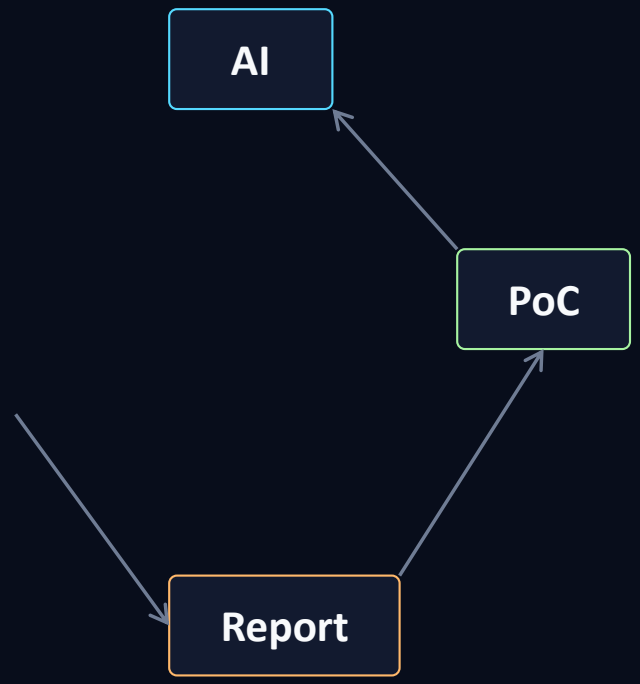
AI駆動バグハント入門

くままぬい



イベント紹介

セキュリティやハッキングについて話す
集会です！

初心者や何もわからない人から上級者まで
大歓迎です！

LTなども募集中です！

https://vrc-ta-hub.com/community/54/

雑なAI報告は、普通に迷惑になる

1 存在しない脆弱性の確認作業を
トリアージャーへ押しつける

2 正当な報告の処理が遅れる

3 報告者が技術的責任を負っていない

AI slopは、ハルシネーションだけの問題ではない

報告を生成するコスト

急減

LLMで報告らしい文章と仮説は
いくらでも作れる

報告が正しいか検証するコスト

あまり減らない

環境構築、再現確認、影響評価は
依然として人間の時間を使う

Bugcrowd 2026年3月：3週間でtriage queueが334%増加。

大部分は証拠の薄い低品質報告。
Bugcrowdはこの態度を「sloptimism」と呼んでいる。

slopは文章品質の問題ではなく、検証コストを他者へ移す問題。

Source: Bugcrowd, Apr. 23, 2026.

受理側はすでに詰まっている

curl / 2025年7月

20%

報告のうちAI slop

5%

genuine vulnerability

Bugcrowd / 2026年3月

大量の低確度報告が triage能力を圧迫

高頻度・低確度の報告が
legitimate submissionsの
検証時間を奪う。

curl側の観測は業界全体の統計ではないが、OSSメンテナー側の負担を示す具体例。

Source: Daniel Stenberg "Death by a thousand slops," Jul. 14, 2025; Bugcrowd, Mar. 10, 2026.

受理側もAIと制限で応答している

HackerOne

Hai Triageが
duplicate・out-of-scope候補を抽出

最終判断には人間が関与

Bugcrowd

submission farming → 永久BAN

10件連続invalid → 審査・30日停止

プログラム開始直後の自動投稿に対処

本人確認の強化

発見側がAIを使えば、受理側もAIとルールで応答する。

Source: HackerOne Hai Triage; Bugcrowd policy changes Mar. 10, 2026.

ただし、AIは本物の脆弱性も見つけている

Anthropic × Mozilla

Claude Opus 4.6がFirefoxに対し
2週間で22件を発見
14件がHigh

ベンダー自身の発表。

OpenAI Codex Security

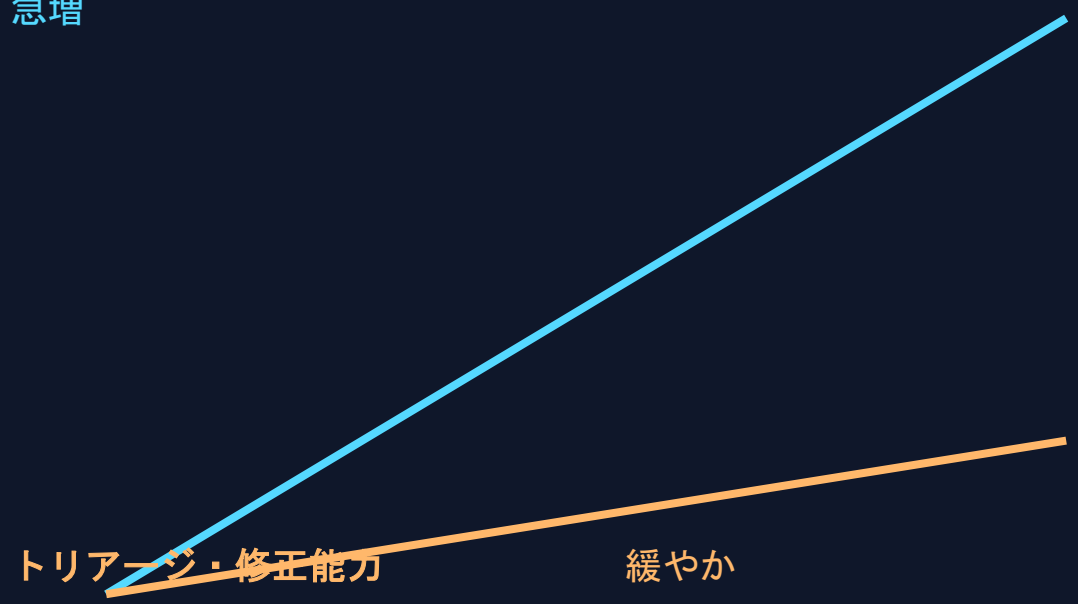
候補列挙だけでなく
sandbox環境でのvalidationと
PoC生成を重視する設計

ベンダー自身の発表。

偽陽性だけが増えたのではない。実在する脆弱性の発見速度も上がっている。

Source: Anthropic, Mar. 6, 2026; OpenAI Codex Security help, 2026.

