0x Guard

Smart contracts security assessment

Final report

Tariff: Standard

Bitcoin.com

February 2022

0xguard.com

hello@0xguard.com

Contents

Introduction

UniswapV2 fork contract with a code update for the new version of solidity and deployment using Minimal Proxy pattern. The code is available in the Github repository. The code was checked in 89956da commit.

Name	Bitcoin.com
Audit date	2022-02-11 - 2022-02-14
Language	Solidity
Platform	SmartBCH

Contracts checked

Name	Address
Multiple contract	
SwapsFactory	https://github.com/bitcoin-portal/bitcoincom-solidity-swap/blob/89956da4ba3e191ebf615ff1109922d51e46ca2d/contracts/SwapsFactory.sol
SwapsPair	https://github.com/bitcoin-portal/bitcoincom-solidity-swap/blob/89956da4ba3e191ebf615ff1109922d51e46ca2d/contracts/SwapsPair.sol
SwapsRouter	https://github.com/bitcoin-portal/bitcoincom-solidity-swap/blob/89956da4ba3e191ebf615ff1109922d51e46ca2d/contracts/SwapsRouter.sol
SwapsERC20	https://github.com/bitcoin-portal/bitcoincom-solidity-swap/blob/89956da4ba3e191ebf615ff1109922d51e46ca2d/contracts/SwapsERC20.sol

Procedure

We perform our audit according to the following procedure:

Automated analysis

- Scanning the project's smart contracts with several publicly available automated Solidity analysis tools
- Manual verification (reject or confirm) all the issues found by the tools

Manual audit

- Manually analyse smart contracts for security vulnerabilities
- Smart contracts' logic check

Known vulnerabilities checked

Title	Check result
Unencrypted Private Data On-Chain	passed
Code With No Effects	passed
Message call with hardcoded gas amount	passed
Typographical Error	passed
DoS With Block Gas Limit	passed
Presence of unused variables	passed
Incorrect Inheritance Order	passed
Requirement Violation	passed
Weak Sources of Randomness from Chain Attributes	passed
Shadowing State Variables	passed

Incorrect Constructor Name	passed
Block values as a proxy for time	passed
Authorization through tx.origin	passed
DoS with Failed Call	passed
Delegatecall to Untrusted Callee	passed
Use of Deprecated Solidity Functions	passed
Assert Violation	passed
State Variable Default Visibility	passed
Reentrancy	passed
Unprotected SELFDESTRUCT Instruction	passed
Unprotected Ether Withdrawal	passed
Unchecked Call Return Value	passed
Floating Pragma	passed
Outdated Compiler Version	passed
Integer Overflow and Underflow	passed
Function Default Visibility	passed

Classification of issue severity

High severity High severity issues can cause a significant or full loss of funds, change of contract ownership, major interference with contract logic. Such issues require immediate attention.

Medium severity Medium severity issues do not pose an immediate risk, but can be detrimental to the client's reputation if exploited. Medium severity issues may lead to a contract failure and can be fixed by modifying the contract state or redeployment. Such issues require attention.

Low severity Low severity issues do not cause significant destruction to the contract's functionality. Such issues are recommended to be taken into consideration.

Issues

High severity issues

No issues were found

Medium severity issues

1. Missing sync() function - FIXED (SwapsPair)

The `sync()` function is essential to safely store deflationary tokens in the pool, see Uniswap documentation.

Recommendation: Consider restoring `sync()` function.

Update: Following smart contracts recheck, Bitcoin.com team fixed the finding by adding sync() function back - pull request link.

Low severity issues

1. Floating Pragma - FIXED (Multiple contract)

Contracts should be deployed with the same compiler version and flags that they have been tested with thoroughly. Locking the pragma helps to ensure that contracts do not accidentally get deployed using, for example, an outdated compiler version that might introduce bugs that affect the contract system negatively.

Recommendation: Lock the pragma version and also consider known bugs (link) for the compiler version that is chosen.

Update: Following smart contracts recheck, Bitcoin.com team fixed the finding by locking pragma version at 0.8.12 - pull request link.

2. Lacks a zero-check on constructor - FIXED (SwapsFactory)

The `constructor()` does not check the input value of the `feeToSetter` variable. In case of incorrect initialization, the commission will come to the wrong address.

Recommendation: Add input validations with `require` in `constructor()`.

Update: Following smart contracts recheck, Bitcoin.com team fixed the finding by adding zero-check for feeToSetter variable - pull request link.

3. Non-cancellable fee - FIXED (SwapsPair)

In this contract, in the `mintFee()`, `mint()`, `burn()` functions, the logic with disabling the fee has been removed. The `feeOn` variable was responsible for this in UniswapV2. If the `SwapsFactory` variable `feeTo` is a zero address, then the fee is still charged.

Recommendation: Double-check your business requirements for this code. It might be worth reverting the fee disable logic from UniswapV2 or adding some input validation in the `SwapsFactory:setFeeTo()` and `SwapsFactory:setFeeToSetter()` functions.

Update: The team decided to leave non-cancellable fee as a default behavior.

4. Redundant code - FIXED (SwapsRouter)

The contract for `L992` uses a line that does not affect the code.

Recommendation: Delete the line

```
delete amountIn;
```

Update: Following smart contracts recheck, Bitcoin.com team fixed the finding by deleting proposed line.

5. Deprecated assert - FIXED (SwapsRouter)

`assert` is a deprecated expression in solidity. `require` fully replaces it and spends less gas.

Recommendation: Replace all `assert` statements with `require` statements.

Update: Following smart contracts recheck, Bitcoin.com team fixed the finding by replacing all asserts with require statements - pull request link.

6. Lacks increase and decrease allowance functions (SwapsERC20)

In the file `SwapsERC20.sol` lacks increase and decrease allowance functions. These functions help to mitigate the frontrun approve attacks. To see more follow the link.

Recommendation: Add the specified functions to these files.

Developer response: The development team decided to leave same functionality as in the original Uniswap code base.

Conclusion

Bitcoin.com Multiple contract, SwapsFactory, SwapsPair, SwapsRouter, SwapsERC20 contracts were audited. 1 medium, 6 low severity issues were found.

Update: the medium severity issue and 5 low severity issues were fixed in the update.

Disclaimer

This report is subject to the terms and conditions (including without limitation, description of services, confidentiality, disclaimer and limitation of liability) set forth in the Services Agreement, or the scope of services, and terms and conditions provided to the Company in connection with the Agreement. This report provided in connection with the Services set forth in the Agreement shall be used by the Company only to the extent permitted under the terms and conditions set forth in the Agreement. This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes without 0xGuard prior written consent.

This report is not, nor should be considered, an “endorsement” or “disapproval” of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any “product” or “asset” created by any team or project that contracts 0xGuard to perform a security assessment. This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model or legal compliance.

