Houston, We Have a (Quality) Problem

Why security audits need an upgrade

By: engn33r

Overview

1. Why audit quality matters
2. Audit quality variability
3. Solutions and implementation

Slides:

https://engn33r.com/ethbelgrade.pdf

Background

- Security at yAudit and yAcademy, auditing and teaching
  - P.S. we offer free security training!
- Formerly in web2 security, before that hardware

yAudit

yAcademy

engn33r.com

Why improving smart contract security matters

- Hacks hurt people, protocols, and the entire ecosystem
- Blockchain tech is open source, but no open source quality metrics for smart contract security exist
- As the ecosystem matures, security should improve (like the early internet)

The problem: audit quality varies

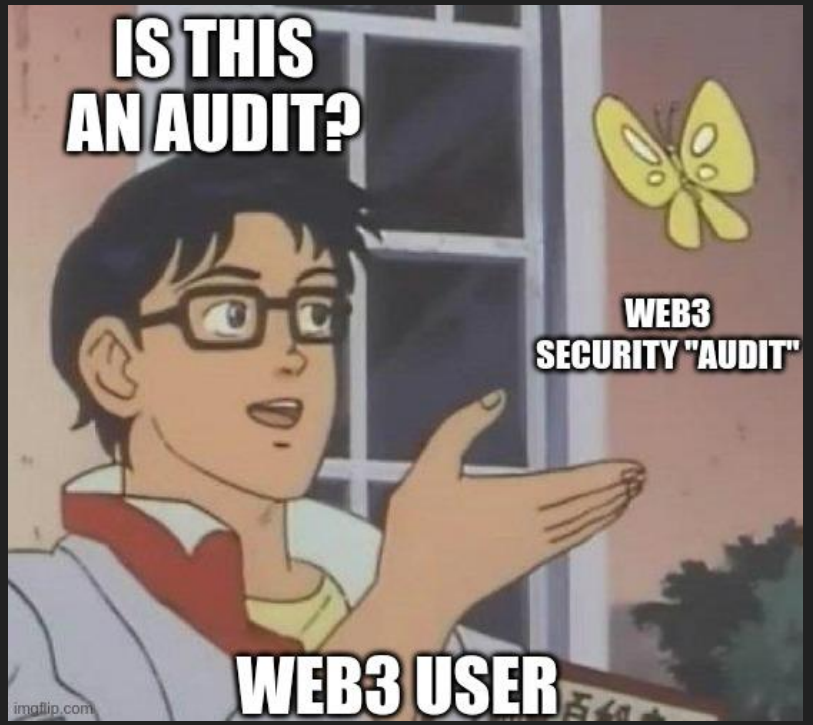
Key factors:

- Time spent with the code
- Skill level of auditors

Common gaps:

- Audited code =? deployed code (see Zellic's Audit Drift analysis)
- Audited code >=? on-chain values (not found in solidity files)
- Risk analysis !=? security audit (backtesting, robustness of weak DeFi protocol designs in variable market conditions, etc.)

2023 web3: the term “auditor” holds little weight



Web3 audit shopping in 2023

Related prior art

- Zellic created a dashboard to track the differences between on-chain code and the audited code. The dashboard is currently offline.
- DeFiSafety provides independent quality and ratings organization that rates DeFi products. DeFiSafety does not perform code audits.

Who is auditing the auditors?

