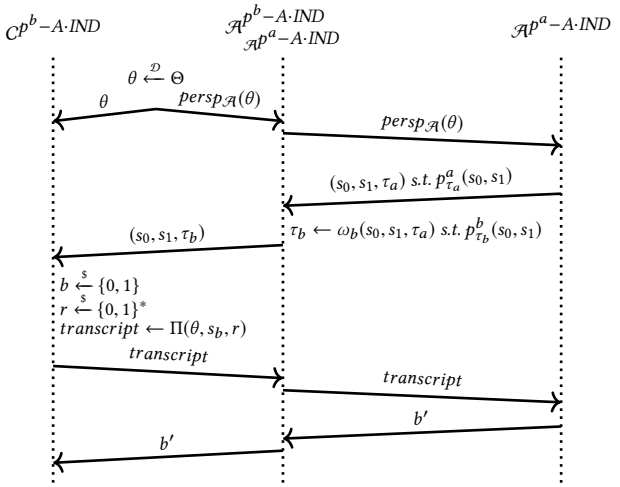

Figure 4: p^b–A·IND $\Rightarrow$ p^a–A·IND

Remark 7. *We denote ω_b as ω_b because it produces the parameter tuple for p^b. In the next subsection, we see the ω_a function, which produces the parameter tuple for p^a.*

Parameterized Properties in E·IND. In A·IND the adversary is restricted by the property because she chooses the attacked pair (s_0, s_1). A more restrictive property results in a harder problem for the adversary, and thus a weaker A·IND notion (see Lemma 6.4). In E·IND, instead, the challenger is restricted by the property. A more restrictive property might exclude the indistinguishable scenarios. Therefore a more restrictive property results in a stronger E·IND notion. Informally p^a–E·IND implies p^b–E·IND, if an efficient parameter transformation exists s.t. $p^a \Rightarrow p^b$. Note that this implication is inverse to the A·IND variant.

There is one more detail: if a scenario-parameter combination is satisfiable with itself, the minimal advantage for attacking this scenario is 0. We hence have to treat these cases separately for our statement, which we prove in Appendix A.4:

Lemma 6.5. *Let p^a and p^b be two parameterized properties with the parameter sets $\mathcal{T}_a$ and $\mathcal{T}_b$ and the parameter tuples $\tau_a \in \mathcal{T}_a$ and $\tau_b \in \mathcal{T}_b$. If an efficient function $\omega_a : S \times \mathcal{T}_b \to \mathcal{T}_a$ exists s.t.*

$$\forall \tau_b \in \mathcal{T}_b . \forall s_0, s_1 \in S : \quad p^b_{\tau_b}(s_0, s_0) \vee \left(p^a_{\omega_a(s_0,\tau_b)}(s_0, s_1) \Rightarrow p^b_{\tau_b}(s_0, s_1)\right)$$

$$\text{then } p^a\text{–}E\cdot IND \;\Rightarrow\; p^b\text{–}E\cdot IND$$

Proof Sketch. *We distinguish the following cases and argue as following in each:*

(1) s_0 is not $p^b_{\tau_b}$-satisfiable: both notions are unachievable.
(2) $p^b_{\tau_b}(s_0, s_0)$ holds: cannot be a successful adversary.
(3) $\neg p^b_{\tau_b}(s_0, s_0)$ and s_0 is $p^b_{\tau_b}$-satisfiable and $p^a_{\omega_a(s_0,\tau_b)}$-satisfiable: we show a reduction proof.
(4) $\neg p^b_{\tau_b}(s_0, s_0)$ and s_0 is $p^b_{\tau_b}$-satisfiable but s_0 is not $p^a_{\omega_a(s_0,\tau_b)}$-satisfiable: p^a–E·IND is unachievable.

Remark 8. *The result for non-parameterized properties is similar. If both properties are non-parameterized, we check if*

$$\forall s_0, s_1 \in S : p^b(s_0, s_0) \;\vee\; \left(p^a(s_0, s_1) \;\Rightarrow\; p^b(s_0, s_1)\right).$$

If p^b is parameterized, but p^a is not, we can use $\omega_a : (s_0, \tau_b) \mapsto ()$. Otherwise, the efficient function ω_a is needed, like for two parameterized properties.

6.3 Conjuncted Properties

Conjuncted properties require that the combination of two or more properties holds. This way individual properties can be used as building blocks for more complex notions. Subsequently, we discuss such combinations for A·IND and E·IND.

6.3.1 Conjuncted Properties in A·IND. For the non-parameterized properties it is clear that p^a–A·IND $\Rightarrow$ (p^a, p^b)–A·IND and p^b–A·IND $\Rightarrow$ (p^a, p^b)–A·IND. This is due to the fact that $(p^a \wedge p^b) \Rightarrow p^a$ and analog for p^b. For parameterized properties the same is true, because if $p^a_{\tau_a}(s_0, s_1) \wedge p^b_{\tau_b}(s_0, s_1)$ holds, then $p^a_{\tau_a}(s_0, s_1)$ holds by definition. We detail this in Appendix A.5.

6.3.2 Conjuncted Properties in E·IND. We find a similar relationship for E·IND. But the direction of the implication is inverted as before. We show in Appendix A.6 that:

Lemma 6.6. *Let p^a and p^b be two parameterized properties with the possible parameter sets $\mathcal{T}_a$ and $\mathcal{T}_b$. Let their conjunction property p^{ab} be $(p^a \wedge p^b)$ with the possible parameter set $\mathcal{T}_{ab} = \mathcal{T}_a \times \mathcal{T}_b$. It holds that p^{ab}–E·IND $\Rightarrow$ p^a–E·IND.*

Proof Sketch. *According to Lemma 6.5 it is sufficient to show that an efficient function $\omega_{ab} : S \times \mathcal{T}_a \to \mathcal{T}_{ab}$ exists. This function picks τ_b at random. If this leads to a $p^{ab}_{(\tau_a,\tau_b)}$-satisfiable combination for every matching s_1 it holds that $p^a_{\tau_a}(s_0, s_1)$. Otherwise p^{ab}–E·IND is unachievable and thus p^{ab}–E·IND $\Rightarrow$ p^a–E·IND holds trivially.*