This report should not be used in any way to make decisions around investment or involvement with any particular project. This report in no way provides investment advice, nor should be leveraged as investment advice of any sort. This report represents an extensive assessing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Slither output

```
SwapsPair._update(uint256,uint256,uint112,uint112) (SwapsPair.sol#104-135) uses a weak
PRNG: "blockTimestamp = uint32(block.timestamp % 2 ** 32) (SwapsPair.sol#118)"
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#weak-PRNG

SwapsPair._mintFee(uint112,uint112,uint256) (SwapsPair.sol#137-162) uses a dangerous
strict equality:
        - _kLast == 0 (SwapsPair.sol#144)
SwapsPair._safeTransfer(address,address,uint256) (SwapsPair.sol#483-506) uses a
dangerous strict equality:
        - require(bool,string)(success == true && (data.length == 0 || abi.decode(data,
(bool))),SwapsPair: TRANSFER_FAILED) (SwapsPair.sol#498-505)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dangerous-
strict-equalities

Reentrancy in SwapsPair.burn(address) (SwapsPair.sol#236-312):
        External calls:
        - _safeTransfer(_token0,_to,amount0) (SwapsPair.sol#282-286)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - _safeTransfer(_token1,_to,amount1) (SwapsPair.sol#288-292)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        State variables written after the call(s):
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#297-302)
                - blockTimestampLast = blockTimestamp (SwapsPair.sol#129)
        - kLast = uint256(reserve0) * reserve1 (SwapsPair.sol#304)
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#297-302)
                - reserve0 = uint112(_balance0) (SwapsPair.sol#126)
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#297-302)
                - reserve1 = uint112(_balance1) (SwapsPair.sol#127)
Reentrancy in SwapsFactory.createPair(address,address) (SwapsFactory.sol#57-132):
        External calls:
        - ISwapsPair(pair).initialize(token0,token1) (SwapsFactory.sol#116-119)
        State variables written after the call(s):
        - getPair[token0][token1] = pair (SwapsFactory.sol#121)
        - getPair[token1][token0] = pair (SwapsFactory.sol#122)
Reentrancy in SwapsPair.swap(uint256,uint256,address,bytes) (SwapsPair.sol#314-403):
        External calls:
        - _safeTransfer(_token0,_to,_amount0Out) (SwapsPair.sol#348)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - _safeTransfer(_token1,_to,_amount1Out) (SwapsPair.sol#349)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - ISwapsCallee(_to).swapsCall(msg.sender,_amount0Out,_amount1Out,_data)
(SwapsPair.sol#351-356)
        State variables written after the call(s):
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#388-393)
                - blockTimestampLast = blockTimestamp (SwapsPair.sol#129)
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#388-393)
                - reserve0 = uint112(_balance0) (SwapsPair.sol#126)
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#388-393)
                - reserve1 = uint112(_balance1) (SwapsPair.sol#127)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-
vulnerabilities-1

SwapsFactory.constructor(address)._feeToSetter (SwapsFactory.sol#27) lacks a zero-check
on :
                - feeToSetter = _feeToSetter (SwapsFactory.sol#29)
                - feeTo = _feeToSetter (SwapsFactory.sol#30)
SwapsFactory.setFeeTo(address)._feeTo (SwapsFactory.sol#135) lacks a zero-check on :
                - feeTo = _feeTo (SwapsFactory.sol#144)
SwapsFactory.setFeeToSetter(address)._feeToSetter (SwapsFactory.sol#148) lacks a zero-
check on :
                - feeToSetter = _feeToSetter (SwapsFactory.sol#157)
SwapsPair.initialize(address,address)._token0 (SwapsPair.sol#72) lacks a zero-check
on :
                - token0 = _token0 (SwapsPair.sol#82)
SwapsPair.initialize(address,address)._token1 (SwapsPair.sol#73) lacks a zero-check
on :
                - token1 = _token1 (SwapsPair.sol#83)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-zero-
address-validation

Reentrancy in SwapsPair.burn(address) (SwapsPair.sol#236-312):
        External calls:
        - _safeTransfer(_token0,_to,amount0) (SwapsPair.sol#282-286)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - _safeTransfer(_token1,_to,amount1) (SwapsPair.sol#288-292)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        State variables written after the call(s):
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#297-302)
                - price0CumulativeLast += uint256(uqdiv(encode(_reserve1),_reserve0)) *
timeElapsed (SwapsPair.sol#122)
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#297-302)
                - price1CumulativeLast += uint256(uqdiv(encode(_reserve0),_reserve1)) *
timeElapsed (SwapsPair.sol#123)
Reentrancy in SwapsFactory.createPair(address,address) (SwapsFactory.sol#57-132):
        External calls:
        - ISwapsPair(pair).initialize(token0,token1) (SwapsFactory.sol#116-119)
        State variables written after the call(s):
        - allPairs.push(pair) (SwapsFactory.sol#124)
Reentrancy in SwapsPair.swap(uint256,uint256,address,bytes) (SwapsPair.sol#314-403):
        External calls:
        - _safeTransfer(_token0,_to,_amount0Out) (SwapsPair.sol#348)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - _safeTransfer(_token1,_to,_amount1Out) (SwapsPair.sol#349)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - ISwapsCallee(_to).swapsCall(msg.sender,_amount0Out,_amount1Out,_data)
(SwapsPair.sol#351-356)
        State variables written after the call(s):
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#388-393)
                - price0CumulativeLast += uint256(uqdiv(encode(_reserve1),_reserve0)) *
timeElapsed (SwapsPair.sol#122)
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#388-393)
                - price1CumulativeLast += uint256(uqdiv(encode(_reserve0),_reserve1)) *
timeElapsed (SwapsPair.sol#123)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-
vulnerabilities-2

Reentrancy in SwapsPair.burn(address) (SwapsPair.sol#236-312):
        External calls:
        - _safeTransfer(_token0,_to,amount0) (SwapsPair.sol#282-286)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
```

```
        - _safeTransfer(_token1,_to,amount1) (SwapsPair.sol#288-292)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        Event emitted after the call(s):
        - Burn(msg.sender,amount0,amount1,_to) (SwapsPair.sol#306-311)
        - Sync(reserve0,reserve1) (SwapsPair.sol#131-134)
                - _update(balance0,balance1,_reserve0,_reserve1)
(SwapsPair.sol#297-302)
Reentrancy in SwapsFactory.createPair(address,address) (SwapsFactory.sol#57-132):
        External calls:
        - ISwapsPair(pair).initialize(token0,token1) (SwapsFactory.sol#116-119)
        Event emitted after the call(s):
        - PairCreated(token0,token1,pair,allPairs.length) (SwapsFactory.sol#126-131)
Reentrancy in SwapsPair.swap(uint256,uint256,address,bytes) (SwapsPair.sol#314-403):
        External calls:
        - _safeTransfer(_token0,_to,_amount0Out) (SwapsPair.sol#348)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - _safeTransfer(_token1,_to,_amount1Out) (SwapsPair.sol#349)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - ISwapsCallee(_to).swapsCall(msg.sender,_amount0Out,_amount1Out,_data)
(SwapsPair.sol#351-356)
        Event emitted after the call(s):
        - Swap(msg.sender,_amount0In,_amount1In,_amount0Out,_amount1Out,_to)
(SwapsPair.sol#395-402)
        - Sync(reserve0,reserve1) (SwapsPair.sol#131-134)
                - _update(balance0,balance1,_reserve0,_reserve1)
(SwapsPair.sol#388-393)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-
vulnerabilities-3

SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)
(SwapsERC20.sol#182-233) uses timestamp for comparisons
        Dangerous comparisons:
        - require(bool,string)(_deadline >= block.timestamp,PERMIT_CALL_EXPIRED)
(SwapsERC20.sol#193-196)
SwapsPair._update(uint256,uint256,uint112,uint112) (SwapsPair.sol#104-135) uses
timestamp for comparisons
        Dangerous comparisons:
        - timeElapsed > 0 && _reserve0 != 0 && _reserve1 != 0 (SwapsPair.sol#121)
```

```
SwapsPair.swap(uint256,uint256,address,bytes) (SwapsPair.sol#314-403) uses timestamp
for comparisons
        Dangerous comparisons:
        - require(bool,string)(_amount0Out < _reserve0 && _amount1Out <
_reserve1,INSUFFICIENT_LIQUIDITY) (SwapsPair.sol#335-339)
        - require(bool,string)(_amount0In > 0 || _amount1In >
0,INSUFFICIENT_INPUT_AMOUNT) (SwapsPair.sol#370-374)
        - require(bool)(balance0Adjusted * balance1Adjusted >= uint256(_reserve0) *
_reserve1 * (1000 ** 2)) (SwapsPair.sol#380-385)
        - balance0 > _reserve0 - _amount0Out (SwapsPair.sol#362-364)
        - balance1 > _reserve1 - _amount1Out (SwapsPair.sol#366-368)
SwapsPair.min(uint256,uint256) (SwapsPair.sol#451-460) uses timestamp for comparisons
        Dangerous comparisons:
        - _x < _y (SwapsPair.sol#459)
SwapsPair.sqrt(uint256) (SwapsPair.sol#462-481) uses timestamp for comparisons
        Dangerous comparisons:
        - _y > 3 (SwapsPair.sol#470)
        - x < z (SwapsPair.sol#473)
        - _y != 0 (SwapsPair.sol#477)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#block-
timestamp

SwapsFactory.constructor(address) (SwapsFactory.sol#26-47) uses assembly
        - INLINE ASM (SwapsFactory.sol#37-44)
SwapsFactory.createPair(address,address) (SwapsFactory.sol#57-132) uses assembly
        - INLINE ASM (SwapsFactory.sol#94-114)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#assembly-usage

