A semantics for BAN logic

Annette Bleeker*
CWI, Amsterdam
annette@cwi.nl

Lambert Meertens
CWI, Amsterdam, and
Department of Computing Science, Utrecht University, The Netherlands
lambert@cwi.nl

0 Introduction

In their ground-breaking paper [BAN89] Burrows, Abadi and Needham give a formalism, BAN logic, that can be used to reason about security properties of protocols. There are, however, several implicit or informally described assumptions. Moreover, the semantics gets very little attention, and is mentioned only in a short, informal description. As pointed out by Syverson [Syv91], having only an intuitive understanding of the semantics directly opposes the usual goal of having semantics at all: to see the logic as part of a system, and to be able to prove statements about a (more or less) existing model formally. Our aim was, therefore, to construct a sound semantics for BAN, as well as a notion of correctness that makes the additional requirements for both specification and protocol explicit.

Various authors have tried to give a semantics to BAN logic; some of them left the logic more or less intact, while others developed semantics for a new logic based on BAN logic. All these approaches are, just like BAN logic itself, restricted in their description of reality — the world can only be described through the (belief) eyes of the participants. However, to be able to judge cryptographic protocols, one cannot avoid looking beyond the individual beliefs of participants. Since all individual beliefs may be wrong, the outside world must be looked at separately. Therefore, we have not only looked for a precise semantics for BAN logic (and a proof of its soundness), but we have also chosen the semantics in such a way that it enables us to reason about knowledge (and, as a result, about the rightness of the participants' beliefs).

For our investigation we use an extension of BAN logic. Its language has, apart from the constructs taken from BAN, a few additional constructs, such as $P\ \mathsf{possesses}\ K$, used to express possession of a key — and the resulting ability to decrypt messages with that key — without necessarily believing that it belongs to a certain pair of principals; and $P\ \mathsf{rightly\_believes}\ \varphi$, which expresses that not only $P\ \mathsf{believes}\ \varphi$, but also φ itself holds.

We present the axioms of the logic in a general form: one can derive statements about the beliefs of principals, but also about the rightness of those beliefs (or of statements in general, independent of any beliefs). Defining a *rectify operation* that maps formulas of the form $P\ \mathsf{believes}\ \varphi$ to $P\ \mathsf{rightly\_believes}\ \varphi$, leaving other formulas intact, leads to a theorem that expresses that principals draw the right conclusions from their beliefs. In other words: if their initial beliefs are right, their conclusions will be right as well.

However, logical soundness does not yet establish that principals draw correct conclusions during a protocol run. We define, using operational semantics, what it means for a protocol to

*This research was supported by the Netherlands Organization for Scientific Research (NWO).

meet its specification. In order to prove that property of a protocol, we need certain restrictions on the protocol, depending on the assumptions. Besides, as it turns out, the assumptions need to be of a certain form as well, in order to secure monotonicity. Those requirements can be checked statically and do not exclude well-known examples of protocols.

1 The logic

In this section we introduce our logic with its language and axioms, while indicating the difference with the original BAN logic. After that, we prove that this logic is stronger than BAN.

1.1 The language

The sorts we distinguish are Principal, Key, Message and Formula. There are (further unspecified) universes of constants for the sorts Principal and Key. We view (logical) formulas as being a subsort of the sort of messages, since messages can also consist of nonces, timestamps or other constants, drawn from some further unspecified universe, as well as encrypted messages. So there is an implicit injection $\mathsf{M} :: \mathsf{Formula} \longrightarrow \mathsf{Message}$.

We use variables $A, B, P, Q, R, \ldots$ for principals, Greek letters $\varphi, \psi, \ldots$ for formulas, $M, X, Y, \ldots$ for messages in general, and $K, \ldots$ for keys.

For formulas, the language of the logic has the logical constant True, the logical operators $\wedge, \vee, \rightarrow$ and $\forall$, and the operator $=$ on the sort Message. Furthermore we have the following operators:

- $(\_,\_) :: \mathsf{Message} \times \mathsf{Message} \longrightarrow \mathsf{Message}$
(an associative, commutative and idempotent operator for message joining which is an extension of $\wedge$, the logical-and operator, so that the joining of two messages that happen to be formulas is interpreted as their conjunction[1])
- $\mathsf{believes} :: \mathsf{Principal} \times \mathsf{Formula} \longrightarrow \mathsf{Formula}$
- $\mathsf{once\_said} :: \mathsf{Principal} \times \mathsf{Message} \longrightarrow \mathsf{Formula}$
(for messages that have been uttered)
- $\mathsf{sees} :: \mathsf{Principal} \times \mathsf{Message} \longrightarrow \mathsf{Formula}$
(for messages that have been received)
- $\mathsf{possesses} :: \mathsf{Principal} \times \mathsf{Key} \longrightarrow \mathsf{Formula}$
(possession of information, in our case a key, does not imply any beliefs about validity or usage)
- $\_\ \mathsf{key\_of}\ (\_,\_) :: \mathsf{Key} \times \mathsf{Principal} \times \mathsf{Principal} \longrightarrow \mathsf{Formula}$
(symmetric in the last two arguments; intuitively, $K\ \mathsf{key\_of}\ (P,Q)$ means that K is a good key between P and Q)
- $\_(\_|\_) :: \mathsf{Key} \times \mathsf{Message} \times \mathsf{Principal} \longrightarrow \mathsf{Message}$
(for encryption; intuitively, $K(X|P)$ denotes X encrypted with K by P)

The "word" operators bind more tightly than the traditional logical operators, so that, e.g., $P\ \mathsf{believes}\ \varphi \wedge \psi$ must be interpreted as $(P\ \mathsf{believes}\ \varphi) \wedge \psi$.

Definition 1 *We define the operators* rightly_believes*,* controls *and* fresh *as follows:*

$$\mathsf{rightly\_believes} :: \mathsf{Principal} \times \mathsf{Formula} \longrightarrow \mathsf{Formula}$$

$$P\ \mathsf{rightly\_believes}\ \varphi := P\ \mathsf{believes}\ \varphi \wedge \varphi$$

[1]Using the injection M mentioned above, we could write: $(\mathsf{M}\varphi, \mathsf{M}\psi) = \mathsf{M}(\varphi \wedge \psi)$.

$$\mathsf{controls} :: \mathsf{Principal} \times \mathsf{Formula} \longrightarrow \mathsf{Formula}$$

$$P\,\mathsf{controls}\,\varphi := P\,\mathsf{believes}\,\varphi \;\rightarrow\; P\,\mathsf{rightly\_believes}\,\varphi$$

$$\mathsf{fresh} :: \mathsf{Message} \longrightarrow \mathsf{Formula}$$

$$\mathsf{fresh}\,X := (\forall P, \varphi :: P\,\mathsf{once\_said}\,(X, \varphi) \;\rightarrow\; P\,\mathsf{believes}\,\varphi)$$

(intuitively: $\mathsf{fresh}\,X$ *if* X *has not been uttered before the current protocol run)*

In BAN logic controls and fresh are primitive operators. We could, equivalently, have introduced these operators as primitives with their definitions turned into axioms.

1.2 Axiomatisation

For the axiomatisation we need to define the *contents* of a message as the collection of submessages when encryption is transparent:

Definition 2 *The function* $cts[\![\_]\!]$ *takes a message and delivers a set of messages:*

$$\begin{array}{ll} cts[\![(X,Y)]\!] & := \{(X,Y)\} \cup cts[\![X]\!] \cup cts[\![Y]\!] \\ cts[\![K(X|P)]\!] & := \{K(X|P)\} \cup cts[\![X]\!] \\ cts[\![X]\!] & := \{X\} \text{ (otherwise)} \end{array}$$

The axiomatisation includes the standard axioms for *equational logic* with Modus Ponens — which subsumes propositional logic — and the standard rules for universal quantification, where a formula $\varphi \leftrightarrow \psi$ is treated as shorthand for $(\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi)$.

Beneath, we introduce a collection of axioms for the specific operators of our logic, and mention the related axiom in the original BAN logic — if such an axiom exists. (Because of the presence of the Modus Ponens rule, we can replace inference rules $\frac{\varphi}{\psi}$ by axioms $\vdash \varphi \rightarrow \psi$.)

For message joining there are all axioms of the forms $\vdash (X,(Y,Z)) = ((X,Y),Z)$, $\vdash (X,Y) = (Y,X)$ and $\vdash (X,X) = X$, and for the key_of operator $\vdash K\,\mathsf{key\_of}\,(P,Q) = K\,\mathsf{key\_of}\,(Q,P)$. Equational logic allows us to apply theorems of the form $\vdash \varphi[x] \wedge x = y \;\rightarrow\; \varphi[y]$.

Furthermore, we have:

Rationality The rationality rule introduces a collection of axioms, one for every theorem of the logic:

$$\frac{\vdash \varphi}{\vdash P\,\mathsf{believes}\,\varphi}$$

This axiom, together with the next one, lifts the level of reasoning from beliefs of principals to general statements.

Believing Modus Ponens Modus Ponens under the believes operator:

$$\vdash P\,\mathsf{believes}\,(\varphi \rightarrow \psi) \;\rightarrow\; (P\,\mathsf{believes}\,\varphi \;\rightarrow\; P\,\mathsf{believes}\,\psi)$$

Saying parts of a joint message Uttering a joint message implies uttering each of the parts:

$$\vdash P\,\mathsf{once\_said}\,(X,Y) \;\rightarrow\; P\,\mathsf{once\_said}\,X$$

The related BAN logic axiom is:

$$\vdash_{BAN} P\,\mathsf{believes}\,Q\,\mathsf{once\_said}\,(X,Y) \;\rightarrow\; P\,\mathsf{believes}\,Q\,\mathsf{once\_said}\,X$$

Saying contents of an encrypted message Uttering an encrypted message, signed by yourself, while you believe that the key is good, implies uttering of the encrypted message:

$$\vdash P \,\mathsf{once\_said}\, K(X|P) \;\wedge\; P \,\mathsf{believes}\, K \,\mathsf{key\_of}\,(P,Q) \;\rightarrow\; P \,\mathsf{once\_said}\, X$$

There is no similar axiom in BAN logic.

Seeing parts of a joint message Seeing a joint message means seeing each part separately as well:

$$\vdash P \,\mathsf{sees}\,(X,Y) \;\rightarrow\; P \,\mathsf{sees}\, X$$

The BAN logic has this axiom (exactly so) as well.

Awareness Awareness of what one sees:

$$\vdash P \,\mathsf{sees}\, X \;\rightarrow\; P \,\mathsf{believes}\, P \,\mathsf{sees}\, X$$

There is no similar axiom in BAN logic.

Possessing keys of a seen key statement If one sees a key statement, one possesses the key that it mentioned:

$$\vdash P \,\mathsf{sees}\, K \,\mathsf{key\_of}\,(Q,R) \;\rightarrow\; P \,\mathsf{possesses}\, K$$

In BAN logic, the notion of possession does not exist.

Possessing believed keys One can only believe that a certain key is good if possessing the key:

$$\vdash P \,\mathsf{believes}\, K \,\mathsf{key\_of}\,(Q,R) \;\rightarrow\; P \,\mathsf{possesses}\, K$$

Decryption Seeing an encrypted message while having the key in possession, means seeing the message itself:

$$\vdash P \,\mathsf{possesses}\, K \wedge P \,\mathsf{sees}\, K(X|Q) \;\rightarrow\; P \,\mathsf{sees}\, X$$

The related BAN logic axiom is **decryption**:

$$\vdash_{BAN} P \,\mathsf{believes}\, K \,\mathsf{key\_of}\,(P,Q) \;\wedge\; P \,\mathsf{sees}\, K(X|Q) \;\rightarrow\; P \,\mathsf{sees}\, X$$

In BAN logic one can only decrypt with keys that are believed to be one's own key. With any other key in possession, decryption cannot be derived, since BAN logic does not have a notion of possession.

