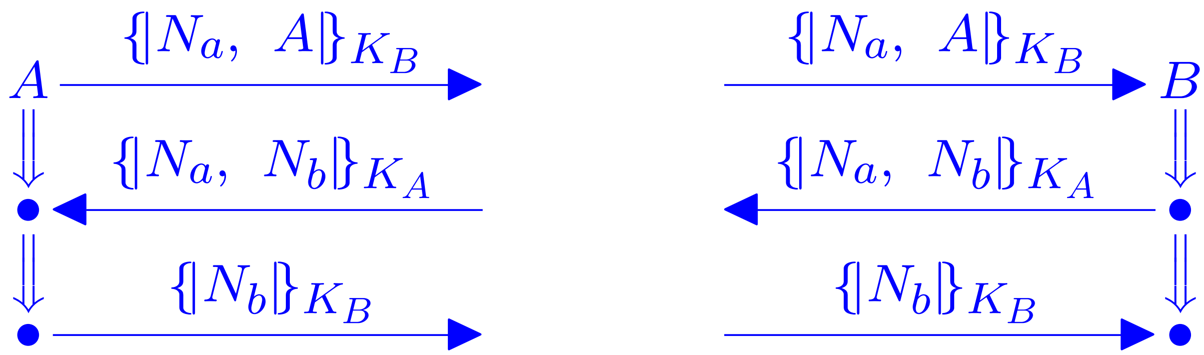
Trust Engineering with Cryptographic Protocols

Joshua D. Guttman, F. Javier Thayer

April 2004

Supported by the **National Security Agency**
and the **MITRE-Sponsored Research** Program

Needham-Schroeder



K_A, K_B	Public (asymmetric) keys of A, B
N_a, N_b	Nonces, one-time random bitstrings
$\{\|t\|\}_K$	Encryption of t with K
$N_a \oplus N_b$	New shared secret

Essence of Cryptography (for today)

Public key cryptography: algorithm using
two related values, one private, the other public

- Encryption: Public key makes ciphertext, only private key owner can decrypt
- Signature: Private key makes ciphertext, anyone can verify signature with public key

A's public key: K_A A's private key: K_A^{-1}

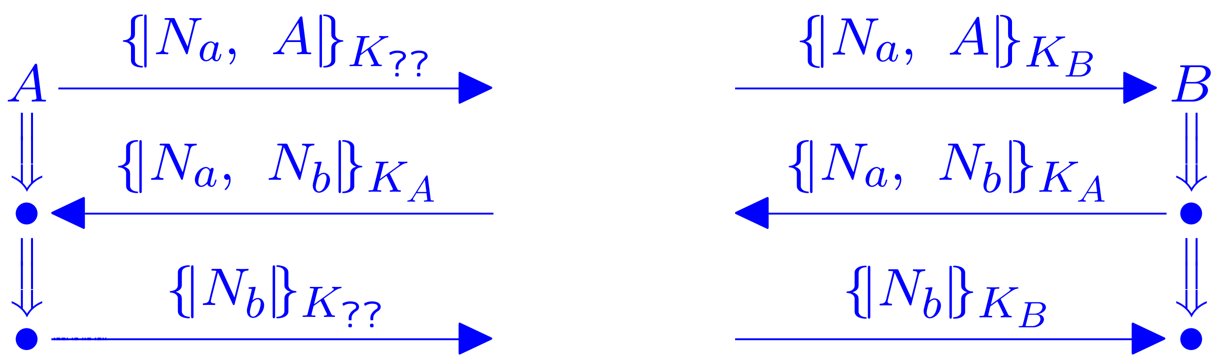
Symmetric key cryptography: algorithm using
a single value, shared as a secret between sender, receiver

- Same key makes ciphertext, extracts plaintext

$K = K^{-1}$

Needham-Schroeder: How does it work?

