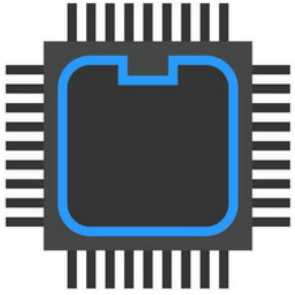
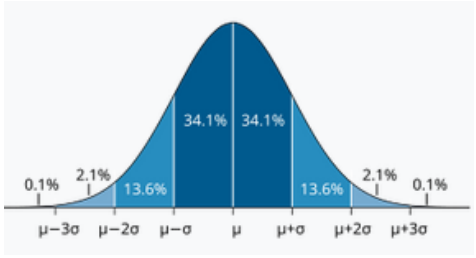
Byzantine General Problems

2105124- Shadman Abid
2105123- Shatabdi Dutta Chowdhury
2105129- Sarowar Alam Roki

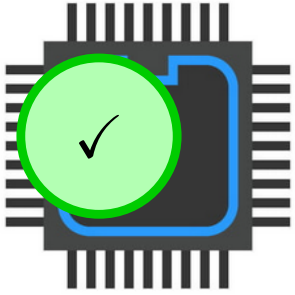
December 11, 2024

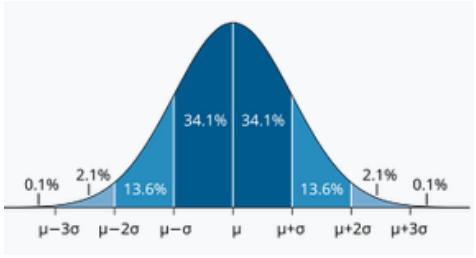
Introduction



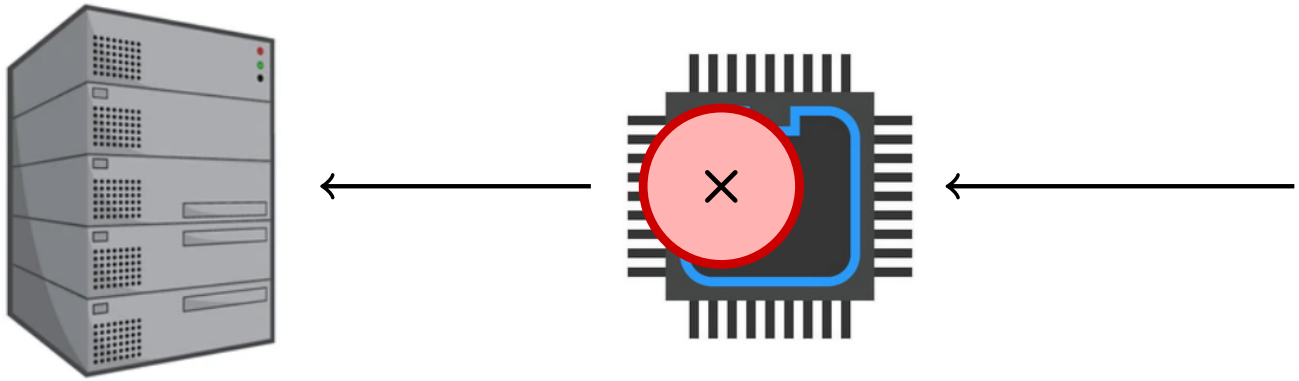


Introduction



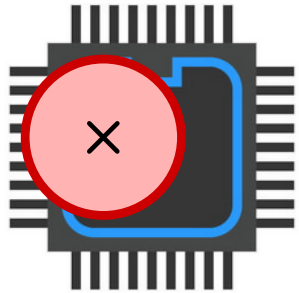


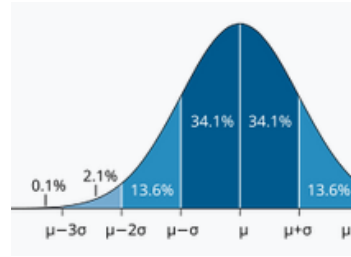
Introduction

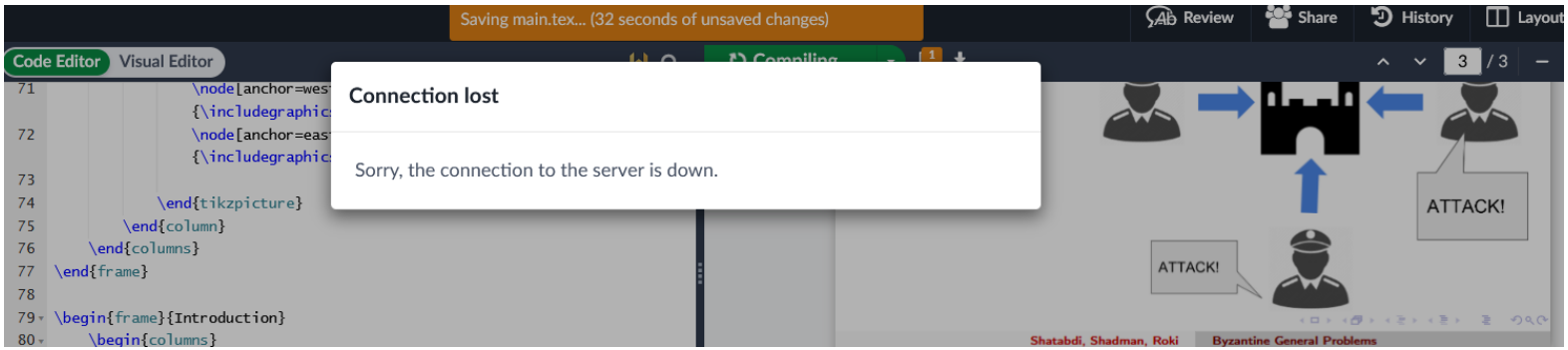



```
} catch (error) {
    console.error("কাজ হচ্ছে না ভাইয়া");
    res.status(500).send('Something went wrong on the server');
}
```

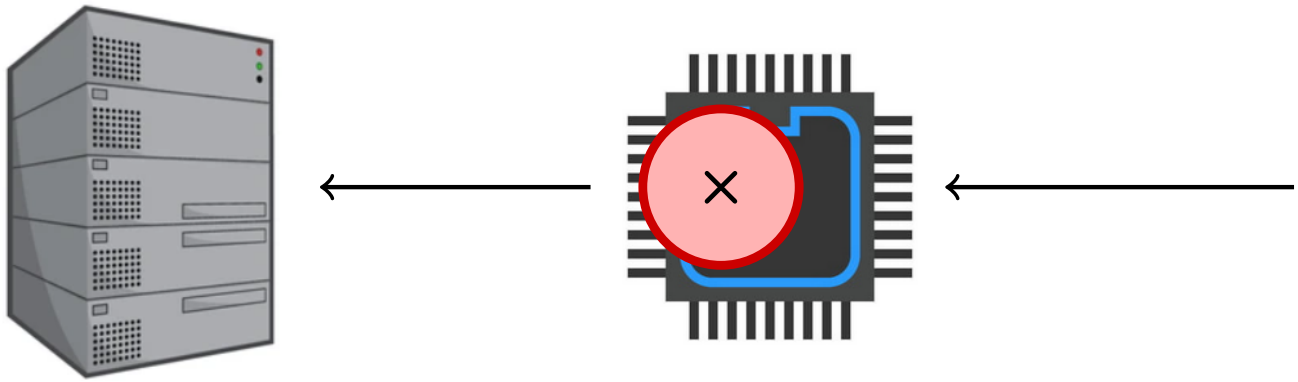
Introduction







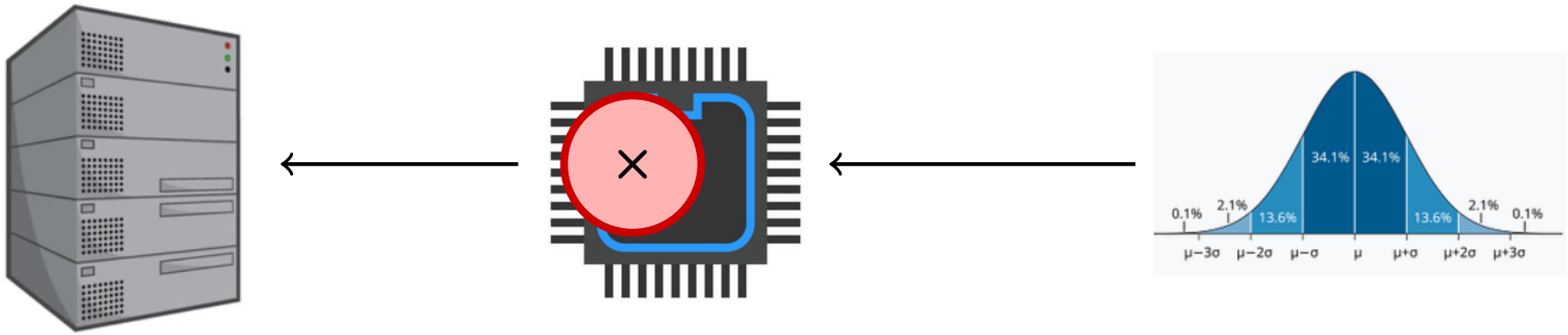
Introduction



- The value of Facebook's share dropped by 5. 5% costing around 7 billion dollars.

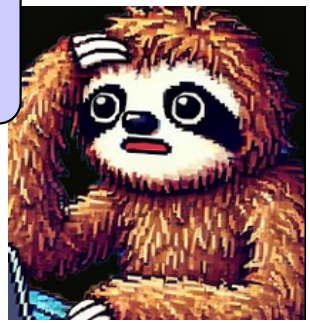
"Wait a second, that's a lot of money, sloth!!!"

Introduction

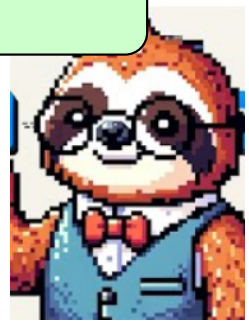


- The value of Facebook's share dropped by 5. 5% costing around 7 billion dollars.
- And also Amazon lost 528 million dollars

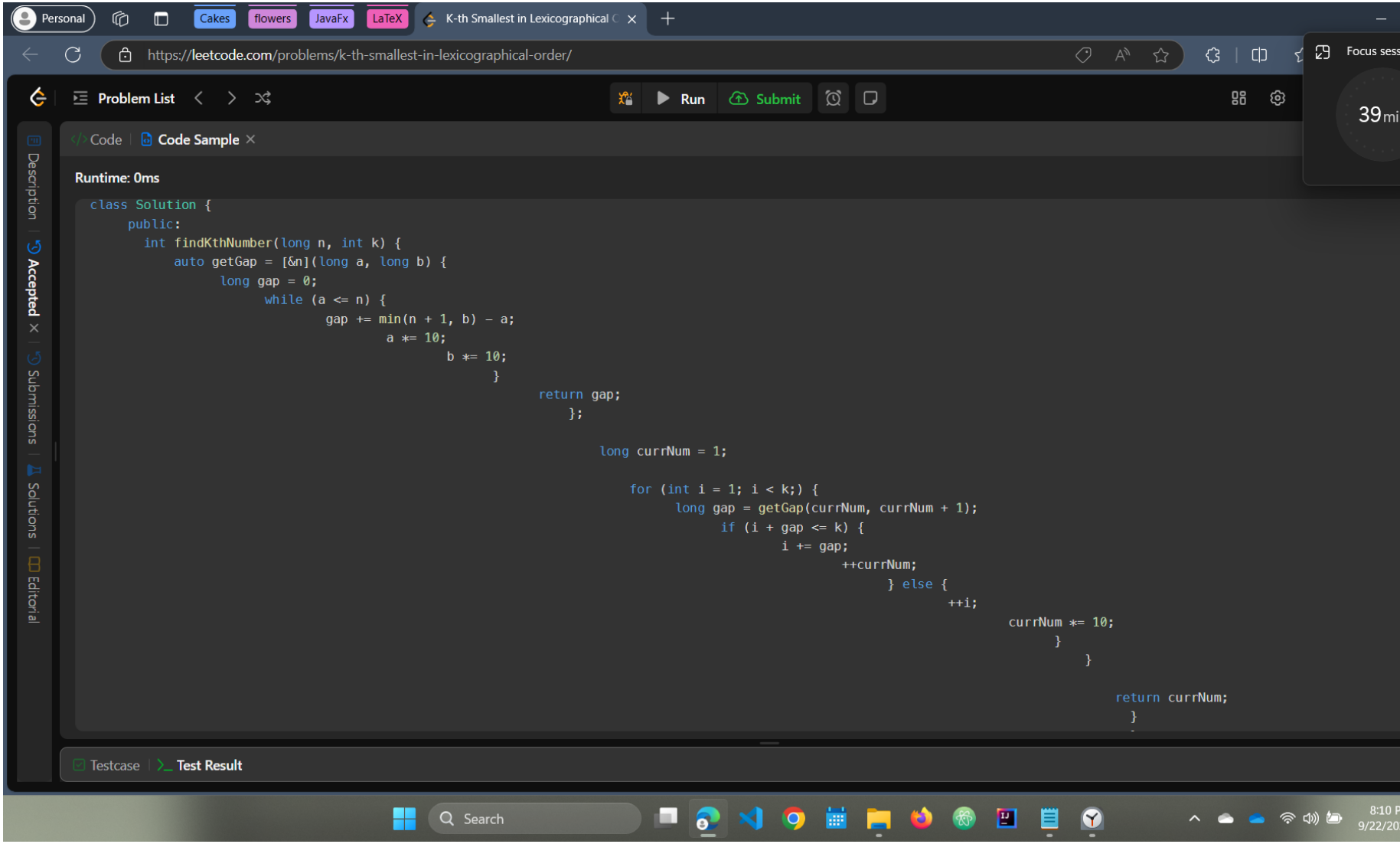
"Wait a second, that's a lot of money, sloth!!!"



"Yep, it is"



Introduction..



The Story Behind...

Hmmm , a white plane with lots of passengers , Seems suspicious!!

The Story Behind...

And ooh look the radar also says so

Hmmm , a white plane with lots of passengers , Seems suspicious!!

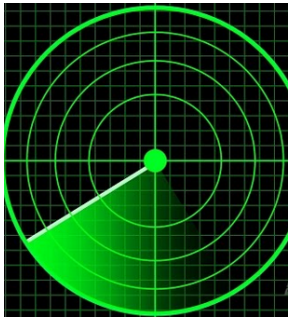
The Story Behind...

Okay boys , Let's nuke that plane , haha

Hmmm , a white plane with lots of passengers , Seems suspicious!!

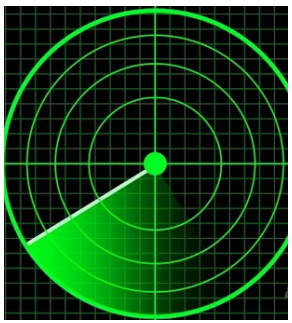
The Story Behind...

Hmm , an F-35 raptor ?
Looks good to me !!



The Story Behind...

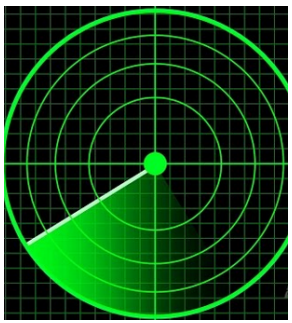
And Look there's nothing on the radar



Hmm , an F-35 raptor ? Looks good to me !!

The Story Behind...

