Security in Computer Networks

Multilateral Security in Distributed and by Distributed Systems

Transparencies for the Last Part of the Lecture:

Security and Cryptography II

Andreas Pfitzmann

Technische Universität Dresden, Faculty of Computer Science, D-01062 Dresden
Nöthnitzer Str. 46, Room 3071
Phone: +49 351 463-38277, e-mail: pfitza@inf.tu-dresden.de, http://dud.inf.tu-dresden.de/

Protection of the recipient: Broadcast

A. Pfitzmann, M. Waidner 1985

Performance? more capable transmission system

Addressing (if possible: switch channels)
explicit addresses: routing
implicit addresses: attribute for the station of the addressee

invisible <==> encryption system
visible example: pseudo random number (generator), associative memory to detect

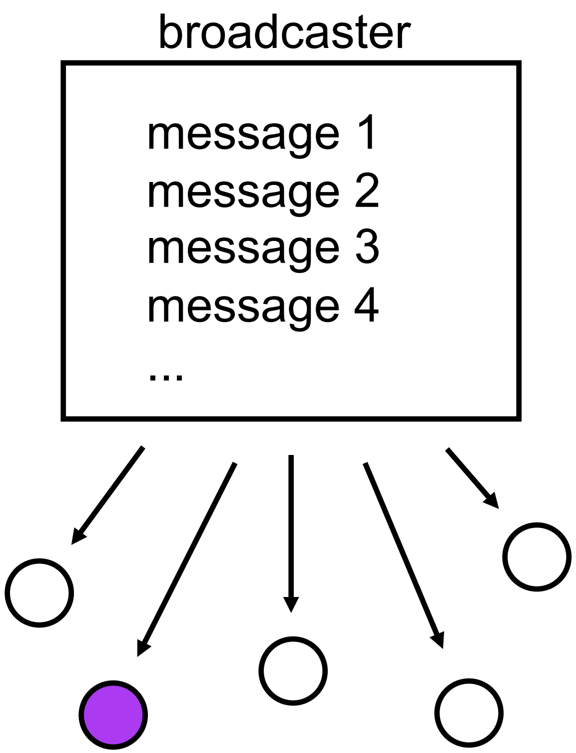
		address distribution	
		public address	private address
implicit address	invisible	very costly, but necessary to establish contact	costly
	visible	should not be used	change after use

Equivalence of Encryption Systems and Implicit Addressing

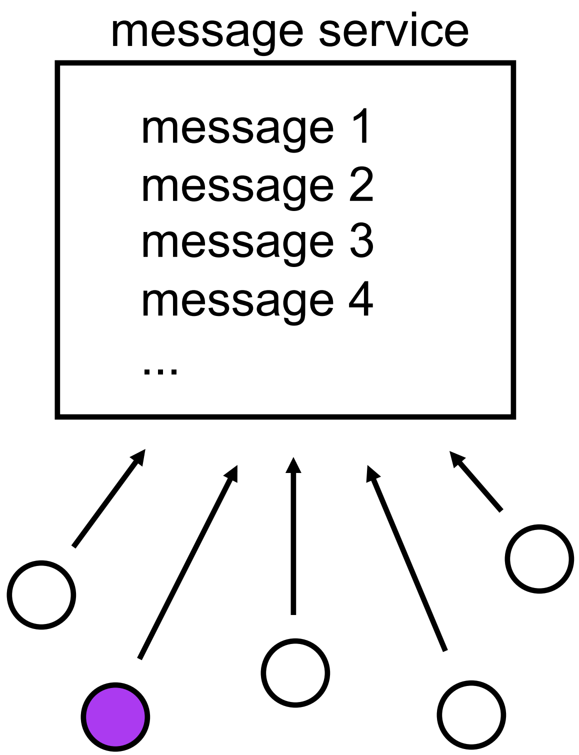
invisible public address <==> asymmetric encryption system

invisible private address <==> symmetric encryption system

Broadcast vs. Queries

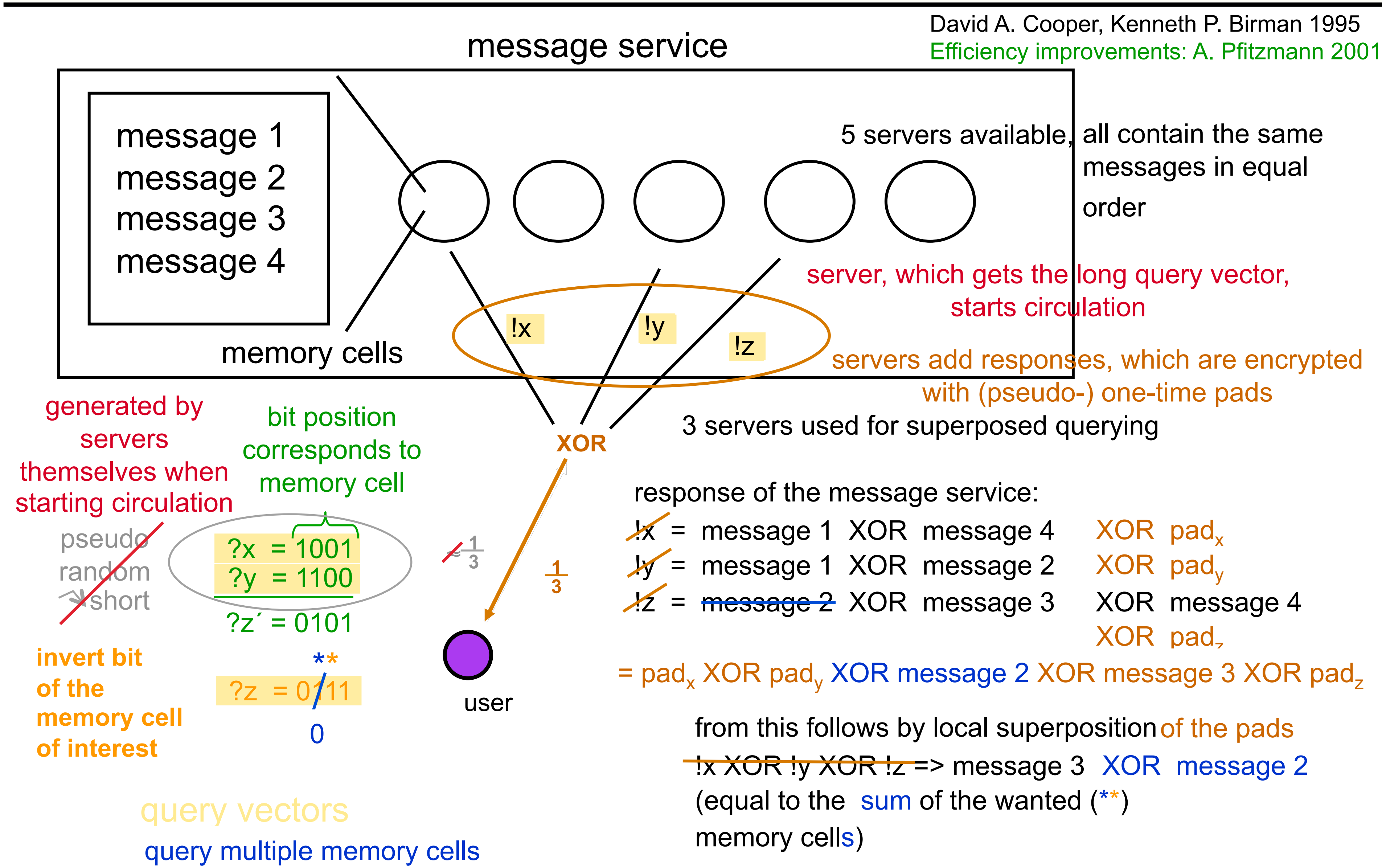


broadcast of separate messages to all recipients



everybody can query all messages

Example for message service



"Query and superpose" instead of "broadcast"

re-writable memory cell = implicit address

re-writing = addition mod 2 (enables to read many cells in one step)

channels trivially realizable

Purposes of implicit addresses

- Broadcast: Efficiency (evaluation of implicit address should be faster than processing the whole message)
- Query and superpose: Medium Access Control; Efficiency (should reduce number of messages to be read)

fixed memory cell = visible implicit address

implementation: fixed query vectors for servers $0 \nearrow \swarrow 1$

Number of addresses *linear* in the expense (of superposing).

Improvement: *Set* of re-writable memory cells = implicit address

Message *m* is stored in a set of *a* memory cells by choosing *a*–1 values randomly and choosing the value of the a^{th} cell such that the sum of all *a* cells is *m*.

For overall *n* memory cells, there are now 2^n–1 usable implicit addresses, but due to overlaps of them, they cannot be used independently.

If collisions occur due to overlap, try retransmit after randomly chosen time intervals.

Any set of cells as well as any set of sets of cells can be queried in one step.

Invisible implicit addresses using "query and superpose" (1)

hopping between memory cells = invisible implicit address

Idea: User who wants to use invisible implicit address at time *t* reads the values from reserved memory cells at time *t*-1. These values identify the memory cell to be used at time *t*.

Impl.:
- Address owner gives each server *s* a PBG_s.
- Each server *s* replaces at each time step *t* the content of its reserved memory cell S_{Adr} with $PBG_s(t)$:

$$S_{Adr} := PBG_s\,(t)$$

- User queries via MIXes $\sum_s PBG_s(t)$. (possible in one step.) user employs $S_{\sum_s PBG_s(t)}$ for message. 1 ↗ ↙ 1
- Address owner generates $\sum_s PBG_s(t)$ and reads using "query and superpose" $S_{\sum_s PBG_s(t)}$ before and after the writing of messages, calculates difference.

  Improvement: for all his invisible implicit addresses together: 1 ↗ ↙ 2 (if ≤ 1 msg)

Address is in so far invisible, that at each point of time only a very little fraction of all possible combinations of the cells S_{Adr} are readable.

Invisible implicit addresses using “query and superpose” (2)

hopping between memory cells = invisible implicit address

can be extended to

hopping between *sets of* memory cells = invisible implicit address

Fault tolerance (and countering modifying attacks)

What if server (intentionally) does

1. **not respond or**
2. **delivers wrong response?**

1. **Submit the same query vector to another server.**
2. **Messages should be authenticated so the user can check their integrity and thereby detect whether at least one server did deliver a wrong response. If so, use a disjoint set of servers or lay traps by sending the same query vector to many servers and checking their responses by comparison.**

Protection of the sender

Dummy messages

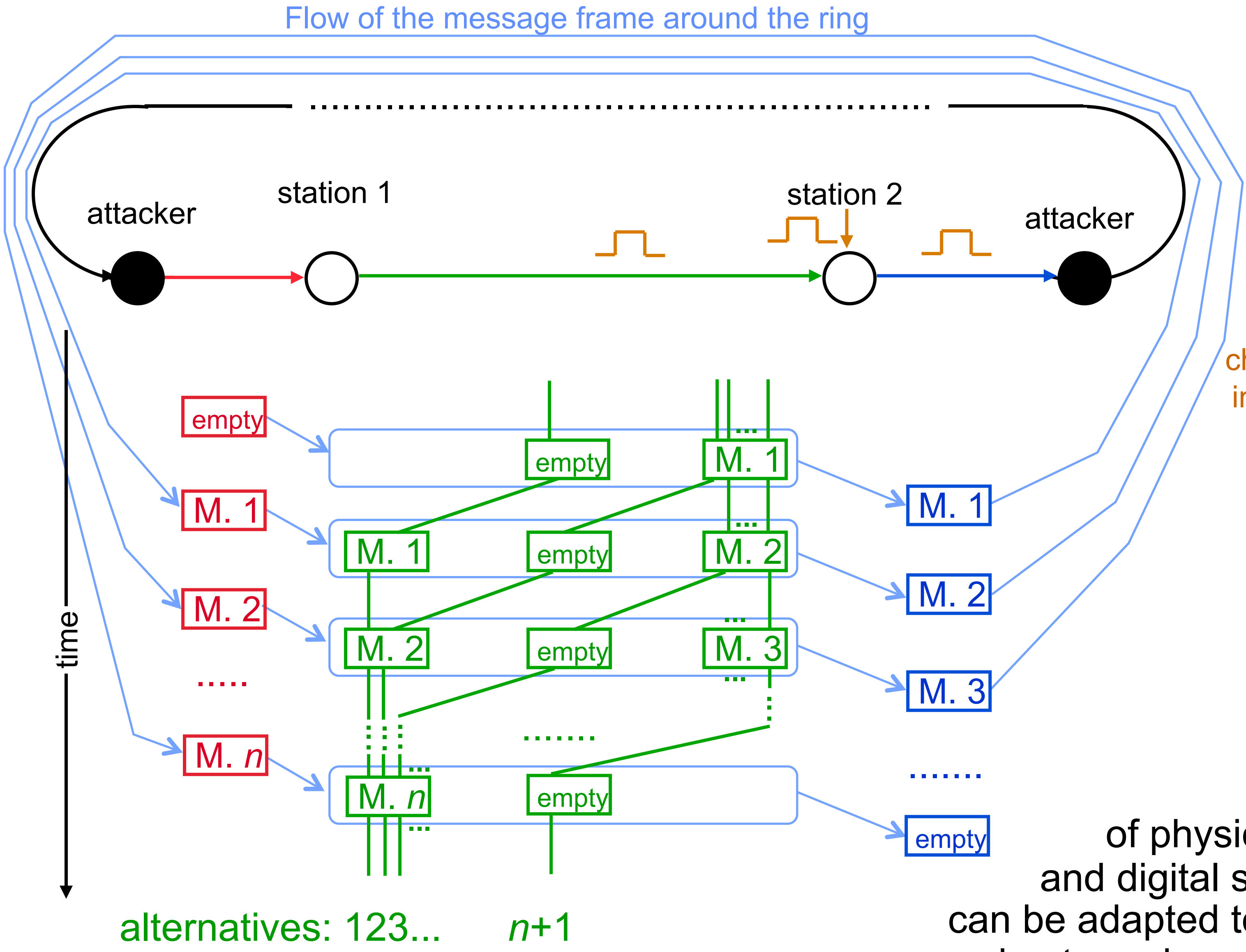
- don’t protect against addressee of meaningful messages
- make the protection of the recipient more inefficient

Unobservability of neighboring lines and stations as well as digital signal regeneration

example: RING-network

Proof of anonymity for a RING access method

A. Pfitzmann 1983 - 1985



Digital signal regeneration:

The analogue characteristics of bits are independent of their true sender.

The idea
of physical unobservability
and digital signal regeneration
can be adapted to other topologies,
i.e. tree-shaped CATV networks;

It reappears in another context in Crowds

Fault tolerance of the RING-network

Requirement

For each possible error, anonymity has to be guaranteed.

Problem

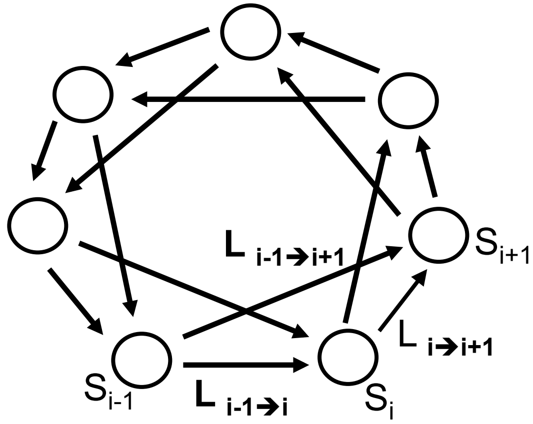
Anonymity: little global information
Fault tolerance: much global information

Principles

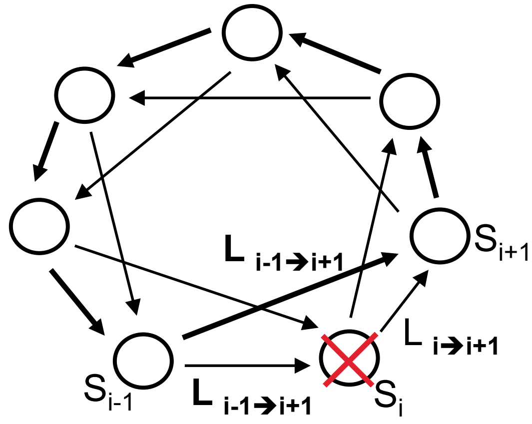
Fault tolerance through weaker anonymity in a single operational mode (anonymity-mode)

Fault tolerance through a special operational mode (fault tolerance-mode)

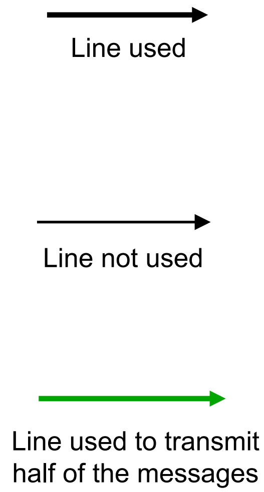
Braided RING

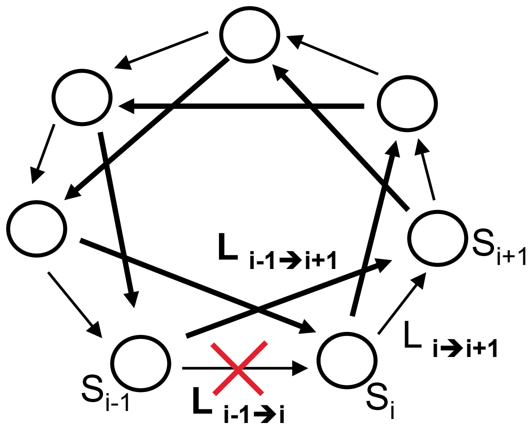


Two RINGs operating if no faults

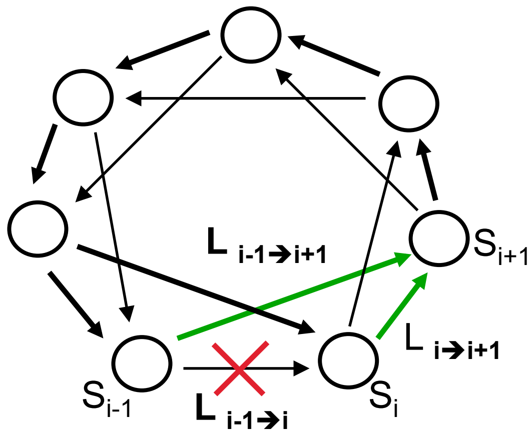


Reconfiguration of the outer RING if a station fails





Reconfiguration of the inner RING if an outer line fails



Reconfiguration of the outer RING if an outer line fails

Modifying attacks

modifying attacks at

covered in RING-network by attacker model

sender anonymity

→ extend the access method

recipient anonymity

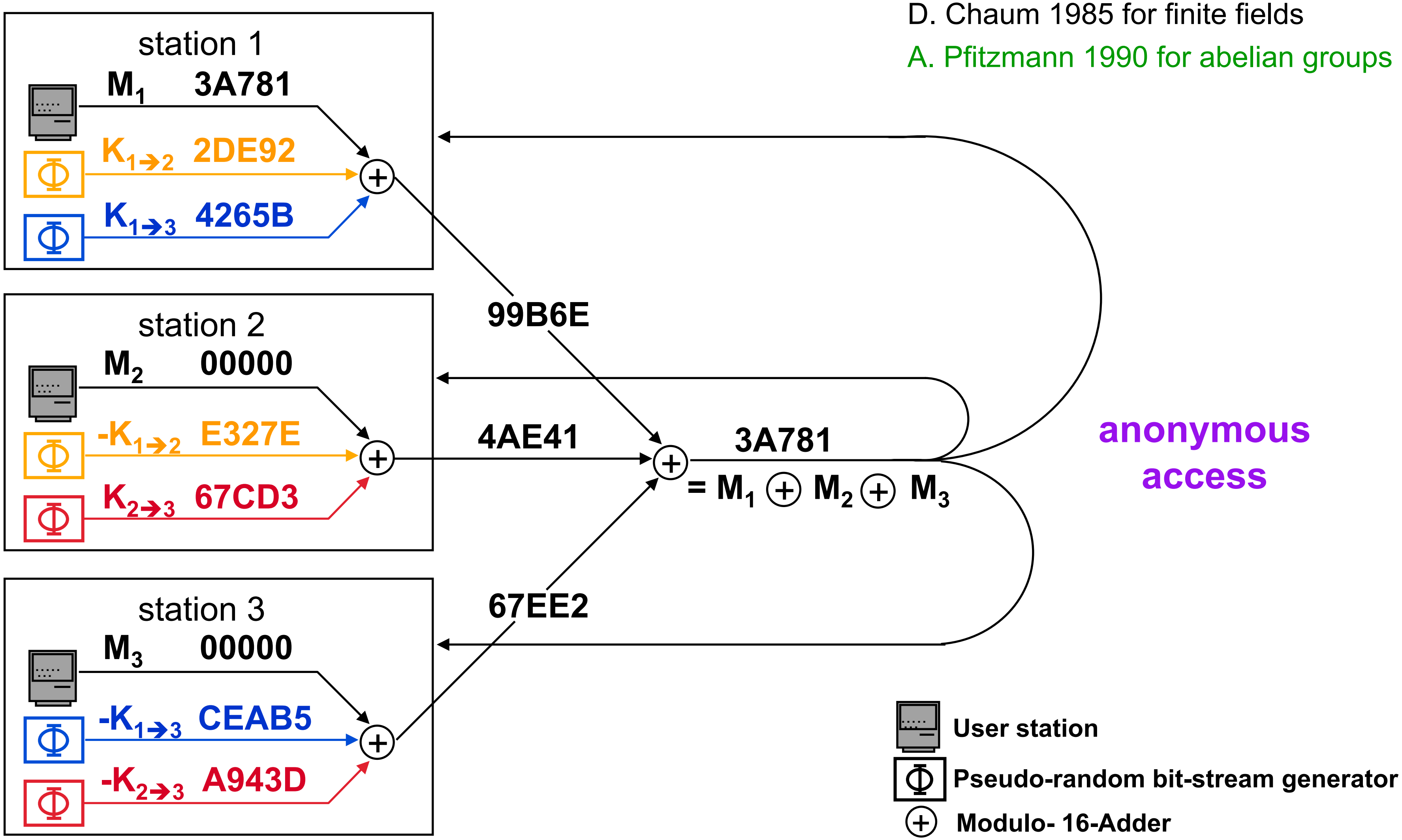
service delivery

publish input and output
if dispute: reconfiguration

Superposed sending (DC-network)

D. Chaum 1985 for finite fields

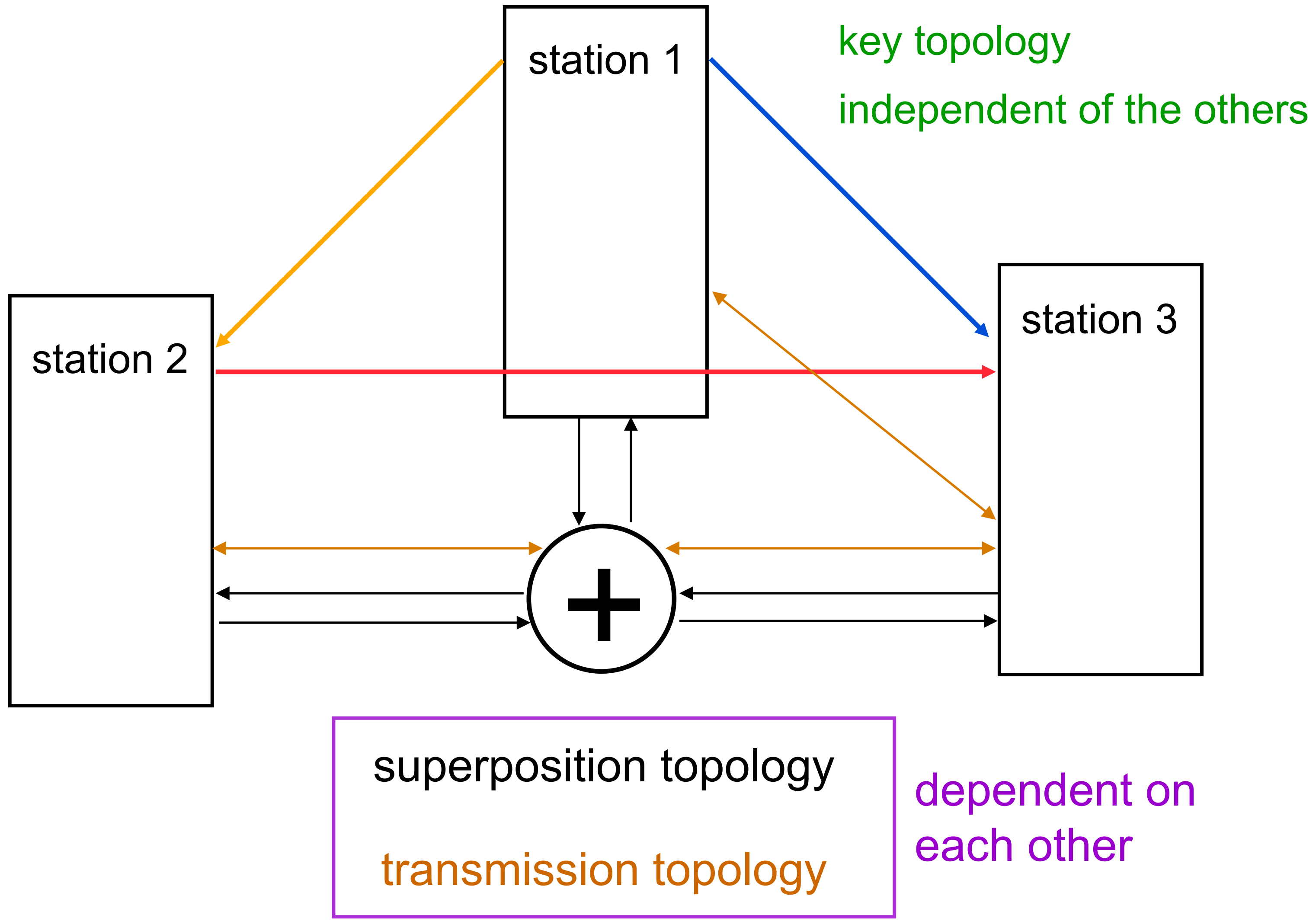
A. Pfitzmann 1990 for abelian groups



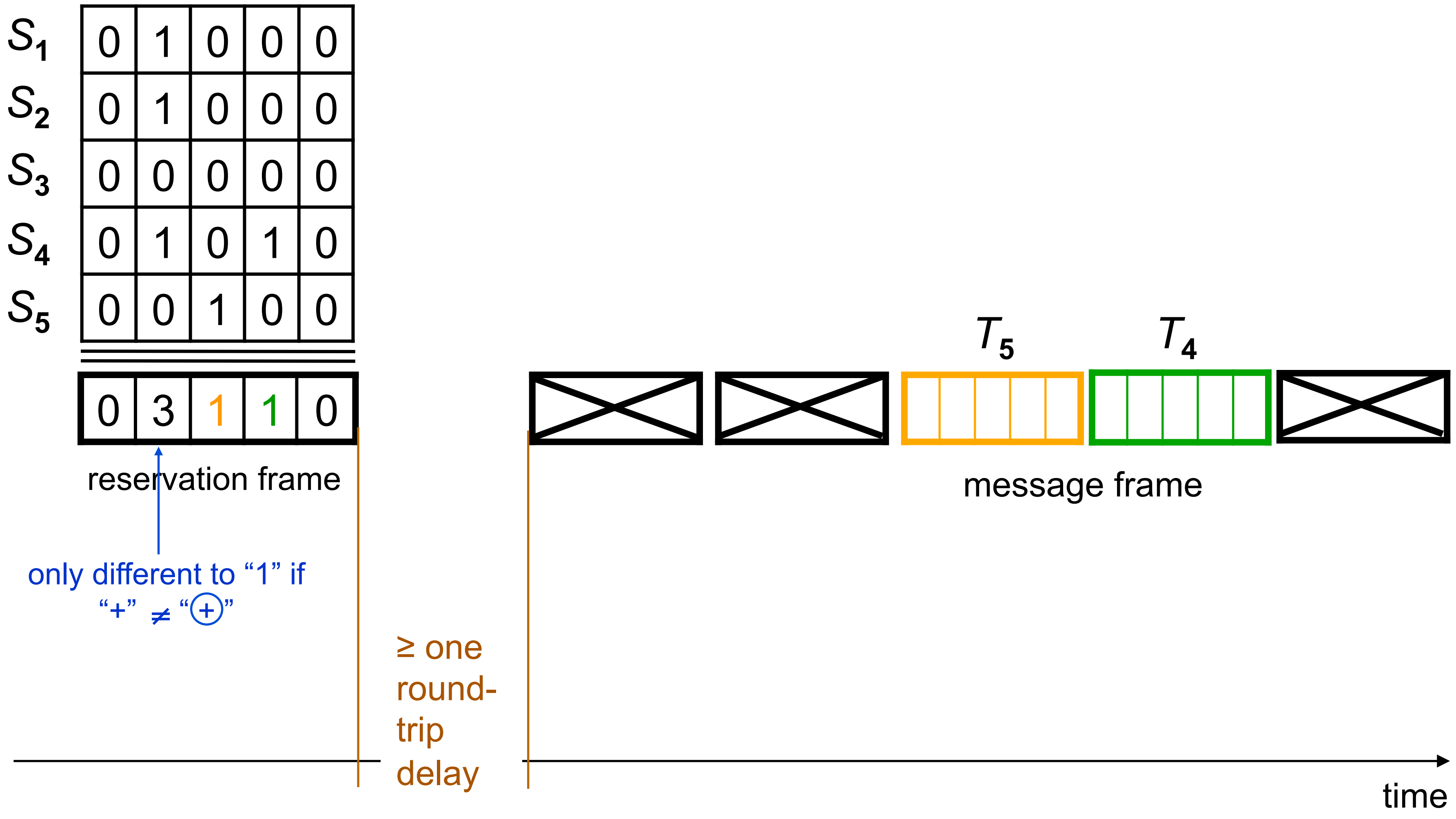
Anonymity of the sender

If stations are connected by keys the value of which is completely unknown to the attacker, tapping all lines does not give him any information about the sender.

Three distinct topologies



Reservation scheme



Superposed receiving

Whoever knows the sum of *n* characters and *n*-1 of these *n* characters, can calculate the *n*-th character.

pairwise superposed receiving (reservation scheme: n=2)

Two stations send simultaneously.
Each subtracts their characters from the sum to receive the character sent by the other station.
==> Duplex channel in the bandwidth of a simplex channel

global superposed receiving (direct transmission: $n \geq 2$)

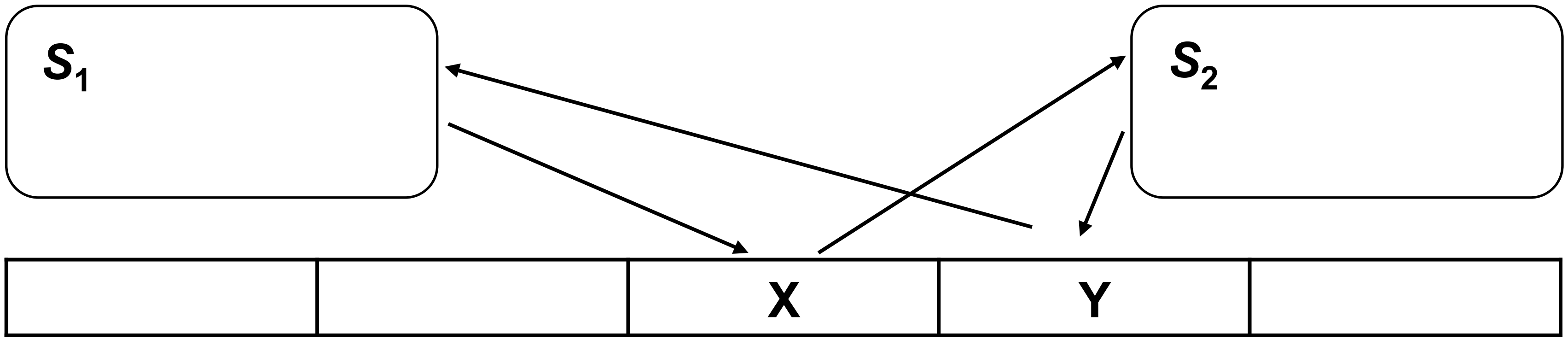
Result of a collision is stored, so that if *n* messages collide, only *n*-1 have to be sent again.

