Defining Anonymity and Privacy
(extended abstract)

Vitaly Shmatikov
SRI International
shmat@csl.sri.com

Dominic J. D. Hughes
Stanford University
dominic@cs.stanford.edu

Abstract

We consider several notions of anonymity and privacy and develop formal definitions of what it means for a communication protocol to be "anonymous." The first group of definitions is based on the attacker's knowledge of the communication graph and is close to the intuitive semantics of anonymity. The second group is based on observational equivalence of protocol instances, and is sound and complete with respect to attacker knowledge. Defining anonymity in this way clarifies the relation between concepts such as sender and recipient anonymity, untraceability, unlinkability, and privacy, and enables formal verification of anonymity properties.

1 Introduction

Anonymity is a long-standing concern of protocol designers in the areas of Internet communications, electronic cash, mobile phone networks, database privacy, etc. [Cha81, Cha88, SGR97, RR98, CB95, SMA95, SS98]. There have been, however, relatively few attempts to give a rigorous formal definition of what it means for a protocol to be "anonymous." The problem is complicated by the confusion between terms such as anonymity, untraceability, and privacy which are often used almost interchangeably.

Our main contribution is to develop formal definitions of several anonymity and privacy properties. Intuitively, both anonymity and privacy deal with the attacker's knowledge of a directed graph of communications between agents. We formalize this intuition with a "graph knowledge" semantics. Since these knowledge-based properties cannot be verified directly using formal techniques, we then define anonymity and privacy properties as observational equivalence of protocol instances, which is sound and complete with respect to knowledge semantics. Observational equivalence-based definitions clarify the relationship between anonymity, untraceability, unlinkability, and privacy, and are amenable to formal verification.

One of the insights is the fundamental difference between two broad classes of identity-related protocol properties: anonymity and privacy. In the rest of this paper, we will say that a protocol provides *anonymity* if observable communication between agents does not leak any information about their identities. A protocol provides *privacy* if no agent can use the protocol to learn how other agents make identity-based decisions. Privacy in this sense can be viewed as protection of secondary information [SMA95]: who is willing to communicate with whom, what is the relation between certain agents, etc.

Other formal definitions of anonymity include a special-purpose epistemic logic of Syverson and Stubblebine [SS99] and a CSP-based definition by Schneider and Sidiroupoulos [SS96]. Our framework is sufficiently general to express most interesting anonymity properties (such as unlinkability in the MIX-net [Cha81] sense) and does not depend on the choice of a particular protocol formalism. Therefore, anonymity properties as defined in this paper can be verified with a wide variety of formal methods and proof techniques.

Acknowledgements We are grateful to Martín Abadi, Hubert Comon, Veronique Cortier, and Dahlia Malkhi for very enlightening discussions that helped us understand the meaning of various anonymity properties and refine the definitions presented in this paper.

2 Informal Definitions

We start by giving informal definitions of anonymity and privacy. Since there is no universally accepted terminology in the field, our understanding of any given term may differ from that preferred by other researchers. This is not intended to be a definitive dictionary of anonymity terms, but rather an intuitive explanation of how they are interpreted in the rest of the paper. Another proposal for anonymity terminology has been developed by Pfitzmann *et al.* [PKS01].

Our informal definitions assume that the protocol specification describes how 2 agents (*sender* and *recipient*) communicate with each other. Communication may involve intermediate agents, *e.g.*, anonymizing servers or Web-based email services, but if their identities need not be protected we do not consider them as part of the abstract communication graph. Instantiating the specification with concrete agent names represents one communication instance. We describe anonymity and privacy properties by stating what the attacker is permitted (and not permitted) to learn by observing one or more instances of communication executed concurrently. In case of properties that are symmetric for senders and recipients, we only define sender properties. Definitions for recipient properties can be obtained by replacing "sender" with "recipient" and "originates from" with "intended for."

Our hierarchy of anonymity properties is shown in figure 1. A solid arrow from property P_1 to property P_2 denotes implication: if the protocol satisfies P_1, then it necessarily satisfies P_2. Dashed arrows are conjunctions.