6.3.3 Comparison including Conjuncted Properties. If p^{ab} is a *completely and efficiently satisfiable* property, we can combine the last lemmas and finally show in Appendix A.7:

Theorem 6.7. *Let p^a and p^b be two parameterized properties with the possible parameter sets $\mathcal{T}_a$ and $\mathcal{T}_b$ and the parameter tuples $\tau_a \in \mathcal{T}_a$ and $\tau_b \in \mathcal{T}_b$ s.t. their conjunction property $p^{ab} : (p^a \wedge p^b)$ is a* completely and efficiently satisfiable *property with the possible parameter set $\mathcal{T}_{ab} = \mathcal{T}_a \times \mathcal{T}_b$ and the parameter tuple $(\tau_a, \tau_b) \in \mathcal{T}_{ab}$. It holds*

$$p^a\text{–}A\cdot IND \;\Rightarrow\; p^b\text{–}E\cdot IND.$$

Proof Sketch. *The previous lemmas show:*

$$p^a\text{–}A\cdot IND \overset{\text{Sec. 6.3.1}}{\Rightarrow} p^{ab}\text{–}A\cdot IND \overset{\text{Lem. 6.3}}{\Rightarrow} p^{ab}\text{–}E\cdot IND \overset{\text{Lem. 6.6}}{\Rightarrow} p^b\text{–}E\cdot IND$$

Remark 9. *If p^{ab} is completely and efficiently satisfiable, then p^a and p^b are completely and efficiently satisfiable too.*

6.3.4 A Different Dimension of Anonymity. The E·IND notions yield a complete new dimension of game based privacy notions. The differences cannot be compensated by just using other properties in the A·IND game, but they are fundamental. To demonstrate this we first define *reasonable* properties. A property is reasonable if the A·IND adversary can attack at least one pair for different scenarios:

Definition 6.8. A property p is denoted *reasonable* iff

$$\exists s_0, s_1 \in S : s_0 \neq s_1 \wedge p(s_0, s_1) = 1$$

We show in Appendix B that there is no reasonable property p such that notion $\overline{\text{Sender1}}_A$–E·IND implies p–A·IND:

Theorem 6.9. *Let p be any* reasonable *property. $\overline{Sender1}_A$–E·IND does not imply p–A·IND:*

$$\overline{Sender1}_A\text{–}E\cdot IND \not\Rightarrow p\text{–}A\cdot IND$$

7 DISCUSSION

In the following, we discuss modifications of our game that allow to express requirements closely related to plausible deniability and to relax the advantage definition in the game (Section 7.1). Moreover, we discuss more fundamental changes to the game that additionally allow to analyze adaptive and active adversaries (Section 7.2), as well as the strength of our plausible deniability notion (Section 7.3)

7.1 Other Variants of Plausible Deniability

We can vary our plausible deniability definition to express related protections. For instance, we can express other requirements for the considered plausible alternative (Section 7.1.1). Also, by varying the advantage requirement, we may allow for distinguishability of a small number of cases (Section 7.1.2).

7.1.1 Changes in the Statement: Varying the Game. Our E·IND definition guarantees that one or some other plausible users *exist*. The accused user might though not know who the plausible alternative is. This is due to our advantage definition in the E·IND game. By iterating over all valid challengers, we model an $\exists$ relationship. However, in some cases a concrete alternative might be required.

Presenting a Concrete Alternative. With a slight modification, we can model this extended goal: In addition to ACN Π, we associate the function $g : \Theta \times S \times \mathcal{T} \to S$ for analysis. g must be efficient and output the corresponding response of the challenger for every protocol state θ, input scenario s_0 and every tuple τ. The returned scenario s_1 must be indistinguishable to achieve this variant of

E·IND. As g is efficient, the blamed user can present the concrete alternative by executing g, given the user has enough information on the protocol state.

Denying an Action without a Single Alternative. Our E·IND definition guarantees that a single, alternative user is also a plausible initiator of the action. Another kind of deniability accepts that no single user is as likely the originator as the suspect, as long as the entirety of alternative users is plausible. To represent this variant in our E·IND game, we allow the challenger to know the randomness of the protocol state before deciding for an alternative scenario s_1 (see Section 4.5, first case). So, the challenger is able to predict every future random decision of the protocol and choose the optimal alternative scenario accordingly. The randomness still cannot be manipulated by the challenger.

7.1.2 Relaxing the Requirement: Varying the Advantage. Requiring a non-negligible advantage of the attack is a strong guarantee, as it allows the protection to fail only in negligibly many cases. We stress however that the advantage definition of our game can easily be adapted to express more relaxed requirements. As in earlier work [15], we can, e.g., consider the case that the protection may fail with small, but non-negligible probability by slightly increasing the allowed advantage.

7.2 Adversary Models

In this Section, we discuss which adversary models require further changes to our E·IND game.

Non-adaptive, passive Adversaries. Our formal definitions are independent of the location of the adversary (local, global, links, nodes etc.). The ACN model Π is capable of representing all non-adaptive and passive adversaries. The only difference between them is the amount of observations included in the transcript.

Adaptive Adversaries. Modeling adaptive adversaries requires a more fundamental adaptation of the game. Kuhn et al. proposed the *Multi-Batch Extension* for the A·IND game [15]. This means that multiple scenarios can be executed consecutively in the ACN, without resetting the protocol state in between. After each execution the adversary gets the transcript and decides for the next scenarios or to submit a guess.

Active Adversaries. To model an active adversary, i.e. one that manipulates the ACN and does not follow the protocol, Kuhn et al. suggest to use *Protocol Queries* [15]. With these queries the adversary can modify the protocol state between adaptive scenarios. For instance the adversary might add or remove nodes from the network. However, the concrete set of queries and their effects depend on the analyzed ACN and adversary model.