Possible solutions

1. Portable ELO ratings for individual auditors from audit competitions
2. Semi-automated metrics for audit reports
3. Standardized security best practices & checklist
4. Post-audit customer reviews

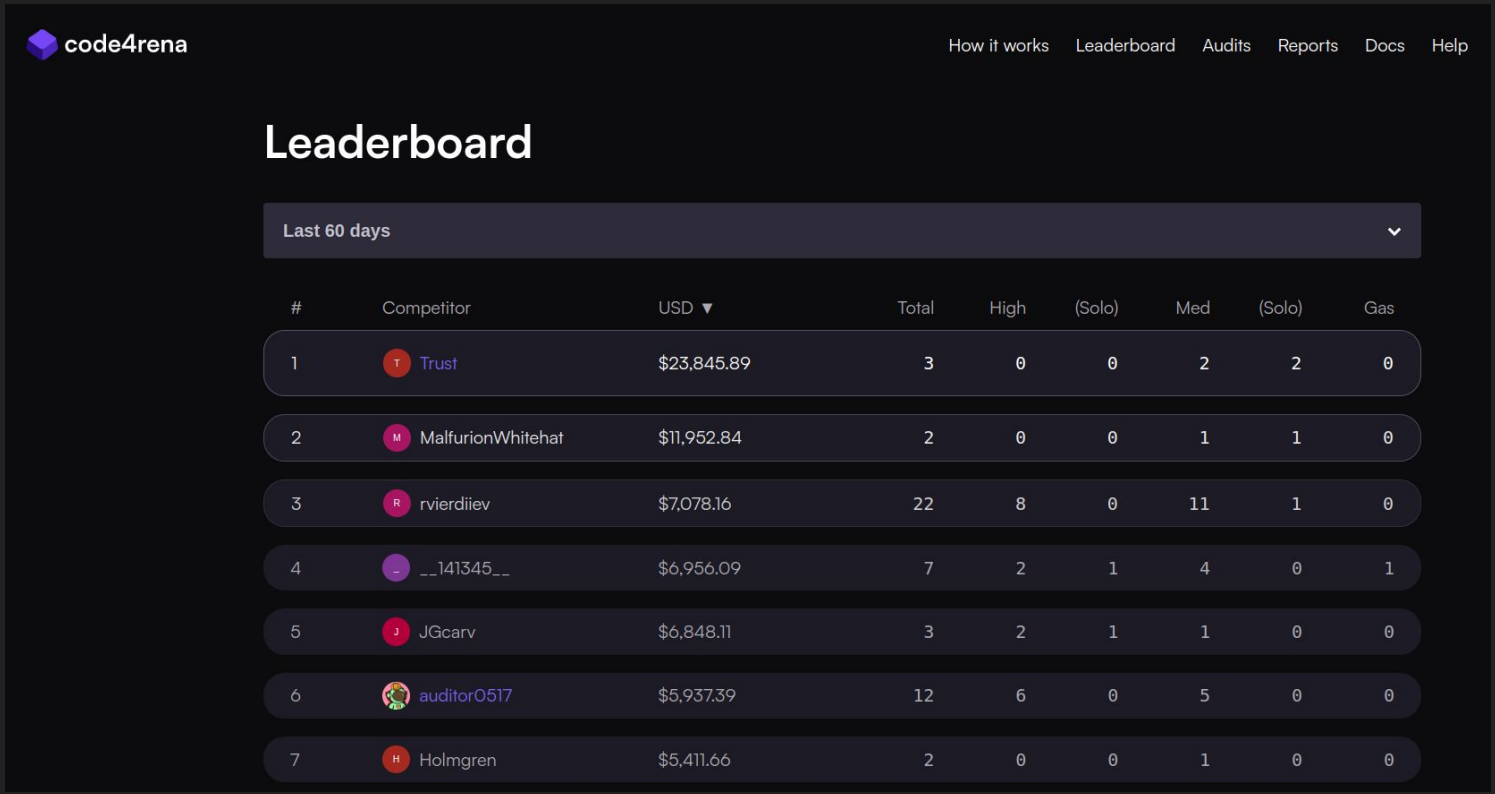
Solution #1: Portable auditor ratings

- We already have data for this from code4rena and Sherlock
- Don’t reinvent the wheel, use a known scoring system
  - Chess ELO analogy: 2200+ → master, 2500+ → grandmaster
- Audit firms have variability, so measure at the individual level
  - Different auditors for each engagement
  - Auditors move between companies

Elo rating system

- Normally for 1-on-1 matches, not for audit contests
- 3rd place != 3rd place. Strength of contest competitors should be factored in
- Glicko > Elo?

