Byzantine General Problem

Murat Osmanoglu

Permissionless Blockchain

Permissionless Blockchain

Bitcoin provides permissionless setting:

Permissionless Blockchain

Bitcoin provides permissionless setting:

- Anyone can participate in the protocol and receive BTC as rewards by performing the PoW-based mining

Permissionless Blockchain

Bitcoin provides permissionless setting:

- Anyone can participate in the protocol and receive BTC as rewards by performing the PoW-based mining

- The mechanism of pouring currency in the system via PoW, that makes it feasible for anyone(possessing sufficient hashing power) to participate

Permissionless Blockchain

Bitcoin provides permissionless setting:

- Anyone can participate in the protocol and receive BTC as rewards by performing the PoW-based mining

- The mechanism of pouring currency in the system via PoW, that makes it feasible for anyone(possessing sufficient hashing power) to participate

- The ledger itself is public, readable, and writeable by anyone

Permissioned Blockchain

Permissioned Blockchain

- Participation is restricted

Permissioned Blockchain

- Participation is restricted
- Producing transactions and/or blocks can only be performed after being authorized by the other nodes

Permissioned Blockchain

- Participation is restricted

- Producing transactions and/or blocks can only be performed after being authorized by the other nodes

- The set of nodes is static : it's fixed and determined at the onset of the protocol execution

Permissioned Blockchain

- Participation is restricted
- Producing transactions and/or blocks can only be performed after being authorized by the other nodes
- The set of nodes is static : it's fixed and determined at the onset of the protocol execution

can also be dynamic, i.e. the initial set of nodes agree on a specific set of rules to accept new players

Permissioned Blockchain

- Prior the system operation the nodes register their certificates, generated by a certificate authority, that are included in the genesis block

Permissioned Blockchain

- Prior the system operation the nodes register their certificates, generated by a certificate authority, that are included in the genesis block

- Using such certificates, all the nodes are capable of authenticating each participant and allowing interaction with the LOG

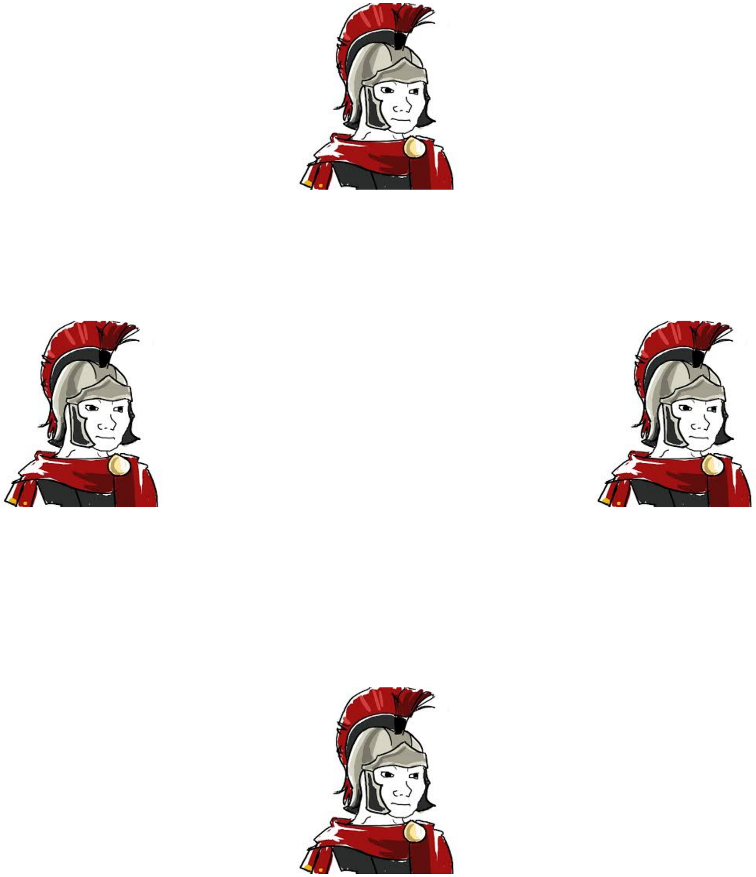
Permissioned Blockchain

- Prior the system operation the nodes register their certificates, generated by a certificate authority, that are included in the genesis block

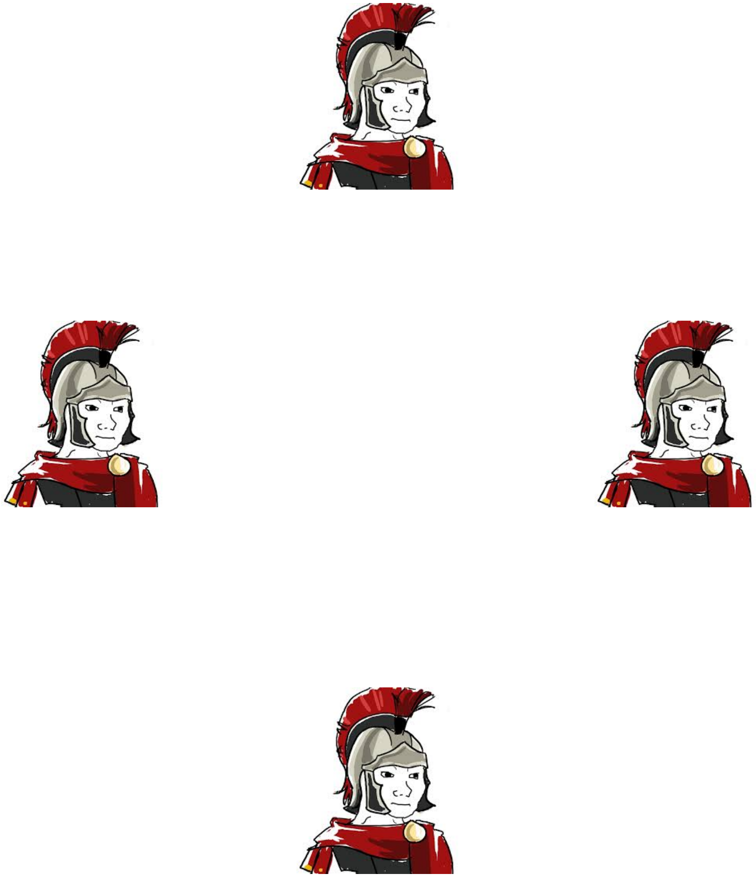
- Using such certificates, all the nodes are capable of authenticating each participant and allowing interaction with the LOG

- Certificates need to be revoked in case that the corresponding secret keys become exposed

The Byzantine Generals Problem

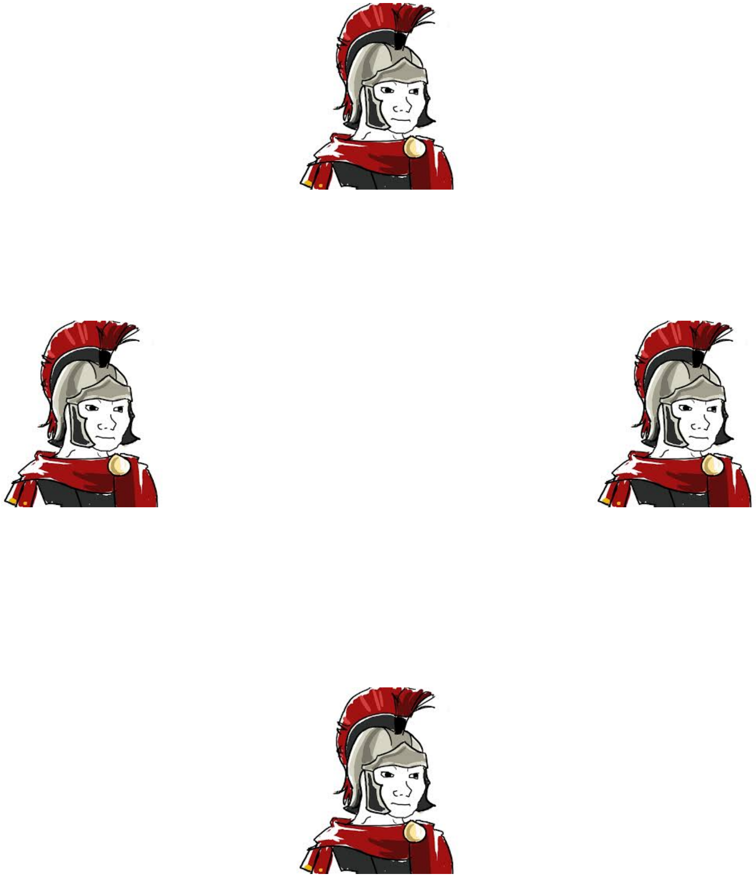


The Byzantine Generals Problem



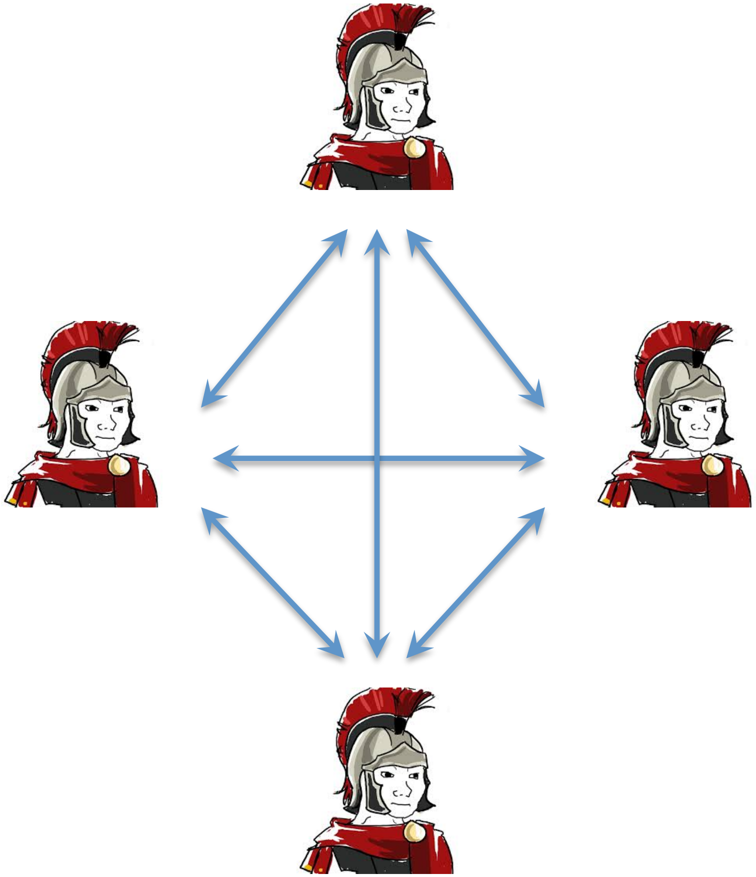
- several divisions of Byzantine army camped outside of an enemy city

The Byzantine Generals Problem



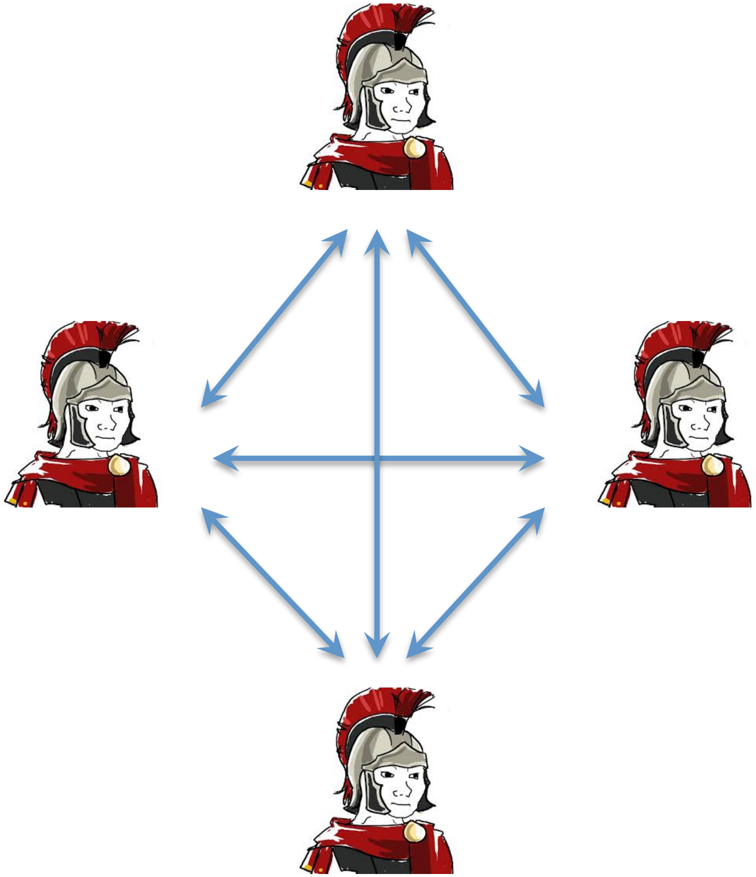
- several divisions of Byzantine army camped outside of an enemy city
- each of them commanded by a general