7.3 On the Strength of Plausible Deniability

We notice that our notion of plausible deniability for anonymous communication is indeed designed to offer a weak privacy protection. If the adversary can get additional information that was not considered during analysis, she might exclude the alternative. Further, some adversaries might also conduct measures against all *potential* culprits if there are only few. In these settings other notions are preferable. However, being able to express and prove even very weak protection allows us to investigate privacy and performance trade-offs, as well as weak ACN protocols in depth.

8 CONCLUSION

The inevitable overhead required to achieve the "All INDistinguishability" (A·IND) notions renders them insufficient in particular scenarios. We demonstrated this with a bound in P2P networks that is based on the well-known predecessor attack. Hence, we designed the new "Exists INDistinguishability" (E·IND) game with weaker guarantees, which enables privacy proofs with the advantage of game-based definitions even for networks that come with less overhead but weaker privacy requirements.

The use of indistinguishability games ensures easy integration of different deniable actions and information. Moreover, it allow us to compare the new notions to related work that already uses such games [15], and to show that they also generalize to notions based on function views [13], which originally have been proposed for information hiding.

To the best of our knowledge, we are thereby the first to find a bridge between the worlds of game and function view based definitions and at the same time provide a general, formal game-based definition for *plausible deniability* in anonymous communication.

With this, we bootstrap a discussion on the trade-offs between A·IND and E·IND as well as other notions. Additionally, the versatile "Exists INDistinguishability" notions open the door to various improvements of anonymous communication, such that we can analyze even weak privacy goals for (P2P) ACNs to uncover flaws, improve on them, and finally provide provable privacy guarantees.

Acknowledgements

We thank the anonymous reviewers for their valuable comments. Further, we thank Aniket Kate for very insightful discussions on the subject and Paul Syverson for initiating the comparison of the two models. This work was supported by the KASTEL Security Research Labs and the Cluster of Excellence 'Centre for Tactile Internet with Human-in-the-Loop' (EXC 2050/1, Project ID 390696704).

REFERENCES

[1] Michael Backes, Aniket Kate, Praveen Manoharan, Sebastian Meiser, and Esfandiar Mohammadi. 2013. AnoA: A Framework for Analyzing Anonymous Communication Protocols. In *2013 IEEE 26th Computer Security Foundations Symposium*. IEEE, New Orleans, LA, 163–178. https://doi.org/10.1109/CSF.2013.18

[2] Mohit Bhargava and Catuscia Palamidessi. 2005. Probabilistic Anonymity. In *CONCUR 2005 - Concurrency Theory, 16th International Conference, CONCUR 2005, San Francisco, CA, USA, August 23-26, 2005, Proceedings (Lecture Notes in Computer Science)*, Martín Abadi and Luca de Alfaro (Eds.), Vol. 3653. Springer, 171–185. https://doi.org/10.1007/11539452_16

[3] Jens-Matthias Bohli and Andreas Pashalidis. 2011. Relations among Privacy Notions. *ACM Transactions on Information and System Security* 14, 1 (May 2011), 1–24. https://doi.org/10.1145/1952982.1952986

[4] David Chaum. 1988. The Dining Cryptographers Problem: Unconditional Sender and Recipient Untraceability. *Journal of cryptology* (1988).

[5] David L Chaum. 1981. Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. 24, 2 (1981), 5.

[6] Ian Clarke, Oskar Sandberg, Brandon Wiley, and Theodore W. Hong. 2000. Freenet: A Distributed Anonymous Information Storage and Retrieval System. In *Designing Privacy Enhancing Technologies, International Workshop on Design Issues in Anonymity and Unobservability, Berkeley, CA, USA, July 25-26, 2000, Proceedings (Lecture Notes in Computer Science)*, Hannes Federrath (Ed.), Vol. 2009. Springer, 46–66. https://doi.org/10.1007/3-540-44702-4_4

[7] Roger Dingledine, Nick Mathewson, and Paul Syverson. 2004. *Tor: The Second-Generation Onion Router:*. Technical Report. Defense Technical Information Center, Fort Belvoir, VA. https://doi.org/10.21236/ADA465464

[8] Prastudy Fauzi, Sarah Meiklejohn, Rebekah Mercer, and Claudio Orlandi. 2019. Quisquis: A New Design for Anonymous Cryptocurrencies. In *Advances in Cryptology – ASIACRYPT 2019*, Steven D. Galbraith and Shiho Moriai (Eds.). Vol. 11921. Springer International Publishing, Cham, 649–678. https://doi.org/10.1007/978-3-030-34578-5_23

[9] Joan Feigenbaum, Aaron Johnson, and Paul F. Syverson. 2007. A Model of Onion Routing with Provable Anonymity. In *Financial Cryptography and Data Security, 11th International Conference, FC 2007, and 1st International Workshop on Usable Security, USEC 2007, Scarborough, Trinidad and Tobago, February 12-16, 2007. Revised Selected Papers (Lecture Notes in Computer Science)*, Sven Dietrich and Rachna Dhamija (Eds.), Vol. 4886. Springer, 57–71. https://doi.org/10.1007/978-3-540-77366-5_9

[10] Nethanel Gelernter and Amir Herzberg. 2013. On the Limits of Provable Anonymity. In *Proceedings of the 12th ACM Workshop on Workshop on Privacy in the Electronic Society - WPES '13*. ACM Press, Berlin, Germany, 225–236. https://doi.org/10.1145/2517840.2517850

[11] Joseph Y Halpern and Kevin R O'Neill. [n. d.]. Anonymity and Information Hiding in Multiagent Systems. ([n. d.]), 14.

[12] Alejandro Hevia and Daniele Micciancio. 2008. An Indistinguishability-Based Characterization of Anonymous Channels. In *Privacy Enhancing Technologies*, Nikita Borisov and Ian Goldberg (Eds.). Vol. 5134. Springer Berlin Heidelberg, Berlin, Heidelberg, 24–43. https://doi.org/10.1007/978-3-540-70630-4_3

[13] Dominic J. D. Hughes and Vitaly Shmatikov. 2004. Information Hiding, Anonymity and Privacy: A Modular Approach. *J. Comput. Secur.* 12, 1 (2004), 3–36.

