Adding new Detectors to Slither

SLITHER

Advanced Topics in Software Engineering(CSE 6324)Fall 2022
Team 4
Final Presentation – 11/29/2022

- Atul Upadhye
- Kundana Vaka
- Yamini Dhulipalla
- Srikar Sai Yarlagadda
- Vigneshwar Selvaraj

GitHub link: https://github.com/AtulUpadhye17/CSE-6324-Team4

Plan:

Iteration 4

- Ordering Detector : Merged all the ordering related detectors into one.(Function, Contract, Library etc.).
- Implemented and integrate the Incorrect ordering detector.
- Optimized the search criteria for the “ Imports on the Top” and "Pragmas on the Top" detectors.
- Test all the implemented detectors and check for any bugs and fix them.
- Created the documentation for the project.

Incorrect Ordering

- Description: Check order of elements in file and inside each contract, according to the style guide [7]
- Incorrect constructor order
- State variable declaration after function
- Library after contract
- Interface after library

Detector Code (Constructor Order)

```
18      def _detect(self):
19          results = []
20          elements_list = []
21          for n in self.slither.crytic_compile.filenames:
22              rootFileName = n.absolute
23              with open(rootFileName, 'r') as f:
24                  for line in f:
25                      elements_list.append(line)
26
27              # check if the order of the constructor is correct or not
28              for contract in self.slither.contracts_derived:
29
30                  list_of_methods=contract.functions
31
32                  for x in range(len(list_of_methods)):
33                      #Check if constructor is present and placed before the functions
34                      if str(list_of_methods[x]).lower == "constructor" and list_of_methods[x]!=list_of_methods[0]:
35
36                          #info to be printed
37                          info = ['Incorrect constructor Order found in ',contract,"\n"]
38
39                          res = self.generate_result(info)
40
41                          results.append(res)
42
```

Detector Code(Function Order)

```
43            # check if the order of elements is correct or not
44            for contract in self.slither.contracts_derived:
45                for function in contract.functions:
46                    for element in elements_list:
47                        if function.name in element:
48                            if element.index(function.name) > elements_list.index(element):
49                                info = ["The function {} is declared before the variable declaration.".format(function.name),"\n"]
50                                res = self.generate_result(info)
51                                results.append(res)
52
53            # check if the library is declared before the contract or not
54            # if the library is declared after the contract, print the line number and the error message.
55            library_index = 0
56            contract_index = 0
57
58            for i in range(len(elements_list)):
59                if elements_list[i].startswith("contract"):
60                    contract_index = i
61                if elements_list[i].startswith("library"):
62                    library_index = i
63            if library_index > contract_index:
64                info = [f"The library is declared after the contract.","\n"]
65                res = self.generate_result(info)
66                results.append(res)
```

Detector Code(Library & Interface Order)

```
68              # check if the interface is declared after the library or not
69              # if the interface is declared before the library, print the line number and the error message.
70
71              interface_index = 0
72              library_index = 0
73
74              for i in range(len(elements_list)):
75                  if elements_list[i].startswith("library"):
76                      library_index = i
77                  if elements_list[i].startswith("interface"):
78                      interface_index = i
79
80              if interface_index < library_index:
81                  info = ["The interface is declared before the library.","\n"]
82                  res = self.generate_result(info)
83                  results.append(res)
84          return results
```

Contract Samples

```
1  // SPDX-License-Identifier: MIT
2
3  contract MyContract {
4      function hello() public{
5
6      }
7      //pragma
8      constructor(){
9      }
10
11
12 }
13
14 library MyLibrary {}
15
16 interface MyInterface {}
17
18 pragma solidity ^0.8.16;
```

Example 1 with pragma on bottom & pragma word commented

incorrect_example.sol

```
1  // SPDX-License-Identifier: MIT
2  pragma solidity ^0.8.16;
3  contract MyContract {
4      function hello() public{
5
6      }
7      //pragma
8      constructor(){
9      }
10
11
12 }
13
14 library MyLibrary {}
15
16 interface MyInterface {}
```

Example 2 with pragma on top & pragma word commented

Output : Example 1

```
(slither-dev) [11/27/22]seed@VM:~/ordering$ slither incorrect_example.sol

solc-0.8.17 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity

Incorrect constructor Order found in MyContract (incorrect_example.sol#4-13)
Reference: https://github.com/trailofbits/slither/wiki/Adding-a-new-detector

The function hello is declared before the variable declaration.
The library is declared after the contract.
Reference: https://github.com/trailofbits/slither/wiki/Adding-a-new-detector

pragma statement should be on top : /home/seed/ordering/incorrect_example.sol
Reference: https://github.com/trailofbits/slither/wiki/Adding-a-new-detector
incorrect_example.sol analyzed (3 contracts with 86 detectors), 5 result(s) found
[2]+  Done                    gedit incorrect_example.sol
(slither-dev) [11/27/22]seed@VM:~/ordering$
```

Output: Example 2

```
(slither-dev) [11/27/22]seed@VM:~/ordering$ slither incorrect_example.sol

solc-0.8.17 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-versions-of-solidity

Incorrect constructor Order found in MyContract (incorrect_example.sol#3-12)
Reference: https://github.com/trailofbits/slither/wiki/Adding-a-new-detector

The function hello is declared before the variable declaration.
The library is declared after the contract.
Reference: https://github.com/trailofbits/slither/wiki/Adding-a-new-detector
incorrect_example.sol analyzed (3 contracts with 86 detectors), 4 result(s) found
(slither-dev) [11/27/22]seed@VM:~/ordering$
```

Integration

- Integrated Detectors :

```
89 from .constructor.incorrect_constructor_name import IncorrectConstructorName
90 from .constructor.incorrect_constructor_order import IncorrectConstructorOrder
91 from .style.imports_on_top import ImportsOnTop
92 from .style.pragmas_on_top import PragmasOnTop
93 from .ordering.incorrect_ordering import IncorrectOrdering
```

Risks

- Complications while integrating all the ordering-related detectors (Function, contract, library) into a single detector. It was time-consuming to fix the errors in this process.
- Finding appropriate sample contracts for testing the detectors we have developed and integrated into slither was difficult.

Customers and Users

- Any security audit firm, smart contract developer, security expert, or academic researcher can use this tool with our new detectors added to Slither to make the smart contract auditing process more efficient.
- Feedback:
  - Shovon shared a few sample contracts with us and asked us to test them and verify if new detectors integrated into Slither are able to detect issues.

References

- [1] Overview. (n.d.). Retrieved October 15, 2022, from https://blog.trailofbits.com/2019/05/27/slither-the-leading-static-analyzer-for-smart-contracts/
- [2] GitHub - crytic/slither: Developer installation. (n.d.). GitHub. Retrieved October 15, 2022, from https://github.com/crytic/slither/wiki/Developer-installation
- [3] Rule Index of Solhint. (n.d.). Solhint. Retrieved October 17, 2022, from https://protofire.github.io/solhint/docs/rules.html
- [4] User Guide — Solium 1.0.0 documentation. (n.d.). https://ethlint.readthedocs.io/en/latest/user-guide.html
- [5] Adding a new detector · crytic/slither Wiki. (n.d.-b). GitHub. https://github.com/crytic/slither/wiki/Adding-a-new-detector
- [6] imports-on-top Solhint. (n.d.). Solhint. https://protofire.github.io/solhint/docs/rules/order/imports-on-top.html
- [7] Style Guide — Solidity 0.8.17 documentation. (n.d.). https://docs.soliditylang.org/en/v0.8.17/style-guide.html

Thank You