Collision resolution algorithm using the mean of messages:

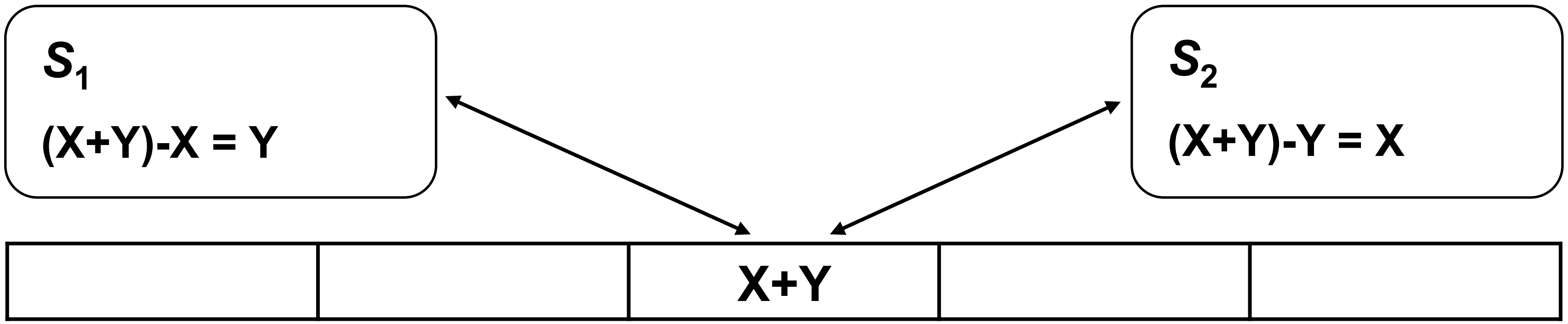
$\leq 2^S - 1$ station

addition mod 2^L

Pairwise superposed receiving

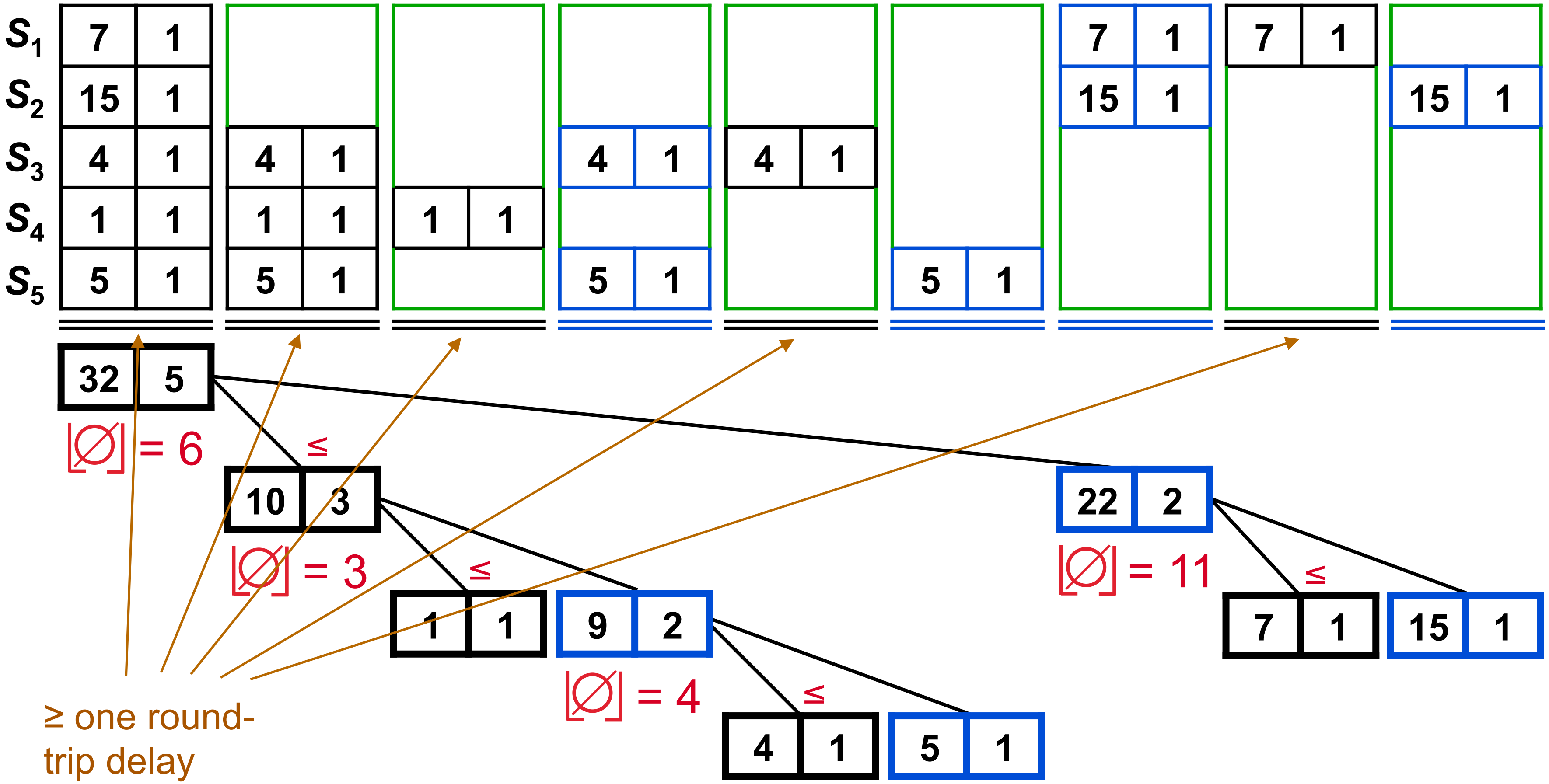


without superposed receiving



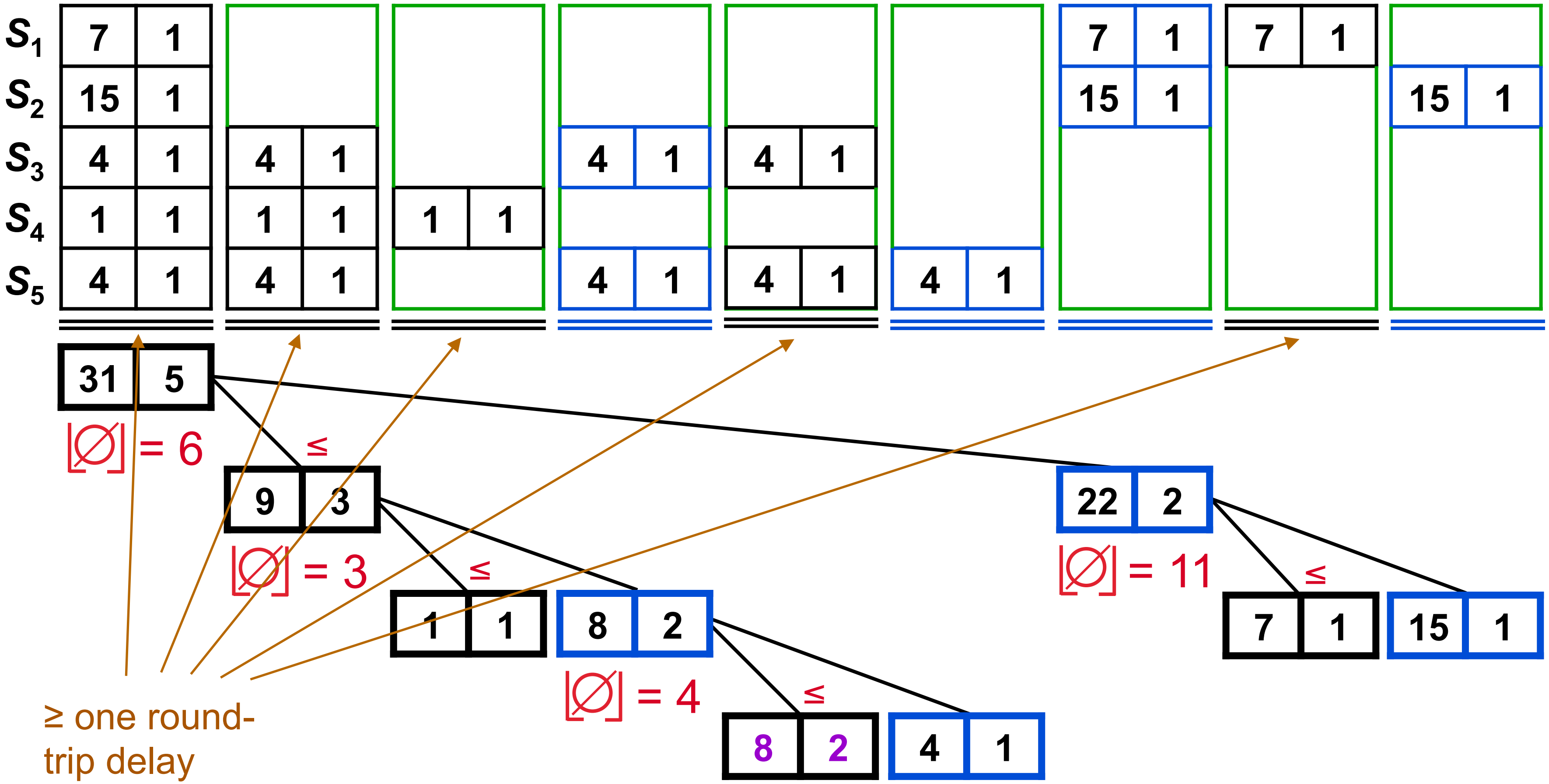
with pairwise superposed receiving

Global superposed receiving



Collision resolution algorithm with mean calculation and superposed receiving

Global superposed receiving (2 messages equal)

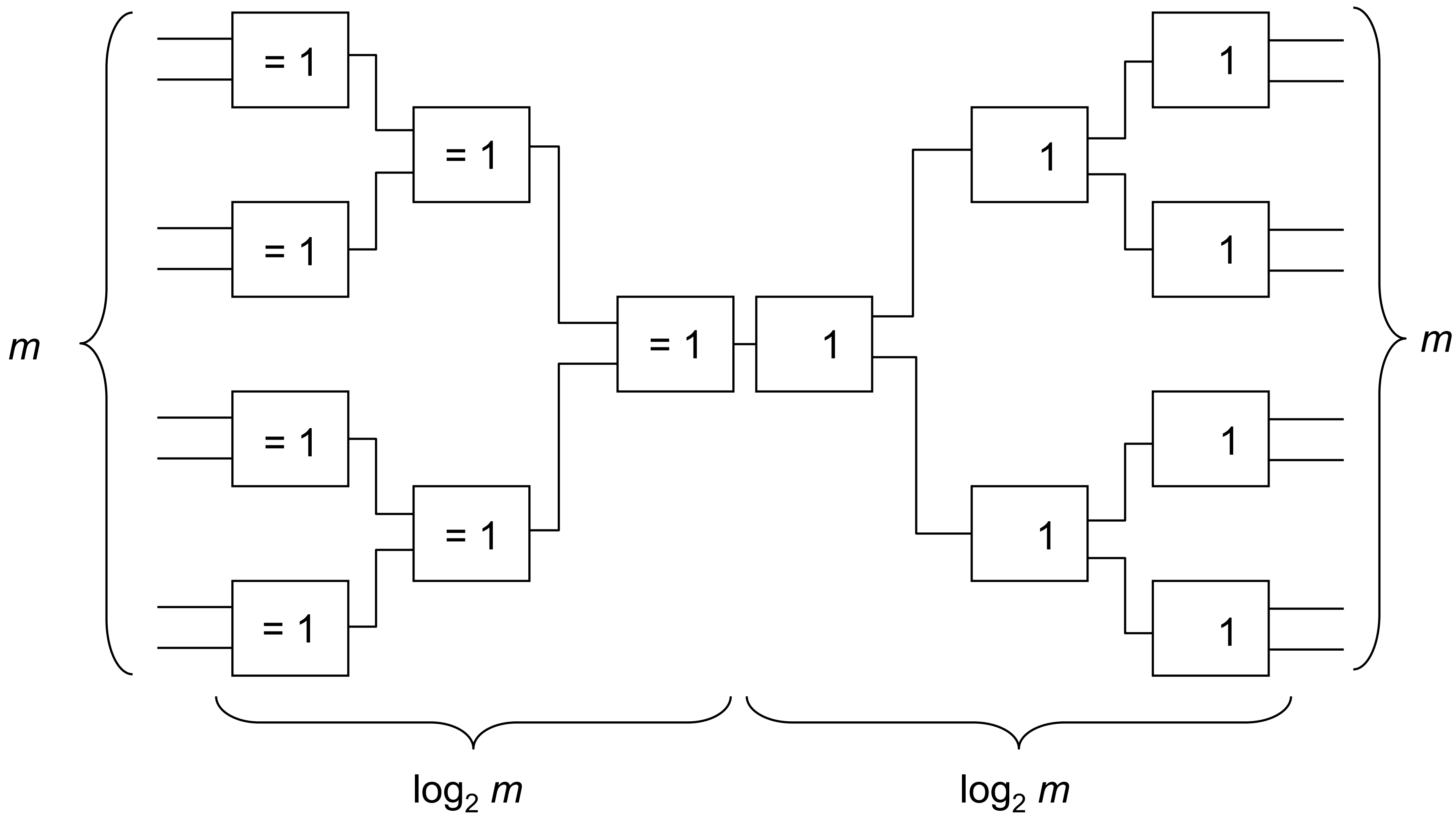


Collision resolution algorithm with mean calculation and superposed receiving

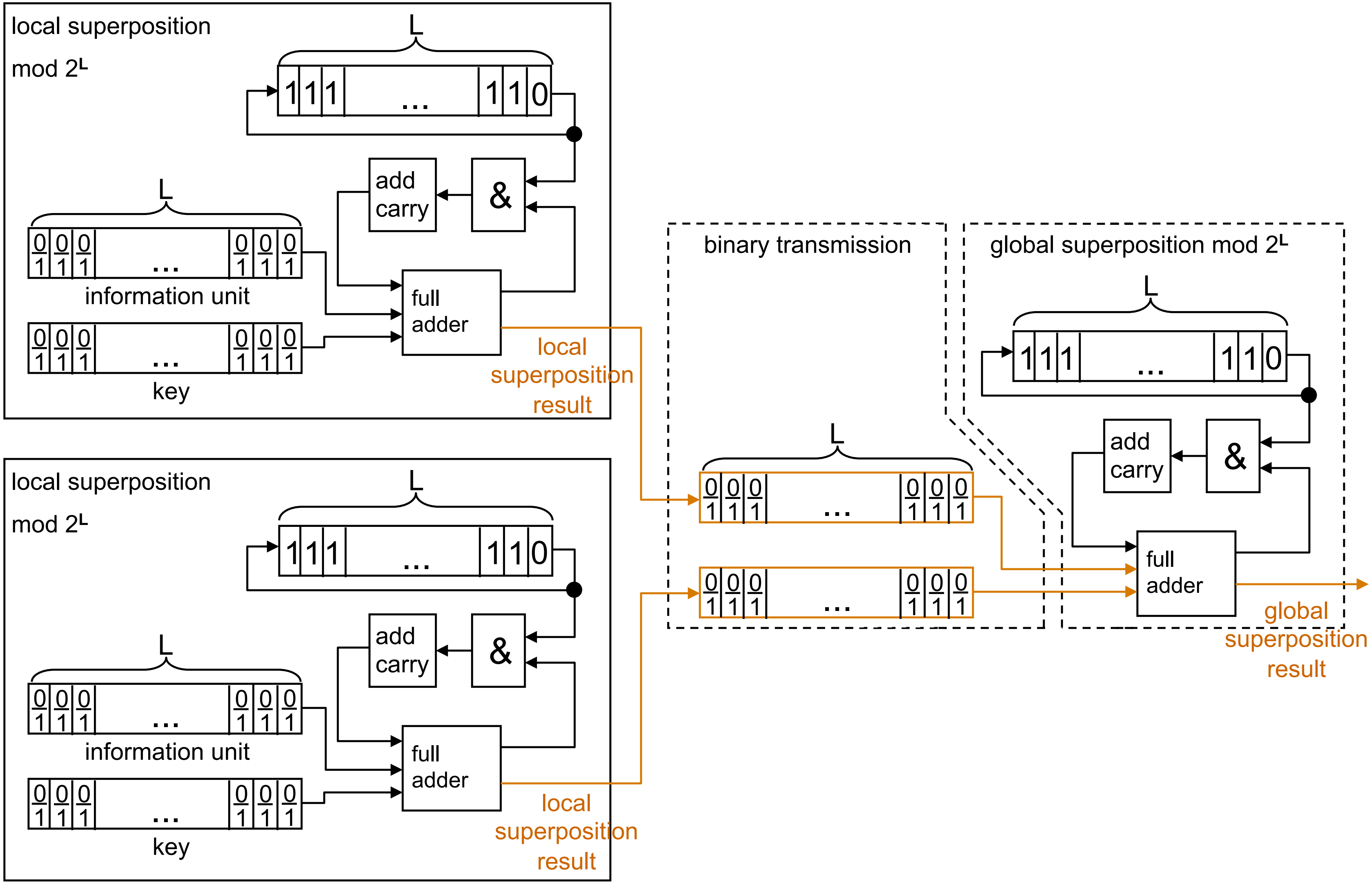
Superposition topology for minimal delay

tree of XOR gates to superpose the output of the user stations

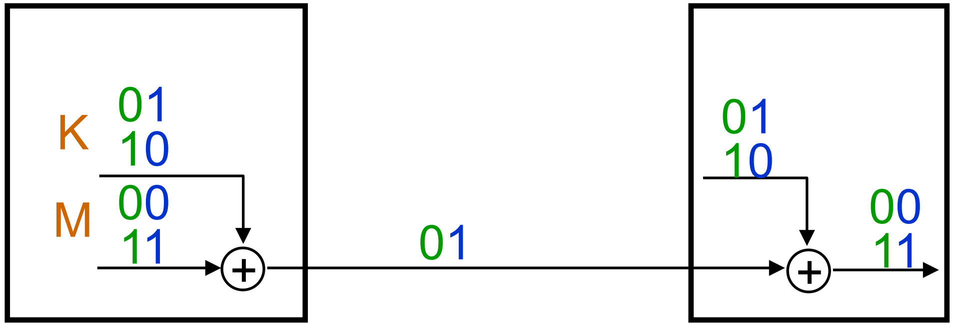
tree of repeaters to amplify the output to the user stations



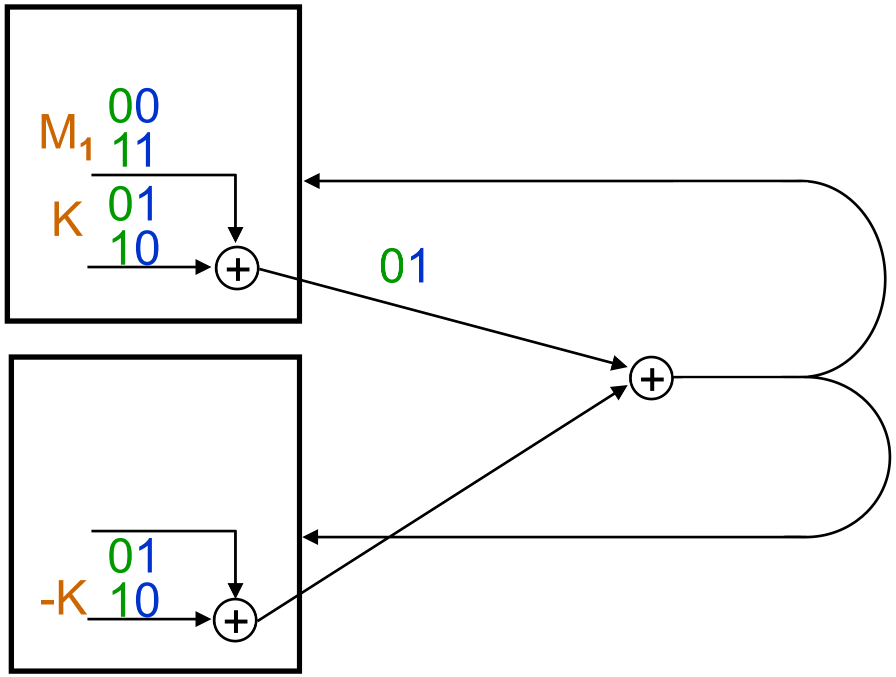
Suitable coding for superposed sending



Analogy between Vernam cipher and superposed sending



$K + M = C \Leftrightarrow M = C - K$ abelian group



$M_1 + K = O_1$

$M_2 - K = O_2$

Proof of sender anonymity: proposition and start of induction

Proposition:

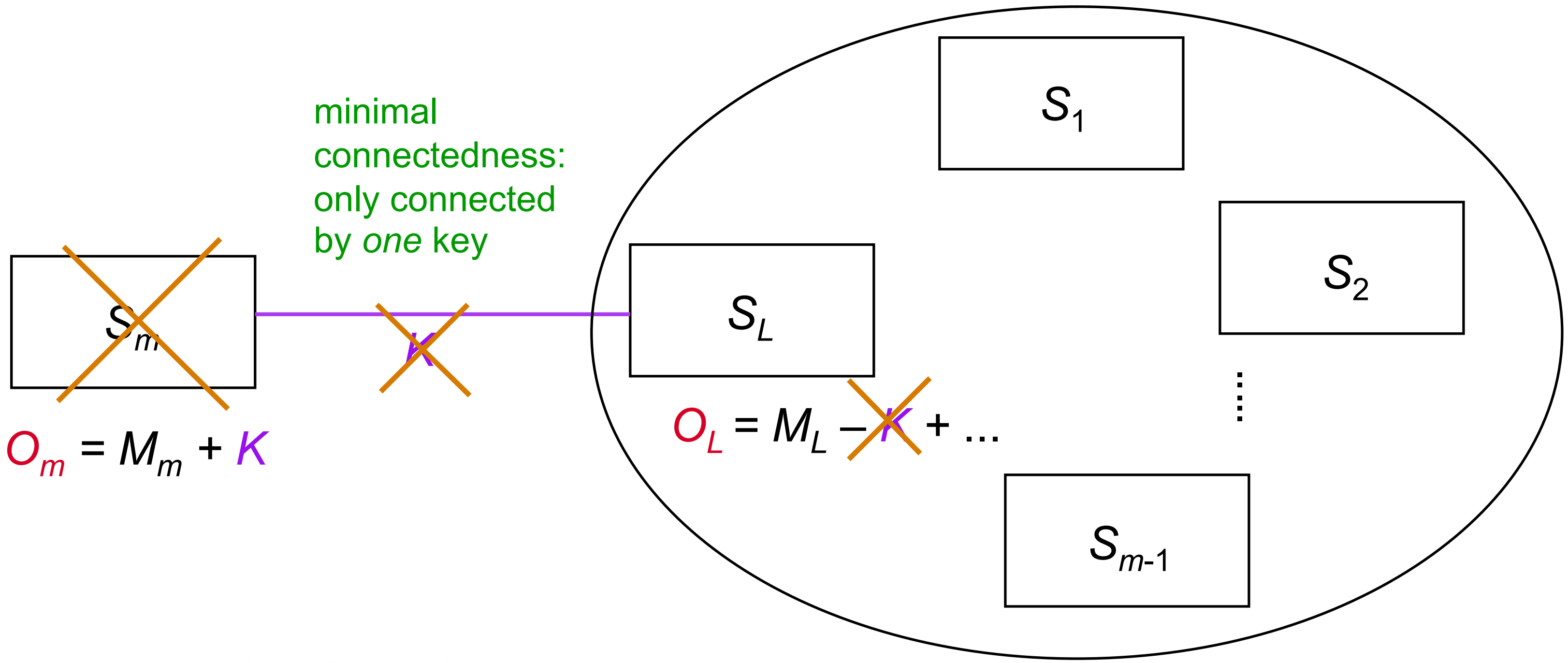
If stations S_i are connected by uniform randomly distributed keys K_j which are unknown to the attacker , by observing all the O_i , the attacker only finds out $\sum_i M_i$ about the M_i.

Proof:

m=1, trivial

step m-1 ➔ m

Proof of sender anonymity: induction step



Attacker observes O_1, O_2, ...O_m.

For each combination of messages M'_1, M'_2, ... M'_m

with $\sum_{i=1}^{m} M'_i = \sum_{i=1}^{m} O_i$ there is exactly one compatible combination of keys : $K' := O_m - M'_m$

The other keys are defined as in the induction assumption, where the output of S_L is taken as $O_L + K'$.

Information-theoretic anonymity in spite of modifying attacks

Problems:

1) The attacker sends messages only to some users. If he gets an answer, the addressee was among these users.
2) To be able to punish a modifying attack at service delivery, corrupted messages have to be investigated. But this may *not* apply to meaningful messages of users truthful to the protocol.

DC⁺-net to protect the recipient even against modifying attacks: if broadcast error then uniformly distributed modification of keys

key between station i and j at time t

at station i at time t broadcast character

(Schief-) field

$$K_{ij}^{t} = a_{ij}^{t} + \sum_{k=t-s}^{t-1} b_{ij}^{t-k} \bullet C_i^k$$

For practical reasons:
Each station has to send within each s successive points in time a random message and observe, whether the broadcast is "correct".

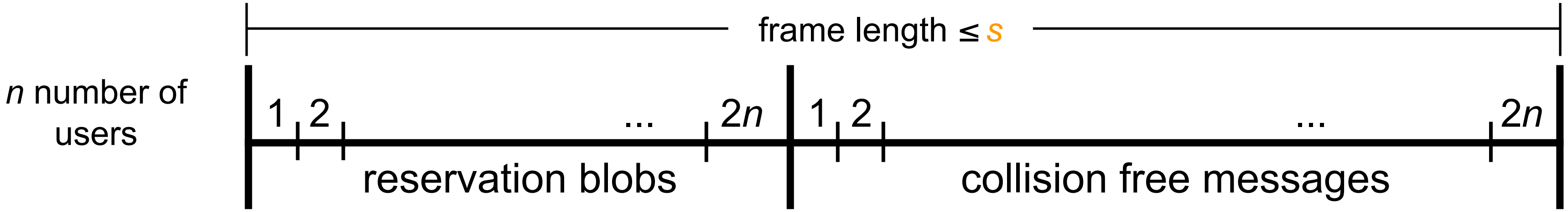
If K_{ij}^{t} is revealed, one will start with C_i^{t-s},..., C_i^{t-1}.

If disput then stop revealing. If revealed, distribute new b_{ij}^{1},..., b_{ij}^{s} .

Let t-s be the first point in time where $V_i^{t-s} \neq V_j^{t-s}$.

$$\begin{pmatrix} K_{ij}^{t+1-s} - K_{ji}^{t+1-s} \\ K_{ij}^{t+2-s} - K_{ji}^{t+2-s} \\ \vdots \\ K_{ij}^{t} - K_{ji}^{t} \end{pmatrix} = \begin{pmatrix} C_i^{t-s} - C_j^{t-s} & 0 & \cdots & 0 \\ C_i^{t+1-s} - C_j^{t+1-s} & C_i^{t-s} - C_j^{t-s} & & 0 \\ & & \ddots & \vdots \\ C_i^{t-1} - C_j^{t-1} & C_i^{t-2} - C_j^{t-2} & \cdots & C_i^{t-s} - C_j^{t-s} \end{pmatrix} \begin{pmatrix} b_{ij}^1 \\ b_{ij}^2 \\ \vdots \\ b_{ij}^s \end{pmatrix}$$

Protection of the sender: anonymous trap protocol



- Each user can cause investigating the reservation blobs directly after their sending if the sending of his reservation blobs did not work.
- Each user can authorize investigating of his “collision-free” random message, by opening the corresponding reservation blob.

Blob := committing to 0 or 1, without revealing the value committed to

1) The user committing the value must not be able to change it, but he must be able to reveal it.

2) The others should not get any information about the value.

In a "digital" world you can get exactly one property without assumptions, the other then requires $\underbrace{\text{a complexity-theoretic assumption}}$.

Example:

Given a prime number p and the prime factors of p -1, as well as a generator α of Z^*_p (multiplicative group mod p). Using y everybody can calculate α^y mod p.

The inverse can not be done efficiently!

1?

$s \in Z^*_p$ randomly chosen

(so user cannot compute e such that $s \equiv \alpha^e$)

$x := s^b \alpha^y \bmod p$ with $0 \le y \le p$-2

commit $\xrightarrow{x}$

open $\xrightarrow{y}$

2?

Let 2^u be the smallest number that does not divide p -1

$y := y_1, b, y_2$ with $0 \le y \le p$-2 and $|y_2| = u$ -1

$x := \alpha^y \bmod p$

commit $\xrightarrow{x}$

open $\xrightarrow{y}$

Blobs based on factoring assumption

1?

$n := p \bullet q$

$s := t^2 \bmod n$

$\xleftarrow{n,\ s}$

$\xleftrightarrow{s \in QR_n}$

commit

$x := y^2\, s^b \bmod n$

$\xrightarrow{x}$

open

$\xrightarrow{y}$

2?

$n := p \bullet q$

$s \notin QR_n,\ \left(\frac{s}{n}\right) = 1$

$\xrightarrow{n,\ s}$

$\xleftrightarrow{n = p \bullet q,\ s \notin QR_n}$

$x := y^2\, s^b \bmod n$

$\xrightarrow{x}$

$\xrightarrow{y}$

Blobs based on asymmetric encryption system

2?

encrypt *b* with asymmetric encryption system (recall: public encryption key and ciphertext together uniquely determine the plaintext)

- has to be probabilistic – otherwise trying all possible values is easy
- communicating the random number used to probabilistically encrypt *b* means opening the blob
- computationally unrestricted attackers can calculate *b* (since they can break any asymmetric encryption system anyway)

Modifying attacks

Modifying attacks at

sender anonymity

recipient anonymity

service delivery

attacker sends message character $\neq 0$,

if the others send their message character as well

➔ no transmission of meaningful information

To be able to punish a modifying attack at service delivery, corrupted messages have to be investigated. But this may *not* apply to meaningful messages of users truthful to the protocol.

Checking the behavior of the stations

To check a station it has to be known:

- All keys with others
- The output of the station
- All the global superposing results received by the station
- At what time the station may send message characters according to the access protocol
  (Can be determined using the global superposition results of the last rounds; These results can be calculated using the outputs of all stations.)

-
-
-

calculated
message characters

compare

-

known = known to *all* stations truthful to the protocol

Modifying attacks in the reservation phase

Collisions in the reservation phase

- cannot be avoided completely
- therefore they *must not* be treated as attack

Problem: Attacker *A* could await the output of the users truthful to the protocol and than *A* could choose his own message so that a collision is generated.

Solution: Each station

1. defines its output using a Blob at first, then
2. awaits the Blobs of all other stations, and finally
3. reveals its own Blob's content.

Fault tolerance: 2 modes of operation

A-mode

anonymous transmission of messages using superposed sending

F-mode

sender and recipient are not protected

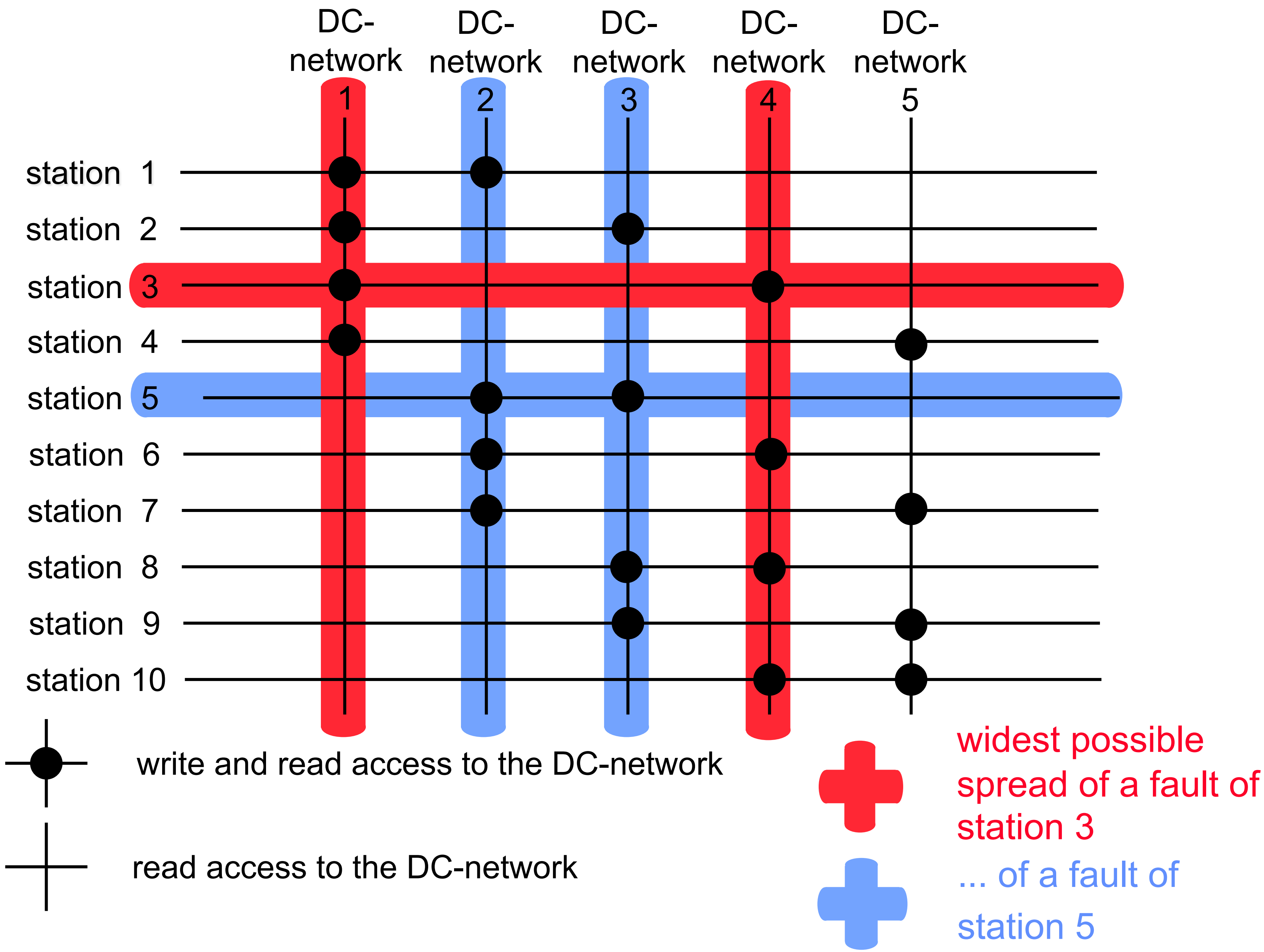
fault detection

fault localization

taking defective components out of operation

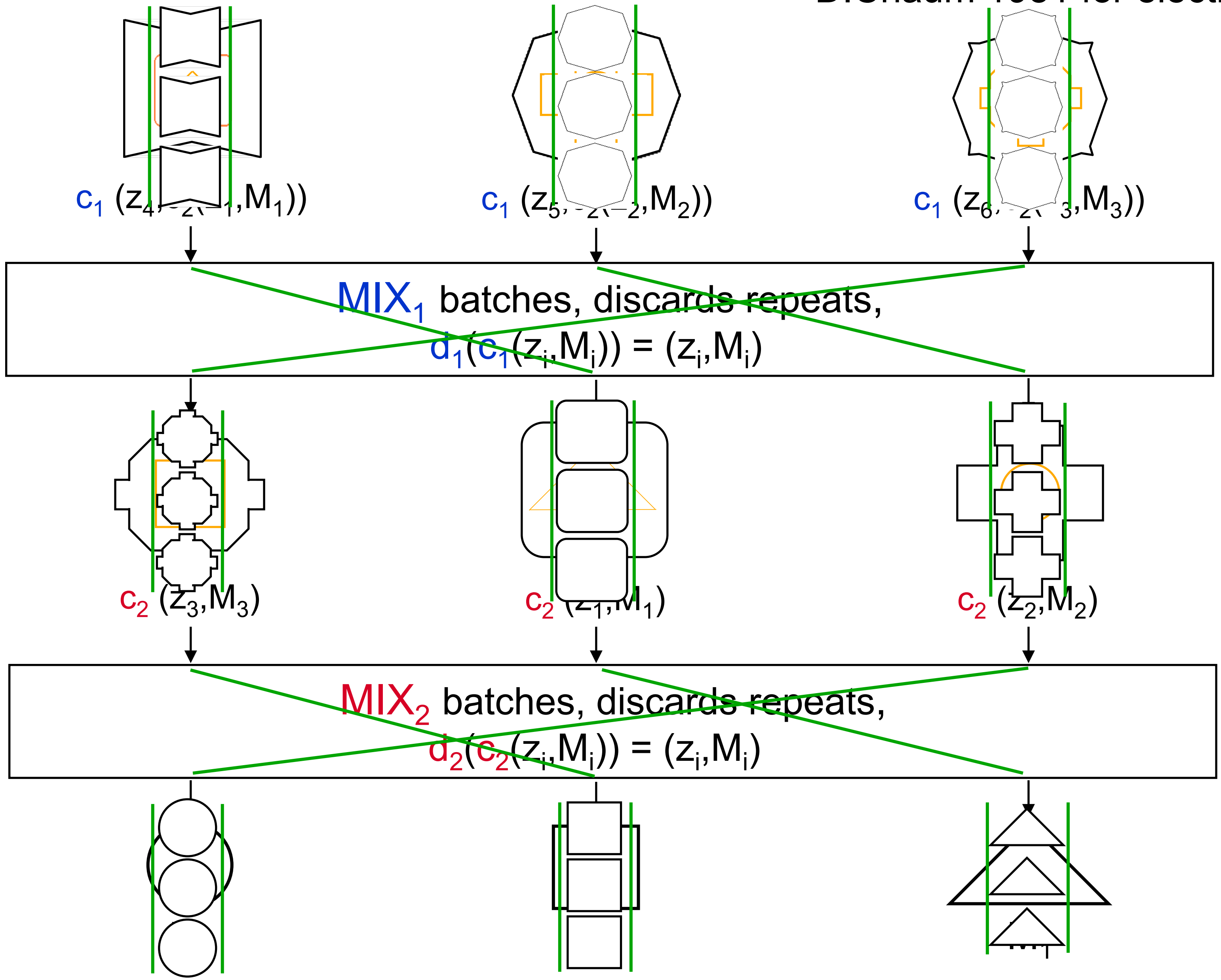
error recovery of the PRGs, initialization of the access protocol

Fault tolerance: sender-partitioned DC-network

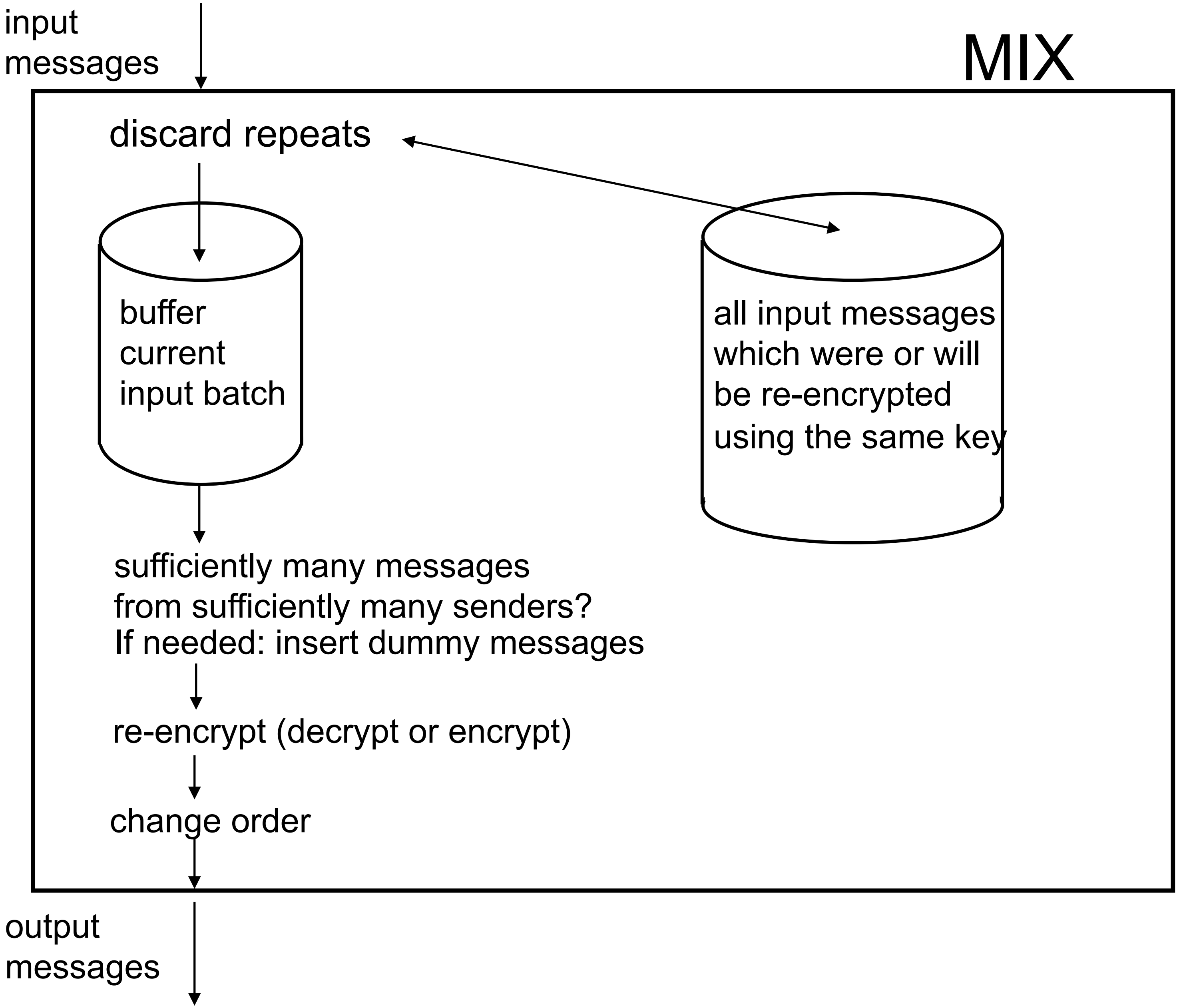


Protection of the communication relation: MIX-network

D.Chaum 1981 for electronic mail



Basic functions of a MIX



min	max
1 HDD access	
10 ms	50 ms
do nothing	test dig. sig.
0 ms	100 ms
asym. encr. special HW	asym. encr. SW
1 ms	100 ms
1 ns	10 µs
11,000001 ms	250,01 ms

Properties of MIXes

MIXes should be designed independently
produced
operated
maintained ...

