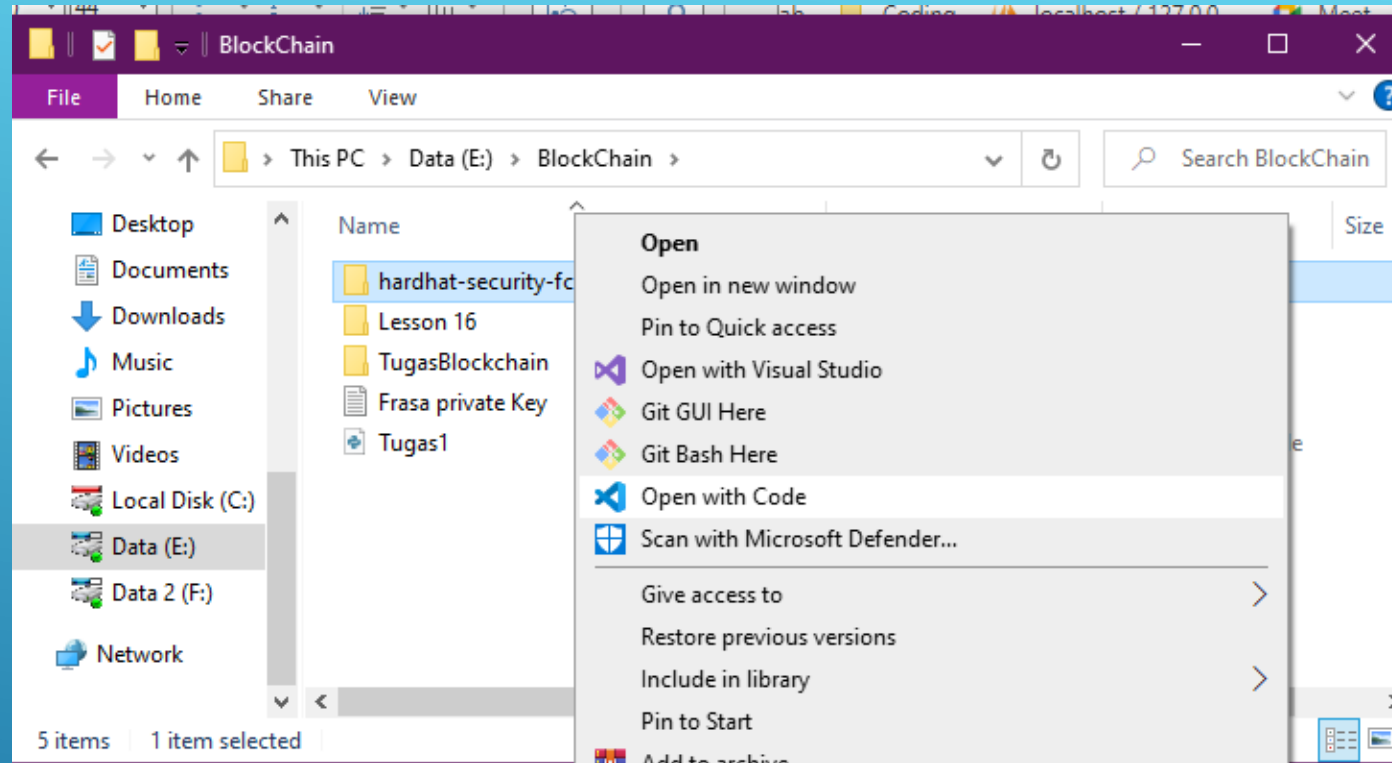


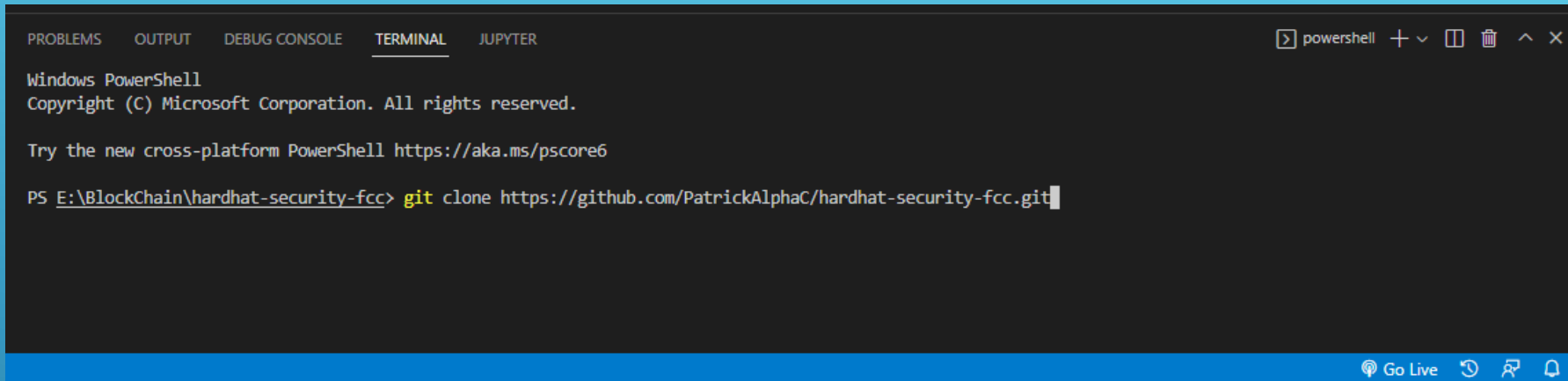
LESSON 18: SECURITY & AUDIT

Slither

Several thin, parallel white lines are drawn diagonally across the right side of the slide, starting from the top right and extending towards the bottom left.



BUAT FOLDER HARDHAT-SECURITY-FCC
DAN MASUK KE TERMINAL DALAM FOLDER
HARDHAT-SECURITY-FCC



The image shows a screenshot of a Visual Studio Code terminal window. The terminal is titled 'powershell' and shows the following text: 'Windows PowerShell', 'Copyright (C) Microsoft Corporation. All rights reserved.', 'Try the new cross-platform PowerShell https://aka.ms/pscore6', and the command 'PS E:\Blockchain\hardhat-security-fcc> git