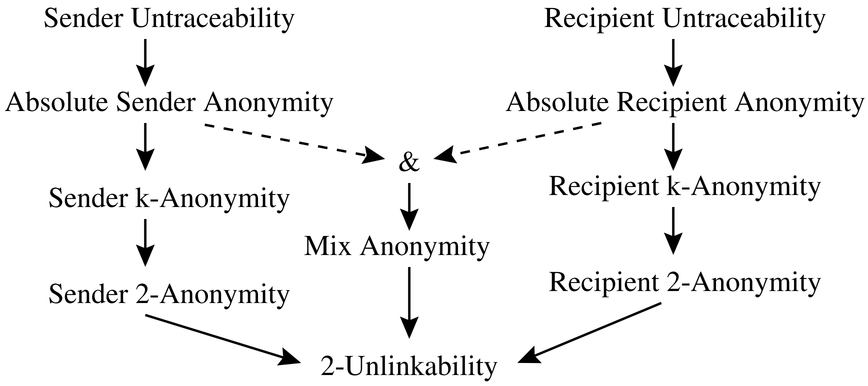

Figure 1: Hierarchy of anonymity properties

Untraceability. We say that a protocol provides *sender untraceability* if the attacker cannot link any observed communication to any subset of senders smaller than the entire set of agents known to the attacker. The attacker should not be able to determine whether two communications originate from the same agent, even if the identity of that agent is unknown.

Anonymity. We say that a protocol provides *absolute sender anonymity* if, as in untraceability, every agent known to the attacker is a plausible sender for every observed communication. The attacker *may* be able, however, to observe that two communications originate from the same sender as long as the latter's identity remains unknown. Absolute anonymity (but not untraceability) can be implemented via persistent pseudonyms, *e.g.*, Web-based email accounts, assuming the attacker cannot monitor communication between the user's computer and the email website.

A protocol provides *sender k-anonymity* if, for every instance of communication, there are at least k agents such that the attacker cannot determine which of them is the actual sender. This notion is similar to k-anonymity as defined by Samarati and Sweeney [SS98] and lineup anonymity [AMP01].

Unlinkability. Unlinkability is the weakest property considered in this paper. We say that a protocol provides *2-unlinkability*, if the attacker cannot determine *both* the sender and the recipient for any instance of communication.

Mix anonymity. Mix anonymity is a stronger form of unlinkability. This is the kind of anonymity provided by approaches based on message blending such as Chaum's MIX-nets [Cha81]. We say that a protocol provides *mix anonymity* if the attacker is permitted to observe which agents executed the protocol as senders or recipients, and to count communications leaving each sender and arriving to each recipient, but not to determine which of the observed agents were involved in any particular instance of communication. This corresponds to a situation where the "Big Brother" attacker may observe the wires connecting everyone's computer to the network, but all communications travel through a secure server(s) which blends them together and distributes to their respective destinations in a manner which prevents the attacker from linking the origin and destination of any given message.

Note that mix anonymity is only meaningful if there are more than 2 communicating agents, in which case it implies 2-unlinkability. If a protocol provides absolute anonymity for both senders and recipients, then it is also mix-anonymous.

Privacy. While anonymity properties defined above deal with identity protection in observable communication, we understand privacy as secrecy of the agents' identity-based decisions. To illustrate by example, consider a medical clinic's website which executes different protocols with patients and non-patients, or a mobile network server which executes different protocols with subscribers, roaming users from other networks, and unauthorized users trying to access it. In both case, the agent's behavior is based on the identity of the other party in the protocol.

We say that a protocol provides *privacy* if the attacker cannot learn how other agents make identity-based decisions. Note that privacy is independent of anonymity. In particular, a protocol may provide anonymity, but not privacy. For example, a medical records retrieval protocol may be such that the attacker cannot learn the identity of any patient or non-patient by observing other agents' communication with the clinic's website, but it can find out whether a particular person is a patient by sending a fake request to the clinic and analyzing the clinic's response (see section 7 for an example).

3 Abstract Communication Model

Let A be a finite set of agent names $\{a_1, \ldots, a_l\}$. Given any communication protocol, consider a concurrent execution of one or more of its instances. Let $L \subseteq A \times A$ be the ***active communication set***, defined as follows: a ***link*** (ordered pair of names) $l = \langle a, b \rangle \in L$ for some $a, b \in A$ *iff* a executes an instance of the protocol with b on an observable communication channel. Say that a is the ***sender*** of l, and b the ***recipient*** of l.