Messages of the same length

buffer
re-encrypt
change order

batch-wise

Each message processed only once!

inside each batch
between the batches

sym. encryption system only for

first
last

MIX

asym. encryption system required

for MIXes in the middle

Possibilities and limits of re-encryption

Aim: (without dummy traffic)

Communication relation can be revealed only by:

- *all* other senders and recipients together or
- *all* MIXes together which were passed through

against the will of the sender or the recipient.

Conclusions:

1. Re-encryption: never decryption directly after encryption

   Reason: to decrypt the encryption the corresponding key is needed;

   ➔ before and after the encoding of the message it is the same

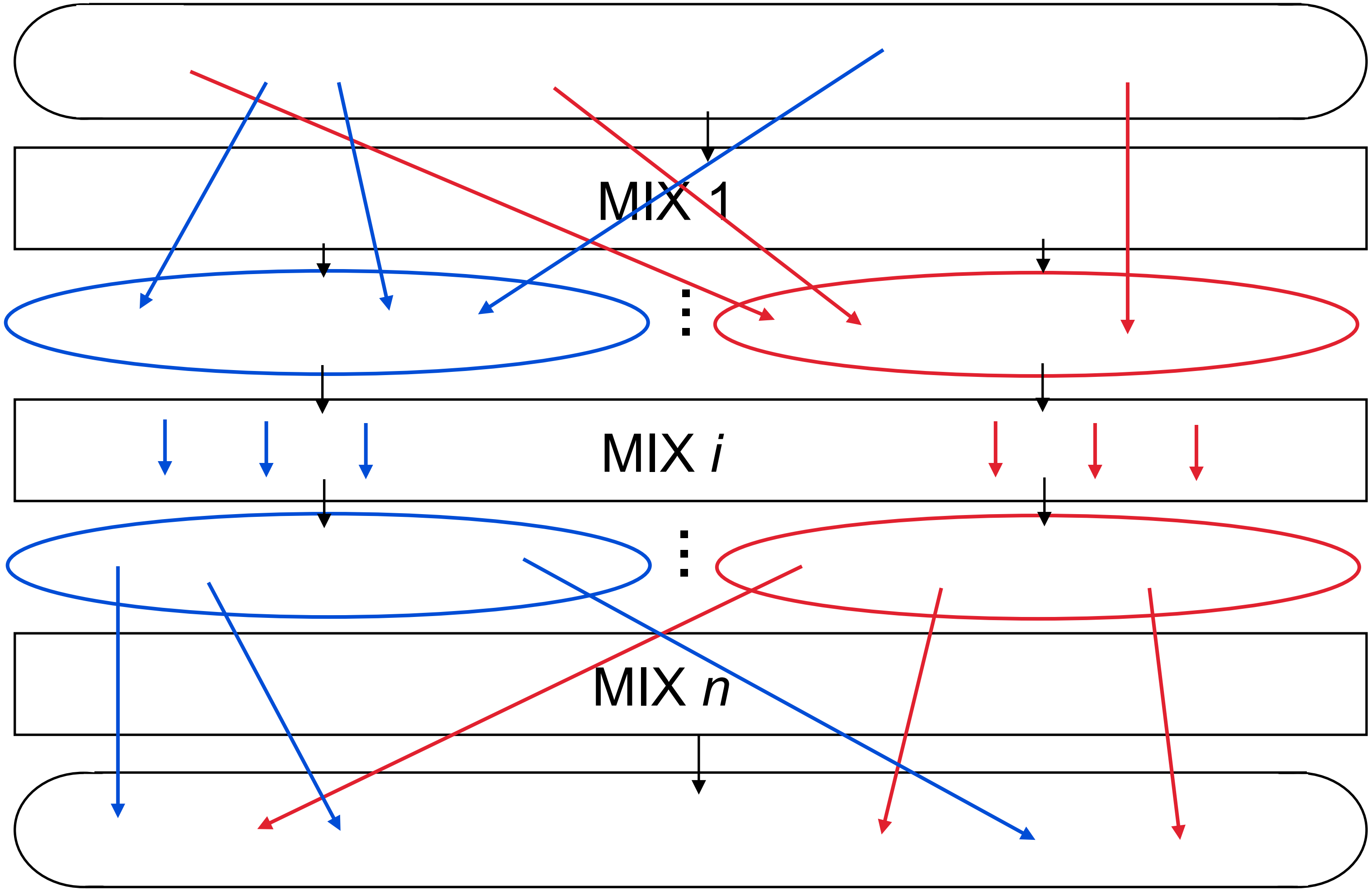
   ➔ re-encryption is irrelevant

2. Maximal protection:

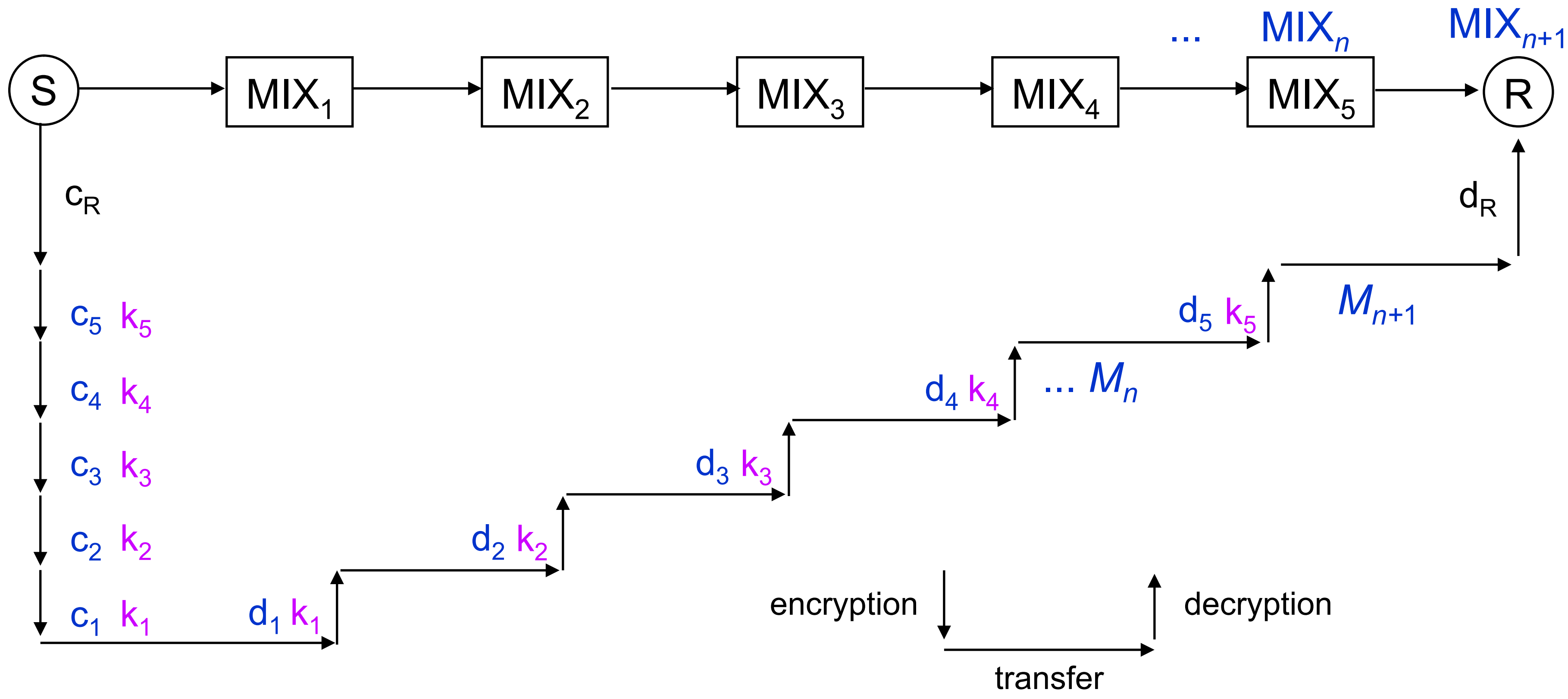
   MIXes are passed through simultaneously and therefore in the same order

Maximal protection

Pass through MIXes in the same order



Re-encryption scheme for sender anonymity



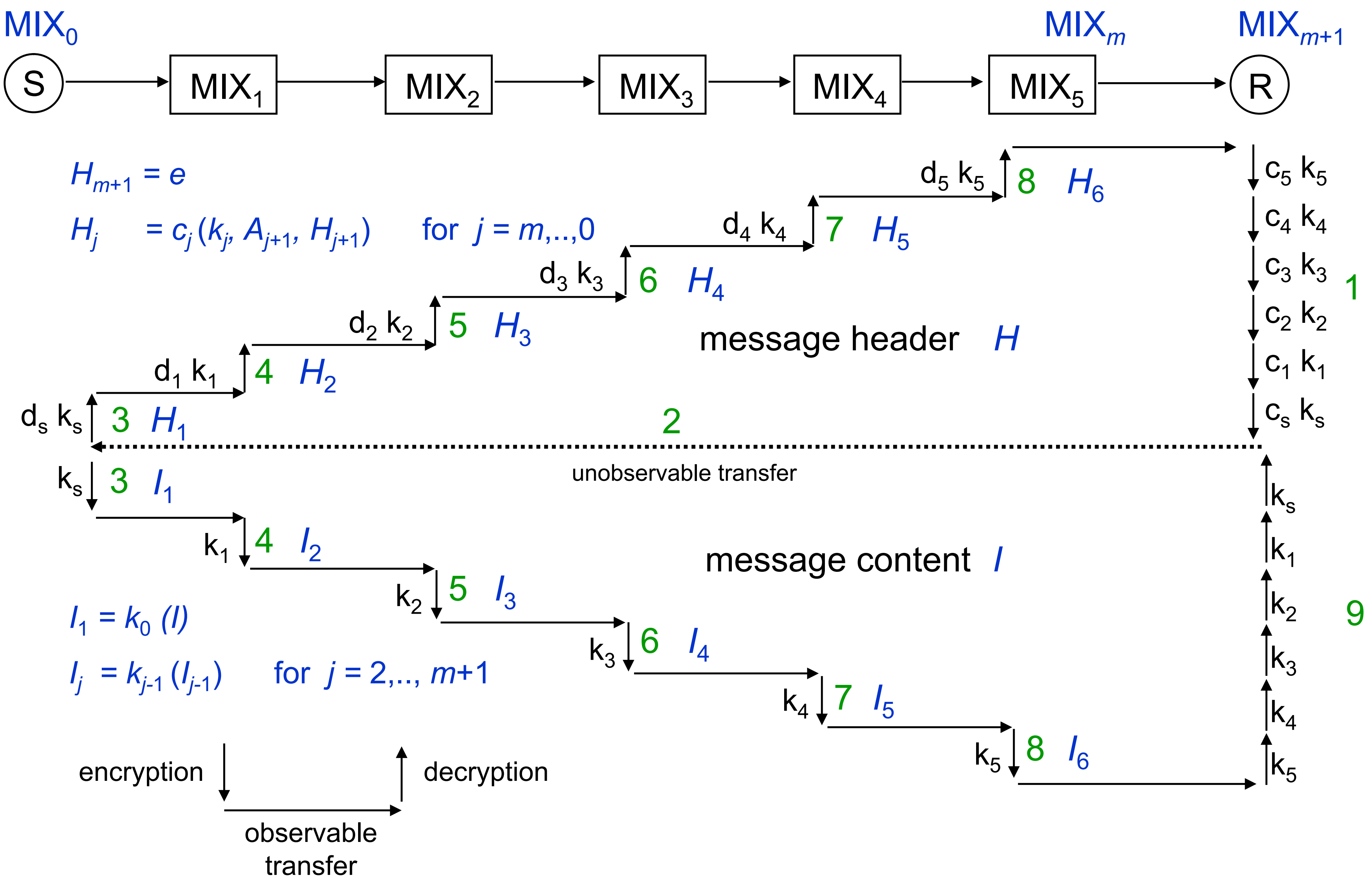
indirect re-encryption scheme for sender anonymity

$M_{n+1} = c_{n+1}(M)$

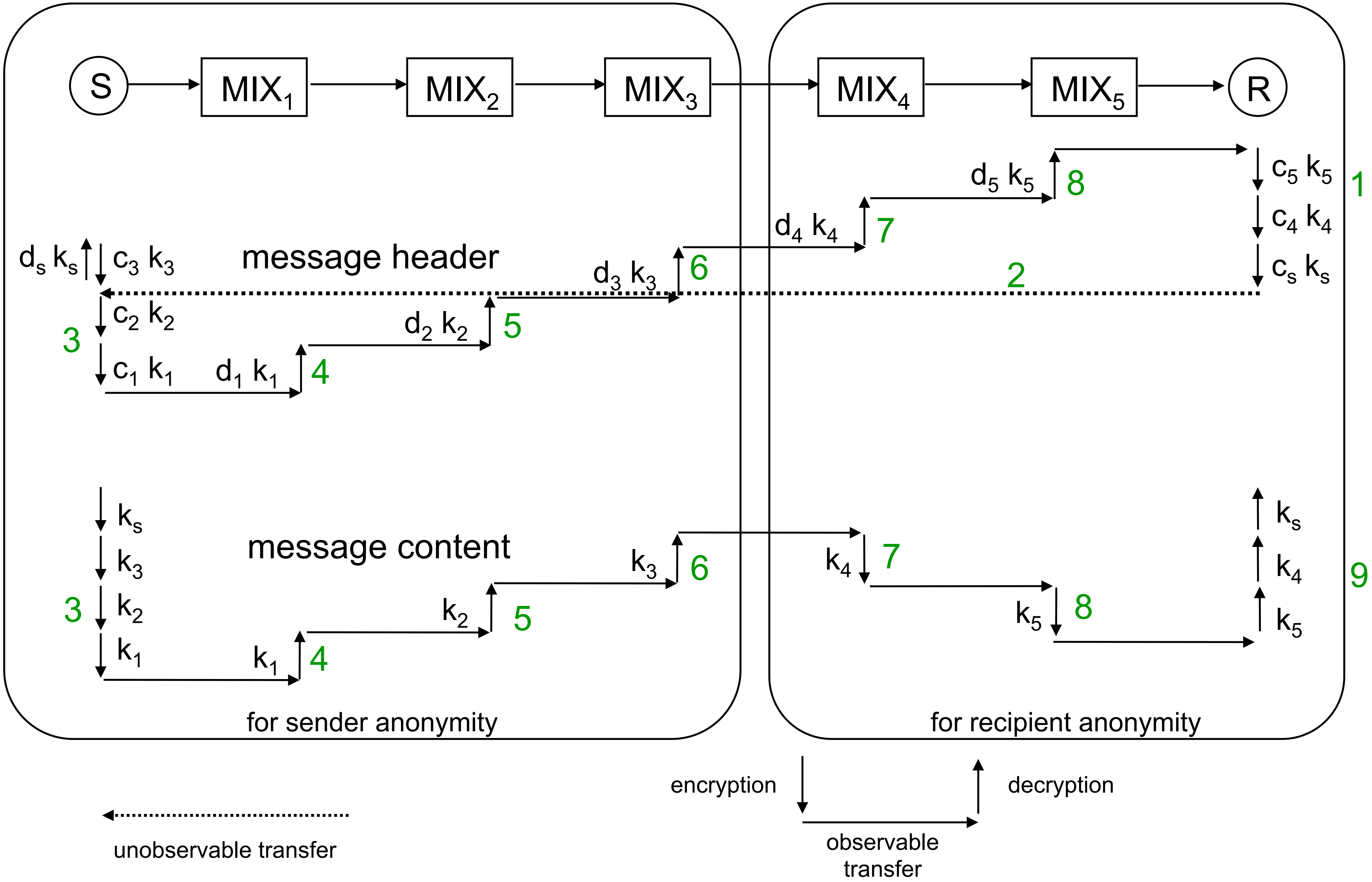
$M_i = c_i(z_i, A_{i+1}, M_{i+1})$ for $i = n,..,1$

$M_i = c_i(k_i, A_{i+1}); k_i(M_{i+1})$

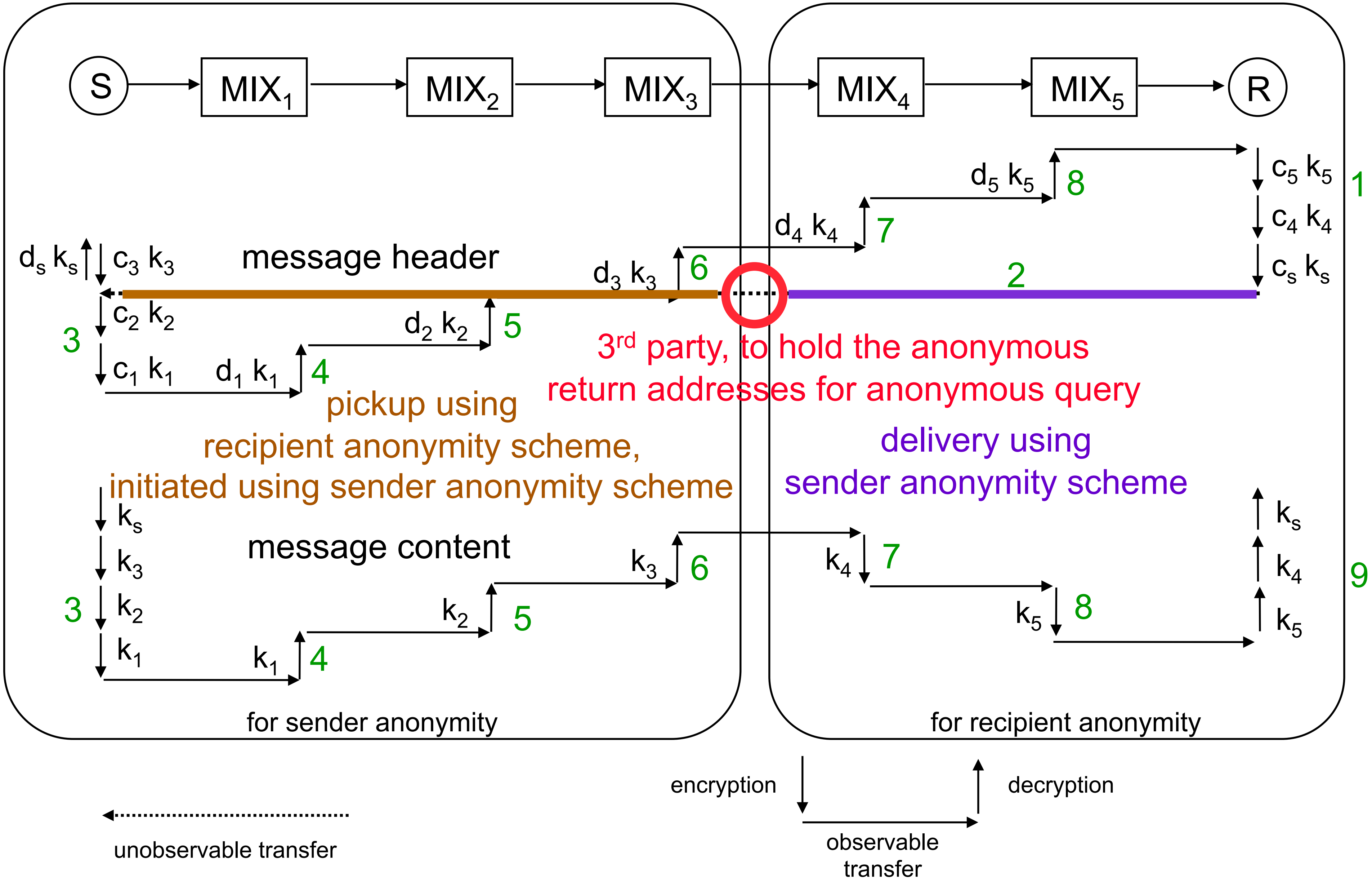
Indirect re-encryption scheme for recipient anonymity



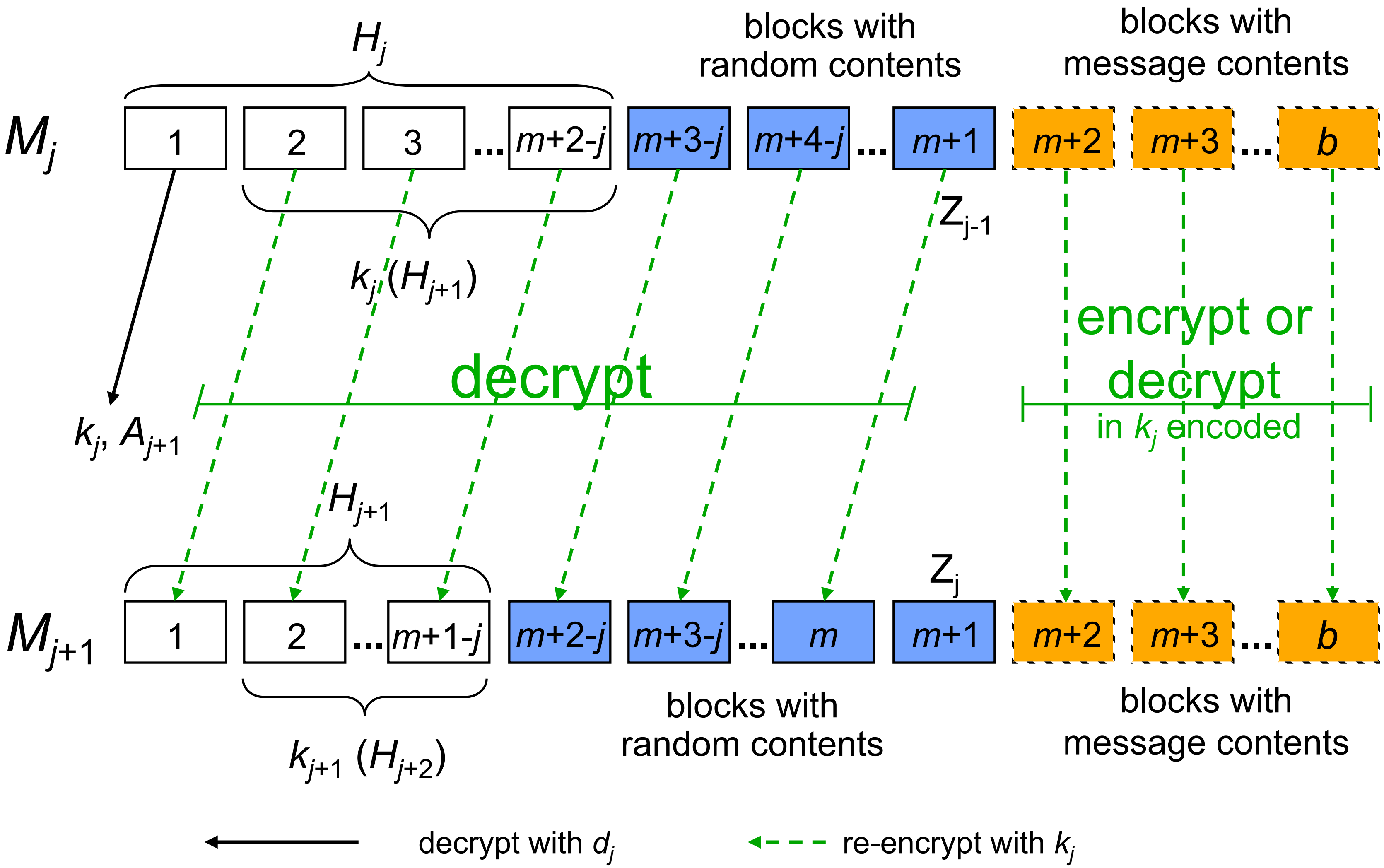
Indirect re-encryption scheme for sender and recipient anonymity



Indirect re-encryption scheme for sender and recipient anonymity



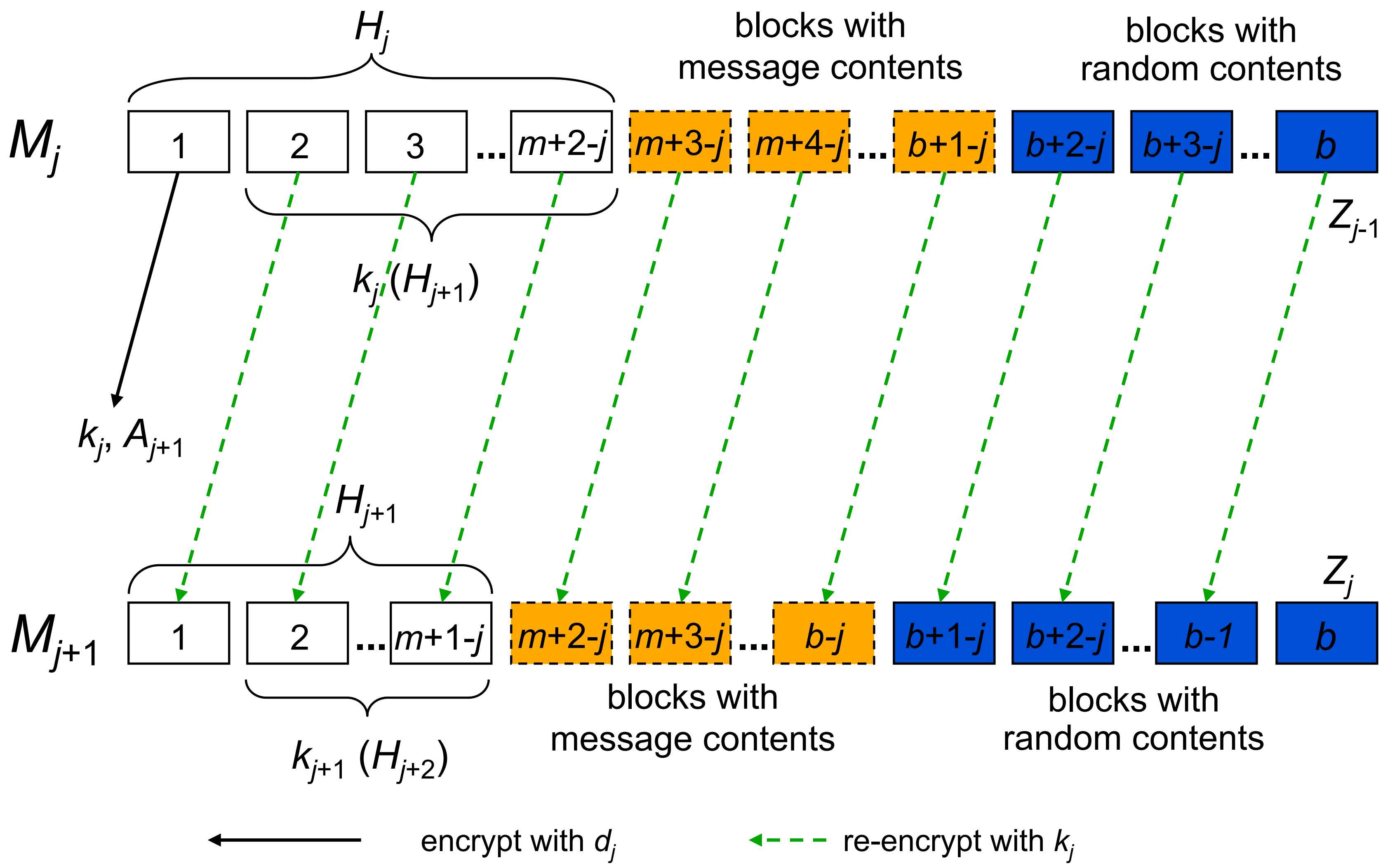
Indirect re-encryption scheme maintaining message length



$H_{m+1} = [e]$

$H_j = [c_j(k_j, A_{j+1})], k_j(H_{j+1})$ for $j = m,..,1$

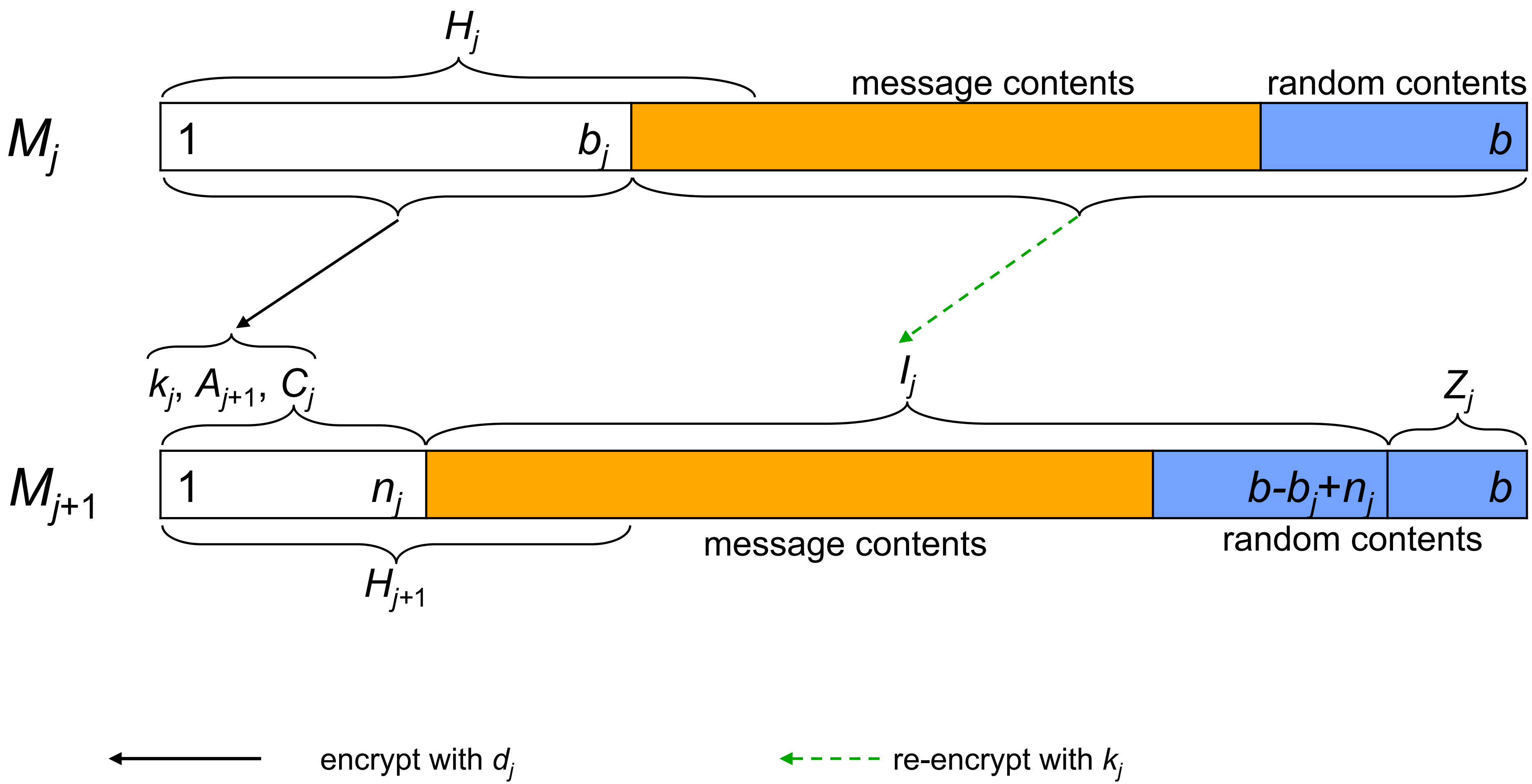
Indirect re-encryption scheme maintaining message length for special symmetric encryption systems



if $k^{-1}(k(M)) = M$

and $k(k^{-1}(M)) = M$

Minimally message expanding re-encryption scheme maintaining message length

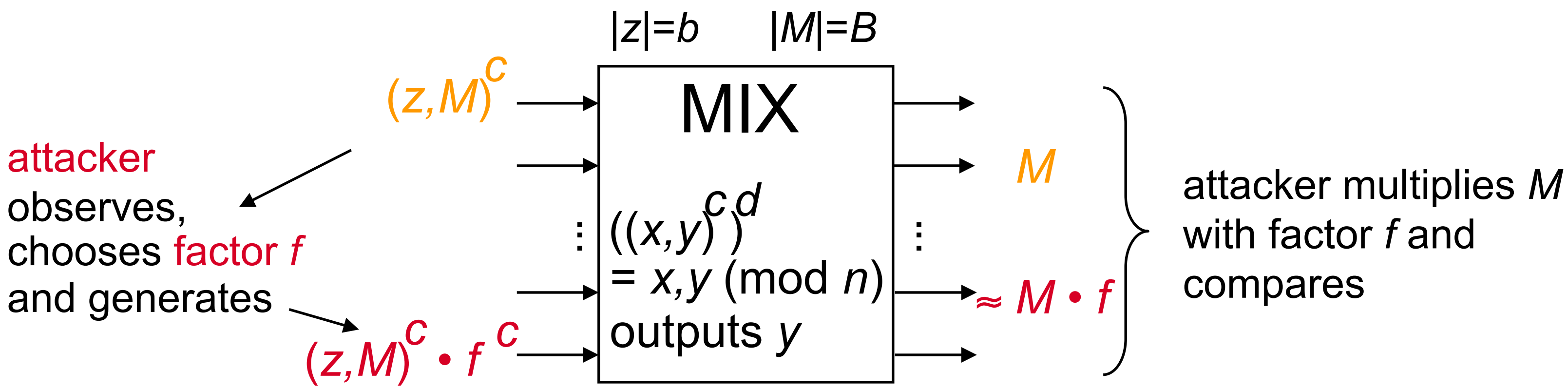


if $k^{-1}(k(M)) = M$

and $k(k^{-1}(M)) = M$

Breaking the direct RSA-implementation of MIXes (1)

Implementation of MIXes using RSA without redundancy predicate and with contiguous bit strings (David Chaum, 1981) is insecure:



Unlinkability, if many factors *f* are possible.