Good key ensures the utterer A collection of axioms, stating that if a key is good, the only ones that use it for encryption are the owners, so if somewhere, someone sees a message that contains a part encrypted with that key, that part must have been said by the key owner who encrypted it:
For all P, Q, R, X, Y, K such that $K(X|Q) \in cts[\![Y]\!]$:

$$\vdash K \,\mathsf{key\_of}\,(P,Q) \wedge R \,\mathsf{sees}\, Y \;\rightarrow\; Q \,\mathsf{once\_said}\, X$$

The related BAN logic axiom is the **message meaning rule**:

$$\vdash_{BAN} P \,\mathsf{believes}\, K \,\mathsf{key\_of}\,(P,Q) \;\wedge\; P \,\mathsf{sees}\, K(X|Q) \;\rightarrow\; P \,\mathsf{believes}\, Q \,\mathsf{once\_said}\, X$$

The BAN axioms that were not mentioned above are:

$$\vdash_{BAN} P \,\mathsf{believes}\, \varphi \;\wedge\; P \,\mathsf{believes}\, \psi \;\rightarrow\; P \,\mathsf{believes}\, (\varphi, \psi)$$

$$\vdash_{BAN} P \,\mathsf{believes}\, (\varphi, \psi) \;\rightarrow\; P \,\mathsf{believes}\, \varphi$$

$$\vdash_{BAN} P \,\mathsf{believes}\, K \,\mathsf{key\_of}\, (Q, R) \;\rightarrow\; P \,\mathsf{believes}\, K \,\mathsf{key\_of}\, (R, Q)$$

$$\vdash_{BAN} P \,\mathsf{believes}\,\mathsf{fresh}\, \varphi \;\wedge\; P \,\mathsf{believes}\, Q \,\mathsf{once\_said}\, \varphi \;\rightarrow\; P \,\mathsf{believes}\, Q \,\mathsf{believes}\, \varphi$$

$$\vdash_{BAN} P \,\mathsf{believes}\,\mathsf{fresh}\, X \;\rightarrow\; P \,\mathsf{believes}\,\mathsf{fresh}\, (X, Y)$$

$$\vdash_{BAN} P \,\mathsf{believes}\, Q \,\mathsf{controls}\, \varphi \;\wedge\; P \,\mathsf{believes}\, Q \,\mathsf{believes}\, \varphi \;\rightarrow\; P \,\mathsf{believes}\, \varphi$$

1.3 Comparison with BAN logic

In order to understand the relation between our logic and BAN logic, we show that every theorem of BAN logic is a theorem of our logic — in other words, that our logic is stronger. Because our logic stripped off the believes operator from most of the axioms, the following lemma will be useful in the proof.

Lemma 3 *If* $\vdash \varphi \rightarrow \psi$, *then* $\vdash P \,\mathsf{believes}\, \varphi \;\rightarrow\; P \,\mathsf{believes}\, \psi$.

Proof

$$\begin{array}{ll} & \vdash \varphi \rightarrow \psi \\ \Rightarrow & \qquad \{\text{Rationality}\} \\ & \vdash P \,\mathsf{believes}\, (\varphi \rightarrow \psi) \\ \Rightarrow & \qquad \{\text{Believing Modus Ponens, Modus Ponens}\} \\ & \vdash P \,\mathsf{believes}\, \varphi \;\rightarrow\; P \,\mathsf{believes}\, \psi \end{array}$$

□

From Believing Modus Ponens and the Rationality Rule we now obtain:

Lemma 4 $\vdash P \,\mathsf{believes}\, (\varphi \wedge \psi) \;\leftrightarrow\; P \,\mathsf{believes}\, \varphi \;\wedge\; P \,\mathsf{believes}\, \psi$

If part of a message is fresh, the whole of the message must be fresh as well:

Lemma 5 $\vdash \mathsf{fresh}\, X \;\rightarrow\; \mathsf{fresh}\, (X, Y)$

Note that the reverse does not hold, since a message can contain "old news" next to new data; the combination is fresh, but every element is not.

Theorem 6 *The logic as defined in this article, is stronger than BAN logic.*

Proof It suffices to prove that all axioms of the BAN logic are theorems of our logic.

$$\vdash_{BAN} P \,\mathsf{sees}\, (X, Y) \;\rightarrow\; P \,\mathsf{sees}\, X$$

is our axiom **Seeing parts of a joint message**, and therefore a theorem.

$$\vdash_{BAN} P \,\mathsf{believes}\, \varphi \;\wedge\; P \,\mathsf{believes}\, \psi \;\rightarrow\; P \,\mathsf{believes}\, (\varphi, \psi)$$

$$\vdash_{BAN} P \,\mathsf{believes}\, (\varphi, \psi) \;\rightarrow\; P \,\mathsf{believes}\, \varphi$$

both follow from Lemma 4 using the identification of the joining operator on formulas with the logical-and operator.

$$\vdash_{BAN} P \,\textsf{believes}\, K \,\textsf{key\_of}\, (Q, R) \;\rightarrow\; P \,\textsf{believes}\, K \,\textsf{key\_of}\, (R, Q)$$

follows from the symmetry of the $\textsf{key\_of}$ operator.

$$\vdash_{BAN} P \,\textsf{believes}\, Q \,\textsf{once\_said}\, (X, Y) \;\rightarrow\; P \,\textsf{believes}\, Q \,\textsf{once\_said}\, X$$

follows from our axiom **Saying parts of a joint message** and Lemma 3.

$$\vdash_{BAN} P \,\textsf{believes}\, K \,\textsf{key\_of}\, (P, Q) \;\wedge\; P \,\textsf{sees}\, K(X|Q) \;\rightarrow\; P \,\textsf{sees}\, X$$

follows from our axioms **Possessing believed keys** and **Decryption**.

$$\vdash_{BAN} P \,\textsf{believes}\, K \,\textsf{key\_of}\, (P, Q) \;\wedge\; P \,\textsf{sees}\, K(X|Q) \;\rightarrow\; P \,\textsf{believes}\, Q \,\textsf{once\_said}\, X$$

follows from our axiom **Awareness** and Lemmas 3 and 4.

$$\vdash_{BAN} P \,\textsf{believes}\,\textsf{fresh}\, \varphi \;\wedge\; P \,\textsf{believes}\, Q \,\textsf{once\_said}\, \varphi \;\rightarrow\; P \,\textsf{believes}\, Q \,\textsf{believes}\, \varphi$$

follows from the definition of $\textsf{fresh}$, the idempotence of $(\_, \_)$, and Lemmas 3 and 4.

$$\vdash_{BAN} P \,\textsf{believes}\,\textsf{fresh}\, X \;\rightarrow\; P \,\textsf{believes}\,\textsf{fresh}\, (X, Y)$$

follows from Lemmas 3 and 5.

$$\vdash_{BAN} P \,\textsf{believes}\, Q \,\textsf{controls}\, \varphi \;\wedge\; P \,\textsf{believes}\, Q \,\textsf{believes}\, \varphi \;\rightarrow\; P \,\textsf{believes}\, \varphi$$

follows from the definitions of $\textsf{controls}$ and $\textsf{rightly\_believes}$ and Lemmas 3 and 4.
□

2 Rectification of formulas

The rectify operation $\mathcal{R}[\![\_]\!]$ maps formulas to formulas. In particular, it maps formulas of the form $P \,\textsf{believes}\, \varphi$ to $P \,\textsf{rightly\_believes}\, \varphi$. It is defined as follows:

Definition 7

$$\begin{array}{lcl}
\mathcal{R}[\![P \,\textsf{believes}\, \varphi]\!] & := & P \,\textsf{rightly_believes}\, \varphi \\
\mathcal{R}[\![\varphi \wedge \psi]\!] & := & \mathcal{R}[\![\varphi]\!] \wedge \mathcal{R}[\![\psi]\!] \\
\mathcal{R}[\![\varphi \vee \psi]\!] & := & \mathcal{R}[\![\varphi]\!] \vee \mathcal{R}[\![\psi]\!] \\
\mathcal{R}[\![\varphi \rightarrow \psi]\!] & := & \mathcal{R}[\![\varphi]\!] \;\rightarrow\; \mathcal{R}[\![\psi]\!] \\
\mathcal{R}[\![\forall x :: \varphi]\!] & := & (\forall x :: \mathcal{R}[\![\varphi]\!]) \\
\mathcal{R}[\![\varphi]\!] & := & \varphi, \text{ other cases}
\end{array}$$

Note the limited recursion, which stops whenever a formula with a "word" operator is encountered. The operation is extended to a set-to-set mapping in the usual way.

Directly from the definition on sets it follows that:

Lemma 8 *If* $\mathcal{A}_1 \subseteq \mathcal{A}_2$, *then* $\mathcal{R}[\![\mathcal{A}_1]\!] \subseteq \mathcal{R}[\![\mathcal{A}_2]\!]$

Theorem 9 *If* $\mathcal{A} \vdash \mathcal{C}$ *then* $\mathcal{R}[\![\mathcal{A}]\!] \vdash \mathcal{R}[\![\mathcal{C}]\!]$.

Proof
We prove it with structural induction on the derivation.

Case trivial proof: A trivial proof is a derivation step of the form $\mathcal{A} \vdash \mathcal{C}$, where $\mathcal{C} \subseteq \mathcal{A}$.

$$
\begin{array}{ll}
 & \mathcal{R}[\![\mathcal{A}]\!] \text{ and } \mathcal{C} \subseteq \mathcal{A} \\
\Rightarrow & \qquad \{\text{Lemma 8}\} \\
 & \mathcal{R}[\![\mathcal{A}]\!] \text{ and } \mathcal{R}[\![\mathcal{C}]\!] \subseteq \mathcal{R}[\![\mathcal{A}]\!] \\
\Rightarrow & \qquad \{\text{trivial proof}\} \\
 & \mathcal{R}[\![\mathcal{C}]\!]
\end{array}
$$

Case deduction: Deduction is the introduction of implication by derivation:

If $\varphi \vdash \psi$, then $\vdash \varphi \rightarrow \psi$.

Using deduction, we may instantiate:

If $\mathcal{R}[\![\varphi]\!] \vdash \mathcal{R}[\![\psi]\!]$, then $\vdash \mathcal{R}[\![\varphi]\!] \rightarrow \mathcal{R}[\![\psi]\!]$.

From here we can derive, under the assumption $\mathcal{R}[\![\varphi]\!] \vdash \mathcal{R}[\![\psi]\!]$:

$$
\begin{array}{ll}
 & \mathsf{True} \\
\Rightarrow & \qquad \{\text{deduction, } \mathcal{R}[\![\varphi]\!] \vdash \mathcal{R}[\![\psi]\!]\} \\
 & \mathcal{R}[\![\varphi]\!] \rightarrow \mathcal{R}[\![\psi]\!] \\
\equiv & \qquad \{\text{definition rectify}\} \\
 & \mathcal{R}[\![\varphi \rightarrow \psi]\!]
\end{array}
$$

Cases axioms of predicate logic: As an example, we prove our claim for the introduction of $\wedge$. Other introductions and eliminations ($\vee$, $\rightarrow$, $\neg$, $\forall$) have analogous proofs.

$$
\begin{array}{ll}
 & \mathcal{R}[\![\varphi]\!] \text{ and } \mathcal{R}[\![\psi]\!] \\
\Rightarrow & \qquad \{\text{introduction } \wedge\} \\
 & \mathcal{R}[\![\varphi]\!] \wedge \mathcal{R}[\![\psi]\!] \\
\equiv & \qquad \{\text{definition rectify}\} \\
 & \mathcal{R}[\![\varphi \wedge \psi]\!]
\end{array}
$$

Case Rationality: Let $\vdash \varphi$. We prove $\vdash \mathcal{R}[\![P \text{ } \mathsf{believes} \text{ } \varphi]\!]$.

$$
\begin{array}{ll}
 & \mathsf{True} \\
\Rightarrow & \qquad \{\text{Rationality, } \vdash \varphi\} \\
 & P \text{ } \mathsf{believes} \text{ } \varphi \wedge \varphi \\
\equiv & \qquad \{\text{definition } \mathsf{rightly\_believes}\} \\
 & P \text{ } \mathsf{rightly\_believes} \text{ } \varphi \\
\equiv & \qquad \{\text{definition rectify}\} \\
 & \mathcal{R}[\![P \text{ } \mathsf{believes} \text{ } \varphi]\!]
\end{array}
$$

Case axiom: For every axiom $\vdash \varphi \rightarrow \psi$, we will prove $\vdash \mathcal{R}[\![\varphi \rightarrow \psi]\!]$, by deriving $\mathcal{R}[\![\varphi]\!] \vdash \mathcal{R}[\![\psi]\!]$.