Assume A's private key K_A^{-1} uncompromised

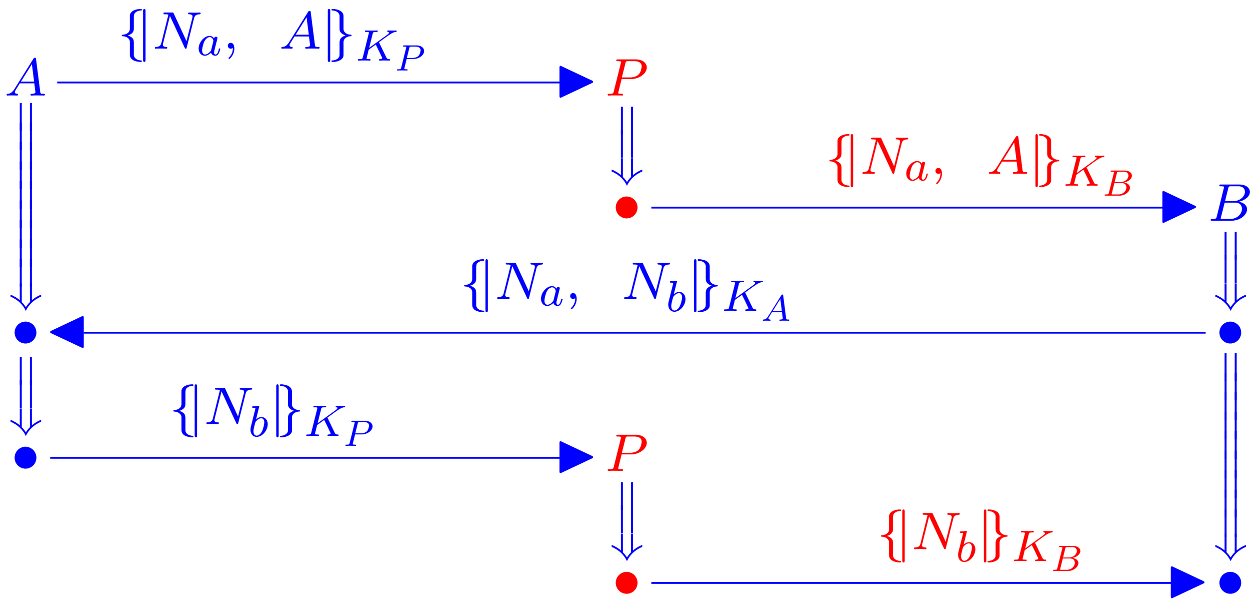


K_A, K_B	Public (asymmetric) keys of A, B
N_a, N_b	Nonces, one-time random bitstrings
$\{\!\|t\|\!\}_K$	Encryption of t with K
$N_a \oplus N_b$	New shared secret

Whoops

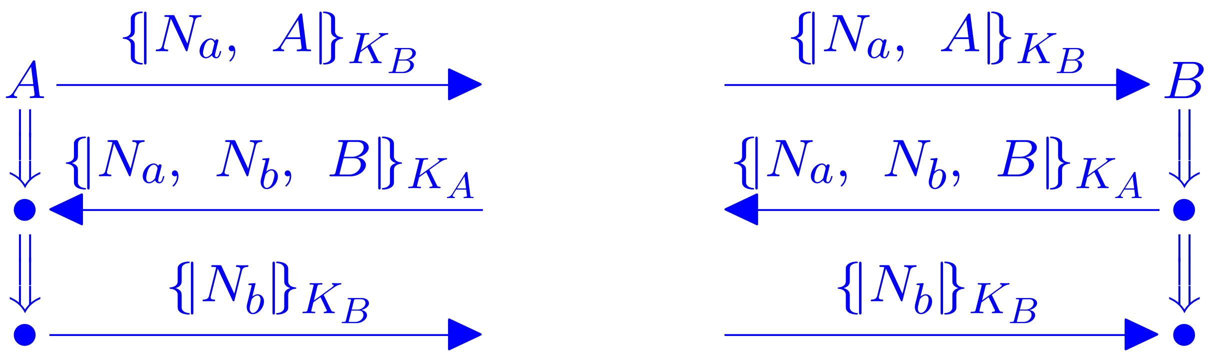
Needham-Schroeder Failure

If ?? $= P$,



(Gavin Lowe)

Needham-Schroeder-Lowe



K_A, K_B	Public (asymmetric) keys of A, B
N_a, N_b	Nonces, one-time random bitstrings
$\{\vert t \vert\}_K$	Encryption of t with K
$N_a \oplus N_b$	New shared secret

Protocol Analysis

Protocol analysis tells us:

- What happened (e.g. authentication properties)
- What didn't happen (e.g. secrecy failures)

Formalized in (e.g.) strand space theory

- Behaviors of regular principals are "strands"
- Adversary actions represented as special strands
- Executions are causally well-founded graphs

Very powerful proof methods: "Authentication tests"

- Compact proofs of many protocols
- Failed proofs suggest attacks
- Useful protocol design heuristics

Authentication test method illustrated on previous slides

Goal for Remainder of Talk

Reason about real world consequences of cryptographic protocols

- Capitalize on methods for protocol analysis and design

Examples:

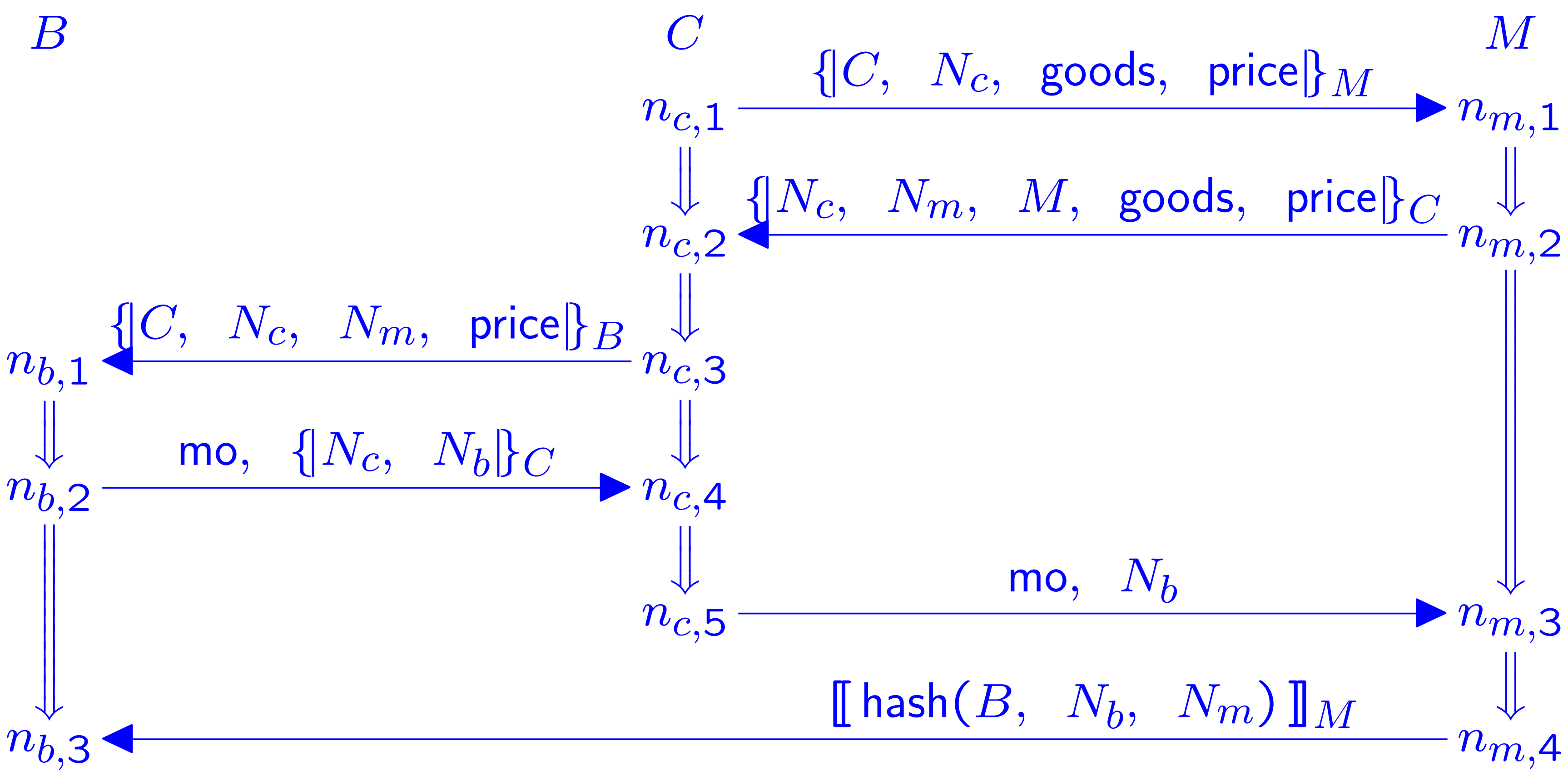
- Distributed access control
  - Principals cooperate to share resources selectively
  - As formulated via trust management logic
- Electronic retail commerce
  - When is customer committed to paying?
  - When is merchant committed to shipping?
  - Whose word did you depend on when deciding?