SwapsPair._safeTransfer(address,address,uint256) (SwapsPair.sol#483-506) compares to a
boolean constant:
        -require(bool,string)(success == true && (data.length == 0 || abi.decode(data,
(bool))),SwapsPair: TRANSFER_FAILED) (SwapsPair.sol#498-505)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#boolean-
equality

Pragma version^0.8.9 (IERC20.sol#3) necessitates a version too recent to be trusted.
Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (ISwapsCallee.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (ISwapsERC20.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
```

```
Pragma version^0.8.9 (ISwapsFactory.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (ISwapsPair.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (SwapsERC20.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (SwapsFactory.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (SwapsPair.sol#3) necessitates a version too recent to be trusted.
Consider deploying with 0.6.12/0.7.6/0.8.7
solc-0.8.11 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-
versions-of-solidity

Low level call in SwapsPair._safeTransfer(address,address,uint256)
(SwapsPair.sol#483-506):
        - (success,data) = _token.call(abi.encodeWithSelector(SELECTOR,_to,_value))
(SwapsPair.sol#490-496)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#low-level-
calls

Function ISwapsERC20.DOMAIN_SEPARATOR() (ISwapsERC20.sol#64-67) is not in mixedCase
Function ISwapsERC20.PERMIT_TYPEHASH() (ISwapsERC20.sol#69-72) is not in mixedCase
Function ISwapsPair.MINIMUM_LIQUIDITY() (ISwapsPair.sol#9-12) is not in mixedCase
Parameter SwapsERC20.approve(address,uint256)._spender (SwapsERC20.sol#130) is not in
mixedCase
Parameter SwapsERC20.approve(address,uint256)._value (SwapsERC20.sol#131) is not in
mixedCase
Parameter SwapsERC20.transfer(address,uint256)._to (SwapsERC20.sol#146) is not in
mixedCase
Parameter SwapsERC20.transfer(address,uint256)._value (SwapsERC20.sol#147) is not in
mixedCase
Parameter SwapsERC20.transferFrom(address,address,uint256)._from (SwapsERC20.sol#162)
is not in mixedCase
Parameter SwapsERC20.transferFrom(address,address,uint256)._to (SwapsERC20.sol#163) is
not in mixedCase
Parameter SwapsERC20.transferFrom(address,address,uint256)._value (SwapsERC20.sol#164)
is not in mixedCase
Parameter
SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._owner
(SwapsERC20.sol#183) is not in mixedCase
```

```
Parameter
SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._spender
(SwapsERC20.sol#184) is not in mixedCase
Parameter
SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._value
(SwapsERC20.sol#185) is not in mixedCase
Parameter
SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._deadline
(SwapsERC20.sol#186) is not in mixedCase
Parameter SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._v
(SwapsERC20.sol#187) is not in mixedCase
Parameter SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._r
(SwapsERC20.sol#188) is not in mixedCase
Parameter SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._s
(SwapsERC20.sol#189) is not in mixedCase
Variable SwapsERC20.DOMAIN_SEPARATOR (SwapsERC20.sol#20) is not in mixedCase
Parameter SwapsFactory.createPair(address,address)._tokenA (SwapsFactory.sol#58) is not
in mixedCase
Parameter SwapsFactory.createPair(address,address)._tokenB (SwapsFactory.sol#59) is not
in mixedCase
Parameter SwapsFactory.setFeeTo(address)._feeTo (SwapsFactory.sol#135) is not in
mixedCase
Parameter SwapsFactory.setFeeToSetter(address)._feeToSetter (SwapsFactory.sol#148) is
not in mixedCase
Parameter SwapsPair.initialize(address,address)._token0 (SwapsPair.sol#72) is not in
mixedCase
Parameter SwapsPair.initialize(address,address)._token1 (SwapsPair.sol#73) is not in
mixedCase
Parameter SwapsPair.mint(address)._to (SwapsPair.sol#165) is not in mixedCase
Parameter SwapsPair.burn(address)._to (SwapsPair.sol#237) is not in mixedCase
Parameter SwapsPair.swap(uint256,uint256,address,bytes)._amount0Out (SwapsPair.sol#315)
is not in mixedCase
Parameter SwapsPair.swap(uint256,uint256,address,bytes)._amount1Out (SwapsPair.sol#316)
is not in mixedCase
Parameter SwapsPair.swap(uint256,uint256,address,bytes)._to (SwapsPair.sol#317) is not
in mixedCase
Parameter SwapsPair.swap(uint256,uint256,address,bytes)._data (SwapsPair.sol#318) is
not in mixedCase
Parameter SwapsPair.encode(uint112)._y (SwapsPair.sol#427) is not in mixedCase
Parameter SwapsPair.uqdiv(uint224,uint112)._x (SwapsPair.sol#439) is not in mixedCase
Parameter SwapsPair.uqdiv(uint224,uint112)._y (SwapsPair.sol#440) is not in mixedCase
Parameter SwapsPair.min(uint256,uint256)._x (SwapsPair.sol#452) is not in mixedCase
Parameter SwapsPair.min(uint256,uint256)._y (SwapsPair.sol#453) is not in mixedCase
Parameter SwapsPair.sqrt(uint256)._y (SwapsPair.sol#463) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-
solidity-naming-conventions

Variable ISwapsPair.swap(uint256,uint256,address,bytes)._amount0Out (ISwapsPair.sol#69)
is too similar to ISwapsPair.swap(uint256,uint256,address,bytes)._amount1Out
(ISwapsPair.sol#70)
Variable SwapsPair.swap(uint256,uint256,address,bytes)._amount0Out (SwapsPair.sol#315)
is too similar to SwapsPair.swap(uint256,uint256,address,bytes)._amount1Out
(SwapsPair.sol#316)
Variable SwapsPair.swap(uint256,uint256,address,bytes).balance0Adjusted
(SwapsPair.sol#377) is too similar to
SwapsPair.swap(uint256,uint256,address,bytes).balance1Adjusted (SwapsPair.sol#378)
Variable SwapsPair.price0CumulativeLast (SwapsPair.sol#28) is too similar to
SwapsPair.price1CumulativeLast (SwapsPair.sol#29)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-
are-too-similar

SwapsFactory.constructor(address) (SwapsFactory.sol#26-47) uses literals with too many
digits:
        - bytecode = type()(SwapsPair).creationCode (SwapsFactory.sol#35)
SwapsFactory.createPair(address,address) (SwapsFactory.sol#57-132) uses literals with
too many digits:
        - mstore(uint256,uint256)(clone_createPair_asm_0,0x3d602d80600a3d3981f3363d3d373
d3d3d363d73000000000000000000000000) (SwapsFactory.sol#98-101)
SwapsFactory.createPair(address,address) (SwapsFactory.sol#57-132) uses literals with
too many digits:
        - mstore(uint256,uint256)(clone_createPair_asm_0 +
0x28,0x5af43d82803e903d91602b57fd5bf30000000000000000000000000000000000)
(SwapsFactory.sol#108-111)
FactoryCodeCheck.factoryCodeHash() (SwapsFactory.sol#163-171) uses literals with too
many digits:
        - keccak256(bytes)(type()(SwapsFactory).creationCode)
(SwapsFactory.sol#168-170)
FactoryCodeCheck.pairCodeHash() (SwapsFactory.sol#173-181) uses literals with too many
digits:
        - keccak256(bytes)(type()(SwapsPair).creationCode) (SwapsFactory.sol#178-180)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#too-many-
digits
```

```
SwapsHelper._pairFor(address,address,address,address) (SwapsHelper.sol#197-228) uses
assembly
        - INLINE ASM (SwapsHelper.sol#218-227)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#assembly-usage

SwapsHelper._safeTransfer(address,address,uint256) (SwapsHelper.sol#129-152) compares
to a boolean constant:
        -require(bool,string)(success == true && (data.length == 0 || abi.decode(data,
(bool))),TRANSFER_FAILED) (SwapsHelper.sol#144-151)
SwapsHelper._safeTransferFrom(address,address,address,uint256)
(SwapsHelper.sol#154-179) compares to a boolean constant:
        -require(bool,string)(success == true && (data.length == 0 || abi.decode(data,
(bool))),TRANSFER_FROM_FAILED) (SwapsHelper.sol#171-178)
SwapsHelper._safeTransferETH(address,uint256) (SwapsHelper.sol#181-195) compares to a
boolean constant:
        -require(bool,string)(success == true,ETH_TRANSFER_FAILED)
(SwapsHelper.sol#191-194)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#boolean-
equality

SwapsHelper._pairFor(address,address,address,address) (SwapsHelper.sol#197-228) is
never used and should be removed
SwapsHelper._safeTransfer(address,address,uint256) (SwapsHelper.sol#129-152) is never
used and should be removed
SwapsHelper._safeTransferETH(address,uint256) (SwapsHelper.sol#181-195) is never used
and should be removed
SwapsHelper._safeTransferFrom(address,address,address,uint256)
(SwapsHelper.sol#154-179) is never used and should be removed
SwapsHelper.sortTokens(address,address) (SwapsHelper.sol#10-34) is never used and
should be removed
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code

Pragma version^0.8.9 (SwapsHelper.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
solc-0.8.11 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-
versions-of-solidity

Low level call in SwapsHelper._safeTransfer(address,address,uint256)
(SwapsHelper.sol#129-152):
        - (success,data) = _token.call(abi.encodeWithSelector(TRANSFER,_to,_value))
(SwapsHelper.sol#136-142)
Low level call in SwapsHelper._safeTransferFrom(address,address,address,uint256)
(SwapsHelper.sol#154-179):
        - (success,data) =
_token.call(abi.encodeWithSelector(TRANSFER_FROM,_from,_to,_value))
(SwapsHelper.sol#162-169)
Low level call in SwapsHelper._safeTransferETH(address,uint256)
(SwapsHelper.sol#181-195):
        - (success) = to.call{value: value}(new bytes(0)) (SwapsHelper.sol#187-189)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#low-level-
calls

Parameter SwapsHelper.sortTokens(address,address)._tokenA (SwapsHelper.sol#11) is not
in mixedCase
Parameter SwapsHelper.sortTokens(address,address)._tokenB (SwapsHelper.sol#12) is not
in mixedCase
Parameter SwapsHelper.quote(uint256,uint256,uint256)._amountA (SwapsHelper.sol#37) is
not in mixedCase
Parameter SwapsHelper.quote(uint256,uint256,uint256)._reserveA (SwapsHelper.sol#38) is
not in mixedCase