The Byzantine Generals Problem



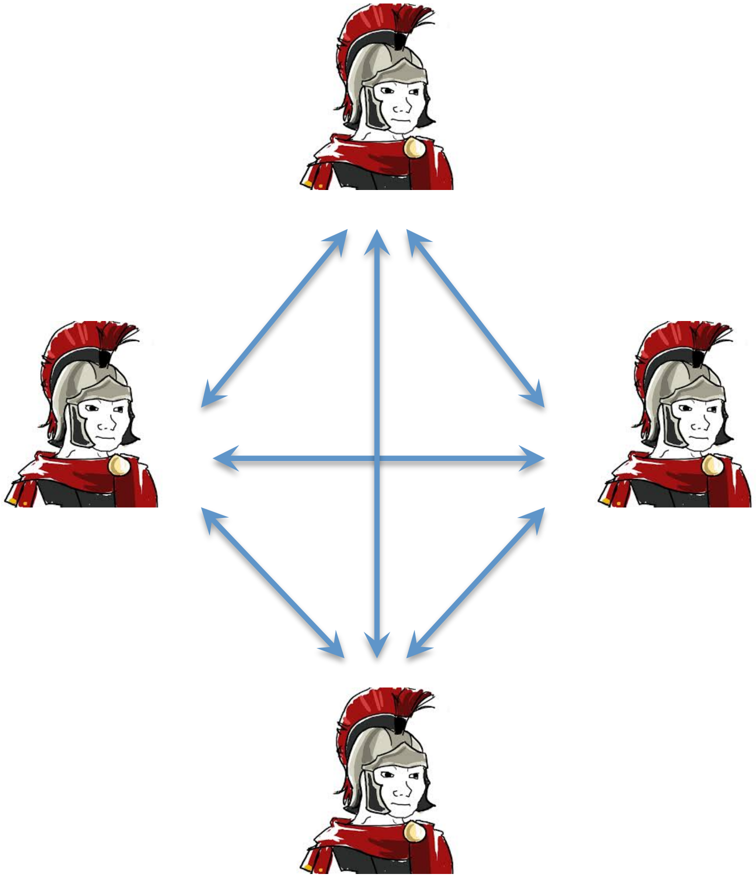
- several divisions of Byzantine army camped outside of an enemy city
- each of them commanded by a general
- generals communicate through messengers

The Byzantine Generals Problem



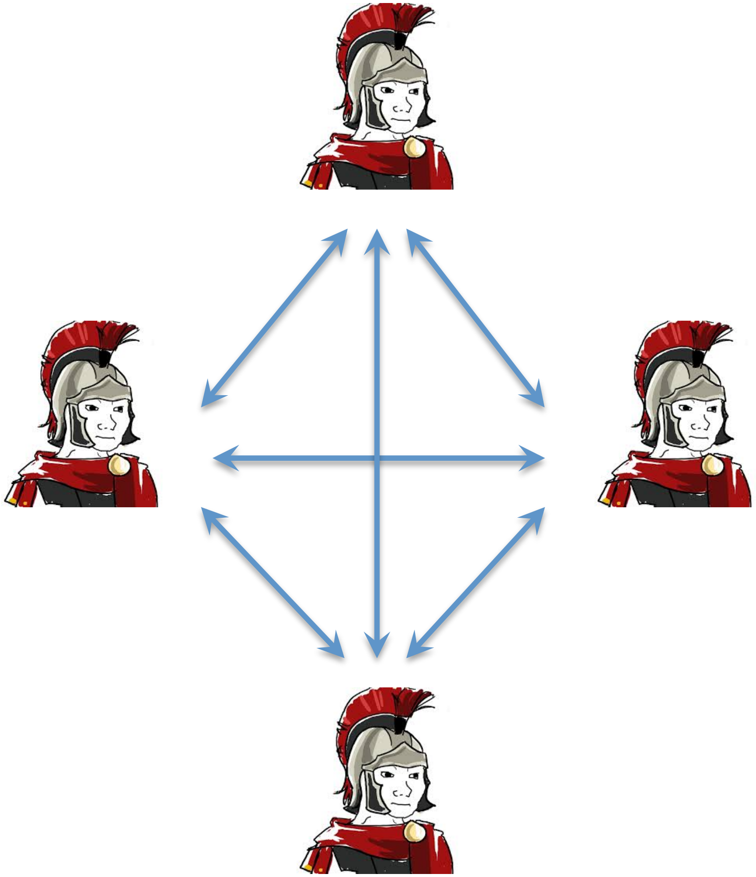
- several divisions of Byzantine army camped outside of an enemy city
- each of them commanded by a general
- generals communicate through messengers
- they try to reach an agreement on a common plan (retreat or attack) while some of them may be traitors that sabotage this process

The Byzantine Generals Problem



The generals must develop an algorithm guaranteeing that

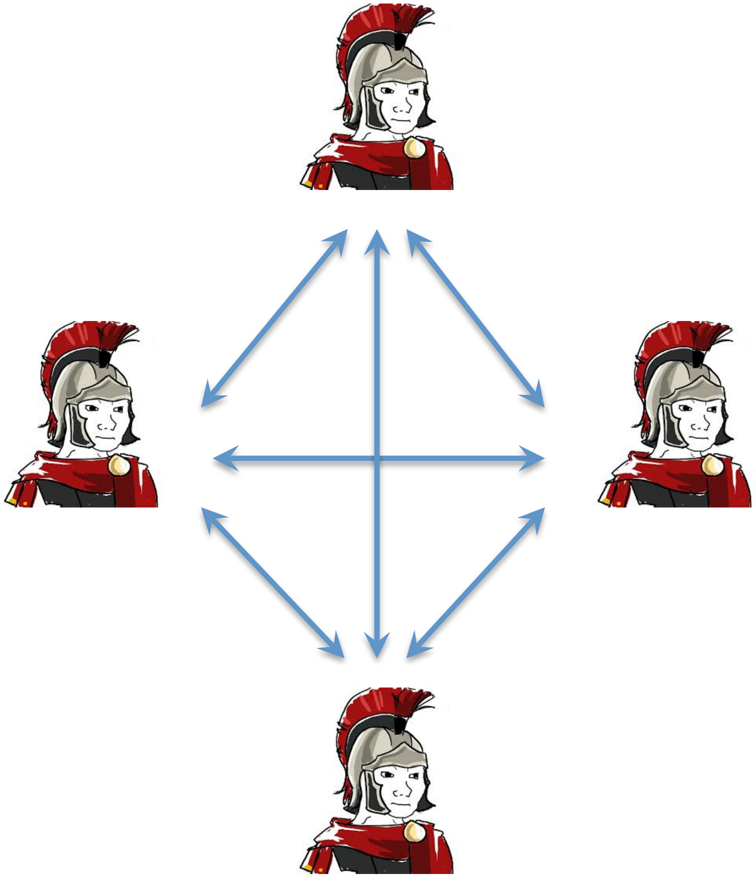
The Byzantine Generals Problem



The generals must develop an algorithm guaranteeing that

- all loyal generals decide on the same plan

The Byzantine Generals Problem

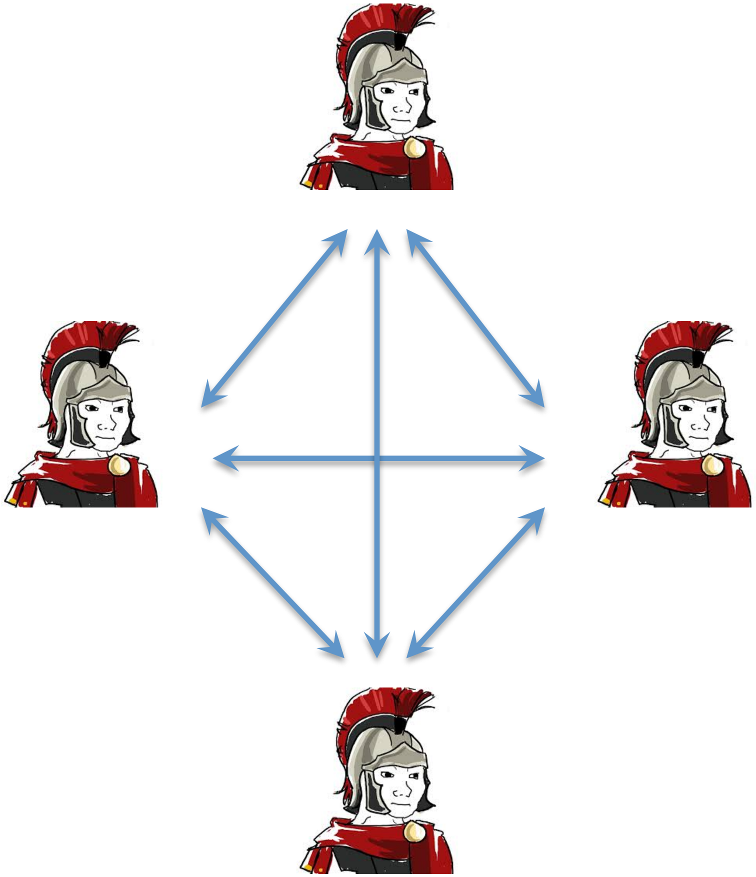


The generals must develop an algorithm guaranteeing that

- all loyal generals decide on the same plan

The algorithm guarantees this regardless of what the traitors do

The Byzantine Generals Problem



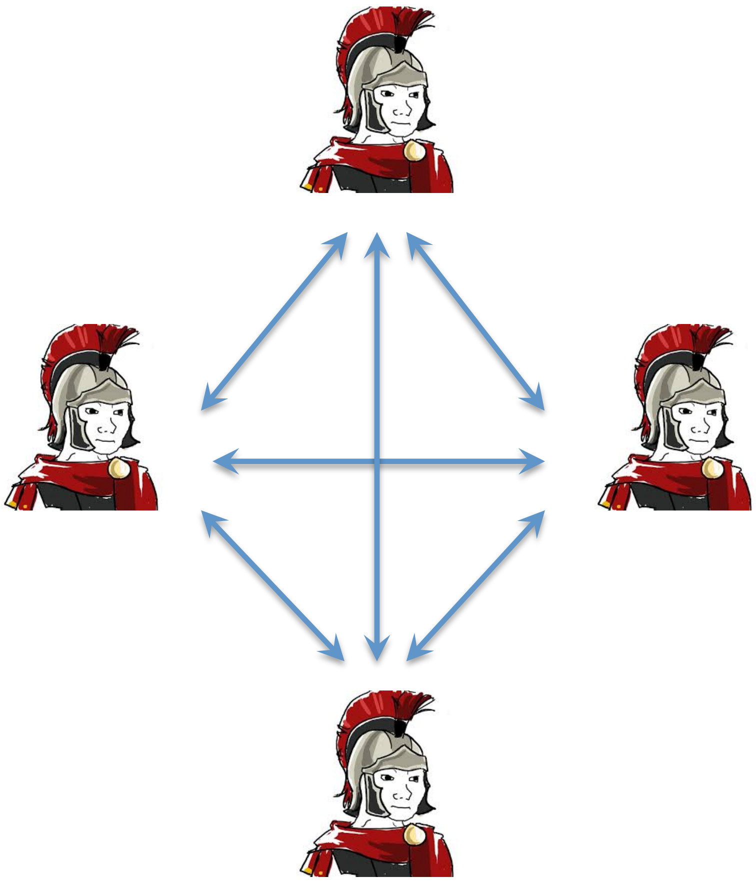
The generals must develop an algorithm guaranteeing that