clone https://github.com/PatrickAlphaC/hardhat-security-fcc.git'. The terminal is part of a larger interface with tabs for 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL', and 'JUPYTER'. The bottom status bar shows 'Go Live' and other icons.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

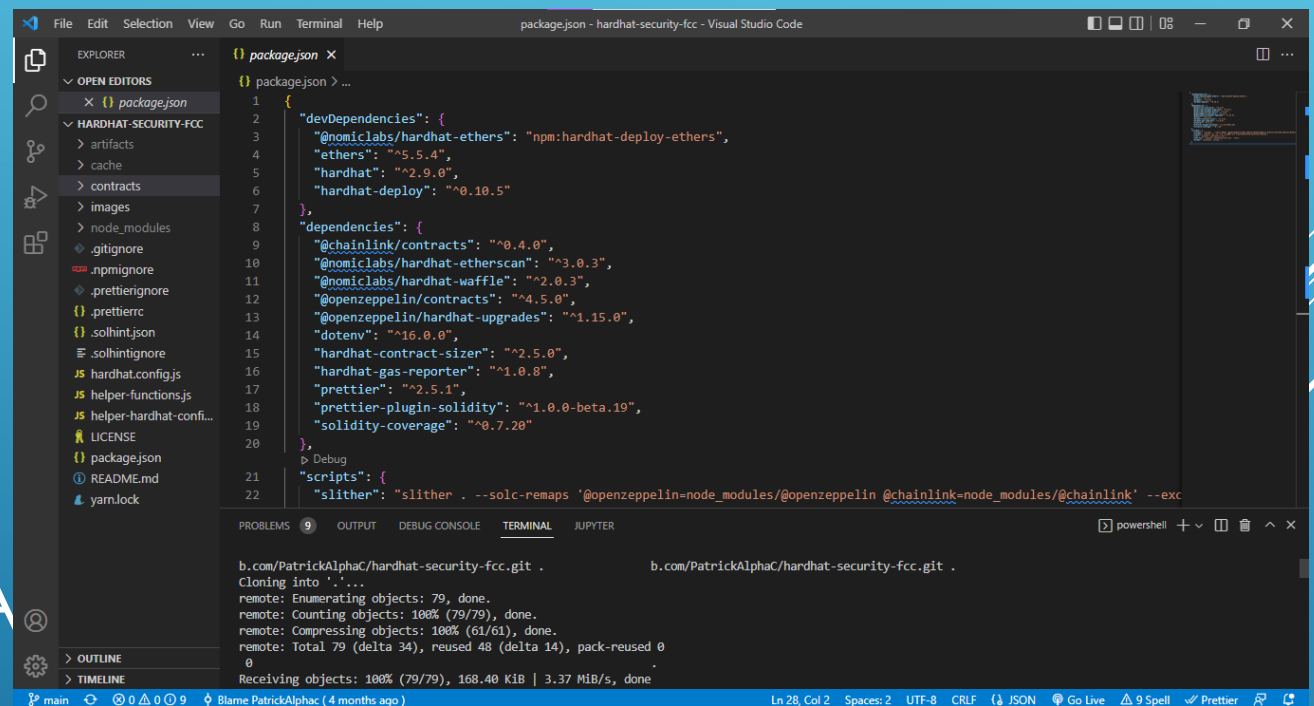
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS E:\Blockchain\hardhat-security-fcc> git clone https://github.com/PatrickAlphaC/hardhat-security-fcc.git
```

MASUK KE VSCODE DAN CLONE CODE DI
GIT
[HTTPS://GITHUB.COM/PATRICKALPHAC/HARDHAT-SECURITY-FCC.GIT](https://github.com/PatrickAlphaC/hardhat-security-fcc.git) SEPERTI YANG
ADA DI VIDEO