case Believing Modus Ponens: We prove it for the equivalent form $\vdash (P \,\mathsf{believes}\,(\varphi \;\rightarrow\; \psi) \wedge P \,\mathsf{believes}\,\varphi) \;\rightarrow\; P \,\mathsf{believes}\,\psi$

$$
\begin{array}{ll}
 & \mathcal{R}[\![P \,\mathsf{believes}\,(\varphi \rightarrow \psi) \wedge P \,\mathsf{believes}\,\varphi]\!] \\
\equiv & \qquad \{\text{definition rectify}\} \\
 & P \,\mathsf{rightly\_believes}\,(\varphi \rightarrow \psi) \wedge P \,\mathsf{rightly\_believes}\,\varphi \\
\equiv & \qquad \{\text{definition } \mathsf{rightly\_believes}\} \\
 & P \,\mathsf{believes}\,(\varphi \rightarrow \psi) \wedge (\varphi \rightarrow \psi) \wedge (P \,\mathsf{believes}\,\varphi) \wedge \varphi \\
\Rightarrow & \qquad \{\text{Believing Modus Ponens, Modus Ponens}\} \\
 & P \,\mathsf{believes}\,\psi \wedge \psi \\
\equiv & \qquad \{\text{definition } \mathsf{rightly\_believes}\} \\
 & P \,\mathsf{rightly\_believes}\,\psi \\
\equiv & \qquad \{\text{definition rectify}\} \\
 & \mathcal{R}[\![P \,\mathsf{believes}\,\psi]\!]
\end{array}
$$

case Saying contents of an encrypted message:

$$
\begin{array}{ll}
 & \mathcal{R}[\![P \,\mathsf{once\_said}\, K(X|P) \wedge P \,\mathsf{believes}\, K \,\mathsf{key\_of}\,(P,Q)]\!] \\
\equiv & \qquad \{\text{definition rectify}\} \\
 & P \,\mathsf{once\_said}\, K(X|P) \wedge P \,\mathsf{rightly\_believes}\, K \,\mathsf{key\_of}\,(P,Q) \\
\Rightarrow & \qquad \{\text{definition } \mathsf{rightly\_believes}\} \\
 & P \,\mathsf{once\_said}\, K(X|P) \wedge P \,\mathsf{believes}\, K \,\mathsf{key\_of}\,(P,Q) \\
\Rightarrow & \qquad \{\text{Saying contents of an encrypted message}\} \\
 & P \,\mathsf{once\_said}\, X \\
\equiv & \qquad \{\text{definition rectify}\} \\
 & \mathcal{R}[\![P \,\mathsf{once\_said}\, X]\!]
\end{array}
$$

case Awareness:

$$
\begin{array}{ll}
 & \mathcal{R}[\![P \,\mathsf{sees}\, X]\!] \\
\equiv & \qquad \{\text{definition rectify}\} \\
 & P \,\mathsf{sees}\, X \\
\Rightarrow & \qquad \{\text{Awareness, introduction } \wedge\} \\
 & P \,\mathsf{believes}\, P \,\mathsf{sees}\, X \wedge P \,\mathsf{sees}\, X \\
\equiv & \qquad \{\text{definition } \mathsf{rightly\_believes}\} \\
 & P \,\mathsf{rightly\_believes}\, P \,\mathsf{sees}\, X \\
\equiv & \qquad \{\text{definition rectify}\} \\
 & \mathcal{R}[\![P \,\mathsf{believes}\, P \,\mathsf{sees}\, X]\!]
\end{array}
$$

case Possessing believed keys:

$$
\begin{array}{ll}
 & \mathcal{R}[\![P \,\mathsf{believes}\, K \,\mathsf{key\_of}\,(Q,R)]\!] \\
\equiv & \qquad \{\text{definition rectify}\} \\
 & P \,\mathsf{rightly\_believes}\, K \,\mathsf{key\_of}\,(Q,R)
\end{array}
$$

$\Rightarrow \qquad \{\text{definition } \mathsf{rightly\_believes}\}$

$P \mathsf{\ believes\ } K \mathsf{\ key\_of\ } (Q, R)$

$\Rightarrow \qquad \{\text{Possessing believed keys}\}$

$P \mathsf{\ possesses\ } K$

$\equiv \qquad \{\text{definition rectify}\}$

$\mathcal{R}[\![P \mathsf{\ possesses\ } K]\!]$

The remaining axioms do not contain a believe operator, so they do not change under the rectify function.
□

3 The model

We view the environment as a system consisting of a finite collection of *principals*. We define for a principal P a *local state* as a tuple $(\mathcal{B}_P, \mathcal{O}_P, \mathcal{S}_P, \mathcal{K}_P)$, with the intuitive interpretation:

- $\mathcal{B}_P$, the set of formulas that P currently believes;
- $\mathcal{O}_P$, the set of (sub-)messages P once said;
- $\mathcal{S}_P$, the set of messages that P has seen so far;
- $\mathcal{K}_P$, the set of keys P possesses.

It is *closed* if it satisfies the following (mutually defined) closure properties, each of which corresponds directly to an axiom:

(i). (Rationality) Principals believe every theorem of the logic:

$$(\vdash \varphi) \Rightarrow \varphi \in \mathcal{B}_P$$

(ii). (Believing Modus Ponens) Principals apply Modus Ponens in their beliefs:

$$(\varphi \rightarrow \psi) \in \mathcal{B}_P \wedge \varphi \in \mathcal{B}_P \Rightarrow \psi \in \mathcal{B}_P$$

(iii). (Saying parts of a joint message) If a principal said a combination of messages at a certain time, then that principal said each of the messages as well:

$$(X, Y) \in \mathcal{O}_P \Rightarrow X \in \mathcal{O}_P \wedge Y \in \mathcal{O}_P$$

The reverse does not hold, since the presence of a joint message in $\mathcal{O}_P$ implies that both components were uttered (as a joint message) at the same time;

(iv). (Saying contents of an encrypted message) If a principal said an encrypted message and believes the key is good, then that principal said the contents of the encrypted message as well:

$$K(X|P) \in \mathcal{O}_P \wedge K \mathsf{\ key\_of\ } (P, Q) \in \mathcal{B}_P \Rightarrow X \in \mathcal{O}_P$$

(v). (Seeing parts of a joint message) If a principal sees a joint message, that principal sees each of the messages as well:

$$(X, Y) \in \mathcal{S}_P \Rightarrow X \in \mathcal{S}_P \wedge Y \in \mathcal{S}_P$$

(Note that the reverse does not hold, since a joint message implies utterance of its components *at the same time*, i.e., within the same message.)

(vi). (Awareness) If a principal sees a message, then that principal also believes he sees it:

$$X \in \mathcal{S}_P \Rightarrow (P\,\mathsf{sees}\,X) \in \mathcal{B}_P$$

(vii). (Possessing keys of a seen key statement) If a principal sees a key statement, then that principal possesses the key mentioned:

$$K\,\mathsf{key\_of}\,(Q,R) \in \mathcal{S}_P \Rightarrow K \in \mathcal{K}_P$$

(viii). (Possessing believed keys) If a principal believes that a certain key is a good key statement, then that principal must possess that key as well:

$$K\,\mathsf{key\_of}\,(Q,R) \in \mathcal{B}_P \Rightarrow K \in \mathcal{K}_P$$

(ix). (Decryption) If a principal P possesses a key K, and if P sees a message X labeled with Q and encrypted with K, then P also sees X itself:

$$K \in \mathcal{K}_P \wedge K(X|Q) \in \mathcal{S}_P \Rightarrow X \in \mathcal{S}_P$$

(Note that there is no closure property corresponding to the axiom **Good key ensures the utterer**.)

The *closure* of a local state s is the least closed local state s' such that $s \subseteq s'$, where the ordering is obtained by component-wise lifting of the set ordering. Note that taking the closure only adds elements to the sets involved, and leaves an already closed local state unchanged.

A *global state* is a mapping from principals to local states (for each principal in the environment). Global states are ordered by lifting the ordering on local states. The closure of a global state s, denoted by $clo(s)$, is defined in the obvious way. The unqualified term "*state*" will, from now on, mean a *closed global* state.

We consistently use the convention that for a state denoted by the variable s its local state for principal P is denoted by the tuple $(\mathcal{B}_P, \mathcal{O}_P, \mathcal{S}_P, \mathcal{K}_P)$, and likewise that for a state s' its local state for principal P is denoted by the tuple $(\mathcal{B}'_P, \mathcal{O}'_P, \mathcal{S}'_P, \mathcal{K}'_P)$

Lemma 10 *The function clo is monotonic, augmenting and idempotent:*

$$x \subseteq y \;\Rightarrow\; clo(x) \subseteq clo(y)$$

$$x \subseteq clo(x)$$

$$clo(clo(x)) = clo(x)$$

4 Semantics of formulas

We define the relation $\models$ between states and formulas (where $s \models \varphi$ means: in state s formula φ holds) inductively on the structure of formulas as the least relation satisfying:

$$
\begin{array}{lll}
s \models \mathsf{True}; & & \\
s \models \varphi \wedge \psi & \text{if} & s \models \varphi \text{ and } s \models \psi; \\
s \models (\varphi, \psi) & \text{if} & s \models \varphi \wedge \psi; \\
s \models \varphi \rightarrow \psi & \text{if} & s \models \varphi \text{ implies } s \models \psi; \\
s \models (\forall x :: \varphi) & \text{if} & s \models \varphi[x \leftarrow u] \text{ holds for all terms } u \text{ of the appropriate kind} \\
 & & \text{not containing unbound variables;} \\
s \models P \,\mathsf{believes}\, \varphi & \text{if} & \varphi \in \mathcal{B}_P; \\
s \models P \,\mathsf{once\_said}\, X & \text{if} & X \in \mathcal{O}_P; \\
s \models P \,\mathsf{sees}\, X & \text{if} & X \in \mathcal{S}_P; \\
s \models P \,\mathsf{possesses}\, K & \text{if} & K \in \mathcal{K}_P; \\
s \models K \,\mathsf{key\_of}\, (P, Q) & \text{if} & \text{for all } S \in \{P, Q\}, \text{ all } R, \text{ and all } X, Y \text{ such that} \\
 & & \qquad K(X|S) \in cts[\![Y]\!] : \\
 & & s \models R \,\mathsf{sees}\, Y \text{ implies } s \models S \,\mathsf{once\_said}\, X.
\end{array}
$$

The notation $\varphi[x \leftarrow u]$ above means: φ with u substituted for the free occurrences of x. If $\psi = (\forall \varphi :: \varphi)$, the whole formula ψ may be substituted for φ, and since $\varphi[\varphi \leftarrow \psi] = \psi$, the "recursion" in the definition of $\models$ is unbounded, whence the appeal above to "least relation".

Note that the relation $\models$ is not monotonic in its left argument, i.e., when $s \subseteq s'$, $s \models \mathcal{A}$ does in general not imply $s' \models \mathcal{A}$.

5 Soundness

Theorem 11 (Soundness) *The logic as described in this article is sound with respect to the newly defined semantics.*

Corollary 12 *BAN logic is sound.*

Proof of Theorem 11 For each axiom $\vdash \varphi \rightarrow \psi$, as summed up in section 1.2, we prove $s \models \varphi \Rightarrow s \models \psi$.

Rationality does not have the form $\vdash \varphi \rightarrow \psi$; instead, we have to show that for each theorem φ (and any P in the environment) the formula $P \,\mathsf{believes}\, \varphi$ is a tautology.

$$
\begin{array}{ll}
 & \vdash \varphi \\
\Rightarrow & \qquad \{\text{closure property (i) }\} \\
 & \text{for all states } s : \ \varphi \in \mathcal{B}_P \\
\equiv & \qquad \{\text{semantics of } \mathsf{believes}\} \\
 & \text{for all states } s : \ s \models P \,\mathsf{believes}\, \varphi
\end{array}
$$

Believing Modus Ponens We prove the soundness of the following, equivalent variant of the axiom: $\vdash P \,\mathsf{believes}\, (\varphi \rightarrow \psi) \wedge P \,\mathsf{believes}\, \varphi \quad \rightarrow \quad P \,\mathsf{believes}\, \psi$.

$$
\begin{array}{ll}
 & s \models P \,\mathsf{believes}\, (\varphi \rightarrow \psi) \ \wedge \ P \,\mathsf{believes}\, \varphi \\
\equiv & \qquad \{\text{semantics of } \mathsf{believes}\} \\
 & (\varphi \rightarrow \psi) \in \mathcal{B}_P \text{ and } \varphi \in \mathcal{B}_P \\
\Rightarrow & \qquad \{\text{closure property (ii)}\} \\
 & \psi \in \mathcal{B}_P
\end{array}
$$

$$
\begin{aligned}
\equiv\quad & \qquad \{\text{semantics of } \mathsf{believes}\} \\
& s \models P \ \mathsf{believes}\ \psi
\end{aligned}
$$

Saying parts of a joint message

$$
\begin{aligned}
& s \models P \ \mathsf{once\_said}\ (X, Y) \\
\equiv\quad & \qquad \{\text{semantics of } \mathsf{once\_said}\} \\
& (X, Y) \in \mathcal{O}_P \\
\Rightarrow\quad & \qquad \{\text{closure property (iii)}\} \\
& X \in \mathcal{O}_P \\
\equiv\quad & \qquad \{\text{semantics of } \mathsf{once\_said}\} \\
& s \models P \ \mathsf{once\_said}\ X
\end{aligned}
$$

Saying contents of an encrypted message

$$
\begin{aligned}
& s \models P \ \mathsf{once\_said}\ K(X|P) \ \wedge\ P \ \mathsf{believes}\ K \ \mathsf{key\_of}\ (P, Q) \\
\equiv\quad & \qquad \{\text{semantics of } \mathsf{once\_said} \text{ and } \mathsf{believes}\} \\
& K(X|P) \in \mathcal{O}_P \text{ and } K \ \mathsf{key\_of}\ (P, Q) \in \mathcal{B}_P \\
\Rightarrow\quad & \qquad \{\text{closure property (iv)}\} \\
& X \in \mathcal{O}_P \\
\equiv\quad & \qquad \{\text{semantics of } \mathsf{once\_said}\} \\
& s \models P \ \mathsf{once\_said}\ X
\end{aligned}
$$