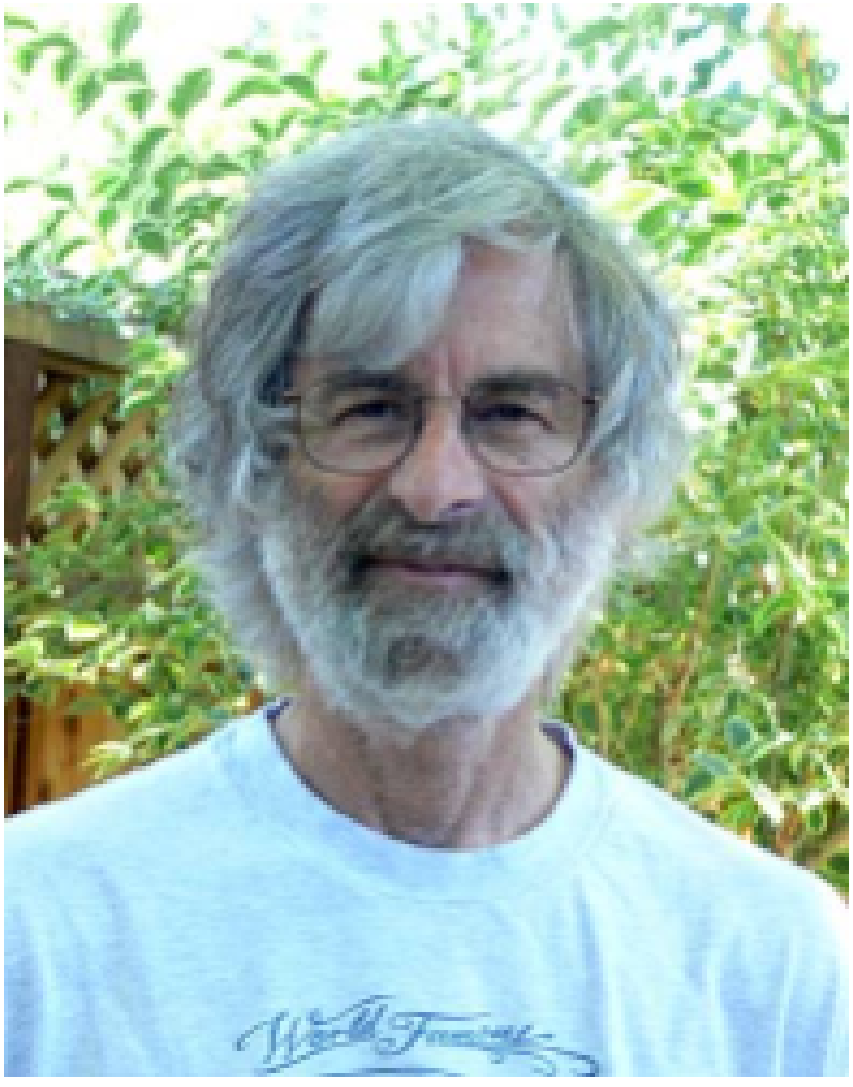
And Look there's nothing on the radar



Hmm , an F-35 raptor ? Looks good to me !!

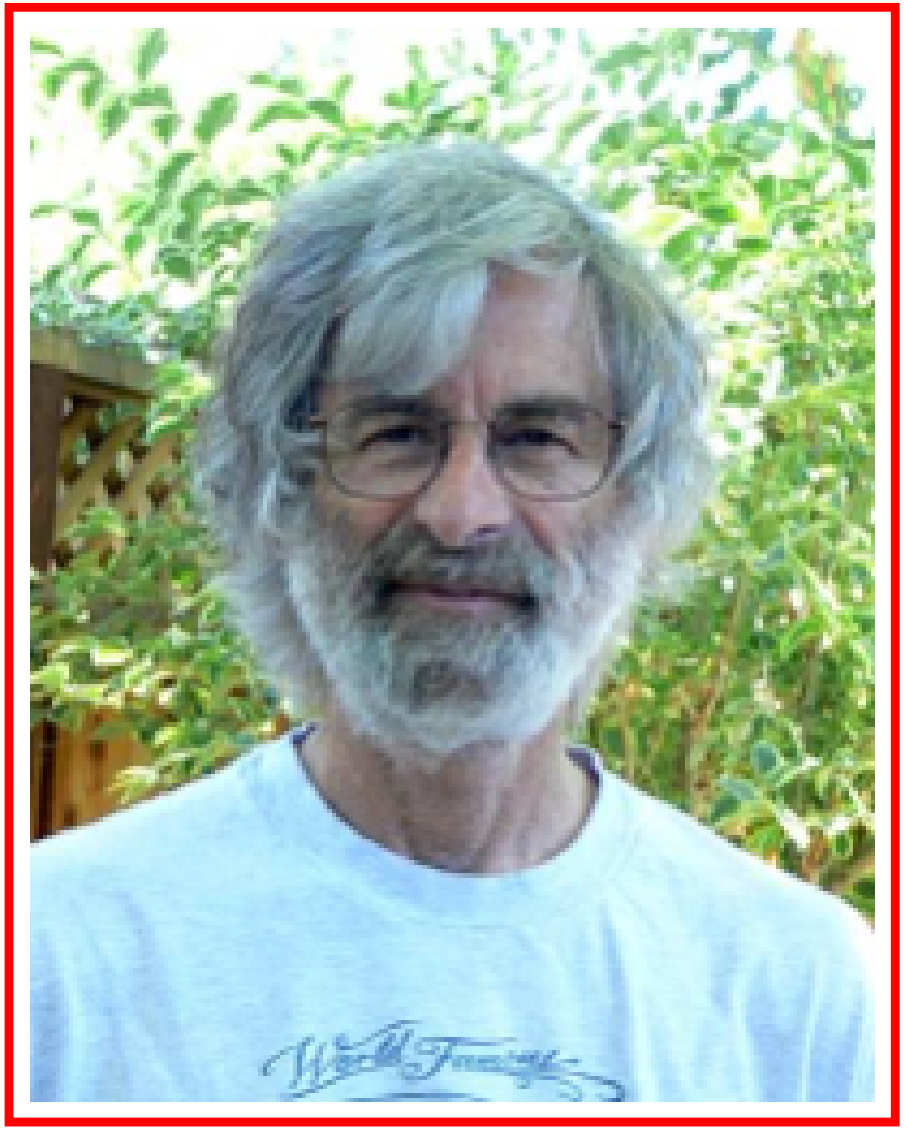
The Story Behind...

Figure: Robert Shoshtack



The Story Behind...