Parameter SwapsHelper.quote(uint256,uint256,uint256)._reserveB (SwapsHelper.sol#39) is
not in mixedCase
Parameter SwapsHelper.getAmountOut(uint256,uint256,uint256)._amountIn
(SwapsHelper.sol#61) is not in mixedCase
Parameter SwapsHelper.getAmountOut(uint256,uint256,uint256)._reserveIn
(SwapsHelper.sol#62) is not in mixedCase
Parameter SwapsHelper.getAmountOut(uint256,uint256,uint256)._reserveOut
(SwapsHelper.sol#63) is not in mixedCase
Parameter SwapsHelper.getAmountIn(uint256,uint256,uint256)._amountOut
(SwapsHelper.sol#87) is not in mixedCase
Parameter SwapsHelper.getAmountIn(uint256,uint256,uint256)._reserveIn
(SwapsHelper.sol#88) is not in mixedCase
Parameter SwapsHelper.getAmountIn(uint256,uint256,uint256)._reserveOut
(SwapsHelper.sol#89) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-
solidity-naming-conventions

SwapsHelper._pairFor(address,address,address,address) (SwapsHelper.sol#197-228) uses
literals with too many digits:
        - mstore(uint256,uint256)
```

```
(ptr__pairFor_asm_0,0x3d602d80600a3d3981f3363d3d373d3d3d363d73000000000000000000000000)
(SwapsHelper.sol#220)
SwapsHelper._pairFor(address,address,address,address) (SwapsHelper.sol#197-228) uses
literals with too many digits:
        - mstore(uint256,uint256)(ptr__pairFor_asm_0 +
0x28,0x5af43d82803e903d91602b57fd5bf3ff00000000000000000000000000000000)
(SwapsHelper.sol#222)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#too-many-
digits

SwapsHelper.UINT256_MAX (SwapsHelper.sol#7) is never used in SwapsHelper
(SwapsHelper.sol#5-229)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unused-state-
variable

quote(uint256,uint256,uint256) should be declared external:
        - SwapsHelper.quote(uint256,uint256,uint256) (SwapsHelper.sol#36-58)
getAmountOut(uint256,uint256,uint256) should be declared external:
        - SwapsHelper.getAmountOut(uint256,uint256,uint256) (SwapsHelper.sol#60-84)
getAmountIn(uint256,uint256,uint256) should be declared external:
        - SwapsHelper.getAmountIn(uint256,uint256,uint256) (SwapsHelper.sol#86-110)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#public-
function-that-could-be-declared-external

SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)
(SwapsERC20.sol#182-233) uses timestamp for comparisons
        Dangerous comparisons:
        - require(bool,string)(_deadline >= block.timestamp,PERMIT_CALL_EXPIRED)
(SwapsERC20.sol#193-196)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#block-
timestamp

SwapsERC20._burn(address,uint256) (SwapsERC20.sol#70-89) is never used and should be
removed
SwapsERC20._mint(address,uint256) (SwapsERC20.sol#49-68) is never used and should be
removed
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code

Pragma version^0.8.9 (SwapsERC20.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
solc-0.8.11 is not recommended for deployment
```

```
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-
versions-of-solidity

Parameter SwapsERC20.approve(address,uint256)._spender (SwapsERC20.sol#130) is not in
mixedCase
Parameter SwapsERC20.approve(address,uint256)._value (SwapsERC20.sol#131) is not in
mixedCase
Parameter SwapsERC20.transfer(address,uint256)._to (SwapsERC20.sol#146) is not in
mixedCase
Parameter SwapsERC20.transfer(address,uint256)._value (SwapsERC20.sol#147) is not in
mixedCase
Parameter SwapsERC20.transferFrom(address,address,uint256)._from (SwapsERC20.sol#162)
is not in mixedCase
Parameter SwapsERC20.transferFrom(address,address,uint256)._to (SwapsERC20.sol#163) is
not in mixedCase
Parameter SwapsERC20.transferFrom(address,address,uint256)._value (SwapsERC20.sol#164)
is not in mixedCase
Parameter
SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._owner
(SwapsERC20.sol#183) is not in mixedCase
Parameter
SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._spender
(SwapsERC20.sol#184) is not in mixedCase
Parameter
SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._value
(SwapsERC20.sol#185) is not in mixedCase
Parameter
SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._deadline
(SwapsERC20.sol#186) is not in mixedCase
Parameter SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._v
(SwapsERC20.sol#187) is not in mixedCase
Parameter SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._r
(SwapsERC20.sol#188) is not in mixedCase
Parameter SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._s
(SwapsERC20.sol#189) is not in mixedCase
Variable SwapsERC20.DOMAIN_SEPARATOR (SwapsERC20.sol#20) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-
solidity-naming-conventions

Pragma version^0.8.9 (ISwapsFactory.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
```

```
solc-0.8.11 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-
versions-of-solidity

SwapsHelper._safeTransfer(address,address,uint256) (SwapsHelper.sol#129-152) uses a
dangerous strict equality:
        - require(bool,string)(success == true && (data.length == 0 || abi.decode(data,
(bool))),TRANSFER_FAILED) (SwapsHelper.sol#144-151)
SwapsHelper._safeTransferETH(address,uint256) (SwapsHelper.sol#181-195) uses a
dangerous strict equality:
        - require(bool,string)(success == true,ETH_TRANSFER_FAILED)
(SwapsHelper.sol#191-194)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dangerous-
strict-equalities

SwapsRouter._swapSupportingFeeOnTransferTokens(address[],address).i
(SwapsRouter.sol#862) is a local variable never initialized
SwapsRouter._swap(uint256[],address[],address).i (SwapsRouter.sol#517) is a local
variable never initialized
SwapsRouter._getAmountsOut(address,uint256,address[]).i (SwapsRouter.sol#1160) is a
local variable never initialized
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#uninitialized-
local-variables

SwapsRouter._addLiquidity(address,address,uint256,uint256,uint256,uint256)
(SwapsRouter.sol#46-119) ignores return value by
ISwapsFactory(FACTORY).createPair(_tokenA,_tokenB) (SwapsRouter.sol#58-61)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unused-return

SwapsRouter.constructor(address,address)._factory (SwapsRouter.sol#28) lacks a zero-
check on :
                - FACTORY = _factory (SwapsRouter.sol#31)
                - PAIR = ISwapsFactory(_factory).cloneTarget() (SwapsRouter.sol#33)
SwapsRouter.constructor(address,address)._WETH (SwapsRouter.sol#29) lacks a zero-check
on :
                - WETH = _WETH (SwapsRouter.sol#32)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-zero-
address-validation

SwapsHelper._pairFor(address,address,address,address) (SwapsHelper.sol#197-228) uses
assembly
```

```
        - INLINE ASM (SwapsHelper.sol#218-227)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#assembly-usage

SwapsHelper._safeTransfer(address,address,uint256) (SwapsHelper.sol#129-152) compares
to a boolean constant:
        -require(bool,string)(success == true && (data.length == 0 || abi.decode(data,
(bool))),TRANSFER_FAILED) (SwapsHelper.sol#144-151)
SwapsHelper._safeTransferFrom(address,address,address,uint256)
(SwapsHelper.sol#154-179) compares to a boolean constant:
        -require(bool,string)(success == true && (data.length == 0 || abi.decode(data,
(bool))),TRANSFER_FROM_FAILED) (SwapsHelper.sol#171-178)
SwapsHelper._safeTransferETH(address,uint256) (SwapsHelper.sol#181-195) compares to a
boolean constant:
        -require(bool,string)(success == true,ETH_TRANSFER_FAILED)
(SwapsHelper.sol#191-194)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#boolean-
equality

Pragma version^0.8.9 (IERC20.sol#3) necessitates a version too recent to be trusted.
Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (ISwapsERC20.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (ISwapsFactory.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (ISwapsPair.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (IWETH.sol#3) necessitates a version too recent to be trusted.
Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (SwapsHelper.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (SwapsRouter.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
solc-0.8.11 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-
versions-of-solidity

Low level call in SwapsHelper._safeTransfer(address,address,uint256)
(SwapsHelper.sol#129-152):
        - (success,data) = _token.call(abi.encodeWithSelector(TRANSFER,_to,_value))
(SwapsHelper.sol#136-142)
Low level call in SwapsHelper._safeTransferFrom(address,address,address,uint256)
(SwapsHelper.sol#154-179):
        - (success,data) =
_token.call(abi.encodeWithSelector(TRANSFER_FROM,_from,_to,_value))
(SwapsHelper.sol#162-169)
Low level call in SwapsHelper._safeTransferETH(address,uint256)
(SwapsHelper.sol#181-195):
        - (success) = to.call{value: value}(new bytes(0)) (SwapsHelper.sol#187-189)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#low-level-
calls

Function ISwapsERC20.DOMAIN_SEPARATOR() (ISwapsERC20.sol#64-67) is not in mixedCase
Function ISwapsERC20.PERMIT_TYPEHASH() (ISwapsERC20.sol#69-72) is not in mixedCase
Function ISwapsPair.MINIMUM_LIQUIDITY() (ISwapsPair.sol#9-12) is not in mixedCase
Parameter SwapsHelper.sortTokens(address,address)._tokenA (SwapsHelper.sol#11) is not
in mixedCase
Parameter SwapsHelper.sortTokens(address,address)._tokenB (SwapsHelper.sol#12) is not
in mixedCase
Parameter SwapsHelper.quote(uint256,uint256,uint256)._amountA (SwapsHelper.sol#37) is
not in mixedCase
Parameter SwapsHelper.quote(uint256,uint256,uint256)._reserveA (SwapsHelper.sol#38) is
not in mixedCase
Parameter SwapsHelper.quote(uint256,uint256,uint256)._reserveB (SwapsHelper.sol#39) is
not in mixedCase
Parameter SwapsHelper.getAmountOut(uint256,uint256,uint256)._amountIn
(SwapsHelper.sol#61) is not in mixedCase
Parameter SwapsHelper.getAmountOut(uint256,uint256,uint256)._reserveIn
