Title not found

Updated as of block [20843568](#) at 9/27/2024, 1:51:23 PM ET

- ID: 25
- Proposer: `0x7dCaCF417BA662840DcD2A35b67f55911815dD7e`
- Start Block: 20826607 (9/25/2024, 5:03:59 AM ET)
- End Block: 20843887 (9/27/2024, 3:00:30 PM ET)
- Targets: `0x31c8EAcBFFdD875c74b94b077895Bd78CF1E64A3` ; `0x8dA8f82d2BbDd896822de723F55D6EdF416130ba` ; `0xD64D01D04498bFc60f04178e0B62a757C5048212`

Table of contents

Proposal Text

> *[RGP-17] Upgrade Governance Contracts to OZ Governor [September 24 Cycle]*

Checks

Reports all state changes from the proposal ✅ Passed

Info:

- RadicleToken (Radworks) at `0x31c8EAcBFFdD875c74b94b077895Bd78CF1E64A3`
  - `balances` key `0x5c04e7808455ee0e22c2773328c151d0dd79dc62` changed from `"0"` to `"8065000000000000000000"`
  - `balances` key `0x8da8f82d2bbdd896822de723f55d6edf416130ba` changed from `"48235149606849567167202010"` to `"48227084606849567167202010"`
- Timelock at `0x8dA8f82d2BbDd896822de723F55D6EdF416130ba`
  - `admin` changed from `0x690e775361ad66d1c4a25d89da9fcd639f5198ed` to `0xd64d01d04498bfc60f04178e0b62a757c5048212`

Decodes target calldata into a human-readable format ✅ Passed

Info:

- On contract `0x31c8EAcBFFdD875c74b94b077895Bd78CF1E64A3`, call `transfer(address dst, uint256 rawAmount)(bool)` with arguments `dst=0x5c04e7808455ee0e22c2773328c151d0dd79dc62` , `rawAmount=80650000000000000000000` (generic)

- On contract `0x8dA8f82d2BbDd896822de723F55D6EdF416130ba` , call `setPendingAdmin(address pendingAdmin_)()` with arguments `pendingAdmin_=0xd64d01d04498bfc60f04178e0b62a757c5048212` (generic)
- On contract `0xD64D01D04498bFc60f04178e0B62a757C5048212` , call `0xb9a61961` (not decoded)

Reports all events emitted from the proposal ✅ Passed

Info:

- RadicleToken (Radworks) at `0x31c8EAcBFFdD875c74b94b077895Bd78CF1E64A3`
  - `Transfer(from: 0x8da8f82d2bbdd896822de723f55d6edf416130ba, to: 0x5c04e7808455ee0e22c2773328c151d0dd79dc62, amount: 80650000000000000000000)`
- Timelock at `0x8dA8f82d2BbDd896822de723F55D6EdF416130ba`
  - `NewPendingAdmin(newPendingAdmin: 0xd64d01d04498bfc60f04178e0b62a757c5048212)`
  - `NewAdmin(newAdmin: 0xd64d01d04498bfc60f04178e0b62a757c5048212)`

Check all targets are verified on Etherscan ✅ Passed

Info:

- `0x31c8EAcBFFdD875c74b94b077895Bd78CF1E64A3` : Contract (verified)
- `0x8dA8f82d2BbDd896822de723F55D6EdF416130ba` : Contract (verified)
- `0xD64D01D04498bFc60f04178e0B62a757C5048212` : Contract (verified)

Check all touched contracts are verified on Etherscan ✅ Passed

Info:

- `0xD73a92Be73EfbFcF3854433A5FcbAbF9c1316073` : EOA (verification not applicable)
- `0x690e775361AD66D1c4A25d89da9fCd639F5198eD` : Contract (verified)
- `0x8dA8f82d2BbDd896822de723F55D6EdF416130ba` : Contract (verified)
- `0x31c8EAcBFFdD875c74b94b077895Bd78CF1E64A3` : Contract (verified)
- `0xD64D01D04498bFc60f04178e0B62a757C5048212` : Contract (verified)

Check all targets do not contain selfdestruct ✅ Passed

Info:

