# Byzantine Generals’ Problem

A situation where communication that requires consensus on a single strategy from all members within a group or party cannot be trusted or verified.

The Byzantine Generals' Problem is a thought experiment that deals with a key question of computer science: is it possible to form a consensus in a computer network composed of independent, geographically distributed nodes?

The problem was proposed in 1982 by researchers from the SRI International Research Institute.

It goes as follows: there are a number of Byzantine generals besieging a city. They can only communicate via sending messengers to each other. The generals must agree on a common plan of action: whether to attack the city or retreat. However, some of the generals are traitorous and actively working against the forming of a consensus; their number and identities are unknown.

The question posed by the problem is what decision-making algorithm the generals should use to devise a common plan — regardless of the traitors’ interference — and whether such an algorithm exists at all.

According to the researchers’ own analysis, such a system is indeed feasible, but the number of loyal generals must strictly exceed two-thirds. For example, in a situation with three generals, one of which is traitorous, the loyal ones can never guarantee that they will be able to reach a consensus.

This problem is highly relevant for **cryptocurrencies** as they are, in essence, distributed computer systems: they are composed of **transaction**-processing nodes that are independent of each other and any central authority and can only communicate remotely. They are the “generals” that need to reach a **consensus** about which transactions have taken place and when.

**Nodes** have the potential to supply faulty data about transactions either by choice or by accident, and their information must be sorted out. Bitcoin (**BTC**) and other cryptocurrencies solve this problem via technical solutions such as the **proof-of-work** and **proof-of-stake** algorithms.