- all loyal generals decide on the same plan

  The algorithm guarantees this regardless of what the traitors do

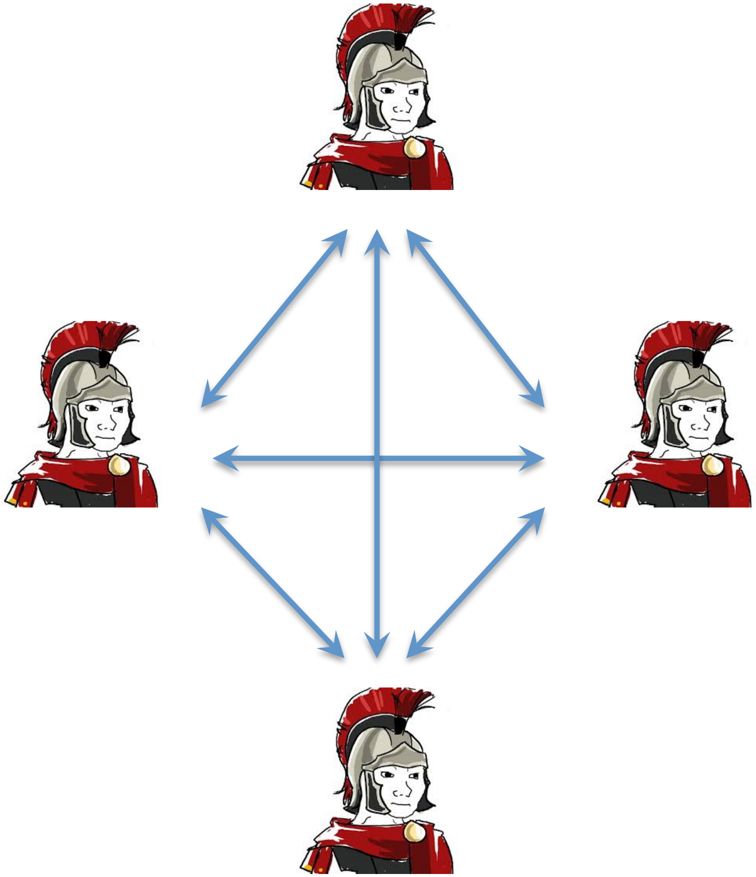
- a small number cannot cause the loyal generals to adopt the wrong plan

The Byzantine Generals Problem



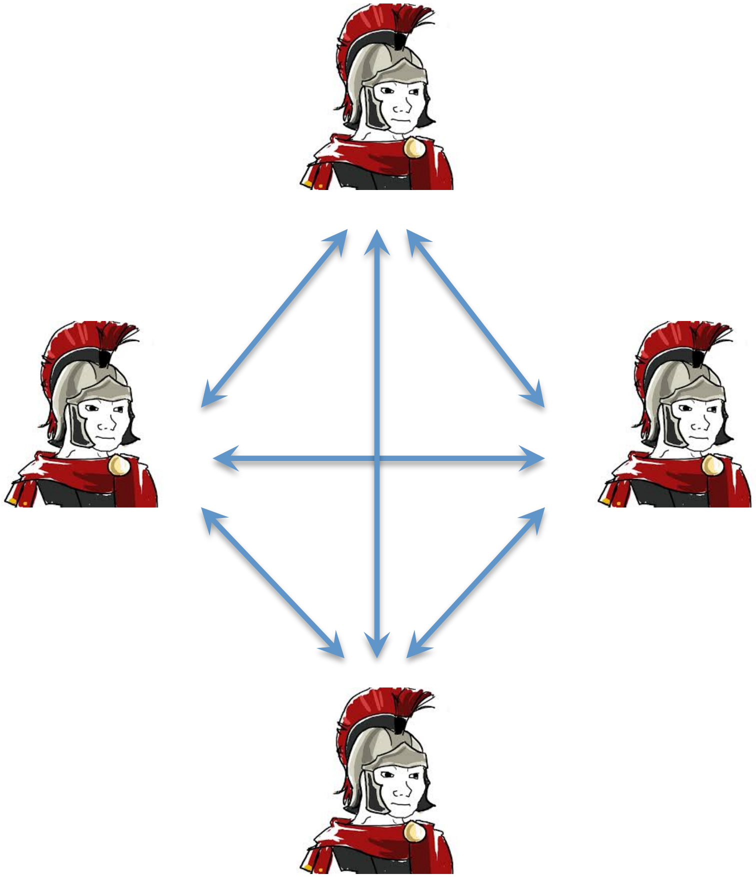
- Each general sends his decision v(i) to each other general by messenger

The Byzantine Generals Problem



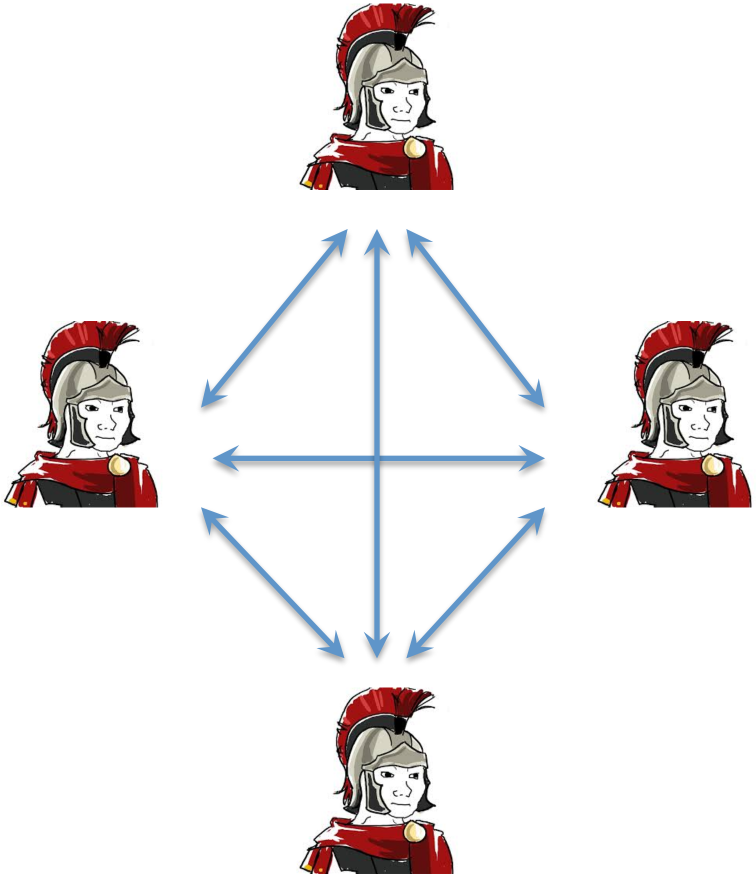
- Each general sends his decision v(i) to each other general by messenger
  - traitors may send different messages to different generals

The Byzantine Generals Problem



- Each general sends his decision v(i) to each other general by messenger
  - traitors may send different messages to different generals
- Two conditions must be satisfied :

The Byzantine Generals Problem

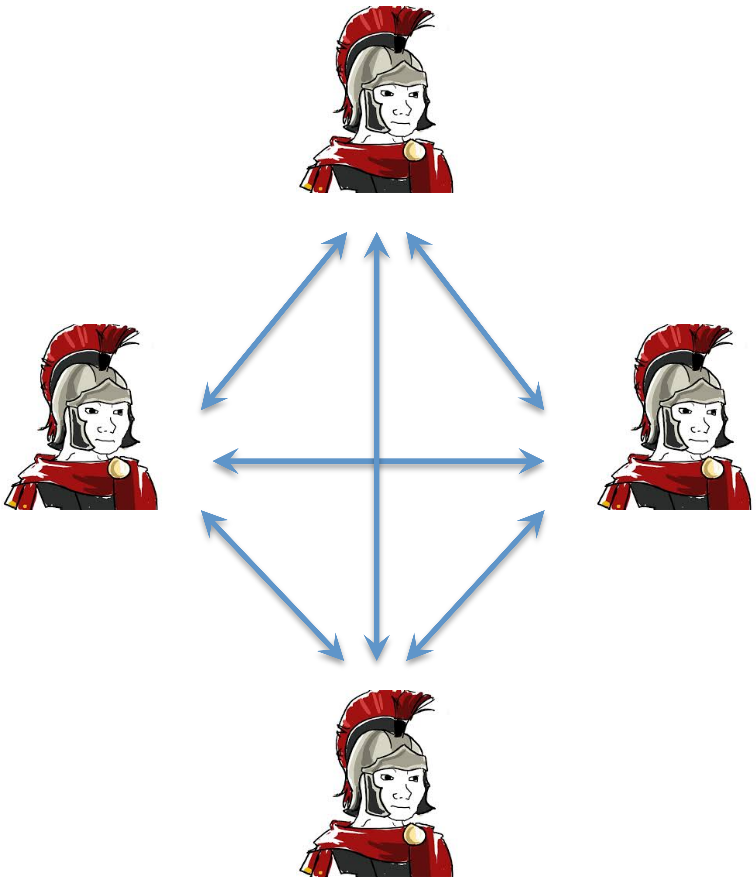


- Each general sends his decision v(i) to each other general by messenger

  traitors may send different messages to different generals

- Two conditions must be satisfied :
  - every loyal general must obtain the same set of messages v(1),…,v(n)

The Byzantine Generals Problem

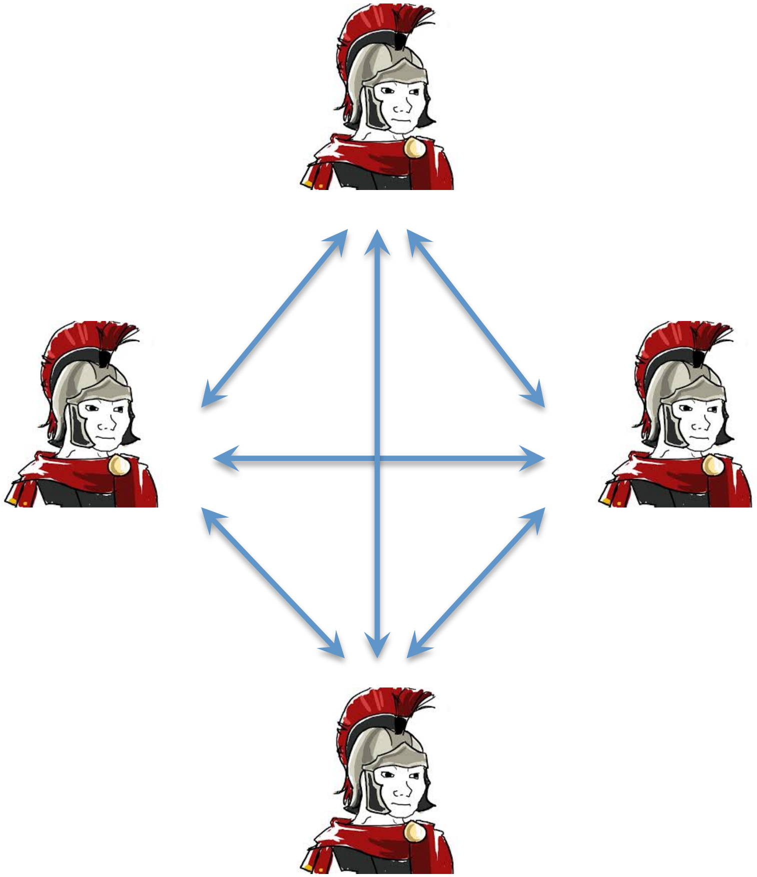


- Each general sends his decision v(i) to each other general by messenger

  traitors may send different messages to different generals

- Two conditions must be satisfied :
  - every loyal general must obtain the same set of messages v(1),….,v(n)
  - if the i-th general is loyal, then the value he sends must be used by every loyal general as v(i)