- `0x31c8EAcBFFdD875c74b94b077895Bd78CF1E64A3` : Contract (looks safe)
- `0x8dA8f82d2BbDd896822de723F55D6EdF416130ba` : Trusted contract (not checked)
- `0xD64D01D04498bFc60f04178e0B62a757C5048212` : Contract (looks safe)

Check all touched contracts do not contain selfdestruct ❗❗ Passed with warnings

Warnings:

- `0xD73a92Be73EfbFcF3854433A5FcbAbF9c1316073` : EOA (may have code later)

Info:

- `0x690e775361AD66D1c4A25d89da9fCd639F5198eD` : Trusted contract (not checked)

- `0x8dA8f82d2BbDd896822de723F55D6EdF416130ba` : Trusted contract (not checked)
- `0x31c8EAcBFFdD875c74b94b077895Bd78CF1E64A3` : Contract (looks safe)
- `0xD64D01D04498bFc60f04178e0B62a757C5048212` : Contract (looks safe)

Reports on whether the caller needs to send ETH with the call ✅ Passed

Info:

No ETH is required to be sent by the account that executes this proposal.

Runs solc against the verified contracts ✅ Passed

Info:

Compiler warnings for RadicleToken (Radworks) at `0x31c8EAcBFFdD875c74b94b077895Bd78CF1E64A3`

```
INFO:CryticCompile:'solc --standard-json --allow-paths
/Users/gargoyles/Development/governance-seatbelt' running
ERROR:CryticCompile:ParserError: ParserError: Source file requires different compiler
version (current compiler is 0.8.27+commit.40a35a09.Darwin.appleclang) - note that
nightly builds are considered to be strictly less than the released version
  --> crytic-export/etherscan-contracts/0x31c8EAcBFFdD875c74b94b077895Bd78CF1E64A3-
RadicleToken.sol:28:1:
   |
28 | pragma solidity ^0.7.5;
   | ^^^^^^^^^^^^^^^^^^^^^^^
```

Compiler warnings for RadworksGovernor at `0xD64D01D04498bFc60f04178e0B62a757C5048212`

```
INFO:CryticCompile:'solc --standard-json --allow-paths
/Users/gargoyles/Development/governance-seatbelt/crytic-export/etherscan-
contracts/0xD64D01D04498bFc60f04178e0B62a757C5048212-RadworksGovernor' running
```

Runs slither against the verified contracts ✅ Passed

Info:

Slither report for RadicleToken (Radworks) at `0x31c8EAcBFFdD875c74b94b077895Bd78CF1E64A3`

```
'solc --standard-json --allow-paths /Users/gargoyles/Development/governance-seatbelt'
running
Traceback (most recent call last):
  File "/opt/homebrew/bin/slither", line 8, in <module>
    sys.exit(main())
             ^^^^^^
  File "/opt/homebrew/Cellar/slither-analyzer/0.10.4/libexec/lib/python3.12/site-
packages/slither/__main__.py", line 776, in main
    main_impl(all_detector_classes=detectors, all_printer_classes=printers)
  File "/opt/homebrew/Cellar/slither-analyzer/0.10.4/libexec/lib/python3.12/site-
packages/slither/__main__.py", line 882, in main_impl
    ) = process_all(filename, args, detector_classes, printer_classes)
        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/homebrew/Cellar/slither-analyzer/0.10.4/libexec/lib/python3.12/site-
packages/slither/__main__.py", line 96, in process_all
```

```
    compilations = compile_all(target, **vars(args))
                   ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/homebrew/Cellar/slither-analyzer/0.10.4/libexec/lib/python3.12/site-packages/crytic_compile/crytic_compile.py", line 722, in compile_all
    compilations.append(CryticCompile(target, **kwargs))
                        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/homebrew/Cellar/slither-analyzer/0.10.4/libexec/lib/python3.12/site-packages/crytic_compile/crytic_compile.py", line 211, in __init__
    self._compile(**kwargs)
  File "/opt/homebrew/Cellar/slither-analyzer/0.10.4/libexec/lib/python3.12/site-packages/crytic_compile/crytic_compile.py", line 633, in _compile
    self._platform.compile(self, **kwargs)
  File "/opt/homebrew/Cellar/slither-analyzer/0.10.4/libexec/lib/python3.12/site-packages/crytic_compile/platform/etherscan.py", line 423, in compile
    solc_standard_json.standalone_compile(
  File "/opt/homebrew/Cellar/slither-analyzer/0.10.4/libexec/lib/python3.12/site-packages/crytic_compile/platform/solc_standard_json.py", line 85, in standalone_compile
    targets_json = run_solc_standard_json(
                   ^^^^^^^^^^^^^^^^^^^^^^^
  File "/opt/homebrew/Cellar/slither-analyzer/0.10.4/libexec/lib/python3.12/site-packages/crytic_compile/platform/solc_standard_json.py", line 204, in run_solc_standard_json
    raise InvalidCompilation(solc_exception_str)
crytic_compile.platform.exceptions.InvalidCompilation: ParserError: ParserError: Source file requires different compiler version (current compiler is 0.8.27+commit.40a35a09.Darwin.appleclang) - note that nightly builds are considered to be strictly less than the released version
  --> crytic-export/etherscan-contracts/0x31c8EAcBFFdD875c74b94b077895Bd78CF1E64A3-RadicleToken.sol:28:1:
   |
28 | pragma solidity ^0.7.5;
   | ^^^^^^^^^^^^^^^^^^^^^^^
```

