Team

Vamsi Praveen Karanam (Amazon)

- Vamsi Praveen Karanam | LinkedIn

Pradeep Karanam (ex-Meta)

- Pradeep Karanam, PMP | LinkedIn

Kartheek Penagamuri (Microsoft)

- Kartheek Penagamuri | LinkedIn

Threat modeling

Threat modeling is a structured process for identifying, evaluating, and addressing potential security threats in a system's design or implementation.

It helps teams understand what can go wrong, prioritize risks based on impact and likelihood, and plan mitigations early in the development lifecycle.

Common frameworks include STRIDE, DFDs, and attack trees, which guide teams in uncovering vulnerabilities and improving security posture before deployment.

PyTM

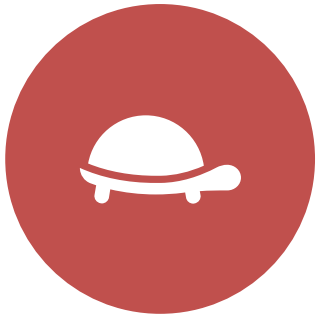
OWASP/pytm: A Pythonic framework for threat modeling

+

MCP Server

AI-powered threat modeling in minutes

Why it Matters



MANUAL THREAT MODELING IS SLOW, ERROR-PRONE, SILOED

SECURITY TEAMS STRUGGLE TO KEEP PACE WITH RAPID FEATURE DELIVERY

DEVELOPERS NEED INSTANT VISUAL FEEDBACK INSIDE THEIR IDE OR CHAT AGENT

The Solution

PyTM MCP Server = OWASP PyTM engine wrapped in the Model Context Protocol

Natural-language prompt → data-flow diagram & STRIDE threat analysis

Python: runs anywhere Python 3.8+ is available

Key Features & Benefits

Feature	Benefit
Easy to use	<1 min setup
Graphviz integration	Instant PNG/SVG diagrams for docs & PRs
MCP-compliant	Plug-and-play with any Copilot-style agent or LLM framework
Extensible Python codebase	Add custom security checks in a few lines

30-Second Workflow

- 1. Clone & run
  - git clone https://github.com/vamsipraveenk/pytm-mcpserver
  - pip install -r requirements.txt
  - Add the MCP server to VS Code
- 2. Prompt your coding agent
  - “Generate a data-flow diagram for a mobile app talking to a web server and DB.”
- 3. Get artifacts
  - Threat modeling diagram → PNG/SVG
  - STRIDE report for quick risk triage

Next Steps

Try different prompts & repos

Extend MCP Server Functionality

Add Evals