The Byzantine Generals Problem



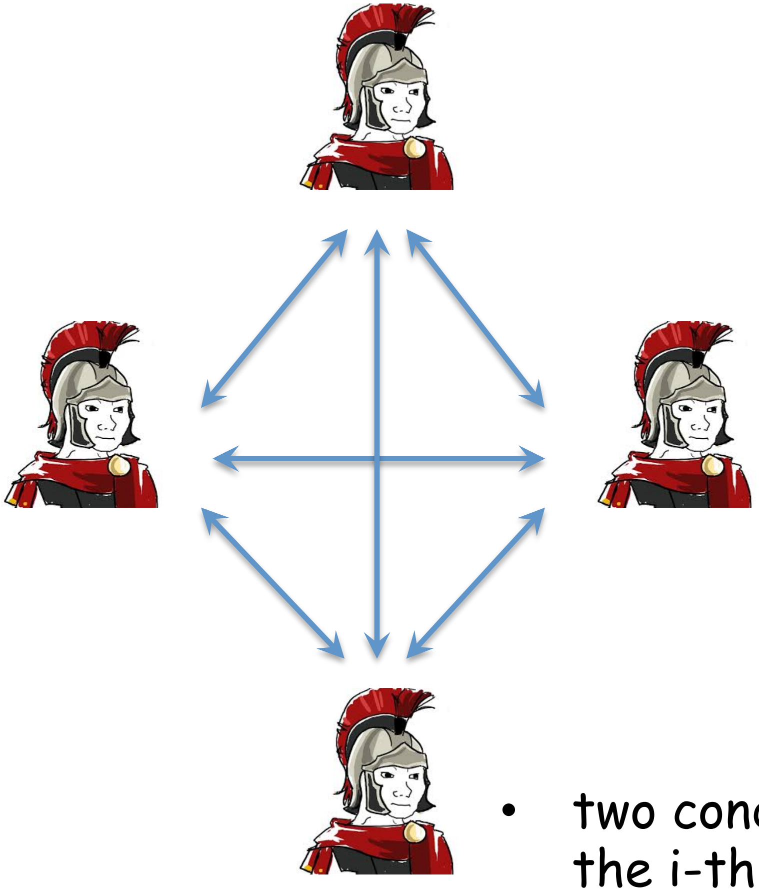
- Each general sends his decision v(i) to each other general by messenger

  traitors may send different messages to different generals

- Two conditions must be satisfied :
  - every loyal general must obtain the same set of messages v(1),…,v(n)

    any two loyal generals use the same v(i)
  - if the i-th general is loyal, then the value he sends must be used by every loyal general as v(i)

The Byzantine Generals Problem



- Each general sends his decision v(i) to each other general by messenger

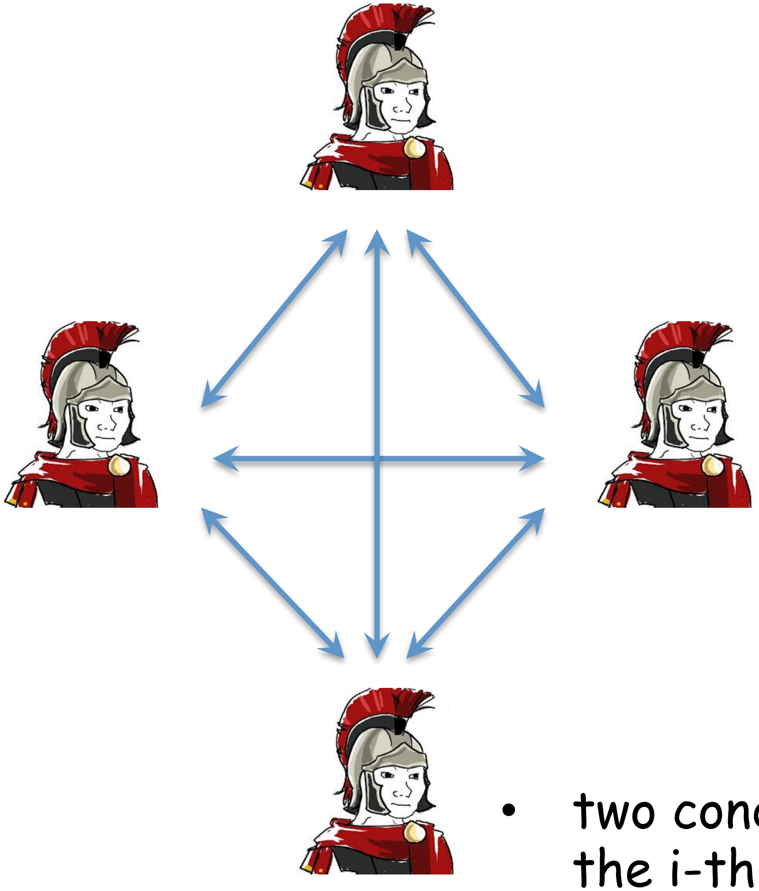
  traitors may send different messages to different generals

- Two conditions must be satisfied :
  - every loyal general must obtain the same set of messages v(1),…,v(n)

    any two loyal generals use the same v(i)
  - if the i-th general is loyal, then the value he sends must be used by every loyal general as v(i)

- two conditions now on the single value v(i) sent by the i-th general.

The Byzantine Generals Problem



- Each general sends his decision v(i) to each other general by messenger

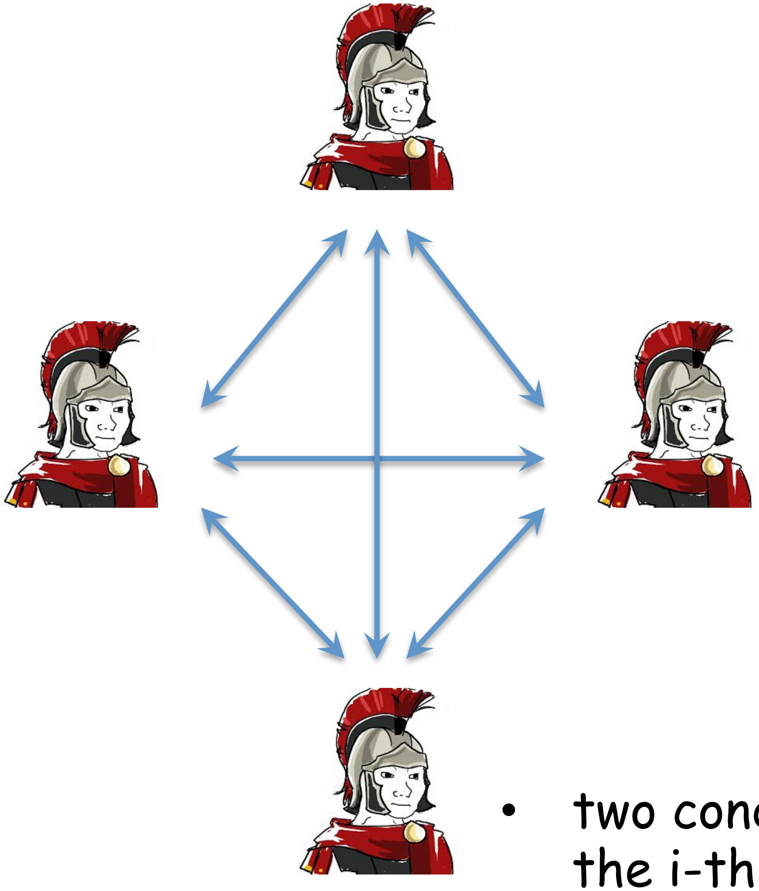
  traitors may send different messages to different generals

- Two conditions must be satisfied :
  - every loyal general must obtain the same set of messages v(1),…,v(n)

    any two loyal generals use the same v(i)
  - if the i-th general is loyal, then the value he sends must be used by every loyal general as v(i)

- two conditions now on the single value v(i) sent by the i-th general.
- turn the problem into a simpler one:

The Byzantine Generals Problem



- Each general sends his decision v(i) to each other general by messenger

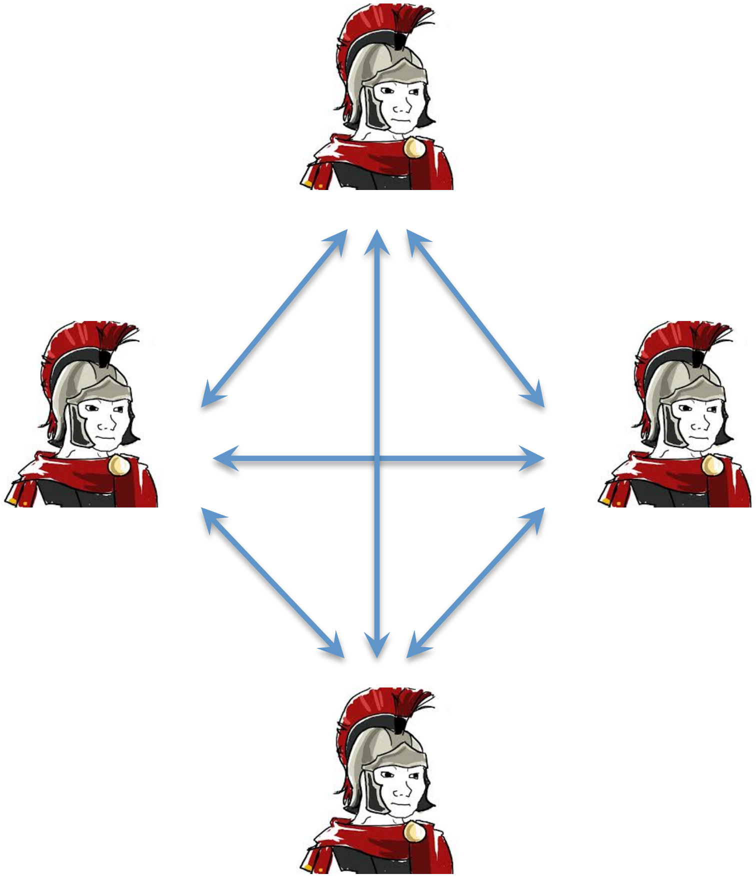
  traitors may send different messages to different generals

- Two conditions must be satisfied :
  - every loyal general must obtain the same set of messages v(1),…,v(n)

    any two loyal generals use the same v(i)
  - if the i-th general is loyal, then the value he sends must be used by every loyal general as v(i)

- two conditions now on the single value v(i) sent by the i-th general.
- turn the problem into a simpler one:
  How a single general sends his value to the others ?