[14] Jon Kleinberg. 2000. The small-world phenomenon: An algorithmic perspective. In *STOC*.

[15] Christiane Kuhn, Martin Beck, Stefan Schiffner, Eduard A. Jorswieck, and Thorsten Strufe. 2019. On Privacy Notions in Anonymous Communication. *Proc. Priv. Enhancing Technol.* 2019, 2 (2019), 105–125. https://doi.org/10.2478/popets-2019-0022

[16] Brian Neil Levine, Marc Liberatore, Brian Lynn, and Matthew Wright. 2020. A Forensically Sound Method of Identifying Downloaders and Uploaders in Freenet. In *CCS '20: 2020 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, USA, November 9-13, 2020*, Jay Ligatti, Xinming Ou, Jonathan Katz, and Giovanni Vigna (Eds.). ACM, 1497–1512. https://doi.org/10.1145/3372297.3417876

[17] Shen Noether and Adam Mackenzie. 2016. Ring Confidential Transactions. *Ledger* 1 (2016), 1–18.

[18] Andreas Pfitzmann and Marit Köhntopp. 2000. Anonymity, Unobservability, and Pseudonymity - A Proposal for Terminology. In *Designing Privacy Enhancing Technologies, International Workshop on Design Issues in Anonymity and Unobservability, Berkeley, CA, USA, July 25-26, 2000, Proceedings (Lecture Notes in Computer Science)*, Hannes Federrath (Ed.), Vol. 2009. Springer, 1–9. https://doi.org/10.1007/3-540-44702-4_1

[19] Michael K. Reiter and Aviel D. Rubin. 1998. Crowds: Anonymity for Web Transactions. *ACM Transactions on Information and System Security* 1, 1 (Nov. 1998), 66–92. https://doi.org/10.1145/290163.290168

[20] Steve A. Schneider and Abraham Sidiropoulos. 1996. CSP and Anonymity. In *Computer Security - ESORICS 96, 4th European Symposium on Research in Computer Security, Rome, Italy, September 25-27, 1996, Proceedings (Lecture Notes in Computer Science)*, Elisa Bertino, Helmut Kurth, Giancarlo Martella, and Emilio Montolivo (Eds.), Vol. 1146. Springer, 198–218. https://doi.org/10.1007/3-540-61770-1_38

[21] Sandra Steinbrecher and Stefan Köpsell. 2003. Modelling Unlinkability. In *Privacy Enhancing Technologies, Third International Workshop, PET 2003, Dresden, Germany, March 26-28, 2003, Revised Papers (Lecture Notes in Computer Science)*, Roger Dingledine (Ed.), Vol. 2760. Springer, 32–47. https://doi.org/10.1007/978-3-540-40956-4_3

[22] Paul F. Syverson and Stuart G. Stubblebine. 1999. Group Principals and the Formalization of Anonymity. In *FM'99 - Formal Methods, World Congress on Formal Methods in the Development of Computing Systems, Toulouse, France, September 20-24, 1999, Proceedings, Volume I (Lecture Notes in Computer Science)*, Jeannette M. Wing, Jim Woodcock, and Jim Davies (Eds.), Vol. 1708. Springer, 814–833. https://doi.org/10.1007/3-540-48119-2_45

[23] Matthew K Wright, Micah Adler, Brian Neil Levine, and Clay Shields. 2004. The predecessor attack: An analysis of a threat to anonymous communications systems. *ACM Transactions on Information and System Security (TISSEC)* 7, 4 (2004), 489–522.

A PROOFS

In the following we show the formal proofs for our claims in the above work. We start with a basic proof on the ~ relation (Appendix A.1). Afterwards provide details on the comparison between A·IND and E·IND (Appendix A.2 - A.7). In the end we will be able to show

$$p^a\text{–A·IND} \Rightarrow p^b\text{–E·IND}$$

For relevant subset of properties.

A.1 Relation ~ is an Equivalence Relation

In order to use our game based definition of ~ ($\mathbf{Adv}^{s_0,s_1}_{\Pi,\mathcal{A}}$ in negligible for every adversary) as the observational equivalence in the function view approach we need to that it is an equivalence relation:

Lemma A.1. *The binary relation ~ is an equivalence relation.*

Proof. We show that ~ is symmetric, transitive and reflexive:

~ **is symmetric**: b is chosen uniformly at random it does not matter if s_0 or s_1 is the first scenario.

~ **is transitive**: The probability distribution on the transcripts of s_0 and s_1 only differs by an negligible amount and if the same holds for s_1 and s_2, the probability distribution of transcripts between s_0 and s_2 can only differ twice negligible. Twice negligible is still negligible.

~ **is reflexive**: The ACN Π is getting the exact same input in both cases. Even if Π is probabilistic, the probability distributions of the possible transcripts is equivalent for both scenarios.

□

A.2 p–A·IND ⇒ p–E·IND

We start the comparison of A·IND and E·IND by showing which one is stronger for a fixed property p.

Lemma A.2. *For every completely and efficiently satisfiable property p with the possible parameter set $\mathcal{T}$ and the parameter tuple $\tau \in \mathcal{T}$ it holds that: p–A·IND ⇒ p–E·IND.*

Proof. Suppose a successful adversary on p–E·IND, is given as $\mathcal{A}^{p\text{–E·IND}}$. We reduce this adversary into a successful adversary $\mathcal{A}^{p\text{–A·IND}}_{\mathcal{A}^{p\text{–E·IND}}}$ on the p–A·IND notion, as shown in Algorithm 3.

Algorithm 3 $\mathcal{A}^{p\text{–A·IND}}_{\mathcal{A}^{p\text{–E·IND}}}$

```
1  Method: getPair
2      Θ' ← Arguments
3      (s_0, τ, state_A) ← A^{p_τ–E·IND}(1^k, getScenario, Θ')
4      s_1 ← γ(s_0, τ)
5      return (s_0, s_1, τ, (state_A, s_1))
6
7  Method: attack
8      (transcript, (state_A, s_1)) ← Arguments
9      b' ← A^{p–E·IND}(1^k, attack, s_1, transcript, state_A)
10     return b'
```

Valid adversary on p–A·IND: As p is an *completely and efficiently satisfiable* property, $\{s \mid s \in S : s_0\ p_\tau\ s\}$ is not empty for any $s_0 \in S$ and a matching $s_1 \in \{ \mid s \in S : p_\tau(s_0, s)\}$ can be always be found efficiently.