Given a set T of ***agent types***, let $\tau : A \times A \to T$ be an A-indexed assignment of types. For example, with $T = \{\mathsf{subscriber}, \mathsf{roaming}, \mathsf{unauthorized}\}$, $\tau(\langle a, b \rangle) = \mathsf{roaming}$ could be interpreted as "cellphone a has roaming rights on network b." In general, τ helps model protocols where participants' behavior varies depending on the identity of their counterparty. A more general framework might express a dynamically changing τ, or even τ that is an arbitrary program.

Let $p(\langle a, b \rangle, \tau)$ be the syntactic specification of a single instance of the protocol between agents a and b in a suitable cryptographic process calculus, *e.g.*, the spi-calculus [AG99]. Given an active communication set $L = \{\langle a_1, b_1\rangle, \ldots, \langle a_n, b_n\rangle\}$, define ***observable communication***

$$P(L, \tau) \stackrel{\triangle}{=} p(\langle a_1, b_1\rangle, \tau) \mid \cdots \mid p(\langle a_n, b_n\rangle, \tau) \mid S$$

where S is the secure system process, if any (*e.g.*, S may model the behavior of the anonymous remailer or central blender). It is possible that $a_i = a_j$ or $b_i = b_j$ for some $i \neq j$, *i.e.*, the same agent may be involved in several conversations. Informally, set L specifies which agents are talking to each other, and type assignment τ determines all identity-based decisions made by the participants.

Communication graphs. Any observable communication can be represented by a communication graph, defined as follows:

DEFINITION 1 *Let M be a set of abstract links, let A be a set of agents, and let T be a set of agent types. A* ***communication graph*** *$C = \langle \mathsf{sr}_C, \tau_C\rangle$ over (M, A, T) is a directed colored multi-graph comprising functions of the following type:*

$$M \xrightarrow{\mathsf{sr}_C} A \times A \xrightarrow{\tau_C} T$$

We write $\mathsf{s}_C : M \to A$ (*sender*) and $\mathsf{r}_C : M \to A$ (*recipient*) for the composites $M \stackrel{\mathsf{sr}_C}{\to} A \times A \stackrel{\pi_1}{\to} A$ and $M \stackrel{\mathsf{sr}_C}{\to} A \times A \stackrel{\pi_2}{\to} A$ respectively, where π_i is i^{th} projection.

From the viewpoint of the attacker, each abstract link can be thought of as an instance of communication with unknown endpoints. Given a graph C, there is a bijection between abstract links $M_C = \{m_1, \ldots, m_n\}$ and concrete links $L(C) = \{\langle \mathsf{s}_C(m_1), \mathsf{r}_C(m_1)\rangle, \ldots, \langle \mathsf{s}_C(m_n), \mathsf{r}_C(m_n)\rangle\}$. The goal of the attacker is to determine how each abstract link is instantiated by finding the mappings s_C and r_C from the links to their senders and recipients.

We write $\mathbb{C}_{M,A,T}$ for the set of communication graphs over M, A, T.

Protocol equivalence. Informally, two processes are *observationally equivalent* if no context can distinguish them. In formal models of security protocols based on process calculi [AG99, LMMS99, BNP99], it is common to define security properties such as secrecy and authentication in terms of observational equivalence (or related concepts such as computational indistinguishability) between an instance of the protocol and that of an "ideal" protocol which is secure by design and serves as the specification of the desired property.

For the purposes of this abstract, we will use observational equivalence somewhat loosely and say that $P \sim_E Q$ *iff* there is no test that can be feasibly performed by the attacker acting in coalition with all agents from set E (or, more precisely, by the attacker who possesses decryption keys of all agents in E) that would distinguish P and Q. Roughly, P and Q are may-testing equivalent. Proof techniques for observational equivalence depend on the choice of a particular process calculus and attacker model - see, *e.g.*, [AG99, BNP99]. Some of the formalisms are amenable to fully mechanized verification (*e.g.*, via trace-by-trace comparison). The technical details of such verification are beyond the scope of this abstract.