```
Cloning into '.'...
remote: Enumerating objects: 79, done.
remote: Counting objects: 100% (79/79), done.
remote: Compressing objects: 100% (61/61), done.
remote: Total 79 (delta 34), reused 48 (delta 14), pack-reused 0
Receiving objects: 100% (79/79), 168.40 KiB | 3.37 MiB/s, done
Resolving deltas: 100% (34/34), done.
```

JIKA CLONE BERHASIL MA
MUNCUL SEPERTI INI DAN



The screenshot shows the Visual Studio Code interface with a project named 'hardhat-security-fcc'. The Explorer sidebar on the left shows the file structure, including 'package.json'. The main editor area displays the content of 'package.json', which lists development dependencies like '@nomiclabs/hardhat-ethers' and production dependencies like '@chainlink/contracts'. The bottom terminal panel shows the output of a git clone command, confirming the successful cloning of the repository and the resolution of deltas.

```
package.json - hardhat-security-fcc - Visual Studio Code

package.json
{
  "devDependencies": {
    "@nomiclabs/hardhat-ethers": "npm:hardhat-deploy-ethers",
    "ethers": "^5.5.4",
    "hardhat": "^2.9.0",
    "hardhat-deploy": "^0.10.5"
  },
  "dependencies": {
    "@chainlink/contracts": "^0.4.0",
    "@nomiclabs/hardhat-etherscan": "^3.0.3",
    "@nomiclabs/hardhat-waffle": "^2.0.3",
    "@openzeppelin/contracts": "^4.5.0",
    "@openzeppelin/hardhat-upgrades": "^1.15.0",
    "dotenv": "^16.0.0",
    "hardhat-contract-sizer": "^2.5.0",
    "hardhat-gas-reporter": "^1.0.8",
    "prettier": "^2.5.1",
    "prettier-plugin-solidity": "^1.0.0-beta.19",
    "solidity-coverage": "^0.7.20"
  },
  "scripts": {
    "slither": "slither . --solc-remaps '@openzeppelin=node_modules/@openzeppelin @chainlink=node_modules/@chainlink' --exc"
  }
}
```

```
main
b.com/PatrickAlphaC/hardhat-security-fcc.git
Cloning into '.'...
remote: Enumerating objects: 79, done.
remote: Counting objects: 100% (79/79), done.
remote: Compressing objects: 100% (61/61), done.
remote: Total 79 (delta 34), reused 48 (delta 14), pack-reused 0
Receiving objects: 100% (79/79), 168.40 KiB | 3.37 MiB/s, done
```

```
PS E:\Blockchain\hardhat-security-fcc> python --version
Python 3.8.5
```

INSTALL PYTHON DAN JIKA SUDAH INSTALL
CEK VERSION PYTHON TERSEBUT (INI
SUDAH PERNAH INSTALL SEBELUMNYA)

```
PS E:\Blockchain\hardhat-security-fcc> pip --version  
pip 22.0.3 from c:\users\fath\appdata\local\programs\python\python38-32\lib\site-packages\pip (python 3.8)
```

INSTALL PIP DAN CEK VERSION PIP
TERSEBUT (INI SUDAH PERNAH INSTALL
SEBELUMNYA)