Slither report for RadworksGovernor at `0xD64D01D04498bFc60f04178e0B62a757C5048212`

```
'solc --standard-json --allow-paths /Users/gargoyles/Development/governance-seatbelt/crytic-export/etherscan-contracts/0xD64D01D04498bFc60f04178e0B62a757C5048212-RadworksGovernor' running
INFO:Detectors:
Governor._execute(uint256,address[],uint256[],bytes[],bytes32) (lib/openzeppelin-contracts/contracts/governance/Governor.sol#363-375) sends eth to arbitrary user
    Dangerous calls:
    - (success,returndata) = targets[i].call{value: values[i]}(calldatas[i]) (lib/openzeppelin-contracts/contracts/governance/Governor.sol#372)
Governor.relay(address,uint256,bytes) (lib/openzeppelin-contracts/contracts/governance/Governor.sol#593-596) sends eth to arbitrary user
    Dangerous calls:
    - (success,returndata) = target.call{value: value}(data) (lib/openzeppelin-contracts/contracts/governance/Governor.sol#594)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#functions-that-send-ether-to-arbitrary-destinations
```

```
INFO:Detectors:
Math.mulDiv(uint256,uint256,uint256) (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#55-134) has bitwise-xor operator ^ instead of
the exponentiation operator **:
     - inverse = (3 * denominator) ^ 2 (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#116)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-
exponentiation
INFO:Detectors:
Governor._name (lib/openzeppelin-contracts/contracts/governance/Governor.sol#50)
shadows:
    - EIP712._name (lib/openzeppelin-
contracts/contracts/utils/cryptography/EIP712.sol#50)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-
variable-shadowing
INFO:Detectors:
Math.mulDiv(uint256,uint256,uint256) (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#55-134) performs a multiplication on the
result of a division:
    - denominator = denominator / twos (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#101)
    - inverse = (3 * denominator) ^ 2 (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#116)
Math.mulDiv(uint256,uint256,uint256) (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#55-134) performs a multiplication on the
result of a division:
    - denominator = denominator / twos (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#101)
    - inverse *= 2 - denominator * inverse (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#120)
Math.mulDiv(uint256,uint256,uint256) (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#55-134) performs a multiplication on the
result of a division:
    - denominator = denominator / twos (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#101)
    - inverse *= 2 - denominator * inverse (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#121)
Math.mulDiv(uint256,uint256,uint256) (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#55-134) performs a multiplication on the
result of a division:
    - denominator = denominator / twos (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#101)
    - inverse *= 2 - denominator * inverse (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#122)
Math.mulDiv(uint256,uint256,uint256) (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#55-134) performs a multiplication on the
result of a division:
    - denominator = denominator / twos (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#101)
    - inverse *= 2 - denominator * inverse (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#123)
Math.mulDiv(uint256,uint256,uint256) (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#55-134) performs a multiplication on the
result of a division:
    - denominator = denominator / twos (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#101)
    - inverse *= 2 - denominator * inverse (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#124)
Math.mulDiv(uint256,uint256,uint256) (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#55-134) performs a multiplication on the
result of a division:
    - denominator = denominator / twos (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#101)
    - inverse *= 2 - denominator * inverse (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#125)
Math.mulDiv(uint256,uint256,uint256) (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#55-134) performs a multiplication on the
result of a division:
    - prod0 = prod0 / twos (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#104)
    - result = prod0 * inverse (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#131)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#divide-
