

Abstract

This paper extends the failures detector approach from crash-stop failures to muteness failures. Muteness failures are malicious failures in which a process stops sending algorithm messages, but might continue to send other messages, e.g., "*I-am-alive*" messages. The paper presents both the specification of a muteness failure detector, denoted by $\Diamond M_{\mathcal{A}}$, and an implementation of $\Diamond M_{\mathcal{A}}$ in a partial synchrony model (there are bounds on message latency and clock skew, but these bounds are unknown and hold only after some point that is itself unknown). We show that, modulo a simple modification, a consensus algorithm that has been designed in a crash-stop model with $\Diamond S$, can be reused in the presence of muteness failures simply by replacing $\Diamond M_{\mathcal{A}}$ with $\Diamond S$.