Security Tools

Visualization

- Sūrya [https://github.com/ConsenSys/surya] - Utility tool for smart contract systems, offering a number of visual outputs and information about the contracts' structure. Also supports querying the function call graph.
- Solgraph [https://github.com/raineorshine/solgraph] - Generates a DOT graph that visualizes function control flow of a Solidity contract and highlights potential security vulnerabilities.
- EVM Lab [https://github.com/ethereum/evmlab] - Rich tool package to interact with the EVM. Includes a VM, Etherchain API, and a trace-viewer.
- ethereum-graph-debugger [https://github.com/fergarrui/ethereum-graph-debugger] - A graphical EVM debugger. Displays the entire program control flow graph.

Static and Dynamic Analysis

- Mythril Classic [https://github.com/ConsenSys/mythril-classic] - Open-source security analyzer for Solidity code and on-chain smart contracts.

- Mythril Platform [https://mythril.ai/] - SaaS platform that allows anyone to build purpose-built security tools.
- Slither [https://github.com/trailofbits/slither] - Static analysis framework with detectors for many common Solidity issues. It has taint and value tracking capabilities and is written in Python.
- Echidna [https://github.com/trailofbits/echidna] - The only available fuzzer for Ethereum software. Uses property testing to generate malicious inputs that break smart contracts.
- Manticore [https://github.com/trailofbits/manticore] - Dynamic binary analysis tool with EVM support [https://asciinema.org/a/haJU2cl0R0Q3jB9wd733LVosL].
- Oyente [https://github.com/melonproject/oyente] - Analyze Ethereum code to find common vulnerabilities, based on this paper [http://www.comp.nus.edu.sg/~loiluu/papers/oyente.pdf].
- Securify [https://securify.chainsecurity.com/] - Fully automated online static analyzer for smart contracts, providing a security report based on vulnerability patterns.
- SmartCheck [https://tool.smartdec.net] - Static analysis of Solidity source code for security vulnerabilities and best practices.

Weakness OSSClassifcation & Test Cases

- SWC-registry [https://github.com/SmartContractSecurity/SWC-registry/] - SWC definitions and a large repository of crafted and real-world samples of vulnerable smart contracts.
- SWC Pages [https://smartcontractsecurity.github.io/SWC-registry/] - The SWC-registry repo published on Github Pages

Test Coverage

- solidity-coverage [https://github.com/sc-forks/solidity-coverage] - Code coverage for Solidity testing.

Linters

Linters improve code quality by enforcing rules for style and composition, making code easier to read and review.

- Solcheck [https://github.com/federicobond/solcheck] - A linter for Solidity code written in JS and heavily inspired by eslint.
- Solint [https://github.com/weifund/solint] - Solidity linting that helps you enforce consistent conventions and avoid errors in your Solidity smart-contracts.
- Solium [https://github.com/duaraghav8/Solium] - Yet another Solidity linting.
- Solhint [https://github.com/protofire/solhint] - A linter for Solidity that provides both Security and Style Guide validations.