before-multiply
INFO:Detectors:
ERC20Votes.CLOCK_MODE() (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#51-55) uses a dangerous
strict equality:
    - require(bool,string)(clock() == block.number,ERC20Votes: broken clock mode)
(lib/openzeppelin-contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#53)
ERC20Votes._writeCheckpoint(ERC20Votes.Checkpoint[],function(uint256,uint256)
returns(uint256),uint256) (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#252-271) uses a dangerous
strict equality:
    - pos > 0 && oldCkpt.fromBlock == clock() (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#265)
Governor.state(uint256) (lib/openzeppelin-
contracts/contracts/governance/Governor.sol#163-197) uses a dangerous strict equality:
    - snapshot == 0 (lib/openzeppelin-contracts/contracts/governance/Governor.sol#176)
GovernorTimelockCompound.state(uint256) (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#51-66) uses a
dangerous strict equality:
    - eta == 0 (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#59)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dangerous-
strict-equalities
INFO:Detectors:
Governor._execute(uint256,address[],uint256[],bytes[],bytes32) (lib/openzeppelin-
contracts/contracts/governance/Governor.sol#363-375) ignores return value by
Address.verifyCallResult(success,returndata,errorMessage) (lib/openzeppelin-
contracts/contracts/governance/Governor.sol#373)
Governor.relay(address,uint256,bytes) (lib/openzeppelin-
contracts/contracts/governance/Governor.sol#593-596) ignores return value by
Address.verifyCallResult(success,returndata,Governor: relay reverted without message)
```

```
(lib/openzeppelin-contracts/contracts/governance/Governor.sol#595)
GovernorTimelockCompound.queue(address[],uint256[],bytes[],bytes32) (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#85-109) ignores
return value by _timelock.queueTransaction(targets[i],values[i],,calldatas[i],eta)
(lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#103)
GovernorTimelockCompound._execute(uint256,address[],uint256[],bytes[],bytes32)
(lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#114-127)
ignores return value by
_timelock.executeTransaction(targets[i],values[i],,calldatas[i],eta)
(lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#125)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unused-return
INFO:Detectors:
ERC20Permit.constructor(string).name (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Permit.sol#44) shadows:
    - ERC20.name() (lib/openzeppelin-contracts/contracts/token/ERC20/ERC20.sol#62-64)
(function)
    - IERC20Metadata.name() (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/IERC20Metadata.sol#17) (function)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#local-
variable-shadowing
INFO:Detectors:
Governor.relay(address,uint256,bytes).target (lib/openzeppelin-
contracts/contracts/governance/Governor.sol#593) lacks a zero-check on :
        - (success,returndata) = target.call{value: value}(data) (lib/openzeppelin-
contracts/contracts/governance/Governor.sol#594)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#missing-zero-
address-validation
INFO:Detectors:
GovernorTimelockCompound.queue(address[],uint256[],bytes[],bytes32) (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#85-109) has
external calls inside a loop: require(bool,string)(!
_timelock.queuedTransactions(keccak256(bytes)
(abi.encode(targets[i],values[i],,calldatas[i],eta))),GovernorTimelockCompound:
identical proposal action already queued) (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#99-102)
GovernorTimelockCompound.queue(address[],uint256[],bytes[],bytes32) (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#85-109) has
external calls inside a loop:
_timelock.queueTransaction(targets[i],values[i],,calldatas[i],eta) (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#103)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation/#calls-
inside-a-loop
INFO:Detectors:
Reentrancy in GovernorTimelockCompound.queue(address[],uint256[],bytes[],bytes32)
(lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#85-109):
    External calls:
    - _timelock.queueTransaction(targets[i],values[i],,calldatas[i],eta)
(lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#103)
    Event emitted after the call(s):
    - ProposalQueued(proposalId,eta) (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#106)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-
vulnerabilities-3
INFO:Detectors:
GovernorTimelockCompound.state(uint256) (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#51-66) uses
timestamp for comparisons
    Dangerous comparisons:
    - eta == 0 (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#59)
    - block.timestamp >= eta + _timelock.GRACE_PERIOD() (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#61)
GovernorTimelockCompound.queue(address[],uint256[],bytes[],bytes32) (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#85-109) uses
timestamp for comparisons
    Dangerous comparisons:
    - require(bool,string)(! _timelock.queuedTransactions(keccak256(bytes)
(abi.encode(targets[i],values[i],,calldatas[i],eta))),GovernorTimelockCompound:
identical proposal action already queued) (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#99-102)
GovernorTimelockCompound._execute(uint256,address[],uint256[],bytes[],bytes32)
(lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#114-127) uses
timestamp for comparisons
    Dangerous comparisons:
    - require(bool,string)(eta > 0,GovernorTimelockCompound: proposal not yet queued)
(lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#122)
GovernorTimelockCompound._cancel(address[],uint256[],bytes[],bytes32)
(lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#133-152) uses
timestamp for comparisons
    Dangerous comparisons:
    - eta > 0 (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#142)
ERC20Permit.permit(address,address,uint256,uint256,uint8,bytes32,bytes32)
(lib/openzeppelin-contracts/contracts/token/ERC20/extensions/ERC20Permit.sol#49-68)
uses timestamp for comparisons
    Dangerous comparisons:
    - require(bool,string)(block.timestamp <= deadline,ERC20Permit: expired deadline)
(lib/openzeppelin-contracts/contracts/token/ERC20/extensions/ERC20Permit.sol#58)
ERC20Votes.delegateBySig(address,uint256,uint256,uint8,bytes32,bytes32)
(lib/openzeppelin-contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#167-184)
uses timestamp for comparisons
    Dangerous comparisons:
    - require(bool,string)(block.timestamp <= expiry,ERC20Votes: signature expired)
(lib/openzeppelin-contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#175)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#block-
timestamp
```

```
INFO:Detectors:
Governor._isValidDescriptionForProposer(address,string) (lib/openzeppelin-
contracts/contracts/governance/Governor.sol#656-696) uses assembly
    - INLINE ASM (lib/openzeppelin-contracts/contracts/governance/Governor.sol#669-
677)
ERC20Votes._unsafeAccess(ERC20Votes.Checkpoint[],uint256) (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#284-289) uses assembly
    - INLINE ASM (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#285-288)
Address._revert(bytes,string) (lib/openzeppelin-
contracts/contracts/utils/Address.sol#231-243) uses assembly
    - INLINE ASM (lib/openzeppelin-contracts/contracts/utils/Address.sol#236-239)
ShortStrings.toString(ShortString) (lib/openzeppelin-
contracts/contracts/utils/ShortStrings.sol#63-73) uses assembly
    - INLINE ASM (lib/openzeppelin-contracts/contracts/utils/ShortStrings.sol#68-71)
StorageSlot.getAddressSlot(bytes32) (lib/openzeppelin-
contracts/contracts/utils/StorageSlot.sol#62-67) uses assembly
    - INLINE ASM (lib/openzeppelin-contracts/contracts/utils/StorageSlot.sol#64-66)
StorageSlot.getBooleanSlot(bytes32) (lib/openzeppelin-
contracts/contracts/utils/StorageSlot.sol#72-77) uses assembly
    - INLINE ASM (lib/openzeppelin-contracts/contracts/utils/StorageSlot.sol#74-76)
StorageSlot.getBytes32Slot(bytes32) (lib/openzeppelin-
contracts/contracts/utils/StorageSlot.sol#82-87) uses assembly
    - INLINE ASM (lib/openzeppelin-contracts/contracts/utils/StorageSlot.sol#84-86)
StorageSlot.getUint256Slot(bytes32) (lib/openzeppelin-
contracts/contracts/utils/StorageSlot.sol#92-97) uses assembly
    - INLINE ASM (lib/openzeppelin-contracts/contracts/utils/StorageSlot.sol#94-96)
StorageSlot.getStringSlot(bytes32) (lib/openzeppelin-
contracts/contracts/utils/StorageSlot.sol#102-107) uses assembly
    - INLINE ASM (lib/openzeppelin-contracts/contracts/utils/StorageSlot.sol#104-106)
StorageSlot.getStringSlot(string) (lib/openzeppelin-
contracts/contracts/utils/StorageSlot.sol#112-117) uses assembly
    - INLINE ASM (lib/openzeppelin-contracts/contracts/utils/StorageSlot.sol#114-116)
StorageSlot.getBytesSlot(bytes32) (lib/openzeppelin-
contracts/contracts/utils/StorageSlot.sol#122-127) uses assembly
    - INLINE ASM (lib/openzeppelin-contracts/contracts/utils/StorageSlot.sol#124-126)