Figure: Robert Shoshtack



Byzantine Generals Problem

Problem Formulation

- Communication through messages.
- Generals must decide upon a plan
- Concensus

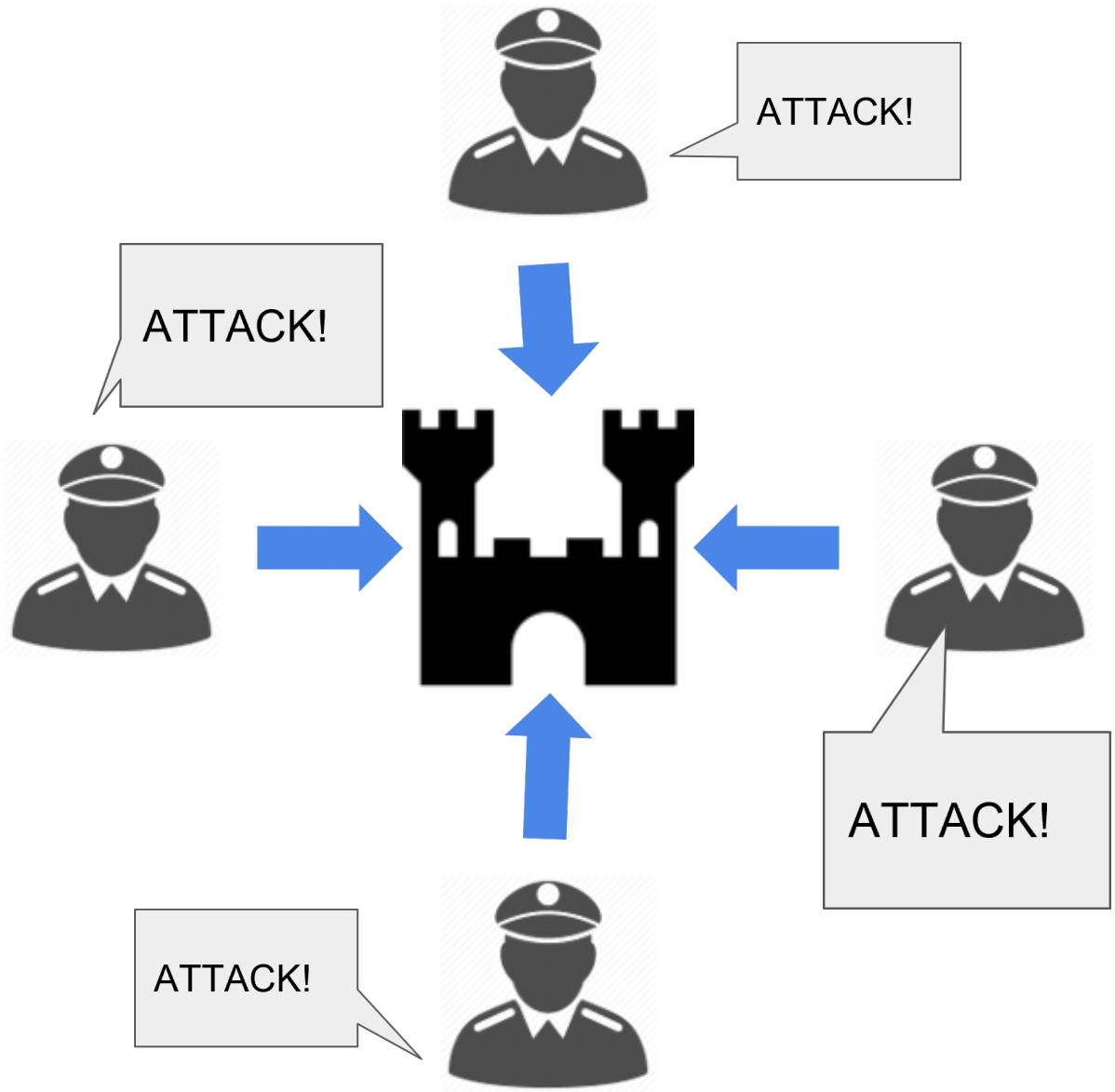




Byzantine Generals Problem

Consensus to Attack

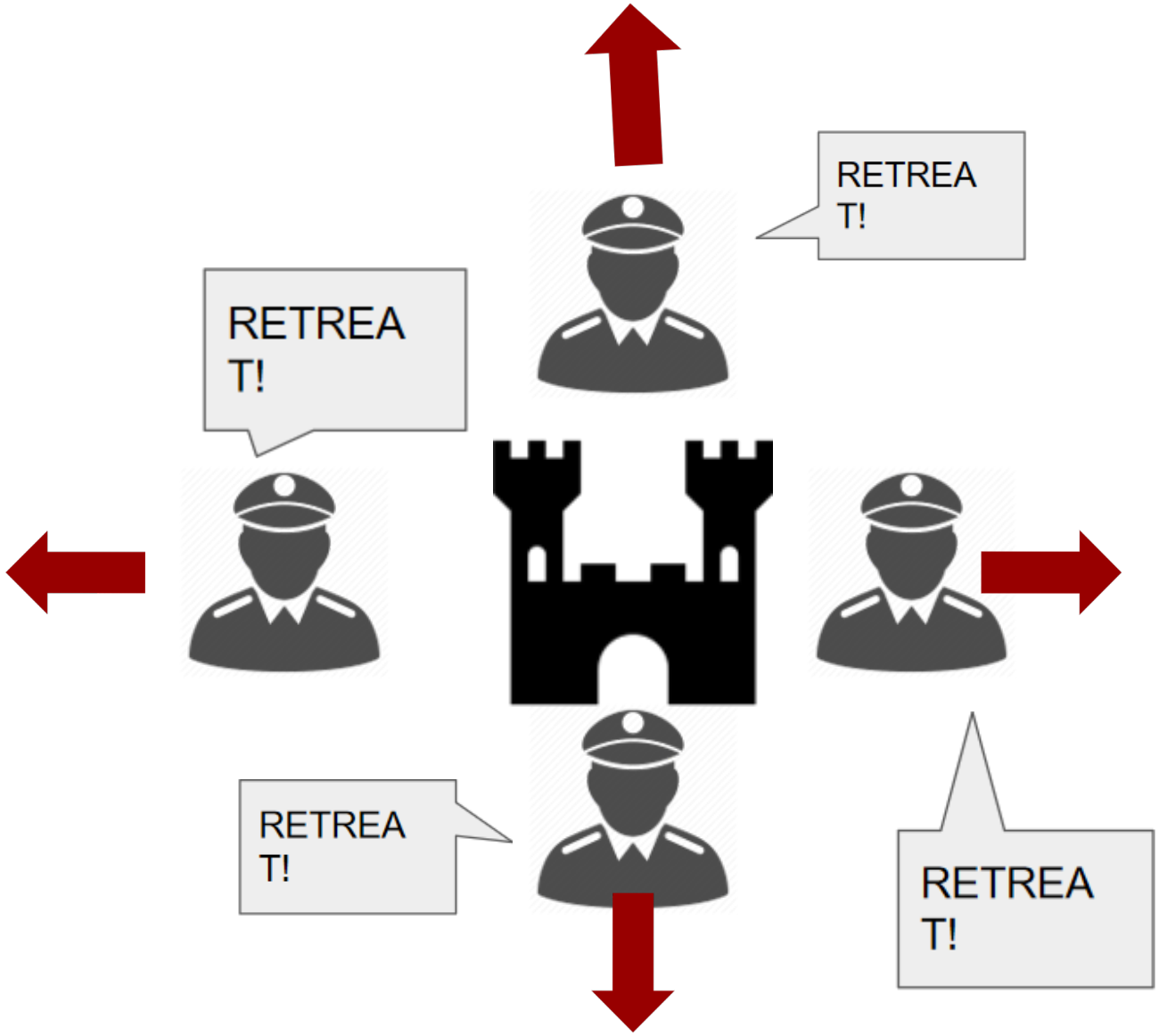
- ***Attack***



Byzantine Generals Problem

Consensus to Retreat

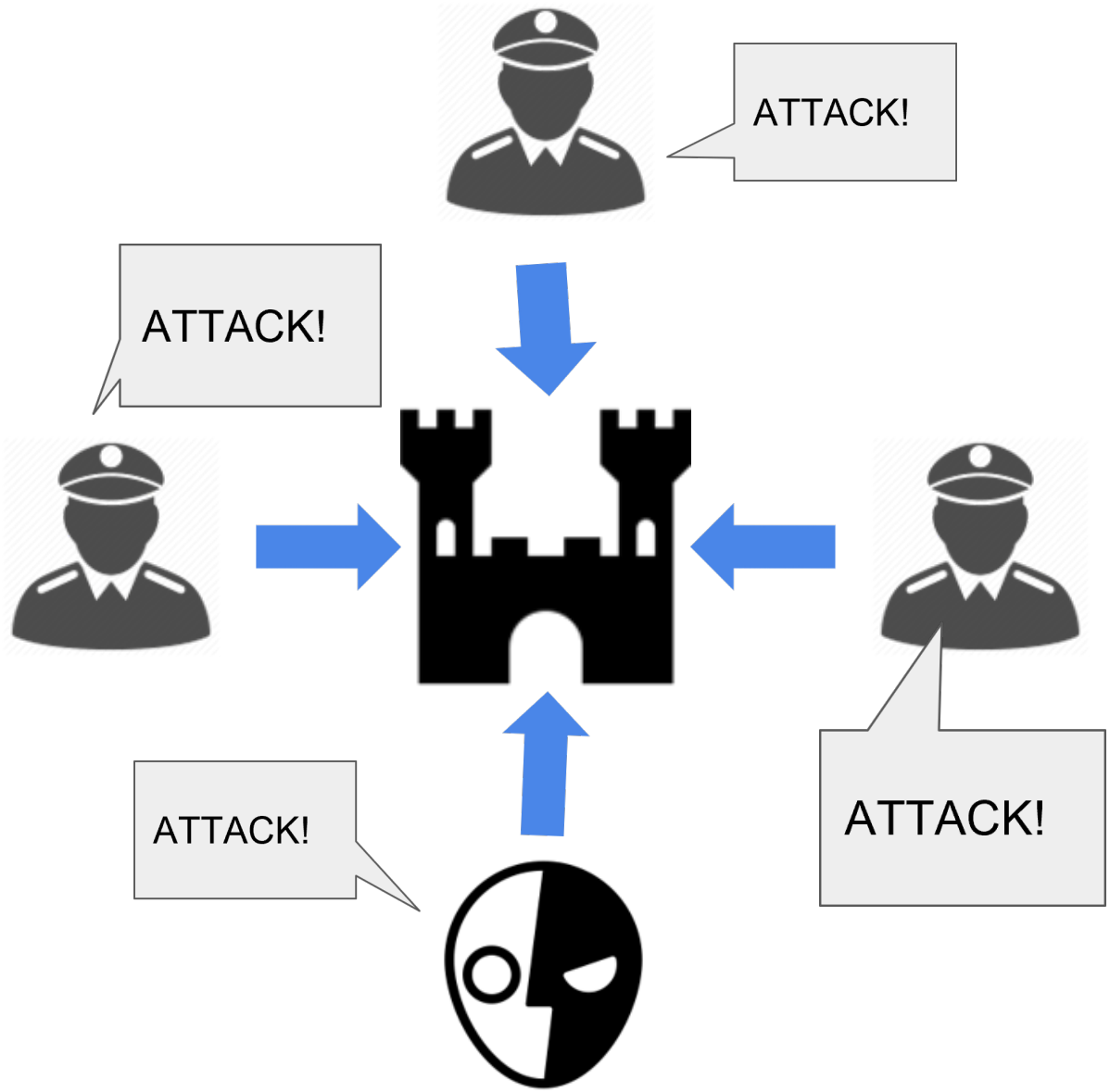
- *Retreat*



Byzantine Generals Problem

Consensus to Attack with Traitor Included

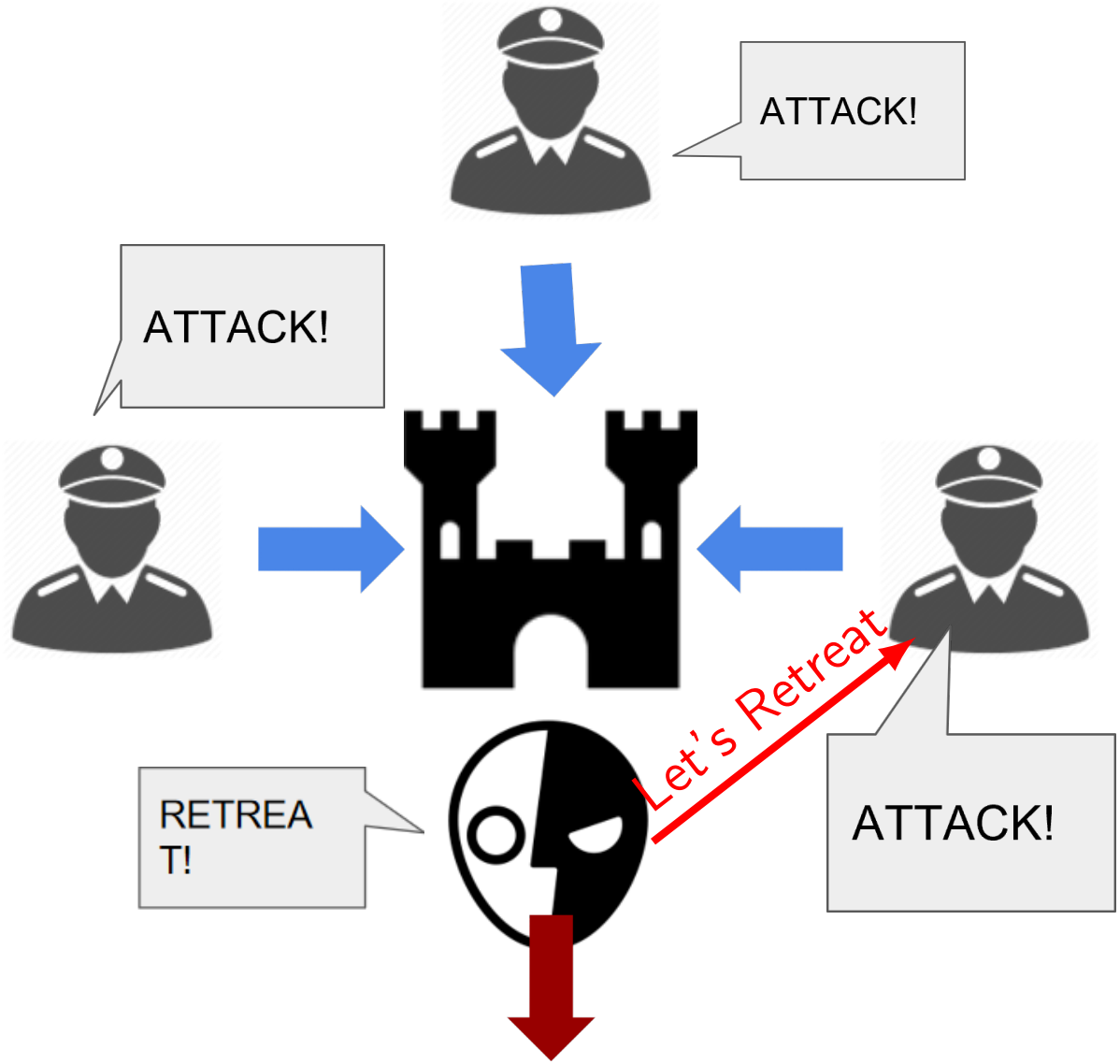
- Traitors
- The *loyal Generals* must reach a *consensus* .



Byzantine Generals Problem

Traitor Interfering

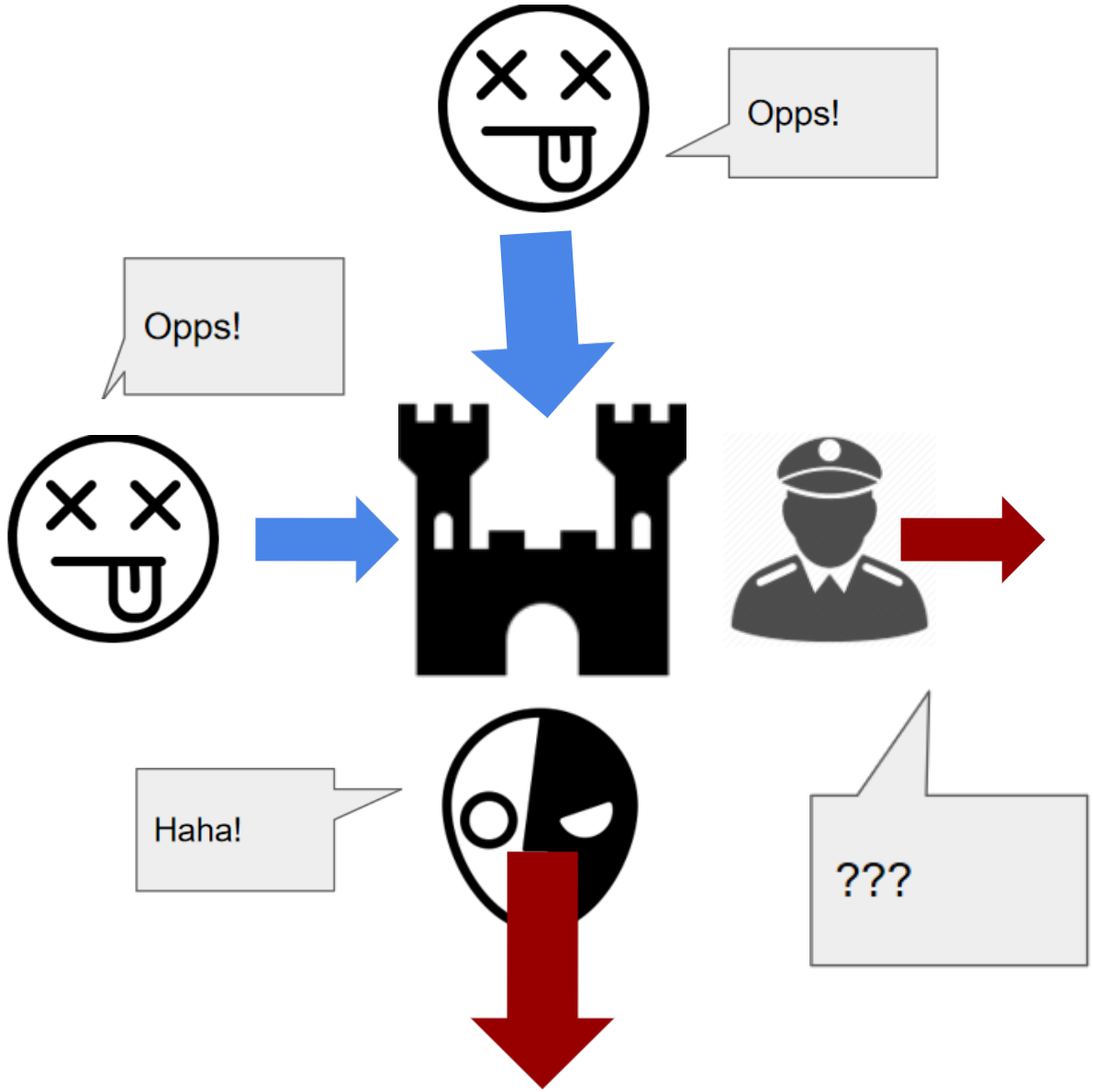
- Traitors can act arbitrarily .



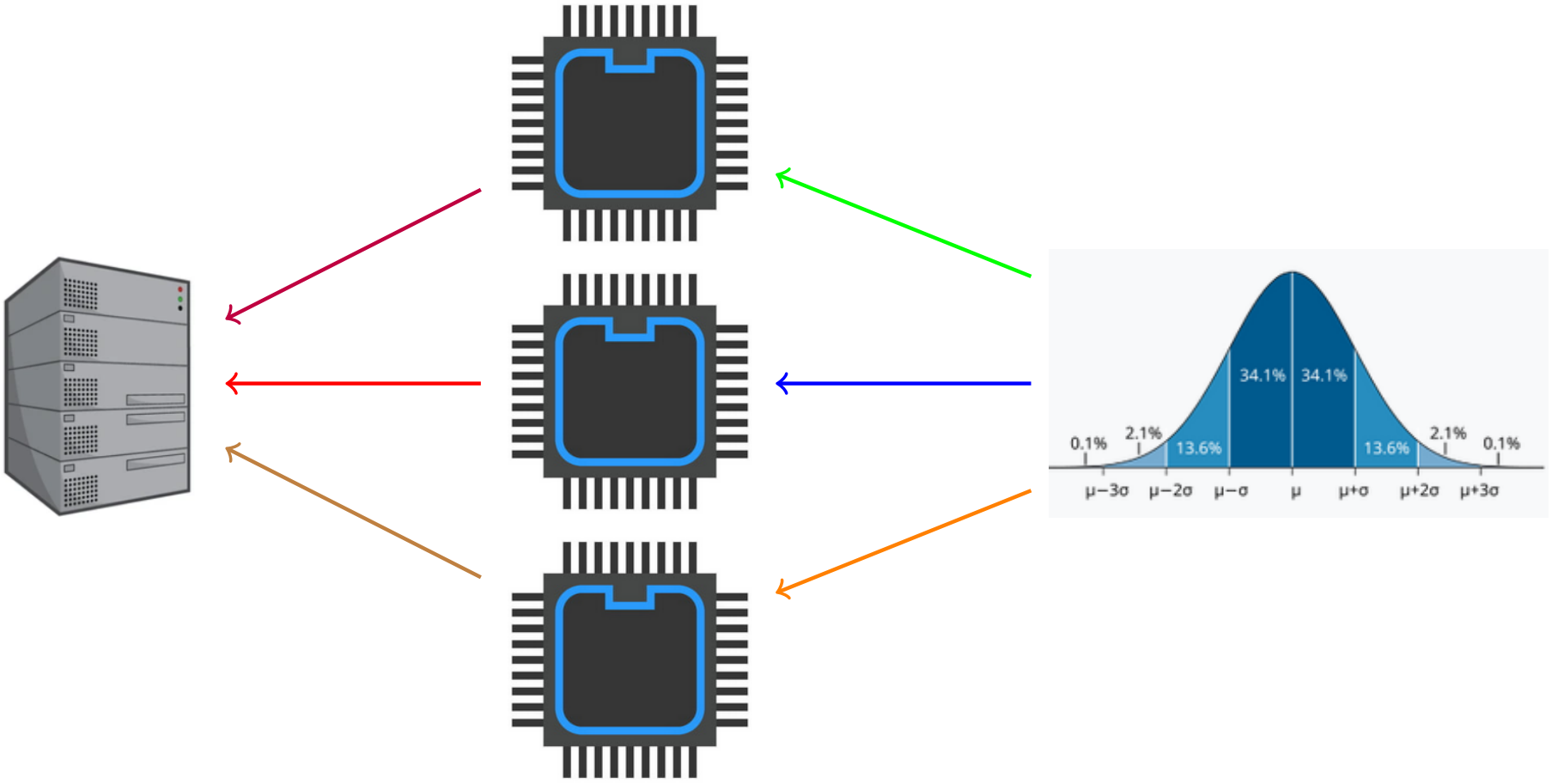
Byzantine Generals Problem

Traitor is Successful

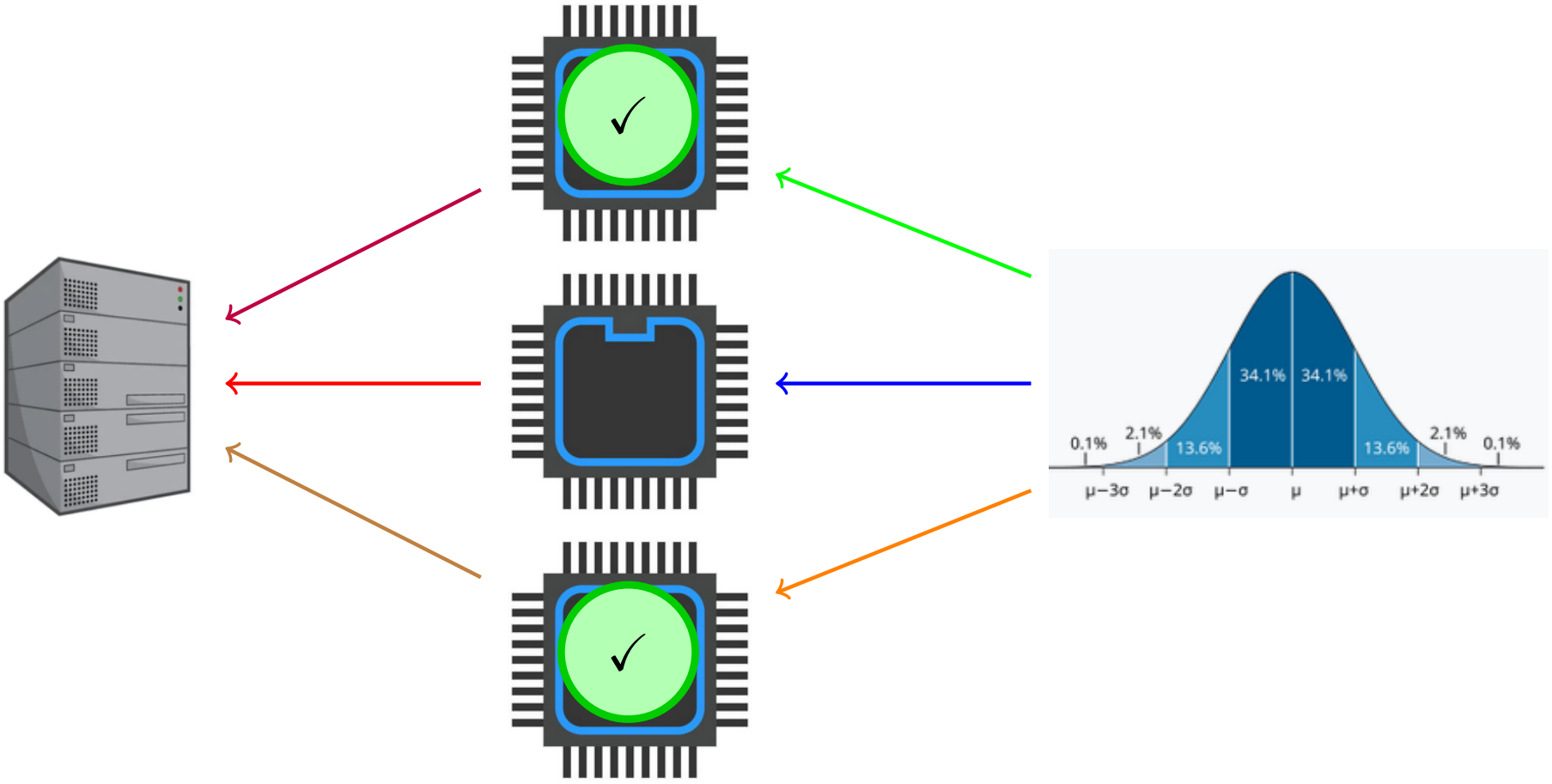
- A traitor can easily foil any effective strategy if he is not dealt with.



