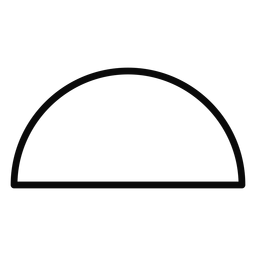
**Il problema delle due armate.**

**Scenario:**

Immagina una collina con il nemico **B** arroccato in cima. Ai lati opposti ci sono due armate alleate: **A1** e **A2**. Per sconfiggere B, devono **attaccare simultaneamente**. Ma possono comunicare solo tramite **messaggeri** che attraversano la valle controllata da B — quindi soggetti a intercettazione.

B ← nemico

A1 ← alleato di A2  A2 ← alleato di A1

**Parallelismo con TCP: flag e numerazione.**

| **Passaggio** | **Mittente → Destinatario** | **Flag** | **SEQ (byte iniziale inviato)** | **ACK (prossimo byte atteso)** |
| --- | --- | --- | --- | --- |
| 1 | A1 → A2 | SYN=1 → richiesta di apertura connessione | SEQ=100 → primo byte trasmesso da A1 | — |
| 2 | A2 → A1 | SYN=1, ACK=1 → risposta e conferma | SEQ=300 → primo byte trasmesso da A2 | ACK=101 → A2 attende il byte successivo da A1 |
| 3 | A1 → A2 | ACK=1 → conferma del SYN ricevuto da A2 | SEQ=101 → byte successivo di A1 | ACK=301 → A1 attende il byte successivo da A2 |
| 4 | A1 → A2 | FIN=1, ACK=1 → A1 chiude la comunicazione | SEQ=102 → byte finale trasmesso da A1 | ACK=301 → conferma ultimo byte ricevuto da A2 |
| 5 | A2 → A1 | FIN=1, ACK=1 → A2 chiude la comunicazione | SEQ=301 → byte finale trasmesso da A2 | ACK=103 → conferma ultimo byte ricevuto da A1 |
| 6 | A1 → A2 | ACK=1 → conferma del FIN ricevuto da A2 | SEQ=103 → ultimo byte trasmesso da A1 | ACK=302 → conferma ultimo byte ricevuto da A2 |

**Cosa dimostra questo?**

* Come nel problema delle armate, **TCP non può garantire certezza assoluta**.
* Se un messaggio o un ACK viene perso, il mittente **non sa se procedere**.
* TCP usa **timeout, ritrasmissione e numerazione** per mitigare l’incertezza.
* Ma il problema delle armate mostra che **il consenso perfetto è impossibile** in ambienti non affidabili.

**Collegamento con sliding windows:**

Il protocollo **sliding windows** estende questa logica:

* Tiene traccia di più pacchetti “in volo”.
* Ogni pacchetto ha un SEQ.
* Gli ACK cumulativi o selettivi indicano cosa è stato ricevuto.
* Se un ACK non arriva entro un certo tempo, il pacchetto viene **ritrasmesso**.

**Collegamento con il three-way handshake:**

Il three-way handshake applica il problema delle due armate alla fase di apertura della connessione:

* Il mittente invia un SYN per iniziare la comunicazione.
* Il destinatario risponde con SYN+ACK per confermare e proporre la propria sequenza.
* Il mittente invia un ACK finale per confermare la conferma.
* Se l’ultimo ACK si perde, il destinatario non sa se procedere: manca la certezza assoluta.