Seeing parts of a joint message

$$
\begin{aligned}
& s \models P \ \mathsf{sees}\ (X, Y) \\
\equiv\quad & \qquad \{\text{semantics of } \mathsf{sees}\} \\
& (X, Y) \in \mathcal{S}_P \\
\Rightarrow\quad & \qquad \{\text{closure property (v)}\} \\
& X \in \mathcal{S}_P \\
\equiv\quad & \qquad \{\text{semantics of } \mathsf{sees}\} \\
& s \models P \ \mathsf{sees}\ X
\end{aligned}
$$

Awareness

$$
\begin{aligned}
& s \models P \ \mathsf{sees}\ X \\
\equiv\quad & \qquad \{\text{semantics of } \mathsf{sees}\} \\
& X \in \mathcal{S}_P \\
\Rightarrow\quad & \qquad \{\text{closure property (vi)}\} \\
& P \ \mathsf{sees}\ X \in \mathcal{B}_P \\
\equiv\quad & \qquad \{\text{semantics of } \mathsf{sees}\} \\
& s \models P \ \mathsf{believes}\ P \ \mathsf{sees}\ X
\end{aligned}
$$

Possessing keys of a seen key statement

$$
\begin{array}{ll}
 & s \models P\ \mathsf{sees}\ K\ \mathsf{key\_of}\ (Q,R) \\
\equiv & \qquad \{\text{semantics of } \mathsf{sees}\} \\
 & K\ \mathsf{key\_of}\ (Q,R) \in \mathcal{S}_P \\
\Rightarrow & \qquad \{\text{closure property (vii)}\} \\
 & K \in \mathcal{K}_P \\
\equiv & \qquad \{\text{semantics of } \mathsf{possesses}\} \\
 & s \models P\ \mathsf{possesses}\ K
\end{array}
$$

Possessing believed keys

$$
\begin{array}{ll}
 & s \models P\ \mathsf{believes}\ K\ \mathsf{key\_of}\ (Q,R) \\
\equiv & \qquad \{\text{semantics of } \mathsf{believes}\} \\
 & K\ \mathsf{key\_of}\ (Q,R) \in \mathcal{B}_P \\
\Rightarrow & \qquad \{\text{closure property (viii)}\} \\
 & K \in \mathcal{K}_P \\
\equiv & \qquad \{\text{semantics of } \mathsf{possesses}\} \\
 & s \models P\ \mathsf{possesses}\ K
\end{array}
$$

Decryption

$$
\begin{array}{ll}
 & s \models P\ \mathsf{possesses}\ K \wedge P\ \mathsf{sees}\ K(X|Q) \\
\equiv & \qquad \{\text{semantics of } \mathsf{possesses} \text{ and } \mathsf{sees}\} \\
 & K \in \mathcal{K}_P \wedge K(X|Q) \in \mathcal{S}_P \\
\Rightarrow & \qquad \{\text{closure property (ix)}\} \\
 & X \in \mathcal{S}_P \\
\equiv & \qquad \{\text{semantics of } \mathsf{sees}\} \\
 & s \models P\ \mathsf{sees}\ X
\end{array}
$$

Good key ensures the utterer We prove the soundness of this equivalent variant of the axiom: $\vdash K\ \mathsf{key\_of}\ (P,Q) \ \rightarrow\ (R\ \mathsf{sees}\ Y \ \rightarrow\ Q\ \mathsf{once\_said}\ X)$, where we may use the assumption that $K(X|Q) \in cts[\![Y]\!]$:

$$
\begin{array}{ll}
 & s \models K\ \mathsf{key\_of}\ (P,Q) \\
\Rightarrow & \qquad \{\text{semantics of } \mathsf{key\_of}\} \\
 & \text{for all } S \in \{P,Q\}, \text{ all } R, \text{ and all } X, Y \text{ such that} K(X|S) \in cts[\![Y]\!] : \\
 & s \models R\ \mathsf{sees}\ Y \text{ implies } s \models S\ \mathsf{once\_said}\ X \\
\Rightarrow & \qquad \{\text{instantiate } S \leftarrow Q,\ R \leftarrow R,\ X \leftarrow X,\ Y \leftarrow Y\} \\
 & \ \text{if } K(X|Q) \in cts[\![Y]\!],\ s \models R\ \mathsf{sees}\ Y \text{ implies } s \models Q\ \mathsf{once\_said}\ X \\
\Rightarrow & \qquad \{\text{discharge the assumption}\} \\
 & s \models R\ \mathsf{sees}\ Y \text{ implies } s \models Q\ \mathsf{once\_said}\ X \\
\equiv & \qquad \{\text{semantics of } \rightarrow\} \\
 & s \models R\ \mathsf{sees}\ Y \ \rightarrow\ Q\ \mathsf{once\_said}\ X
\end{array}
$$

□

6 Protocols

In the previous section we have proven our logic sound, so every theorem of the logic is a tautology in the model. This, however, does not by itself establish in any way the suitability of the logic as a tool for proving protocols correct. In fact, protocols were not referred to at all.

Definition 13 *A* protocol *is a finite sequence of* actions*, where each protocol action has the form $P \to Q : X$, signifying the sending of a message X from P to Q.*

The empty protocol is denoted by $\emptyset$, and the sequencing operator "$;$" is used to denote concatenation of protocols.

The participants *of a protocol are the principals mentioned in its actions.*

We model actions as transitions from states to states. If in state s the action $P \to Q : X$ is performed, only the local states for P and Q will be changed. The new state s' is the least (closed global) state such that $s \subseteq s'$, $X \in \mathcal{O}'_P$ and $X \in \mathcal{S}'_Q$. From the closure properties we know that the local states for other principals stay unchanged. Formally:

Definition 14 *We define the transition function tra, mapping a state and an action to a new state, by*

$$tra(s, (P \to Q : X)) \;:=\; clo(s[\mathcal{O}_P \leftarrow \mathcal{O}_P \cup \{X\},\, \mathcal{S}_Q \leftarrow \mathcal{S}_Q \cup \{X\}])$$

that is, for the action of sending a message X from P to Q, X is added to both $\mathcal{O}_P$ and $\mathcal{S}_Q$, and then the closure is taken (to make it a well-formed state).

We define the extension of tra to protocols by:

$$\begin{array}{lll} tra(s, \emptyset) & := & s \\ tra(s, \mathcal{P}_1; \mathcal{P}_2) & := & tra(tra(s, \mathcal{P}_1), \mathcal{P}_2) \end{array}$$

Using Lemma 10 it follows immediately from the definition of *tra* that the transition function is augmenting for a fixed protocol:

Lemma 15 $s \subseteq tra(s, \mathcal{P})$

A *specification* of a protocol is given by two sets of formulas, $\mathcal{A}$ (the *assumptions*) and $\mathcal{C}$ (the *conclusions*). Protocol $\mathcal{P}$ *satisfies* such a specification (notation: $\{\mathcal{A}\}\,\mathcal{P}\,\{\mathcal{C}\}$) when "$\mathcal{A}$ holds initially" guarantees that "$\mathcal{C}$ holds finally", i.e., after running $\mathcal{P}$. We extend the relation $\models$ between states and formulas to a relation between states and *sets* of formulas, as follows:

$$s \models \mathcal{F} \quad := \quad \text{for all } \varphi \in \mathcal{F} : \; s \models \varphi$$

Using this, the semantics of the *specification triple* $\{\mathcal{A}\}\,\mathcal{P}\,\{\mathcal{C}\}$ is now defined as follows:

$$\models \{\mathcal{A}\}\,\mathcal{P}\,\{\mathcal{C}\} \quad \text{iff} \quad \text{for all states } s : \; s \models \mathcal{A} \text{ implies } tra(s, \mathcal{P}) \models \mathcal{C}.$$

Lemma 16 *The specifications may be composed: If $\models \{\mathcal{A}\}\,\mathcal{P}_1\,\{\mathcal{B}\}$ and $\models \{\mathcal{B}\}\,\mathcal{P}_2\,\{\mathcal{C}\}$, then $\models \{\mathcal{A}\}\,\mathcal{P}_1; \mathcal{P}_2\,\{\mathcal{C}\}$*

Proof

$$\begin{array}{ll} & \models \{\mathcal{A}\}\,\mathcal{P}_1\,\{\mathcal{B}\} \text{ and } \models \{\mathcal{B}\}\,\mathcal{P}_2\,\{\mathcal{C}\} \\ \equiv & \qquad \{\text{semantics of specification triples}\} \\ & \text{for all states } s : \; s \models \mathcal{A} \text{ implies } tra(s, \mathcal{P}_1) \models \mathcal{B} \text{ and} \\ & \text{for all states } s' : \; s' \models \mathcal{B} \text{ implies } tra(s', \mathcal{P}_2) \models \mathcal{C} \\ \Rightarrow & \qquad \{\text{instantiate } s' \leftarrow tra(s, \mathcal{P}_1)\} \\ & \text{for all states } s : \; s \models \mathcal{A} \text{ implies } tra(s, \mathcal{P}_1) \models \mathcal{B} \text{ and} \\ & tra(s, \mathcal{P}_1) \models \mathcal{B} \text{ implies } tra(tra(s, \mathcal{P}_1), \mathcal{P}_2) \models \mathcal{C} \end{array}$$

$$
\begin{array}{ll}
\Rightarrow & \quad \{\text{transitivity of implication}\} \\
& \text{for all states } s: \ s \models \mathcal{A} \text{ implies } tra(tra(s,\mathcal{P}_1),\mathcal{P}_2) \models \mathcal{C} \\
\equiv & \quad \{\text{definition of } tra\} \\
& \text{for all states } s: \ s \models \mathcal{A} \text{ implies } tra(s,\mathcal{P}_1;\mathcal{P}_2) \models \mathcal{C} \\
\equiv & \quad \{\text{semantics of specification triples}\} \\
& \models \{\mathcal{A}\}\,\mathcal{P}_1;\mathcal{P}_2\,\{\mathcal{C}\}
\end{array}
$$

□

Similarly, one can prove:

Lemma 17 *If* $\models \{\mathcal{A}\}\,\mathcal{P}\,\{\mathcal{B}\}$ *and* $\models \{\mathcal{A}'\}\,\mathcal{P}\,\{\mathcal{B}'\}$, *then* $\models \{\mathcal{A}\cup\mathcal{A}'\}\,\mathcal{P}\,\{\mathcal{B}\cup\mathcal{B}'\}$

7 Correctness of protocols

Our aim now is to prove that if $\mathcal{C}$ can be derived in our logic from $\mathcal{A}$ together with (some yet-to-be-defined logic translation of) $\mathcal{P}$, then $\models \{\mathcal{A}\}\,\mathcal{P}\,\{\mathcal{C}\}$ holds. From that we can conclude a rectified version, so that we know that participants in protocols draw correct conclusions.

As we have seen, the sending of a message is modelled by adding the message to the sender's $\mathcal{O}$ set of messages once-said, and to the receiver's $\mathcal{S}$ set of messages seen, and then to apply the closure as mentioned in section 3. We now define the logical equivalent of this:

Definition 18 *For an action $P \rightarrow Q : X$, i.e., the sending of a message X from P to Q, we define its* logic translation $\mathbf{T}(P \rightarrow Q : X)$ *as a (singleton) set of formulas:*

$$\mathbf{T}(P \rightarrow Q : X) \quad := \quad \{P\,\mathsf{once\_said}\,X \ \wedge \ Q\,\mathsf{sees}\,X\}$$

The extension to protocols is recursively defined:

$$
\begin{array}{lcl}
\mathbf{T}(\emptyset) & := & \emptyset \\
\mathbf{T}(\mathcal{P}_1;\mathcal{P}_2) & := & \mathbf{T}(\mathcal{P}_1)\cup\mathbf{T}(\mathcal{P}_2)
\end{array}
$$

Lemma 19 *For all actions a:* $\models \{\emptyset\}\,a\,\{\mathbf{T}(a)\}$

Proof Let a be $P \rightarrow Q : X$.

$$
\begin{array}{ll}
& \models \{\emptyset\}\,P \rightarrow Q : X\,\{\mathbf{T}(P \rightarrow Q : X)\} \\
\equiv & \quad \{\text{semantics of specification triples}\} \\
& \text{for all } s: \ s \models \emptyset \text{ implies } tra(s, P \rightarrow Q : X) \models \mathbf{T}(P \rightarrow Q : X) \\
\equiv & \quad \{\text{definition of } \mathbf{T}\} \\
& \text{for all } s: \ s \models \emptyset \text{ implies } tra(s, P \rightarrow Q : X) \models \{P\,\mathsf{once\_said}\,X \wedge Q\,\mathsf{sees}\,X\} \\
\equiv & \quad \{\text{definition of } \models \text{ on sets of formulas}\} \\
& \text{for all } s: \ tra(s, P \rightarrow Q : X) \models P\,\mathsf{once\_said}\,X \wedge Q\,\mathsf{sees}\,X \\
\equiv & \quad \{\text{definition of } tra\} \\
& \text{for all } s: \ clo(s[\mathcal{O}_P \leftarrow \mathcal{O}_P \cup \{X\}, \mathcal{S}_Q \leftarrow \mathcal{S}_Q \cup \{X\}]) \models P\,\mathsf{once\_said}\,X \wedge Q\,\mathsf{sees}\,X
\end{array}
$$

