# libfuzzer

# libfuzzer

> *Catalogue généré le 2026-05-11*

## En une phrase

libFuzzer est un "fuzzer" intégré au compilateur LLVM qui bombarde une fonction C/C++ avec des milliers d'entrées aléatoires pour trouver les bugs et plantages cachés — c'est l'outil le plus simple pour démarrer.

## Quand l'utiliser

- Tester une fonction C ou C++ avec un setup minimal et rapide.
- Démarrer du fuzzing sans s'embêter avec des configurations complexes.
- Fuzzer un projet compilé avec Clang.
- Quand un seul cœur CPU suffit (pas de campagne géante).
- Prototyper avant de basculer plus tard sur AFL++ si besoin.

## Comment l'invoquer

- **Slash command** : `/libfuzzer`
- **Phrases déclencheurs (texte)** : "fuzz C/C++ with libFuzzer", "quick fuzzing setup", "LLVM fuzzer"
- **Auto-invocation** : Sur demande explicite

# Description détaillée

libFuzzer est le fuzzer "starter pack" pour le C et C++. Un fuzzer, c'est un robot qui essaie de casser ton programme en lui passant des entrées tordues, aléatoires, jamais prévues. Le but : voir s'il plante, accède à de la mémoire interdite ou se met dans un état bizarre. libFuzzer est fait par les équipes LLVM (les mêmes qui font le compilateur Clang), donc c'est super intégré au pipeline de compilation moderne.

Tu écris juste une petite fonction appelée `LLVMFuzzerTestOneInput` qui reçoit des octets aléatoires, tu compiles avec `-fsanitize=fuzzer` et c'est parti. C'est le plus rapide à mettre en route, mais il a deux limitations : il tourne sur un seul cœur CPU et il est en mode "maintenance" depuis fin 2022 (pas de nouvelles features).

Bonne nouvelle : les harnesses (les petits programmes d'entrée écrits pour libFuzzer) sont compatibles avec AFL++. Donc tu commences avec libFuzzer pour aller vite, et si tu veux scaler sur plusieurs cœurs plus tard, tu réutilises le même code avec AFL++. C'est un excellent premier pas avant les outils plus avancés.

# Pour aller plus loin

Pour les exemples concrets, options de configuration et patterns avancés, voir le SKILL.md original.

# Source

- **Plugin** : `trailofbits/testing-handbook-skills`
- **Nom interne** : `libfuzzer`
- **Fichier** : `/home/thymon/.claude/plugins/cache/trailofbits/testing-handbook-skills/1.0.1/skills/libfuzzer/SKILL.md`

---

Revision #2
Created 2026-05-11 21:19:48 UTC by thymon
Updated 2026-05-11 21:37:23 UTC by thymon