$2^b \cdot 2^B \leq n-1$ hold always and normally $b << B$.

If the random bit strings are the most significant bits, it holds

$(z,M) = z \cdot 2^B + M$ and

$(z,M) \cdot f \equiv (z \cdot 2^B + M) \cdot f \equiv z \cdot 2^B \cdot f + M \cdot f.$

Breaking the direct RSA-implementation of MIXes (2)

Let the identifiers *z'* and *M'* be defined by

$$(z,M)\cdot f \equiv z'\cdot 2^B + M' \Rightarrow$$

$$z\cdot 2^B\cdot f + M\cdot f \equiv z'\cdot 2^B + M' \Rightarrow$$

$$2^B\cdot (z\cdot f - z') \equiv M' - M\cdot f \Rightarrow$$

$$z\cdot f - z' \equiv (M' - M\cdot f) \cdot (2^B)^{-1} \qquad (1)$$

If the attacker chooses $f \leq 2^b$, it holds

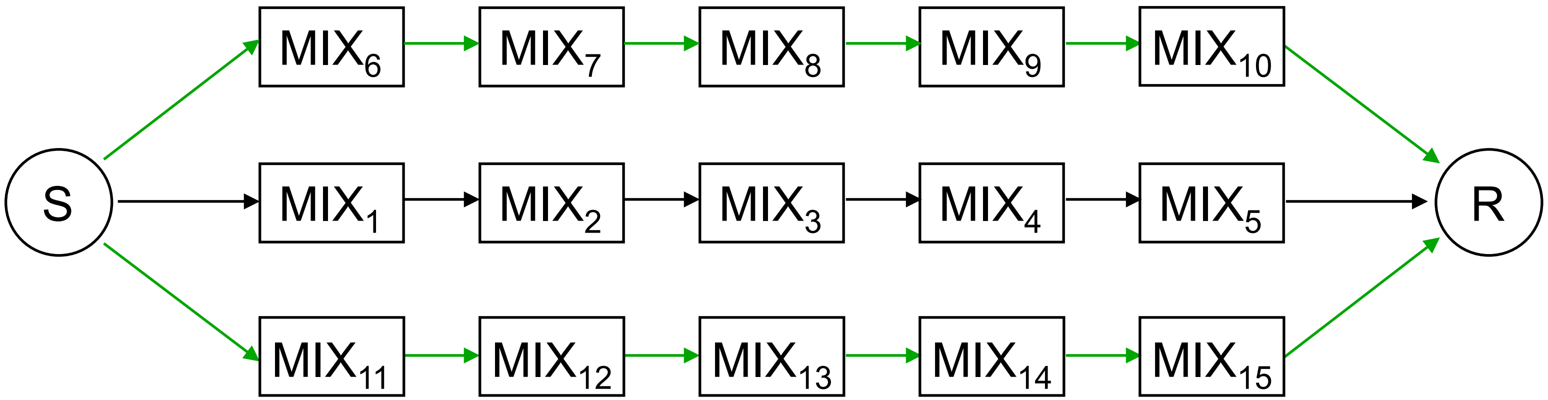
$$-2^b < z\cdot f - z' < 2^{2b} \qquad (2)$$

The attacker replaces in (1) *M* and *M'* by all output-message pairs of the batch and tests (2).

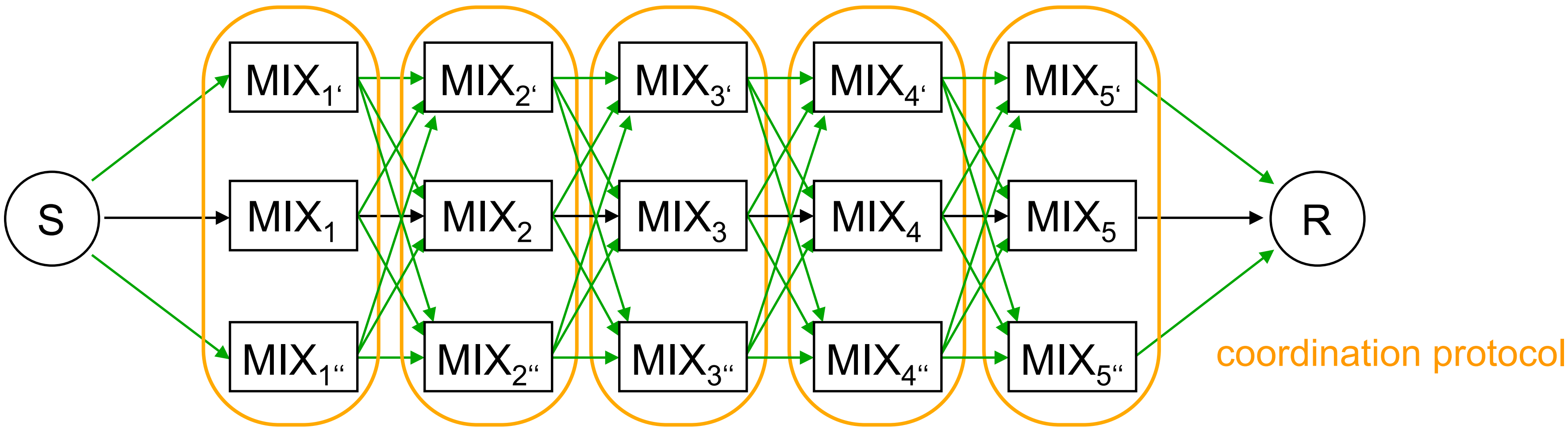
(2) holds, if $b \ll B$, very probably only for one pair (P1,P2). P1 is output message to $(z,M)^c$, P2 to $(z,M)^c\cdot f^c$.

If (2) holds for several pairs, the attack is repeated with another factor.

Fault tolerance in MIX-networks (1)

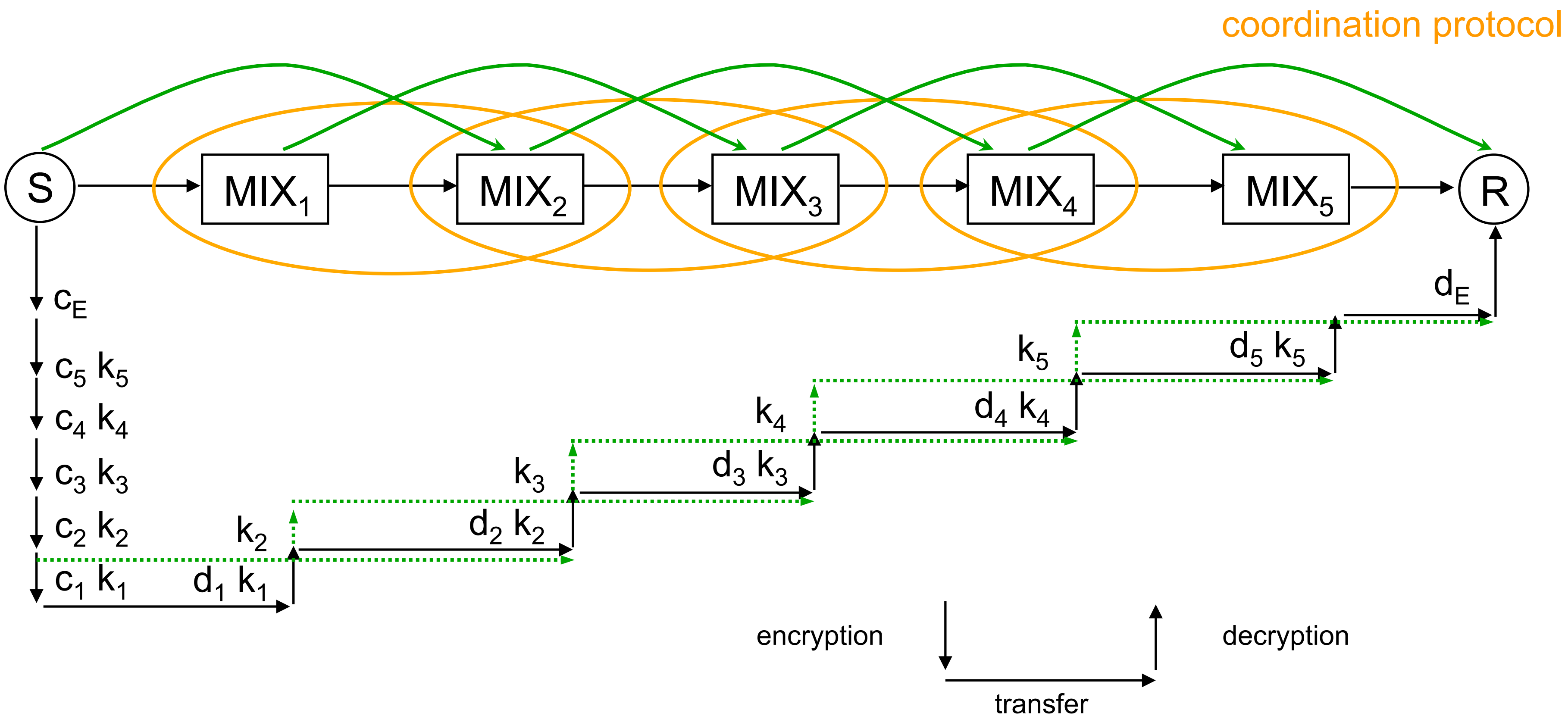


2 alternative routes via disjoint MIXes



$MIX_{i‘}$ or $MIX_{i“}$ can substitute MIX_i

Fault tolerance in MIX-networks (2)



In each step, one MIX can be skipped

Complexity of the basic methods

	unobservability of neighboring lines and stations as well as digital signal regeneration RING-network	DC-network	MIX-network
attacker model	physically limited	computationally restricted w.r.t. service delivery --- computationally restricted • cryptographically strong • well analyzed	computationally restricted not even well analyzed asymmetric encryption systems are known which are secure against adaptive active attacks
expense per user	$O(n)$ $(\geq \frac{n}{2})$ transmission	$O(n)$ $(\geq \frac{n}{2})$ transmission $O(k \cdot n)$ key	$O(k)$, practically: ≈ 1 transmission on the last mile ... in the core network $O(k^2)$, practically: $\approx k$

n = number of users

k = connectedness key graph of DC-networks respectively number of MIXes

Encryption in layer models

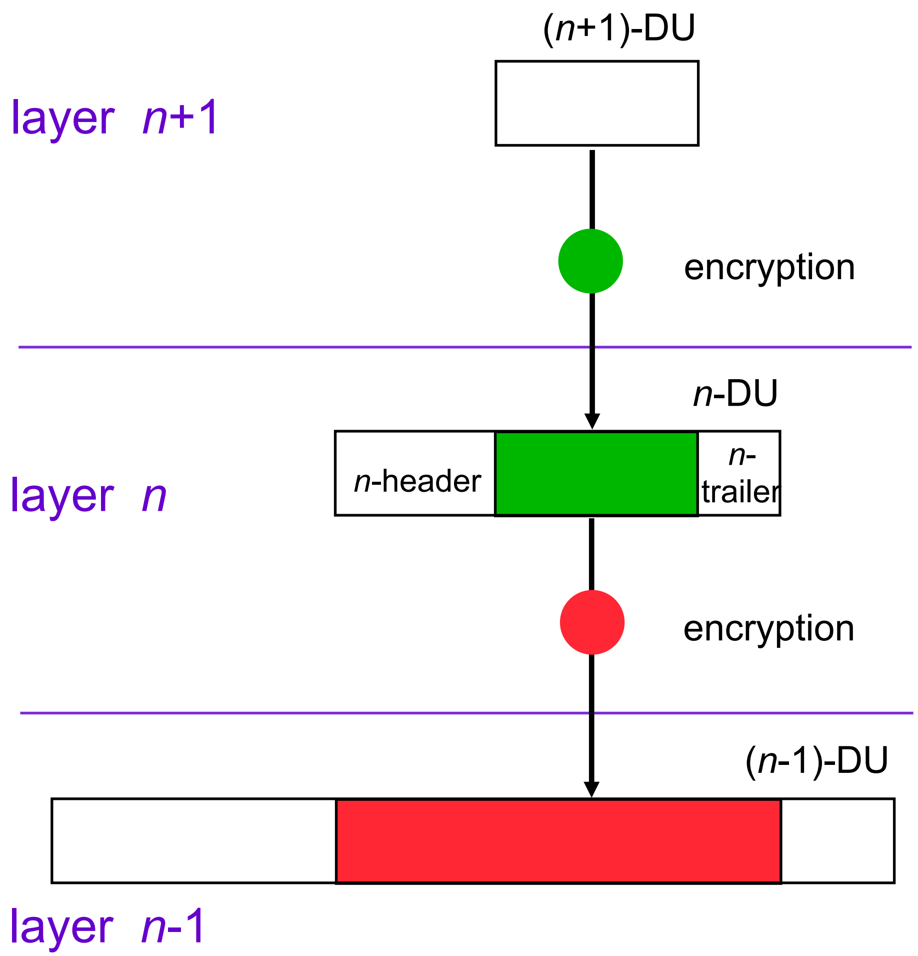
In the OSI model it holds:

Layer *n* doesn't have to look at Data Units (DUs) of layer *n*+1 to perform its service. So layer *n*+1 can deliver (*n*+1)-DUs encrypted to layer *n*.

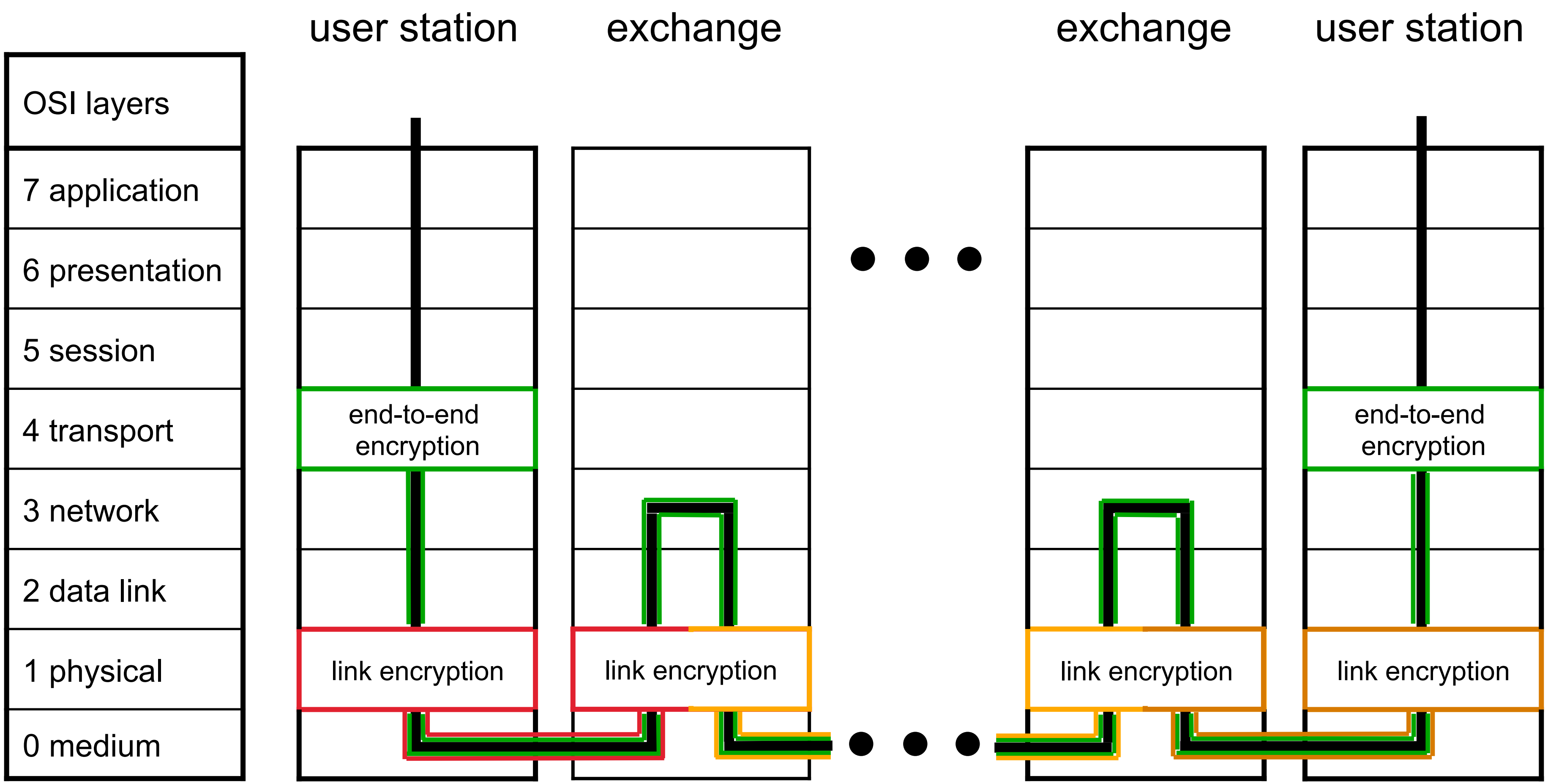
For packet-oriented services, the layer *n* typically furnishes the (*n*+1)-DUs with a *n*-header and possibly with an *n*-trailer, too, and delivers this as *n*-DU to layer *n*-1. This can also be done encrypted again.

and so on.

All encryptions are independent with respect to both the encryption systems and the keys.



Arranging it into the OSI layers (1)



Arranging it into the OSI layers (2)

OSI layers	broadcast		query	MIX-network	DC-network	RING-network
7 application						
6 presentation						
5 session						
4 transport	implicit addressing		implicit addressing			
3 network	broad-cast		query and superpose	buffer and re-encrypt		
2 data link					anonymous access	anonymous access
1 physical		channel selection			superpose keys and messages	digital signal regeneration
0 medium						ring

- yellow: has to preserve anonymity against the communication partner
- green: end-to-end encryption
- orange: has to preserve anonymity
- brown: realizable without consideration of anonymity

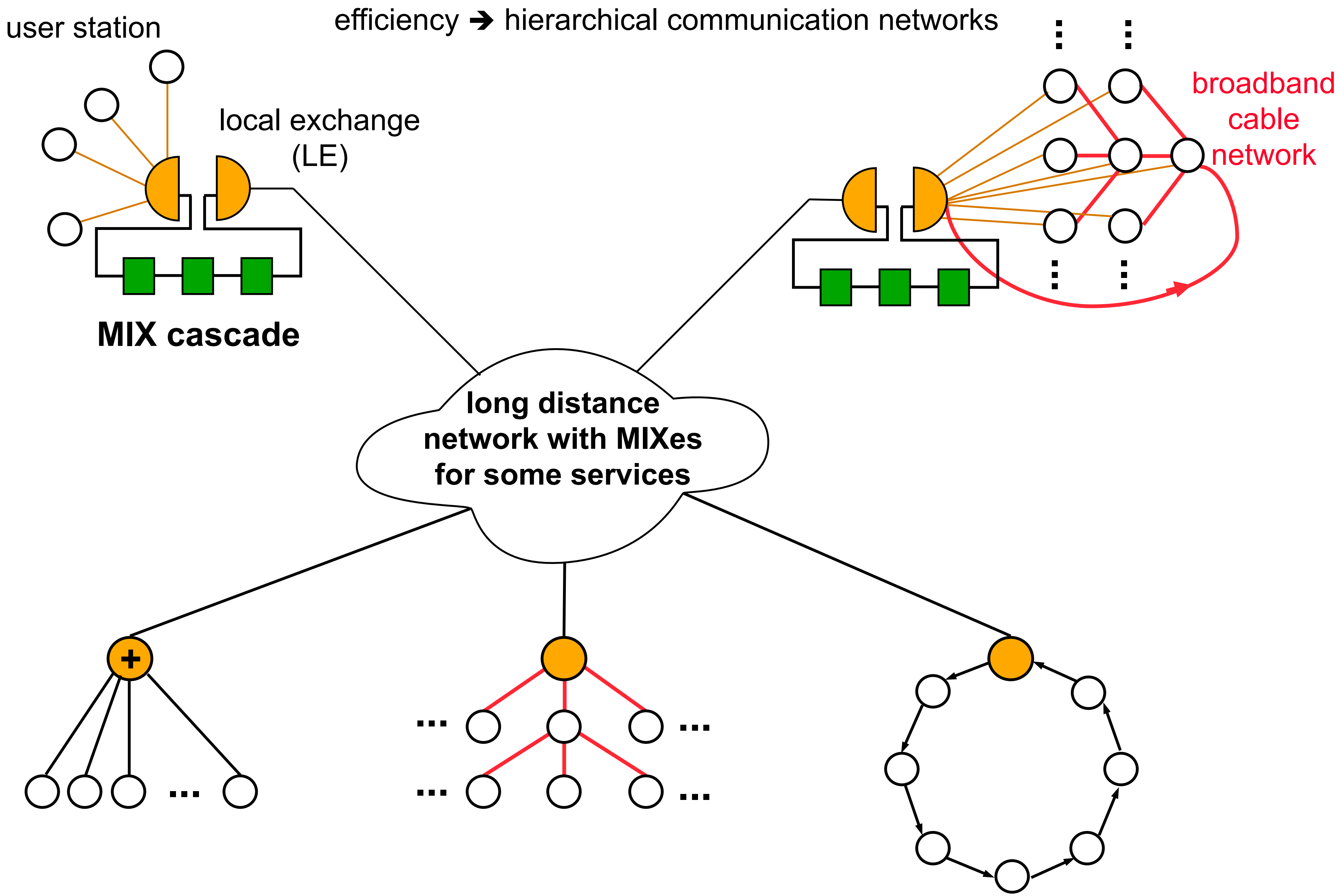
Tolerating errors and active attacks

Problems: series systems w.r.t. availability

maintain the anonymity of „honest“ users

There are adequate extensions.

Network extension by stages



Solution for the ISDN: telephone MIXes

Aims: ISDN services on ISDN transmission system

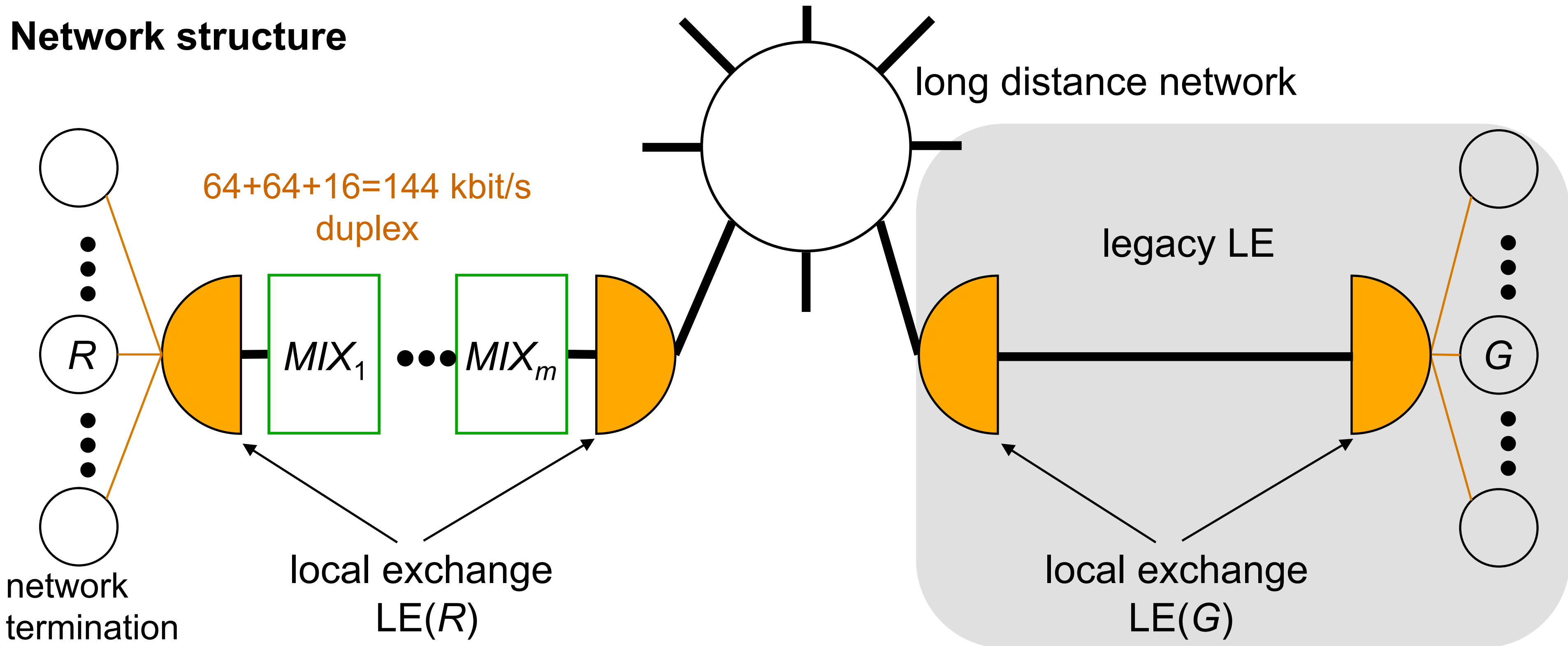
2 independent 64-kbit/s duplex channels on a 144-kbit/s subscriber line

hardly any additional delay on established channels

establish a channel within 3 s

no additional traffic on the long distance network

Network structure



Solution for the ISDN: telephone MIXes

Aims: ISDN services on ISDN transmission system

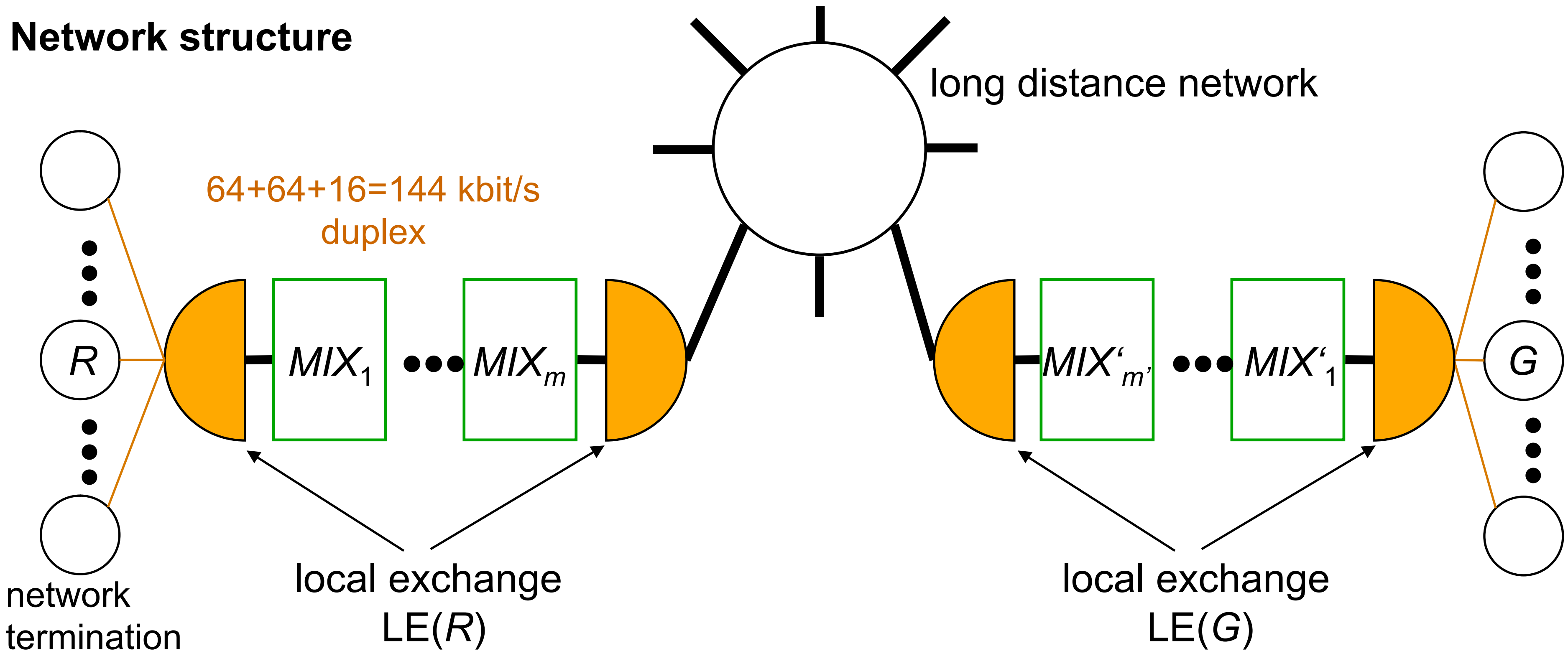
2 independent 64-kbit/s duplex channels on a 144-kbit/s subscriber line

hardly any additional delay on established channels

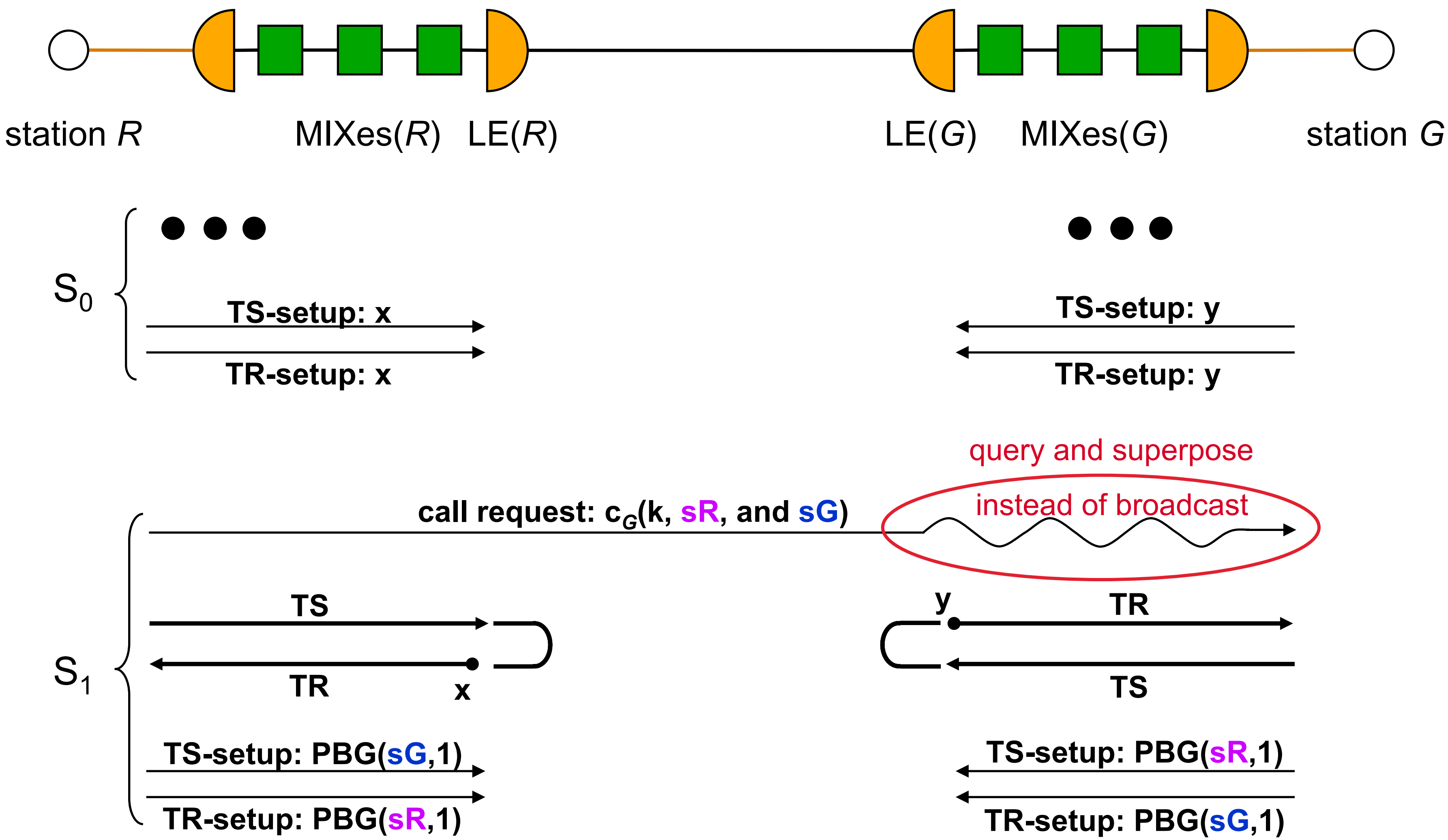
establish a channel within 3 s

no additional traffic on the long distance network

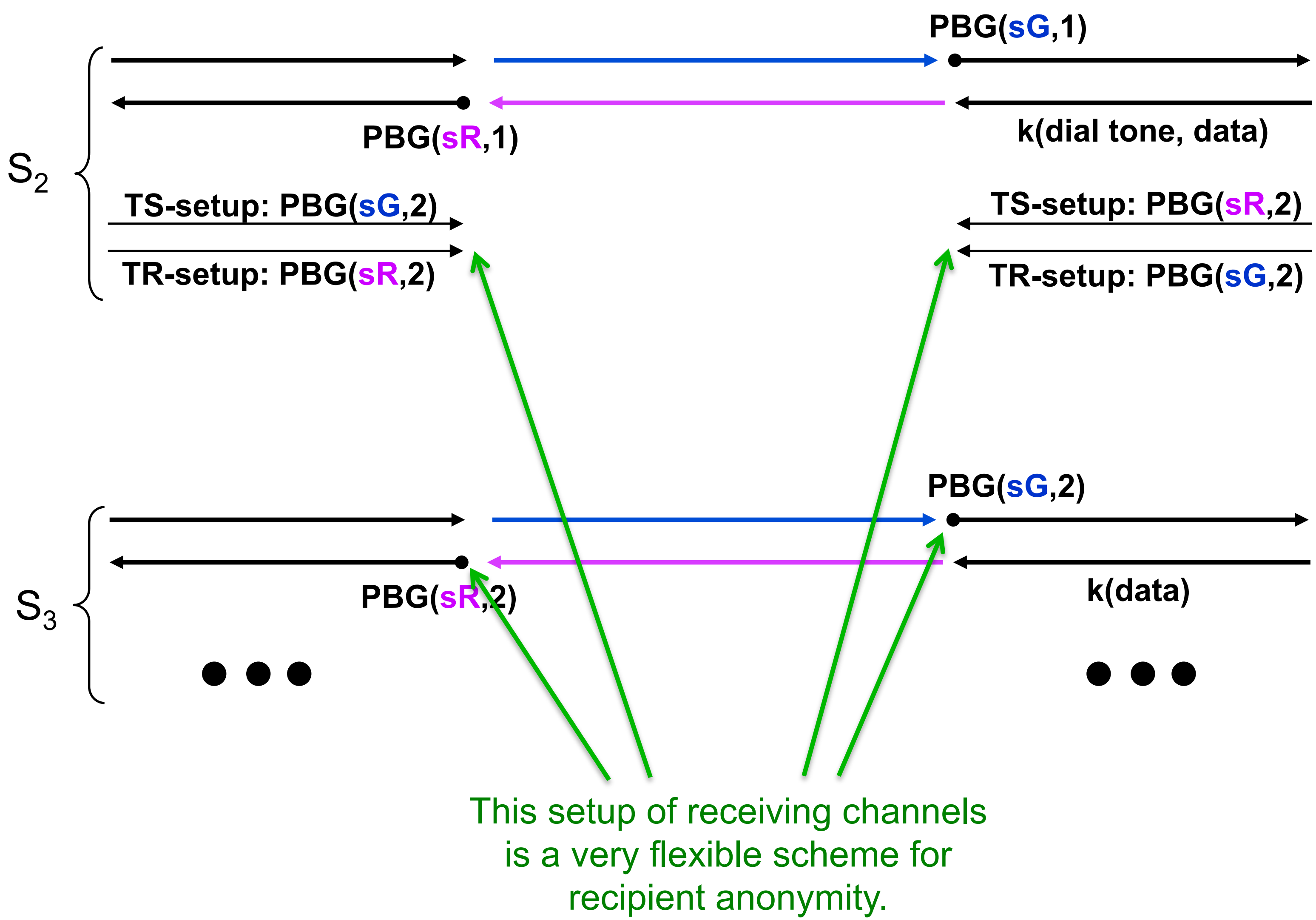
Network structure



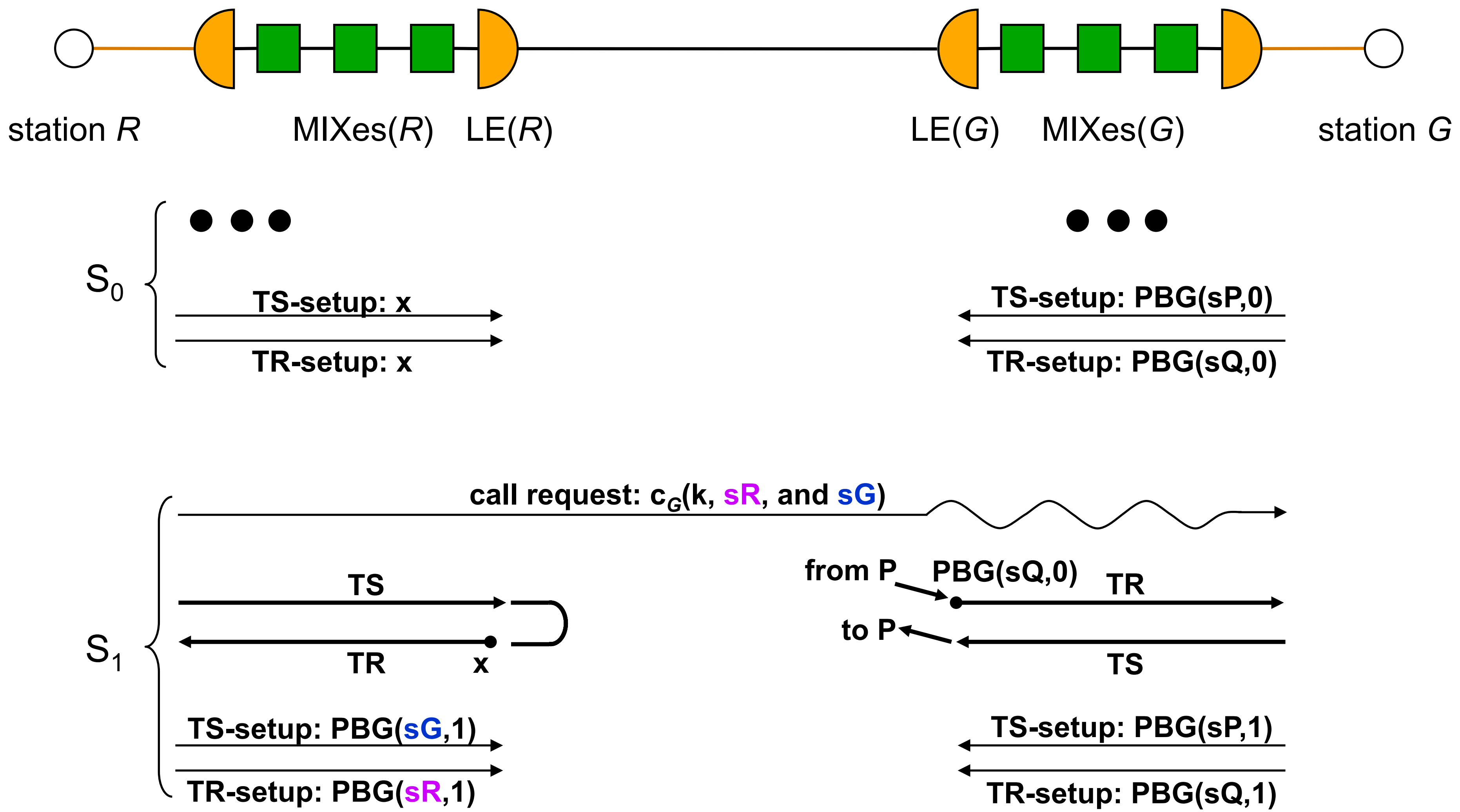
Time-slice channels (1)



