

Forum: [Hack Forums](#)

Thread: [Threat Modeling for Developers Who Don't Want Another Giant Document](#)

watsica12 - *October 19, 2026, 2:28 am*

I've been trying to make threat modeling more practical by applying it to a simple web application with authentication, user profiles, and file uploads. Drawing a data-flow diagram helps identify where information moves and where trust boundaries exist. From there, frameworks such as STRIDE provide a structured way to think about spoofing, tampering, repudiation, information disclosure, denial of service, and privilege escalation.

The difficult part isn't producing a list of possible threats. It's deciding which risks matter most, identifying realistic mitigations, and ensuring the resulting tasks actually reach the engineering backlog. A threat model also needs to change when the architecture changes, otherwise it quickly becomes documentation describing an application that no longer exists.

I'm curious how other developers approach this in smaller projects where there isn't a dedicated security team. Do you use formal frameworks, lightweight checklists, architecture diagrams, or collaborative review sessions? What is the smallest threat-modeling exercise you've done that led to a meaningful security improvement?