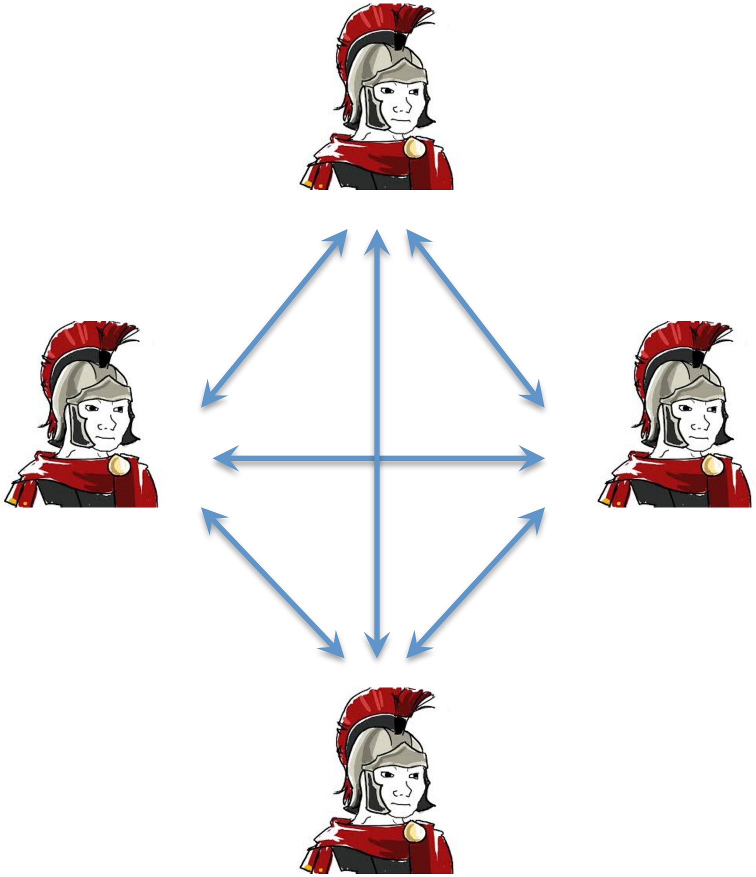
The Byzantine Generals Problem



Definition

A general must send an order to his n – 1 lieutenant generals in a way that :

The Byzantine Generals Problem

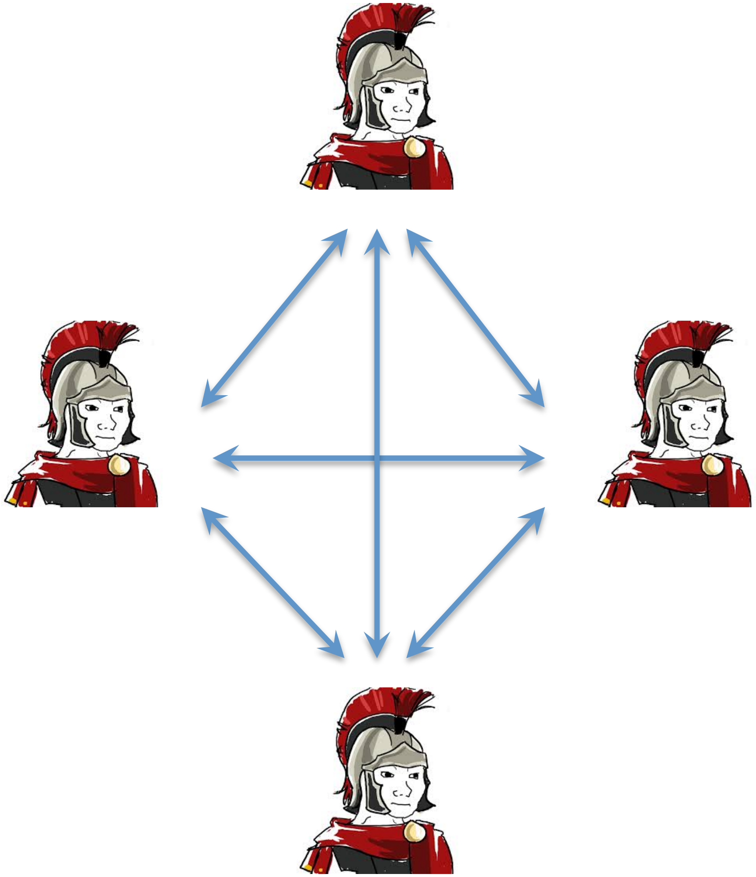


Definition

A general must send an order to his n – 1 lieutenant generals in a way that :

- all loyal lieutenants obey the same order

The Byzantine Generals Problem



Definition

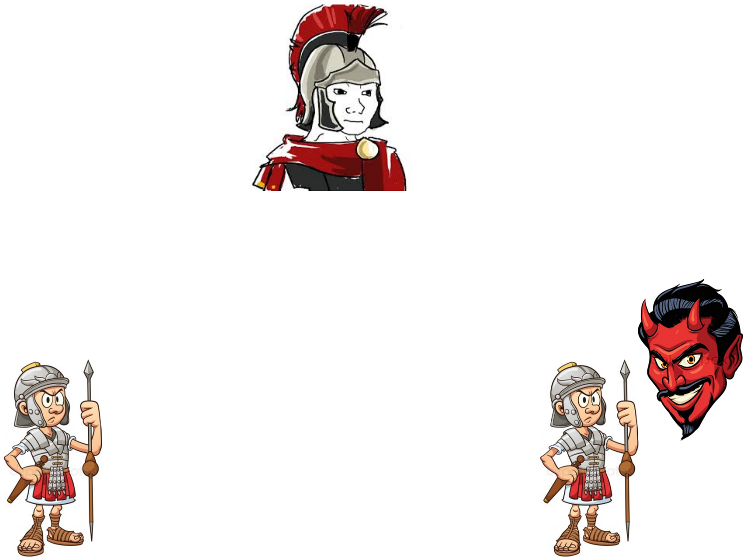
A general must send an order to his n – 1 lieutenant generals in a way that :

- all loyal lieutenants obey the same order
- if the general is loyal, then every loyal lieutenant obeys the order he sends

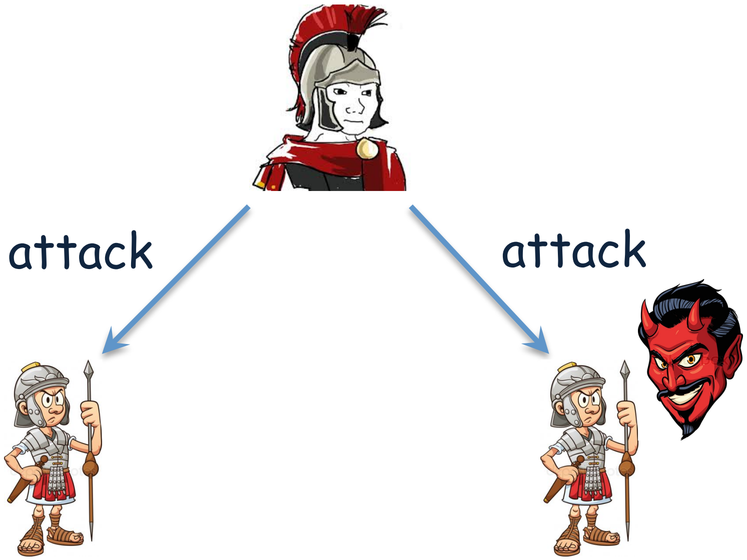
The Byzantine Generals Problem

The Byzantine Generals Problem

The Byzantine Generals Problem



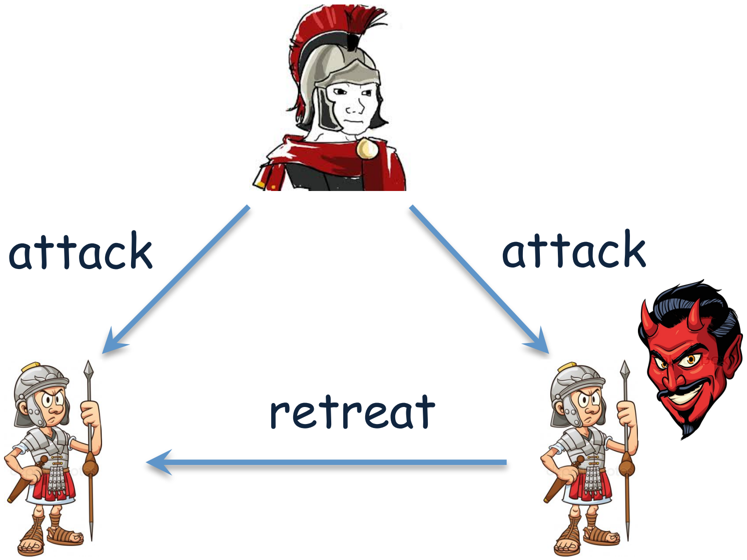
The Byzantine Generals Problem



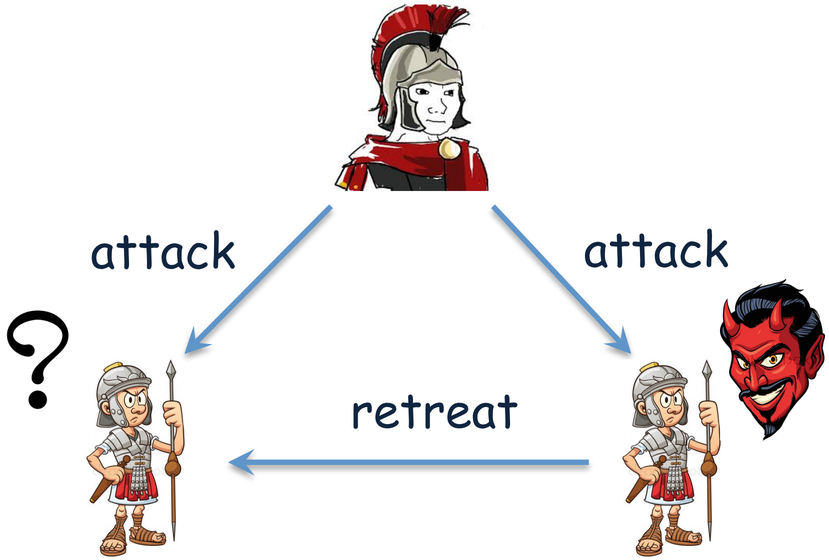
attack

attack

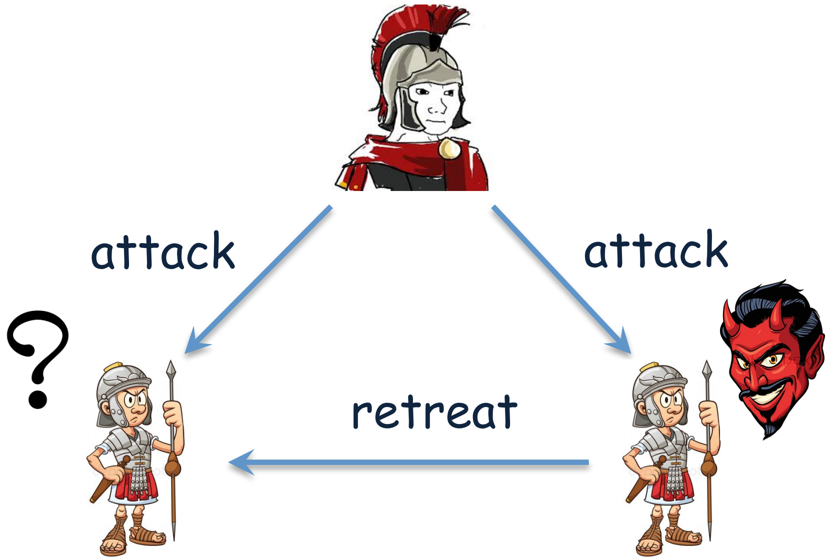
The Byzantine Generals Problem



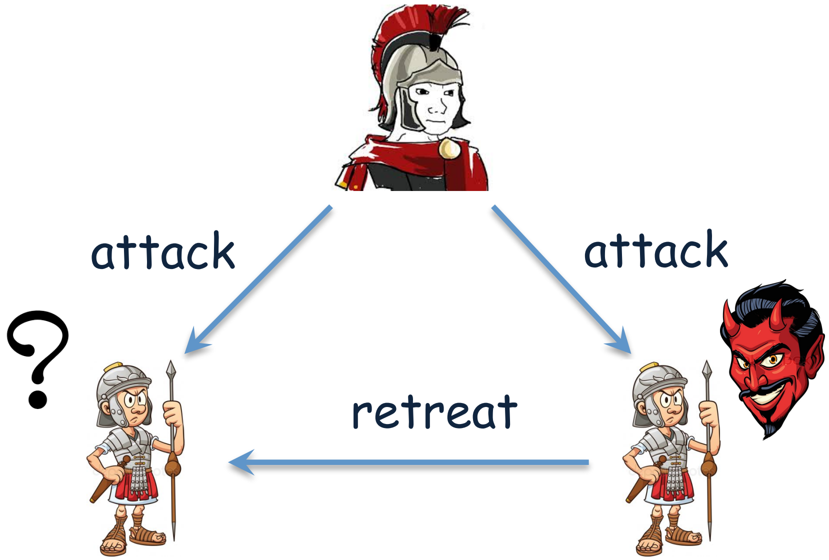
The Byzantine Generals Problem



The Byzantine Generals Problem



The Byzantine Generals Problem

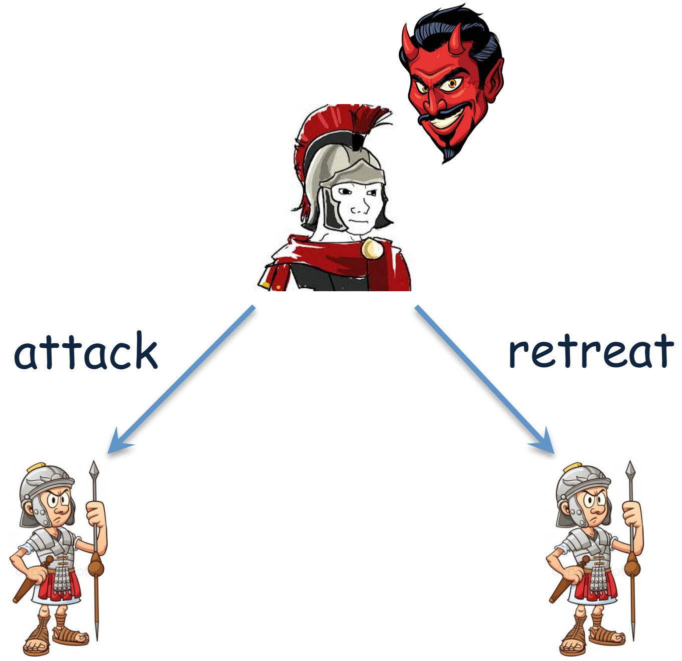


attack

attack

?