Valid challenger on p–E·IND: As $s_1 \in \{s \mid s \in S : p_\tau(s_0, s)\}$ and *transcript* is either of $\Pi(s_0)$ or $\Pi(s_1)$, the algorithm is a *valid* challenger for p–E·IND.

Advantage: $\mathcal{A}^{p\text{–E·IND}}$ has an advantage higher then negligible against every *valid* challenger and therefore also against $\mathcal{A}^{p\text{–A·IND}}_{\mathcal{A}^{p}\text{–E·IND}}$. Thus, $\mathcal{A}^{p\text{–A·IND}}_{\mathcal{A}^{p}\text{–E·IND}}$ is successful with more than negligible probability.

This proof is visualized in Fig. 5.

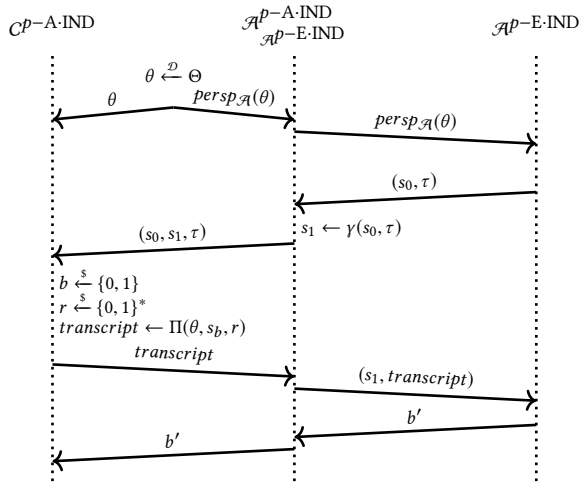

Figure 5: p–A·IND $\Rightarrow$ p–E·IND

□

A.3 p^b–A·IND $\Rightarrow$ p^a–A·IND

In Remark 1 we have already outlined that p^b–A·IND $\Rightarrow$ p^a–A·IND holds if $p^a \Rightarrow p^b$. For parameterized properties we need a way to transform the parameters for p^a into parameters for p^b. In the following lemma we therefore require an efficient parameter transformation function ω_b that outputs the parameters τ_b.

Lemma A.3. *Let p^a and p^b be two parameterized properties with the possible parameter sets $\mathcal{T}_a$ and $\mathcal{T}_b$ and the parameter tuples τ_a and τ_b. If an efficient parameter transformation function $\omega_b : S \times S \times \mathcal{T}_a \rightarrow \mathcal{T}_b$ exist s.t.*

$$\forall \tau_a \in \mathcal{T}_0 . \forall s_0, s_1 \in S : \quad p^a_{\tau_a}(s_0, s_1) \quad \Rightarrow \quad p^b_{\omega_b(s_0, s_1, \tau_a)}(s_0, s_1) \tag{2}$$

$$\textit{then } p^b\textit{–A·IND} \Rightarrow p^a\textit{–A·IND.}$$

Proof. Suppose a successful p^a–A·IND-adversary is given, then we can reduce her into an adversary on p^b–A·IND as shown in Algorithm 4. This proof is visualized in Fig. 6.

Valid adversary on p^b–A·IND: $\mathcal{A}^{p^a\text{–A·IND}}$ outputs a holding combination (s_0, s_1, τ_a) s.t. $p^a_{\tau_a}(s_0, s_1)$. According to Eq. (2) then $p^b_{\tau_b}(s_0, s_1)$, where $\tau_b = \omega_b(s_0, s_1, \tau_a)$, holds too.

Valid challenger on p^a–A·IND: *transcript* is either $\Pi(s_0)$ or $\Pi(s_1)$ and $p^a_{\tau_a}(s_0, s_1)$ holds because s_0, s_1 and τ_a are initially chosen by $\mathcal{A}^{p^a\text{–A·IND}}$. Consequently the Algorithm 4 is a *valid* challenger.

Algorithm 4 $\mathcal{A}^{p^b\text{–A·IND}}_{\mathcal{A}^{p^a}\text{–A·IND}}$

```
Method: getPair
    Θ' ← Arguments
    (s_0, s_1, τ_a, state_A) ← A^{p^a–A·IND}(1^k, getPair, Θ')
    τ_b ← ω_b(s_0, s_1, τ_a)
    return (s_0, s_1, τ_b, state_A)

Method: attack
    (transcript, state_A) ← Arguments
    b' ← A^{p^a–A·IND}(1^k, attack, transcript, state_A)
    return b'
```

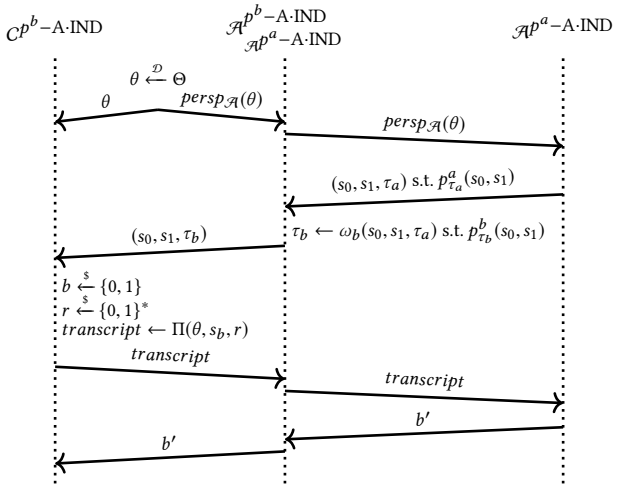

Figure 6: p^b–A·IND $\Rightarrow$ p^a–A·IND

Advantage: For every successful attack of the p^a–A·IND adversary, the p^b–A·IND adversary is also successful. We assumed that the p^a–A·IND adversary has more than negligible advantage. Hence the p^b–A·IND adversary has a advantage, bigger than negligible too. □

A.4 p^a–E·IND $\Rightarrow$ p^b–E·IND

For E·IND the direction of the implication is flipped. Still we need to transform the parameters. This time we require an efficient parameter transformation function ω_a to output parameters for p^a. The signature is different to the ω_b function, shown in Appendix A.3 as we are in the E·IND game now.