(SwapsHelper.sol#62) is not in mixedCase
Parameter SwapsHelper.getAmountOut(uint256,uint256,uint256)._reserveOut
(SwapsHelper.sol#63) is not in mixedCase
Parameter SwapsHelper.getAmountIn(uint256,uint256,uint256)._amountOut
(SwapsHelper.sol#87) is not in mixedCase
Parameter SwapsHelper.getAmountIn(uint256,uint256,uint256)._reserveIn
(SwapsHelper.sol#88) is not in mixedCase
Parameter SwapsHelper.getAmountIn(uint256,uint256,uint256)._reserveOut
(SwapsHelper.sol#89) is not in mixedCase
Parameter SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,addre
ss,uint256)._tokenA (SwapsRouter.sol#122) is not in mixedCase
Parameter SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,addre
ss,uint256)._tokenB (SwapsRouter.sol#123) is not in mixedCase
Parameter SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,addre
ss,uint256)._amountADesired (SwapsRouter.sol#124) is not in mixedCase
Parameter SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,addre
ss,uint256)._amountBDesired (SwapsRouter.sol#125) is not in mixedCase
Parameter SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,addre
ss,uint256)._amountAMin (SwapsRouter.sol#126) is not in mixedCase
Parameter SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,addre
ss,uint256)._amountBMin (SwapsRouter.sol#127) is not in mixedCase
Parameter SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,addre
ss,uint256)._to (SwapsRouter.sol#128) is not in mixedCase
Parameter SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,addre
ss,uint256)._deadline (SwapsRouter.sol#129) is not in mixedCase
Parameter
SwapsRouter.addLiquidityETH(address,uint256,uint256,uint256,address,uint256)._token
(SwapsRouter.sol#173) is not in mixedCase
Parameter SwapsRouter.addLiquidityETH(address,uint256,uint256,uint256,address,uint256)._
amountTokenDesired (SwapsRouter.sol#174) is not in mixedCase
Parameter SwapsRouter.addLiquidityETH(address,uint256,uint256,uint256,address,uint256)._
amountTokenMin (SwapsRouter.sol#175) is not in mixedCase
Parameter SwapsRouter.addLiquidityETH(address,uint256,uint256,uint256,address,uint256)._
amountETHMin (SwapsRouter.sol#176) is not in mixedCase
Parameter
SwapsRouter.addLiquidityETH(address,uint256,uint256,uint256,address,uint256)._to
(SwapsRouter.sol#177) is not in mixedCase
Parameter
SwapsRouter.addLiquidityETH(address,uint256,uint256,uint256,address,uint256)._deadline
(SwapsRouter.sol#178) is not in mixedCase
Parameter SwapsRouter.removeLiquidity(address,address,uint256,uint256,uint256,address,ui
nt256)._tokenA (SwapsRouter.sol#236) is not in mixedCase
Parameter SwapsRouter.removeLiquidity(address,address,uint256,uint256,uint256,address,ui
nt256)._tokenB (SwapsRouter.sol#237) is not in mixedCase
Parameter SwapsRouter.removeLiquidity(address,address,uint256,uint256,uint256,address,ui
nt256)._liquidity (SwapsRouter.sol#238) is not in mixedCase
Parameter SwapsRouter.removeLiquidity(address,address,uint256,uint256,uint256,address,ui
nt256)._amountAMin (SwapsRouter.sol#239) is not in mixedCase
Parameter SwapsRouter.removeLiquidity(address,address,uint256,uint256,uint256,address,ui
nt256)._amountBMin (SwapsRouter.sol#240) is not in mixedCase
Parameter SwapsRouter.removeLiquidity(address,address,uint256,uint256,uint256,address,ui
nt256)._to (SwapsRouter.sol#241) is not in mixedCase
Parameter SwapsRouter.removeLiquidity(address,address,uint256,uint256,uint256,address,ui
nt256)._deadline (SwapsRouter.sol#242) is not in mixedCase
Parameter
SwapsRouter.removeLiquidityETH(address,uint256,uint256,uint256,address,uint256)._token
(SwapsRouter.sol#292) is not in mixedCase
```

```
Parameter SwapsRouter.removeLiquidityETH(address,uint256,uint256,uint256,address,uint256
)._liquidity (SwapsRouter.sol#293) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETH(address,uint256,uint256,uint256,address,uint256
)._amountTokenMin (SwapsRouter.sol#294) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETH(address,uint256,uint256,uint256,address,uint256
)._amountETHMin (SwapsRouter.sol#295) is not in mixedCase
Parameter
SwapsRouter.removeLiquidityETH(address,uint256,uint256,uint256,address,uint256)._to
(SwapsRouter.sol#296) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETH(address,uint256,uint256,uint256,address,uint256
)._deadline (SwapsRouter.sol#297) is not in mixedCase
Parameter SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,
address,uint256,bool,uint8,bytes32,bytes32)._tokenA (SwapsRouter.sol#333) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,
address,uint256,bool,uint8,bytes32,bytes32)._tokenB (SwapsRouter.sol#334) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,
address,uint256,bool,uint8,bytes32,bytes32)._liquidity (SwapsRouter.sol#335) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,
address,uint256,bool,uint8,bytes32,bytes32)._amountAMin (SwapsRouter.sol#336) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,
address,uint256,bool,uint8,bytes32,bytes32)._amountBMin (SwapsRouter.sol#337) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,
address,uint256,bool,uint8,bytes32,bytes32)._to (SwapsRouter.sol#338) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,
address,uint256,bool,uint8,bytes32,bytes32)._deadline (SwapsRouter.sol#339) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,
address,uint256,bool,uint8,bytes32,bytes32)._approveMax (SwapsRouter.sol#340) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,
address,uint256,bool,uint8,bytes32,bytes32)._v (SwapsRouter.sol#341) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,
address,uint256,bool,uint8,bytes32,bytes32)._r (SwapsRouter.sol#342) is not in
mixedCase
```

```
Parameter SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,
address,uint256,bool,uint8,bytes32,bytes32)._s (SwapsRouter.sol#343) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermit(address,uint256,uint256,uint256,addre
ss,uint256,bool,uint8,bytes32,bytes32)._token (SwapsRouter.sol#381) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermit(address,uint256,uint256,uint256,addre
ss,uint256,bool,uint8,bytes32,bytes32)._liquidity (SwapsRouter.sol#382) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermit(address,uint256,uint256,uint256,addre
ss,uint256,bool,uint8,bytes32,bytes32)._amountTokenMin (SwapsRouter.sol#383) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermit(address,uint256,uint256,uint256,addre
ss,uint256,bool,uint8,bytes32,bytes32)._amountETHMin (SwapsRouter.sol#384) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermit(address,uint256,uint256,uint256,addre
ss,uint256,bool,uint8,bytes32,bytes32)._to (SwapsRouter.sol#385) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermit(address,uint256,uint256,uint256,addre
ss,uint256,bool,uint8,bytes32,bytes32)._deadline (SwapsRouter.sol#386) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermit(address,uint256,uint256,uint256,addre
ss,uint256,bool,uint8,bytes32,bytes32)._approveMax (SwapsRouter.sol#387) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermit(address,uint256,uint256,uint256,addre
ss,uint256,bool,uint8,bytes32,bytes32)._v (SwapsRouter.sol#388) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermit(address,uint256,uint256,uint256,addre
ss,uint256,bool,uint8,bytes32,bytes32)._r (SwapsRouter.sol#389) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermit(address,uint256,uint256,uint256,addre
ss,uint256,bool,uint8,bytes32,bytes32)._s (SwapsRouter.sol#390) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHSupportingFeeOnTransferTokens(address,uint256,ui
nt256,uint256,address,uint256)._token (SwapsRouter.sol#427) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHSupportingFeeOnTransferTokens(address,uint256,ui
nt256,uint256,address,uint256)._liquidity (SwapsRouter.sol#428) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHSupportingFeeOnTransferTokens(address,uint256,ui
nt256,uint256,address,uint256)._amountTokenMin (SwapsRouter.sol#429) is not in
mixedCase
Parameter SwapsRouter.removeLiquidityETHSupportingFeeOnTransferTokens(address,uint256,ui
nt256,uint256,address,uint256)._amountETHMin (SwapsRouter.sol#430) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHSupportingFeeOnTransferTokens(address,uint256,ui
nt256,uint256,address,uint256)._to (SwapsRouter.sol#431) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHSupportingFeeOnTransferTokens(address,uint256,ui
nt256,uint256,address,uint256)._deadline (SwapsRouter.sol#432) is not in mixedCase
```

```
Parameter SwapsRouter.removeLiquidityETHWithPermitSupportingFeeOnTransferTokens(address,
uint256,uint256,uint256,address,uint256,bool,uint8,bytes32,bytes32)._token
(SwapsRouter.sol#465) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermitSupportingFeeOnTransferTokens(address,
uint256,uint256,uint256,address,uint256,bool,uint8,bytes32,bytes32)._liquidity
(SwapsRouter.sol#466) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermitSupportingFeeOnTransferTokens(address,
uint256,uint256,uint256,address,uint256,bool,uint8,bytes32,bytes32)._amountTokenMin
(SwapsRouter.sol#467) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermitSupportingFeeOnTransferTokens(address,
uint256,uint256,uint256,address,uint256,bool,uint8,bytes32,bytes32)._amountETHMin
(SwapsRouter.sol#468) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermitSupportingFeeOnTransferTokens(address,
uint256,uint256,uint256,address,uint256,bool,uint8,bytes32,bytes32)._to
(SwapsRouter.sol#469) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermitSupportingFeeOnTransferTokens(address,
uint256,uint256,uint256,address,uint256,bool,uint8,bytes32,bytes32)._deadline
(SwapsRouter.sol#470) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermitSupportingFeeOnTransferTokens(address,