control access
(or actions) via distribu
logical deduction

Main idea: Enrich strand space framework with
formulas from a trust management logic

- Formulas for message transmissions are guaranteed by sender
- Formulas for message receipt are assumptions the receiver relies on

An Example: EPMO



Electronic Purchase using Money Order

$\text{mo} = [\![\text{hash}(C, N_c, N_b, N_m, \text{price})]\!]_B$

Nonce-based cryptographic protocols

Authenticate peer

- Demonstrable to third party (in some protocols)

Guarantee loosely synchronous interaction

- Unpredictable nonce establishes causal ordering
- Message recent if it incorporates recently generated nonce

Establish shared secrets

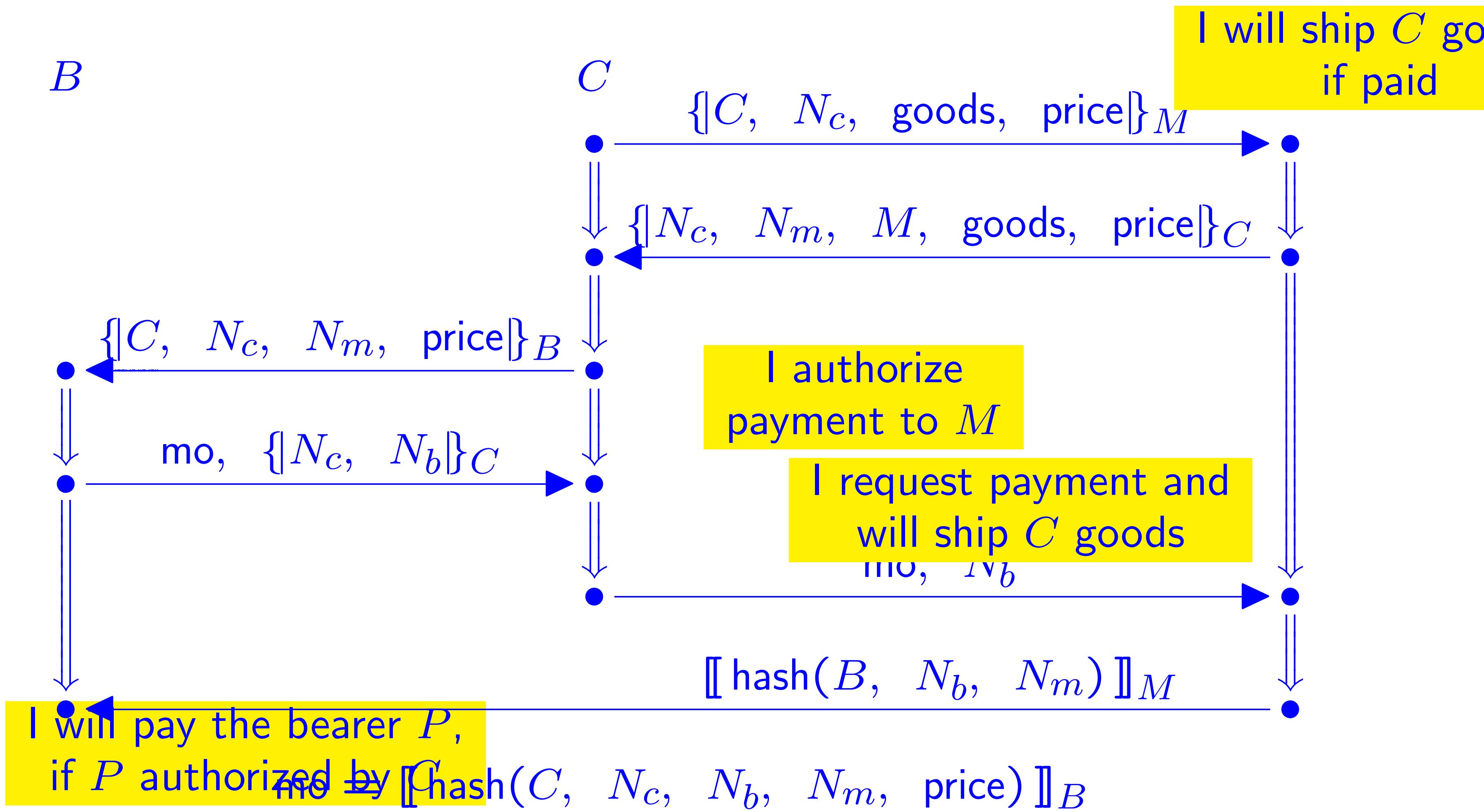
- Temporary secrets — M, B
- Permanent secrets — price
- Secrets shared among subset of principals — goods

