entry-point-analyzer

entry-point-analyzer

> *Catalogue généré le 2026-05-11*

En une phrase

Cartographie toutes les « portes d'entrée » d'un smart contract — c'est-à-dire les fonctions que n'importe qui peut appeler depuis l'extérieur pour modifier l'état du contrat — afin de savoir où concentrer un audit de sécurité.

Quand l'utiliser

- Au tout début d'un audit de smart contract (Solidity, Vyper, Solana, Move, TON, CosmWasm…), pour dresser la liste des points exposés.
- Pour comprendre quelles fonctions sont accessibles publiquement et lesquelles sont réservées à l'admin.
- Quand tu veux vérifier le « contrôle d'accès » — autrement dit, qui a le droit de faire quoi dans le contrat.
- Avant de chercher des vulnérabilités spécifiques : on commence toujours par savoir où l'attaquant peut frapper.

Comment l'invoquer

- **Slash command** : `/entry-point-analyzer` (ou `/entry-points`).
- **Phrases déclencheurs (texte)** : "find entry points", "external functions", "audit flows", "access control", "privileged operations".
- **Auto-invocation** : Sur demande explicite.

Description détaillée

Un smart contract, c'est un programme qui vit sur la blockchain. Il expose des fonctions, comme des boutons qu'on peut presser depuis l'extérieur. Certains boutons sont réservés à l'admin (par exemple : retirer de l'argent du contrat), d'autres sont accessibles à tout le monde (par exemple : transférer ses propres tokens). Le skill `entry-point-analyzer` dresse la liste exhaustive de ces boutons et indique qui a le droit de les actionner.

Il ne s'intéresse qu'aux fonctions qui changent l'état du contrat (celles qui peuvent déplacer de l'argent, modifier des balances, changer des permissions). Les fonctions purement « lecture seule » (`view`, `pure`) sont exclues parce qu'elles ne peuvent pas, à elles seules, faire perdre des fonds. Pour les contrats en Solidity, le skill peut s'appuyer sur Slither, un outil d'analyse statique très utilisé, qui produit automatiquement le tableau des entry points.

En sortie, tu obtiens un rapport Markdown structuré : la liste des fonctions exposées, leur niveau d'accès (public, admin, restreint à un rôle, accessible seulement par un autre contrat), et les « modifiers » de sécurité appliqués (les garde-fous comme `onlyOwner`). C'est la base de travail à partir de laquelle tout audit sérieux démarre.

Pour aller plus loin

Pour les détails techniques, exemples et patterns spécifiques, voir le SKILL.md original.

Source

- **Plugin** : `trailofbits/entry-point-analyzer`
- **Nom interne** : `entry-point-analyzer`
- **Fichier** : `/home/thymon/.claude/plugins/cache/trailofbits/entry-point-analyzer/1.0.0/skills/entry-point-analyzer/SKILL.md`

Revision #2
Created 2026-05-11 21:19:24 UTC by thymon
Updated 2026-05-11 21:37:02 UTC by thymon