uint256,uint256,uint256,address,uint256,bool,uint8,bytes32,bytes32)._approveMax
(SwapsRouter.sol#471) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermitSupportingFeeOnTransferTokens(address,
uint256,uint256,uint256,address,uint256,bool,uint8,bytes32,bytes32)._v
(SwapsRouter.sol#472) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermitSupportingFeeOnTransferTokens(address,
uint256,uint256,uint256,address,uint256,bool,uint8,bytes32,bytes32)._r
(SwapsRouter.sol#473) is not in mixedCase
Parameter SwapsRouter.removeLiquidityETHWithPermitSupportingFeeOnTransferTokens(address,
uint256,uint256,uint256,address,uint256,bool,uint8,bytes32,bytes32)._s
(SwapsRouter.sol#474) is not in mixedCase
Parameter SwapsRouter.swapExactTokensForTokens(uint256,uint256,address[],address,uint256
)._amountIn (SwapsRouter.sol#556) is not in mixedCase
Parameter SwapsRouter.swapExactTokensForTokens(uint256,uint256,address[],address,uint256
)._amountOutMin (SwapsRouter.sol#557) is not in mixedCase
Parameter
SwapsRouter.swapExactTokensForTokens(uint256,uint256,address[],address,uint256)._path
(SwapsRouter.sol#558) is not in mixedCase
Parameter
SwapsRouter.swapExactTokensForTokens(uint256,uint256,address[],address,uint256)._to
(SwapsRouter.sol#559) is not in mixedCase
Parameter SwapsRouter.swapExactTokensForTokens(uint256,uint256,address[],address,uint256
)._deadline (SwapsRouter.sol#560) is not in mixedCase
Parameter SwapsRouter.swapTokensForExactTokens(uint256,uint256,address[],address,uint256
)._amountOut (SwapsRouter.sol#597) is not in mixedCase
Parameter SwapsRouter.swapTokensForExactTokens(uint256,uint256,address[],address,uint256
)._amountInMax (SwapsRouter.sol#598) is not in mixedCase
Parameter
SwapsRouter.swapTokensForExactTokens(uint256,uint256,address[],address,uint256)._path
(SwapsRouter.sol#599) is not in mixedCase
Parameter
SwapsRouter.swapTokensForExactTokens(uint256,uint256,address[],address,uint256)._to
(SwapsRouter.sol#600) is not in mixedCase
Parameter SwapsRouter.swapTokensForExactTokens(uint256,uint256,address[],address,uint256
)._deadline (SwapsRouter.sol#601) is not in mixedCase
Parameter
SwapsRouter.swapExactETHForTokens(uint256,address[],address,uint256)._amountOutMin
(SwapsRouter.sol#638) is not in mixedCase
Parameter SwapsRouter.swapExactETHForTokens(uint256,address[],address,uint256)._path
(SwapsRouter.sol#639) is not in mixedCase
Parameter SwapsRouter.swapExactETHForTokens(uint256,address[],address,uint256)._to
(SwapsRouter.sol#640) is not in mixedCase
Parameter
SwapsRouter.swapExactETHForTokens(uint256,address[],address,uint256)._deadline
(SwapsRouter.sol#641) is not in mixedCase
Parameter
SwapsRouter.swapTokensForExactETH(uint256,uint256,address[],address,uint256)._amountOut
(SwapsRouter.sol#688) is not in mixedCase
Parameter SwapsRouter.swapTokensForExactETH(uint256,uint256,address[],address,uint256)._
amountInMax (SwapsRouter.sol#689) is not in mixedCase
Parameter
SwapsRouter.swapTokensForExactETH(uint256,uint256,address[],address,uint256)._path
(SwapsRouter.sol#690) is not in mixedCase
Parameter
SwapsRouter.swapTokensForExactETH(uint256,uint256,address[],address,uint256)._to
(SwapsRouter.sol#691) is not in mixedCase
Parameter
SwapsRouter.swapTokensForExactETH(uint256,uint256,address[],address,uint256)._deadline
(SwapsRouter.sol#692) is not in mixedCase
Parameter
SwapsRouter.swapExactTokensForETH(uint256,uint256,address[],address,uint256)._amountIn
(SwapsRouter.sol#743) is not in mixedCase
Parameter SwapsRouter.swapExactTokensForETH(uint256,uint256,address[],address,uint256)._
```

```
amountOutMin (SwapsRouter.sol#744) is not in mixedCase
Parameter
SwapsRouter.swapExactTokensForETH(uint256,uint256,address[],address,uint256)._path
(SwapsRouter.sol#745) is not in mixedCase
Parameter
SwapsRouter.swapExactTokensForETH(uint256,uint256,address[],address,uint256)._to
(SwapsRouter.sol#746) is not in mixedCase
Parameter
SwapsRouter.swapExactTokensForETH(uint256,uint256,address[],address,uint256)._deadline
(SwapsRouter.sol#747) is not in mixedCase
Parameter
SwapsRouter.swapETHForExactTokens(uint256,address[],address,uint256)._amountOut
(SwapsRouter.sol#798) is not in mixedCase
Parameter SwapsRouter.swapETHForExactTokens(uint256,address[],address,uint256)._path
(SwapsRouter.sol#799) is not in mixedCase
Parameter SwapsRouter.swapETHForExactTokens(uint256,address[],address,uint256)._to
(SwapsRouter.sol#800) is not in mixedCase
Parameter
SwapsRouter.swapETHForExactTokens(uint256,address[],address,uint256)._deadline
(SwapsRouter.sol#801) is not in mixedCase
Parameter SwapsRouter.swapExactTokensForTokensSupportingFeeOnTransferTokens(uint256,uint
256,address[],address,uint256)._amountIn (SwapsRouter.sol#925) is not in mixedCase
Parameter SwapsRouter.swapExactTokensForTokensSupportingFeeOnTransferTokens(uint256,uint
256,address[],address,uint256)._amountOutMin (SwapsRouter.sol#926) is not in mixedCase
Parameter SwapsRouter.swapExactTokensForTokensSupportingFeeOnTransferTokens(uint256,uint
256,address[],address,uint256)._path (SwapsRouter.sol#927) is not in mixedCase
Parameter SwapsRouter.swapExactTokensForTokensSupportingFeeOnTransferTokens(uint256,uint
256,address[],address,uint256)._to (SwapsRouter.sol#928) is not in mixedCase
Parameter SwapsRouter.swapExactTokensForTokensSupportingFeeOnTransferTokens(uint256,uint
256,address[],address,uint256)._deadline (SwapsRouter.sol#929) is not in mixedCase
Parameter SwapsRouter.swapExactETHForTokensSupportingFeeOnTransferTokens(uint256,address
[],address,uint256)._amountOutMin (SwapsRouter.sol#960) is not in mixedCase
Parameter SwapsRouter.swapExactETHForTokensSupportingFeeOnTransferTokens(uint256,address
[],address,uint256)._path (SwapsRouter.sol#961) is not in mixedCase
Parameter SwapsRouter.swapExactETHForTokensSupportingFeeOnTransferTokens(uint256,address
[],address,uint256)._to (SwapsRouter.sol#962) is not in mixedCase
Parameter SwapsRouter.swapExactETHForTokensSupportingFeeOnTransferTokens(uint256,address
[],address,uint256)._deadline (SwapsRouter.sol#963) is not in mixedCase
Parameter SwapsRouter.swapExactTokensForETHSupportingFeeOnTransferTokens(uint256,uint256
,address[],address,uint256)._amountIn (SwapsRouter.sol#1008) is not in mixedCase
Parameter SwapsRouter.swapExactTokensForETHSupportingFeeOnTransferTokens(uint256,uint256
,address[],address,uint256)._amountOutMin (SwapsRouter.sol#1009) is not in mixedCase
Parameter SwapsRouter.swapExactTokensForETHSupportingFeeOnTransferTokens(uint256,uint256
,address[],address,uint256)._path (SwapsRouter.sol#1010) is not in mixedCase
Parameter SwapsRouter.swapExactTokensForETHSupportingFeeOnTransferTokens(uint256,uint256
,address[],address,uint256)._to (SwapsRouter.sol#1011) is not in mixedCase
Parameter SwapsRouter.swapExactTokensForETHSupportingFeeOnTransferTokens(uint256,uint256
,address[],address,uint256)._deadline (SwapsRouter.sol#1012) is not in mixedCase
Parameter SwapsRouter.pairFor(address,address,address)._factory (SwapsRouter.sol#1059)
is not in mixedCase
Parameter SwapsRouter.pairFor(address,address,address)._tokenA (SwapsRouter.sol#1060)
is not in mixedCase
Parameter SwapsRouter.pairFor(address,address,address)._tokenB (SwapsRouter.sol#1061)
is not in mixedCase
Parameter SwapsRouter.getAmountsOut(uint256,address[])._amountIn (SwapsRouter.sol#1076)
is not in mixedCase
Parameter SwapsRouter.getAmountsOut(uint256,address[])._path (SwapsRouter.sol#1077) is
not in mixedCase
Parameter SwapsRouter.getAmountsIn(uint256,address[])._amountOut (SwapsRouter.sol#1091)
is not in mixedCase
Parameter SwapsRouter.getAmountsIn(uint256,address[])._path (SwapsRouter.sol#1092) is
not in mixedCase
Parameter SwapsRouter.getReserves(address,address,address)._factory
(SwapsRouter.sol#1106) is not in mixedCase
Parameter SwapsRouter.getReserves(address,address,address)._tokenA
(SwapsRouter.sol#1107) is not in mixedCase
Parameter SwapsRouter.getReserves(address,address,address)._tokenB
(SwapsRouter.sol#1108) is not in mixedCase
Variable SwapsRouter.FACTORY (SwapsRouter.sol#13) is not in mixedCase
Variable SwapsRouter.WETH (SwapsRouter.sol#14) is not in mixedCase
Variable SwapsRouter.PAIR (SwapsRouter.sol#15) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-
solidity-naming-conventions

Variable ISwapsPair.swap(uint256,uint256,address,bytes)._amount0Out (ISwapsPair.sol#69)
is too similar to ISwapsPair.swap(uint256,uint256,address,bytes)._amount1Out
(ISwapsPair.sol#70)
Variable SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,addres
s,uint256)._amountADesired (SwapsRouter.sol#124) is too similar to SwapsRouter.addLiquid
ity(address,address,uint256,uint256,uint256,uint256,address,uint256)._amountBDesired
(SwapsRouter.sol#125)
Variable SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,a
ddress,uint256,bool,uint8,bytes32,bytes32)._amountAMin (SwapsRouter.sol#336) is too
similar to SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,addr
ess,uint256)._amountBMin (SwapsRouter.sol#127)
Variable
SwapsRouter._addLiquidity(address,address,uint256,uint256,uint256,uint256)._amountAMin
(SwapsRouter.sol#51) is too similar to
SwapsRouter._addLiquidity(address,address,uint256,uint256,uint256,uint256)._amountBMin
(SwapsRouter.sol#52)
Variable SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,a
ddress,uint256,bool,uint8,bytes32,bytes32)._amountAMin (SwapsRouter.sol#336) is too
similar to SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256
,address,uint256,bool,uint8,bytes32,bytes32)._amountBMin (SwapsRouter.sol#337)
Variable SwapsRouter.removeLiquidity(address,address,uint256,uint256,uint256,address,uin
t256)._amountAMin (SwapsRouter.sol#239) is too similar to SwapsRouter.removeLiquidity(ad