Contest data: Code4rena



#	Competitor	USD ▼	Total	High	(Solo)	Med	(Solo)	Gas
1	Trust	$23,845.89	3	0	0	2	2	0
2	MalfurionWhitehat	$11,952.84	2	0	0	1	1	0
3	rvierdiiev	$7,078.16	22	8	0	11	1	0
4	__141345__	$6,956.09	7	2	1	4	0	1
5	JGcarv	$6,848.11	3	2	1	1	0	0
6	auditor0517	$5,937.39	12	6	0	5	0	0
7	Holmgren	$5,411.66	2	0	0	1	0	0

Contest data: Sherlock

LEADERBOARD

#	Auditor	Points	Contest days	Payouts (USDC)
1	lllllll	**538**	79.0	190,840.22
2	0xRajeev	**514**	14.0	39,039.67
3	WATCHPUG	**402**	51.0	138,337.61
4	thec00n	**330**	35.0	81,151.3
5	hyh	**243**	147.0	246,839.08
6	roguereddwarf	**230**	28.0	18,869.94
7	xiaoming90	**225**	72.0	142,276.03

Solution #1: Implementation steps

- Adapt ELO scoring formula for competitions without 1-on-1 results
- Create and maintain a database of auditor ELO ratings, including historical data to show progress over time
- Optional: Code4rena and Sherlock help with this effort by making data easier to pull with an API

Solution #2: Metrics for audit reports (1 of 2)

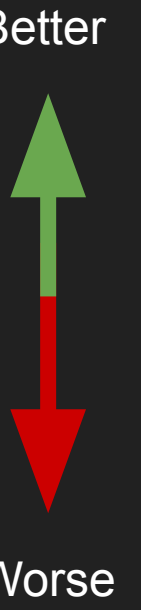
- Critical + high findings ÷ total findings = % serious findings
  - Low risk findings are more easily automated
- Misrated findings ÷ total findings = % of findings with incorrect risk
  - Audit firms are incentivized to inflate the severity of their findings
  - This requires substantial manual effort

Solution #2: Metrics for audit reports (2 of 2)

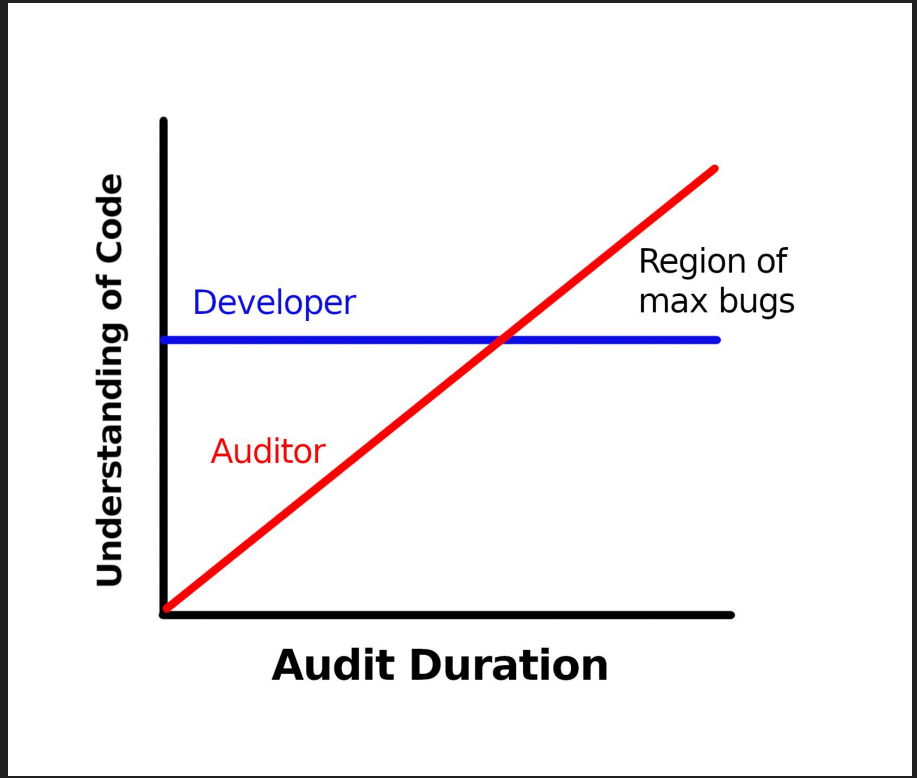
- Unique audit findings ÷ total audit findings = % unique findings
  - Measures the amount of manual effort for each audit
- Lines of code ÷ (auditors * days of auditing) = LOC per person day
  - Slower audit speed may indicate more attention to detail