```
PS E:\Blockchain\hardhat-security-fcc> pip3 install solc-select  
  Downloading solc_select-0.2.1-py3-none-any.whl (16 kB)  
Installing collected packages: solc-select  
Successfully installed solc-select-0.2.1
```

LALU PADA PIP KITA INSTALL SOLC-
SELECT

```
PS E:\BlockChain\hardhat-security-fcc> pip3 install slither-analyzer
Collecting slither-analyzer
  Downloading slither_analyzer-0.8.3-py3-none-any.whl (547 kB)
    ━━━━━━━━━━━━━━━━━━━ 547.9/547.9 kB 6.9 MB/s eta 0:00:00
Requirement already satisfied: pysha3>=1.0.2 in c:\users\fath\appdata\local\programs\python\python38-32\lib\site-packages (from slither-analyzer) (1.0.2)
Collecting crytic-compile>=0.2.3
  Downloading crytic_compile-0.2.3-py3-none-any.whl (87 kB)
    ━━━━━━━━━━━━━━━━━━━ 87.2/87.2 kB 5.1 MB/s eta 0:00:00
Collecting prettytable>=0.7.2
  Downloading prettytable-3.3.0-py3-none-any.whl (26 kB)
Collecting wcwidth
  Downloading wcwidth-0.2.5-py2.py3-none-any.whl (30 kB)
Installing collected packages: wcwidth, prettytable, crytic-compile, slither-analyzer
Successfully installed crytic-compile-0.2.3 prettytable-3.3.0 slither-analyzer-0.8.3 wcwidth-0.2.5
```

LALU KITA INSTALL JUGA SLITHER-ANALYZER

CEK JIKA SLITHER SUDAH TERINSTALL DENGAN BAIK MENGUNAKAN COMMAND SLITHER --HELP, JIKA BERJALAN DENGAN BAIK DAN TIDAK ADA ERROR MAKA SLITHER TERINSTALL DENGAN BAIK

```
PS E:\Blockchain\hardhat-security-fcc> slither --help
usage: slither target [flag]

target can be:
  - file.sol // a Solidity file
  - project_directory // a project directory. See https://github.com/crytic/crytic-compile/#crytic-compile for the supported platforms
  - 0x.. // a contract on mainnet
  - NETWORK:0x.. // a contract on a different network. Supported networks: mainnet, ropsten, kovan, rinkeby, goerli, tobalab, bsc, testnet.bsc, arbi, testnet.arbi, poly, avax, testnet.avax, ftn

For usage information, see
https://github.com/crytic/slither/wiki/Usage

optional arguments:
  -h, --help            show this help message and exit
  --version              displays the current version

Compile options:
  --compile-force-framework COMPILE_FORCE_FRAMEWORK
                        Force the compile to a given framework (solc, truffle, embark, dapp, etherlime, etherscan, vyper, waffle, brownie, solc-json, buidler, hardhat, foundry, standard, archive)
  --compile-remove-metadata
                        Remove the metadata from the bytecode
  --compile-custom-build COMPILE_CUSTOM_BUILD
                        Replace platform specific build command
  --ignore-compile       Do not run compile of any platform

Solc options:
  --solc SOLC            solc path
  --solc-remaps SOLC_REMAPS
```

```
Additional options:
  --json JSON           Export the results as a JSON file ("--json -" to export to stdout)
  --sarif SARIF         Export the results as a SARIF JSON file ("--sarif -" to export to stdout)
  --json-types JSON_TYPES
                        Comma-separated list of result types to output to JSON, defaults to detectors, printers. Available types: compilations, console, detectors, printers, list-detectors, list-printers
  --zip ZIP             Export the results as a zipped JSON file
  --zip-type ZIP_TYPE   Zip compression type. One of lzma, stored, deflated, bzip2. Default lzma
  --markdown-root MARKDOWN_ROOT
                        URL for markdown generation
  --filter-paths FILTER_PATHS
                        Comma-separated list of paths for which results will be excluded
  --triage-mode          Run triage mode (save results in slither.db.json)
  --config-file CONFIG_FILE
                        Provide a config file (default: slither.config.json)
  --solc-ast            Provide the contract as a json AST
  --generate-patches     Generate patches (json output only)
```