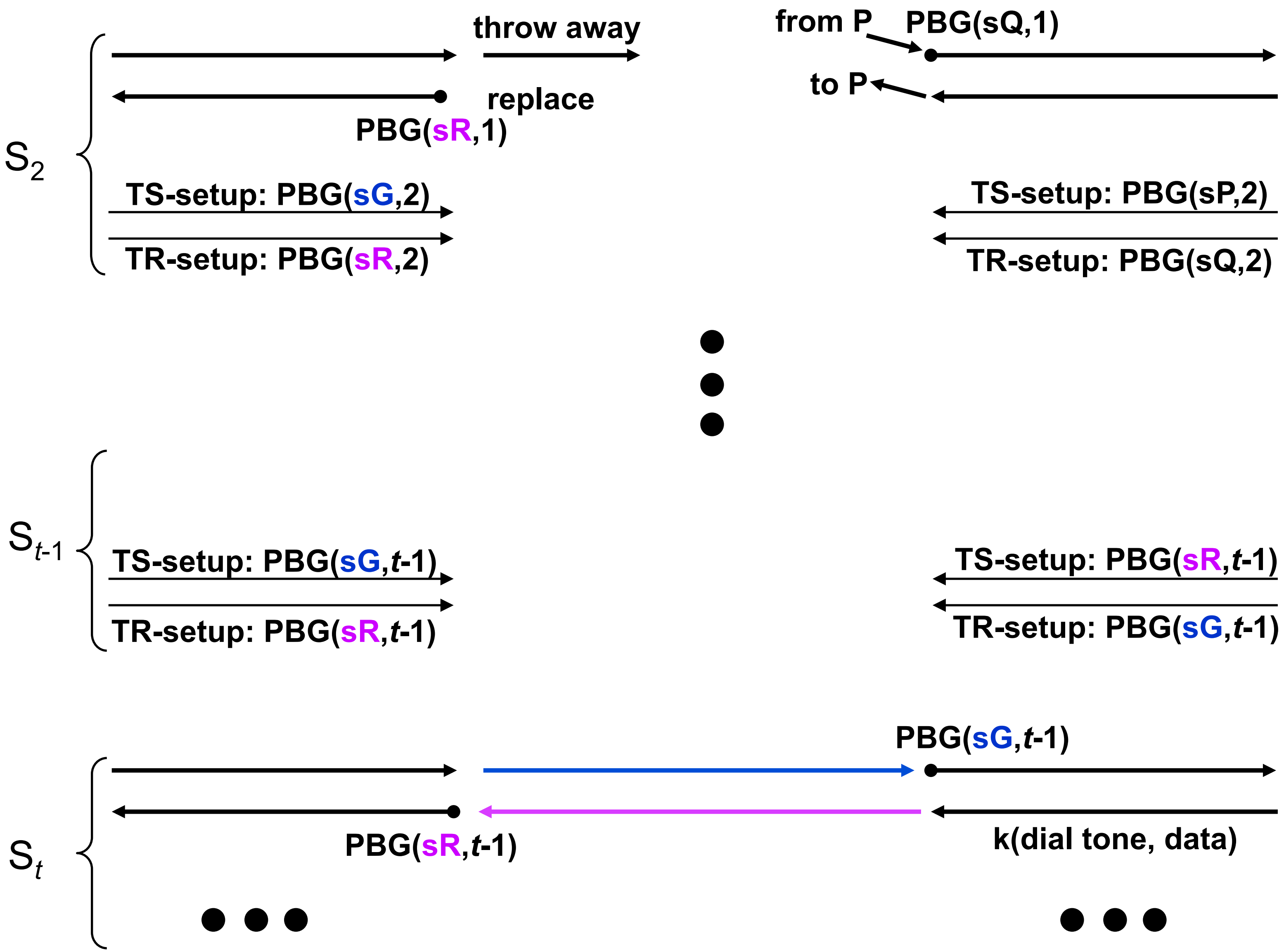
Time-slice channels (2)



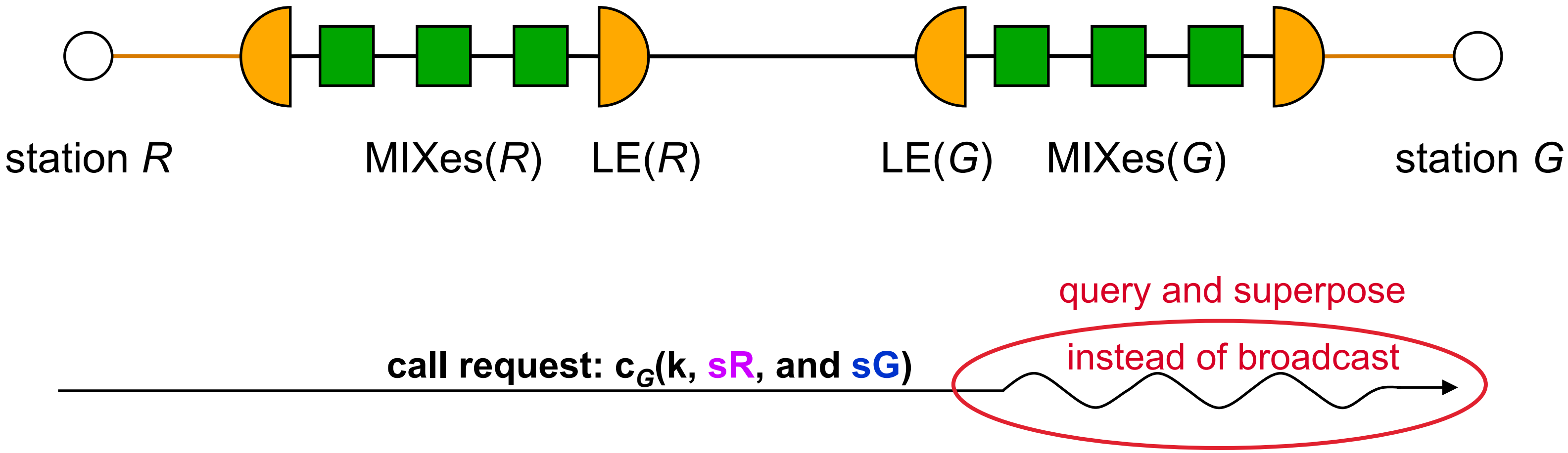
Connection configuration later (1)



Connection configuration later (2)



Query and superpose to receive the call requests



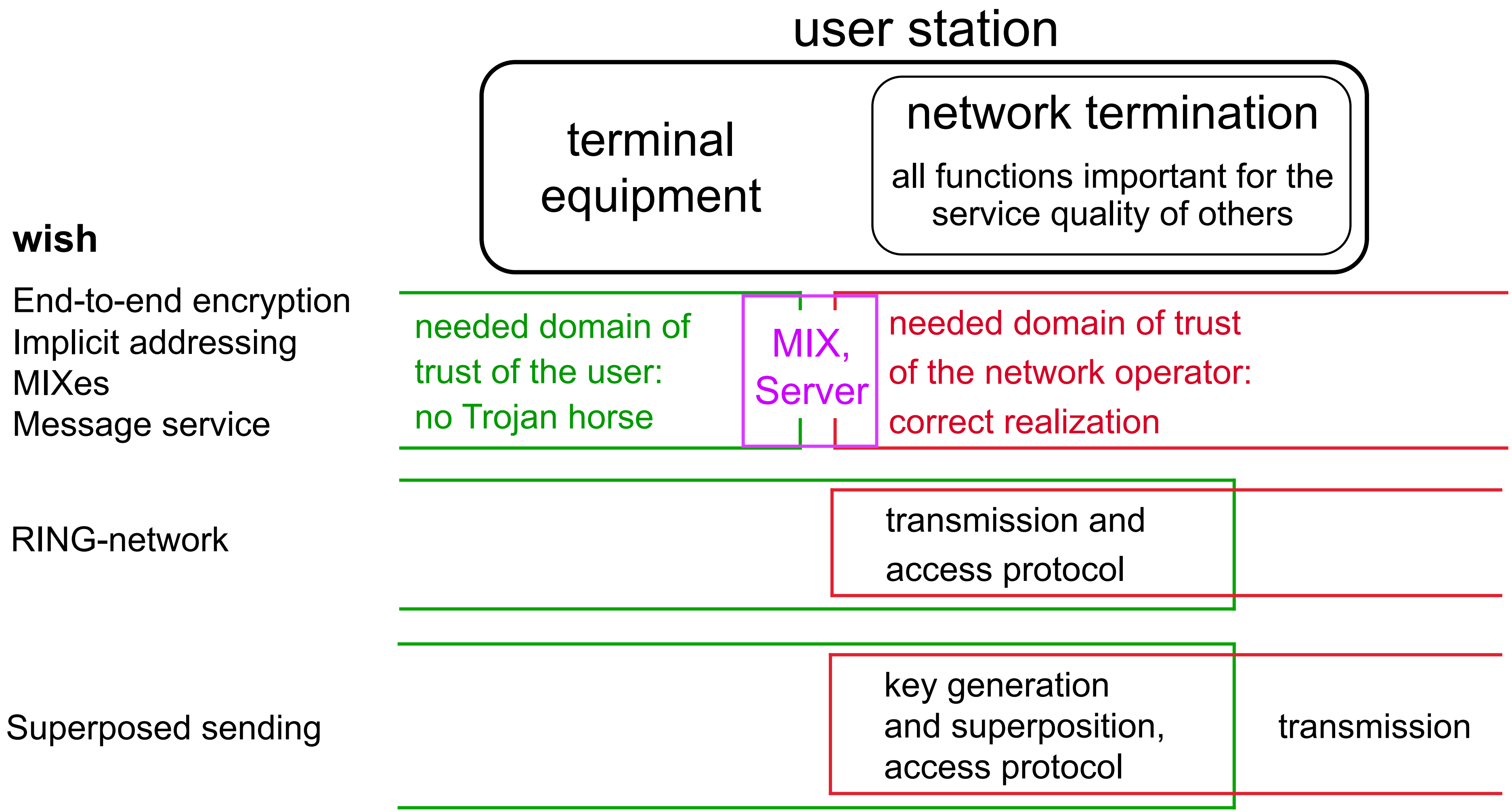
Query and superpose:

- *Each* station has to query in each time slice (else the anonymity set degenerates)
- *Each* station should inquiry *all* its implicit addresses at each query.

  (possible both for visible and invisible addresses without additional expense)

–> The size of the anonymity set is no longer limited by the transmission capacity on the user line, but only by the addition performance of the message servers.

Operatorship of the network components



Problems here are easier than at switching centers:

1. Network terminations are less complex
2. … cannot be changed quickly (hardware, no remote maintenance)

MIXes, Servers: technically easier; organizationally w.r.t. confidence more problematic

Superposed sending: technically more expensive; organizationally easier

Outlook (1)

Using the network → transactions between anonymous partners

→ explicit proof of identity is possible at any time

Protection of traffic data
and data on interests requires
appropriate network structure

→ consider early enough

keep options

Networks offering anonymity can be operated in a "trace users mode" without huge losses in performance,
the converse is not true!

Outlook (2)

Trustworthy data protection in general or only at individual payment for interested persons?

- Concerning traffic data, the latter is technically inefficient.
- The latter has the contrary effect (suspicion).
- Everyone should be able to afford fundamental rights!

Radio networks (1)

Difference to wired networks

- Bandwidth of transmission remains scarce
- The current place of the user is also to be protected

Assumptions

- Mobile user station is *always* identifiable and locatable if the station sends.
- Mobile user station is *not* identifiable and locatable if the station only (passively) receives.

not commend-able

not applic-able

Which measures are applicable?

+ end-to-end encryption

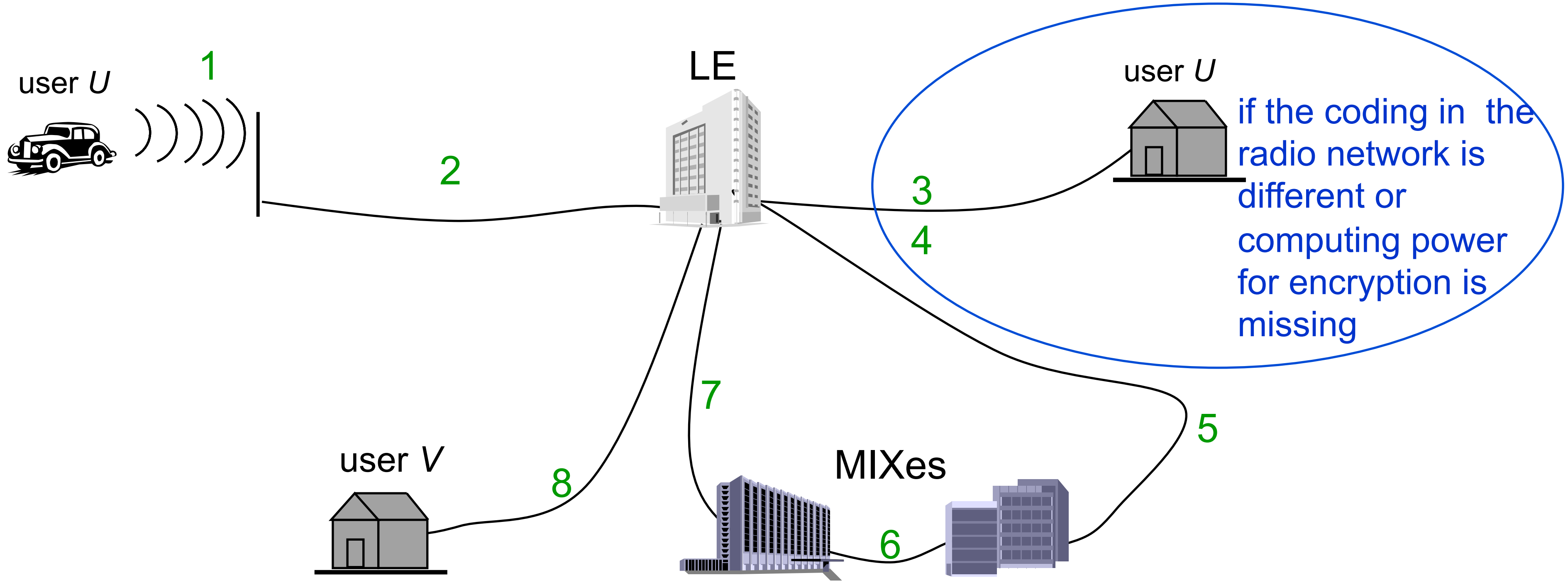
+ link encryption

- dummy messages, unobservability of neighboring lines and stations as well digital signal regeneration, superposed sending

➔ all measures to protect traffic data and data on interests have to be handled in the wired part of the communication network

Radio networks (2)

+ MIXes



+ Broadcast the call request in the whole radio network, only then the mobile station answers. After this the transmission proceeds in one radio cell only.

+ Filter + Generation of visible implicit addresses + Restrict the region

+ Keep the user and SIM anonymous towards the mobile station used.

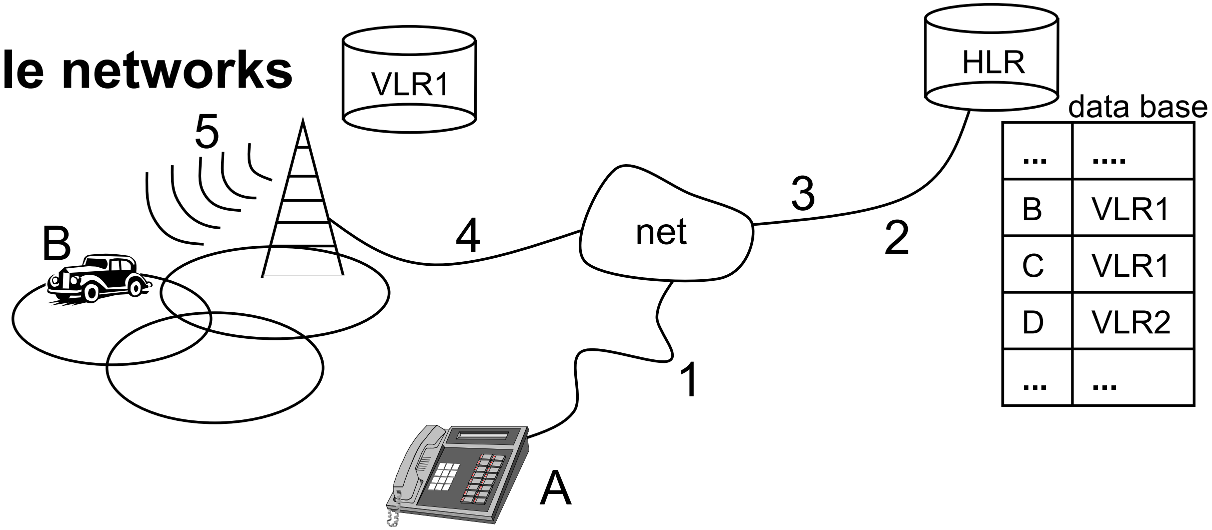
No movement profiles in radio networks

GSM/UMTS – cellular mobile networks

- roaming information

  in central data bases
- operators of the network can

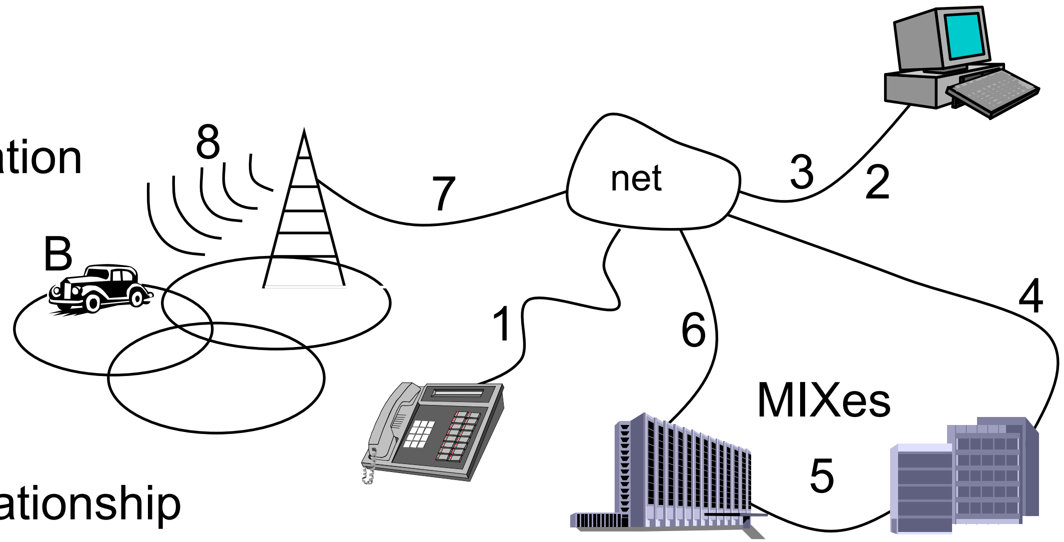
  record the information



Alternative concept

- Maintenance of the roaming information

  in a domain of trust
  - at home (HPC)
  - at trustworthy organizations
- Protection of the communication relationship using MIXes



Electronic Banking

Motivation

- Banking using paper forms – premium version

  Customer gets the completely personalized forms from the bank in which only the value has to be filled in. No signature!

  Electronic banking – usual version

  Customer gets card and PIN, TAN from his/her bank.

  Upcoming

  Customer gets chip card from Bank with

  or → key for MAC
  → key pair for digital signature

- Map exercise of US secret services: observe the citizens of the USSR (1971, Foy 75)

Main part (Everything a little bit more precise)

- Payment system is secure ...

  MAC, digital signature

  payment system using digital signatures

- Pseudonyms (person identifier $\nleftrightarrow$ role-relationship pseudonyms)

Security properties of digital payment systems

digital (integrity, availability)

Payment system is **secure** if

via communication network
immaterial, digital

- user can transfer the rights received,
- user can loose a right only if he is willing to,
- if a user who is willing to pay uniquely denotes another user as recipient, only this entity receives the right,
- user can prove transfers of rights to a third party if necessary (receipt problem), and
- the users cannot increase their rights even if they collaborate,

without the committer being identified.

Problem: messages can be copied perfectly

Solution: witness accepts only the *first* (copy of a) message

Pseudonyms examples

- **person pseudonyms**
  - **public person pseudonym** — phone number
  - **non-public person pseudonym** — account number
  - **anonymous-person pseudonym** — biometric, DNA (as long as no register)
- **role pseudonyms**
  - **business-relationship pseudonym** — pen name
  - **transaction pseudonym** — one-time password

Scalability concerning the protection

A n o n y m i t y

Pseudonyms: Linkability in detail

Distinction between:

1. **Initial linking** between the pseudonym and its holder
2. Linkability due to the **use** of the pseudonym **across different contexts**

Pseudonyms: Initial linking to holder

Public pseudonym:

The linking between pseudonym and its holder may be publicly known from the very beginning.

Phone number with its owner listed in public directories

Initially non-public pseudonym:

The linking between pseudonym and its holder may be known by certain parties (trustees for identity), but is not public at least initially.

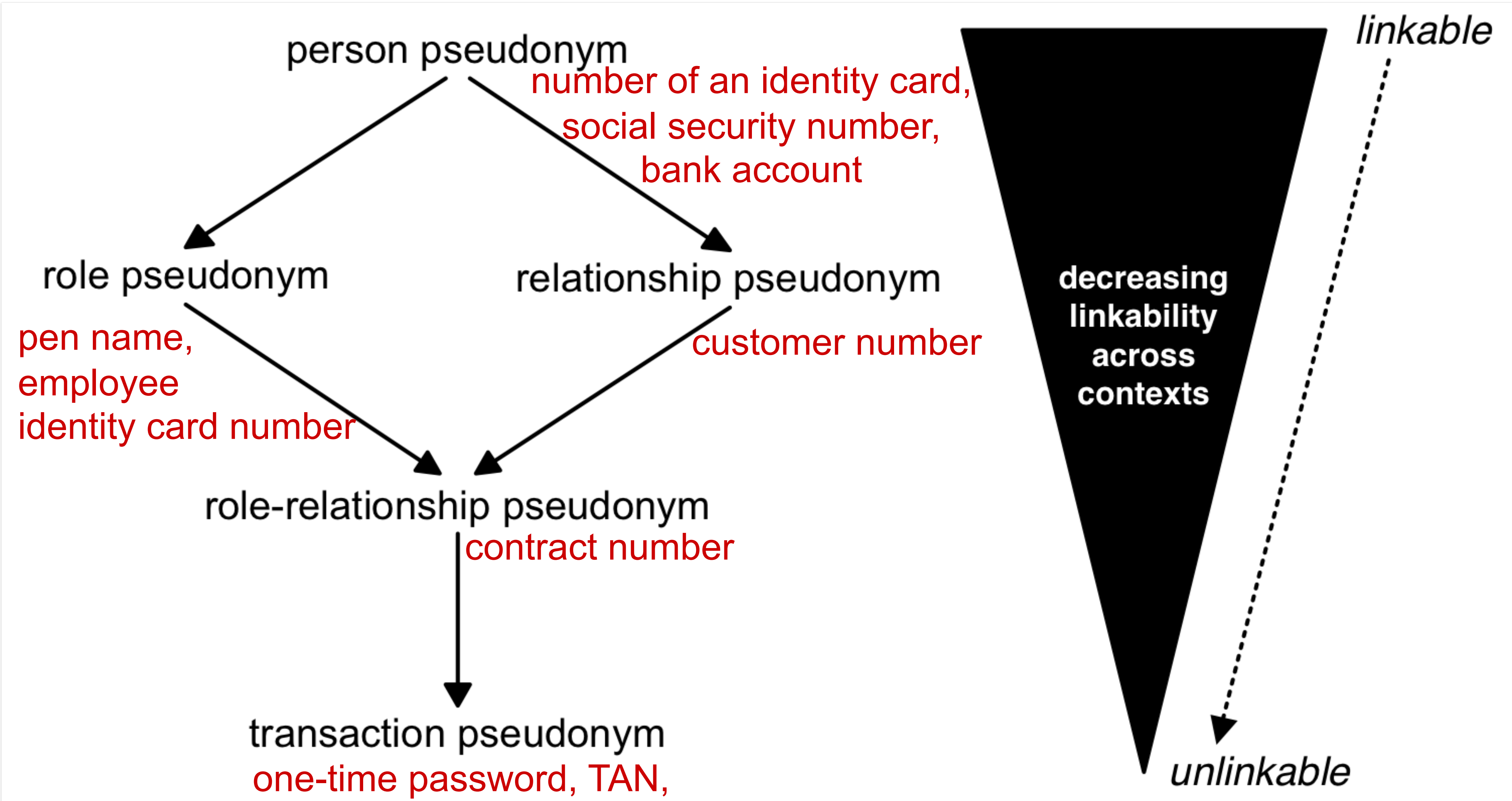
Bank account with bank as trustee for identity,
Credit card number ...

Initially unlinked pseudonym:

The linking between pseudonym and its holder is – at least initially – not known to anybody (except the holder).

Biometric characteristics; DNA (as long as no registers)

Pseudonyms: Use across different contexts => partial order



one-time use public-key pair

A → B stands for “B enables stronger unlinkability than A”

Notations: transfer of a signed message from *X* to *Y*

functional notation

signing
the message *M*:

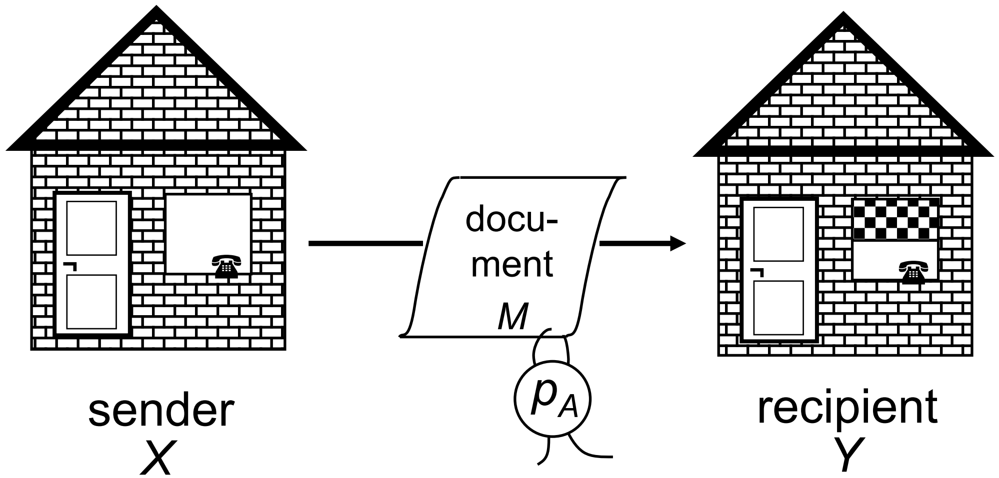
$s_A(M)$

$X \xrightarrow{M,\ s_A(M)} Y$

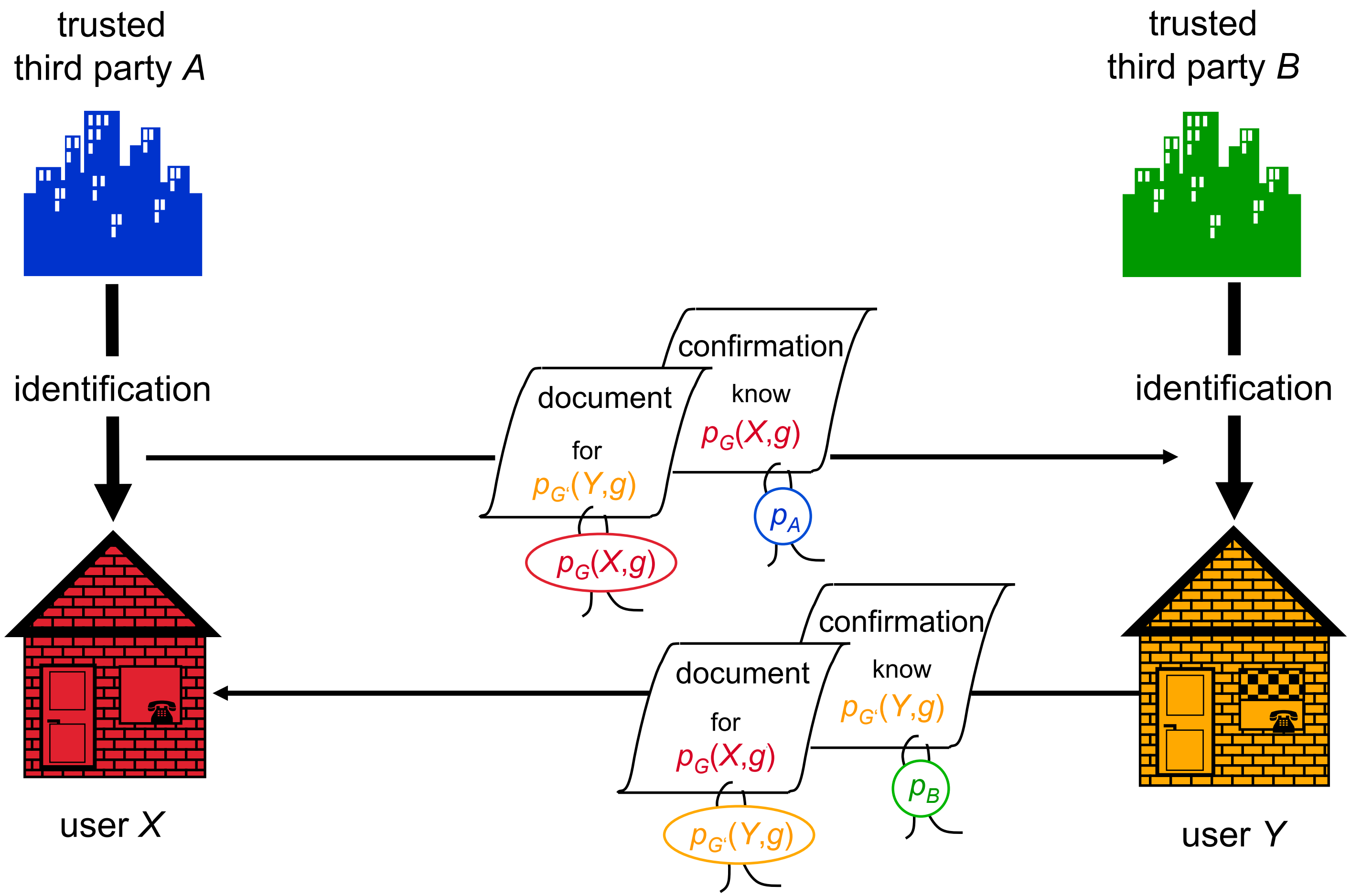
test the
signature:

$t_A\ (M,\ s_A(M))$?

graphical notation



Authenticated anonymous declarations between business partners that can be de-anonymized



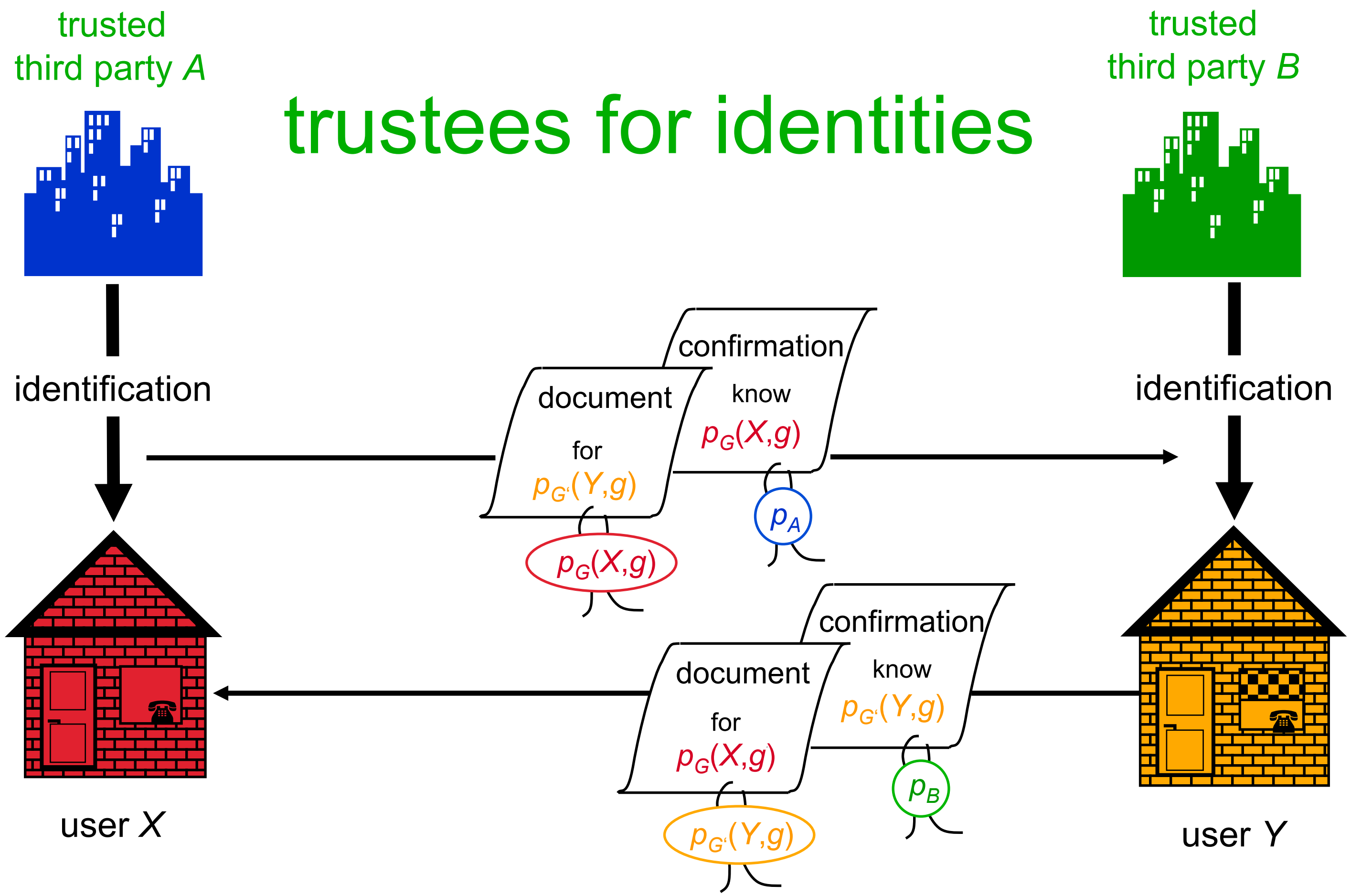
Generalization:

$$X \rightarrow B_1 \rightarrow B_2 \rightarrow \ldots \rightarrow B_n \rightarrow Y$$

$$X \searrow B'_1 \rightarrow B'_2 \rightarrow \ldots \rightarrow B'_m \nearrow Y$$

error / attack tolerance (cf. MIXes)

Authenticated anonymous declarations between business partners that can be de-anonymized

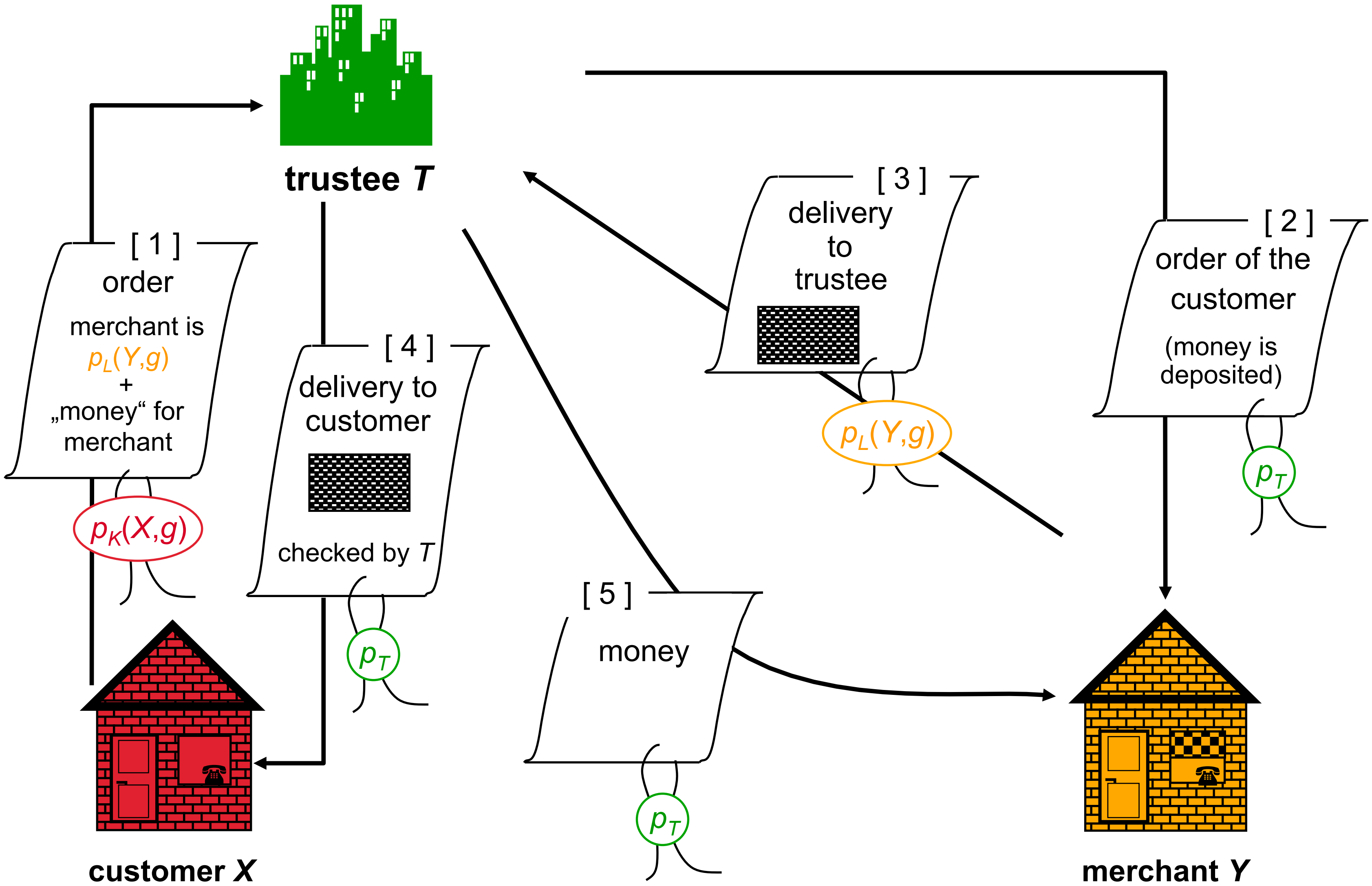


Generalization:

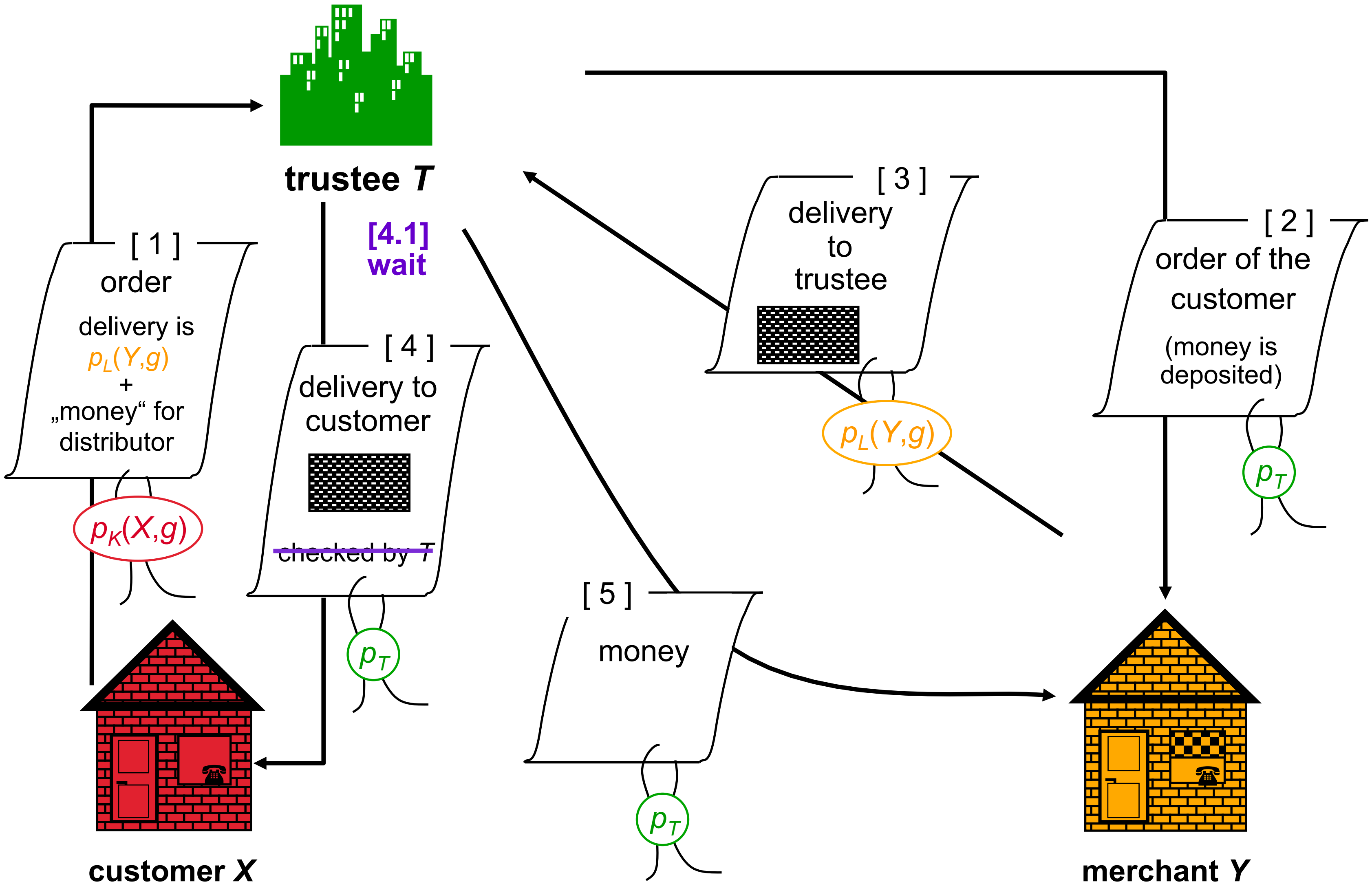
$$\begin{array}{l} X \rightarrow B_1 \rightarrow B_2 \rightarrow \ldots \rightarrow B_n \rightarrow Y \\ \quad \searrow B'_1 \rightarrow B'_2 \rightarrow \ldots \rightarrow B'_m \nearrow \end{array}$$

error / attack tolerance (cf. MIXes)

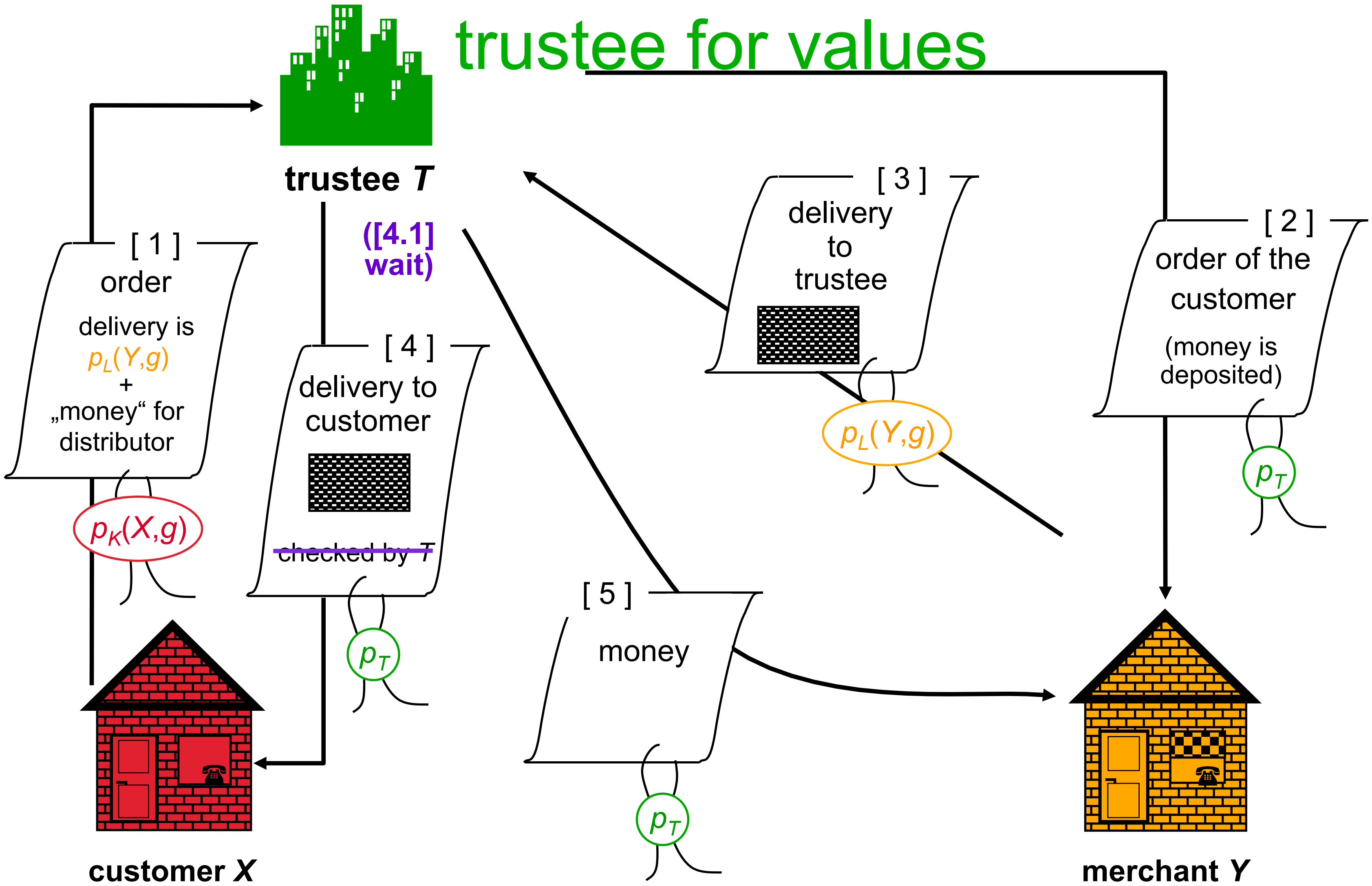
Security for completely anonymous business partners using active trustee who can check the goods



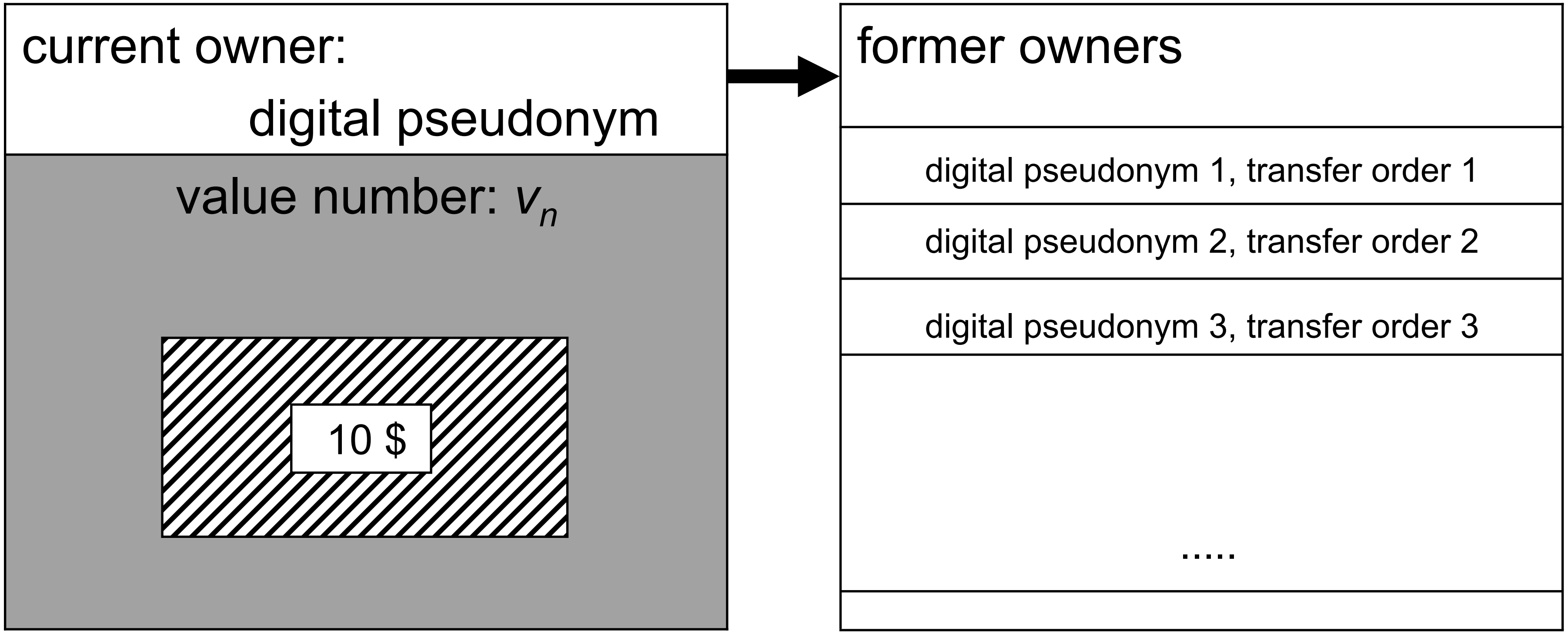
Security for completely anonymous business partners using active trustee who can not check the goods



Security for completely anonymous business partners using active trustee who can (not) check the goods

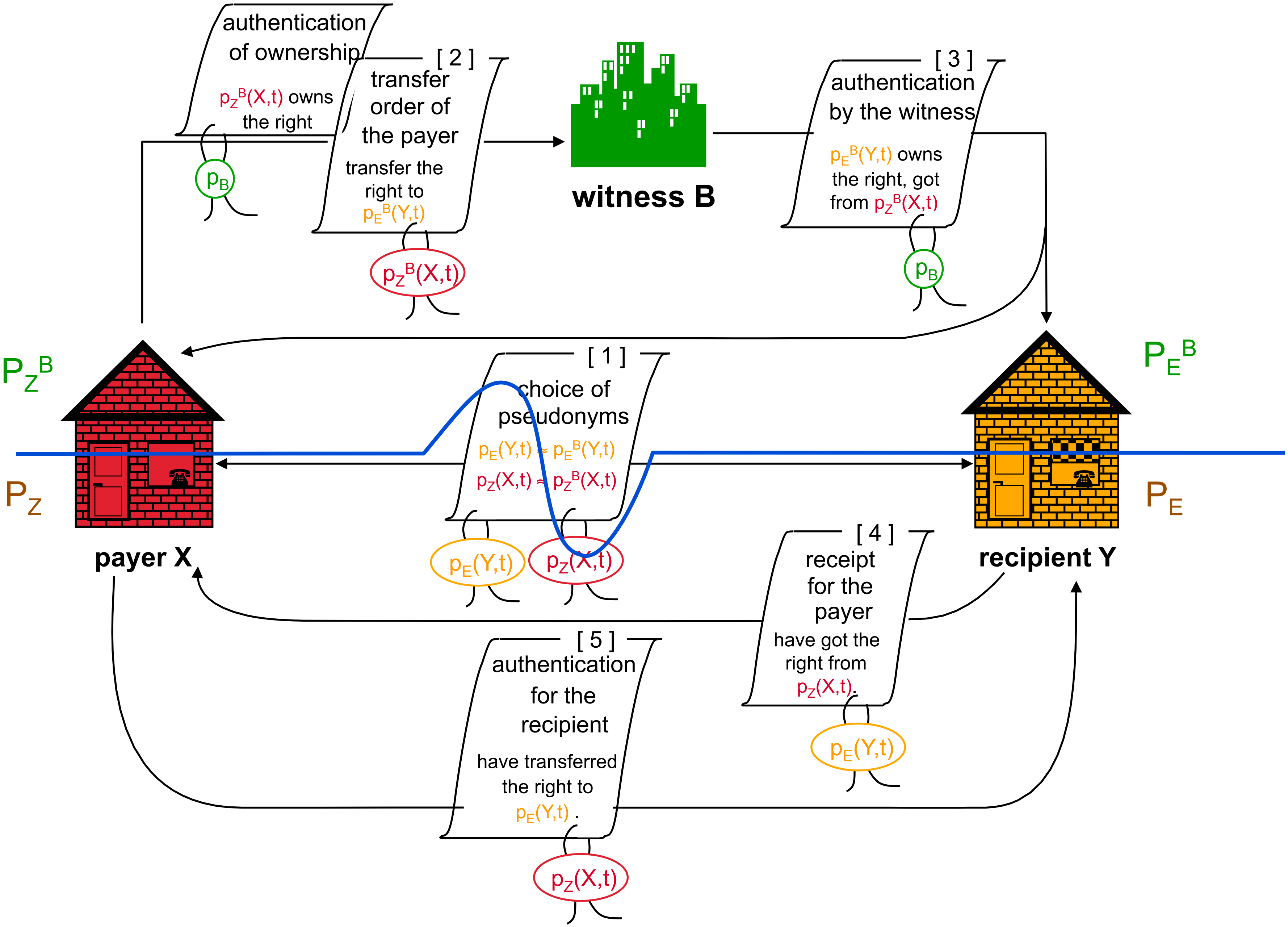


Anonymously transferable standard values

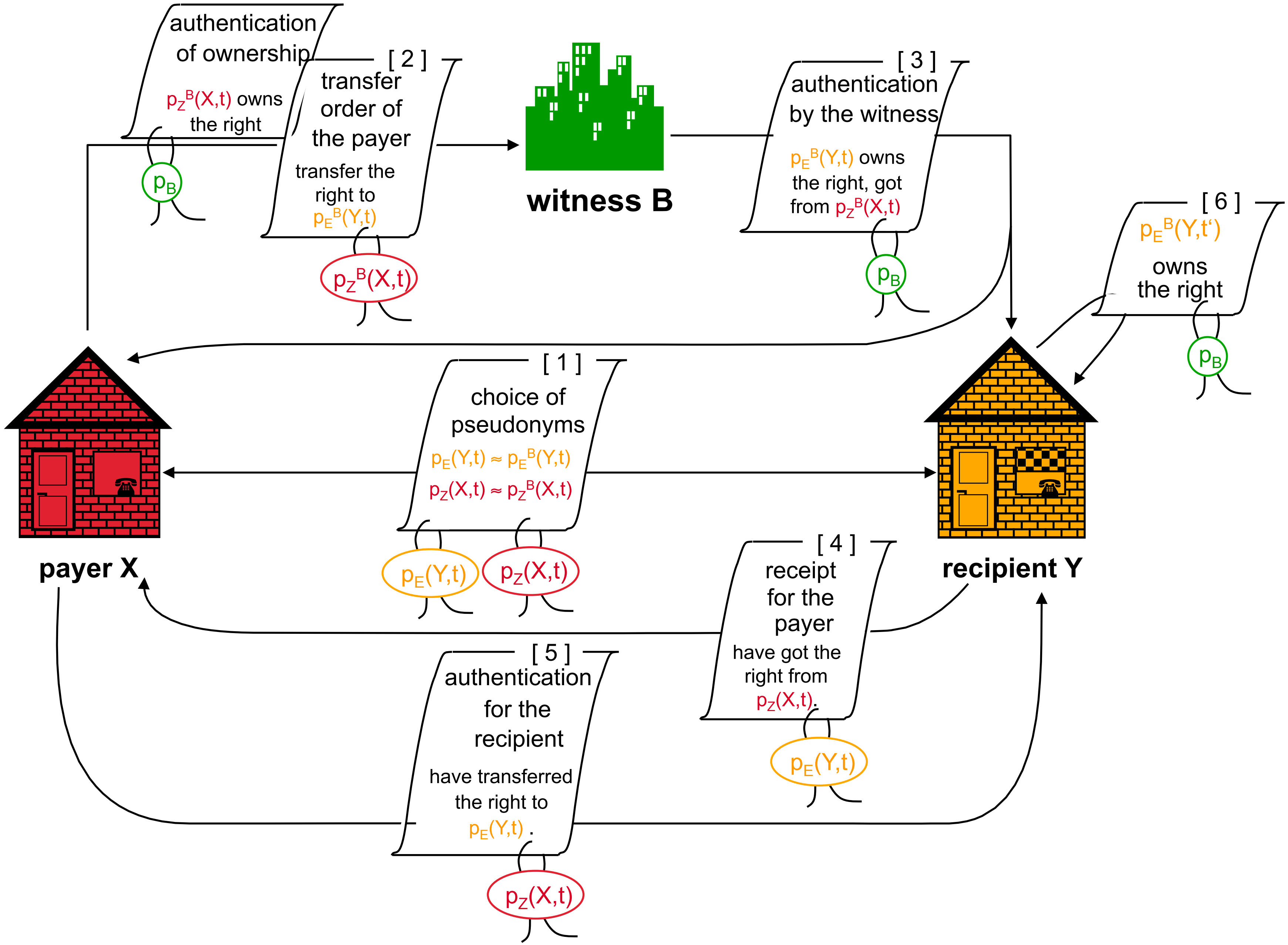


Anonymously transferable standard value

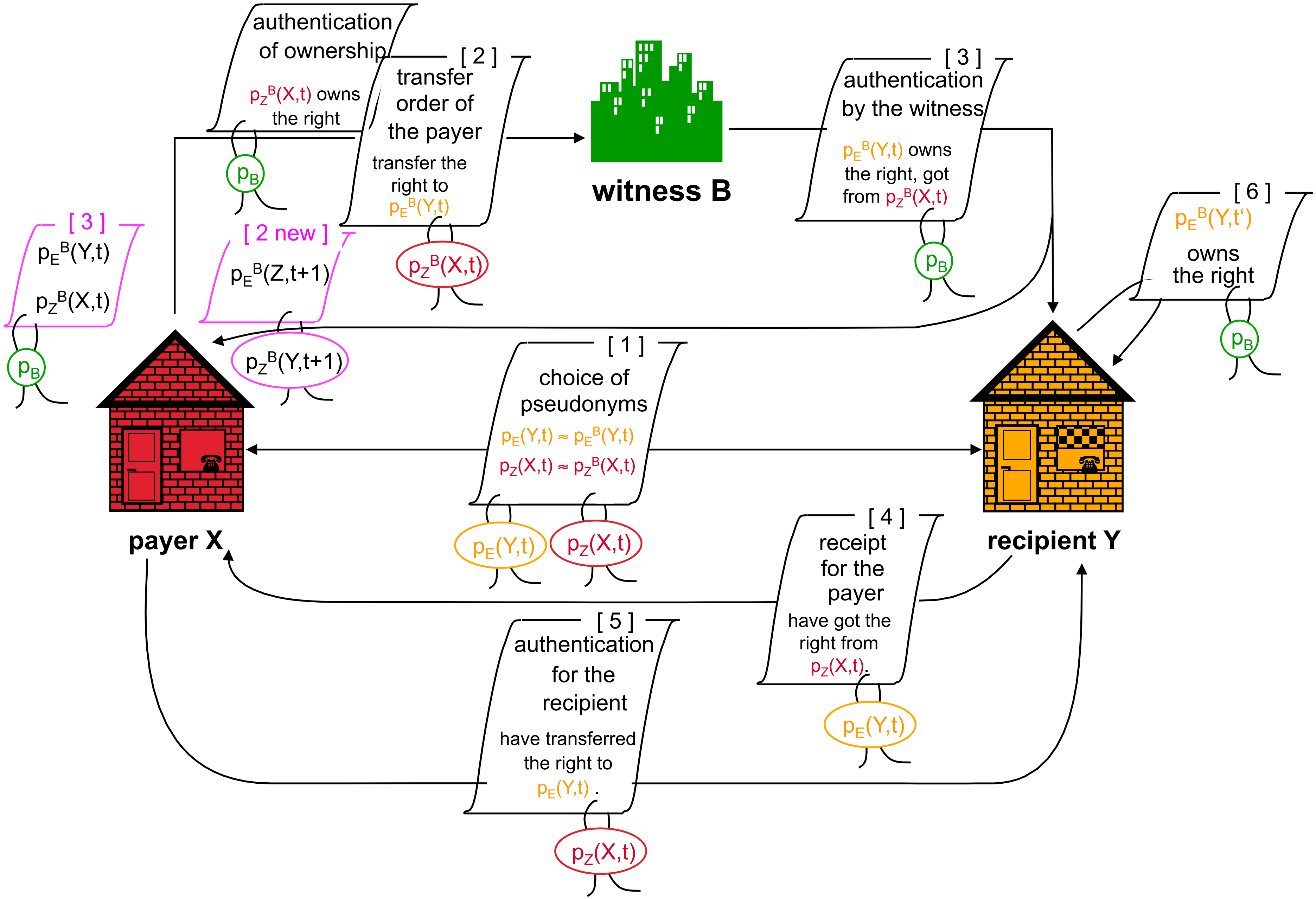
Basic scheme of a secure and anonymous digital payment system



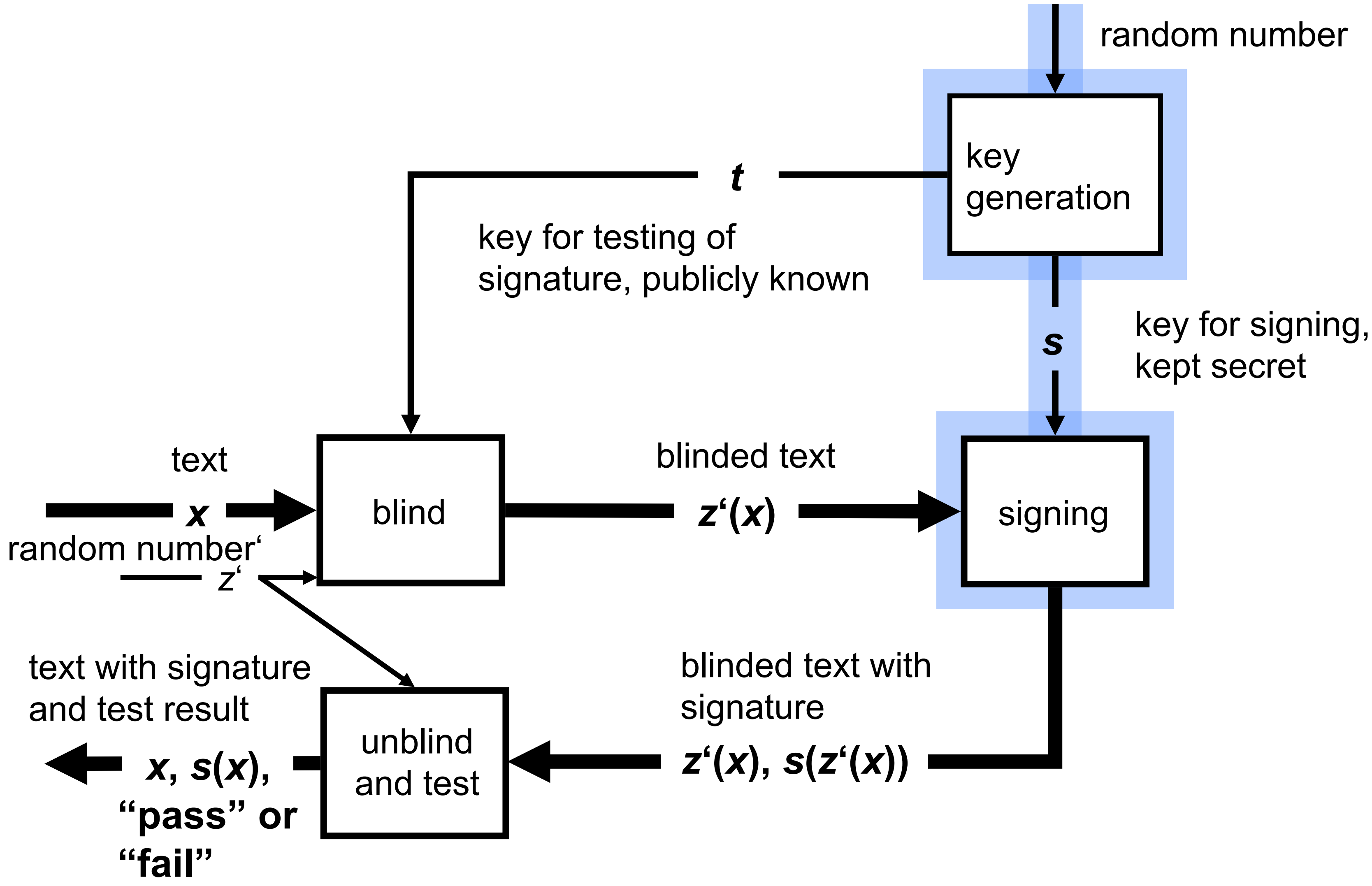
Transformation of the authentication by the witness



