Replay Protection

Levente Buttyán

CrySyS Lab, BME

buttyan@crysys.hu

Message sequence numbers

- replay protection is often ensured by sequence numbering messages, and requiring that sequence numbers of received messages are monotonically increasing

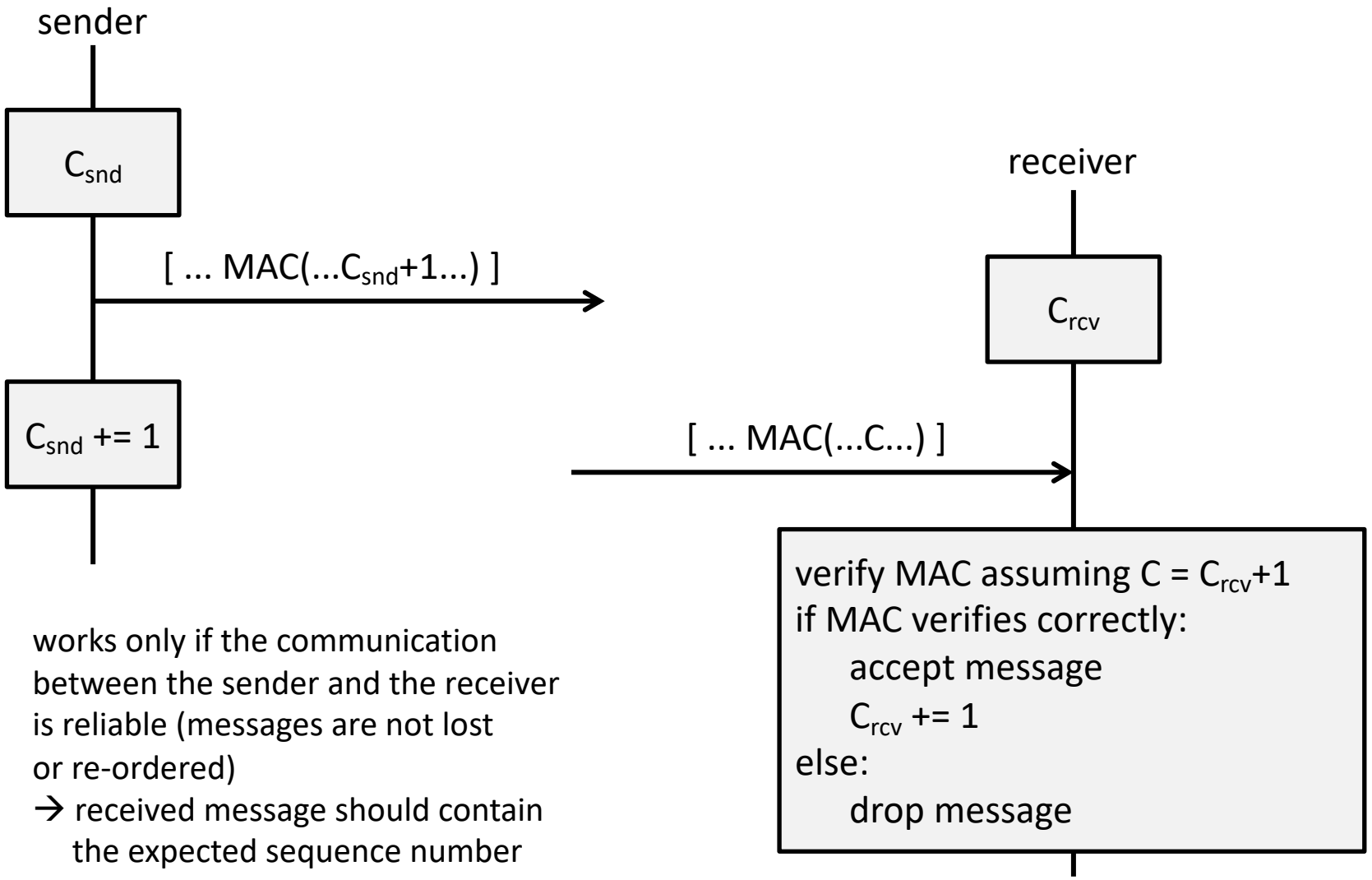
- the sender maintains a ”sending” sequence number C_{snd}
  - stored locally by the sender
  - its value is the sequence number of the last successfully sent message
  - its initial value can be 0
  - the next message is sent with sequence number $C_{snd}+1$
    - » MAC calculation for the message should depend on $C_{snd}+1$
    - » $C_{snd}+1$ may be explicitly included in a message header
  - if message is sent successfully, C_{snd} is incremented

