Financial Impact of Ethereum Vulnerability Detectors

Ing. Andrey Bortnikov, Faculty of Information Technology, Czech Technical University in Prague | Supervisor: Ing. Josef Gattermayer, Ph.D.

Motivation and Objectives

Smart contract exploits in decentralized finance continue to cause substantial losses, making early, automated detection of risky patterns critical to user protection and trust. This work examines how effective **Ethereum** vulnerability detectors are within the development lifecycle and estimates their financial impact when applied at scale. The thesis studies the **Wake** framework's detectors, constructs a real-world mainnet dataset, evaluates detector findings against value at risk, and designs an AI-assisted semantic detector that flags mismatches between documentation and implementation. The core objective is to quantify the degree to which static analysis can meaningfully reduce potential losses and reputational risks.

Methods and Dataset

The analysis leverages **Wake**, a static analysis toolkit for **Solidity** that provides AST/IR introspection, control- and data-flow utilities, and static analysis tools. The evaluation focuses on impactful classes such as **reentrancy**, **tx.origin** misuse, **unsafe delegatecall**, and **unchecked ERC-20 return values**. A mainnet dataset of recent, active contracts was compiled by filtering the **DexScreener** page for **Ethereum** (using trending and liquidity thresholds) and retrieving verified sources from Etherscan; token value was approximated via **DexScreener** for a practical **total value locked (TVL)** metric. On top of existing detectors, a new AI-based documentation-diff check compares NatSpec documentation to actual logic to identify semantic inconsistencies that could mislead users, auditors, or integrators.

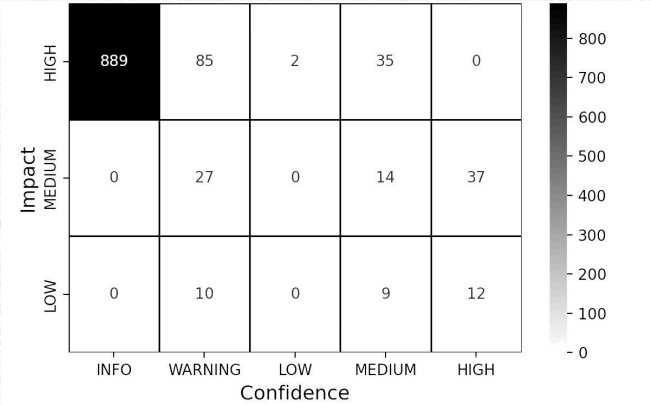
Model and Evaluation

To align technical findings with real-world stakes, the study adopts a value-at-risk metric that sums base-token **TVL** associated with high-impact findings while considering detector confidence levels. One hundred contracts were analyzed in automated batches, with normalized outputs to support aggregation and comparison. While no formal precision/recall benchmarking is reported, the methodology emphasizes severity and confidence, prioritizing clear, high-impact categories over low-signal patterns. The result is a pragmatic view of potential exposure if high-severity weaknesses reach production unaddressed.

Key results

- **1,120** total findings across severities.
- **32** high-impact reentrancy instances.
- Addressing flagged issues corresponds to an estimated potential exploit cost of **$2,701,486.78**.

"Classic" pitfalls remain dominant drivers of risk and warrant prioritized mitigation.



AI Detector (documentation-diff)

The documentation-diff detector uses an **LLM** to identify semantic mismatches between documentation and implementation, such as missing access-control details, incomplete trading rules, or ambiguous economic logic. While categorized at **WARNING** impact with **MEDIUM** confidence, correcting these gaps is associated with an estimated **$1,823,454.58** of protected value by reducing reputational and operational risk. Representative cases include R0AR TOKEN **$1,215,257.45**, TORN **$298,700.15**, and Meme Index **$62,796.43**, illustrating how clearer, truthful documentation can prevent misaligned expectations and integration errors.

Contract	Type	TVL
R0AR TOKEN	documentation-diff	$1,215,257.45
RandomDEX	documentation-diff	$8,300.00
MyStandard	documentation-diff	$45,500.00
E280	documentation-diff	$12,354.78
TWGToken	documentation-diff	$25,564.30
TORN	documentation-diff	$298,700.15
CATERC20	documentation-diff	$8,034.76
Meme Index	documentation-diff	$62,796.43
LegalXToken	documentation-diff	$41,481.91
DecentralizedEURO	documentation-diff	$105,464.80

Application, Limits and Impact

For developers and projects, integrating these checks into **IDEs** and **CI/CD** pipelines offers inexpensive, early feedback that can gate risky releases and focus fixes on high-value exposures. Auditors can use value-based summaries to focus on changes with the highest financial impact. There are limits: the TVL-based metric doesn't model exploit likelihood, static analysis can produce false positives, and the dataset leans toward recent, trending pairs. Despite these constraints, the results suggest that a combined static and semantic approach can measurably reduce both the probability and consequence of defects reaching production.

Conclusions and Next Steps

The results show that common detectors, especially when combined with documentation-aware semantic checks, can prevent multi-million-dollar losses and strengthen user trust. Next, we should build labeled datasets to measure precision and recall, widen detector coverage to more vulnerability classes and chains, and improve the economic model to include likelihood, composability effects, and on-chain signals for calibration. The trajectory points to practical gains from integrating static analysis and LLM-based semantics into everyday smart-contract engineering.