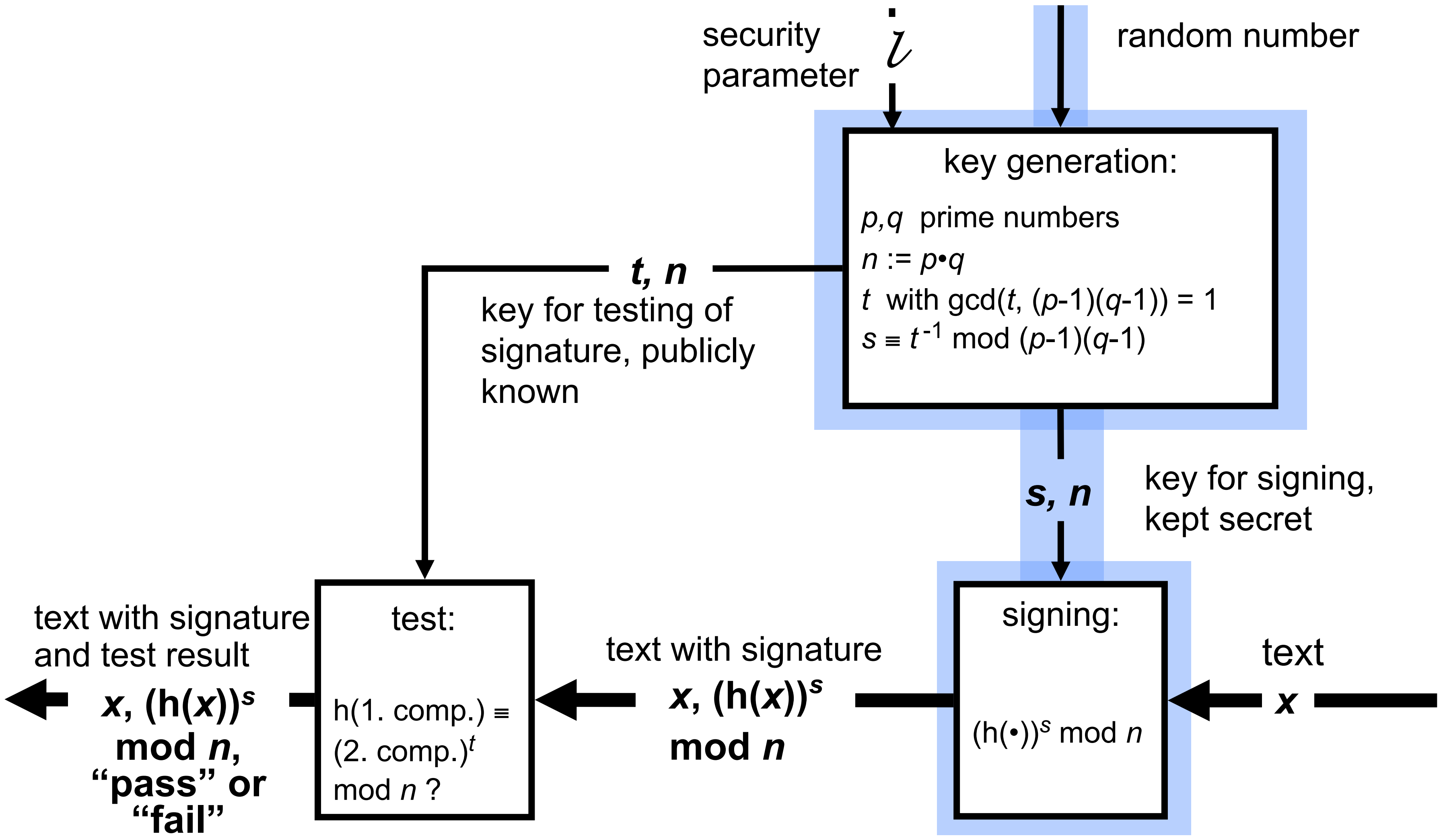
The next round: Y in the role payer to recipient Z



Signature system for signing blindly



RSA as digital signature system with collision-resistant hash function h



One time convertible authentication

Recipient

choose pseudonym

$\boldsymbol{p}$

(test key of arbitrary sign. system)

Collision-resistant hash function **h**

$\boldsymbol{p},\mathbf{h}(\boldsymbol{p})$

choose $\boldsymbol{r} \in_R Z_n^*$

$(\boldsymbol{p},\mathbf{h}(\boldsymbol{p}))\bullet\boldsymbol{r}^{t} \longrightarrow$

$(\boldsymbol{p},\mathbf{h}(\boldsymbol{p}))^{s}\bullet\boldsymbol{r} \longleftarrow$

multiply with

$\boldsymbol{r}^{-1}$

get

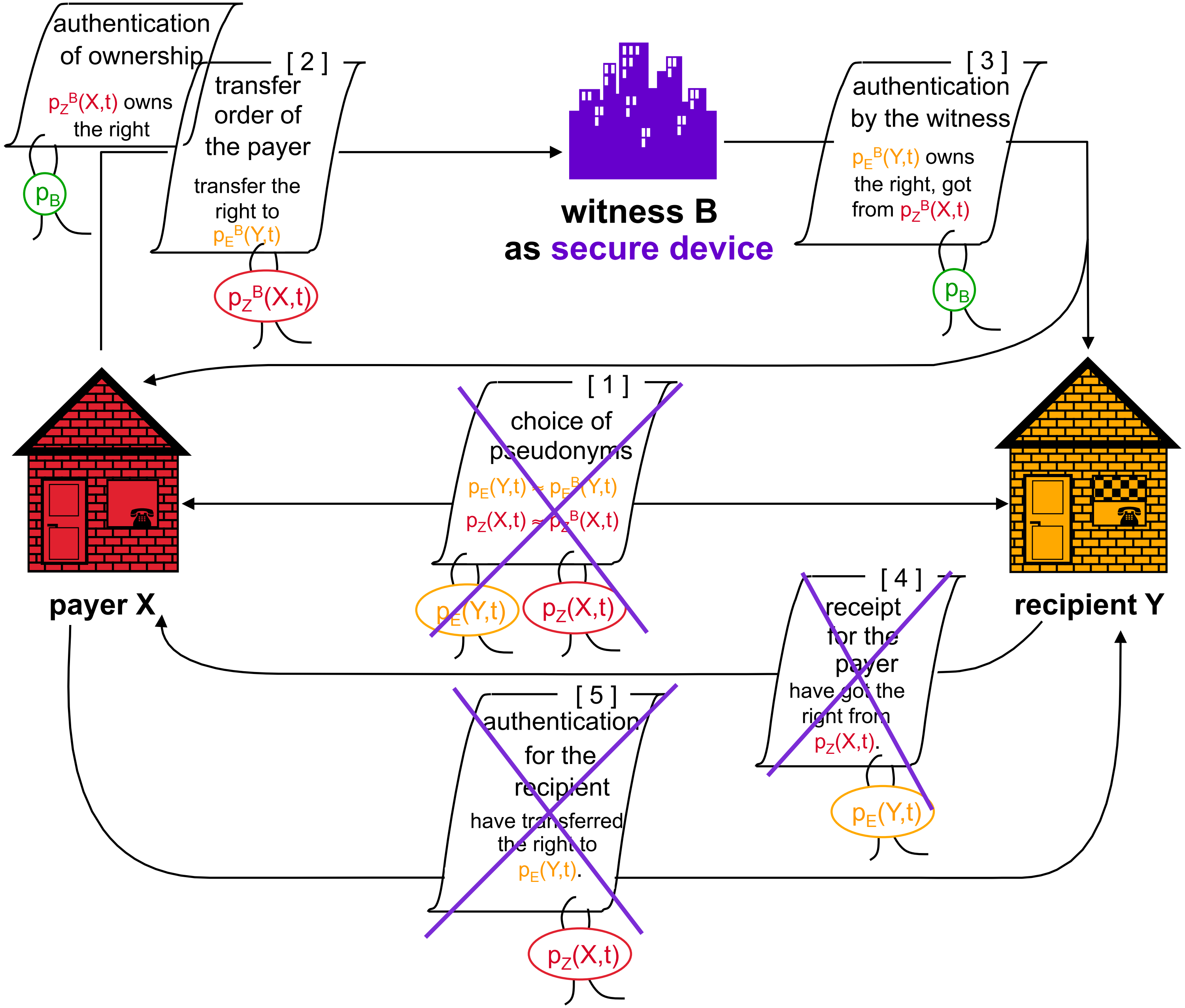
$(\boldsymbol{p},\mathbf{h}(\boldsymbol{p}))^{s}$

Issuer (i.e. witness)

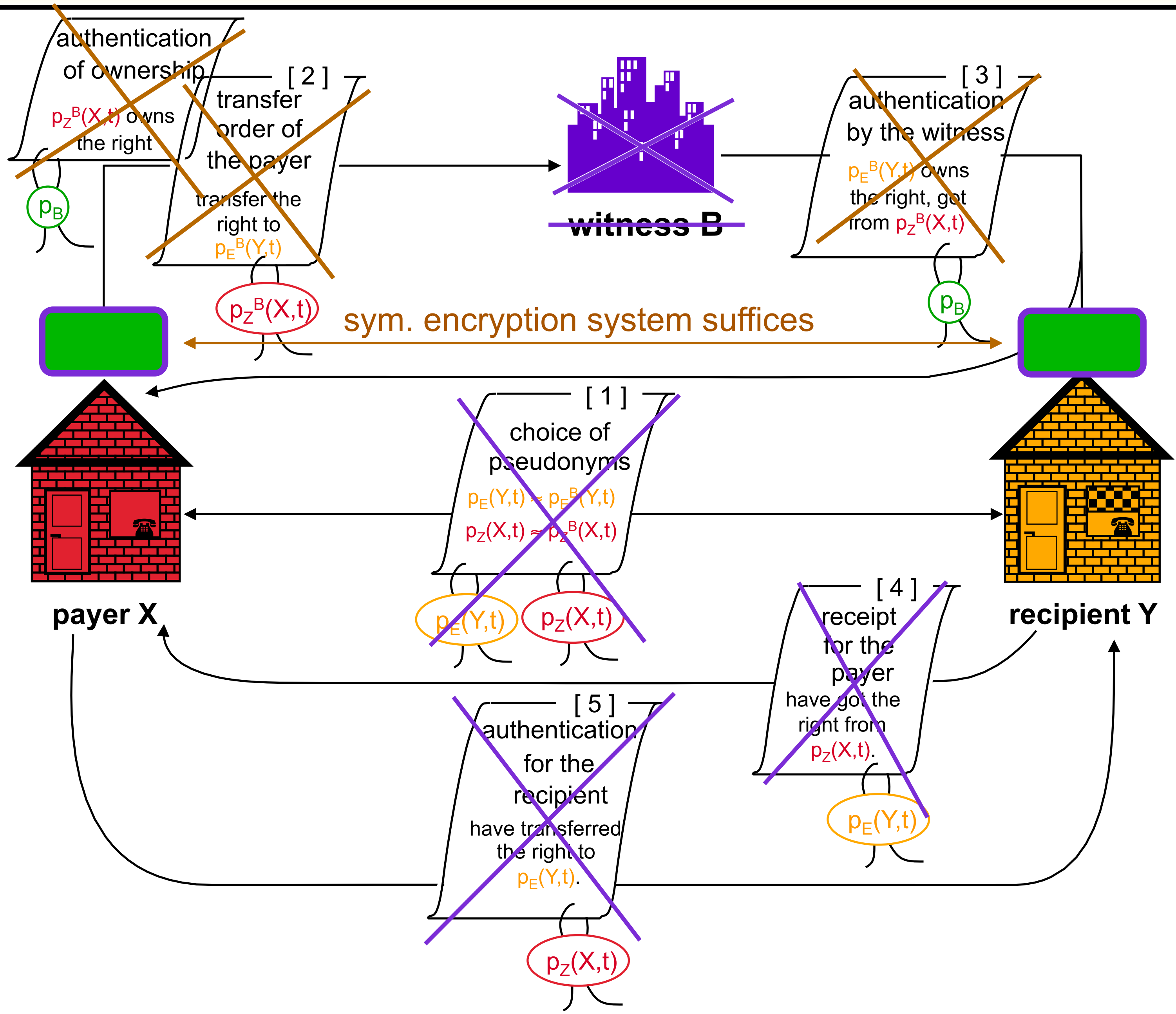
RSA test key ***t***,***n***, publicly known

$((\boldsymbol{p},\mathbf{h}(\boldsymbol{p}))\bullet\boldsymbol{r}^{t})^{s}$

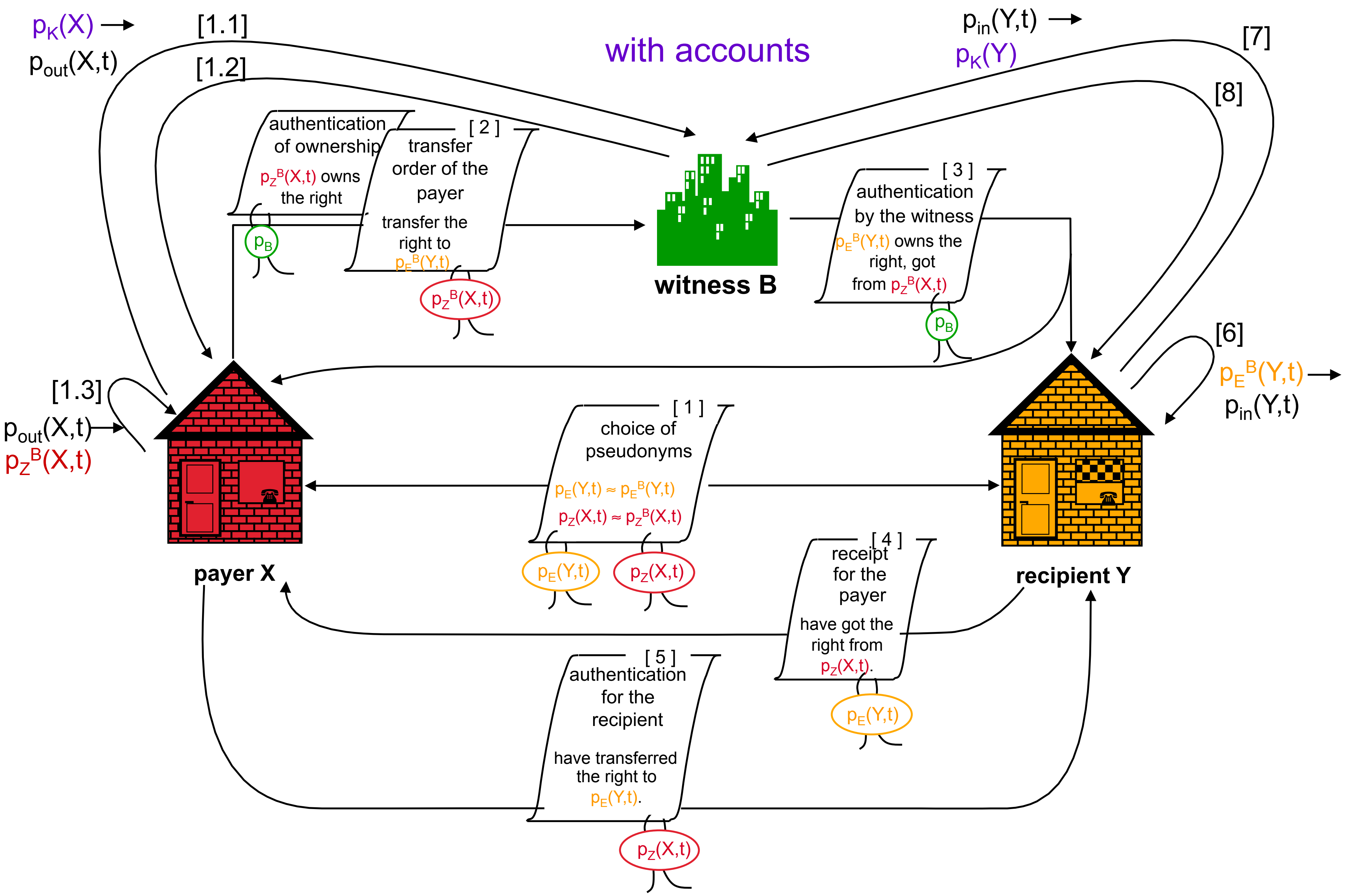
Secure device: 1st possibility



Secure device: 2nd possibility



Secure and anonymous digit. payment system with accounts



Offline payment system

Payment systems with security by Deanonymizability

k	security parameter
I	identity of the entity giving out the banknote
r_i	randomly chosen ($1 \leq i \leq k$)
C	commitment scheme with information theoretic secrecy

blindly signed banknote:

$$s_{Bank}(C(r_1), C(r_1 \oplus I), C(r_2), C(r_2 \oplus I), \ldots, C(r_k), C(r_k \oplus I)),$$

recipient decides, whether he wants to get revealed r_i **or** $r_i \oplus I$.
(one-time pad preserves anonymity.)

Hand-over to two honest recipients:
probability ($\exists i$: bank gets to know r_i and $r_i \oplus i$) $\geq 1-e^{-c \cdot k}$
(original owner identifiable)

Outlook

legal certainty vs. liability

online / offline

debit = pre-paid / pay-now / credit

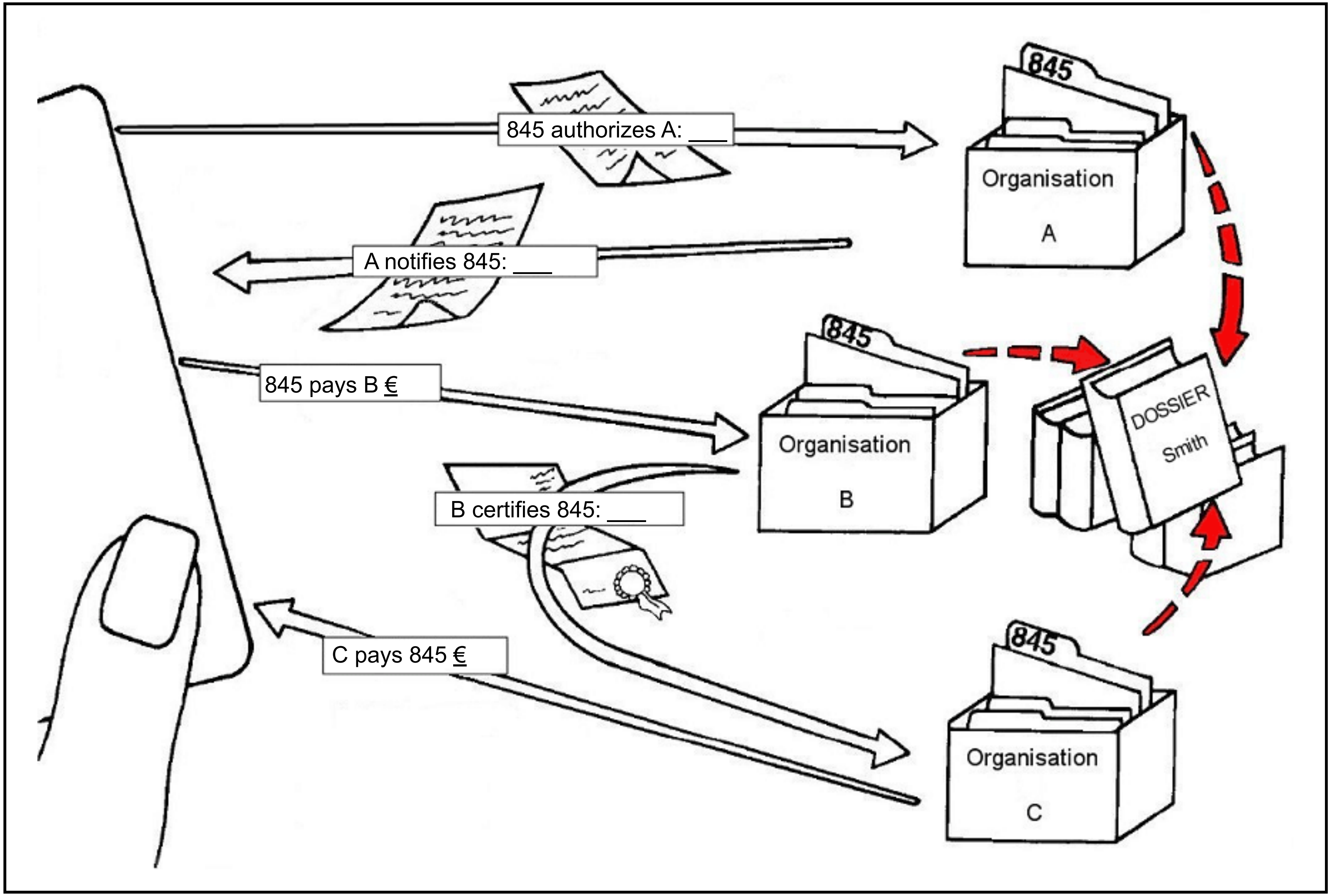
only special software or hardware, too ?

universal means of payment or multifaceted bonus systems ?

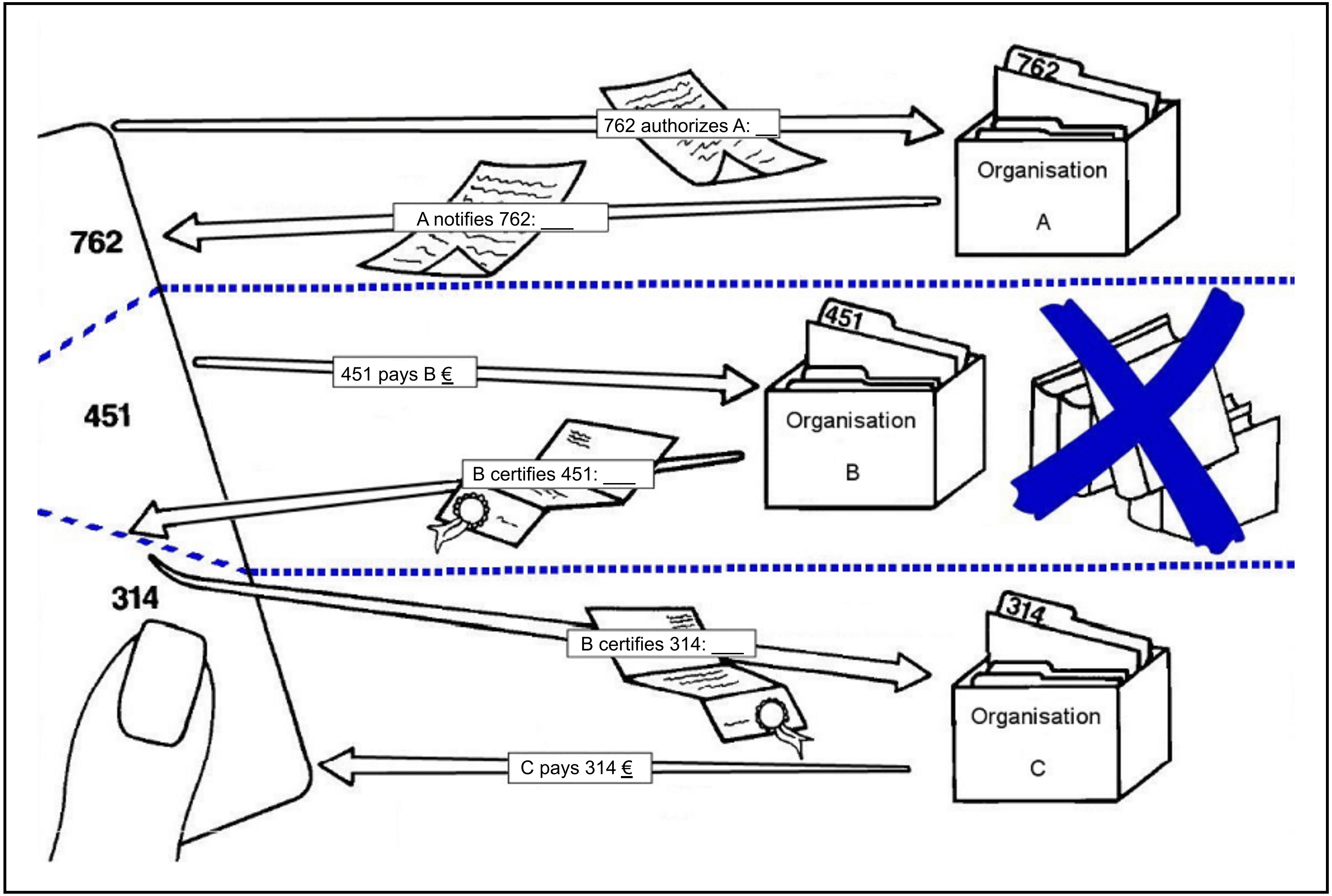
one or multiple currencies ?

one or multiple systems ?

Personal identifier



Role pseudonyms (business-relationship and transaction pseudonyms)



Multilateral security in digital payment systems

Identification in case of fraud using anonymous payment systems

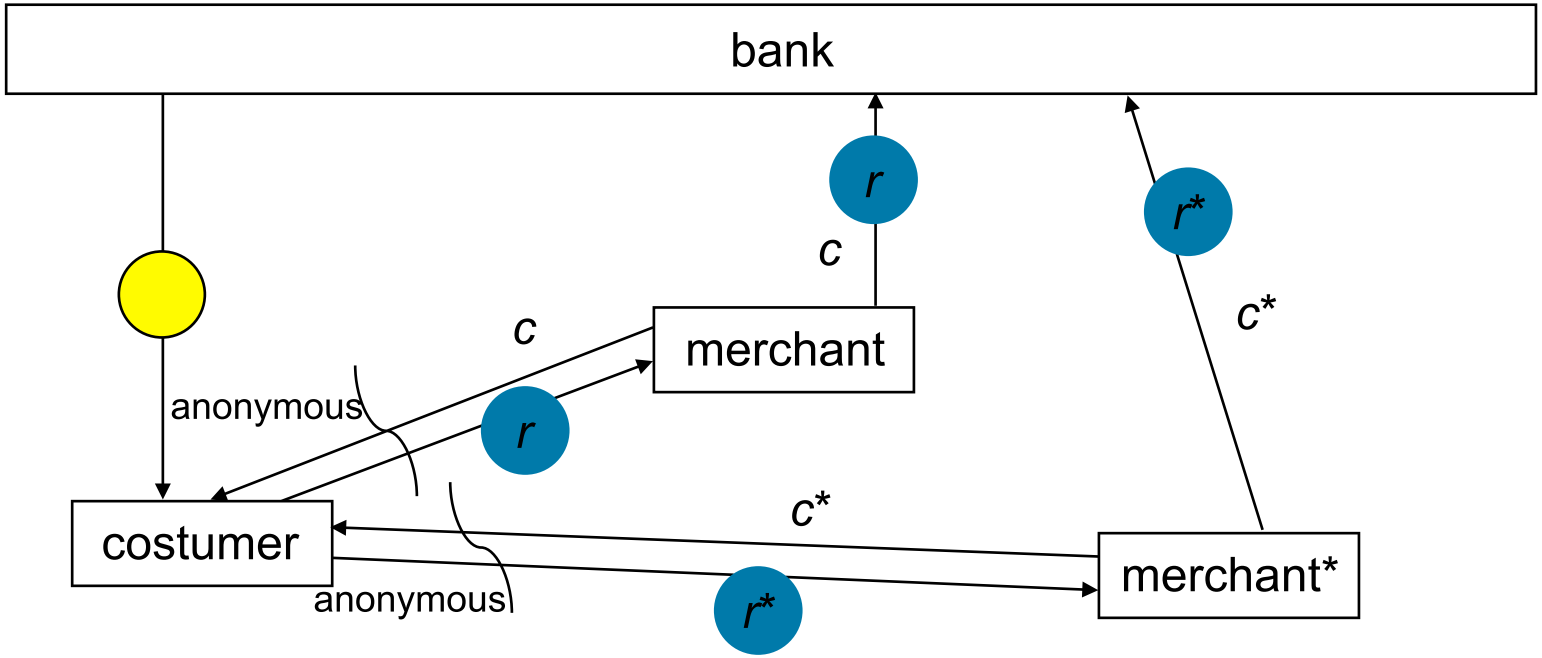


fig.: identification in case of fraud

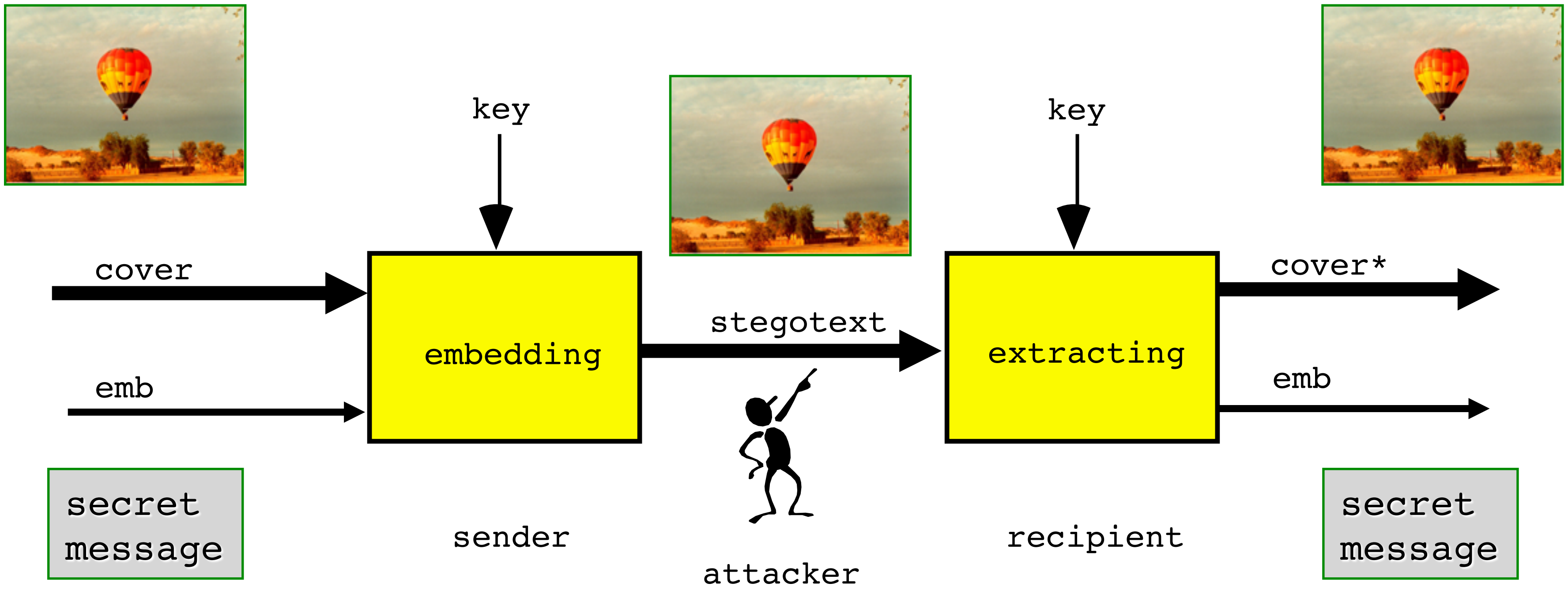
c, *c** challenges (with merchant ID)
r, *r** responses

conclusive identification of the costumer is possible using different responses to same digital coin

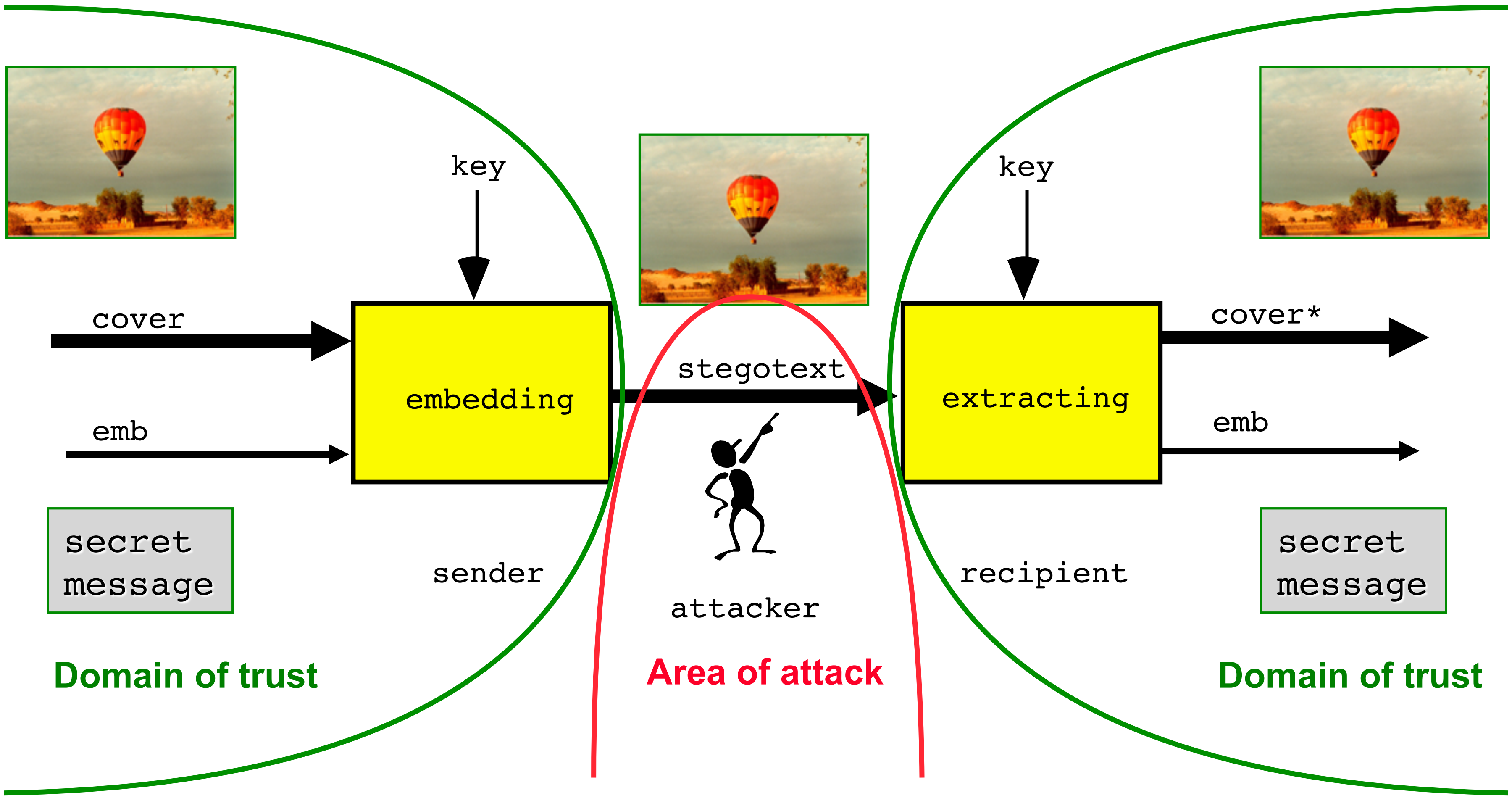
Cryptography and the impossibility of its legal regulation

- Cryptography *(you already know)*
- Steganography
- Proposals to regulate cryptography
- Technical limits of regulating cryptography
  - Secure digital signatures → Secure encryption
  - Key Escrow encryption without permanent surveillance → Encryption without Key Escrow
  - Symmetric authentication → Encryption
  - Multimedia communication → Steganography
  - Keys for communication and secret signature keys can be replaced at any time → Key Escrow to backup keys is nonsense
- Proposals to regulate cryptography harm the good guys only