Strand space theory focuses on

- Causal structure of protocol interactions
- Properties of protocols mentioned above

and provides strong protocol design methods

EPMO: Commitments on sends



Trust management and protocols

Each principal P

- Reasons locally in Th_P
- Derives guarantee before transmitting message
- Relies on assertions of others as premises

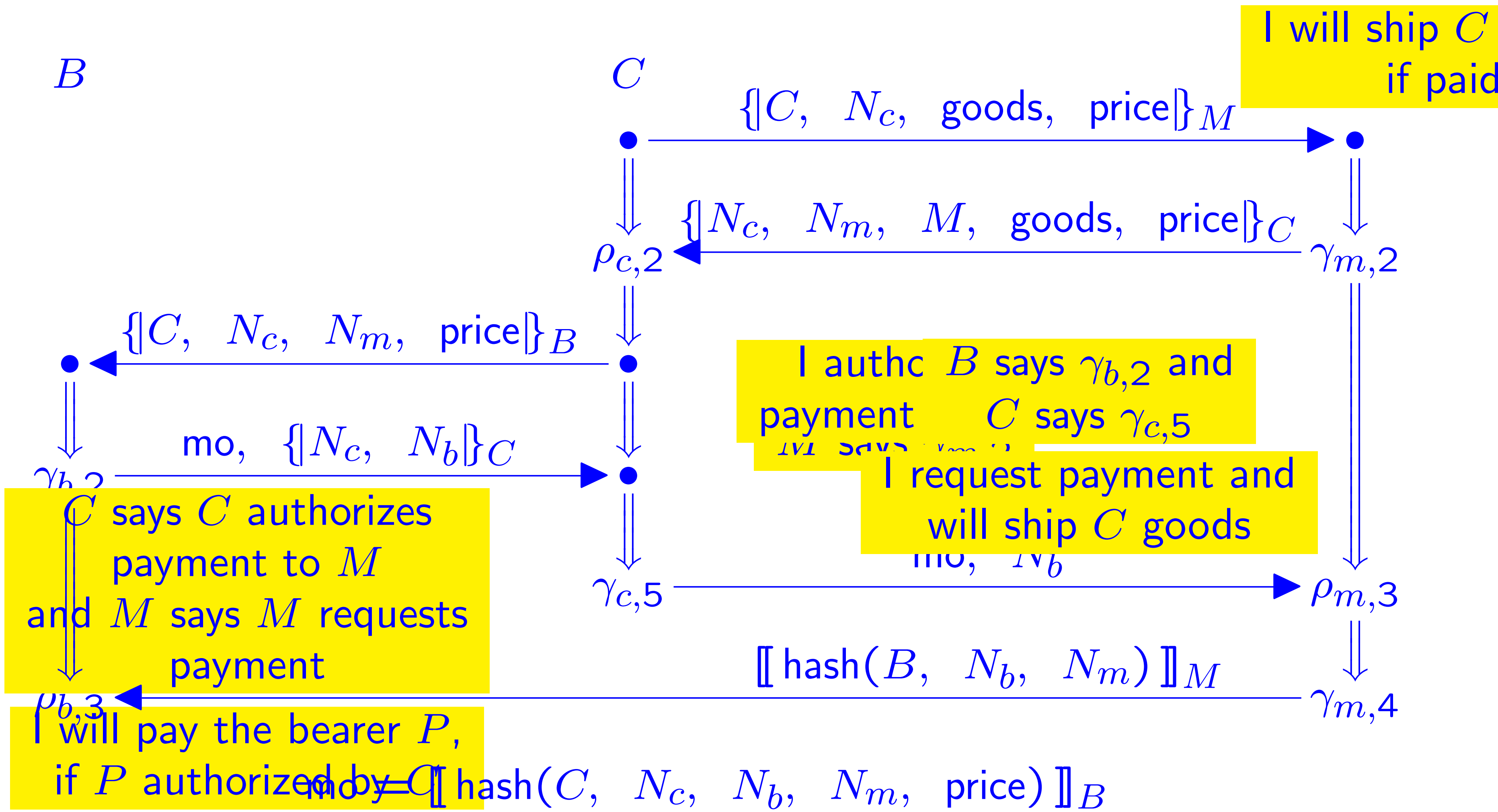
Premises: formulas associated with message receptions

- Specifies what recipient may rely on, e.g.
  "B says 'I will transfer funds if authorized' "
- Provides local representation of remote guarantee
- Th_P determines whether ϕ follows from P' says ϕ

Role of protocol

- When I rely on you having asserted a formula,
  then you did guarantee that assertion
- Coordination mechanism for rely/guarantees
- Sound protocol: "relies" always backed by "guarantees"

EPMO: Rely/Guarantee Formulas



Contrast: Earlier Work

The BAN tradition

- Messages are formulas or formulas idealize messages
- Who asserted the formulas?
- Who drew consequences from formulas?