Now let s be any state, and put $s' = clo(s[\mathcal{O}_P \leftarrow \mathcal{O}_P \cup \{X\}, \mathcal{S}_Q \leftarrow \mathcal{S}_Q \cup \{X\}])$. We have

$$
\begin{array}{ll}
& X \in \mathcal{O}_P \cup \{X\} \wedge X \in \mathcal{S} \cup \{X\} \\
\Rightarrow & \quad \{clo \text{ is augmenting}\} \\
& X \in \mathcal{O}'_P \wedge X \in \mathcal{S}'_Q \\
\equiv & \quad \{\text{semantics of } \mathsf{once\_said} \text{ and } \mathsf{sees}\} \\
& s' \models P\,\mathsf{once\_said}\,X \wedge Q\,\mathsf{sees}\,X
\end{array}
$$

$\square$

Definition 20 *For a collection of predicates $\mathcal{A}$ and a protocol step $a = P \to Q : X$, the predicate $\mathcal{A}$ allows a is defined recursively with respect to the structure of the message X:*

$$\begin{array}{lcl}
\mathcal{A}\ \mathit{allows}\ P \to Q : K(X|P) & := & \textit{for some } S:\ \mathcal{A} \vdash P\ \textsf{believes}\ K\ \textsf{key_of}\ (P,S) \\
 & & \textit{and } \mathcal{A}\ \mathit{allows}\ P \to S : X \\
\mathcal{A}\ \mathit{allows}\ P \to Q : K(X|R), R \neq P & := & \mathcal{A} \vdash P\ \textsf{sees}\ K(X|R) \\
\mathcal{A}\ \mathit{allows}\ P \to Q : (X,Y) & := & \mathcal{A}\ \mathit{allows}\ P \to Q : X \text{ and } \mathcal{A}\ \mathit{allows}\ P \to Q : Y \\
\mathcal{A}\ \mathit{allows}\ P \to Q : \varphi & := & \mathcal{A} \vdash P\ \textsf{believes}\ \varphi \\
\mathcal{A}\ \mathit{allows}\ P \to Q : X & := & \textit{true (all other cases)}
\end{array}$$

Longer protocols are allowed if each of the steps is allowed in the respective states:

$$\begin{array}{lcl}
\mathcal{A}\ \mathit{allows}\ \emptyset & := & \mathit{true} \\
\mathcal{A}\ \mathit{allows}\ (a;\mathcal{P}) & := & \mathcal{A}\ \mathit{allows}\ a \text{ and } (\mathcal{A} \cup \mathbf{T}(a))\ \mathit{allows}\ \mathcal{P}
\end{array}$$

Lemma 21 *If $\mathcal{A}$ allows a, then $\mathcal{R}[\![\mathcal{A}]\!]$ allows a.*

Proof We prove $\mathcal{R}[\![\mathcal{A}]\!]\ \mathit{allows}\ P \to Q : X$ from $\mathcal{A}\ \mathit{allows}\ P \to Q : X$ with induction on X.

Case $K(X|P)$:

$$\begin{array}{ll}
 & \mathcal{A}\ \mathit{allows}\ P \to Q : K(X|P) \\
\equiv & \qquad \{\text{definition } \mathit{allows}\} \\
 & \text{for some } S:\ \mathcal{A} \vdash P\ \textsf{believes}\ K\ \textsf{key_of}\ (P,S) \\
 & \text{and } \mathcal{A}\ \mathit{allows}\ P \to S : X \\
\Rightarrow & \qquad \{\text{Theorem 9}\} \\
 & \text{for some } S:\ \mathcal{R}[\![\mathcal{A}]\!] \vdash \mathcal{R}[\![P\ \textsf{believes}\ K\ \textsf{key_of}\ (P,S)]\!] \\
 & \text{and } \mathcal{A}\ \mathit{allows}\ P \to S : X \\
\equiv & \qquad \{\text{definition rectify}\} \\
 & \text{for some } S:\ \mathcal{R}[\![\mathcal{A}]\!] \vdash P\ \textsf{rightly_believes}\ K\ \textsf{key_of}\ (P,S) \\
 & \text{and } \mathcal{A}\ \mathit{allows}\ P \to S : X \\
\Rightarrow & \qquad \{\text{definition } \textsf{rightly_believes}\} \\
 & \text{for some } S:\ \mathcal{R}[\![\mathcal{A}]\!] \vdash P\ \textsf{believes}\ K\ \textsf{key_of}\ (P,S) \\
 & \text{and } \mathcal{A}\ \mathit{allows}\ P \to S : X \\
\Rightarrow & \qquad \{\text{Induction Hypothesis}\} \\
 & \text{for some } S:\ \mathcal{R}[\![\mathcal{A}]\!] \vdash P\ \textsf{believes}\ K\ \textsf{key_of}\ (P,S) \\
 & \text{and } \mathcal{R}[\![\mathcal{A}]\!]\ \mathit{allows}\ P \to S : X \\
\equiv & \qquad \{\text{definition } \mathit{allows}\} \\
 & \mathcal{R}[\![\mathcal{A}]\!]\ \mathit{allows}\ P \to Q : K(X|P)
\end{array}$$

Case $K(X|R), R \neq P$:

$$\begin{array}{ll}
 & \mathcal{A}\ \mathit{allows}\ P \to Q : K(X|R) \\
\equiv & \qquad \{\text{definition } \mathit{allows}\} \\
 & \mathcal{A} \vdash P\ \textsf{sees}\ K(X|R) \\
\Rightarrow & \qquad \{\text{Theorem 9}\} \\
 & \mathcal{R}[\![\mathcal{A}]\!] \vdash \mathcal{R}[\![P\ \textsf{sees}\ K(X|R)]\!] \\
\equiv & \qquad \{\text{definition rectify}\}
\end{array}$$

$$
\begin{array}{ll}
 & \mathcal{R}[\![\mathcal{A}]\!] \vdash P\ \mathsf{sees}\ K(X|R) \\
\equiv & \qquad \{\text{definition } \ \textit{allows}\ \} \\
 & \mathcal{R}[\![\mathcal{A}]\!]\ \textit{allows}\ P \rightarrow Q : K(X|R)
\end{array}
$$

Case (X, Y):

$$
\begin{array}{ll}
 & \mathcal{A}\ \textit{allows}\ P \rightarrow Q : (X, Y) \\
\equiv & \qquad \{\text{definition } \ \textit{allows}\ \} \\
 & \mathcal{A}\ \textit{allows}\ P \rightarrow Q : X \ \text{ and } \ \mathcal{A}\ \textit{allows}\ P \rightarrow Q : Y \\
\Rightarrow & \qquad \{\text{Induction Hypothesis}\} \\
 & \mathcal{R}[\![\mathcal{A}]\!]\ \textit{allows}\ P \rightarrow Q : X \ \text{ and } \ \mathcal{R}[\![\mathcal{A}]\!]\ \textit{allows}\ P \rightarrow Q : Y \\
\equiv & \qquad \{\text{definition } \ \textit{allows}\ \} \\
 & \mathcal{R}[\![\mathcal{A}]\!]\ \textit{allows}\ P \rightarrow Q : (X, Y)
\end{array}
$$

Case φ:

$$
\begin{array}{ll}
 & \mathcal{A}\ \textit{allows}\ P \rightarrow Q : \varphi \\
\equiv & \qquad \{\text{definition } \ \textit{allows}\ \} \\
 & \mathcal{A} \vdash P\ \mathsf{believes}\ \varphi \\
\Rightarrow & \qquad \{\text{Theorem } 9\} \\
 & \mathcal{R}[\![\mathcal{A}]\!] \vdash \mathcal{R}[\![P\ \mathsf{believes}\ \varphi]\!] \\
\equiv & \qquad \{\text{definition rectify}\} \\
 & \mathcal{R}[\![\mathcal{A}]\!] \vdash P\ \mathsf{rightly\_believes}\ \varphi \\
\Rightarrow & \qquad \{\text{definition } \ \mathsf{rightly\_believes}\ \} \\
 & \mathcal{R}[\![\mathcal{A}]\!] \vdash P\ \mathsf{believes}\ \varphi \\
\equiv & \qquad \{\text{definition } \ \textit{allows}\ \} \\
 & \mathcal{R}[\![\mathcal{A}]\!]\ \textit{allows}\ P \rightarrow Q : \varphi
\end{array}
$$

Case other X:

$$
\begin{array}{ll}
 & \mathcal{A}\ \textit{allows}\ P \rightarrow Q : X \\
\equiv & \qquad \{\text{definition } \ \textit{allows}\ \} \\
 & \text{true} \\
\equiv & \qquad \{\text{definition } \ \textit{allows}\ \} \\
 & \mathcal{R}[\![\mathcal{A}]\!]\ \textit{allows}\ P \rightarrow Q : X
\end{array}
$$

$\square$

Definition 22 *We define* positive *formulas as the least set such that:*

φ	*is positive if*	$\vdash \varphi$;
$K\ \mathsf{key\_of}\ (P, Q)$	*is positive*;	
$P\ \mathsf{possesses}\ K$	*is positive*;	
$P\ \mathsf{believes\ fresh}\ X$	*is positive*;	
$P\ \mathsf{believes}\ \varphi$, *other formulas*	*is positive if*	φ *is positive;*
$(P\ \mathsf{controls}\ \varphi) \wedge \varphi$	*is positive if*	φ *is positive;*
$(P\ \mathsf{believes}\ Q\ \mathsf{controls}\ \varphi) \wedge (R\ \mathsf{believes}\ \varphi)$	*is positive if*	φ *is positive;*
$P\ \mathsf{sees}\ X$	*is positive*;	
$P\ \mathsf{once\_said}\ X$	*is positive*;	
$\varphi \wedge \psi$	*is positive if*	φ *is positive and* ψ *is positive;*
$\varphi \vee \psi$	*is positive if*	φ *is positive and* ψ *is positive;*
$(\forall x :: \varphi)$	*is positive if*	$\varphi[x \leftarrow u]$ *is positive for all terms* u *of the appropriate kind not containing unbound variables.*

This is extended to finite sets of formulas: $\mathcal{F}$ is positive whenever the formula $\bigwedge \mathcal{F}$ is.

Lemma 23 *If* $\mathcal{A} \cup \mathbf{T}(a)$ *is positive and* $\mathcal{A}$ *allows* a*, then* $\models \{\mathcal{A}\}\, a\, \{\mathcal{A}\}$

This follows immediately from the following lemma:

Lemma 24 *If* $\mathcal{A}$ *allows* a*,* $\mathcal{A} \vdash \varphi$ *and* φ *positive, then* $\models \{\mathcal{A}\}\, a\, \{\varphi\}$

The proof of this lemma can be found in the appendix.

Lemma 25 *If* $\mathcal{A} \cup \mathbf{T}(a)$ *positive and* $\mathcal{A}$ *allows* a*, then* $\models \{\mathcal{A}\}\, a\, \{\mathcal{A}, \mathbf{T}(a)\}$.

Proof This follows now directly from Lemmas 17, 19 and 23. □

Definition 26 *We define a collection of predicates* $\mathcal{A}$ *to be* safe *iff* $\mathcal{A}$ *positive or there exists a positive collection* $\mathcal{A}'$ *such that* $\mathcal{A} = \mathcal{R}[\![\mathcal{A}']\!]$.

The following lemma follows immediately from the definition of safe.

Lemma 27 *If* $\mathcal{A}$ *is positive, then* $\mathcal{R}[\![\mathcal{A}]\!]$ *is safe.*

We now formulate Lemma 23 for the weaker requirement of $\mathcal{A}$ being safe, rather than positive.

Lemma 28 *If* $\mathcal{A} \cup \mathbf{T}(a)$ safe and $\mathcal{A}$ *allows* a*, then* $\models \{\mathcal{A}\}\, a\, \{\mathcal{A}\}$

The proof of this lemma can be found in the appendix.

Lemma 29 *If* $\mathcal{A} \cup \mathbf{T}(a)$ *safe and* $\mathcal{A}$ *allows* a*, then* $\models \{\mathcal{A}\}\, a\, \{\mathcal{A}, \mathbf{T}(a)\}$.

Proof This follows now directly from Lemmas 17, 19 and 23.
□

Theorem 30 *If* $\mathcal{A} \cup \mathbf{T}(\mathcal{P})$ *safe,* $\mathcal{A}$ *allows* $\mathcal{P}$ *and* $\mathcal{A} \cup \mathbf{T}(\mathcal{P}) \vdash \mathcal{C}$*, then* $\models \{\mathcal{A}\}\, \mathcal{P}\, \{\mathcal{C}\}$.