Example metrics

	% Unique Findings	% Serious Findings	Misrated Findings	LOC per person day
Audit Firm 1	60	33	5	70
Audit Firm 2	45	26	12	180
Audit Firm 3	30	22	24	300
Audit Firm 4	10	14	37	900



Why longer audits matter



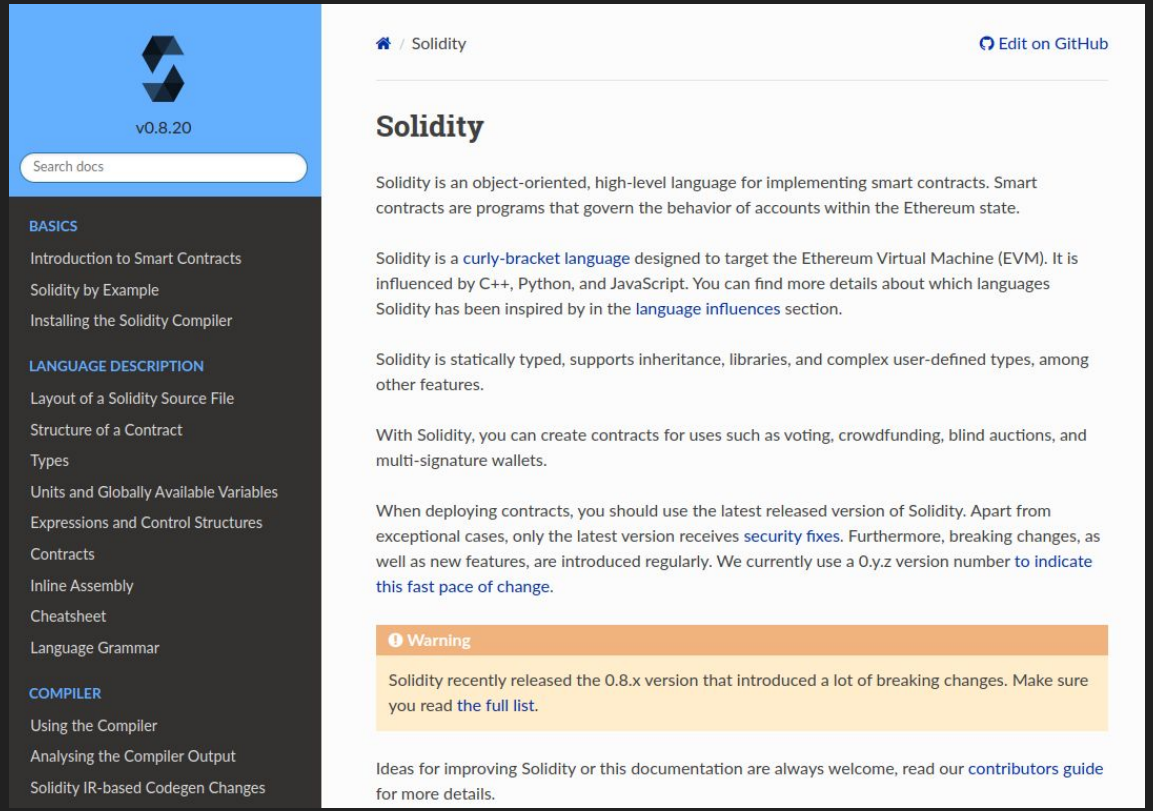
Solution #2: Implementation steps

- Create parsers for different audit reports (like Masamune)
- Enlist unbiased experts to comment on the accuracy of chosen metrics (i.e. security experts not working at audit firms)
- Optional: Organize a group of audit firms to maintain and enhance the collected metrics, potentially adding manual non-automated metrics
- Optional: audit firms contribute parsers for their own reports to enable transparent metrics on their results

Solution #3: Standard Best Practices & Checklist

- An open source list of common bugs can reduce repetitive mistakes
- Standardization across audit firms will improve consistency
- Information sharing across the industry helps everyone level-up

Information repository example



Solution #3: Implementation steps

- Start a repository of common security mistakes and best practices
  - https://github.com/YAcademy-Residents/security-checklist
- Recruit other experts to contribute to the knowledge repository

Solution #4: Post-audit customer reviews

- We need restaurant reviews for audits
  - Sharing information about the audit process is valuable
- Possible categories for feedback:
  - Was the cost fair?
  - Were the auditors communicative?
  - Did the audit miss some findings?
  - Did the final report meet expectation?

Customer reviews