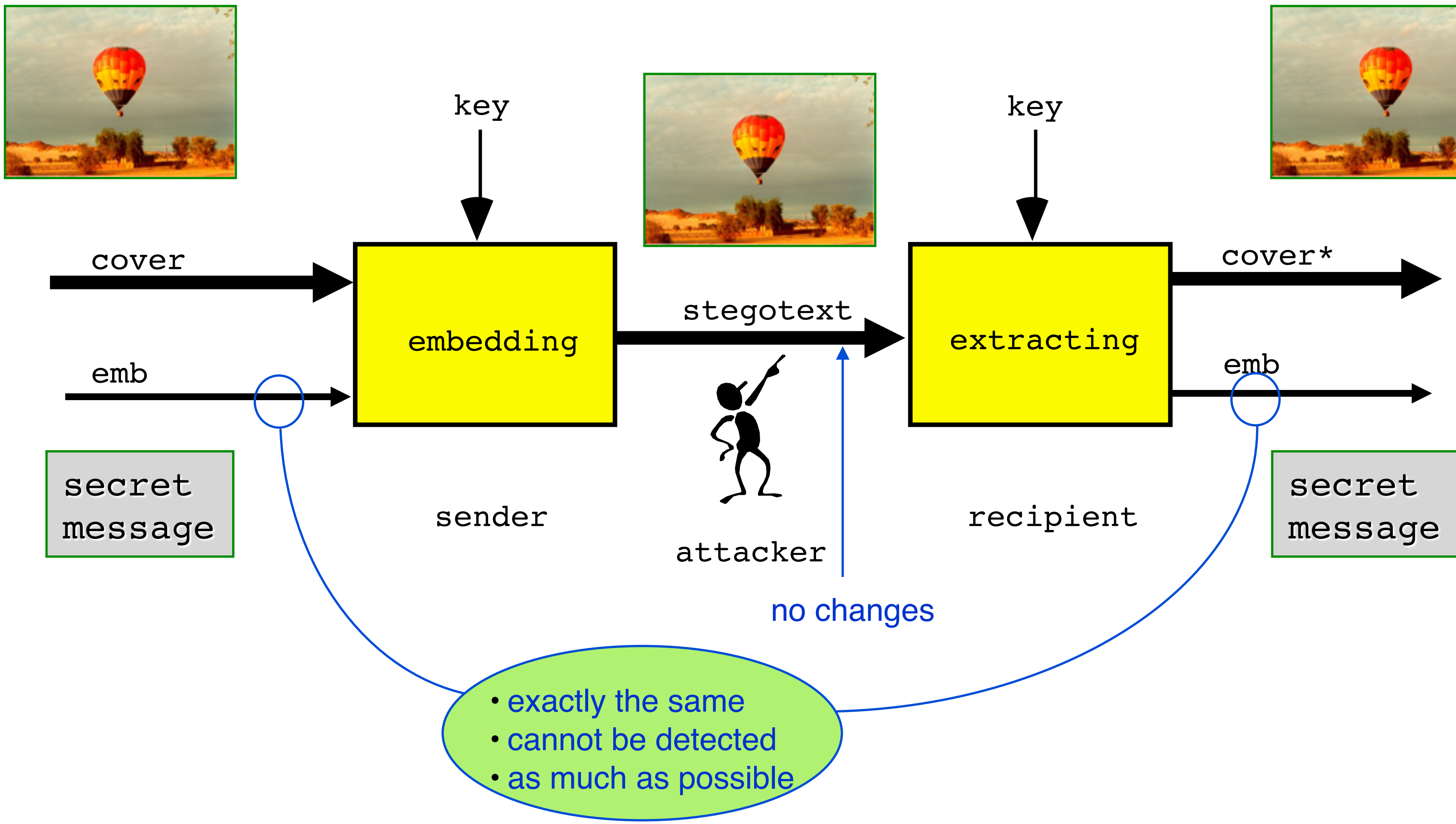
Steganography



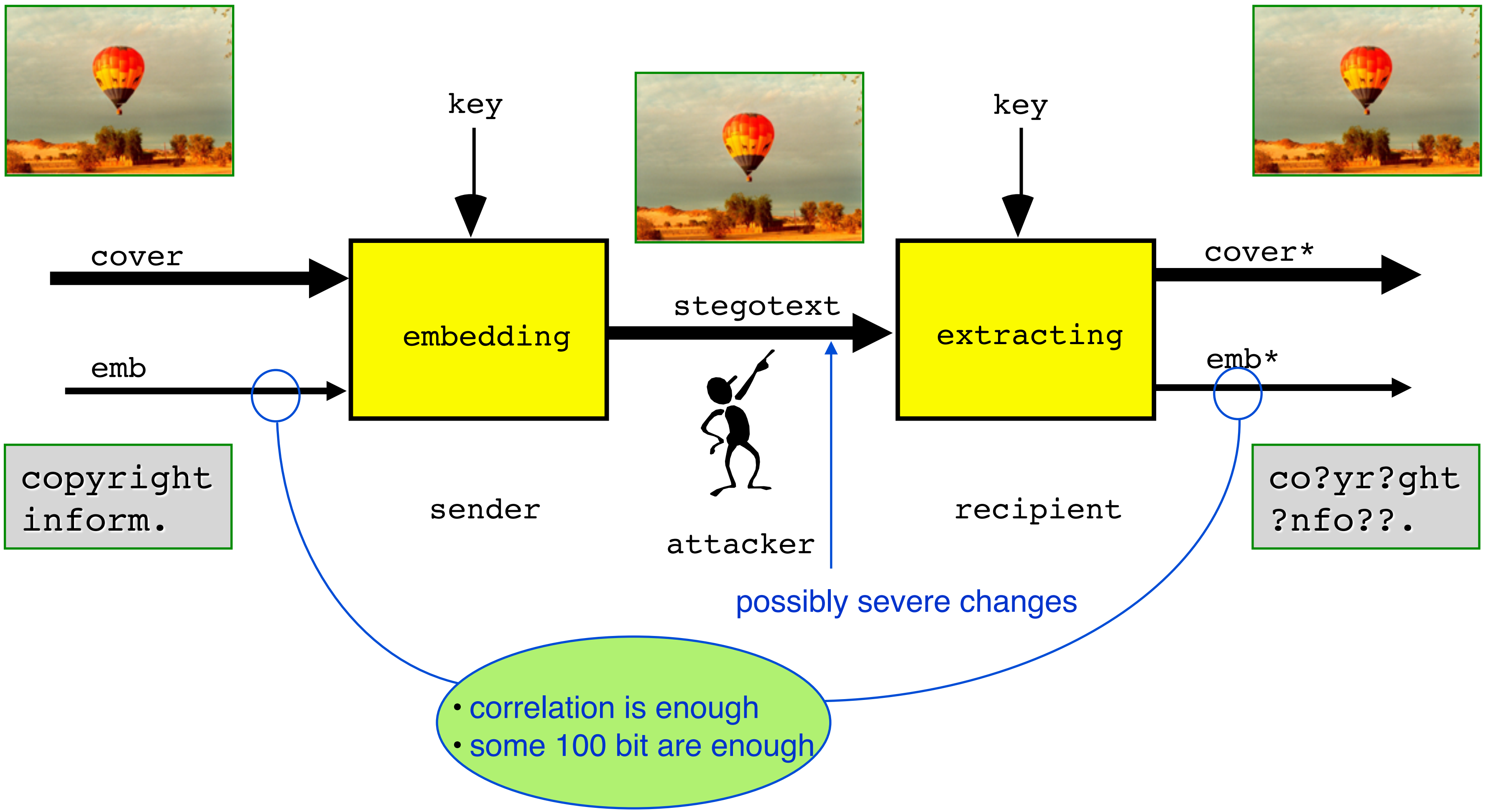
Steganography



Steganography: Secrecy of secrecy



Steganography: Watermarking and Fingerprinting



Proposals to regulate cryptography ?

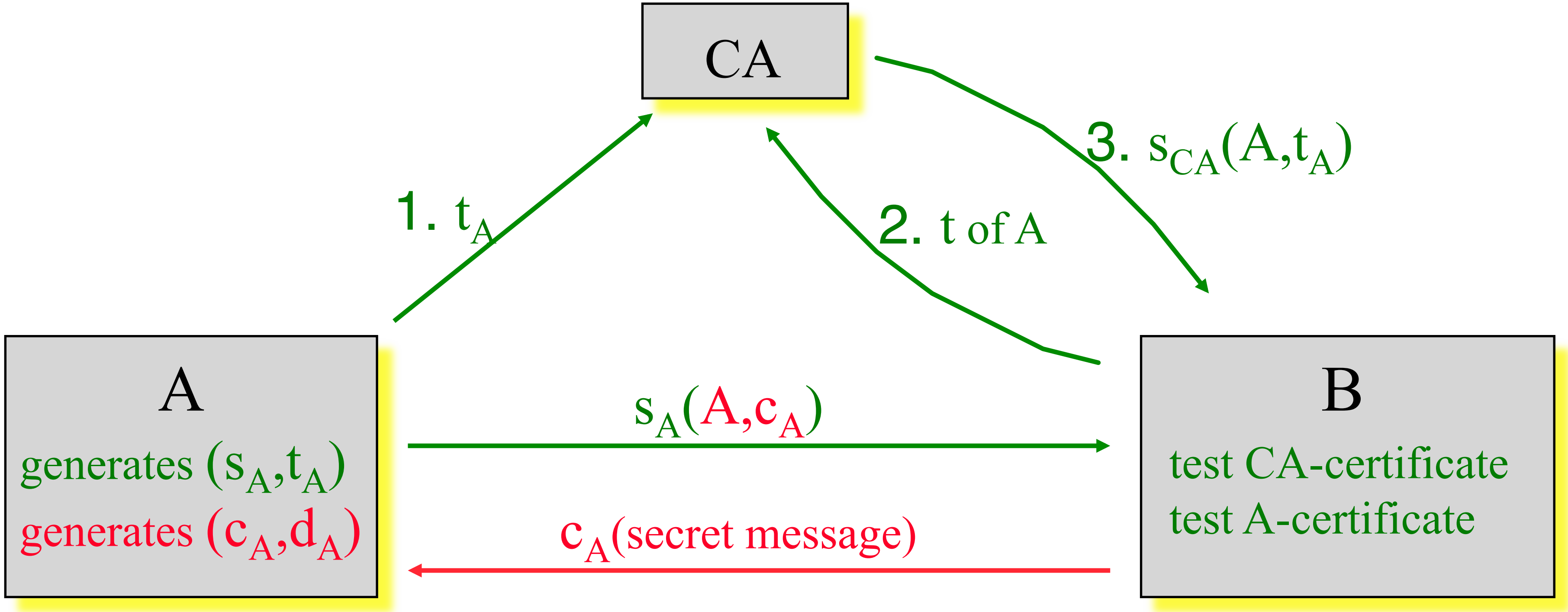
?

- Would you regulate cryptography to help fight crime ?
- If so: How ?

Proposals to regulate cryptography !

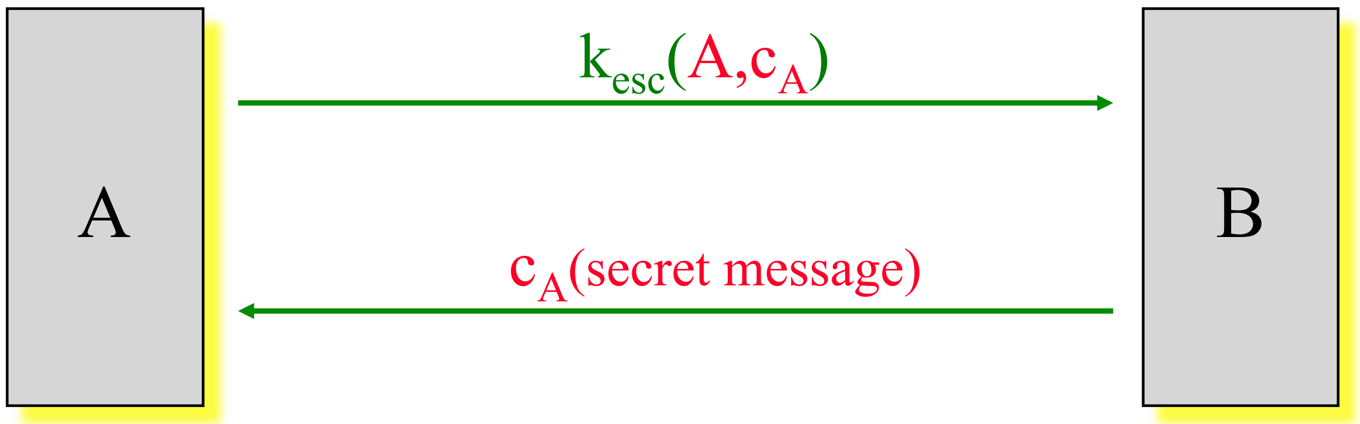
- Outlaw encryption
- Outlaw encryption – with the exception of small key lengths
- Outlaw encryption – with the exception of Key Escrow or Key Recovery systems
- Publish public encryption keys only within PKI if corresponding secret key is escrowed
- Obligation to hand over decryption key to law enforcement during legal investigation

Secure digital signatures —> Secure encryption



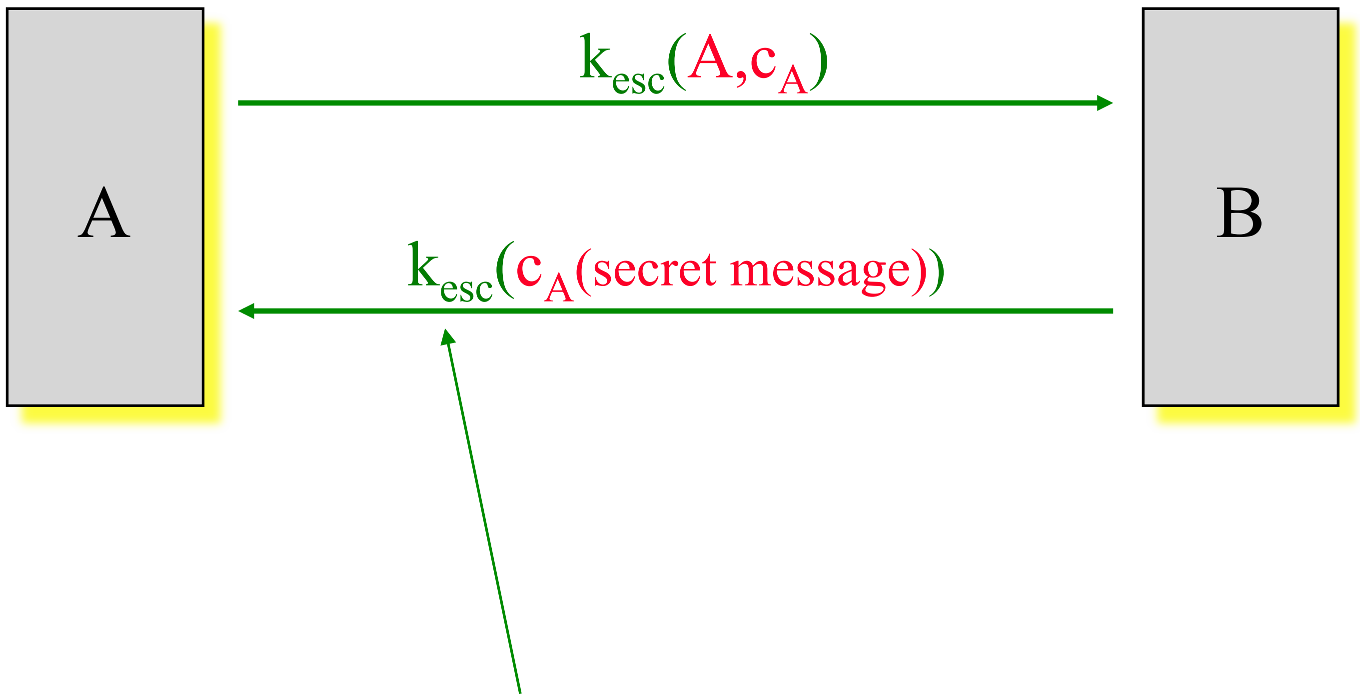
A does not need a certificate for c_A issues by CA

Key Escrow encryption without permanent surveillance



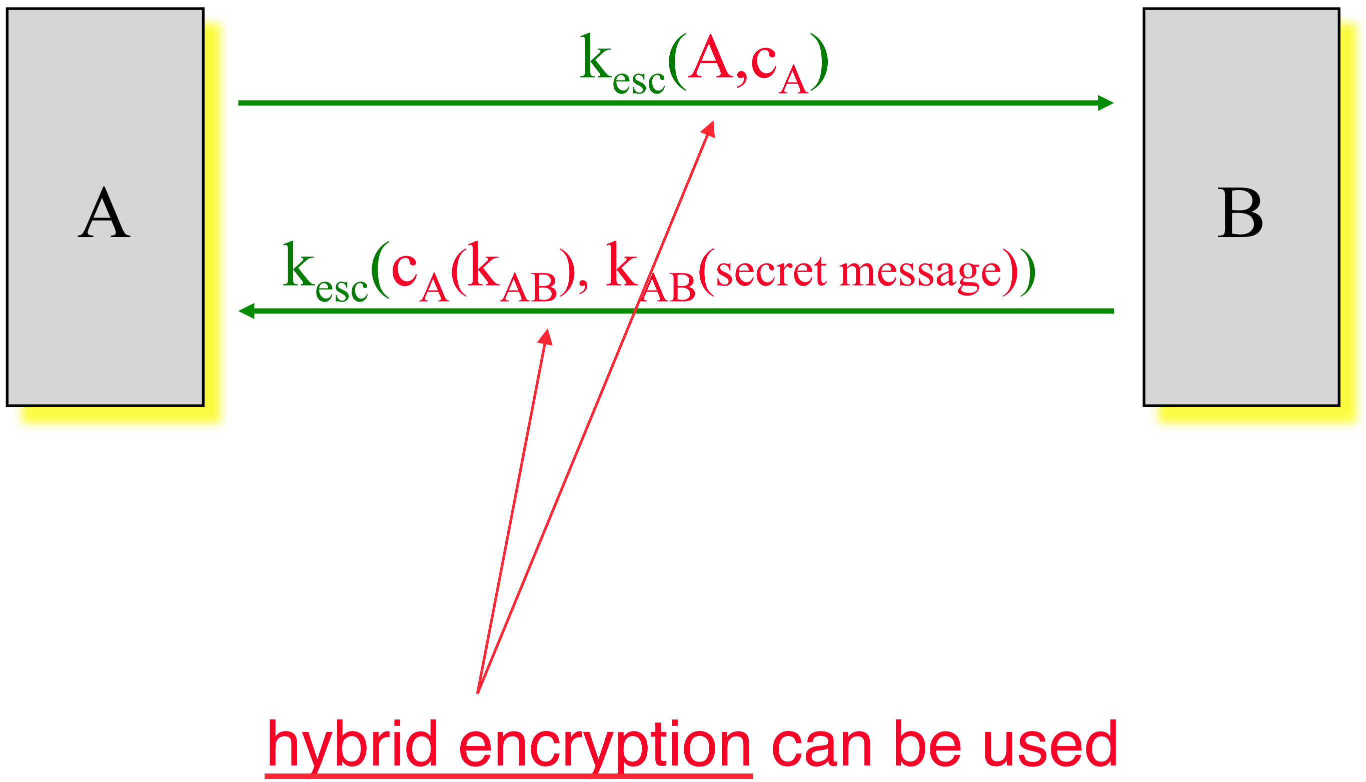
—> Encryption without Key Escrow

Key Escrow encryption without permanent surveillance

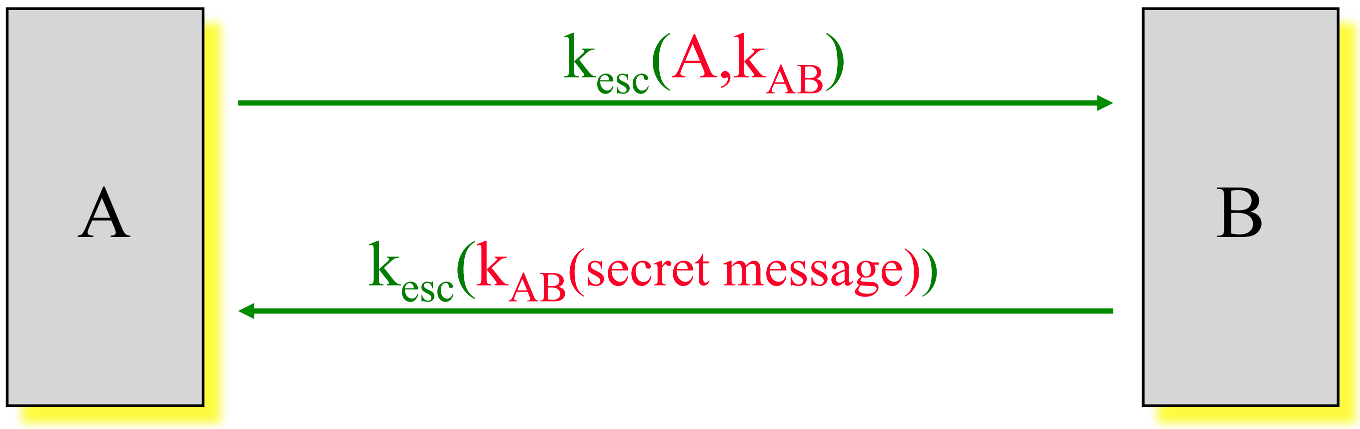


employ <u>Key Escrow additionally</u>
to keep your encryption without Key Escrow secret

Key Escrow encryption without permanent surveillance



Key Escrow encryption without permanent surveillance



if surveillance is not done or even cannot be done retroactively, symmetric encryption alone does the job

Symmetric authentication → Encryption

Sender A

Kennt k_{AB}

Zu übertragen sei Nachricht
$b_1, \dots b_n$ mit $b_i \in \{0, 1\}$

Berechnet
$\mathrm{MAC}_1 := \mathrm{code}(k_{AB}, b_1) \dots \mathrm{MAC}_n := \mathrm{code}(k_{AB}, b_n)$

Sei $a_1, \dots a_n$ die bitweise invertierte Nachricht.

Wählt zufällig $\mathrm{MAC'}_1 \dots \mathrm{MAC'}_n$ mit
$\mathrm{MAC'}_1 \neq \mathrm{code}(k_{AB}, a_1) \dots \mathrm{MAC'}_n \neq \mathrm{code}(k_{AB}, a_n)$

form

falsely authenticated messages

Überträgt (die Mengenklammern bedeuten „zufällige Reihenfolge")
$\{(b_1, \mathrm{MAC}_1), (a_1, \mathrm{MAC'}_1)\} \dots$
$\{(b_n, \mathrm{MAC}_n), (a_n, \mathrm{MAC'}_n)\}$ ——————>

intermingle

Empfänger B

Kennt k_{AB}

Probiert, ob
$\{\mathrm{MAC}_1 = \mathrm{code}(k_{AB}, b_1)$ oder
$\mathrm{MAC'}_1 = \mathrm{code}(k_{AB}, a_1)\}$
und empfängt den passenden Wert b_1
...
probiert, ob
$\{\mathrm{MAC}_n = \mathrm{code}(k_{AB}, b_n)$ oder
$\mathrm{MAC'}_n = \mathrm{code}(k_{AB}, a_n)\}$
und empfängt den passenden Wert b_n

separate

Ronald L. Rivest: Chaffing and Winnowing: Confidentiality without Encryption; MIT Lab for Computer Science, March 22, 1998; http://theory.lcs.mit.edu/~rivest/chaffing.txt

Symmetric authentication → Encryption

Sender *A*

Kennt k_{AB}

Zu übertragen sei Nachricht
$b_1, \ldots b_n$ mit $b_i \in \{0, 1\}$

Berechnet
$\mathrm{MAC}_1 := \mathrm{code}(k_{AB}, b_1) \ldots \mathrm{MAC}_n := \mathrm{code}(k_{AB}, b_n)$

Überträgt
$(1, b_1, \mathrm{MAC}_1), \ldots (n, b_n, \mathrm{MAC}_n)$

Empfänger *B*

Kennt k_{AB}

falsely authenticated messages

Komplementgenerierer

Hört die Nachricht $b_1, \ldots b_n$ ab.

Bildet $a_1, \ldots a_n$, die bitweise invertierte Nachricht.
Wählt zufällig $\mathrm{MAC'}_1 \ldots \mathrm{MAC'}_n$ und mischt in
den Nachrichtenstrom von Sender A
an die passenden Stellen
$(1, a_1, \mathrm{MAC'}_1), \ldots (n, a_n, \mathrm{MAC'}_n)$

Überträgt die Mischung

form and intermingle
without knowing the key

separate

normales Authentikationsprotokoll
Ignoriert Nachrichten mit falscher Seque
Ignoriert Nachrichten mit falscher Authe
gibt die übrigbleibenden weiter
empfangen wird mit größter Wahrschein
$b_1, \ldots b_n$

Abhörer
kann a_i und b_i nicht unterscheiden

Key exchange for steganography ?

Exchanging keys outside the communication network is easy for **small closed groups**, in particular it is easy for criminals and terrorists.

Large open groups need a method of key exchange which works without transmitting suspicious messages within the communication network – asymmetric encryption cannot be used directly for key exchange.

Solution:

Diffie-Hellman Public-Key Agreement

Uses public keys of a commonly used digital signature systems (DSS, developed and standardized by NSA and NIST, USA)

Key exchange without message exchange

Diffie-Hellman Public-Key Agreement

secret: x y

public: g^x g^y

$$(g^y)^x = g^{yx} = g^{xy} = (g^x)^y$$

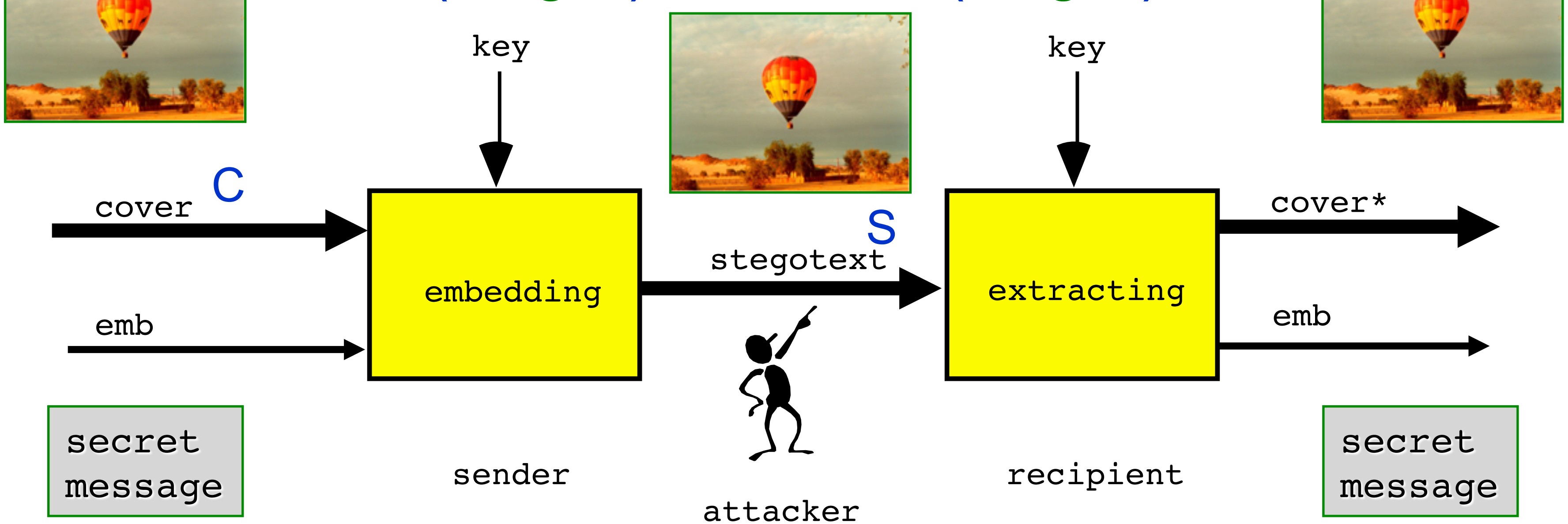
Key exchange for steganography !

Diffie-Hellman Public-Key Agreement

secret:	x	y
public:	g^x	g^y

$$(g^y)^x = g^{yx} = g^{xy} = (g^x)^y$$

$$f(C, g^{yx}) = f(S, g^{xy})$$

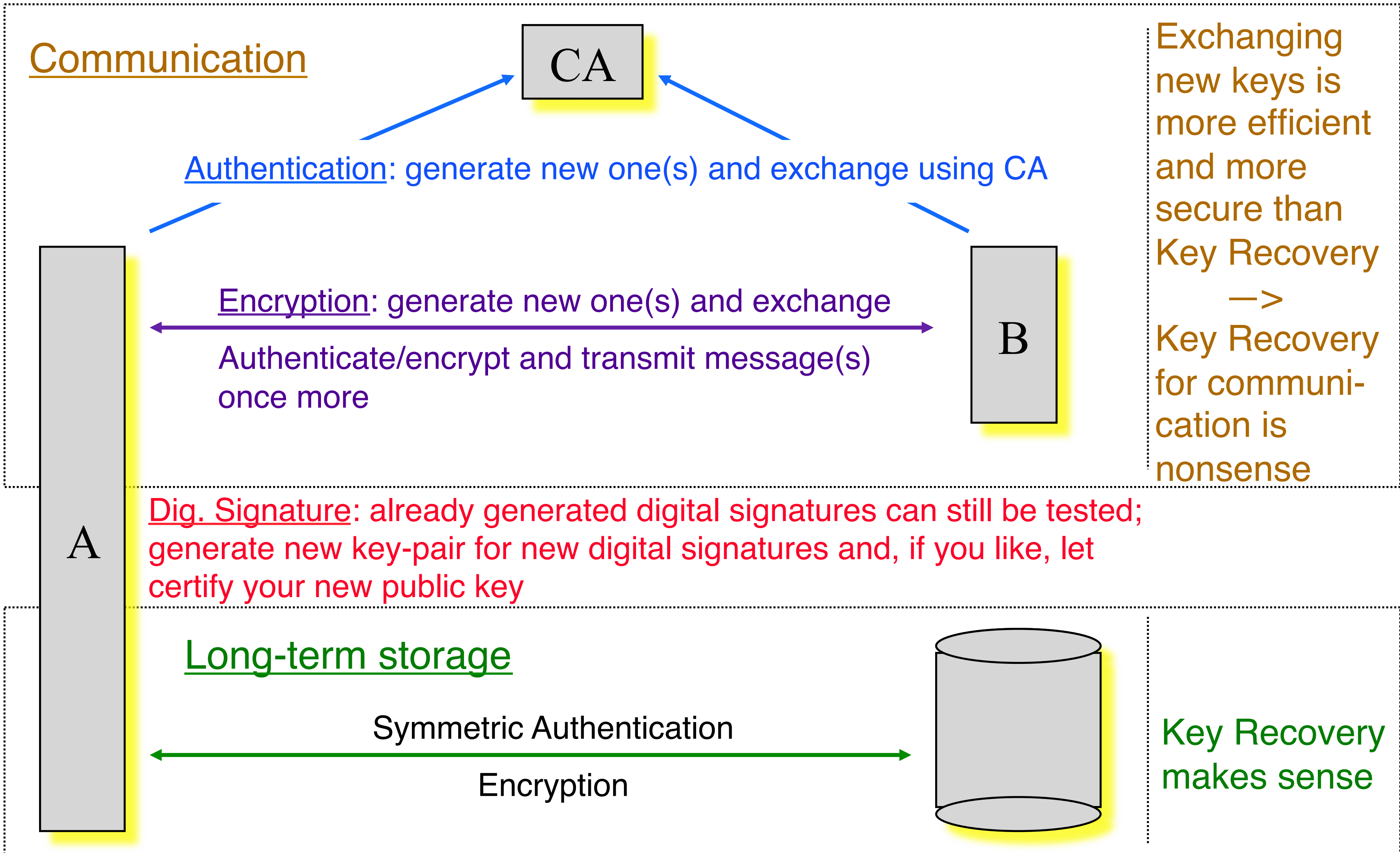


Summary

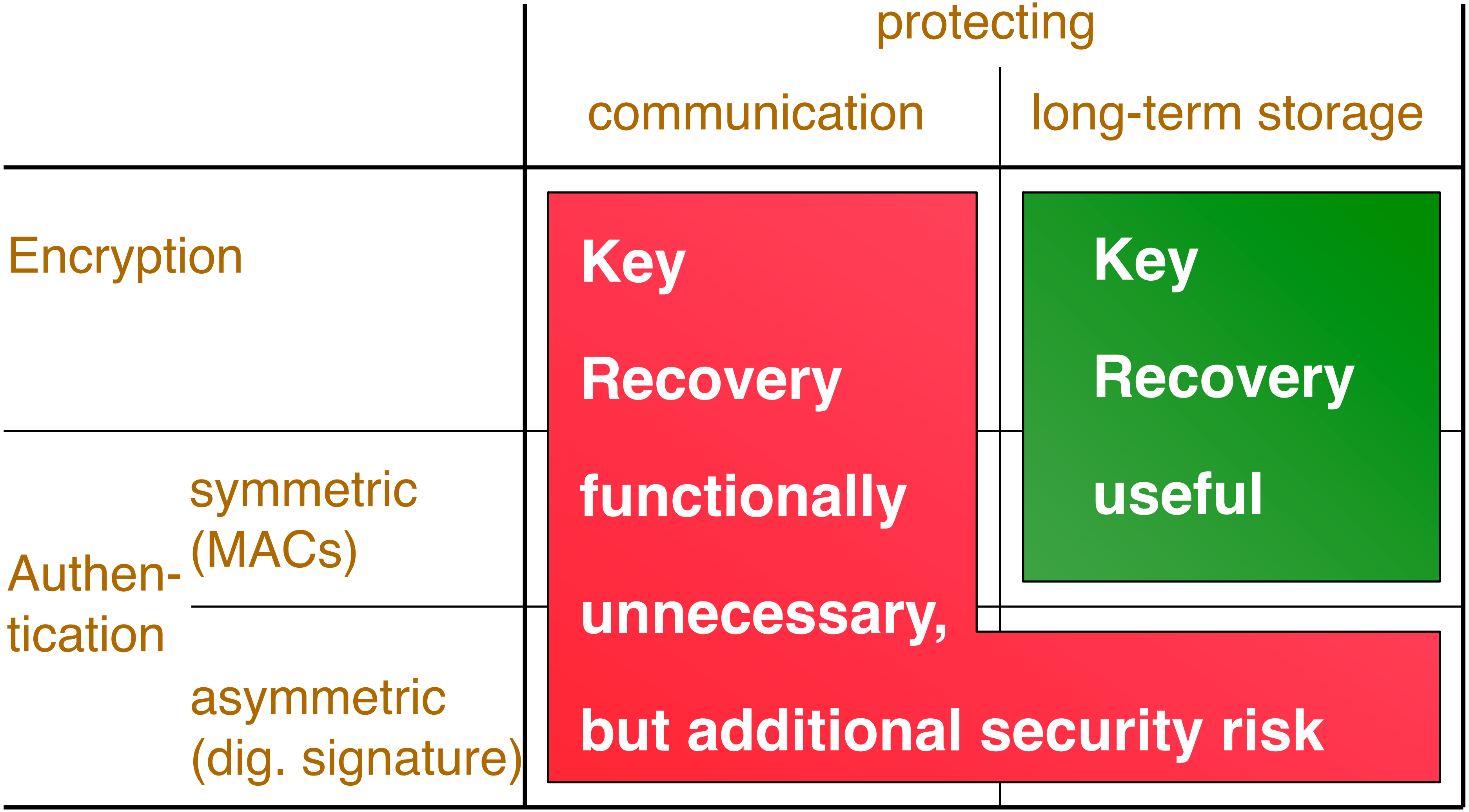
Digital Signatures	→	Encryption
Key Escrow without permanent surveillance	→	Key exchange, multiple encryption
Multimedia communication	→	Steganography

Cryptoregulation ignores technical constraints

Loosing secret keys



Key Recovery – for which keys ?



Proposals to regulate cryptography harm the good guys only

- Outlaw encryption
- Outlaw encryption – with the exception of small key lengths
- Outlaw encryption – with the exception of Key Escrow or Key Recovery systems
- Publish public encryption keys only within PKI if corresponding secret key is escrowed
- Obligation to hand over decryption key to law enforcement during legal investigation

➢ Steganography

➢ In addition steganography

➢ Use Key Escrow or Key Recovery system for bootstrap

➢ Run PKI for your public encryption keys yourself

➢ Calculate one-time-pad accordingly

(Im-)Possibility to regulate anonymous/pseudonymous communication

- Explicit techniques *(you already know the theory)*
- Workarounds

(Im-)Possibility to regulate anonymous/pseudonymous communication

Anon-Proxies

MIXes

- **Cascade: AN.ON**
- **P2P: TOR**

All this exists abroad without regulation – as long as we do not have a global home policy

(Im-)Possibility to regulate anonymous/pseudonymous communication

But even domestic:

- Public phones,
- Prepaid phones,
- open unprotected WLANs,
- insecure Bluetooth mobile phones,
- ...

Data retention is nearly nonsense,

since „criminals“ will use workarounds, cf. above