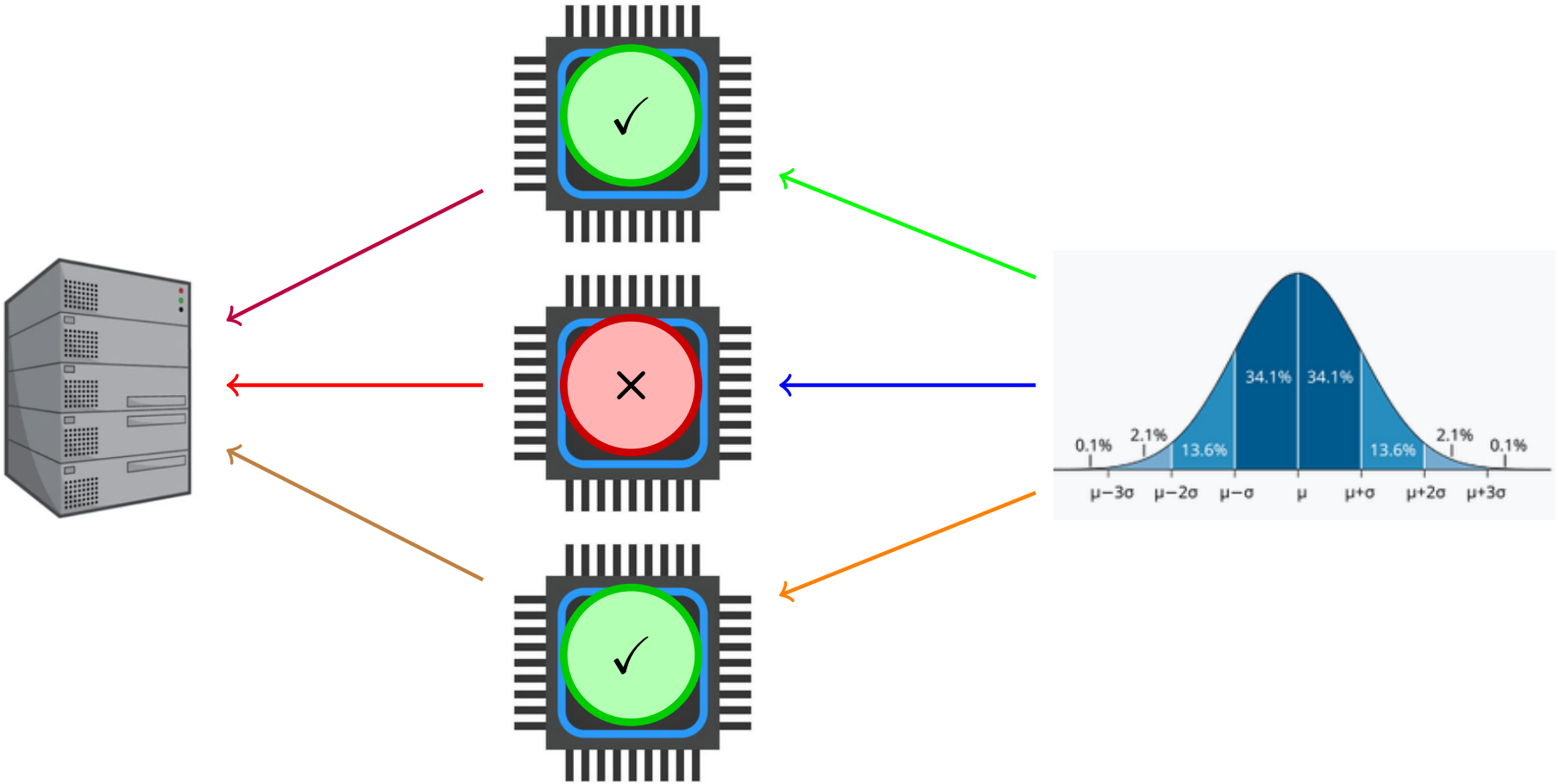
Relevance of Byzantine Generals Problem to Computer Science..



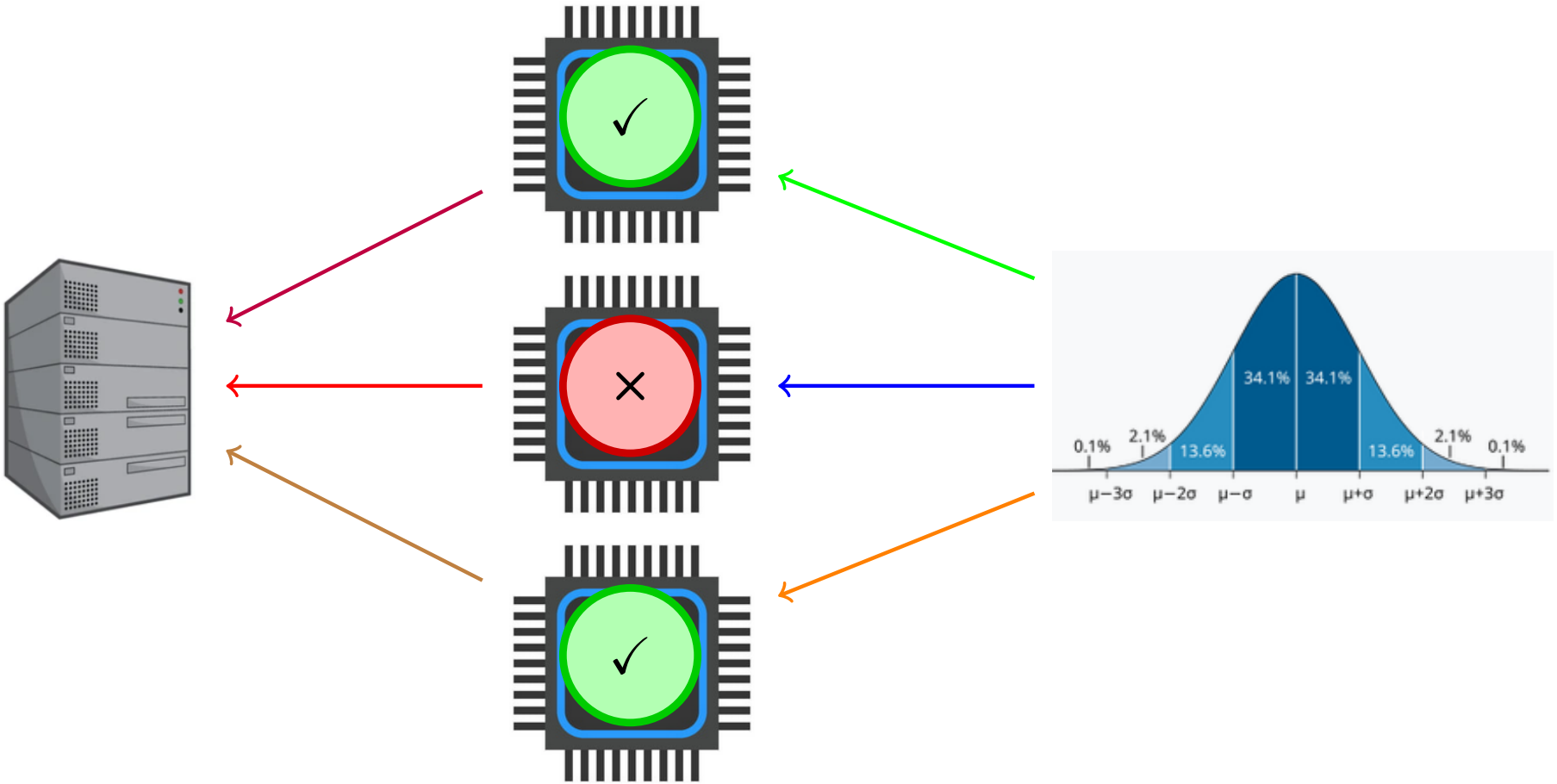
Relevance of Byzantine Generals Problem to Computer Science..



Relevance of Byzantine Generals Problem to Computer Science..



Relevance of Byzantine Generals Problem to Computer Science..



Same goes for radars and sensitive identification technology.

The Problem Revisited...

The Byzantine Generals Problem

The **Byzantine Generals Problem** is a way to understand how computer systems can agree on a decision, even when some parts might not work correctly or try to deceive others.

- We can relate it with the situation where how a group of generals, represented as computer nodes or processes, must agree on a decision (e.g., whether to attack or retreat), despite the presence of traitors among them.

A Simplified Version

There are a total of n officers on the battlefield.

Figure: Battlefield Scenario with Generals

A Simplified Version...

There are a total of n officers on the battlefield.

- One officer is the **commanding general**, who will send order to the remaining n-1 are **lieutenant generals** such that
  - **Condition 1:** All loyal lieutenants must obey the same order.
  - **Condition 2:** If the commanding general is loyal, all loyal lieutenants must follow the order he sends.

Figure: Battlefield Scenario with Generals

On the way of solution...

What is Byzantine generals problem

- All loyal lieutenants obey the same order.
- If the commanding general is loyal, then every loyal lieutenant obeys the order he sends.

On the way of solution...

What is Byzantine generals problem

- Consistency/Agreement
- If the commanding general is loyal, then every loyal lieutenant obeys the order he sends.

On the way of solution...

What is Byzantine generals problem

- Consistency/Agreement
- **If the commanding general is loyal**, then every loyal lieutenant obeys the order he sends.

On the way of solution...

What is Byzantine generals problem

- Consistency/Agreement
- Validity

On the way of solution...

What is Byzantine generals problem

- Consistency/Agreement
- Validity
- Termination

Impossibility Result

Impossibility Result

Now we will go through two main strategies.

- Oral messages
- Signed messages

Impossibility Result: Oral Messages

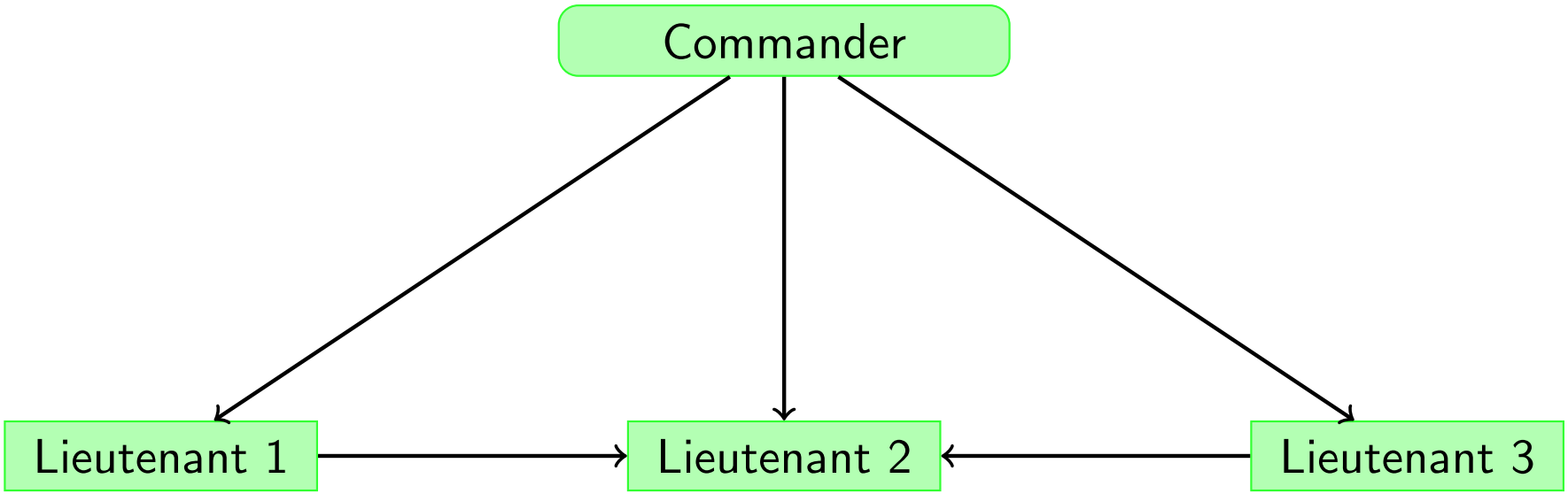
"if the generals can send only oral messages, then no solution will work unless more than $\frac{2}{3}$ of the generals are loyal."

Oral Messages

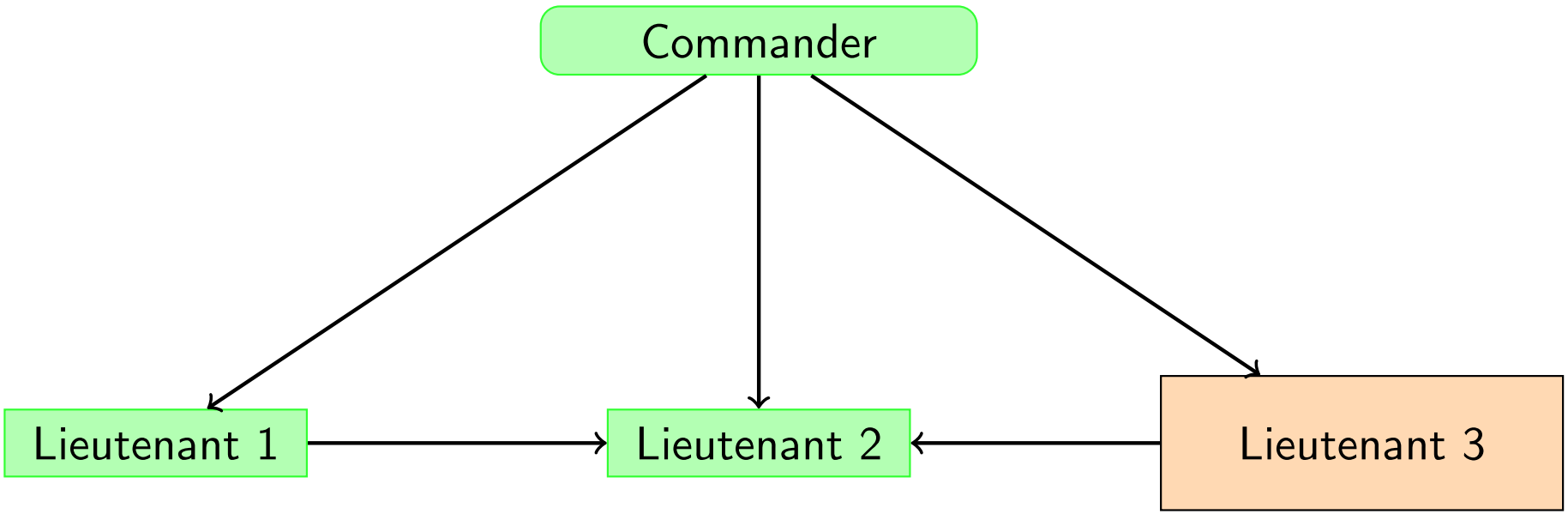
What are ORAL MESSAGES ???