Lemma A.4. *Let p^a and p^b be two parameterized properties with the parameter sets $\mathcal{T}_a$ and $\mathcal{T}_b$ and the parameter tuples $\tau_a \in \mathcal{T}_a$ and $\tau_b \in \mathcal{T}_b$. If an efficient function $\omega_a : S \times \mathcal{T}_b \rightarrow \mathcal{T}_a$ exists s.t.*

$$\forall \tau_b \in \mathcal{T}_b . \forall s_0, s_1 \in S : \quad p^b_{\tau_b}(s_0, s_0) \vee \left(p^a_{\omega_a(s_0, \tau_b)}(s_0, s_1) \Rightarrow p^b_{\tau_b}(s_0, s_1)\right) \tag{3}$$

$$\textit{then } p^a\textit{–E·IND} \Rightarrow p^b\textit{–E·IND}$$

Proof. We show that a successful adversary on p^b–E·IND does imply a successful adversary on p^a–E·IND. Algorithm 5 describes this reduced adversary $\mathcal{A}^{p^a\text{–E·IND}}_{\mathcal{A}^{p^b}\text{–E·IND}}$ formally.

Algorithm 5 $\mathcal{A}^{p^a\text{–E·IND}}_{\mathcal{A}^{p^b}\text{–E·IND}}$

```
Method: getScenario
    Θ' ← Arguments
    (s_0, τ_b, state_A) ← A^{p^b–E·IND}(1^k, getScenario, Θ')
    if (p^b_{τ_b}(s_0, s_0)):
        // self satisfiable combination
        s'_0 ←$ S
        τ_a ←$ τ_a
        return (s'_0, τ_a, (state_A, s'_0, 1, θ'))
    else:
        τ_a ← ω_a(s_0, τ_b)
        return (s_0, τ_a, (state_A, s_0, 0, θ')

Method: attack
    (s_1, transcript, (state_A, s_0, wasSelfsatis., Θ')) ← Args
    if wasSelfsatisfiable:
        // simulating a valid challenger
        θ'' ←$ θ'
        r ←$ {0,1}*
        transcript ← Π(θ'', s_0, r)
        A^{p^b–E·IND}(1^k, attack, s_0, transcript, state_A)
        //and guessing at random
        return b' ←$ {0,1}
    else:
        b' ← A^{p^b–E·IND}(1^k, attack, s_1, transcript, state_A)
        return b'
```

For this proof we run a proof by cases. The cases are

(1) s_0 is not $p^b_{\tau_b}$-*satisfiable*
(2) $p^b_{\tau_b}(s_0, s_0)$ holds
(3) $\neg p^b_{\tau_b}(s_0, s_0)$ and s_0 is $p^b_{\tau_b}$-*satisfiable* and $p^a_{\omega_a(s_0,\tau_b)}$-*satisfiable*
(4) $\neg p^b_{\tau_b}(s_0, s_0)$ and s_0 is $p^b_{\tau_b}$-*satisfiable* but is not $p^a_{\omega_a(s_0,\tau_b)}$-*satisfiable*

Case s_0 is not $p^b_{\tau_b}$-satisfiable: In this case p^b–E·IND is *unachievable*. Due to Eq. (3) s_0 is not $p^a_{\omega_a(s_0,\tau_b)}$-*satisfiable*. Thus, p^a–E·IND is *unachievable* too. In this case p^a–E·IND $\Rightarrow$ p^b–E·IND holds trivial.

Case $p^b_{\tau_b}(s_0, s_0)$ holds: If s_0 is $p^b_{\tau_b}$-*satisfiable* with itself (self-satisfiable) the challenger returning $s_1 := s_0$ is *valid* and the advantage of $\mathcal{A}^{p^b}$–E·IND is 0 for this round. Algorithm 5 chooses a random attack combination (s'_0, τ_a) and guesses b' at random for this round. Thus, she has at least the same advantage as $\mathcal{A}^{p^b}$–E·IND for this round[6]. As θ'' is picked form all the possible protocol states and and as $\mathcal{A}^{p^b}$–E·IND has the same perspectiveon the protocol state, she cannot detect that she is communicating with a simulated challenger. If $p^b_{\tau_b}(s_0, s_0)$ holds for every combination (s_0, τ_b), then the property p^b is *reflexive*. Thus, p^b–E·IND is achieved in general and the implication p^a–E·IND $\Rightarrow$ p^b–E·IND holds trivial. Otherwise, if p^b is not *reflexive*, every successful $\mathcal{A}^{p^b}$–E·IND must choose another attack combination at some point in polynomial time, for which $\neg p^b_{\tau_b}(s_0, s_0)$ holds.

Case $\neg p^b_{\tau_b}(s_0, s_0)$ and s_0 is $p^b_{\tau_b}$ and $p^a_{\omega_a(s_0,\tau_b)}$-satisfiable: In Eq. (3) we assumed that a function ω_a exists, that transforms a parameter tuple τ_b of p^b into a parameter tuple $\tau_0 \leftarrow \omega_a(s_0, \tau_b)$ for p^a s.t. $p^a_{\tau_0}(s_0, s_1) \Rightarrow p^b_{\tau_b}(s_0, s_1)$. Due to Eq. (3), the p^a–E·IND challenger, who chooses a s_1 s.t. $s_0\ p^a_{\tau_0}\ s_1$, is also valid for $p^b_{\tau_b}$. The *transcript* is either $\Pi(s_0)$ or $\Pi(s_1)$. Thus, the Algorithm 5 is successful for every successful attack of $\mathcal{A}^{p^b}$–E·IND. We assumed that $\mathcal{A}^{p^b}$–E·IND is successful with more than negligible probability. Algorithm 5 has the same advantage and thus the reduction holds.