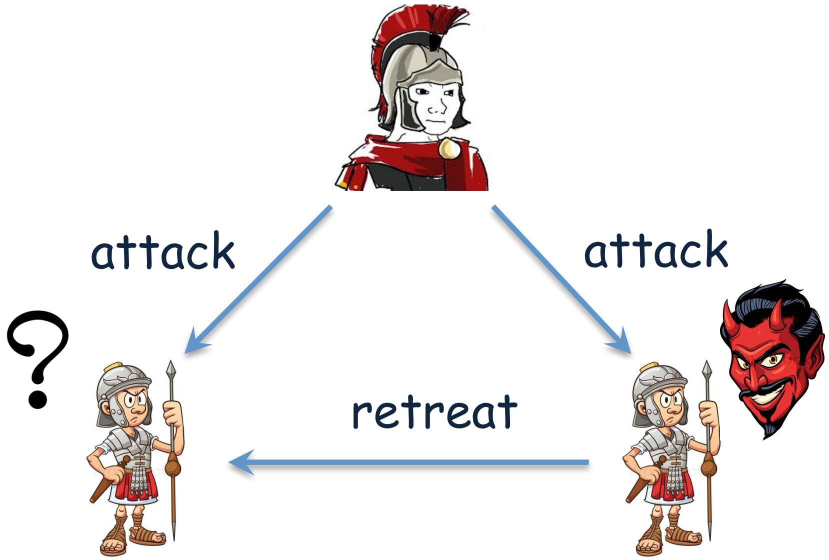
retreat



attack

retreat

The Byzantine Generals Problem

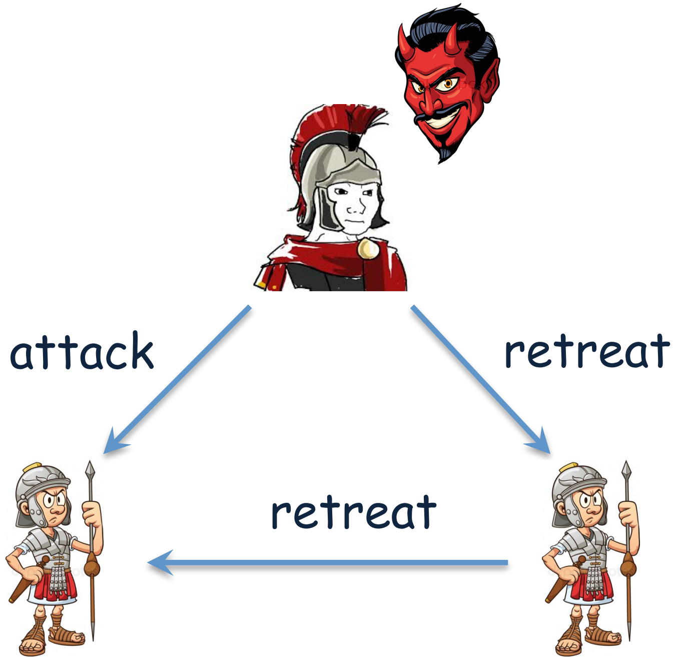


attack

attack

?

retreat

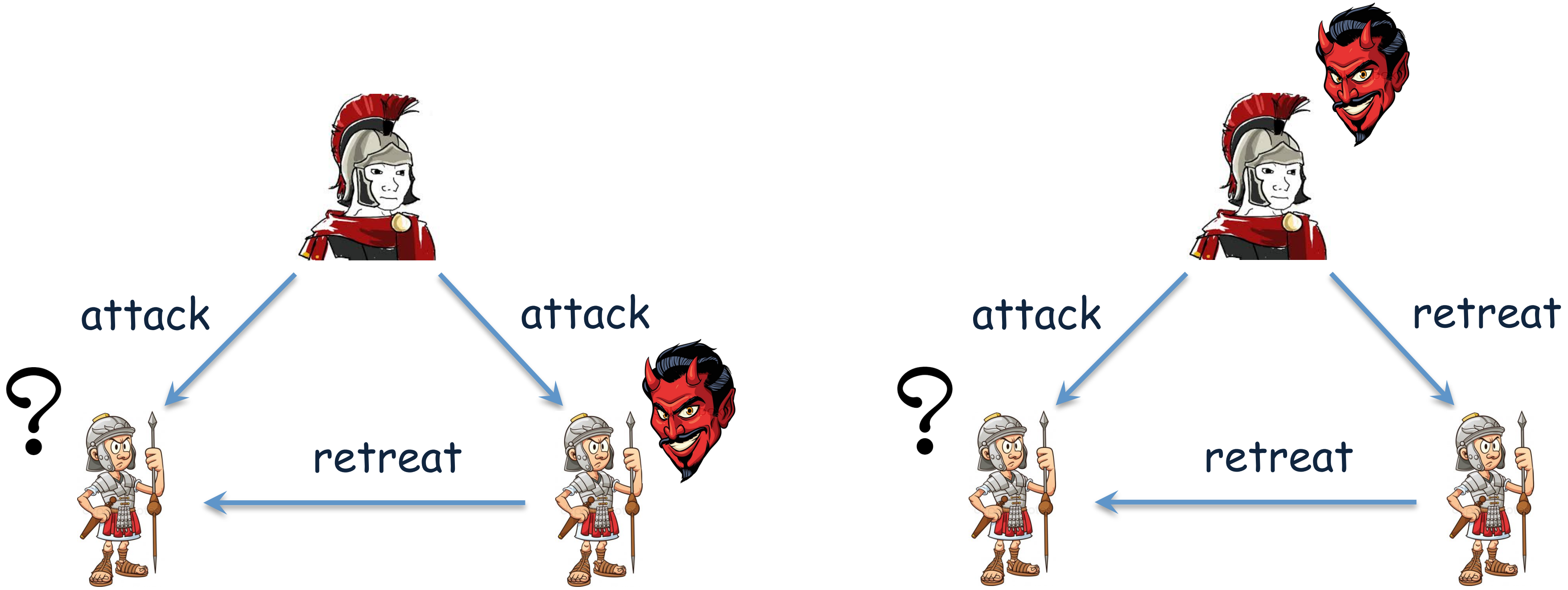


attack

retreat

retreat

The Byzantine Generals Problem



attack

attack

?

retreat

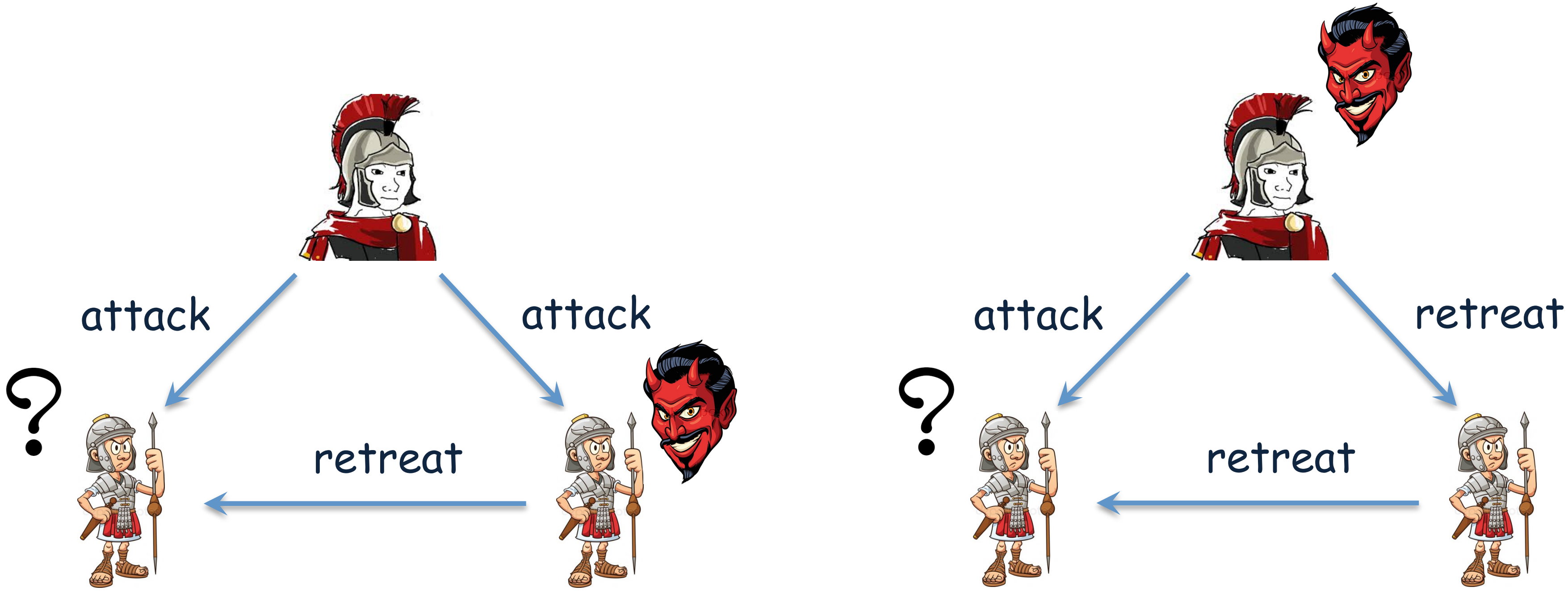
attack

retreat

?

retreat

The Byzantine Generals Problem



- no solution can work in the presence of a single traitor if there are only three generals

The Byzantine Generals Problem

Assumptions

The Byzantine Generals Problem

Assumptions

- every message is delivered correctly

The Byzantine Generals Problem

Assumptions

- every message is delivered correctly
- the receiver of a message knows who sent it

  prevents a traitor from interfering with the communication between other two generals.