```
PS E:\Blockchain\hardhat-security-fcc> npm install --global ya
```

```
found 0 vulnerabilities
```

```
npm notice
```

```
npm notice New minor version of npm available! 8.7.0 -> 8.13.2
```

```
npm notice Changelog: https://github.com/npm/cli/releases/tag/v8.13.2
```

```
npm notice Run npm install -g npm@8.13.2 to update!
```

```
npm notice
```

IN
NPM INSTALL --GLOBAL YARN

```
PS E:\Blockchain\hardhat-security-fcc> yarn --version  
1.22.19
```

LALU KITA CEK VERSION DARI YARN
TERSEBUT, AGAR MENANDAKAN YARN
SUDAH TERINSTALL

```
PS E:\Blockchain\hardhat-security-fcc> yarn
yarn install v1.22.19
[1/4] Resolving packages...
[2/4] Fetching packages...
[3/4] Linking dependencies...
warning " > @nomiclabs/hardhat-waffle@2.0.3" has incorrect peer
dependency "@nomiclabs/hardhat-ethers@^2.0.0".
warning " > @nomiclabs/hardhat-waffle@2.0.3" has unmet peer de
pendency "ethereum-waffle@^3.2.0".
warning " > @openzeppelin/hardhat-upgrades@1.19.0" has incorre
ct peer dependency "@nomiclabs/hardhat-ethers@^2.0.0".
warning " > @openzeppelin/hardhat-upgrades@1.19.0" has incorre
ct peer dependency "@nomiclabs/hardhat-etherscan@^3.1.0".
warning " > hardhat-deploy@0.10.5" has unmet peer dependency "
@ethersproject/hardware-wallets@^5.0.14".
[4/4] Building fresh packages...
Done in 142.14s.
```

LALU KITA HIDUPKAN YARN TERSEBUT
DENGAN COMMAND YARN

LAI DE FILE TER

```
package.json X
package.json > {} scripts > slither

6   "hardhat-deploy": "^0.10.5"
7   },
8   "dependencies": {
9     "@chainlink/contracts": "^0.4.0",
10    "@nomiclabs/hardhat-etherscan": "^3.0.3",
11    "@nomiclabs/hardhat-waffle": "^2.0.3",
12    "@openzeppelin/contracts": "^4.5.0",
13    "@openzeppelin/hardhat-upgrades": "^1.15.0",
14    "dotenv": "^16.0.0",
15    "hardhat-contract-sizer": "^2.5.0",
16    "hardhat-gas-reporter": "^1.0.8",
17    "prettier": "^2.5.1",
18    "prettier-plugin-solidity": "^1.0.0-beta.19",
19    "solidity-coverage": "^0.7.20"
20  },
21  > Debug
22  "scripts": {
23    "slither": "slither --solc-remaps '@openzeppelin=node_modules/@openzeppelin @chainlink=node_modules/@chainlink' --exc",
24    "toolbox": "docker run -it --rm -v $PWD:/src trailofbits/eth-security-toolbox",
25    "lint": "solhint 'contracts/*.sol'",
26    "lint:fix": "solhint 'contracts/**/*.sol' --fix",
27    "format": "prettier --write ."
28  }
29 }
```