Case $\neg p^b_{\tau_b}(s_0, s_0)$ and s_0 is $p^b_{\tau_b}$-satisfiable and is not $p^a_{\omega_a(s_0,\tau_b)}$-satisfiable: In this case we found an *unsatisfiable* combination for p^a and p^a–E·IND is therefore *unachievable*. In this case

$$p^a\text{–E·IND} \Rightarrow p^b\text{–E·IND}$$

holds trivial.

□

A.5 p^a–A·IND $\Rightarrow$ p^{ab}–A·IND

In preparation for Theorem A.7 we discuss conjuncted properties. In order to make the natural implication p^a–A·IND $\Rightarrow$ p^{ab}–A·IND work for parameterized properties we need to give an efficient parameter transformation, that outputs τ_a given τ_{ab}. It is clear that this transformation is rather simple:Âť

Lemma A.5. *Let p^a and p^b be two parameterized properties with the possible parameter sets $\mathcal{T}_a$ and $\mathcal{T}_b$ and the parameter tuples $\tau_a \in \mathcal{T}_a$ and $\tau_b \in \mathcal{T}_b$. Let their conjunction property p^{ab} be $p^a \wedge p^b$ with the possible parameter set $\mathcal{T}_{ab} = \mathcal{T}_a \times \mathcal{T}_b$ and the parameter tuple (τ_a, τ_b). It holds that*

$$p^a\text{–A·IND} \Rightarrow p^{ab}\text{–A·IND}$$

Proof. According to Lemma A.3 it is sufficient to show that an efficient function $\omega_a : S \times S \times \mathcal{T}_{ab} \rightarrow \mathcal{T}_a$ exists s.t.

$$\forall(\tau_a, \tau_b) \in \mathcal{T}_{ab}.\forall s_0, s_1 \in S :$$
$$p^{ab}_{(\tau_a,\tau_b)}(s_0, s_1) \Rightarrow p^a_{\omega_a(s_0,s_1,(\tau_a,\tau_b))}(s_0, s_1)$$

ω_a is given by

$$\omega_a : (s_0, s_1, (\tau_a, \tau_b)) \mapsto (\tau_a)$$

□

[6]Her advantage is higher, if she picks a combination s'_0, τ_0 s.t. s'_0 is not $p^a_{\tau_0}$-*satisfiable*. In this case p^a–E·IND is *unachievable* and p^a–E·IND $\Rightarrow$ p^b–E·IND holds trivial

Remark 10. *Please note the naming ω_a and its signature is inconsistent with Lemma A.3 and actually resembles ω_b from Lemma A.4. This is due to the fact that the indices are named to match the conjuncted properties. ω_a still means it produces parameters for p^a.*

A.6 p^{ab}–E·IND ⇒ p^a–E·IND

For E·IND the direction is flipped. We hence need a parameter transformation from τ_a to τ_{ab}. We can generate the τ_b part of the parameter at random:

Lemma A.6. *Let p^a and p^b be two parameterized properties with the possible parameter sets $\mathcal{T}_a$ and $\mathcal{T}_b$. Let their conjunction property p^{ab} be $(p^a \wedge p^b)$ with the possible parameter set $\mathcal{T}_{ab} = \mathcal{T}_a \times \mathcal{T}_b$. It holds that p^{ab}–E·IND ⇒ p^a–E·IND.*

Proof. According to Lemma A.4 it is sufficient to show that an efficient function $\omega_{ab} : S \times \mathcal{T}_a \to \mathcal{T}_{ab}$ exists s.t.

$$\forall \tau_a \in \mathcal{T}_a. \forall s_0, s_1 \in S: \quad p^a_{\tau_a}(s_0, s_0) \vee \left(p^{ab}_{\omega_{ab}(s_0,\tau_a)}(s_0, s_1) \Rightarrow p^a_{\tau_a}(s_0, s_1)\right)$$

ω_{ab} is given by $\omega_{ab} : (s_0, \tau_a) \mapsto (\tau_a, \tau_b \xleftarrow{\$} \mathcal{T}_b)$. τ_b is picked at random, we hence distinguish between the two cases:

- τ_b is s.t. $p^{ab}_{(\tau_a,\tau_b)}(s_0, \cdot)$ is *satisfiable*
- τ_b is s.t. $p^{ab}_{(\tau_a,\tau_b)}(s_0, \cdot)$ is not *satisfiable*

Case $p^{ab}_{(\tau_a,\tau_b)}(s_0, \cdot)$ is satisfiable: For every matching s_1 also $p^a_{\tau_a}(s_0, s_1)$ holds. The $p^a_{\tau_a}$–E·IND adversary has more than negligible advantage for every such pair. Hence, the $p^{ab}_{(\tau_a,\tau_b)}$–E·IND adversary has more than negligible advantage too.

Case $p^{ab}_{(\tau_a,\tau_b)}(s_0, \cdot)$ is not satisfiable: The notion p^{ab}–E·IND is *unachievable* and thus p^{ab}–E·IND ⇒ p^a–E·IND holds trivial. □

A.7 p^a–A·IND ⇒ p^b–E·IND

Finally we can show p^a–A·IND ⇒ p^b–E·IND for a relevant subset of properties:

Theorem A.7. *Let p^a and p^b be two parameterized properties with the possible parameter sets $\mathcal{T}_a$ and $\mathcal{T}_b$ and the parameter tuples $\tau_a \in \mathcal{T}_a$ and $\tau_b \in \mathcal{T}_b$ s.t. their conjunction property $p^{ab} : (p^a \wedge p^b)$ is a completely and efficiently satisfiable property with the possible parameter set $\mathcal{T}_{ab} = \mathcal{T}_a \times \mathcal{T}_b$ and the parameter tuple $(\tau_a, \tau_b) \in \mathcal{T}_{ab}$.*

If p^{ab} is completely and efficiently satisfiable then p^a–A·IND ⇒ p^b–E·IND.