Embedding formulas explicitly inside messages

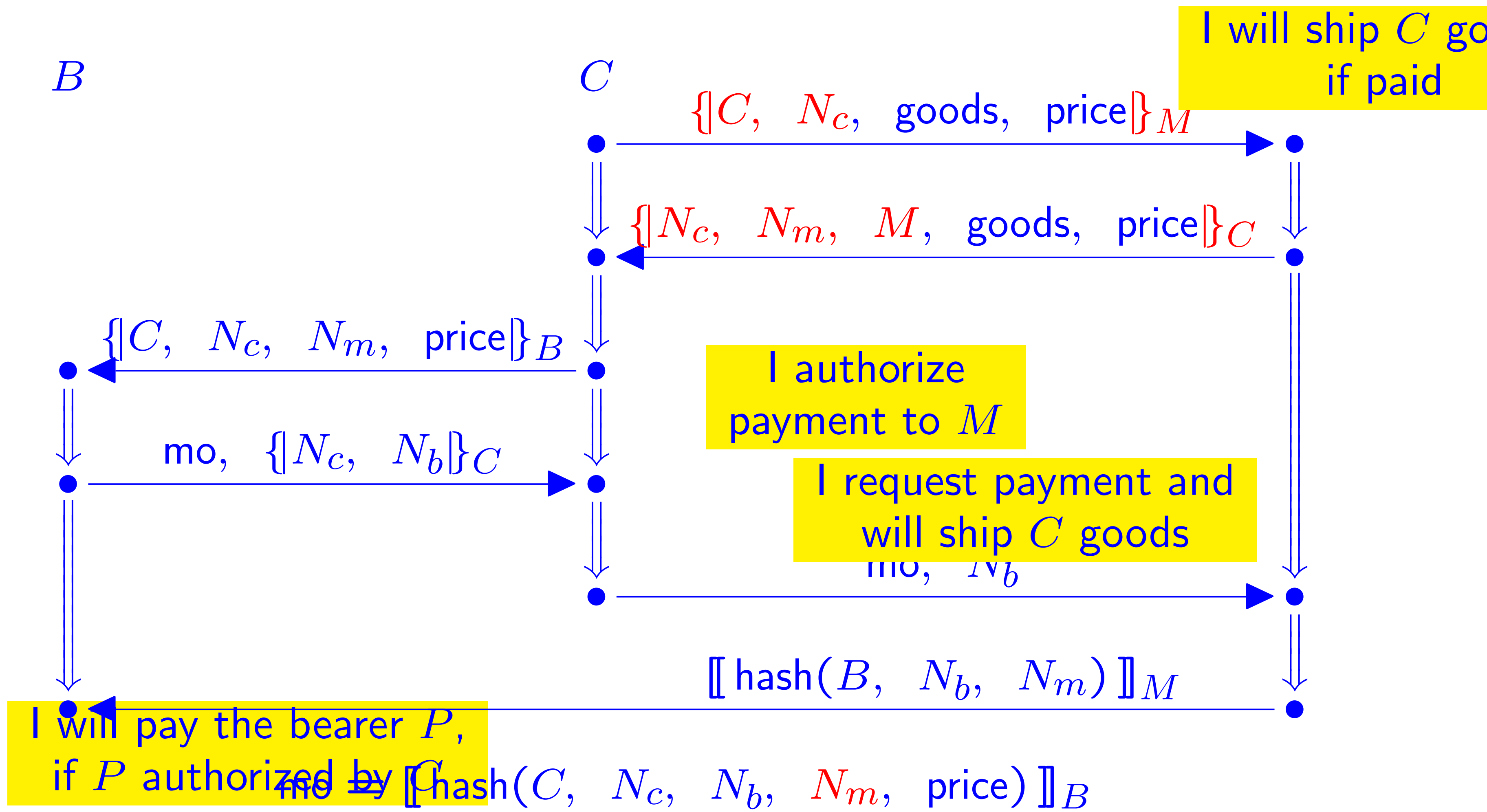
starts
with LAWB

- Main view of logical trust mgt
- Formulas parsed out of certificates
- Problem of partial information?