The Byzantine Generals Problem

Assumptions

- every message is delivered correctly
- the receiver of a message knows who sent it

  prevents a traitor from interfering with the communication between other two generals.
- the absence of a message can be detected

  prevents a traitor to sabotage a decision by not sending messages

The Byzantine Generals Problem

Assumptions

- every message is delivered correctly
- the receiver of a message knows who sent it

  prevents a traitor from interfering with the communication between other two generals.
- the absence of a message can be detected

  prevents a traitor to sabotage a decision by not sending messages

Algorithm

The Byzantine Generals Problem

Assumptions

- every message is delivered correctly
- the receiver of a message knows who sent it

  prevents a traitor from interfering with the communication between other two generals.
- the absence of a message can be detected

  prevents a traitor to sabotage a decision by not sending messages

Algorithm

- The commander sends his decision to every lieutenant

The Byzantine Generals Problem

Assumptions

- every message is delivered correctly
- the receiver of a message knows who sent it

  prevents a traitor from interfering with the communication between other two generals.
- the absence of a message can be detected

  prevents a traitor to sabotage a decision by not sending messages

Algorithm

- The commander sends his decision to every lieutenant
- For each i, let v(i) be the value lieutenant i receives from the commander, else be RETREAT if he receives no value

The Byzantine Generals Problem

Assumptions

- every message is delivered correctly
- the receiver of a message knows who sent it

  prevents a traitor from interfering with the communication between other two generals.
- the absence of a message can be detected

  prevents a traitor to sabotage a decision by not sending messages

Algorithm

- The commander sends his decision to every lieutenant
- For each i, let v(i) be the value lieutenant i receives from the commander, else be RETREAT if he receives no value
- Lieutenant i sends the value v(i) to all other lieutenants

The Byzantine Generals Problem

Assumptions

- every message is delivered correctly
- the receiver of a message knows who sent it

  prevents a traitor from interfering with the communication between other two generals.
- the absence of a message can be detected

  prevents a traitor to sabotage a decision by not sending messages

Algorithm

- The commander sends his decision to every lieutenant
- For each i, let v(i) be the value lieutenant i receives from the commander, else be RETREAT if he receives no value
- Lieutenant i sends the value v(i) to all other lieutenants
- For each i, and each j≠i, let v(j) be the value lieutenant i received from lieutenant j, else be RETREAT if he receives no value

The Byzantine Generals Problem

Assumptions

- every message is delivered correctly
- the receiver of a message knows who sent it

  prevents a traitor from interfering with the communication between other two generals.
- the absence of a message can be detected

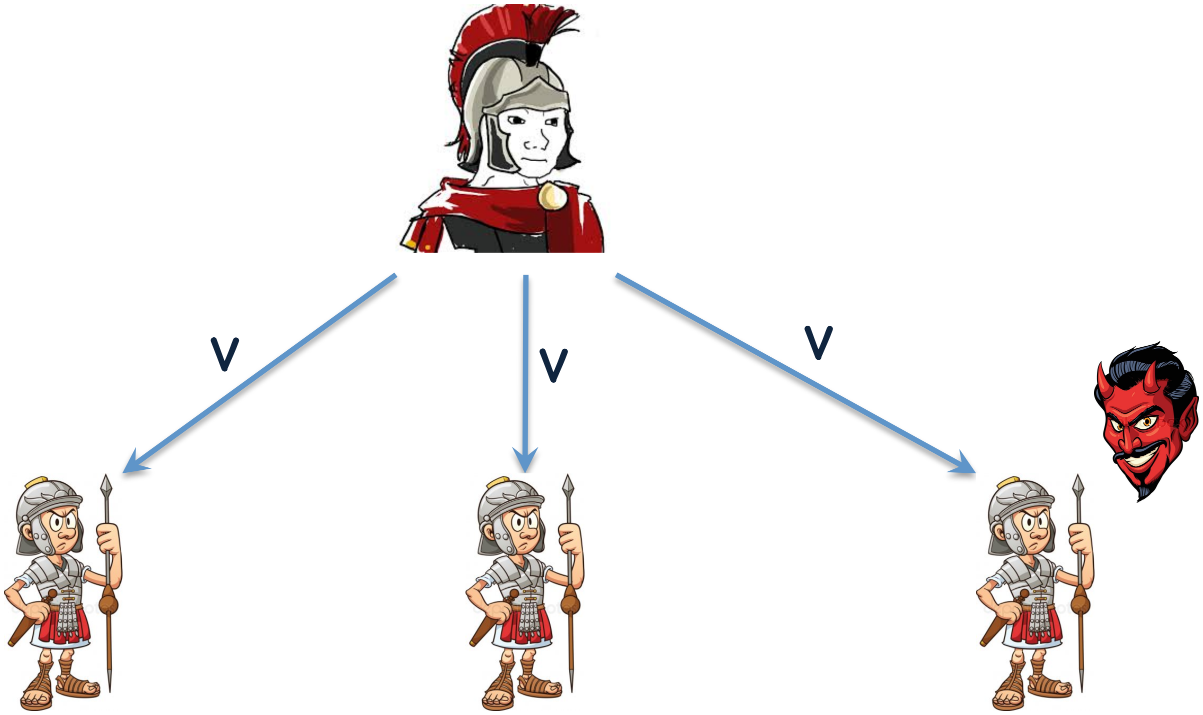
  prevents a traitor to sabotage a decision by not sending messages

Algorithm

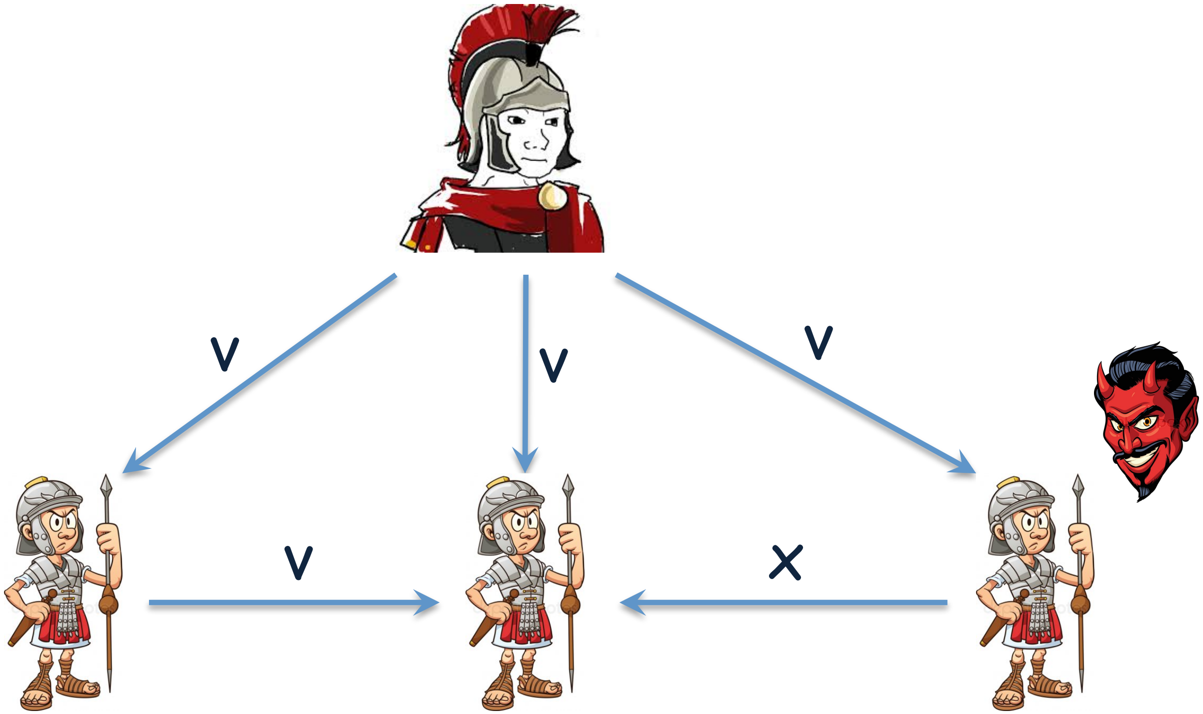
- The commander sends his decision to every lieutenant
- For each i, let v(i) be the value lieutenant i receives from the commander, else be RETREAT if he receives no value
- Lieutenant i sends the value v(i) to all other lieutenants
- For each i, and each j≠i, let v(j) be the value lieutenant i received from lieutenant j, else be RETREAT if he receives no value
- Lieutenant i computes the final decision as majority(v(1),…,v(n-1))