Graph equivalence. Any protocol P induces an equivalence relation on the set of communication graphs over M, A, T as follows:

Definition 2 (Graph equivalence) *For C, C' communication graphs over the same triple M, A, T, define*

$$C \simeq_P C' \quad \triangleq \quad P(L(C), \tau_C) \sim_{A \setminus \mathsf{actv}(C,C')} P(L(C'), \tau_{C'})$$

where $\mathsf{actv}(C, C') = \mathsf{s}_C(X) \cup \mathsf{r}_C(X) \mid X = \{m \in M \text{ s.t. } \mathsf{sr}_C(m) \neq \mathsf{sr}_{C'}(m)\}$ is the set of agents whose communication differs between C and C'.

4 Graph Knowledge Semantics

4.1 Representation of knowledge

Suppose that a function $f : X \to Y$ can be determined only up to its membership in a set of functions F.

- ***Value opaqueness***
  - Given $k \geq 2$, f is ***value k-opaque*** if the value of f can be determined only up to a lineup of size k:

$$\forall\, x \in X.\, \exists\, f_1, \ldots, f_k \in F.\, [i \neq j \Rightarrow f_i(x) \neq f_j(x)]$$

  - Given $Z \subseteq Y$, f is ***value Z-opaque*** if the value of f can be determined only up to the lineup Z:

$$\forall\, x \in X.\, \forall y \in Z.\, \exists\, g \in F.\, g(x) = y$$

  - f is ***absolutely value opaque*** if f is value Y-opaque.

- f is ***equality opaque*** if it is impossible to determine whether f produces equal values on distinct inputs:

$$\forall\, x \neq x' \in X.\, \exists\, g \in F.\, g(x) \neq g(x')$$

- f is ***image opaque*** if it is impossible to determine any point in the image of f:

$$\forall\, y \in Y.\, \exists\, g \in F.\, y \notin \mathsf{im}\, g = \{g(x) \mid x \in X\}$$

Sender untraceability	Attacker may not know anything about senders	s_C is absolutely value opaque and equality opaque
Absolute sender anonymity	Attacker may not know senders but may know whether 2 communications originate from the same (unknown) sender	s_C is absolutely value opaque
Sender k-anonymity	Attacker may only know senders up to a lineup of size k	s_C is value k-opaque
Mix anonymity	Attacker may know the sets of actual senders and recipients	s_C is value $\mathsf{im}\,\mathsf{s}_C$-opaque; r_C is value $\mathsf{im}\,\mathsf{r}_C$-opaque
Unlinkability	Attacker may not know both sender and recipient	sr_C is value 2-opaque
Privacy	Attacker may not determine relation between any 2 agents	τ_C is value 2-opaque

Table 1: Knowledge-based definitions.

Note that for any k value k-opaqueness implies image opaqueness, and Z-opaqueness implies image opaqueness when $Z \not\subseteq \mathsf{im}\, f$.

Define a set F of functions $X \to Y$ to be value k-opaque (resp. value Z-opaque, absolutely value opaque, equality opaque, image opaque) if each $f \in F$ is value k-opaque (resp. value Z-opaque, absolutely value opaque, equality opaque, image opaque). Note that this is independent of the choice of f.

Attacker's approximate knowledge of a communication graph C over M, A, T can be expressed in terms of opaqueness of functions underlying the definition of communication graph: $\mathsf{sr} : M \to A \times A$, $\mathsf{s} : M \to A$, $\mathsf{r} : M \to A$, and the typing function $\tau : A \times A \to T$.

4.2 Definitions based on graph knowledge

In table 1, we formalize the anonymity and privacy properties introduced informally in section 2 by stating them in terms of the attacker's knowledge of the communication graph C associated with the protocol. Recipient properties are identical to sender properties upon substituting r_C for s_C.

5 Observational Equivalences

Our usage of observational equivalence is somewhat different from conventional equivalence-based definitions. Instead of requiring equivalence between the protocol and its ideal specification, we define both anonymity and privacy as observational equivalence of particular combinations of protocol instances that we will call "scenarios." Our scenarios are constructed in such a way that an attacker with a certain kind of knowledge about protocol participants can feasibly tell them apart. Therefore, observational equivalence of the scenarios implies that the attacker cannot extract the needed knowledge from its observations of the protocol, and vice versa.

We define anonymity properties (untraceablility, sender and recipient anonymity, mix anonymity, unlinkability) via observational equivalence of communication graphs representing different communication scenarios. By contrast, privacy is defined via observational equivalence of the same communication graph under different typings, which represent different relations between agents. This brings to light the difference between anonymity and privacy. Anonymity deals with invariance of observable communication w.r.t. agent identities, whereas privacy means that an active attacker cannot learn anything about relations between agents other than herself.