```
PS E:\BlockChain\hardhat-security-fcc> slither . -  
ps '@openzeppelin=node_modules/@openzeppelin @chai  
modules/@chainlink' --exclude naming-convention,e  
nction,low-level-calls  
'npx hardhat compile --force' running  
Downloading compiler 0.8.7  
Compiled 12 Solidity files successfully
```

JIKA BERHASIL AKAN MUNCUL SEPERTI INI,
NANTINYA SLITHER AKAN MENGANALISA
SEMUA CONTRACT YANG ADA DI FOLDER
INI.

Reentrancy in EtherStore.withdraw() (contracts/Reentrancy.sol#15-21):

External calls:

- (success) = msg.sender.call{value: balance, gas: 50000} (contracts/Reentrancy.sol#18)

State variables written after the call(s):

- balances[msg.sender] = 0 (contracts/Reentrancy.sol#20)

Reference: <https://github.com/crytic/slither/wiki/documentation#reentrancy-vulnerabilities>

MetamorphicContract.owner (contracts/MetamorphicContract.sol#6) is never initialized. It is used in:

- MetamorphicContract.kill() (contracts/MetamorphicContract.sol#8-11)

Reference: <https://github.com/crytic/slither/wiki/documentation#uninitialized-state-variables>

Address.verifyCallResult(bool,bytes,string) (node_modules/@openzeppelin/contracts/utils/Address.sol#201-221) used

- INLINE ASM (node_modules/@openzeppelin/contracts/Address.sol#213-216)

Reference: <https://github.com/crytic/slither/wiki/documentation#assembly-usage>

VaultFuzzTest.echidna_test_find_password() (contracts/fuzzing/VaultFuzzTest.sol#9-11) compares to a boolean

- s_locked == true (contracts/test/fuzzing/VaultFuzzTest.sol#10)

Reference: <https://github.com/crytic/slither/wiki/documentation#boolean-equality>

Different versions of Solidity are used:

- Version used: ['0.8.7', '^0.8.0', '^0.8.7']

- ^0.8.0 (node_modules/@openzeppelin/contracts/Initializable.sol#4)

- ^0.8.0 (node_modules/@openzeppelin/contracts/ERC20/ERC20.sol#4)

- ^0.8.0 (node_modules/@openzeppelin/contracts/ERC20/IERC20.sol#4)

- ^0.8.0 (node_modules/@openzeppelin/contracts/ERC20/extensions/IERC20Metadata.sol#4)

- ^0.8.1 (node_modules/@openzeppelin/contracts/Address.sol#4)

- ^0.8.0 (node_modules/@openzeppelin/contracts/Context.sol#4)

- ^0.8.0 (node_modules/@openzeppelin/contracts/Context.sol#4)

- ^0.8.0 (contracts/BadRNG.sol#2)

- ^0.8.0 (contracts/LiquidityPoolAsOracle.sol#2)

- 0.8.7 (contracts/MetamorphicContract.sol#2)

- ^0.8.0 (contracts/Reentrancy.sol#2)

- ^0.8.0 (contracts/Vault.sol#8)

- ^0.8.7 (contracts/test/fuzzing/VaultFuzzTest.sol#10)

Reference: <https://github.com/crytic/slither/wiki/documentation#different-pragma-directives-are-used>

Pragma version^0.8.0 (node_modules/@openzeppelin/contracts/Initializable.sol#4) allows old version

Pragma version^0.8.0 (node_modules/@openzeppelin/contracts/ERC20/ERC20.sol#4) allows old versions

Pragma version^0.8.0 (node_modules/@openzeppelin/contracts/ERC20/IERC20.sol#4) allows old versions

Pragma version^0.8.0 (node_modules/@openzeppelin/contracts/ERC20/extensions/IERC20Metadata.sol#4) allows old versions

Pragma version^0.8.1 (node_modules/@openzeppelin/contracts/Address.sol#4) allows old versions

Pragma version^0.8.0 (node_modules/@openzeppelin/contracts/Context.sol#4) allows old versions

Pragma version^0.8.0 (contracts/BadRNG.sol#2) allows old versions

Pragma version^0.8.0 (contracts/LiquidityPoolAsOracle.sol#2) allows old versions

Pragma version^0.8.0 (contracts/Reentrancy.sol#2) allows old versions

Pragma version^0.8.7 (contracts/test/fuzzing/VaultFuzzTest.sol#10) allows old versions

MUNCUL SEPTEMBER

```
Pragma version^0.8.0 (contracts/Reentrancy.sol#2)
versions
Pragma version^0.8.0 (contracts/Vault.sol#8) allow
ions
Reference: https://github.com/crytic/slither/wiki/
ocumentation#incorrect-versions-of-solidity

BadRNG.enterRaffle() (contracts/BadRNG.sol#11-14)
als with too many digits:
- require(bool)(msg.value >= 10000000000000000000) (contracts/BadRNG.sol#12)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#too-many-digits

MetamorphicContract.owner (contracts/MetamorphicContract.sol#6) should be constant
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant
```

TECATAT ADA 17 MASALAH YANG

analyzed (12 contracts with 75 detectors), 17 result(s) found

MAUPUN YANG KECIL

Hasil yang berwarna merah itu berarti adalah issue yang mempunyai high impact jika diperbaiki, maka saat kita mendeploy contract yang sudah diperbaiki dan contract yang belum diperbaiki akan terlihat banyak perbedaan setelah perubahan dan kita akan menemukan issue yang sangat banyak pada contract yang belum diperbaiki.

Sedangkan hasil yang berwarna hijau adalah issu yang mempunyai low impact jika diperbaiki, contohnya seperti version solidity yang berbeda sedikit di semua file, lalu jika ada version solidity yang sudah tua, dan lainnya.

PENJELASAN HASIL ANALISIS SLITHER PADA CONTRACT