Message sequence numbers

- the receiver maintains a ”receiving” sequence number C_{rcv}
  - stored locally by the receiver
  - its value is the sequence number of the last message received
  - its initial value can be 0
  - next message is accepted only if its sequence number is larger than C_{rcv}
  - acceptance of the message usually also depends on other factors (e.g., success of MAC verification)
  - if message is finally accepted, C_{rcv} is set to the sequence number of the message
  - otherwise, C_{rcv} is not changed

- for bi-directional communication, the parties need to maintain different sequence numbers in the two directions
  - both parties will have a ”sending” and a ”receiving” sequence number

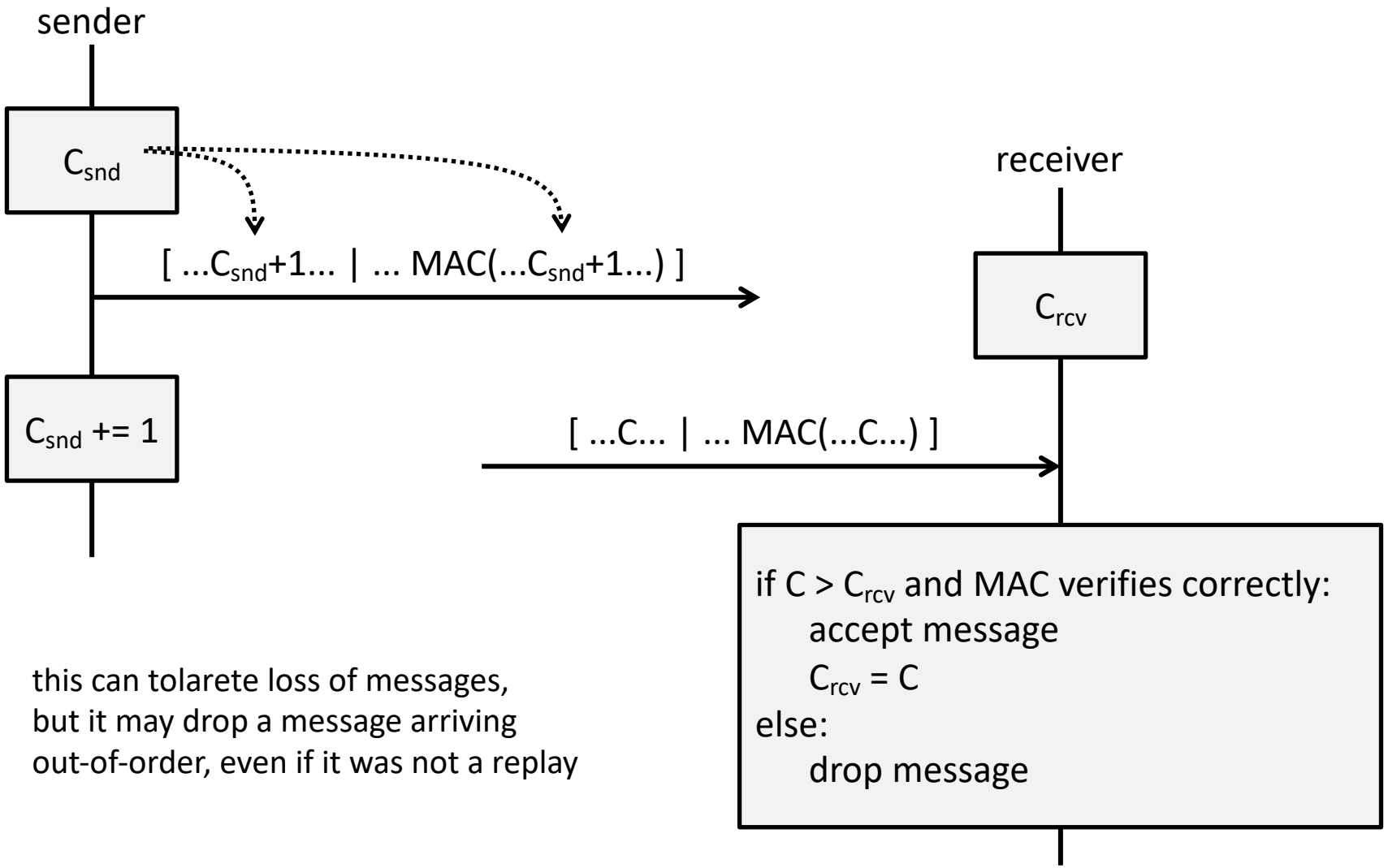
Implicit sequence numbering



works only if the communication between the sender and the receiver is reliable (messages are not lost or re-ordered)

→ received message should contain the expected sequence number

Explicit sequence numbering



this can tolarete loss of messages,
but it may drop a message arriving
out-of-order, even if it was not a replay