発見側だけが速くなっている



ボトルネックが
発見から
検証・報告・修正へ移る。

curl側も、2026年には高品質な
脆弱性報告まで急増し、バックログが
さらに悪化すると予測。

Source: Daniel Stenberg "High quality chaos," Apr. 22, 2026.

事例

2025/11/30　Google Antigravity / Gemini 3 Pro

pre-RCEの
IPC境界を列挙
→
数十件の候補
→
5〜6件を
検証対象に
→
PoC成立
→
CVE

CVE-2026-8945

Firefox / Firefox Focus for Android
サンドボックスエスケープ / High
報奨金 $4,000
共同報告者2名、Firefox 151にて修正

当時のAntigravityは個人向け無償プレビュー。
大量の候補を試せたこと自体が成果の条件。

Source: Mozilla MFSA 2026-46; Google Antigravity, Nov. 18, 2025.

実態は、かなり丸投げしている

recon	**環境構築**	**デコンパイル**	**コード分析**
仮説生成	**PoC作成**	**インパクト拡張**	**レポート初稿**

これらを当時はGemini、現在は主にCodexへ任せている。

個別作業をAIへ任せ、人間は調査全体の進行を制御している。

人間に残ったのは、探索資源の配分

介入する場面

同じ場所を繰り返し調査している

成果のないコンポーネントへ時間を使い続けている

重要でない情報を掘り続けている

新しい証拠が増えなくなった

介入時に行うこと

現状の要約をAIに出させる

調査済み領域のマッピング

狙っているバグクラスの再確認

次に見るコンポーネントの再選定

進捗の客観的な確認

一番重いのは「探索上のslop」

認識上のslop

存在しないバグ

インパクトの飛躍

PoCで削れる

探索上のslop

同じ場所の周回

成果のない調査の継続

時間とトークンを
消費してから見える

伝達上のslop

長いレポート

不要な背景・過剰な添付

編集で削れる

まだ調査を続けるか

現在の調査

→

判断材料

primitiveはあるか

未確認の機能はあるか

まだ狩れそうか

インパクトを実証で拡張できるか

→

AIによる再評価

↓

人間が判断

継続 / 移動 / 終了

AIの再評価と人間の判断を併用する。

提出までに通すゲート

スコープ内 — 同一環境で複数回再現 — クリーンVMで再現 — 人間の操作でも再現 — 期待と実際の差を説明 — 影響が実証範囲内

中心基準はPoCが成功していること。

全コードの人間による再読や、AI全出力の逐語的な監査は提出条件にしない。

インパクトはPoCから書く

やりがちな流れ

静的解析で限定的なOOBを発見

↓

「RCEにつながる可能性がある」

↓

High / Critical

提出時の考え方

PoCで実際に確認したprimitive

↓

成立条件

↓

実証できた影響 → レポートへ記載

AIが追加インパクトを探索するのは止めない。最終レポートへ採用するかをPoCで決める。

AIレポートは、そのままだと長い

AIが入れがちなもの

- 長い背景説明
- 類似CVE
- 自明な説明
- Highと判断する理由の長文
- AI特有の定型句
- 各節の総括
- 同じ内容の言い換え
- 実証していない最大インパクト
- 全検証ログ
- 不要なPoC・補助ファイル
- 結論より前に大量の説明

文体修正ではなく、順序変更と削除が中心。

AI出力から、提出物へ

AIが生成した調査レポート

背景・類似例
全証拠・総括

→

編集

1. 結論を先頭へ移す
2. 背景・定型句を削る
3. 未実証の影響を修正
4. PoCと添付を最小化

→

提出版

概要

CVSS ※専用欄がない場合

対象

原因

再現手順

PoCファイルの説明

影響

修正案 ※必要な場合

証拠を捨てるのではなく、初回提出から外す

初回提出に入れる

最短の再現手順

最も明確なPoC

原因

実証済みの影響

手元に置く

他の条件での検証結果

追加ログ

反論への先回り

複数のE2E証拠

要求されたときに追加で出す。読み手の仕事を減らしつつ、反論への備えは保持する。

まとめ

AIは実在する脆弱性を見つけられる。

人間に残るのは、
探索資源の配分・提出判断・情報の圧縮。

AIの出力を増やすより、
送る情報を減らす方が重要である。

参考資料

- Bugcrowd, "Sloptimism is breaking any system built on human validation," Apr. 23, 2026
- Bugcrowd, "Bugcrowd policy changes to address AI slop submissions," Mar. 10, 2026
- Daniel Stenberg, "Death by a thousand slops," Jul. 14, 2025
- Daniel Stenberg, "High quality chaos," Apr. 22, 2026
- HackerOne, Hai Triage product page / press release, Jul. 22, 2025
- Anthropic, "Partnering with Mozilla to improve Firefox's security," Mar. 6, 2026
- OpenAI Help Center, "Codex Security," 2026
- Mozilla Foundation Security Advisory MFSA 2026-46, May 19, 2026
- Google Antigravity, public preview announcement, Nov. 18, 2025

報奨金額・候補数・運用実感は、発表者およびチームの実践記録に基づく。