Proof We prove the theorem by induction on the protocol.

Case $\emptyset$:

$$
\begin{array}{ll}
& \mathcal{A} \cup \mathbf{T}(\emptyset) \vdash \mathcal{C} \\
\equiv & \qquad \{\text{definition of } \mathbf{T}(\emptyset)\} \\
& \mathcal{A} \vdash \mathcal{C} \\
\Rightarrow & \qquad \{\text{soundness (Theorem 11)}\} \\
& \text{for all states } s\text{: } s \models \mathcal{A} \text{ implies } s \models \mathcal{C} \\
\equiv & \qquad \{\text{definition of } \mathit{tra}(s,\emptyset)\} \\
& \text{for all states } s\text{: } s \models \mathcal{A} \text{ implies } \mathit{tra}(s,\emptyset) \models \mathcal{C} \\
\equiv & \qquad \{\text{definition of } \models \{\mathcal{A}\}\, \mathcal{P}\, \{\mathcal{C}\}\} \\
& \models \{\mathcal{A}\}\, \emptyset\, \{\mathcal{C}\}
\end{array}
$$

Case $(a;\mathcal{P})$:

$$
\begin{array}{ll}
& \mathcal{A} \cup \mathbf{T}(a;\mathcal{P}) \text{ safe, } \mathcal{A}\ \mathit{allows}\ (a;\mathcal{P}) \text{ and } \mathcal{A} \cup \mathbf{T}(a;\mathcal{P}) \vdash \mathcal{C} \\
\Rightarrow & \qquad \{\text{definitions of } \mathbf{T} \text{ and } \mathit{allows}\} \\
& \mathcal{A} \cup \mathbf{T}(a) \cup \mathbf{T}(\mathcal{P}) \text{ safe, } \mathcal{A}\ \mathit{allows}\ a,\ \mathcal{A} \cup \mathbf{T}(a)\ \mathit{allows}\ \mathcal{P} \text{ and } \mathcal{A} \cup \mathbf{T}(a) \cup \mathbf{T}(\mathcal{P}) \vdash \mathcal{C} \\
\Rightarrow & \qquad \{\text{Induction hypothesis}\} \\
& \mathcal{A} \cup \mathbf{T}(a) \text{ safe, } \mathcal{A}\ \mathit{allows}\ a \text{ and } \models \{\mathcal{A} \cup \mathbf{T}(a)\}\, \mathcal{P}\, \{\mathcal{C}\} \\
\Rightarrow & \qquad \{\text{Lemma 29}\} \\
& \models \{\mathcal{A}\}\, a\, \{\mathcal{A} \cup \mathbf{T}(a)\} \text{ and } \models \{\mathcal{A} \cup \mathbf{T}(a)\}\, \mathcal{P}\, \{\mathcal{C}\} \\
\Rightarrow & \qquad \{\text{composition of specifications (Lemma 16)}\} \\
& \models \{\mathcal{A}\}\, a;\mathcal{P}\, \{\mathcal{C}\}
\end{array}
$$

□

From Theorem 9 our main theorem now follows: a proof of a specification in the logic indeed ensures the right conclusions of the principals during the protocol.

Theorem 31 *If* $\mathcal{A} \cup \mathbf{T}(\mathcal{P})$ *positive,* $\mathcal{A}\ \mathit{allows}\ \mathcal{P}$ *and* $\mathcal{A} \cup \mathbf{T}(\mathcal{P}) \vdash \mathcal{C}$*, then* $\models \{\mathcal{R}[\![\mathcal{A}]\!]\}\, \mathcal{P}\, \{\mathcal{R}[\![\mathcal{C}]\!]\}$.

Proof Let $\mathcal{A} \cup \mathbf{T}(\mathcal{P})$ be positive, $\mathcal{A}\ \mathit{allows}\ \mathcal{P}$ and $\mathcal{A} \cup \mathbf{T}(\mathcal{P}) \vdash \mathcal{C}$. Then $\mathcal{R}[\![\mathcal{A} \cup \mathbf{T}(\mathcal{P})]\!]$ is safe (Lemma 27), $\mathcal{R}[\![\mathcal{A}]\!]\ \mathit{allows}\ \mathcal{P}$ (Lemma 21) and $\mathcal{R}[\![\mathcal{A}]\!] \cup \mathbf{T}(\mathcal{P}) \vdash \mathcal{R}[\![\mathcal{C}]\!]$ (Lemma 9). Now we can apply Theorem 30. □

8 Conclusion

The original BAN logic has proved successful for finding many unintended errors in security protocols. It does, though, not spot all potential security breaches and thus is less suited for finding intentional (possibly malicious) errors.

Our formalism resulted from a systematic attempt to formulate precise restrictions under which the changing beliefs of principals during a protocol run accurately reflect the changing state of affairs while communication takes place. The restrictions that emerged from our investigation are rather unelegant and complicated, the reason being that we wanted to keep them such that they can be checked statically and do not exclude well-known examples of protocols. They are, however, necessary; for most of the restrictions we have an example of a simple concrete protocol (not obeying the restriction) that leads to a false conclusion.

The semantic correctness criterium we have formulated is weak; it does not take the threats imposed by intruders and impostors into account. It remains to be investigated whether a stronger correctness criterium requires further restrictions. Extrapolating from our experience, we think in fact that that is rather likely. In view of the complications already engendered, an appropriate question is whether the approach of translating protocols into a logical framework is the most felicitous one.

References

[AT91] Martín Abadi and Mark R. Tuttle. *A Semantics for a Logic of Authentication (Extended Abstract),* ACM Symposium on Principles of Distributed Computing, 1991, Montreal, Quebec, Canada.

[BAN89] Michael Burrows, Martín Abadi and Roger Needham. *A Logic of Authentication,* Report 39, Digital Systems Research Center, Palo Alto, CA.

[BAN90] Michael Burrows, Martín Abadi and Roger Needham. *A Logic of Authentication,* ACM Transactions on Computer Systems, Vol. 8, No. 1, February 1990, pp 18-36.

[GEN69] G. Gentzen, M.E. Szabo (ed.). *The collected papers of Gerhard Gentzen,* North-Holland, Amsterdam, 1969.

[GNY90] Li Gong, Roger Needham and Raphael Yahalom. *Reasoning about Belief in Cryptographic Protocols,* IEEE Computer Society Symposium on Research in Security and Privacy, 1990.

[Ness90] Dan M. Nessett. *A critique of the Burrows, Abadi and Needham Logic,* Operating Systems Review, Vol. 24, No. 2, April 1990, pp 35-38

[Snek91] Einar Snekkenes. *Exploring the BAN Approach to Protocol Analysis,* IEEE Computer Society Symposium on Research in Security and Privacy, 1991, Oakland, CA.

[Syv91] Paul Syverson. *The Use of Logic in the Analysis of Cryptographic Protocols,* IEEE Computer Society Symposium on Research in Security and Privacy, 1991, Oakland, CA.

[YW93] Alec F. Yasinsac and William A. Wulf. *Evaluating Cryptographic Protocols,* Report CS-93-66, CS & IPC, University of Virginia.

Appendix

Proof of Lemma 24

Considering the semantics of specification triples, it is sufficient to prove for every positive formula φ with $\mathcal{A} \vdash \varphi$ that for arbitrary s, $tra(s,a) \models \varphi$ follows from $\mathcal{A} \cup \mathbf{T}(a)$ positive, $\mathcal{A}$ *allows* a and $s \models \mathcal{A}$. We will prove this by induction on the structure of (positive) φ. We write $a = A \to B : M$ and $s' = tra(s,a)$.

Case φ, for $\vdash \varphi$:

$$
\begin{array}{ll}
 & \vdash \varphi \\
\Rightarrow & \qquad \{\text{soundness}\} \\
 & \models \varphi \\
\Rightarrow & \qquad \{\text{instantiation for all}\} \\
 & s' \models \varphi
\end{array}
$$

Case $K\,\mathsf{key\_of}\,(P,Q)$: Note that, according to the semantics of $K\,\mathsf{key\_of}\,(P,Q)$, we can assume:

$$
\begin{array}{l}
\text{For all } R, X, Y, S \text{ such that } S \in \{P,Q\} \text{ and } K(Y|S) \in cts[\![X]\!] : \\
s \models R\,\mathsf{sees}\,X \text{ implies } s \models S\,\mathsf{once\_said}\,Y
\end{array}
$$

We have to prove $s' \models K\,\mathsf{key\_of}\,(P,Q)$, i.e. (for clarity we accompany s' with variables R', X', etc.):

$$
\begin{array}{l}
\text{For all } R', X', Y', S' \text{ such that } S' \in \{P,Q\} \text{ and } K(Y'|S') \in cts[\![X']\!] : \\
s' \models R'\,\mathsf{sees}\,X' \text{ implies } s' \models S'\,\mathsf{once\_said}\,Y'
\end{array}
$$

Let such R', X', Y', S' be given. Instantiating $Y = Y', S = S'$ in the assumption, we have:

$$
\begin{array}{l}
\text{For all } R, X \text{ such that } K(Y'|S') \in cts[\![X]\!] : \\
s \models R\,\mathsf{sees}\,X \text{ implies } s \models S'\,\mathsf{once\_said}\,Y'
\end{array}
$$

First, assume that for some R: $X' \in cts[\![\mathcal{S}_R]\!]$. Then:

$$
\begin{array}{ll}
 & X' \in cts[\![\mathcal{S}_R]\!] \\
\equiv & \qquad \{\text{definition of } cts[\![\,\_\,]\!] \text{ on sets of messages; for some } X\text{: }\} \\
 & X \in \mathcal{S}_R \wedge X' \in cts[\![X]\!] \\
\Rightarrow & \qquad \{K(Y'|S') \in cts[\![X']\!], \text{ transitivity of ``} \in cts[\![\,\_\,]\!]\text{''}\} \\
 & X \in \mathcal{S}_R \wedge K(Y'|S') \in cts[\![X]\!] \\
\equiv & \qquad \{\text{semantics of } \mathsf{sees}\} \\
 & s \models R\,\mathsf{sees}\,X \text{ and } K(Y'|S') \in cts[\![X]\!] \\
\Rightarrow & \qquad \{\text{instantiated assumption } s \models K\,\mathsf{key\_of}\,(P,Q)\} \\
 & s \models S'\,\mathsf{once\_said}\,Y' \\
\equiv & \qquad \{\text{semantics of } \mathsf{once\_said}\} \\
 & Y' \in \mathcal{O}_{S'} \\
\Rightarrow & \qquad \{\mathcal{O}_{S'} \text{ increases, i.e. } \mathcal{O}_{S'} \subseteq \mathcal{O}'_{S'}\} \\
 & Y' \in \mathcal{O}'_{S'} \\
\equiv & \qquad \{\text{semantics of } \mathsf{once\_said}\} \\
 & s' \models S'\,\mathsf{once\_said}\,Y'
\end{array}
$$

$\Rightarrow$ {introduction implication}

$s' \models R'\,\mathsf{sees}\,X'$ implies $s' \models S'\,\mathsf{once\_said}\,Y'$

Now assume that for no R: $X' \in cts[\![\mathcal{S}_R]\!]$; in particular, $X' \notin cts[\![\mathcal{S}_{R'}]\!]$. We want to prove that also in that case $s' \models R'\,\mathsf{sees}\,X'$ implies $s' \models S'\,\mathsf{once\_said}\,Y'$. So assume $s' \models R'\,\mathsf{sees}\,X'$, i.e., $X' \in \mathcal{S}'_{R'}$, and therefore $X' \in cts[\![\mathcal{S}'_{R'}]\!]$. The only participant R' that can have $\mathcal{S}_{R'} \neq \mathcal{S}'_{R'}$ is B. From $X' \in cts[\![\mathcal{S}'_B]\!] \setminus cts[\![\mathcal{S}_B]\!]$ and $cts[\![\mathcal{S}'_B]\!] = cts[\![\mathcal{S}_B]\!] \cup cts[\![M]\!]$ we see that $X' \in cts[\![M]\!]$, and so $K(Y'|S') \in cts[\![M]\!]$. Note further that $s' \models A\,\mathsf{once\_said}\,M$. We now prove by induction on the structure of messages that for all M:

If $s \models \mathcal{A}$, $\mathcal{A}$ *allows* $A \to B : M$, $K(Y'|S') \in cts[\![M]\!]$ and $s' \models A\,\mathsf{once\_said}\,M$,
then $s \models S'\,\mathsf{once\_said}\,Y'$

case $K(Y'|A)$:

$\mathcal{A}$ *allows* $A \to B : K(Y'|A)$

$\equiv$ {definition of *allows* ; for some P':}

$\mathcal{A} \vdash A\,\mathsf{believes}\,K\,\mathsf{key\_of}\,(A, P')$ and $\mathcal{A}$ *allows* $A \to P' : Y'$