5.1 Definitions based on observational equivalence

Absolute sender anonymity. Attacker cannot determine the sender of any abstract link m, since for any agent a there is an observationally equivalent graph in which a is the sender of m:

$$\begin{array}{rcl}\mathsf{ASA}_{M,A,T}(P) & \stackrel{\triangle}{=} & \forall C \in \mathbb{C}_{M,A,T}\,.\,\forall m \in M.\,\forall a \in A.\,\exists C' \in \mathbb{C}_{M,A,T}\,. \\ & & \quad C' \simeq_P C \;\wedge\; \mathsf{s}_{C'}(m) = a\end{array}$$

Sender untraceability. Absolute sender anonymity, together with attacker not being able to detect when two abstract links come from the same sender. Hence for any pair of abstract links that have the same sender, there exists an observationally equivalent graph in which they have distinct senders.

$$\begin{array}{rcl}\mathsf{SUT}_{M,A,T}(P) & \stackrel{\triangle}{=} & \mathsf{ASA} \;\wedge\; \forall C \in \mathbb{C}_{M,A,T}\,.\,\forall m \in M.\,\forall n \in M. \\ & & \quad \mathsf{s}_C(n) = \mathsf{s}_C(m).\,\exists C' \in \mathbb{C}_{M,A,T}\,. \\ & & \quad C' \simeq_P C \;\wedge\; \mathsf{s}_{C'}(m) \neq \mathsf{s}_{C'}(n)\end{array}$$

Sender k-anonymity. Similar to absolute sender anonymity, but weakening the existence of one observationally equivalent graph for every agent to the existence of k graphs yielding a lineup of k possible senders for any given abstract link.

$$\begin{aligned}\mathsf{S}k\mathsf{A}_{M,A,T}(P) \;\triangleq\; & \forall C \in \mathbb{C}_{M,A,T}\,.\,\forall m \in M.\,\exists C_1,\ldots,C_{k-1} \in \mathbb{C}_{M,A,T}\,.\\ & C_i \simeq_P C \;\wedge\; \mathsf{s}_{C_i}(m) \neq \mathsf{s}_C(m) \;\wedge\; [i \neq j \Rightarrow \mathsf{s}_{C_i}(m) \neq \mathsf{s}_{C_j}(m)]\end{aligned}$$

Recipient properties. Recipient properties ARA, RUT and $\mathsf{R}k\mathsf{A}$ are are omitted since they are obtained from sender properties by substituting recipient r for sender s throughout.

Mix anonymity. Mix anonymity is the conjunction of a sender clause and a recipient clause. Attacker knows who sent and who received communication, but does not know who sent or received what: for any abstract link m there exists an observationally equivalent graph in which m was sent by any one of the other senders $a \in \mathsf{im}\,\mathsf{s}$, and an observationally equivalent world in which m was received by one of the other recipients $a \in \mathsf{im}\,\mathsf{r}$.

$$\begin{aligned}\mathsf{MIX}_{M,A,T}(P) \;\triangleq\; & \forall C \in \mathbb{C}_{M,A,T}\,.\,\forall m \in M.\\ & [\forall a \in \mathsf{im}\,\mathsf{s}_C\,.\,\exists\, C' \in \mathbb{C}_{M,A,T}\,.C' \simeq_P C \;\wedge\; \mathsf{s}_{C'}(m) = a] \;\wedge\\ & [\forall a \in \mathsf{im}\,\mathsf{r}_C\,.\,\exists\, C' \in \mathbb{C}_{M,A,T}\,.C' \simeq_P C \;\wedge\; \mathsf{r}_{C'}(m) = a]\end{aligned}$$

2-Unlinkability. Analogous to sender 2-anonymity, but with sender-recipient pair in place of sender. Thus for any link m there exists an observationally equivalent graph in which *either* m has a different sender, *or* m has a different recipient.

$$\begin{aligned}\mathsf{2UL}_{M,A,T}(P) \;\triangleq\; & \forall C \in \mathbb{C}_{M,A,T}\,.\,\forall m \in M.\,\exists C' \in \mathbb{C}_{M,A,T}\,.\\ & C' \simeq_P C \;\wedge\; \mathsf{sr}_{C'}(m) \neq \mathsf{sr}_C(m)\end{aligned}$$