Belgrade Fortress

Београдска тврђава

4.7 ★★★★½ (39,819)

Fortress

★★★★★ 8 months ago

Beautiful fortress and park that adds to the rich history of Belgrade. Perfect place for a stroll or picnic . Lots of view points overlooking the city across the river.

★☆☆☆☆ 2 years ago

OK. It deserves 5 stars, but I was a bit nervous when Google asked me for my opinion, so I put one to vent some frustration. Google just doesn't know when to ask.

Solution #4: Implementation steps

- Create a website to submit and view reviews
- Develop sybil resistance or submission verification for review submission
  - Only 1 review per audit, reviewer must be the correct protocol
- Optional: allow reviews to be edited if later audits catch bugs that the first one missed, giving a new perspective on the earlier audit

Why does any of this matter?

- What gets measured, gets managed
  - Auditors and audit firms will improve quality
- Quality metrics will improve decision making
  - Developers will make informed decisions for audits, enabling fairer pricing based on the targeted security goals
- Improving quality of security reduces hacks

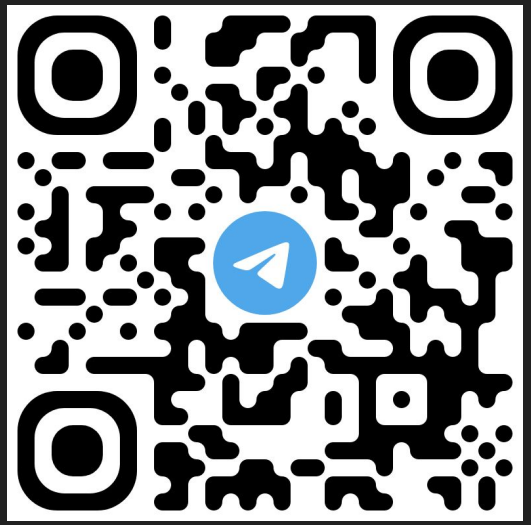
Summary

1. Audit quality is variable
2. Quality metrics possible at auditor and report level
3. Better data improves decision making for everyone
4. Automated metrics that can keep pace with this space are possible with some coordination

Join & participate

The question is not "can this be built?" but "should this be built?"

TG Coordination Group



Slides: https://engn33r.com/ethbelgrade.pdf