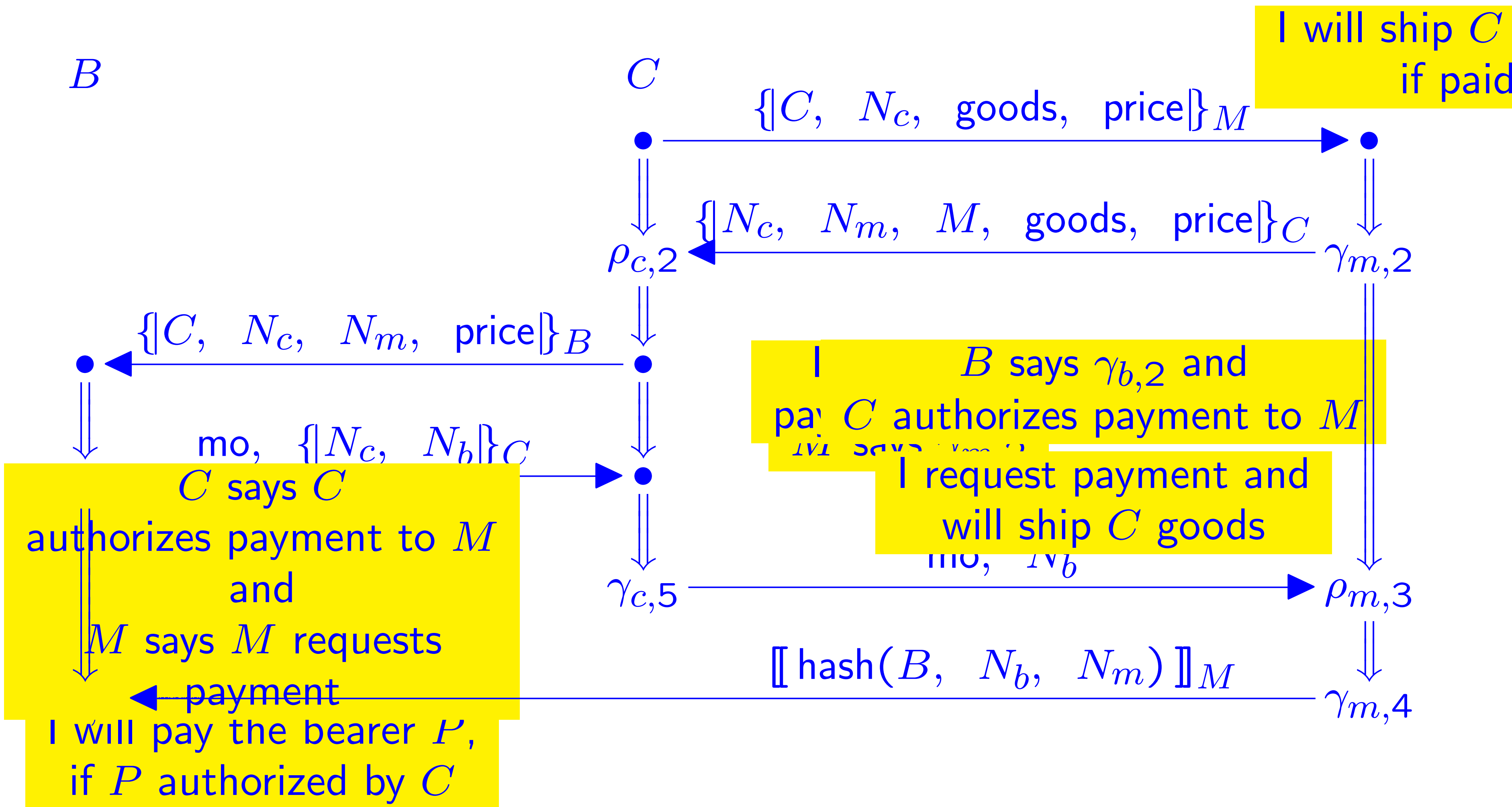
Our view: Formulas part of transmission/reception, not msg

- Compatible with many insights of earlier views
- Independent method to determine what events happened
- Clarity about who makes assertions, who infers consequences
- Partial information easy to handle
- Rigorous notion of soundness