$\Rightarrow$ {soundness, $s \models \mathcal{A}$}

$s \models A\,\mathsf{believes}\,K\,\mathsf{key\_of}\,(A, P')$

$\equiv$ {semantics of $\mathsf{believes}$}

$K\,\mathsf{key\_of}\,(A, P') \in \mathcal{B}_A$

$\Rightarrow$ {$\mathcal{B}_A$ increases}

$K\,\mathsf{key\_of}\,(A, P') \in \mathcal{B}'_A$

$\Rightarrow$ {assumption}

$K\,\mathsf{key\_of}\,(A, P') \in \mathcal{B}'_A$ and $s' \models A\,\mathsf{once\_said}\,K(Y'|S')$

$\Rightarrow$ {closure rule (iv), $A = S'$}

$s' \models S'\,\mathsf{once\_said}\,Y'$

case $K(Y'|S'), S' \neq A$:

$\mathcal{A}$ *allows* $A \to B : K(Y'|S')$

$\equiv$ {definition of *allows* }

$\mathcal{A} \vdash A \neq S' \wedge A\,\mathsf{sees}\,K(Y'|S')$

$\Rightarrow$ {soundness, $s \models \mathcal{A}$}

$s \models A\,\mathsf{sees}\,K(Y'|S')$

$\Rightarrow$ {semantics of $s \models K\,\mathsf{key\_of}\,(P, Q)$}

$s \models S'\,\mathsf{once\_said}\,Y'$

case $K_1(M|A)$: (with $K(Y'|S') \in cts[\![M]\!]$)

$\mathcal{A}$ *allows* $A \to B : K_1(M|A)$

$\equiv$ {definition of *allows* ; for some P':}

$\mathcal{A} \vdash A\,\mathsf{believes}\,K_1\,\mathsf{key\_of}\,(A, P')$ and $\mathcal{A}$ *allows* $A \to P' : M$

$\Rightarrow$ {soundness, $s \models \mathcal{A}$}

$s \models A\,\mathsf{believes}\,K_1\,\mathsf{key\_of}\,(A, P')$ and $\mathcal{A}$ *allows* $A \to P' : M$

$\Rightarrow$ {closure rule (iv)}

$$
\begin{array}{ll}
 & s' \models A \,\mathsf{once\_said}\, M \text{ and } \mathcal{A}\ \mathit{allows}\ A \to P' : M \\
\Rightarrow & \qquad \{\text{Induction Hypothesis}\} \\
 & s \models S' \,\mathsf{once\_said}\, Y'
\end{array}
$$

case $K_1(M|C), C \neq A$: (with $K(Y'|S') \in \mathit{cts}[\![M]\!]$)

$$
\begin{array}{ll}
 & \mathcal{A}\ \mathit{allows}\ A \to B : K_1(M|C) \\
\equiv & \qquad \{\text{definition of } \mathit{allows}\} \\
 & \mathcal{A} \vdash A \,\mathsf{sees}\, K_1(M|C) \\
\Rightarrow & \qquad \{\text{soundness, } s \models \mathcal{A}\} \\
 & s \models A \,\mathsf{sees}\, K_1(M|C) \\
\Rightarrow & \qquad \{\text{definition of } \mathit{cts}[\![\_]\!]\} \\
 & K(Y'|S') \in \mathit{cts}[\![K_1(M|C)]\!] \text{ and } s \models A \,\mathsf{sees}\, K_1(M|C) \\
\Rightarrow & \qquad \{\text{semantics of } s \models K \,\mathsf{key\_of}\, (P,Q)\} \\
 & s \models S' \,\mathsf{once\_said}\, Y'
\end{array}
$$

case (M_1, M_2):

$$
\begin{array}{ll}
 & \mathcal{A}\ \mathit{allows}\ A \to B : (M_1, M_2), K(Y'|S') \in \mathit{cts}[\![(M_1, M_2)]\!] \\
 & \text{ and } s' \models A \,\mathsf{once\_said}\, (M_1, M_2) \\
\Rightarrow & \qquad \{\text{definition of } \mathit{cts}[\![\_]\!]\} \\
 & \mathcal{A}\ \mathit{allows}\ A \to B : (M_1, M_2), (K(Y'|S') \in \mathit{cts}[\![M_1]\!] \text{ or } K(Y'|S') \in \mathit{cts}[\![M_2]\!]) \\
 & \text{ and } s' \models A \,\mathsf{once\_said}\, (M_1, M_2) \\
\Rightarrow & \qquad \{\text{for some } i \in \{1,2\}\} \\
 & \mathcal{A}\ \mathit{allows}\ A \to B : (M_1, M_2), K(Y'|S') \in \mathit{cts}[\![M_i]\!] \text{ and } s' \models A \,\mathsf{once\_said}\, (M_1, M_2) \\
\Rightarrow & \qquad \{\text{definition of } \mathit{allows}, \text{ semantics of } \mathsf{once\_said}\} \\
 & \mathcal{A}\ \mathit{allows}\ A \to B : M_i, K(Y'|S') \in \mathit{cts}[\![M_i]\!] \text{ and } s' \models A \,\mathsf{once\_said}\, M_i \\
\Rightarrow & \qquad \{\text{Induction Hypothesis}\} \\
 & s \models S' \,\mathsf{once\_said}\, Y'
\end{array}
$$

cases φ, other X: For these cases the assumption $K(Y'|S') \in \mathit{cts}[\![M]\!]$ cannot hold.

This proves by induction the claim

> If $s \models \mathcal{A}$, $\mathcal{A}\ \mathit{allows}\ A \to B : M$, $K(Y'|S') \in \mathit{cts}[\![M]\!]$ and $s' \models A \,\mathsf{once\_said}\, M$, then $s \models S' \,\mathsf{once\_said}\, Y'$

which concludes Case $K \,\mathsf{key\_of}\, (P,Q)$ of the induction of Lemma 23. We continue with the remaining Cases.

Case $(P \,\mathsf{controls}\, \varphi) \wedge \varphi$:

$$
\begin{array}{ll}
 & s \models (P \,\mathsf{controls}\, \varphi) \wedge \varphi \\
\Rightarrow & \qquad \{\text{semantics of } \wedge, \text{ elimination and}\} \\
 & s \models \varphi \\
\Rightarrow & \qquad \{\varphi \text{ positive, Induction Hypothesis}\} \\
 & s' \models \varphi
\end{array}
$$

$$
\begin{array}{ll}
\Rightarrow & \quad \{\text{introduction implies, and}\} \\
& (s' \models P\,\mathsf{believes}\,\varphi \text{ implies } s' \models \varphi) \text{ and } s' \models \varphi \\
\equiv & \quad \{\text{semantics and, implies, definition of } \mathsf{controls}\} \\
& s' \models (P\,\mathsf{controls}\,\varphi) \wedge \varphi
\end{array}
$$

Case $(P\,\mathsf{believes}\,Q\,\mathsf{controls}\,\varphi) \wedge (R\,\mathsf{believes}\,\varphi)$:

$$
\begin{array}{ll}
& s \models (P\,\mathsf{believes}\,Q\,\mathsf{controls}\,\varphi) \wedge (R\,\mathsf{believes}\,\varphi) \\
\equiv & \quad \{\text{semantics of } \wedge \text{ and } \mathsf{believes}\} \\
& Q\,\mathsf{controls}\,\varphi \in \mathcal{B}_P \text{ and } \varphi \in \mathcal{B}_R \\
\Rightarrow & \quad \{\mathcal{B}_P \text{ and } \mathcal{B}_R \text{ increase}\} \\
& Q\,\mathsf{controls}\,\varphi \in \mathcal{B}'_P \text{ and } \varphi \in \mathcal{B}'_R \\
\equiv & \quad \{\text{semantics of } \wedge \text{ and } \mathsf{believes}\} \\
& s' \models (P\,\mathsf{believes}\,Q\,\mathsf{controls}\,\varphi) \wedge (R\,\mathsf{believes}\,\varphi)
\end{array}
$$

Cases $P\,\mathsf{possesses}\,K$, $P\,\mathsf{sees}\,X$, $P\,\mathsf{once\_said}\,X$, $P\,\mathsf{believes}\,\varphi$ and $P\,\mathsf{believes\,fresh}\,X$:

$$
\begin{array}{ll}
& s \models P\,\mathsf{possesses}\,K \\
\equiv & \quad \{\text{semantics of } \mathsf{possesses}\} \\
& K \in \mathcal{K}_P \\
\Rightarrow & \quad \{\mathcal{K}_P \text{ increases, i.e. } \mathcal{K}_P \subseteq \mathcal{K}'_P\} \\
& K \in \mathcal{K}'_P \\
\equiv & \quad \{\text{semantics of } \mathsf{possesses}\} \\
& s' \models P\,\mathsf{possesses}\,K
\end{array}
$$

Since $\mathcal{S}_P$, $\mathcal{O}_P$ and $\mathcal{B}_P$ are also increase, the proofs for $P\,\mathsf{sees}\,X$, $P\,\mathsf{once\_said}\,X$, $P\,\mathsf{believes}\,\varphi$ and $P\,\mathsf{believes\,fresh}\,X$ are analogous.

Cases $\varphi \wedge \psi$, $\varphi \vee \psi$, $\forall x :: \varphi$:

$$
\begin{array}{ll}
& s \models \varphi \wedge \psi \\
\equiv & \quad \{\text{semantics of } \wedge\} \\
& s \models \varphi \text{ and } s \models \psi \\
\equiv & \quad \{\text{definition of } \varphi \wedge \psi \text{ positive}\} \\
& \varphi \text{ and } \psi \text{ positive, and } s \models \varphi \text{ and } s \models \psi \\
\Rightarrow & \quad \{\text{Induction Hypothesis}\} \\
& s' \models \varphi \text{ and } s' \models \psi \\
\equiv & \quad \{\text{semantics of } \wedge\} \\
& s' \models \varphi \wedge \psi
\end{array}
$$

The proofs for $\varphi \vee \psi$ and $\forall x :: \varphi$ are analogous.
□

Proof of Lemma 28 The proof is analogous to the proof of Lemma 24. According to the semantics of specification triples, it is sufficient to prove for every formula ψ in $\mathcal{A}$ that for arbitrary s, $\mathit{tra}(s,a) \models \psi$ follows from $\mathcal{A} \cup \mathbf{T}(a)$ safe, $\mathcal{A}$ *allows* a and $s \models \mathcal{A}$. For positive $\psi \in \mathcal{A}$ Lemma 24 ensures that $s' \models \psi$. For $\psi = \mathcal{R}[\![\varphi]\!]$ (where φ is positive) we must prove that either $\mathcal{R}[\![\varphi]\!]$ is positive, or $s' \models \mathcal{R}[\![\varphi]\!]$ under the assumptions mentioned. We prove this by induction on the structure of (positive) φ. Before embarging on the induction, note that formulas of each of the forms $K\,\mathsf{key\_of}\,(P,Q)$, $P\,\mathsf{possesses}\,K$, $P\,\mathsf{sees}\,X$ and $P\,\mathsf{once\_said}\,X$ are invariant under the mapping $\mathcal{R}[\![-]\!]$, so that we can skip them in the case analysis.

Case $\varphi, \vdash \varphi$:

$$\begin{array}{ll} & \vdash \varphi \\ \Rightarrow & \qquad \{\text{Theorem 9}\} \\ & \vdash \mathcal{R}[\![\varphi]\!] \\ \Rightarrow & \qquad \{\text{definition positive}\} \\ & \mathcal{R}[\![\varphi]\!] \text{ positive} \end{array}$$

Case $P\,\mathsf{believes\ fresh}\,X$: We postpone this complicated case to the end of the proof.

Case $P\,\mathsf{believes}\,\varphi$, φ positive:

$$\begin{array}{ll} & \mathcal{R}[\![P\,\mathsf{believes}\,\varphi]\!] \\ = & \qquad \{\text{definition of } \mathcal{R}[\![\_]\!]\} \\ & P\,\mathsf{rightly\_believes}\,\varphi \\ = & \qquad \{\text{definition of } \mathsf{rightly\_believes}\} \\ & P\,\mathsf{believes}\,\varphi \wedge \varphi \end{array}$$

which is positive, because φ is.