A Possible Scenerio

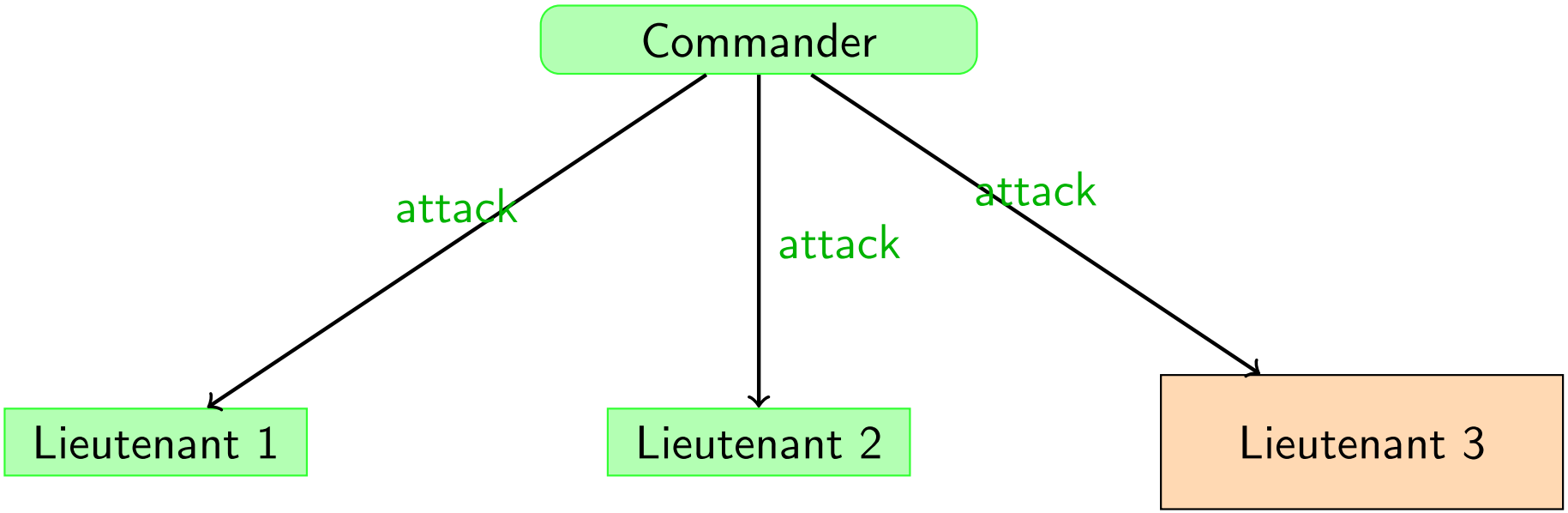
Flowchart possibility Result



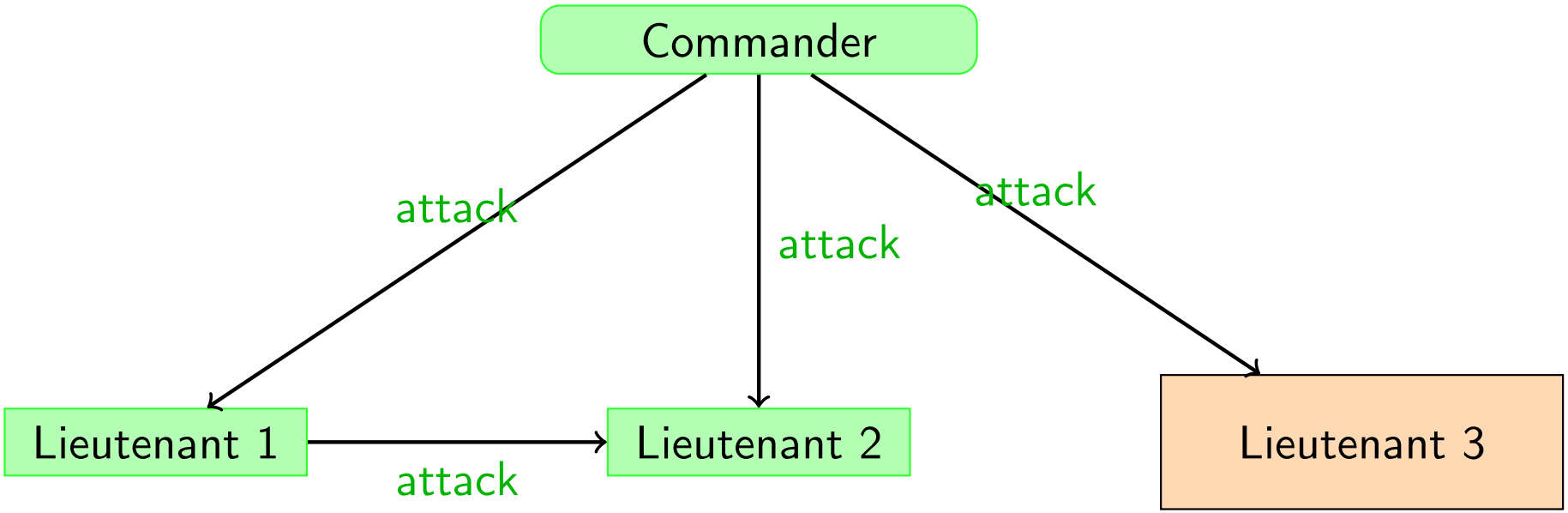
Flowchart Possibility Result



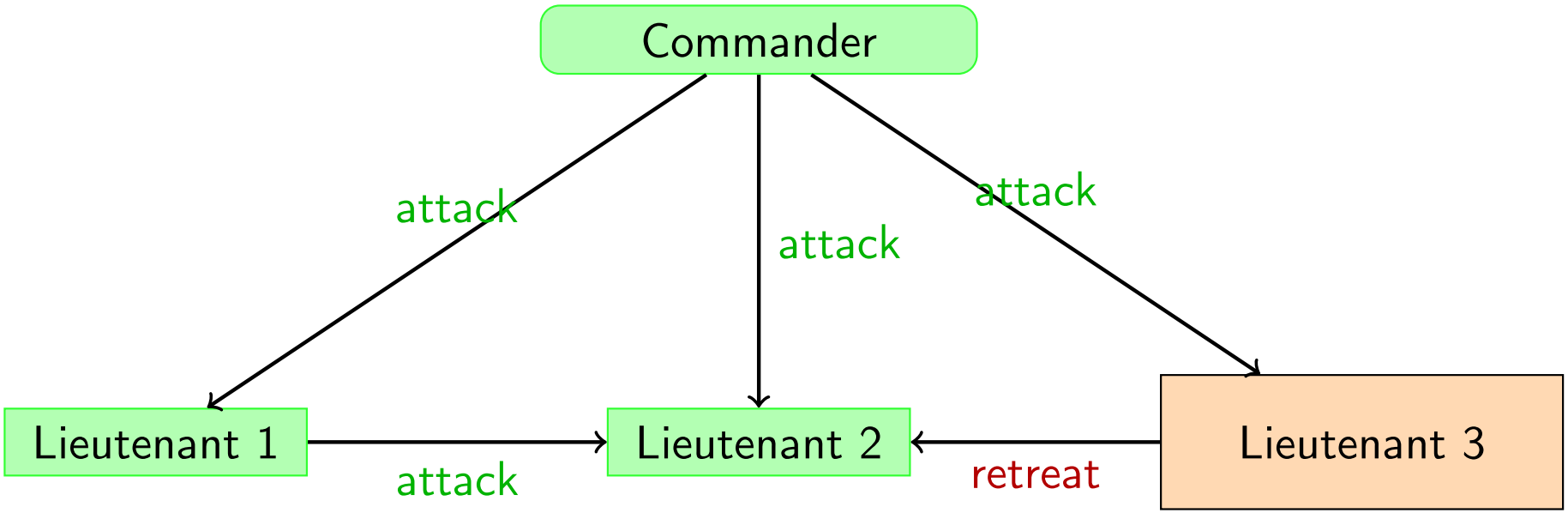
Flowchart Possibility Result



Flowchart Possibility Result



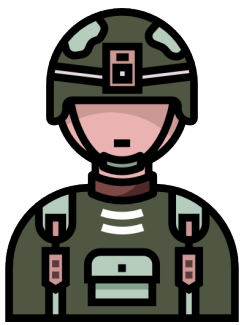
Flowchart Possibility Result

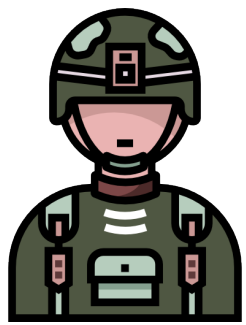


Impossiblity Proof

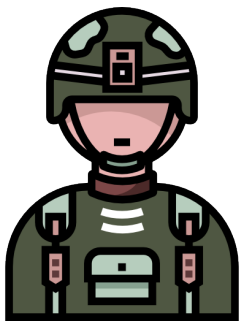
When only one of the generals or the commander is traitor.So the traitor is only one memeber. So the member , m=1

Impossibility Result Proof[m=1]

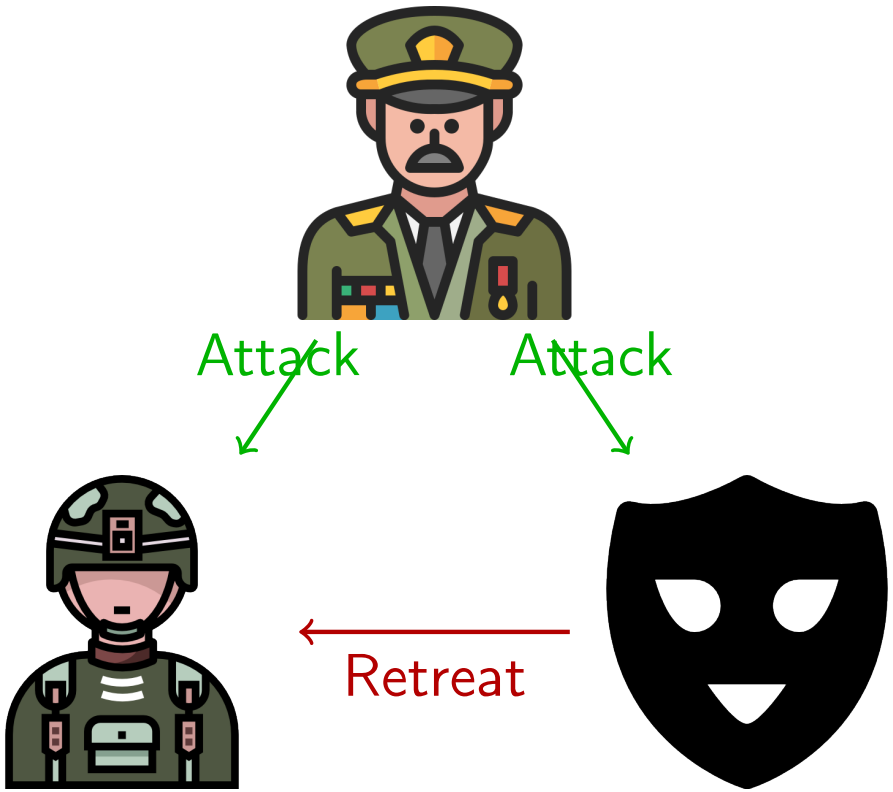




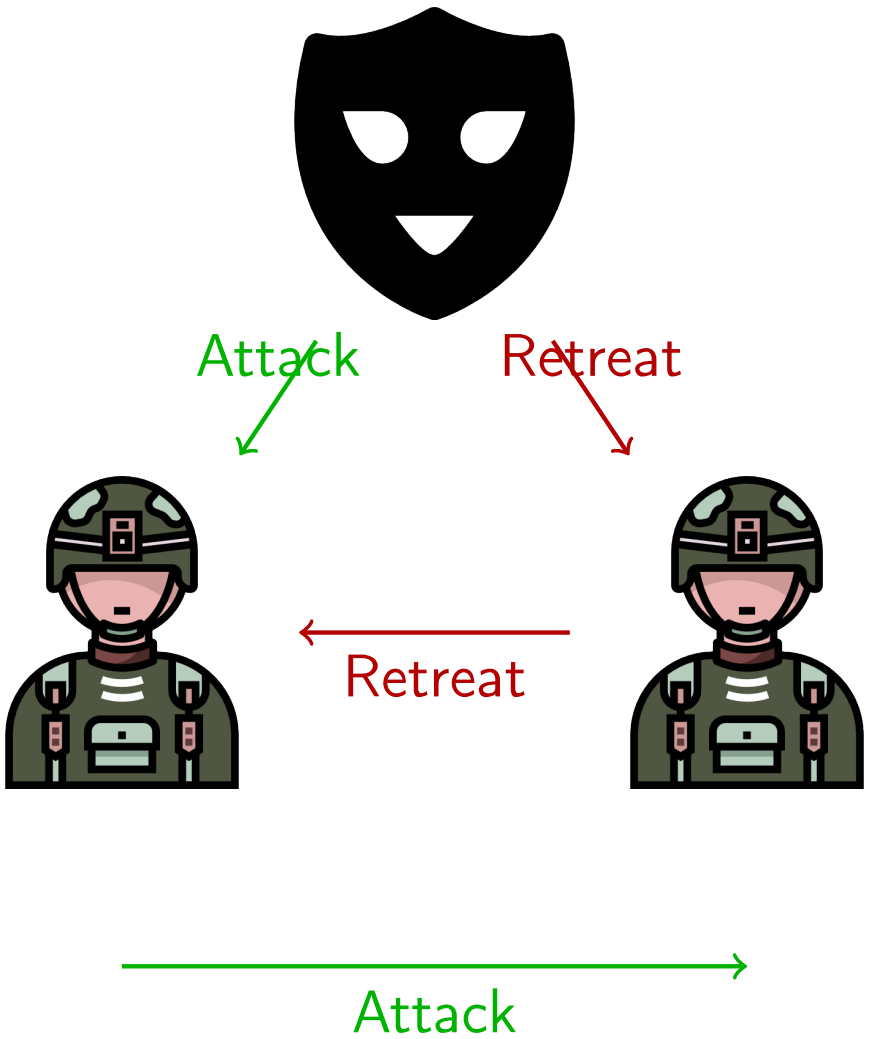
Impossibility Result Proof[m=1]



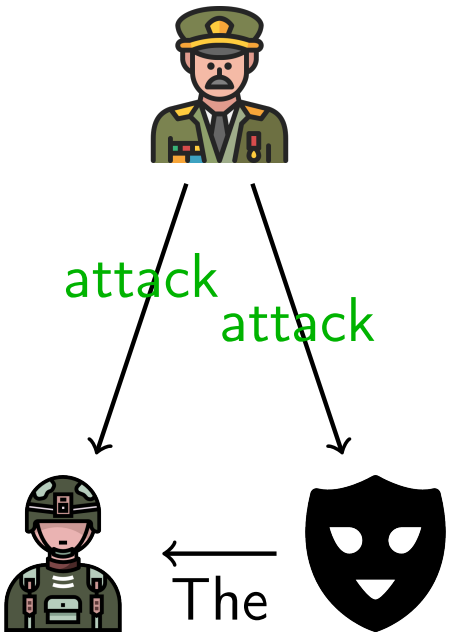
Impossibility Result Proof[m=1]

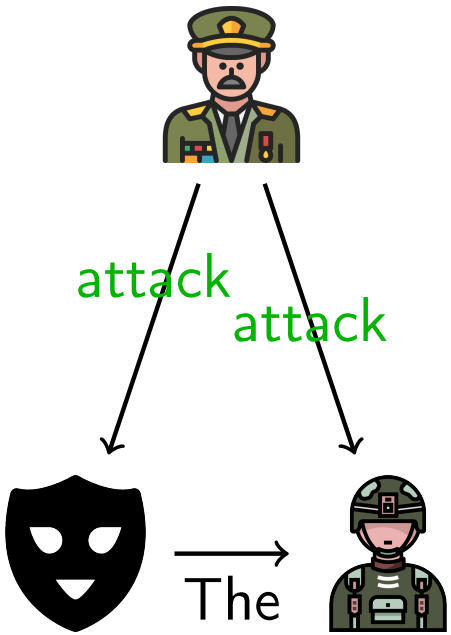


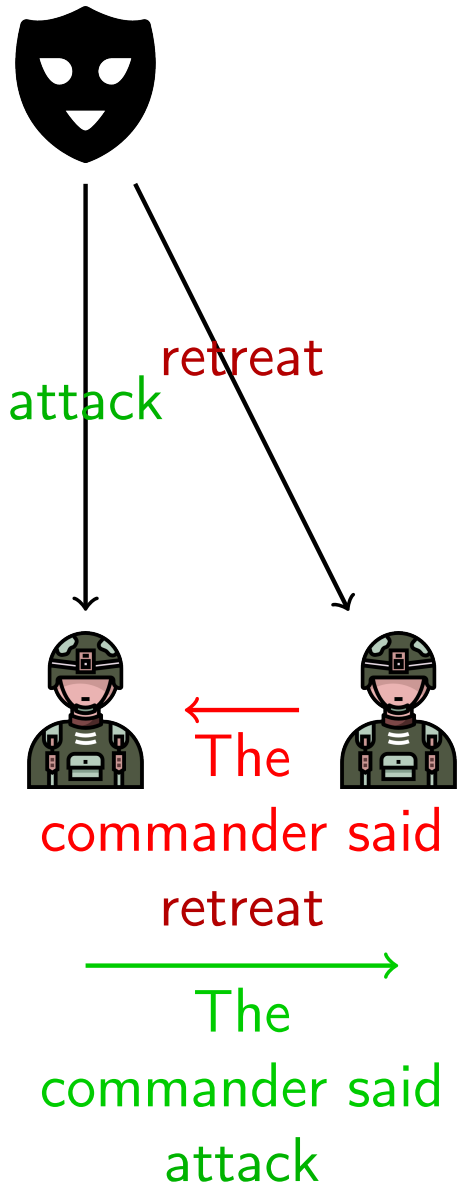
Impossibility Result Proof[m=1]