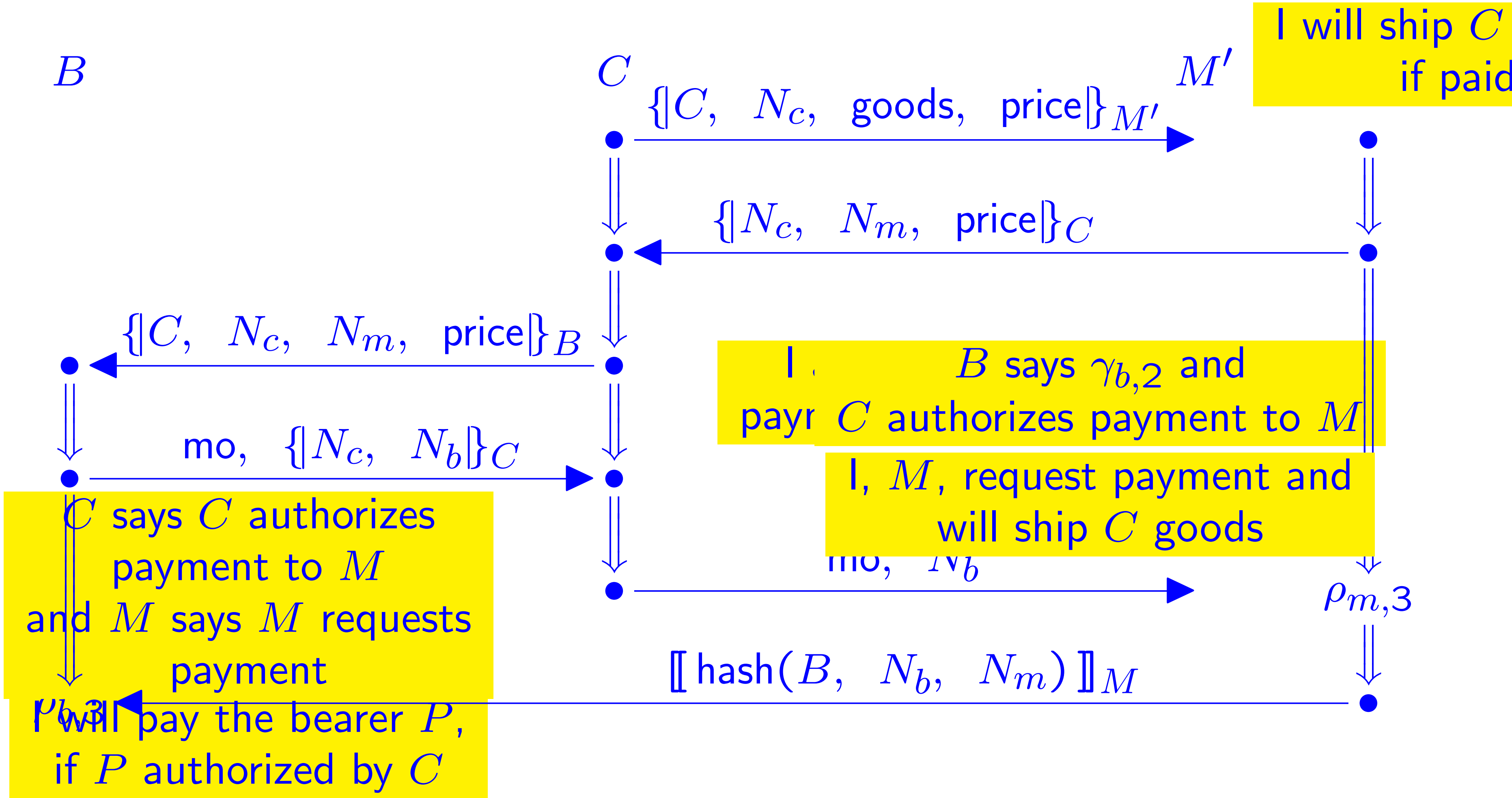
EPMO and Needham-Schroeder-Lowe



EPMO Weakened



Lowe-style attack



Soundness

Let Π be an annotated protocol, i.e.

- A set of roles (parametrized behaviors)
  - A role is a sequence of transmissions/receptions (nodes)
- For each transmission node n, a guarantee γ_n
- For each reception n, a rely formula ρ_n
- The principal active on node n is prin(n)

γ_n, ρ_n may refer to message ingredients

Π is sound if, for all executions $\mathcal{B}$, and message receptions $n \in \mathcal{B}$

$$\{\text{prin}(m) \text{ says } \gamma_m \colon m \prec_{\mathcal{B}} n\} \quad \longrightarrow_{\mathcal{L}} \quad \rho_n$$

where $\longrightarrow_{\mathcal{L}}$ is the consequence relation of the underlying logic

Soundness follows from authentication properties

- Authentication tests a good tool
- Recency easy to incorporate

One case of soundness

$$\rho_{m,3} = \begin{array}{l} B \text{ says } \gamma_{b,2} \\ \text{and } C \text{ says } \gamma_{c,5} \end{array}$$

Suppose $n_{m,3} \in \mathcal{B}$
where $m \in \text{Merchant}[B, C, M, p, g, N_c, N_m, N_b]$
necessary keys uncompromised, nonces u.o.

Then $n_{b,2}, n_{c,5} \in \mathcal{B}$ for some
$b \in \text{Bank}[B, C, *, p, N_c, N_m, N_b]$ and
$c \in \text{Customer}[B, C, M, p, g, N_c, N_m, N_b]$

Moreover, $n_{m,1} \preceq_{\mathcal{B}} n_{b,2}$ and $n_{m,1} \preceq_{\mathcal{B}} n_{c,5}$

Same form as an authentication result with recency

In weakened EPMO, only know

$$c \in \text{Customer}[B, C, X, p, g, N_c, N_m, N_b]$$

Four Tenets of Logical Trust Management

1. Principal theories: Each principal P holds a theory Th_P; P derives conclusions using Th_P
   - May rely on formulas P' says ψ as additional premises
   - P says ϕ only when P derives ϕ
2. Trust in others: "P trusts P' for a subject ψ" means
   - P says $((P'$ says $\psi) \supset \psi)$
3. Syntactic authority: Certain formulas, e.g.
   - P says ϕ
   - P authorizes ϕ

   are true whenever P utters them
4. Access control via deduction: P may control resource r; P takes action $\phi(r, P')$ on behalf of P' when P derives
   - P' requests $\phi(r, P')$
   - P' deserves $\phi(r, P')$

Trust and Protocols

Nonce-based, cryptographic protocols for real tasks:

- Rely on formula after message receipt
- Guard message transmissions by guarantee
- Stop if you fail to infer guard

Key technical idea: Soundness

- Annotated protocol is sound if (in every execution) each rely supported by earlier guarantees
- Strand space authentication tests establish soundness

Clean method to export pure properties of protocol
to support trust needs of real systems

Permissible Bundles

Let $\mathcal{B}$ a bundle; let each P hold theory Th_P

$\mathcal{B}$ is permissible if

$$\{\rho_m \colon m \Rightarrow^+ n\} \longrightarrow_{\mathsf{Th}_P} \gamma_n$$

for each positive,
regular $n \in \mathcal{B}$

Means, every principal derives guarantee before sending each message

- permissible is vertical (strand-by-strand)
- sound is horizontal (cross-strand)

What trust is needed in permissible bundles of a sound protocol?

For which P' and ψ must P accept

$$P \text{ says } ((P' \text{ says } \psi) \supset \psi)$$

Trust Mgt Reasoning for EPMO, 1: Bank

$\gamma_{b,2}$	$\forall P_M$	**if**	C authorizes transfer$(B, \text{price}, P_M, N_m)$,
		and	P_M requests transfer$(B, \text{price}, P_M, N_m)$,
		then	transfer$(B, \text{price}, P_M, N_m)$.
$\rho_{b,3}$			C says C authorizes transfer$(B, \text{price}, M, N_m)$,
		and	M says M requests transfer$(B, \text{price}, M, N_m)$.