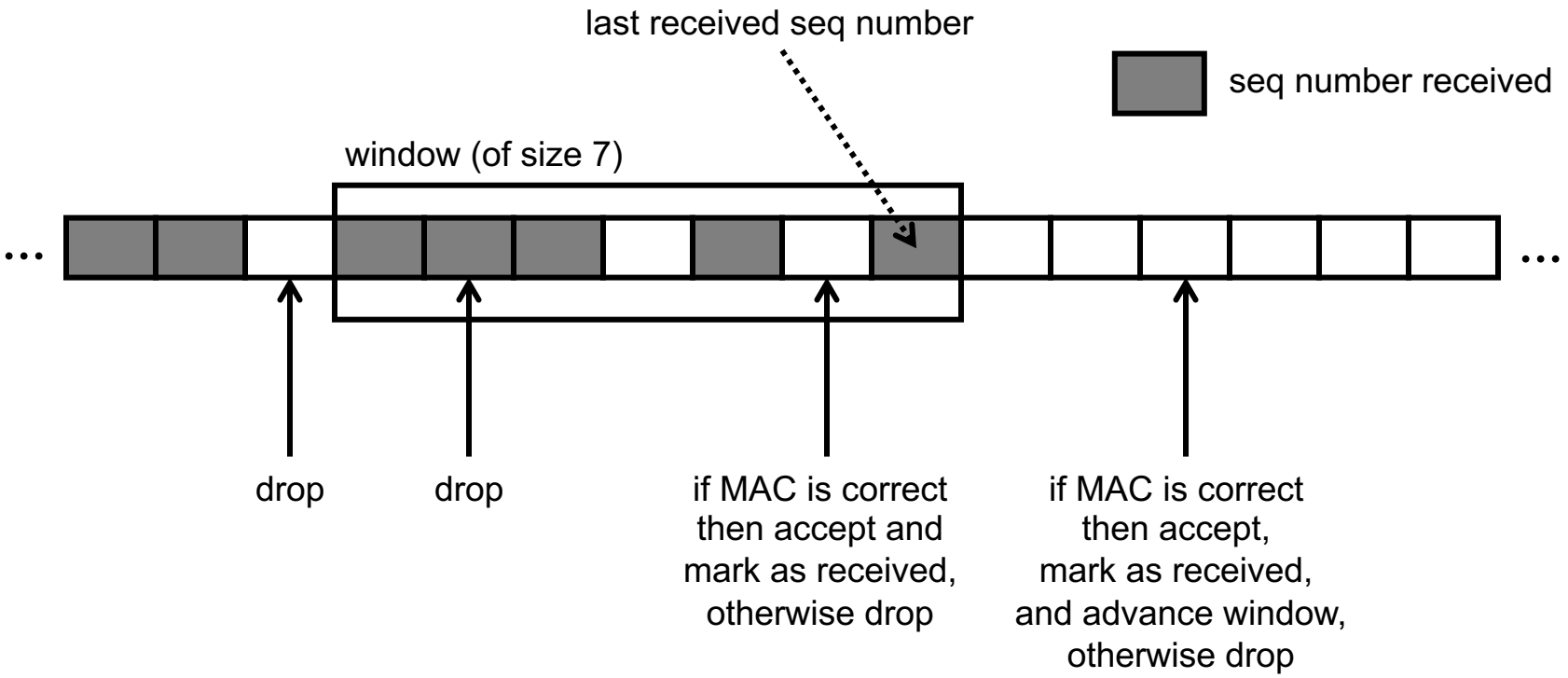
Replay detection in unreliable networks

- in some cases, accepting a message only if its sequence number is larger than the ”receiving” sequence number of the receiver is too strict
  - e.g., in the Internet, the IP protocol does not guarantee that IP packets will arrive in the same order as they were sent
  - in this case, receiving messages out of order is not necessarily an attack
- but we still want to detect replayed messages!

Replay detection in unreliable networks

- to handle this, the receiver may maintain not just a ”receiving” sequence number, but a ”receiving window”
  - the reciever would keep track of received sequence numbers within a window of a given size
  - if a received message has a sequence number smaller than the left edge of the window, then it is dropped
  - if a received message has a sequence number larger than the right edge of the window (and its MAC is correct), then it is accepted and the window is advanced such that its right edge matches the largest sequence number received so far
  - if a received message falls within the window, it is accepted only if its sequence number has not been seen before (and its MAC is correct)

Replay detection window illustrated



Control questions

- How can message sequence numbers be used for replay detection?
- What is the difference between an explicit and an implicit message sequence numbering scheme?
- How can replay protection be solved in an unreliable network?