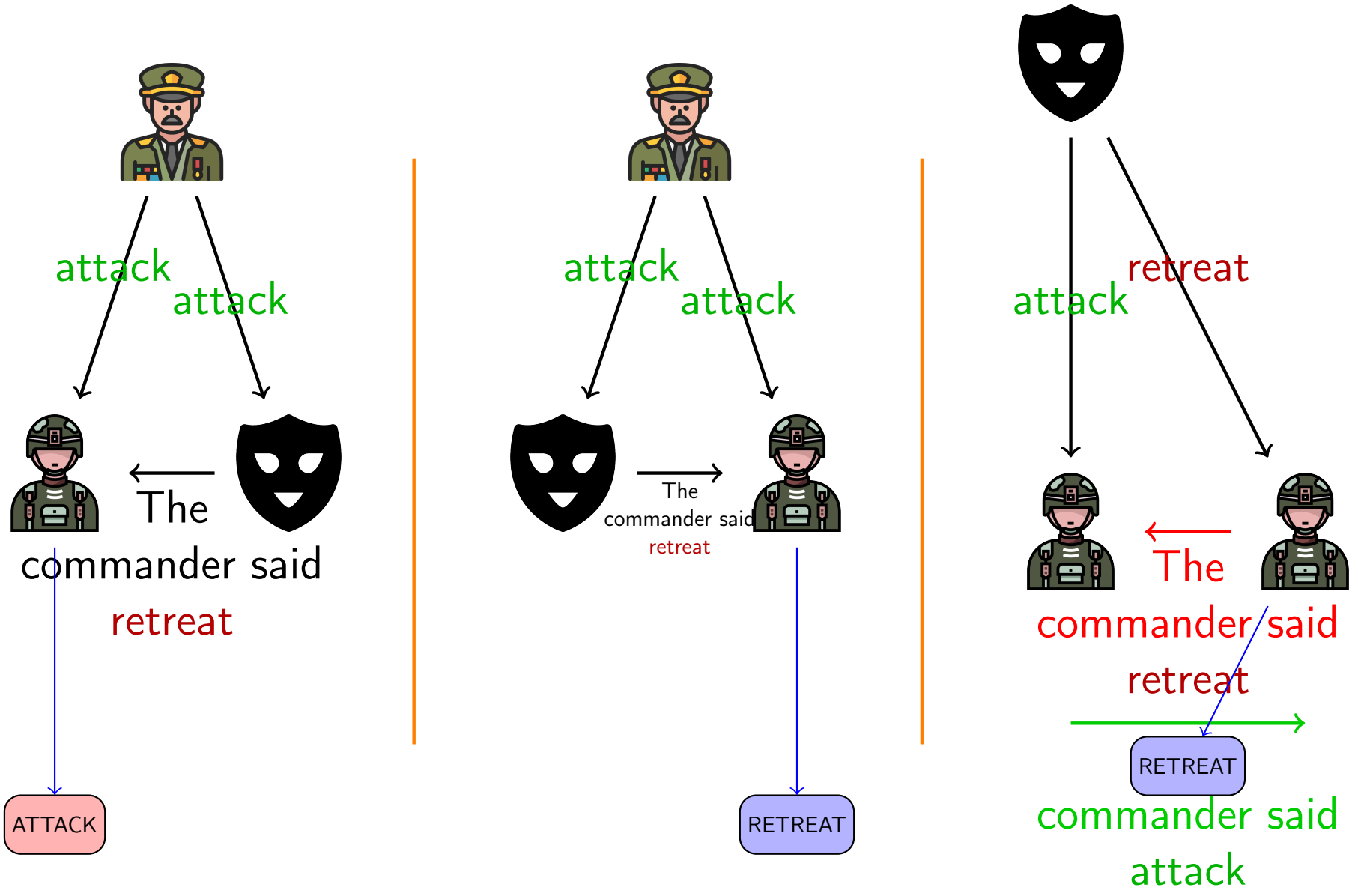
Three Scenarios [m=1]



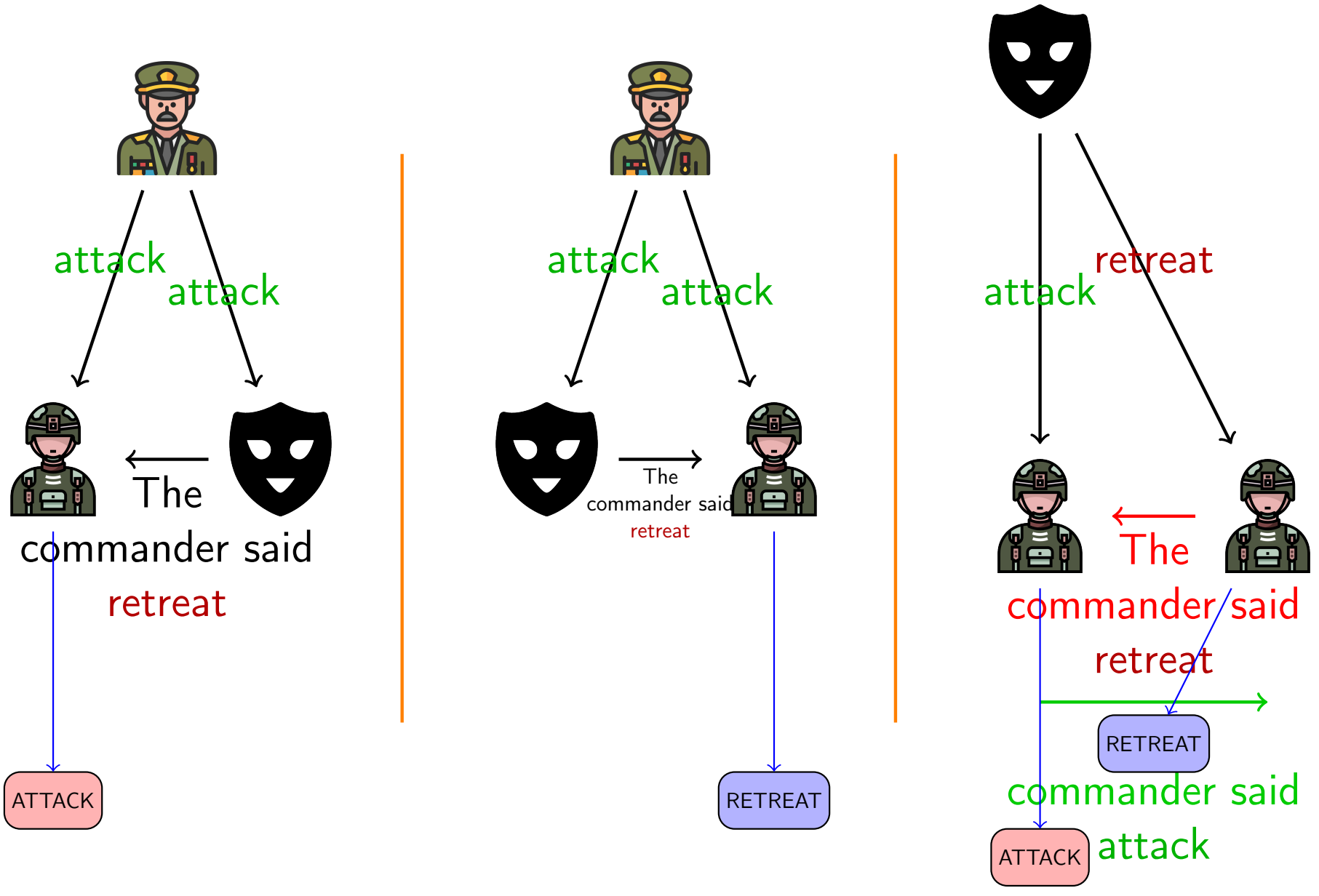




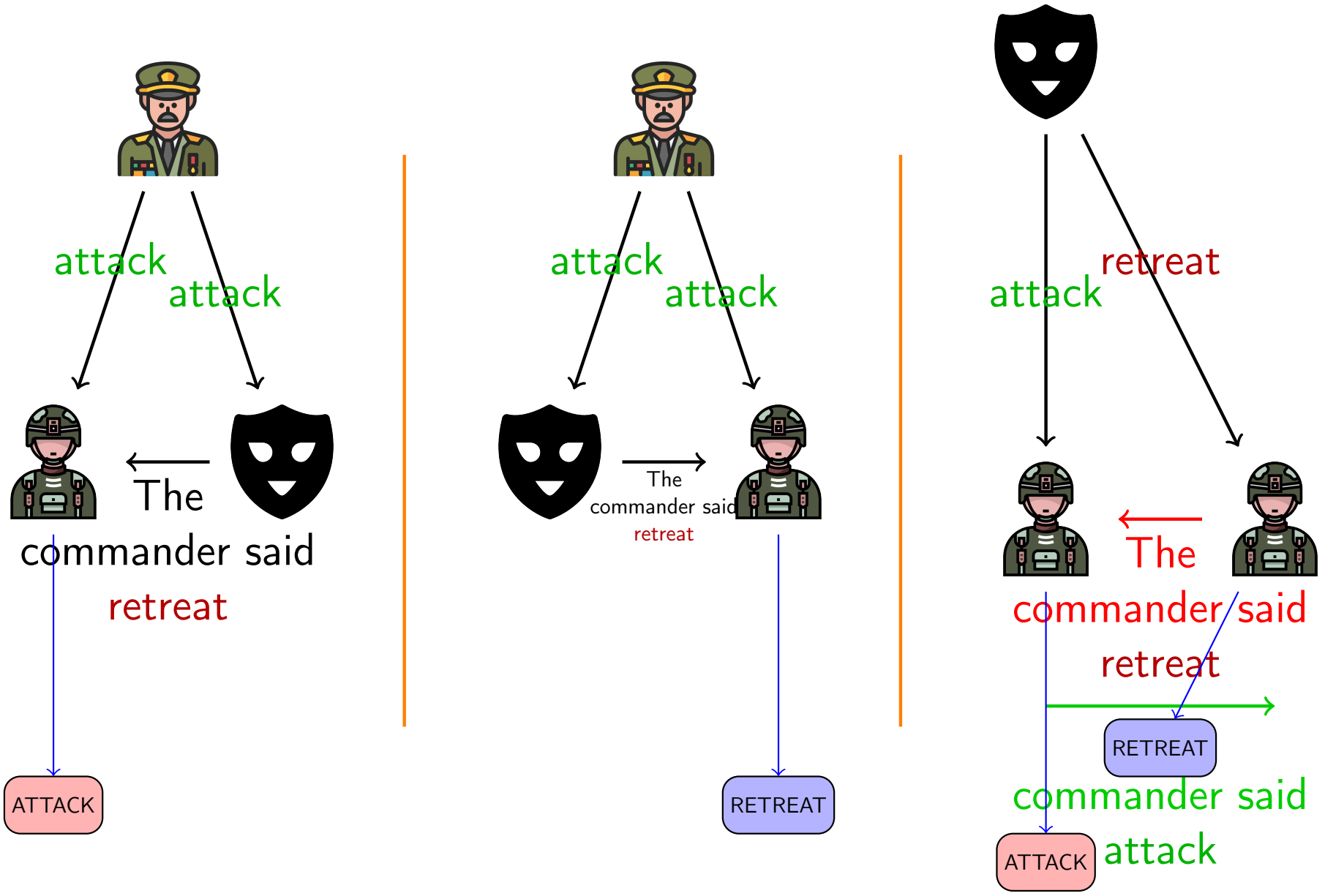
Three Scenarios



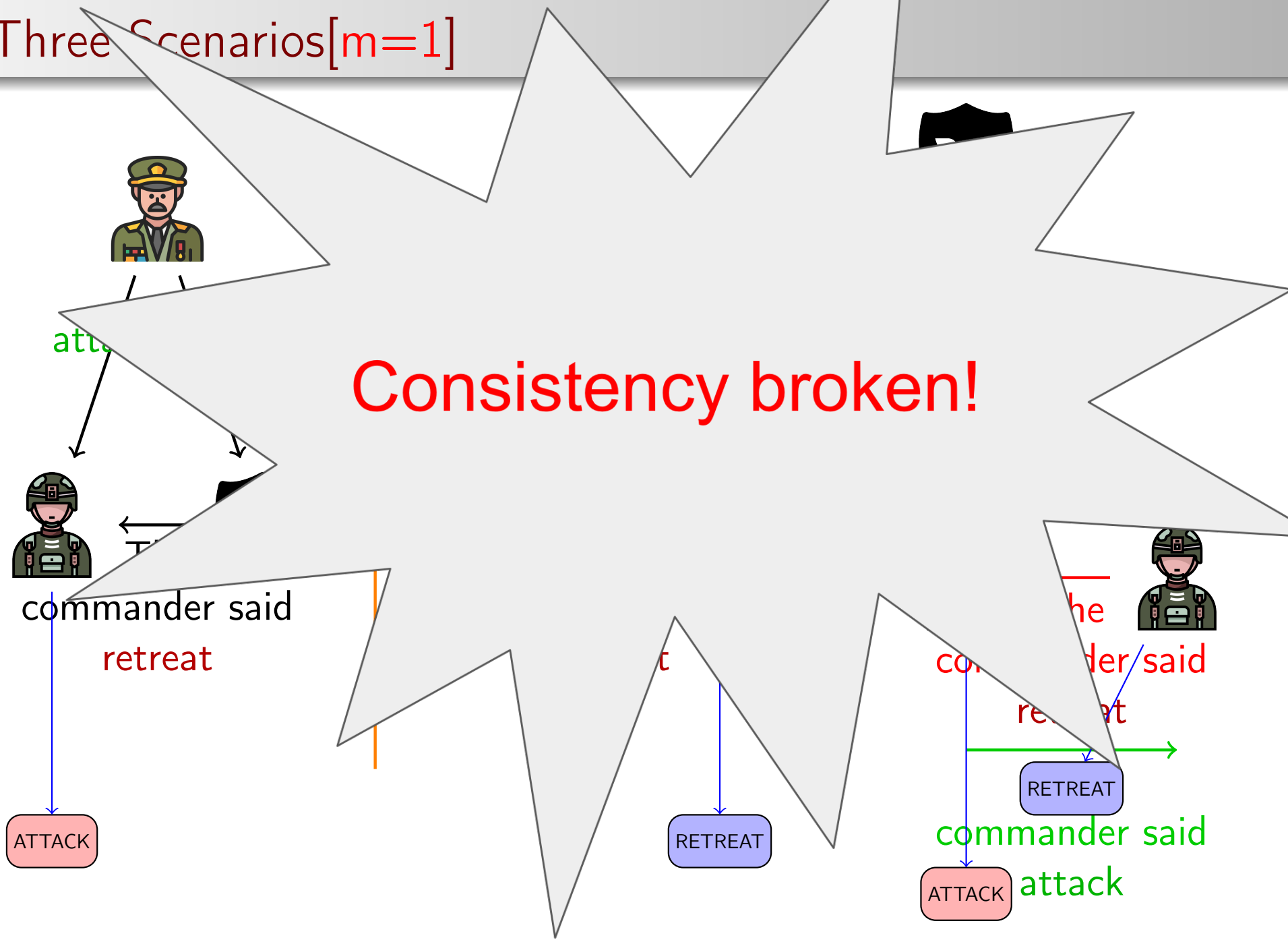
Three Scenarios



Three Scenarios

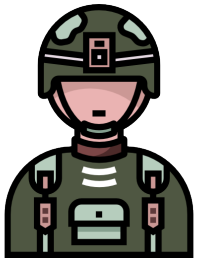


Three Scenarios[m=1]

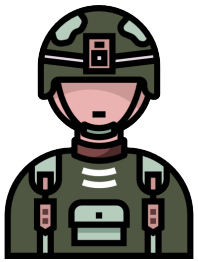


Impossibility Result m>1

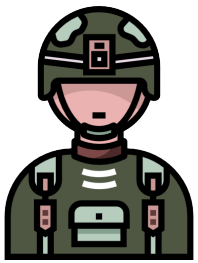
Proof by cointradiction:



x



y

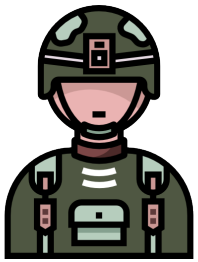


z

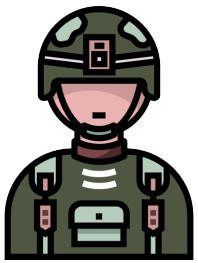
Impossibility Result m>1

According to protocol f

x



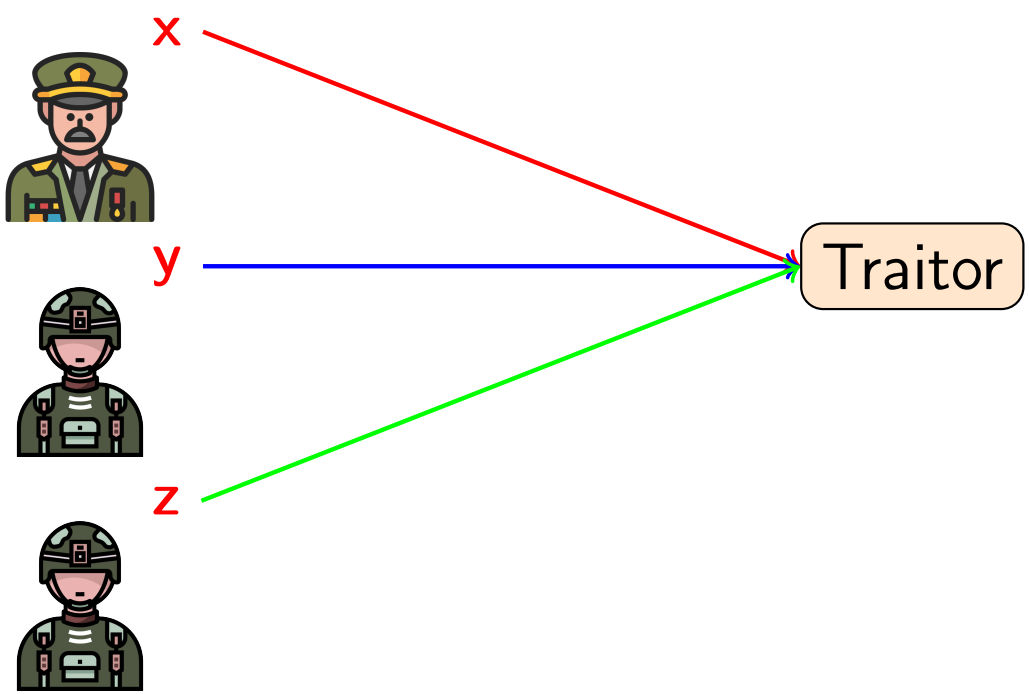
y



z

Impossibility Result[m>1]

According to protocol f



To be continued...