Case $P\,\mathsf{controls}\,\varphi \wedge \varphi$, φ positive:

$$\begin{array}{ll} & s \models \mathcal{R}[\![P\,\mathsf{controls}\,\varphi \wedge \varphi]\!] \\ \equiv & \qquad \{\text{definition of } \mathcal{R}[\![\_]\!]\} \\ & s \models \mathcal{R}[\![P\,\mathsf{controls}\,\varphi]\!] \wedge \mathcal{R}[\![\varphi]\!] \\ \Rightarrow & \qquad \{\text{proposition calculus on the semantic level}\} \\ & s \models \mathcal{R}[\![\varphi]\!] \\ \Rightarrow & \qquad \{\text{Induction Hypothesis, } \varphi \text{ positive}\} \\ & s' \models \mathcal{R}[\![\varphi]\!] \\ \equiv & \qquad \{\text{proposition calculus on the semantic level}\} \\ & s' \models (P\,\mathsf{rightly\_believes}\,\varphi \;\rightarrow\; (P\,\mathsf{rightly\_believes}\,\varphi \wedge \mathcal{R}[\![\varphi]\!])) \wedge \mathcal{R}[\![\varphi]\!] \\ \equiv & \qquad \{\text{definition of } \mathcal{R}[\![\_]\!]\} \\ & s' \models \mathcal{R}[\![(P\,\mathsf{believes}\,\varphi \;\rightarrow\; P\,\mathsf{believes}\,\varphi \wedge \varphi)]\!] \wedge \mathcal{R}[\![\varphi]\!] \\ \equiv & \qquad \{\text{definition of } \mathsf{rightly\_believes}\} \\ & s' \models \mathcal{R}[\![(P\,\mathsf{believes}\,\varphi \;\rightarrow\; P\,\mathsf{rightly\_believes}\,\varphi)]\!] \wedge \mathcal{R}[\![\varphi]\!] \\ \equiv & \qquad \{\text{definition of } \mathsf{controls}\} \\ & s' \models \mathcal{R}[\![P\,\mathsf{controls}\,\varphi]\!] \wedge \mathcal{R}[\![\varphi]\!] \\ \equiv & \qquad \{\text{definition of } \mathcal{R}[\![\_]\!]\} \\ & s' \models \mathcal{R}[\![P\,\mathsf{controls}\,\varphi \wedge \varphi]\!] \end{array}$$

Case $(P\,\mathsf{believes}\,Q\,\mathsf{controls}\,\varphi) \wedge (R\,\mathsf{believes}\,\varphi)$, φ positive:

$$\begin{array}{ll} & s \models \mathcal{R}[\![(P\,\mathsf{believes}\,Q\,\mathsf{controls}\,\varphi) \wedge (R\,\mathsf{believes}\,\varphi)]\!] \\ \equiv & \qquad \{\text{definition of } \mathcal{R}[\![\_]\!]\} \\ & s \models P\,\mathsf{rightly\_believes}\,Q\,\mathsf{controls}\,\varphi \wedge R\,\mathsf{rightly\_believes}\,\varphi \\ \equiv & \qquad \{\text{definition } \mathsf{rightly\_believes}\} \\ & s \models P\,\mathsf{believes}\,Q\,\mathsf{controls}\,\varphi \wedge Q\,\mathsf{controls}\,\varphi \wedge R\,\mathsf{believes}\,\varphi \wedge \varphi \\ \Rightarrow & \qquad \{\text{semantics } \wedge,\ \mathsf{believes},\ \mathcal{B}_P \text{ and } \mathcal{B}_R \text{ monotonous, } \varphi \text{ positive}\} \end{array}$$

$s' \models P\,\mathsf{believes}\,Q\,\mathsf{controls}\,\varphi \wedge R\,\mathsf{believes}\,\varphi \wedge \varphi$

$\Rightarrow$ {introduction implies}

$s' \models P\,\mathsf{believes}\,Q\,\mathsf{controls}\,\varphi \wedge ((Q\,\mathsf{believes}\,\varphi) \;\rightarrow\; \varphi) \wedge R\,\mathsf{believes}\,\varphi \wedge \varphi$

$\equiv$ {definition $\mathsf{controls}$}

$s' \models P\,\mathsf{believes}\,Q\,\mathsf{controls}\,\varphi \wedge Q\,\mathsf{controls}\,\varphi \wedge R\,\mathsf{believes}\,\varphi \wedge \varphi$

$\equiv$ {definition $\mathsf{rightly\_believes}$}

$s' \models P\,\mathsf{rightly\_believes}\,Q\,\mathsf{controls}\,\varphi \wedge R\,\mathsf{rightly\_believes}\,\varphi$

$\equiv$ {definition of $\mathcal{R}[\![-]\!]$}

$s' \models \mathcal{R}[\![(P\,\mathsf{believes}\,Q\,\mathsf{controls}\,\varphi) \wedge (R\,\mathsf{believes}\,\varphi)]\!]$

Cases $\varphi_1 \wedge \varphi_2$, $\varphi_1 \vee \varphi_2$ and $\forall x :: \varphi$:

$s \models \varphi \wedge \psi$

$\equiv$ {semantics $\wedge$}

$s \models \varphi$ and $s \models \psi$

$\equiv$ {definition $\varphi \wedge \psi$ positive}

φ and ψ positive, and $s \models \varphi$ and $s \models \psi$

$\Rightarrow$ {Induction Hypothesis}

$s' \models \varphi$ and $s' \models \psi$

$\equiv$ {semantics $\wedge$}

$s' \models \varphi \wedge \psi$

The proofs for $\varphi \vee \psi$ and $\forall x :: \varphi$ are analogous.

Finally, the postponed

Case $P\,\mathsf{believes}\,\mathsf{fresh}\,X$: Observe that $\mathcal{R}[\![P\,\mathsf{believes}\,\mathsf{fresh}\,X]\!] = P\,\mathsf{rightly\_believes}\,\mathsf{fresh}\,X = P\,\mathsf{believes}\,\mathsf{fresh}\,X \wedge \mathsf{fresh}\,X$. The first conjunct of this expression is positive, which is enough to prove our claim. The second conjunct is not positive, hence we need to prove $s' \models \mathsf{fresh}\,X$ separately.

According to the definition of $\mathsf{fresh}\,X$, we can assume:

For all P, φ : $s \models P\,\mathsf{once\_said}\,(X, \varphi) \;\rightarrow\; P\,\mathsf{believes}\,\varphi$

We have to prove:

For all P, φ : $s' \models P\,\mathsf{once\_said}\,(X, \varphi) \;\rightarrow\; P\,\mathsf{believes}\,\varphi$

Let P and φ be given. We now distinguish three cases to prove the above implication: $(X, \varphi) \in \mathcal{O}_P$, $(X, \varphi) \notin \mathcal{O}'_P$ and finally $(X, \varphi) \in \mathcal{O}'_P \backslash \mathcal{O}_P$. Note that it is sufficient to prove $s' \models P\,\mathsf{believes}\,\varphi$.

$(X, \varphi) \in \mathcal{O}_P$

$\equiv$ {semantics of $\mathsf{once\_said}$}

$s \models P\,\mathsf{once\_said}\,(X, \varphi)$

$\Rightarrow$ {assumption $s \models \mathsf{fresh}\,X$}

$s \models P\,\mathsf{believes}\,\varphi$

$\equiv$ {semantics of $\mathsf{believes}$}

$\varphi \in \mathcal{B}_P$

$\Rightarrow$ {$\mathcal{B}_P$ increases}

$$
\begin{array}{ll}
& \varphi \in \mathcal{B}'_P \\
\equiv & \qquad \{\text{semantics of } \mathsf{believes}\} \\
& s' \models P \,\mathsf{believes}\, \varphi
\end{array}
$$

$$
\begin{array}{ll}
& (X,\varphi) \notin \mathcal{O}'_P \\
\equiv & \qquad \{\text{semantics of } \mathsf{once\_said}\} \\
& \text{not } (s' \models P \,\mathsf{once\_said}\, (X,\varphi)) \\
\Rightarrow & \qquad \{\text{introduction implies}\} \\
& s' \models P \,\mathsf{once\_said}\, (X,\varphi) \text{ implies } s' \models P \,\mathsf{believes}\, \varphi \\
\equiv & \qquad \{\text{semantics implies}\} \\
& s' \models P \,\mathsf{once\_said}\, (X,\varphi) \ \rightarrow\ P \,\mathsf{believes}\, \varphi
\end{array}
$$

For $(X,\varphi) \in \mathcal{O}'_P \backslash \mathcal{O}_P$, we first observe that $P = A$ (only the sender's $\mathcal{O}$ set has changed). We now prove the following statement by induction on M:

> If $(X,\varphi) \in \mathcal{O}'_A \backslash \mathcal{O}_A$, $(X,\varphi) \in cts[\![M]\!]$, $\mathcal{A}$ *allows* $A \rightarrow B : M$, $s \models \mathcal{A}$ and $s' \models A \,\mathsf{once\_said}\, M$, then $s' \models A \,\mathsf{believes}\, \varphi$.

case (X,φ):

$$
\begin{array}{ll}
& \mathcal{A} \textit{ allows } A \rightarrow B : (X,\varphi) \\
\equiv & \qquad \{\text{definition of } \textit{allows}\} \\
& \mathcal{A} \textit{ allows } A \rightarrow B : X \text{ and } \mathcal{A} \textit{ allows } A \rightarrow B : \varphi \\
\Rightarrow & \qquad \{\text{definition of } \textit{allows}\} \\
& \mathcal{A} \vdash A \,\mathsf{believes}\, \varphi \\
\Rightarrow & \qquad \{\text{soundness}\} \\
& s \models A \,\mathsf{believes}\, \varphi \\
\Rightarrow & \qquad \{\text{semantics of } \mathsf{believes}, \text{ monotonicity } \mathcal{B}_A\} \\
& s' \models A \,\mathsf{believes}\, \varphi
\end{array}
$$

case (M_1, M_2): $\quad ((X,\varphi) \in cts[\![M_1]\!] \cup cts[\![M_2]\!])$

$$
\begin{array}{ll}
& (X,\varphi) \in cts[\![M_1]\!] \text{ or } (X,\varphi) \in cts[\![M_2]\!] \\
\Rightarrow & \qquad \{\text{two analogous cases}\} \\
& (X,\varphi) \in cts[\![M_i]\!] \\
\equiv & \qquad \{\text{assumptions}\} \\
& (X,\varphi) \in cts[\![M_i]\!], \mathcal{A} \textit{ allows } A \rightarrow B : (M_1, M_2) \text{ and } s' \models A \,\mathsf{once\_said}\, (M_1, M_2) \\
\Rightarrow & \qquad \{\text{definition of } \textit{allows}, \text{ semantics of } \mathsf{once\_said}, \text{ closure rule (iii)}\} \\
& (X,\varphi) \in cts[\![M_i]\!], \mathcal{A} \textit{ allows } A \rightarrow B : M_i \text{ and } s' \models A \,\mathsf{once\_said}\, M_i \\
\Rightarrow & \qquad \{\text{Induction Hypothesis}\} \\
& s' \models A \,\mathsf{believes}\, \varphi
\end{array}
$$

case $K(Y|A)$: $\quad ((X,\varphi) \in cts[\![Y]\!])$

$$
\begin{array}{ll}
& \mathcal{A}\ \mathit{allows}\ A \to B : K(Y|A) \\
\Rightarrow & \qquad \{\text{definition of } \mathit{allows}\text{; for some } P'\text{:}\} \\
& \mathcal{A}\ \mathit{allows}\ A \to P' : Y \\
\Rightarrow & \qquad \{\text{assumptions}\} \\
& \mathcal{A}\ \mathit{allows}\ A \to P' : Y, s' \models A\, \textsf{once\_said}\, K(Y|A) \text{ and } (X,\varphi) \in \mathit{cts}[\![Y]\!] \\
\Rightarrow & \qquad \{\text{closure rule (iv)}\} \\
& \mathcal{A}\ \mathit{allows}\ A \to P' : Y, s' \models A\, \textsf{once\_said}\, Y \text{ and } (X,\varphi) \in \mathit{cts}[\![Y]\!] \\
\Rightarrow & \qquad \{\text{Induction Hypothesis}\} \\
& s' \models A\, \textsf{believes}\, \varphi
\end{array}
$$

case $K(Y|Q), Q \neq A$: This case cannot occur:

$$
\begin{array}{ll}
& (X,\varphi) \in \mathcal{O}'_A \backslash \mathcal{O}_A \\
\equiv & \qquad \{\text{definition closure}\} \\
& (X,\varphi) \in (\mathit{clo}(\mathcal{O}_A \cup \{K(Y|Q)\})) \backslash \mathcal{O}_A \\
\equiv & \qquad \{Q \neq A, \text{ closure rule (iv)}\} \\
& (X,\varphi) \in (\mathcal{O}_A \cup \{K(Y|Q)\}) \backslash \mathcal{O}_A \\
\Rightarrow & \qquad \{\text{set subtraction}\} \\
& (X,\varphi) \in \{K(Y|Q)\} \\
\equiv & \qquad \{\text{singleton}\} \\
& (X,\varphi) = K(Y|Q) \\
\equiv & \qquad \{\text{joint messages containing formulas are distinct from encrypted messages}\} \\
& \text{false}
\end{array}
$$

This proves by induction the claim

> If $(X,\varphi) \in \mathcal{O}'_A \backslash \mathcal{O}_A$, $(X,\varphi) \in \mathit{cts}[\![M]\!]$, $\mathcal{A}\ \mathit{allows}\ A \to B : M$, $s \models \mathcal{A}$ and $s' \models A\, \textsf{once\_said}\, M$, then $s' \models A\, \textsf{believes}\, \varphi$.

which concludes Case $\textsf{fresh}\, X$ of the induction of Lemma 28.
□