Proof. We prove this in three steps.

$$p^a\text{–A·IND} \overset{\text{Step 1}}{\Rightarrow} p^{ab}\text{–A·IND} \overset{\text{Step 2}}{\Rightarrow} p^{ab}\text{–E·IND} \overset{\text{Step 3}}{\Rightarrow} p^b\text{–E·IND}$$

Step 1: Follows directly from Lemma A.5.

Step 2: Follows directly from Lemma A.2. For every *completely and efficiently satisfiable* property p^{ab} Step 2 holds.

Step 3: Follows directly from $p^{ab} \Rightarrow p^a$ and Lemma A.4 □

B E·IND IS A DIFFERENT DIMENSION OF ANONYMITY

We demonstrate based on the example $\overline{\text{Sender1}}_A$–E·IND that E·IND and A·IND are indeed two different dimensions of anonymity.

B.1 Example: $\overline{\text{Sender1}}_A$–E·IND

No *reasonable* A·IND notion is weaker than $\overline{\text{Sender1}}_A$–E·IND. As a reasonable A·IND notion we understand a notion in which the adversary is allowed to attack at least one pair of differing scenarios. We describe an $\overline{\text{Sender1}}_A$–E·IND system by two equivalence classes $eqcls_0$ and $eqcls_1$ of scenarios. $eqcls_0$ contains a very simple set of scenarios, s.t. this class achieves $\overline{\text{Sender1}}_A$–E·IND, while $eqcls_1$ contains the remaining scenarios and is also $\overline{\text{Sender1}}_A$–E·IND. Next, we show that this whole system is still $\overline{\text{Sender1}}_A$–E·IND, if we move any arbitrary scenario from $eqcls_1$ to $eqcls_0$ and vice versa. We hence, show that we can separate every arbitrary pair of scenarios and still be $\overline{\text{Sender1}}_A$–E·IND. As we assume that at least one comparable pair of differing scenario exists for any reasonable p–A·IND notion, we can separate exactly this pair. Hence, the p–A·IND adversary is successful.

Theorem B.1. *Let p be any* reasonable *property. $\overline{Sender1}_A$–E·IND does not imply p–A·IND:*

$$\overline{Sender1}_A\text{–E·IND} \not\Rightarrow p\text{–A·IND}$$

Proof. This lemma is equivalent to:

$$\begin{aligned} &\forall \text{ reas. } p.: && \overline{\text{Sender1}}_A\text{–E·IND} \not\Rightarrow p\text{–A·IND} \\ &\qquad\Longleftrightarrow \\ &\forall \text{ reas. } p. \exists sys \in Sys: && sys \text{ is } \overline{\text{Sender1}}_A\text{–E·IND} \wedge \\ &&& sys \text{ is not } p\text{–A·IND} \end{aligned}$$

For every such p, there exists at least one p-comparable pair $s_0, s_1 \in S$ with $s_0 \neq s_1$. We provide a $\overline{\text{Sender1}}_A$–E·IND system for every possible case of s_0, s_1. Note that φ represents the empty scenario with no real communication. As mentioned in Remark 6 we can give a system by its $\sim$ equivalence classes. We make use of this fact for this proof.

Case (1) $s_0 \neq \varphi \wedge s_1 \neq \varphi$: We describe the counterexample system by two equivalence classes $eqcls_0$ and $eqcls_1$ as follows:

$$\begin{aligned} eqcls_0 &:= \{\varphi, s_0\} \\ eqcls_1 &:= S \setminus eqcls_0 \end{aligned}$$

$\overline{Sender1}_A$–E·IND: If the $\overline{\text{Sender1}}_A$–E·IND adversary attacks $eqcls_1$, the challenger returns an indistinguishable scenario from $eqcls_1$ where a is not active. The existence of such a scenario is trivial. Otherwise, if the $\overline{\text{Sender1}}_A$–E·IND adversary attacks $eqcls_0$, the challenger returns the indistinguishable φ scenario.

Not p–A·IND: $p(s_0, s_1)$ holds and s_0, s_1 are in different equivalence classes, and thus p–A·IND is broken.

Case (2) $s_0 = \varphi \wedge s_1 \neq \varphi$ or $s_0 \neq \varphi \wedge s_1 = \varphi$: The counterexample system consists of two equivalence classes again:

$$\begin{aligned} eqcls_0 &:= \{\varphi\} \\ eqcls_1 &:= S \setminus \{\varphi\} \end{aligned}$$

$\overline{Sender1}_A$–E·IND: Same argumentation as in case (1).

Not p–A·IND: Same argumentation as in case (1).

In this system the adversary can decide if any real communication takes place or not. This proof does also hold for more p-comparable pairs, because one p-comparable and distinguishable pair is enough to break the A·IND notion. □

We can not find any A·IND notion equivalent to $\overline{\text{Sender1}}_A$–E·IND. In every reasonable notion at least one pair of differing scenarios is comparable i.e., $s_0 \neq s_1$ and $p(s_0, s_1)$. If such a pair exists, the A·IND notion is not achieved in general. The concept of this proof is equivalent for lot of E·IND notions we observed. E·IND is therefore a completely different dimension of anonymity and cannot be represented in the proposed A·IND game of Kuhn et al.[15].

B.2 The other way around

On the other hand we show that *Sender Unobservability* ($S\overline{O}$–A·IND) is stronger than $\overline{\text{Sender1}}_A$–E·IND. According to Theorem 6.7 it is sufficient to show that their conjunction property $E_S \wedge \overline{\text{Sender1}}_A$ is a completely and efficiently satisfiable property. To show this, we describe the construction γ function: Given a input scenario $s_0 \in S$ and the parameter $a \in A$ for the $\overline{\text{Sender1}}_A$ property we can easily construct a satisfying s_1 by choosing a random agent $a^* \xleftarrow{\$} A \setminus \{a\}$ and replacing every occurrence of sender a by a^*. This yields the new scenario s_1 in which a is not longer active. It holds that $E_S(s_0, s_1) \wedge \overline{\text{Sender1}}_A(s_0, s_1)$. Note that this also holds for $s_0 = \varphi$, the scenario with no communication.