At most m simulated traitors

Protocol $\mathbf{f}$ can solve it...

Impossibility Result m>1

Assumption: $m > 1$ by contradiction

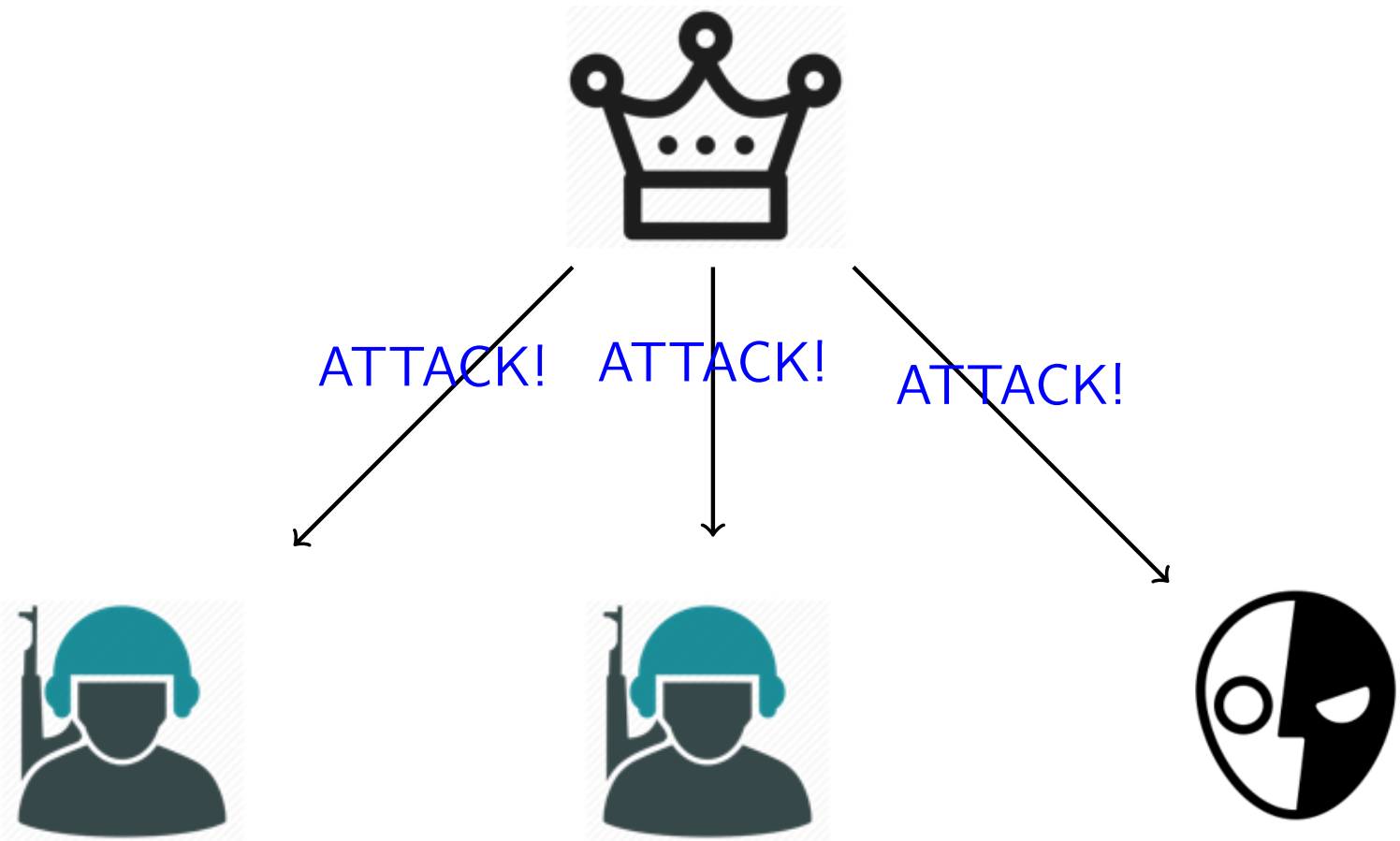
- If $m > 1$, then $m = 1$
- $m = 1$ can't be solved.
- (contradiction).

Oral Messages Fault!!!

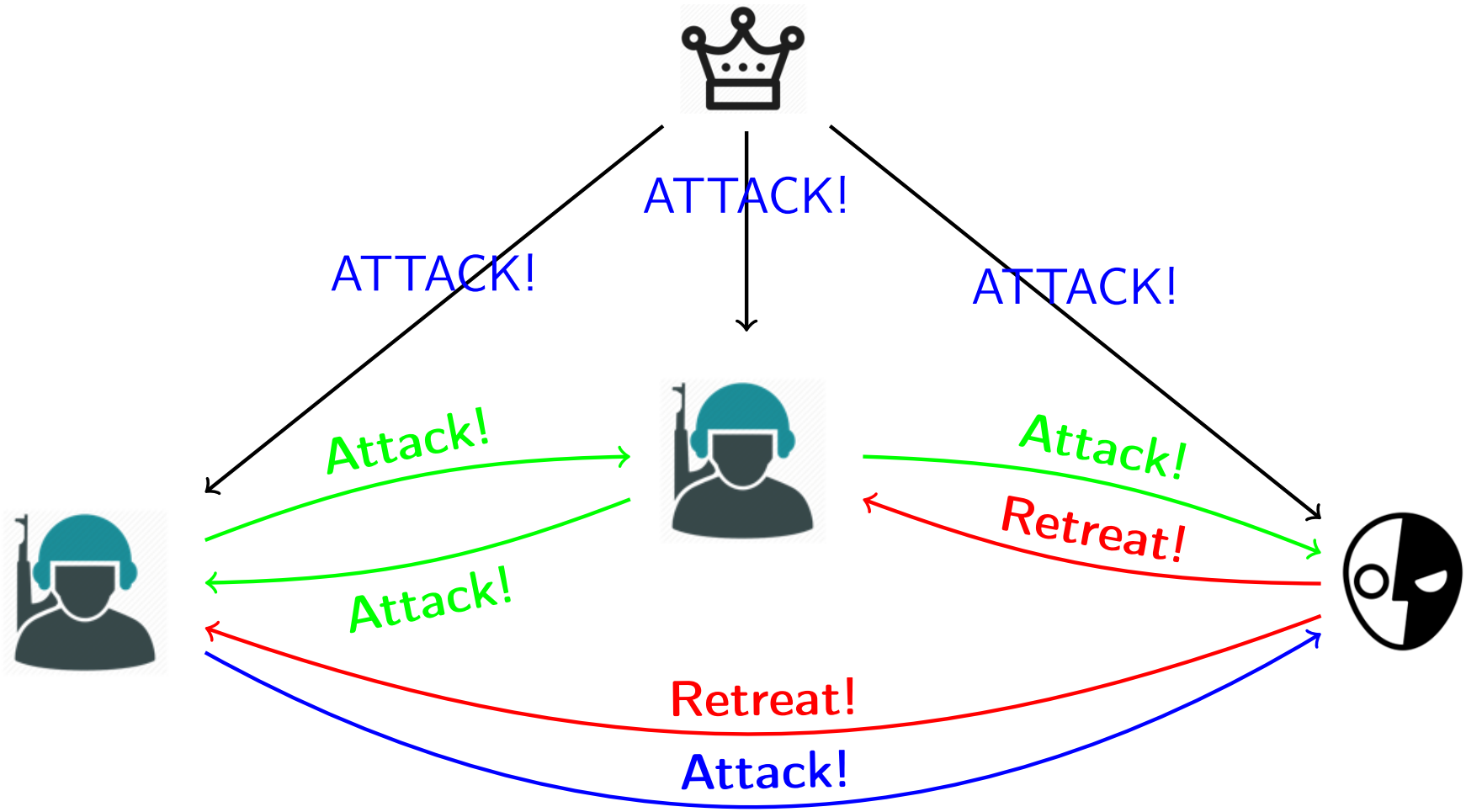
Solutions

So.. The Solution???

Solutions 1:Oral Messages->OM(1)



OM(1)-3*OM(0)


ATTACK!
ATTACK!
ATTACK!
Attack!
Attack!
Attack!
Retreat!
Retreat!
Attack!

Oral Messages

- Intuition: For every message M received, we solve a smaller BGP containing all but the current commander to tell others M has been received.
- OM(m) solvable for m traitors when 3m<n

Oral Messages

- Intuition: For every message M received, we solve a smaller BGP containing all but the current commander to tell others M has been received.
- **OM(m)** solvable for m traitors when 3m<n

Oral Messages

Formally...

- OM(k)
  - k==0
    - Commander sends value to everyone and everyone sends back the value they received
  - k>0
    - Commander sends value to everyone
    - Everyone starts a smaller BGP OM(K-1) where current Lieutenant becomes the new Commander
    - Everyone participated n-1 OM(k-1) and get n-1 values, return the majority

Complexity:(n-1)*MC(OM(k-1)) + n-1 = $O(n^m)$

Oral Messages

- OM(k)
  - k==0
    - Commander sends value to everyone and everyone sends back the value they received
  - k>0
    - Commander sends value to everyone
    - Everyone starts a smaller BGP OM(K-1) where current Lieutenant becomes the new Commander
    - Everyone participated n-1 OM(k-1) and get n-1 values, return the majority

Complexity:(n-1)*MC(OM(k-1)) + n-1 = $O(n^m)$

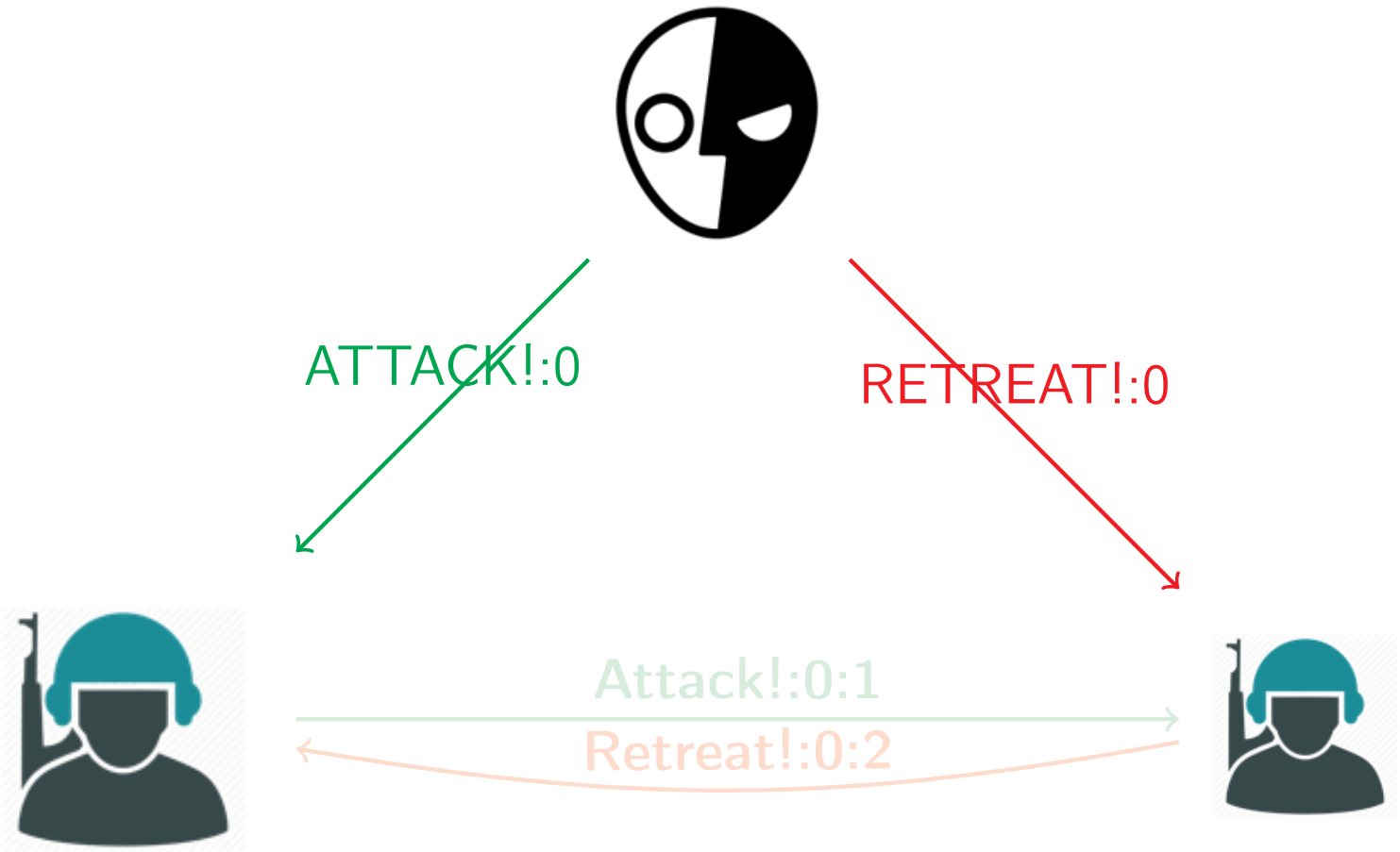
Oral Messages

Finally...