5.2 A word of caution

The implications of definitions given in sections 4.2 and 5.1 can be quite subtle. Consider, for example, a protocol that provides mix-anonymity and the following communication graph:

$$\begin{tikzcd} a \arrow[r, "m"] & c \\ b \arrow[ur, shift left, "p"] \arrow[ur, shift right, "q"'] \arrow[r, "n"'] & d \end{tikzcd}$$

If the attacker is permitted to observe the senders and the recipients, it can infer that b is communicating with c because b is involved in 2 communications, and since d is involved in only 1 communication, one of b's communications must be with c. Even though this knowledge could be considered an attack on anonymity, it is *not* a violation of mix-anonymity as defined above. For any given instance of communication, the attacker cannot determine the sender or the recipient. It knows that *either* p, *or* q is an instance of b communicating with c, but if the protocol is mix-anonymous it cannot know which one.

That our definition of mix-anonymity permits this kind of inference does not indicate a problem with the definition. In fact, a realistic mix-anonymity system employing, for example, a central anonymizing blender, would be exposed to the same attack. The user must take care to fully understand what is—and is not—implied by various anonymity properties defined above.

5.3 Privacy

Our definition of privacy is quite subtle. In particular, it permits the attacker to observe the difference if the typing of communicating agent pairs changes. Consider the case when the recipient executes observably different protocols with different senders. For example, a clinic may execute different protocols with patients and non-patients. The attacker may observe that the clinic's website is communicating with a patient, but as long as the identity of the patient is protected, this knowledge is not harmful.

No agent should be able, however, to determine the typing of any agent pair that does not involve himself. In the clinic example, the attacker may learn that the clinic has patients and non-patients, and that it executes different protocols with them, but not whether a particular agent is a patient.

$$\begin{array}{rcl}\mathsf{PRV}_{M,A,T}(P) & \stackrel{\triangle}{=} & \forall C \in \mathbb{C}_{M,A,T}\,.\,\forall \tau : A \times A \to T.\, \exists \tau' : A \times A \to T. \\ & & \text{s.t. } \tau' \neq \tau \wedge \forall m \in M_C\ \tau(\mathsf{sr}_C(m)) = \tau'(\mathsf{sr}_C(m)) \\ & & \quad P(L(C), \tau) \sim_{A \setminus \mathsf{actv}(\tau,\tau')} P(L(C), \tau')\end{array}$$

where $\mathsf{actv}(\tau, \tau') = \{a \in A \mid \exists x \text{ s.t. } \tau(\langle a, x\rangle) \neq \tau'(\langle a, x\rangle) \text{ or } \tau(\langle x, a\rangle) \neq \tau'(\langle x, a\rangle)\}$.

6 Soundness and Completeness

In this section we verify the correctness of our observational equivalence definitions of anynomity and privacy properties with respect to attacker knowledge semantics.

Let Π be any of the anonymity or privacy properties defined in section 2, and write Π^{obs} for the associated observational equivalence predicate, as defined in section 5. Define the ***knowledge predicate*** Π^{kn} associated with Π, on protocols P, as follows: $\Pi^{\text{kn}}_{M,A,T}(P)$ holds *iff* every $\simeq_P$-equivalence class of communication graphs is opaque in every sense listed in the third column of the row associated with Π in table 1.

THEOREM 1 (SOUNDNESS AND COMPLETENESS) *For each anonymity and privacy property* Π *introduced in section 2, the associated observational equivalence and knowledge predicates coincide. In other words, for all* M, A, T *and protocols* P,

$$\Pi^{kn}_{M,A,T}(P) \quad \Leftrightarrow \quad \Pi^{obs}_{M,A,T}(P)$$

7 Example: Private Authentication

We use the private authentication protocol by Martín Abadi [Aba01] to illustrate our definitions. The goal of the protocol is to authenticate sender A to recipient B without revealing their identities to other parties. All agents communicate on a single channel where establishing the source of a message is difficult (*e.g.*, the agents are mobile devices broadcasting on a single frequency). Each agent has a list of "friends" with whom it is willing to communicate. In our framework, this is modeled by function τ s.t. $\tau(\langle a, b\rangle)$ is true *iff* b is willing to communicate with a. The protocol works as follows:

$$\begin{array}{lcl} A \to B & : & m_A = "hi", \{"hi", N_A, A\}_{K_B} \\ B \to A & : & \left\{ \begin{array}{l} m^1_B = "ack", \{"ack", N_A, N_B, A\}_{K_A} \\ \qquad\qquad \text{if } m_A \text{ is encrypted with } K_B \text{ and } \tau(\langle A, B\rangle) = \mathsf{true} \\ m^2_B = "ack", \{N'_B\}_K \\ \qquad\qquad \text{otherwise } (K \text{ is s.t. only } B \text{ knows } K^{-1}) \end{array} \right. \end{array}$$