StorageSlot.getBytesSlot(bytes) (lib/openzeppelin-
contracts/contracts/utils/StorageSlot.sol#132-137) uses assembly
    - INLINE ASM (lib/openzeppelin-contracts/contracts/utils/StorageSlot.sol#134-136)
Strings.toString(uint256) (lib/openzeppelin-contracts/contracts/utils/Strings.sol#19-
39) uses assembly
    - INLINE ASM (lib/openzeppelin-contracts/contracts/utils/Strings.sol#25-27)
    - INLINE ASM (lib/openzeppelin-contracts/contracts/utils/Strings.sol#31-33)
ECDSA.tryRecover(bytes32,bytes) (lib/openzeppelin-
contracts/contracts/utils/cryptography/ECDSA.sol#55-72) uses assembly
    - INLINE ASM (lib/openzeppelin-
contracts/contracts/utils/cryptography/ECDSA.sol#63-67)
ECDSA.toEthSignedMessageHash(bytes32) (lib/openzeppelin-
contracts/contracts/utils/cryptography/ECDSA.sol#165-174) uses assembly
    - INLINE ASM (lib/openzeppelin-
contracts/contracts/utils/cryptography/ECDSA.sol#169-173)
ECDSA.toTypedDataHash(bytes32,bytes32) (lib/openzeppelin-
contracts/contracts/utils/cryptography/ECDSA.sol#197-206) uses assembly
    - INLINE ASM (lib/openzeppelin-
contracts/contracts/utils/cryptography/ECDSA.sol#199-205)
Math.mulDiv(uint256,uint256,uint256) (lib/openzeppelin-
contracts/contracts/utils/math/Math.sol#55-134) uses assembly
    - INLINE ASM (lib/openzeppelin-contracts/contracts/utils/math/Math.sol#62-66)
    - INLINE ASM (lib/openzeppelin-contracts/contracts/utils/math/Math.sol#85-92)
    - INLINE ASM (lib/openzeppelin-contracts/contracts/utils/math/Math.sol#99-108)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#assembly-
usage
INFO:Detectors:
5 different versions of Solidity are used:
    - Version constraint ^0.8.0 is used by:
        -^0.8.0 (lib/openzeppelin-contracts/contracts/governance/Governor.sol#4)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/governance/IGovernor.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/governance/compatibility/GovernorCompatibilityBravo.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/governance/compatibility/IGovernorCompatibilityBravo.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorSettings.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorVotesComp.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/governance/extensions/IGovernorTimelock.sol#4)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/governance/utils/IVotes.sol#3)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/interfaces/IERC165.sol#4)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/interfaces/IERC5267.sol#4)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/interfaces/IERC5805.sol#4)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/interfaces/IERC6372.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/token/ERC1155/IERC1155Receiver.sol#4)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/token/ERC20/ERC20.sol#4)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/token/ERC20/IERC20.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Permit.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20VotesComp.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/IERC20Metadata.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/IERC20Permit.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/token/ERC721/IERC721Receiver.sol#4)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/utils/Context.sol#4)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/utils/Counters.sol#4)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/utils/StorageSlot.sol#5)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/utils/Strings.sol#4)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/utils/cryptography/ECDSA.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/utils/introspection/ERC165.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/utils/introspection/IERC165.sol#4)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/utils/math/Math.sol#4)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/utils/math/SafeCast.sol#5)
        -^0.8.0 (lib/openzeppelin-contracts/contracts/utils/math/SignedMath.sol#4)
        -^0.8.0 (lib/openzeppelin-
contracts/contracts/vendor/compound/ICompoundTimelock.sol#4)
    - Version constraint ^0.8.1 is used by:
        -^0.8.1 (lib/openzeppelin-contracts/contracts/utils/Address.sol#4)
    - Version constraint ^0.8.8 is used by:
        -^0.8.8 (lib/openzeppelin-contracts/contracts/utils/ShortStrings.sol#4)
        -^0.8.8 (lib/openzeppelin-contracts/contracts/utils/cryptography/EIP712.sol#4)
    - Version constraint ^0.8.4 is used by:
        -^0.8.4 (lib/openzeppelin-