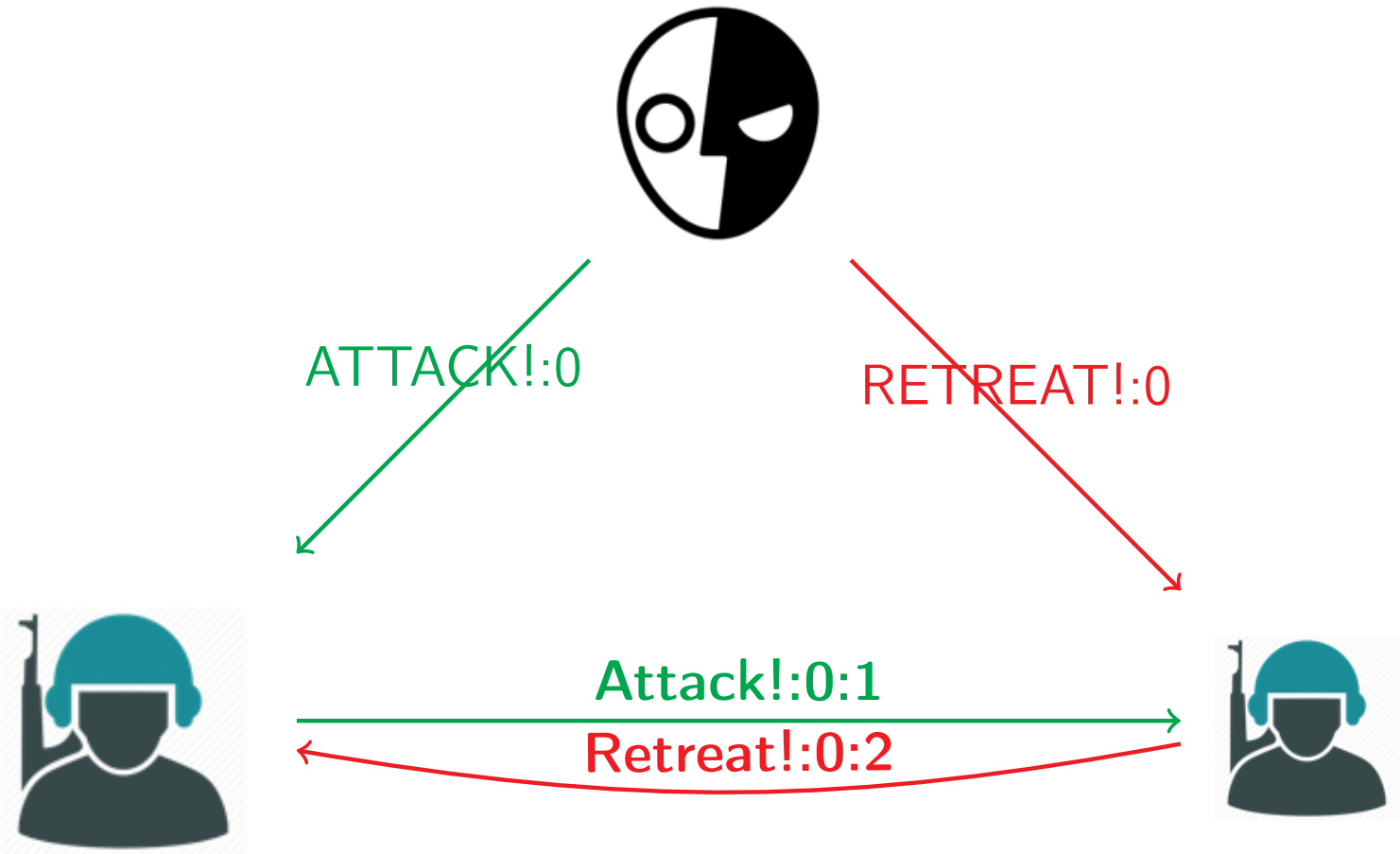
- OM(k)
  - k==0
    - Commander sends value to everyone and everyone sends back the value they received
  - k>0
    - Commander sends value to everyone
    - Everyone starts a smaller BGP OM(K-1) where current Lieutenant becomes the new Commander
    - Everyone participated n-1 OM(k-1) and get n-1 values, return the majority

Complexity: (n-1)*MC(OM(k-1)) + n-1 = $O(n^m)$

Solutions 2: Signed Messages

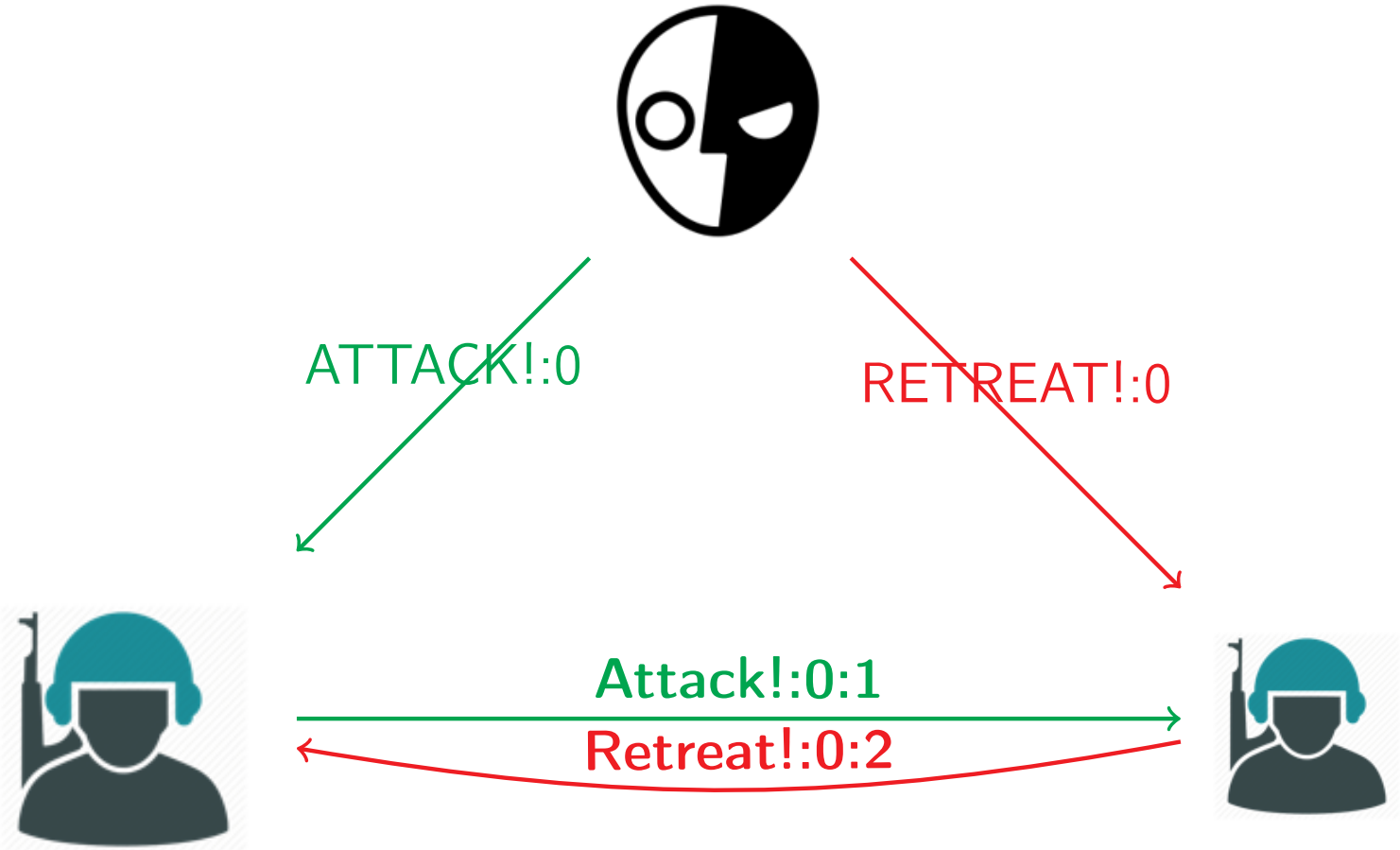


Solutions 2: Signed Messages



$V(1) == V(2)$

Solutions 2: Signed Messages



Choice(V(1))==Choice(V(2))

Minimum Number

Minimum number required for which an *f*-resilient consensus protocol exists

	synchrony	asynchrony	partial synchrony
fail-stop	f+1	inf	2f+1
crash	f+1	inf	2f+1 (Paxos)
byzantine with digital signature	f+1 (SM(f+1))	inf	
byzantine with authenticated channel	3f+1 (OM(f))	inf	

Minimum Number

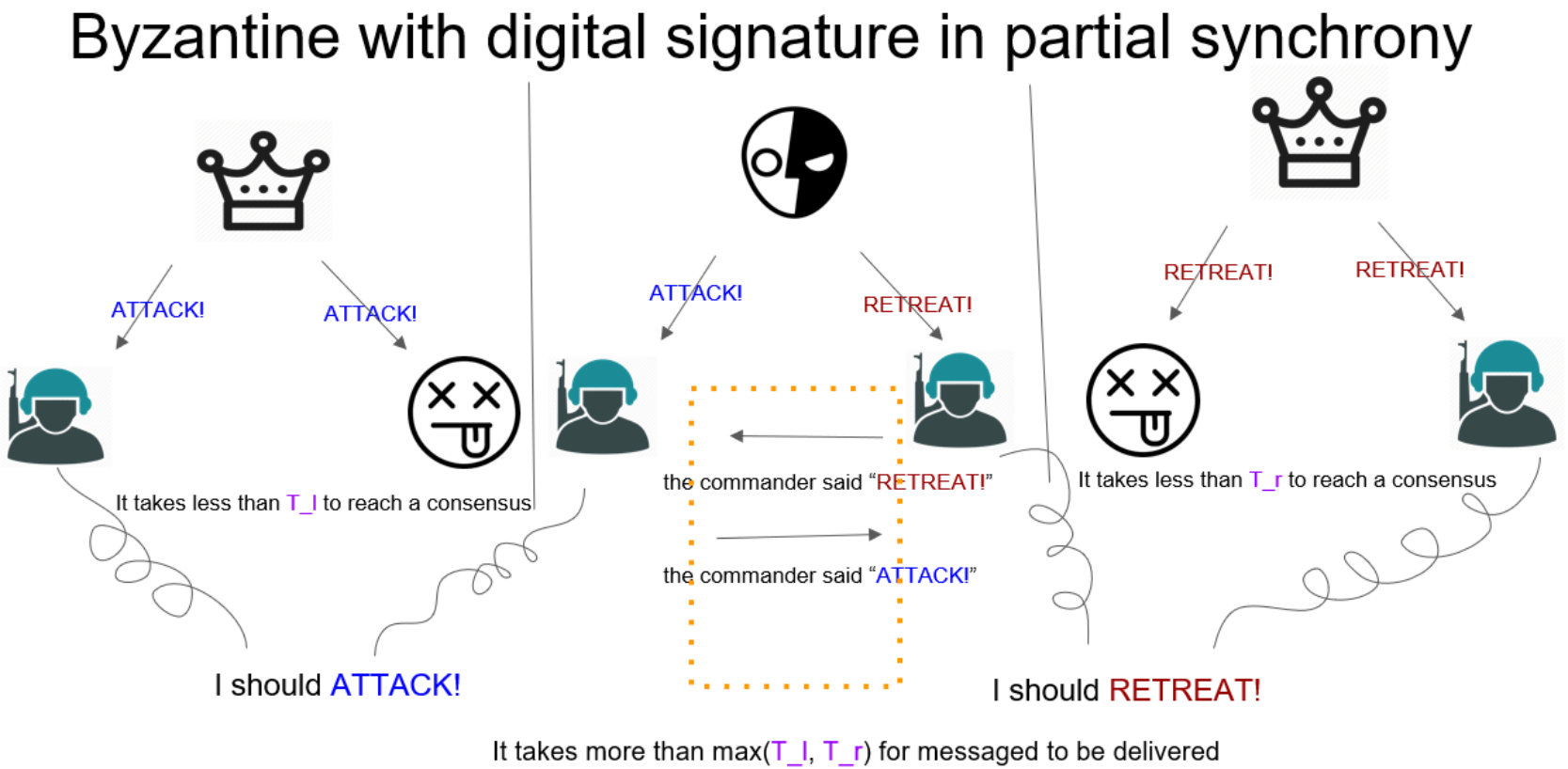
Minimum number required for which
an *f*-resilient consensus protocol exists

	synchrony	asynchrony	partial synchrony
fail-stop	f+1	inf	2f+1
crash	f+1	inf	2f+1 (Paxos)
byzantine with digital signature	f+1 (SM(f+1))	inf	???
byzantine with authenticated channel	3f+1 (OM(f))	inf	

Byzantine with digital signature in partial synchrony

1. synchronous 1/3 faults.
2. Sound familiar?
3. Assume there exist a protocol that can solve it.

Partial Synchrony



Practical Byzantine Fault Tolerance

- Commander sends the value to every lieutenant
- Every lieutenant
  - If it receives a new value v, broadcast `(prepare, v)`
  - If it receives $2f+1$ `(prepare, v)`, broadcast `(commit, v)`
  - If it receives $2f+1$ `(commit, v)`, broadcast `(committed, v)`
  - If it receives $f+1$ `(committed, v)`, broadcast `(committed, v)`
- Ensure agreement
- Ensure liveness under a loyal commander
- What if the commander is faulty?
  - we need view change

Practical Byzantine Fault Tolerance

- Commander sends the value to every lieutenant
- Every lieutenant
  - If it receives a new value v, broadcast `(prepare, v)`
  - If it receives $2f+1$ `(prepare, v)`, broadcast `(commit, v)`
  - If it receives $2f+1$ `(commit, v)`, broadcast `(committed, v)`
  - If it receives $f+1$ `(committed, v)`, broadcast `(committed, v)`
- Ensure agreement
- Ensure liveness under a loyal commander
- What if the commander is faulty?
  - we need view change

Minimum number required for which an f-resilient consensus protocol exists

Minimum number required for which an f-resilient consensus protocol exists

	synchrony	asynchrony	partial synchrony
fail-stop	f+1	inf	2f+1
crash	f+1	inf	2f+1 (Paxos)
byzantine with digital signature	f+1 (SM(f+1))	inf	3f+1(PBFT)
byzantine with authenticated channel	3f+1 (OM(f))	inf	

Application

Blockchain and Cryptocurrencies

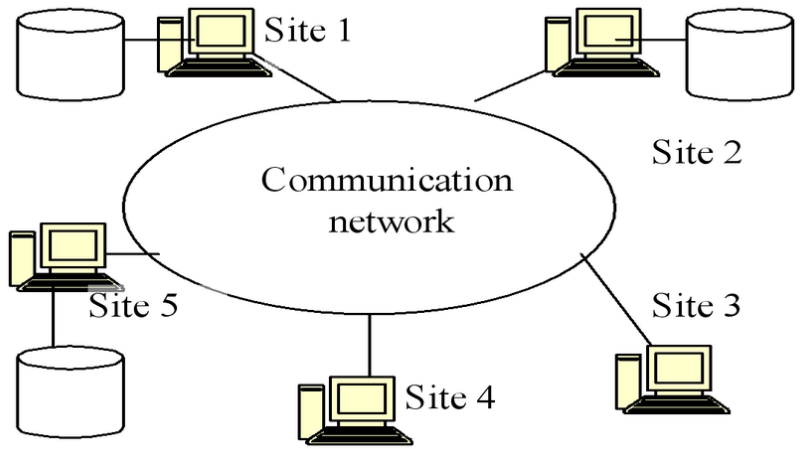
- **Problem Context:** In decentralized systems like Bitcoin, there are no central authorities to ensure all nodes agree on the state of the ledger.
- **Solution:** Byzantine Fault Tolerance (BFT) algorithms ensure consensus despite malicious or faulty nodes.

Blockchain and Cryptocurrencies

- **Problem Context:** In decentralized systems like Bitcoin, there are no central authorities to ensure all nodes agree on the state of the ledger.
- **Solution:** Byzantine Fault Tolerance (BFT) algorithms ensure consensus despite malicious or faulty nodes.

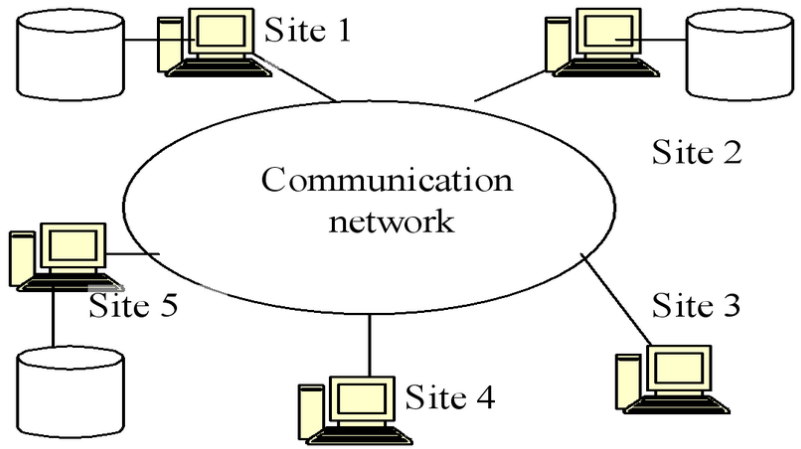
Distributed Databases

- Challenge is to ensure that all nodes have consistent data.
- Byzantine fault tolerance can help maintain data consistency.



Distributed Databases

- Challenge is to ensure that all nodes have consistent data.
- Byzantine fault tolerance can help maintain data consistency.



Conclusion

Key Takeaway

Byzantine Fault Tolerance (BFT) is the cornerstone of reliability in distributed systems. It enables:

- **Consensus:** Agreement among nodes, even with faulty or malicious participants.
- **Scalability:** Essential for decentralized systems like blockchain.
- **Resilience:** Smooth functioning in unpredictable environments.

"By leveraging BFT protocols, we build systems that stand strong amidst failures."

The End