If m_A did not contain N_A under encryption, the protocol would not provide sender anonymity. Assuming there are 2 agents a_1 and a_2, $P(\{\langle a_1, b\rangle\}, \tau) \not\simeq P(\{\langle a_2, b\rangle\}, \tau)$ even if $\tau(\langle a_1, b\rangle) = \tau(\langle a_2, b\rangle)$. The attacker can compute $m_1 = "hi", \{"hi", a_1\}_{K_b}$ and $m_2 = "hi", \{"hi", a_2\}_{K_b}$, and detect the difference between the two protocol instances by comparing the observed m_A with $m_{\{1,2\}}$.

The purpose of the decoy message m^2_B is more subtle. If, for example, $m^2_B = \{\}$ (*i.e.*, B simply does not respond if it does not want to talk to A), then an active attacker can learn the value of $\tau(\langle A, B\rangle)$ for any A by computing the corresponding m_A and sending it to B. It won't be able to read m^1_B since it is encrypted by A's key, but it can observe the difference between m^1_B and m^2_B. Privacy equivalence $P(\emptyset, \tau) \simeq P(\emptyset, \tau')$ does not hold.

References

[Aba01] M. Abadi. Private authentication. Manuscript, May 2001.

[AG99] M. Abadi and A. Gordon. A calculus for cryptographic protocols: the spi-calculus. *Information and Computation*, 143:1–70, 1999.

[AMP01] J. Andonieh, D. Malkhi, and E. Pavlov. A security failure of the Crowds method for anonymity. Manuscript, July 2001.

[BNP99] M. Boreale, R. De Nicola, and R. Pugliese. Proof techniques for cryptographic processes. In *Proc. LICS*, pages 157–166, 1999.

[CB95] D. Cooper and K. Birman. Preserving privacy in a network of mobile computers. In *Proc. IEEE Symposium on Security and Privacy*, pages 26–38, 1995.

[Cha81] D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–88, 1981.

[Cha88] D. Chaum. The Dining Cryptographers problem: unconditional sender and recipient untraceability. *J. Cryptology*, 1(1):65–75, 1988.

[LMMS99] P. Lincoln, J.C. Mitchell, M. Mitchell, and A. Scedrov. Probabilistic polynomial-time equivalence and security analysis. In *Proc. World Congress on Formal Methods*, volume 1708 of *LNCS*, pages 776–793. Springer-Verlag, 1999.

[PKS01] A. Pfitzmann, M. Köhntopp, and A. Shostack. Anonymity, unobservability, and pseudonymity - a proposal for terminology. Manuscript (available from A. Pfitzmann), June 2001.

[RR98] M. Reiter and A. Rubin. Crowds: Anonymity for web transactions. *ACM Transactions on Information and System Security*, 1(1):66–92, 1998.

[SGR97] P. Syverson, D. Goldschlag, and M. Reed. Anonymous connections and onion routing. In *Proc. IEEE Symposium on Security and Privacy*, pages 44–54, 1997.

[SMA95] D. Samfat, R. Molva, and N. Asokan. Anonymity and untraceability in mobile networks. In *Proc. MOBICOM*, pages 26–36, 1995.

[SS96] S. Schneider and A. Sidiroupoulos. CSP and anonymity. In *Proc. ESORICS*, volume 1146 of *LNCS*, pages 198–218. Springer-Verlag, 1996.

[SS98] P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical Report SRI-CSL-98-03, Computer Science Laboratory, SRI, 1998.

[SS99] P. Syverson and S. Stubblebine. Group principals and the formalization of anonymity. In *Proc. World Congress on Formal Methods*, volume 1708 of *LNCS*, pages 814–833. Springer-Verlag, 1999.