Universal quantifier $\forall P_M$ expresses "payable to bearer"

After node $n_{b,3}$, B can deduce

$$\text{transfer}(B, \text{price}, P_M, N_m)$$

Uses syntactic authority (authorizes, requests) but not trust

Trust Mgt Reasoning for EPMO, 2: Merchant

$\gamma_{m,2}$	$\forall P_B$	**if**	transfer$(P_B, \text{price}, M, N_m)$,
		then	ship(M, goods, C).
$\rho_{m,3}$			B says $\gamma_{b,2}$,
		and	C says $\gamma_{c,5}$.
$\gamma_{m,4}$			M requests transfer$(B, \text{price}, M, N_m)$,
		and	ship(M, goods, C).

After node $n_{m,3}$, can M can deduce ship(M, goods, C)?
Yes, if M requests transfer and accepts

$$B \text{ says } \gamma_{b,2} \quad \text{implies} \quad \gamma_{b,2}$$

i.e. M trusts B to transfer the funds as promised

$\gamma_{b,2}$	$\forall P_M$	**if**	C authorizes transfer$(B, \text{price}, P_M, N_m)$,
		and	P_M requests transfer$(B, \text{price}, P_M, N_m)$,
		then	transfer$(B, \text{price}, P_M, N_m)$.

Trust Mgt Formulas for EPMO, 3: Customer

Customer:

$\rho_{c,2}$	M says $\gamma_{m,2}$.
$\rho_{c,4}$	B says $\gamma_{b,2}$.
$\gamma_{c,5}$	C authorizes transfer$(B, \text{price}, M, N_m)$.

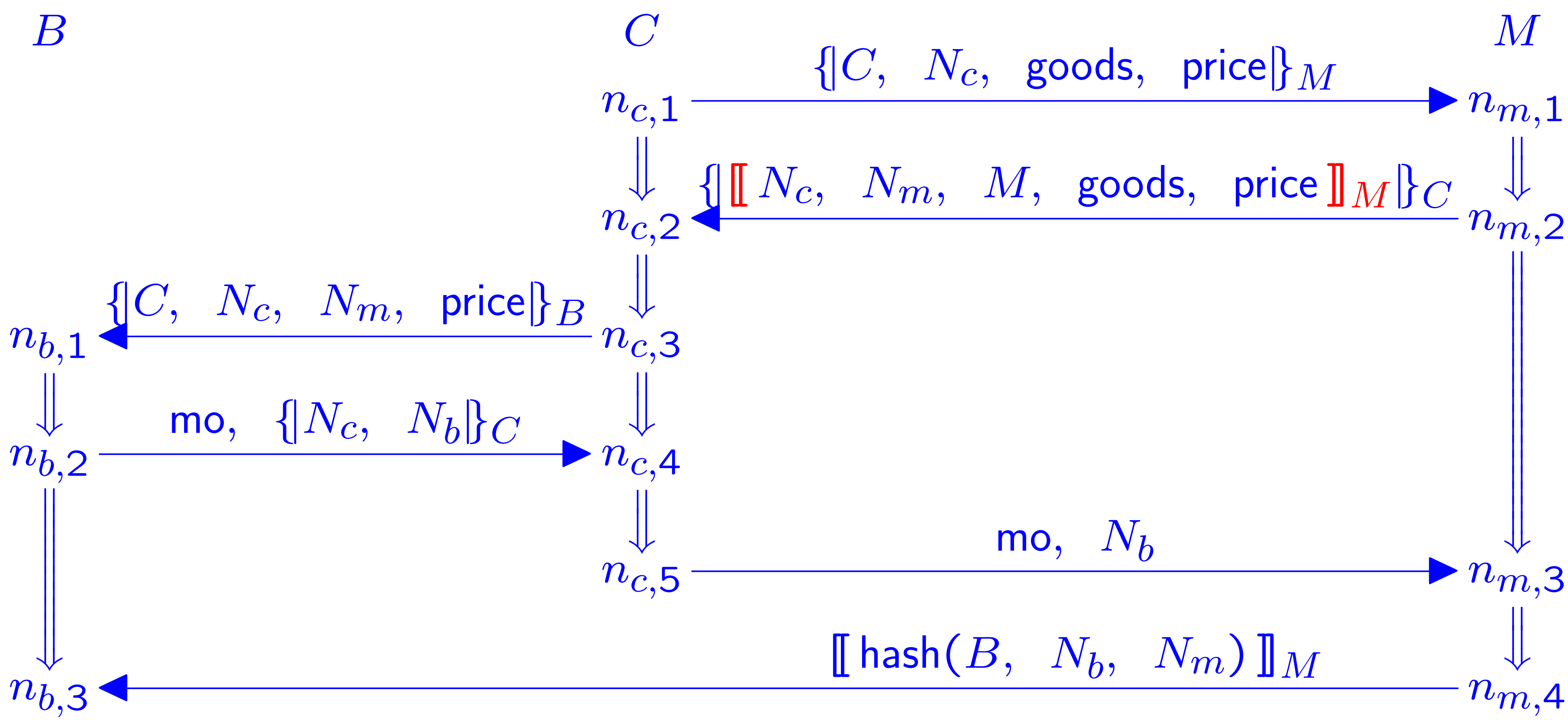
Decision to assert $\gamma_{c,5}$ depends on C's trust in M:

$$M \text{ says } \gamma_{m,2} \quad \text{implies} \quad \gamma_{m,2}$$

and C's trust in B:

$$B \text{ says } \gamma_{b,2} \quad \text{implies} \quad \gamma_{b,2}$$

A Signed Alternate: SEPMO



Signed Electronic Purchase using Money Order

$\mathsf{mo} = [\![\mathsf{hash}(C,\ N_c,\ N_b,\ N_m,\ \mathsf{price})]\!]_B$