The Byzantine Generals Problem

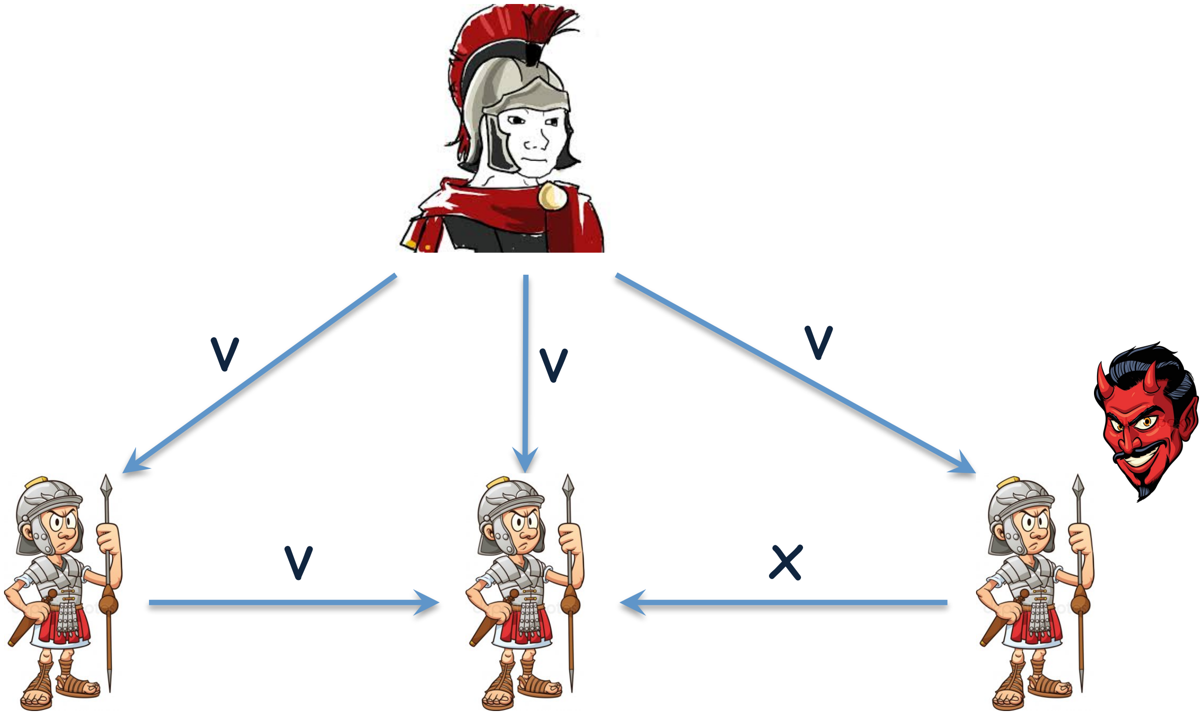
The Byzantine Generals Problem



The Byzantine Generals Problem



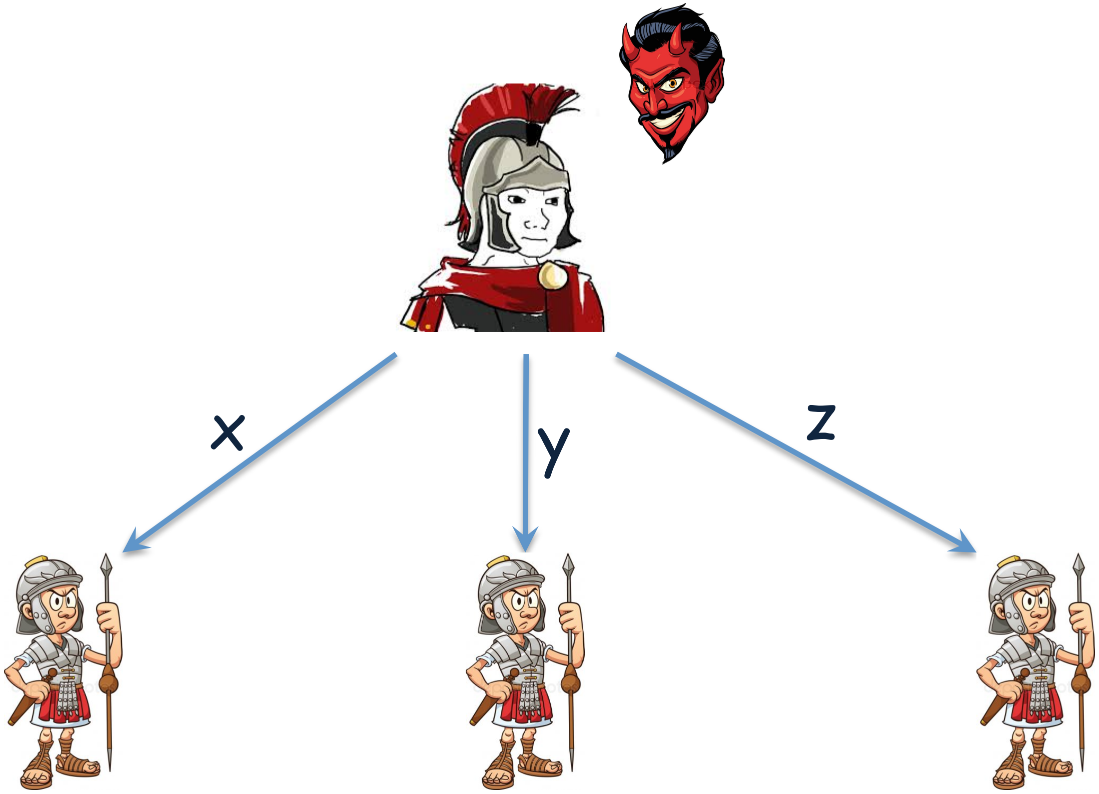
The Byzantine Generals Problem



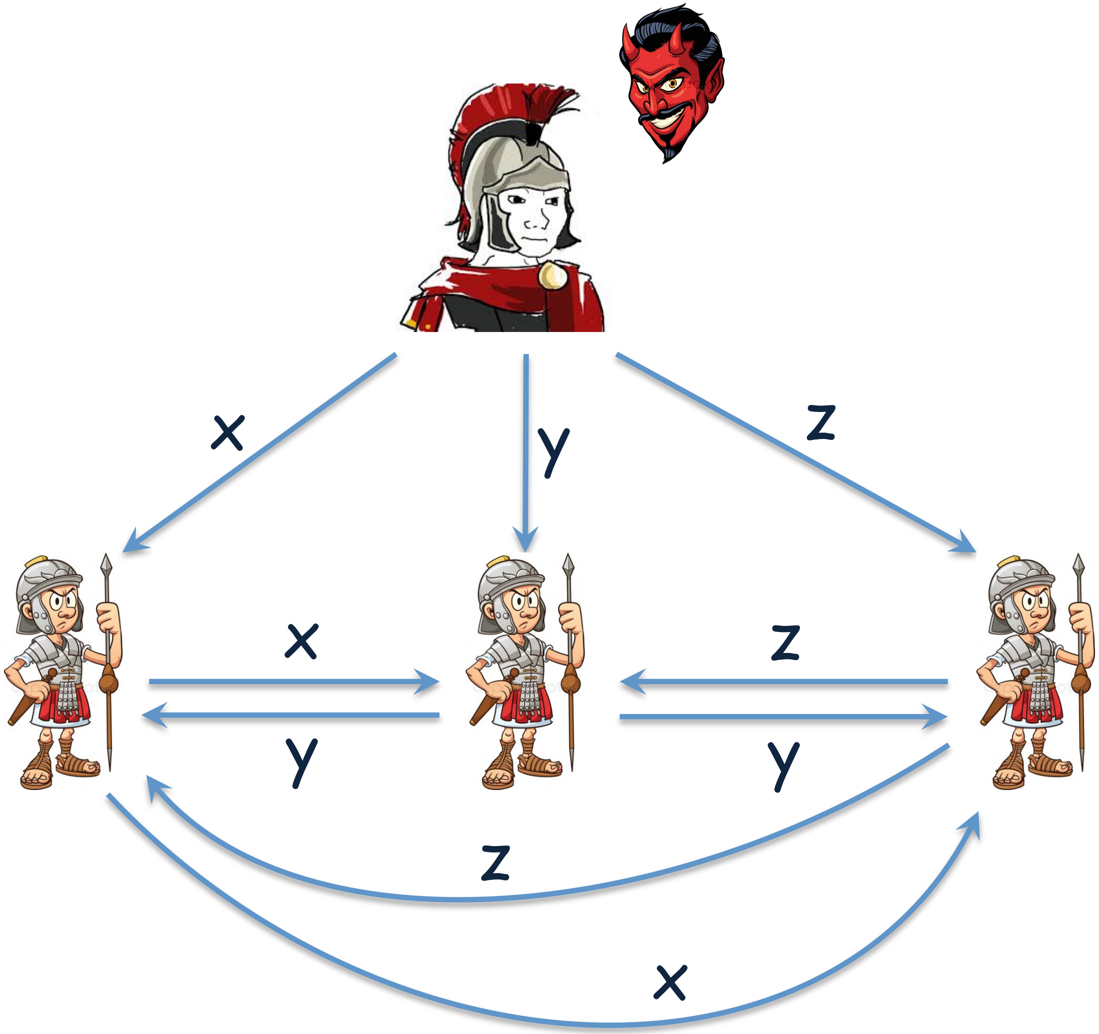
majority(v, v, x)

The Byzantine Generals Problem

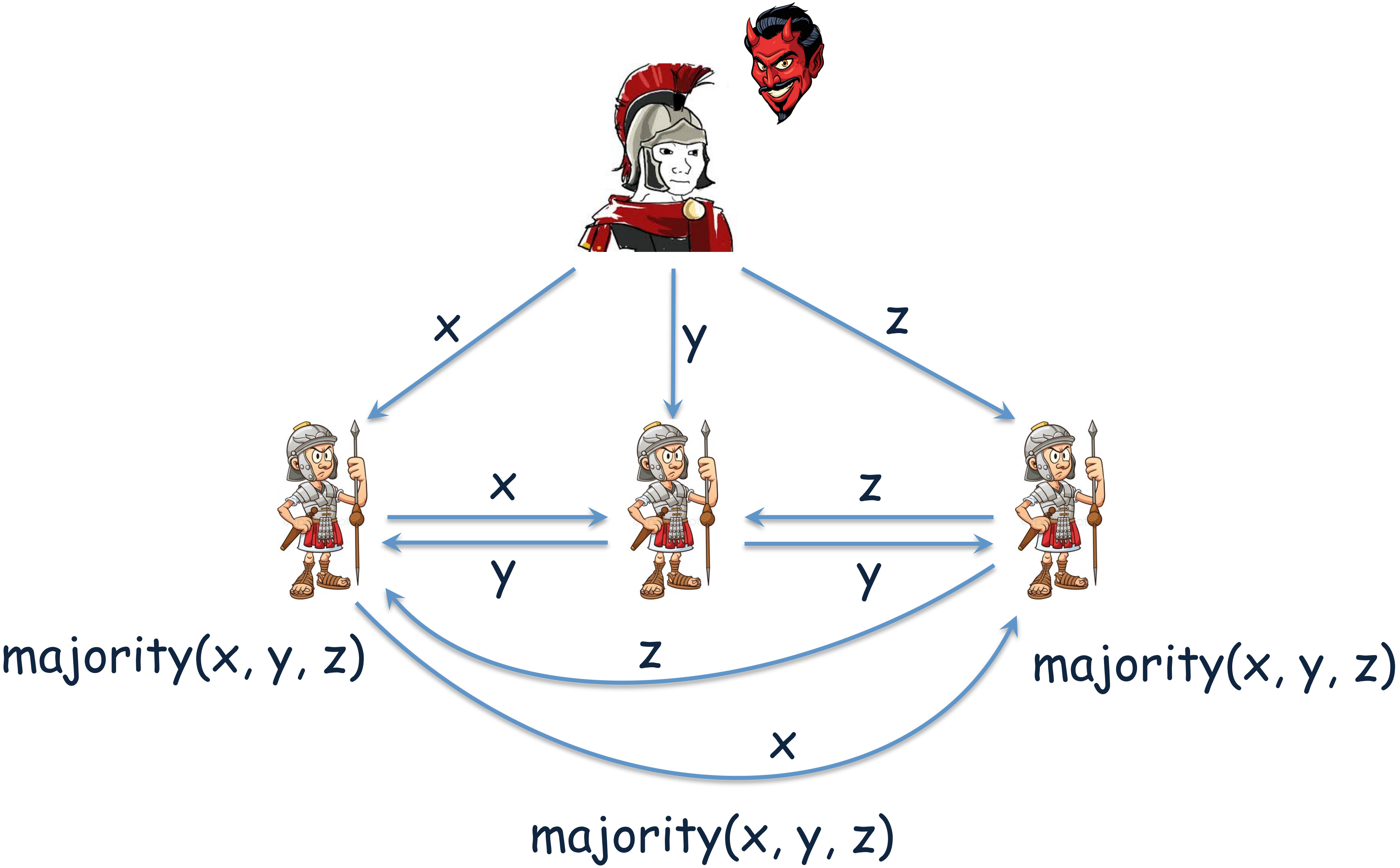
The Byzantine Generals Problem



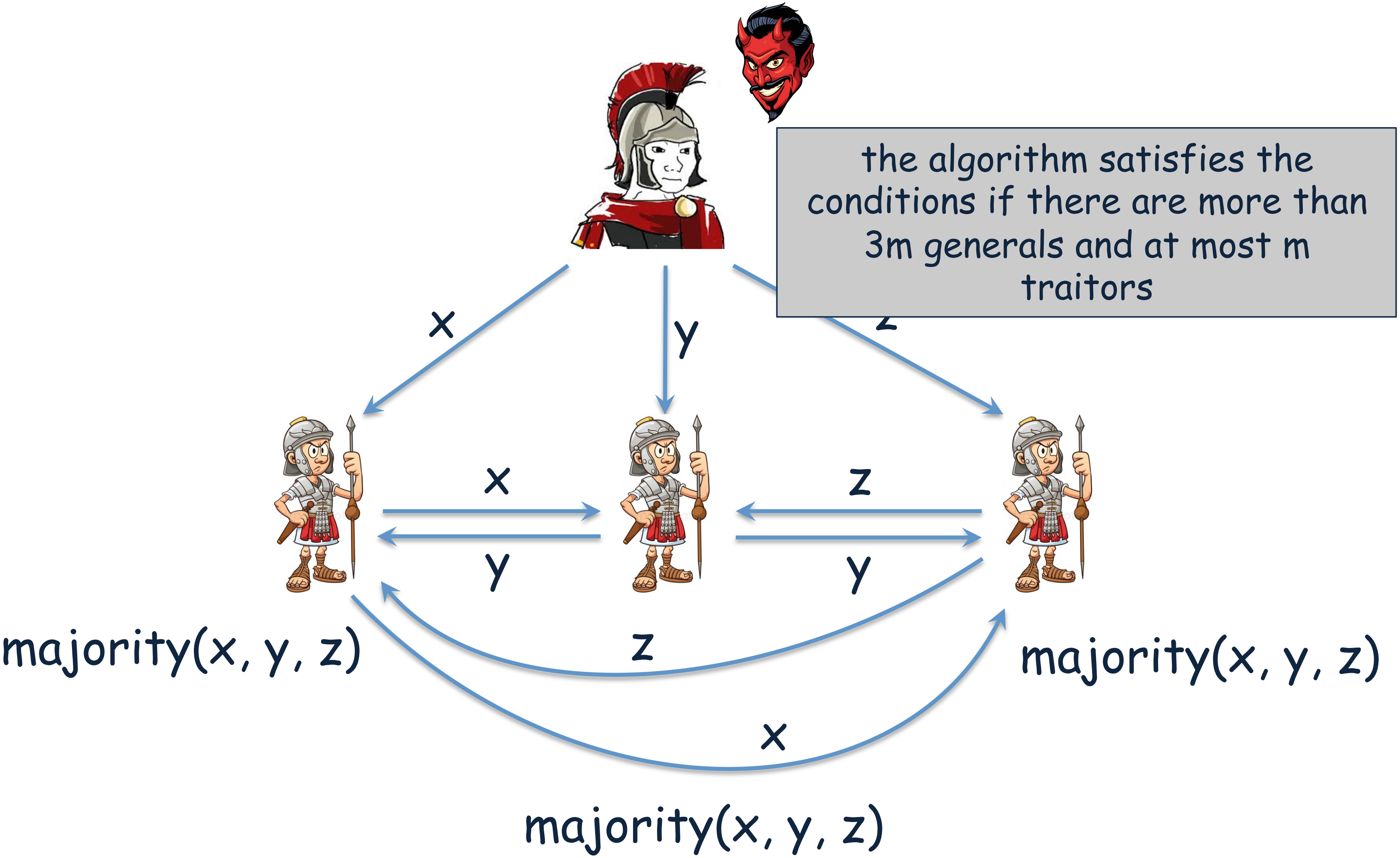
The Byzantine Generals Problem



The Byzantine Generals Problem



The Byzantine Generals Problem



the algorithm satisfies the conditions if there are more than 3m generals and at most m traitors