dress,address,uint256,uint256,uint256,address,uint256)._amountBMin
(SwapsRouter.sol#240)
Variable
SwapsRouter._addLiquidity(address,address,uint256,uint256,uint256,uint256)._amountAMin
(SwapsRouter.sol#51) is too similar to SwapsRouter.removeLiquidity(address,address,uint2
56,uint256,uint256,address,uint256)._amountBMin (SwapsRouter.sol#240)
Variable SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,addres
s,uint256)._amountAMin (SwapsRouter.sol#126) is too similar to SwapsRouter.removeLiquidi
ty(address,address,uint256,uint256,uint256,address,uint256)._amountBMin
(SwapsRouter.sol#240)
Variable
SwapsRouter._addLiquidity(address,address,uint256,uint256,uint256,uint256)._amountAMin
(SwapsRouter.sol#51) is too similar to SwapsRouter.addLiquidity(address,address,uint256,
uint256,uint256,uint256,address,uint256)._amountBMin (SwapsRouter.sol#127)
Variable SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,addres
s,uint256)._amountAMin (SwapsRouter.sol#126) is too similar to SwapsRouter.addLiquidity(
address,address,uint256,uint256,uint256,uint256,address,uint256)._amountBMin
(SwapsRouter.sol#127)
Variable
SwapsRouter._addLiquidity(address,address,uint256,uint256,uint256,uint256)._amountAMin
(SwapsRouter.sol#51) is too similar to SwapsRouter.removeLiquidityWithPermit(address,add
ress,uint256,uint256,uint256,address,uint256,bool,uint8,bytes32,bytes32)._amountBMin
(SwapsRouter.sol#337)
Variable SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,a
ddress,uint256,bool,uint8,bytes32,bytes32)._amountAMin (SwapsRouter.sol#336) is too
similar to SwapsRouter.removeLiquidity(address,address,uint256,uint256,uint256,address,u
int256)._amountBMin (SwapsRouter.sol#240)
```

```
Variable SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256)._amountADesired (SwapsRouter.sol#124) is too similar to SwapsRouter._addLiquidity(address,address,uint256,uint256,uint256,uint256)._amountBDesired (SwapsRouter.sol#50)
Variable SwapsRouter._addLiquidity(address,address,uint256,uint256,uint256,uint256)._amountADesired (SwapsRouter.sol#49) is too similar to SwapsRouter._addLiquidity(address,address,uint256,uint256,uint256,uint256)._amountBDesired (SwapsRouter.sol#50)
Variable SwapsRouter._addLiquidity(address,address,uint256,uint256,uint256,uint256)._amountADesired (SwapsRouter.sol#49) is too similar to SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256)._amountBDesired (SwapsRouter.sol#125)
Variable SwapsRouter.removeLiquidity(address,address,uint256,uint256,uint256,address,uint256)._amountAMin (SwapsRouter.sol#239) is too similar to SwapsRouter._addLiquidity(address,address,uint256,uint256,uint256,uint256)._amountBMin (SwapsRouter.sol#52)
Variable SwapsRouter.removeLiquidity(address,address,uint256,uint256,uint256,address,uint256)._amountAMin (SwapsRouter.sol#239) is too similar to SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256)._amountBMin (SwapsRouter.sol#127)
Variable SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256)._amountAMin (SwapsRouter.sol#126) is too similar to SwapsRouter._addLiquidity(address,address,uint256,uint256,uint256,uint256)._amountBMin (SwapsRouter.sol#52)
Variable SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,address,uint256,bool,uint8,bytes32,bytes32)._amountAMin (SwapsRouter.sol#336) is too similar to SwapsRouter._addLiquidity(address,address,uint256,uint256,uint256,uint256)._amountBMin (SwapsRouter.sol#52)
Variable SwapsRouter.removeLiquidity(address,address,uint256,uint256,uint256,address,uint256)._amountAMin (SwapsRouter.sol#239) is too similar to SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,address,uint256,bool,uint8,bytes32,bytes32)._amountBMin (SwapsRouter.sol#337)
Variable SwapsRouter.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256)._amountAMin (SwapsRouter.sol#126) is too similar to SwapsRouter.removeLiquidityWithPermit(address,address,uint256,uint256,uint256,address,uint256,bool,uint8,bytes32,bytes32)._amountBMin (SwapsRouter.sol#337)
Variable SwapsRouter._addLiquidity(address,address,uint256,uint256,uint256,uint256).amountAOptimal (SwapsRouter.sol#100-104) is too similar to SwapsRouter._addLiquidity(address,address,uint256,uint256,uint256,uint256).amountBOptimal (SwapsRouter.sol#81-85)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-are-too-similar
```

```
SwapsHelper._pairFor(address,address,address,address) (SwapsHelper.sol#197-228) uses
literals with too many digits:
        - mstore(uint256,uint256)
(ptr__pairFor_asm_0,0x3d602d80600a3d3981f3363d3d373d3d3d363d73000000000000000000000000)
(SwapsHelper.sol#220)
SwapsHelper._pairFor(address,address,address,address) (SwapsHelper.sol#197-228) uses
literals with too many digits:
        - mstore(uint256,uint256)(ptr__pairFor_asm_0 +
0x28,0x5af43d82803e903d91602b57fd5bf3ff0000000000000000000000000000000000)
(SwapsHelper.sol#222)
RouterCodeCheck.routerCodeHash() (SwapsRouter.sol#1223-1231) uses literals with too
many digits:
        - keccak256(bytes)(type()(SwapsRouter).creationCode)
(SwapsRouter.sol#1228-1230)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#too-many-
digits

Pragma version^0.8.9 (ISwapsERC20.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
solc-0.8.11 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-
versions-of-solidity

Function ISwapsERC20.DOMAIN_SEPARATOR() (ISwapsERC20.sol#64-67) is not in mixedCase
Function ISwapsERC20.PERMIT_TYPEHASH() (ISwapsERC20.sol#69-72) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-
solidity-naming-conventions

Pragma version^0.8.9 (ISwapsERC20.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (ISwapsPair.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
solc-0.8.11 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-
versions-of-solidity

Function ISwapsERC20.DOMAIN_SEPARATOR() (ISwapsERC20.sol#64-67) is not in mixedCase
Function ISwapsERC20.PERMIT_TYPEHASH() (ISwapsERC20.sol#69-72) is not in mixedCase
Function ISwapsPair.MINIMUM_LIQUIDITY() (ISwapsPair.sol#9-12) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-
```

```
solidity-naming-conventions

Variable ISwapsPair.swap(uint256,uint256,address,bytes)._amount0Out (ISwapsPair.sol#69)
is too similar to ISwapsPair.swap(uint256,uint256,address,bytes)._amount1Out
(ISwapsPair.sol#70)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-
are-too-similar

Pragma version^0.8.9 (IWETH.sol#3) necessitates a version too recent to be trusted.
Consider deploying with 0.6.12/0.7.6/0.8.7
solc-0.8.11 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-
versions-of-solidity

SwapsPair._update(uint256,uint256,uint112,uint112) (SwapsPair.sol#104-135) uses a weak
PRNG: "blockTimestamp = uint32(block.timestamp % 2 ** 32) (SwapsPair.sol#118)"
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#weak-PRNG

SwapsPair._mintFee(uint112,uint112,uint256) (SwapsPair.sol#137-162) uses a dangerous
strict equality:
        - _kLast == 0 (SwapsPair.sol#144)
SwapsPair._safeTransfer(address,address,uint256) (SwapsPair.sol#483-506) uses a
dangerous strict equality:
        - require(bool,string)(success == true && (data.length == 0 || abi.decode(data,
(bool))),SwapsPair: TRANSFER_FAILED) (SwapsPair.sol#498-505)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dangerous-
strict-equalities

Reentrancy in SwapsPair.burn(address) (SwapsPair.sol#236-312):
        External calls:
        - _safeTransfer(_token0,_to,amount0) (SwapsPair.sol#282-286)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - _safeTransfer(_token1,_to,amount1) (SwapsPair.sol#288-292)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        State variables written after the call(s):
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#297-302)
                - blockTimestampLast = blockTimestamp (SwapsPair.sol#129)
        - kLast = uint256(reserve0) * reserve1 (SwapsPair.sol#304)
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#297-302)
```

```
                - reserve0 = uint112(_balance0) (SwapsPair.sol#126)
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#297-302)
                - reserve1 = uint112(_balance1) (SwapsPair.sol#127)
Reentrancy in SwapsPair.swap(uint256,uint256,address,bytes) (SwapsPair.sol#314-403):
        External calls:
        - _safeTransfer(_token0,_to,_amount0Out) (SwapsPair.sol#348)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - _safeTransfer(_token1,_to,_amount1Out) (SwapsPair.sol#349)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - ISwapsCallee(_to).swapsCall(msg.sender,_amount0Out,_amount1Out,_data)
(SwapsPair.sol#351-356)
        State variables written after the call(s):
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#388-393)
                - blockTimestampLast = blockTimestamp (SwapsPair.sol#129)
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#388-393)
                - reserve0 = uint112(_balance0) (SwapsPair.sol#126)
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#388-393)
                - reserve1 = uint112(_balance1) (SwapsPair.sol#127)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-
vulnerabilities-1

SwapsPair.initialize(address,address)._token0 (SwapsPair.sol#72) lacks a zero-check
on :
                - token0 = _token0 (SwapsPair.sol#82)
SwapsPair.initialize(address,address)._token1 (SwapsPair.sol#73) lacks a zero-check
on :
                - token1 = _token1 (SwapsPair.sol#83)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-zero-
address-validation

Reentrancy in SwapsPair.burn(address) (SwapsPair.sol#236-312):
        External calls:
        - _safeTransfer(_token0,_to,amount0) (SwapsPair.sol#282-286)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - _safeTransfer(_token1,_to,amount1) (SwapsPair.sol#288-292)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        State variables written after the call(s):
```

```
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#297-302)
                - price0CumulativeLast += uint256(uqdiv(encode(_reserve1),_reserve0)) *
timeElapsed (SwapsPair.sol#122)
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#297-302)
                - price1CumulativeLast += uint256(uqdiv(encode(_reserve0),_reserve1)) *
timeElapsed (SwapsPair.sol#123)
Reentrancy in SwapsPair.swap(uint256,uint256,address,bytes) (SwapsPair.sol#314-403):
        External calls:
        - _safeTransfer(_token0,_to,_amount0Out) (SwapsPair.sol#348)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - _safeTransfer(_token1,_to,_amount1Out) (SwapsPair.sol#349)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - ISwapsCallee(_to).swapsCall(msg.sender,_amount0Out,_amount1Out,_data)
(SwapsPair.sol#351-356)
        State variables written after the call(s):
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#388-393)
                - price0CumulativeLast += uint256(uqdiv(encode(_reserve1),_reserve0)) *
timeElapsed (SwapsPair.sol#122)
        - _update(balance0,balance1,_reserve0,_reserve1) (SwapsPair.sol#388-393)
                - price1CumulativeLast += uint256(uqdiv(encode(_reserve0),_reserve1)) *
timeElapsed (SwapsPair.sol#123)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-
vulnerabilities-2

Reentrancy in SwapsPair.burn(address) (SwapsPair.sol#236-312):
        External calls:
        - _safeTransfer(_token0,_to,amount0) (SwapsPair.sol#282-286)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - _safeTransfer(_token1,_to,amount1) (SwapsPair.sol#288-292)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        Event emitted after the call(s):
        - Burn(msg.sender,amount0,amount1,_to) (SwapsPair.sol#306-311)
        - Sync(reserve0,reserve1) (SwapsPair.sol#131-134)
                - _update(balance0,balance1,_reserve0,_reserve1)
(SwapsPair.sol#297-302)
Reentrancy in SwapsPair.swap(uint256,uint256,address,bytes) (SwapsPair.sol#314-403):
        External calls:
```

```
        - _safeTransfer(_token0,_to,_amount0Out) (SwapsPair.sol#348)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - _safeTransfer(_token1,_to,_amount1Out) (SwapsPair.sol#349)
                - (success,data) =
_token.call(abi.encodeWithSelector(SELECTOR,_to,_value)) (SwapsPair.sol#490-496)
        - ISwapsCallee(_to).swapsCall(msg.sender,_amount0Out,_amount1Out,_data)
(SwapsPair.sol#351-356)
        Event emitted after the call(s):
        - Swap(msg.sender,_amount0In,_amount1In,_amount0Out,_amount1Out,_to)
(SwapsPair.sol#395-402)
        - Sync(reserve0,reserve1) (SwapsPair.sol#131-134)
                - _update(balance0,balance1,_reserve0,_reserve1)
(SwapsPair.sol#388-393)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-
vulnerabilities-3

SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)
(SwapsERC20.sol#182-233) uses timestamp for comparisons
        Dangerous comparisons:
        - require(bool,string)(_deadline >= block.timestamp,PERMIT_CALL_EXPIRED)
(SwapsERC20.sol#193-196)
SwapsPair._update(uint256,uint256,uint112,uint112) (SwapsPair.sol#104-135) uses
timestamp for comparisons
        Dangerous comparisons:
        - timeElapsed > 0 && _reserve0 != 0 && _reserve1 != 0 (SwapsPair.sol#121)
SwapsPair.swap(uint256,uint256,address,bytes) (SwapsPair.sol#314-403) uses timestamp
for comparisons
        Dangerous comparisons:
        - require(bool,string)(_amount0Out < _reserve0 && _amount1Out <
_reserve1,INSUFFICIENT_LIQUIDITY) (SwapsPair.sol#335-339)
        - require(bool,string)(_amount0In > 0 || _amount1In >
0,INSUFFICIENT_INPUT_AMOUNT) (SwapsPair.sol#370-374)
        - require(bool)(balance0Adjusted * balance1Adjusted >= uint256(_reserve0) *
_reserve1 * (1000 ** 2)) (SwapsPair.sol#380-385)
        - balance0 > _reserve0 - _amount0Out (SwapsPair.sol#362-364)
        - balance1 > _reserve1 - _amount1Out (SwapsPair.sol#366-368)
SwapsPair.min(uint256,uint256) (SwapsPair.sol#451-460) uses timestamp for comparisons
        Dangerous comparisons:
        - _x < _y (SwapsPair.sol#459)
SwapsPair.sqrt(uint256) (SwapsPair.sol#462-481) uses timestamp for comparisons
```

```
        Dangerous comparisons:
        - _y > 3 (SwapsPair.sol#470)
        - x < z (SwapsPair.sol#473)
        - _y != 0 (SwapsPair.sol#477)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#block-
timestamp

SwapsPair._safeTransfer(address,address,uint256) (SwapsPair.sol#483-506) compares to a
boolean constant:
        -require(bool,string)(success == true && (data.length == 0 || abi.decode(data,
(bool))),SwapsPair: TRANSFER_FAILED) (SwapsPair.sol#498-505)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#boolean-
equality

Pragma version^0.8.9 (IERC20.sol#3) necessitates a version too recent to be trusted.
Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (ISwapsCallee.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (ISwapsFactory.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (SwapsERC20.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
Pragma version^0.8.9 (SwapsPair.sol#3) necessitates a version too recent to be trusted.
Consider deploying with 0.6.12/0.7.6/0.8.7
solc-0.8.11 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-
versions-of-solidity

Low level call in SwapsPair._safeTransfer(address,address,uint256)
(SwapsPair.sol#483-506):
        - (success,data) = _token.call(abi.encodeWithSelector(SELECTOR,_to,_value))
(SwapsPair.sol#490-496)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#low-level-
calls

Parameter SwapsERC20.approve(address,uint256)._spender (SwapsERC20.sol#130) is not in
mixedCase
Parameter SwapsERC20.approve(address,uint256)._value (SwapsERC20.sol#131) is not in
mixedCase
Parameter SwapsERC20.transfer(address,uint256)._to (SwapsERC20.sol#146) is not in
mixedCase
```

```
Parameter SwapsERC20.transfer(address,uint256)._value (SwapsERC20.sol#147) is not in
mixedCase
Parameter SwapsERC20.transferFrom(address,address,uint256)._from (SwapsERC20.sol#162)
is not in mixedCase
Parameter SwapsERC20.transferFrom(address,address,uint256)._to (SwapsERC20.sol#163) is
not in mixedCase
Parameter SwapsERC20.transferFrom(address,address,uint256)._value (SwapsERC20.sol#164)
is not in mixedCase
Parameter
SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._owner
(SwapsERC20.sol#183) is not in mixedCase
Parameter
SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._spender
(SwapsERC20.sol#184) is not in mixedCase
Parameter
SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._value
(SwapsERC20.sol#185) is not in mixedCase
Parameter
SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._deadline
(SwapsERC20.sol#186) is not in mixedCase
Parameter SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._v
(SwapsERC20.sol#187) is not in mixedCase
Parameter SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._r
(SwapsERC20.sol#188) is not in mixedCase
Parameter SwapsERC20.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)._s
(SwapsERC20.sol#189) is not in mixedCase
Variable SwapsERC20.DOMAIN_SEPARATOR (SwapsERC20.sol#20) is not in mixedCase
Parameter SwapsPair.initialize(address,address)._token0 (SwapsPair.sol#72) is not in
mixedCase
Parameter SwapsPair.initialize(address,address)._token1 (SwapsPair.sol#73) is not in
mixedCase
Parameter SwapsPair.mint(address)._to (SwapsPair.sol#165) is not in mixedCase
Parameter SwapsPair.burn(address)._to (SwapsPair.sol#237) is not in mixedCase
Parameter SwapsPair.swap(uint256,uint256,address,bytes)._amount0Out (SwapsPair.sol#315)
is not in mixedCase
Parameter SwapsPair.swap(uint256,uint256,address,bytes)._amount1Out (SwapsPair.sol#316)
is not in mixedCase
Parameter SwapsPair.swap(uint256,uint256,address,bytes)._to (SwapsPair.sol#317) is not
in mixedCase
Parameter SwapsPair.swap(uint256,uint256,address,bytes)._data (SwapsPair.sol#318) is
not in mixedCase
```

```
Parameter SwapsPair.encode(uint112)._y (SwapsPair.sol#427) is not in mixedCase
Parameter SwapsPair.uqdiv(uint224,uint112)._x (SwapsPair.sol#439) is not in mixedCase
Parameter SwapsPair.uqdiv(uint224,uint112)._y (SwapsPair.sol#440) is not in mixedCase
Parameter SwapsPair.min(uint256,uint256)._x (SwapsPair.sol#452) is not in mixedCase
Parameter SwapsPair.min(uint256,uint256)._y (SwapsPair.sol#453) is not in mixedCase
Parameter SwapsPair.sqrt(uint256)._y (SwapsPair.sol#463) is not in mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-to-
solidity-naming-conventions

Variable SwapsPair.swap(uint256,uint256,address,bytes)._amount0Out (SwapsPair.sol#315)
is too similar to SwapsPair.swap(uint256,uint256,address,bytes)._amount1Out
(SwapsPair.sol#316)
Variable SwapsPair.swap(uint256,uint256,address,bytes).balance0Adjusted
(SwapsPair.sol#377) is too similar to
SwapsPair.swap(uint256,uint256,address,bytes).balance1Adjusted (SwapsPair.sol#378)
Variable SwapsPair.price0CumulativeLast (SwapsPair.sol#28) is too similar to
SwapsPair.price1CumulativeLast (SwapsPair.sol#29)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-
are-too-similar

Pragma version^0.8.9 (IERC20.sol#3) necessitates a version too recent to be trusted.
Consider deploying with 0.6.12/0.7.6/0.8.7
solc-0.8.11 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-
versions-of-solidity

Pragma version^0.8.9 (ISwapsCallee.sol#3) necessitates a version too recent to be
trusted. Consider deploying with 0.6.12/0.7.6/0.8.7
solc-0.8.11 is not recommended for deployment
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-
versions-of-solidity
. analyzed (31 contracts with 77 detectors), 387 result(s) found
```

0x Guard