contracts/contracts/utils/structs/DoubleEndedQueue.sol#3)
    - Version constraint ^0.8.20 is used by:
        -^0.8.20 (src/RadworksGovernor.sol#2)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#different-
pragma-directives-are-used
INFO:Detectors:
ERC20._burn(address,uint256) (lib/openzeppelin-
contracts/contracts/token/ERC20/ERC20.sol#277-293) is never used and should be removed
ERC20._mint(address,uint256) (lib/openzeppelin-
contracts/contracts/token/ERC20/ERC20.sol#251-264) is never used and should be removed
ERC20Votes._add(uint256,uint256) (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#273-275) is never used and
should be removed
ERC20Votes._burn(address,uint256) (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#206-210) is never used and
should be removed
ERC20Votes._maxSupply() (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#189-191) is never used and
should be removed
ERC20Votes._mint(address,uint256) (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#196-201) is never used and
should be removed
ERC20Votes._subtract(uint256,uint256) (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#277-279) is never used and
should be removed
ERC20VotesComp._maxSupply() (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20VotesComp.sol#43-45) is never used and
should be removed
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#dead-code
INFO:Detectors:
Version constraint ^0.8.0 contains known severe issues
(https://solidity.readthedocs.io/en/latest/bugs.html)
    - FullInlinerNonExpressionSplitArgumentEvaluationOrder
    - MissingSideEffectsOnSelectorAccess
    - AbiReencodingHeadOverflowWithStaticArrayCleanup
    - DirtyBytesArrayToStorage
    - DataLocationChangeInInternalOverride
    - NestedCalldataArrayAbiReencodingSizeValidation
    - SignedImmutables
    - ABIDecodeTwoDimensionalArrayMemory
    - KeccakCaching.
It is used by:
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/governance/Governor.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/governance/IGovernor.sol#4)
    - ^0.8.0 (lib/openzeppelin-
contracts/contracts/governance/compatibility/GovernorCompatibilityBravo.sol#4)
    - ^0.8.0 (lib/openzeppelin-
contracts/contracts/governance/compatibility/IGovernorCompatibilityBravo.sol#4)
    - ^0.8.0 (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorSettings.sol#4)
    - ^0.8.0 (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#4)
    - ^0.8.0 (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorVotesComp.sol#4)
    - ^0.8.0 (lib/openzeppelin-
contracts/contracts/governance/extensions/IGovernorTimelock.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/governance/utils/IVotes.sol#3)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/interfaces/IERC165.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/interfaces/IERC5267.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/interfaces/IERC5805.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/interfaces/IERC6372.sol#4)
    - ^0.8.0 (lib/openzeppelin-
contracts/contracts/token/ERC1155/IERC1155Receiver.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/token/ERC20/ERC20.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/token/ERC20/IERC20.sol#4)
    - ^0.8.0 (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Permit.sol#4)
    - ^0.8.0 (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#4)
    - ^0.8.0 (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20VotesComp.sol#4)
    - ^0.8.0 (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/IERC20Metadata.sol#4)
    - ^0.8.0 (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/IERC20Permit.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/token/ERC721/IERC721Receiver.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/utils/Context.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/utils/Counters.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/utils/StorageSlot.sol#5)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/utils/Strings.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/utils/cryptography/ECDSA.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/utils/introspection/ERC165.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/utils/introspection/IERC165.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/utils/math/Math.sol#4)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/utils/math/SafeCast.sol#5)
    - ^0.8.0 (lib/openzeppelin-contracts/contracts/utils/math/SignedMath.sol#4)
    - ^0.8.0 (lib/openzeppelin-
contracts/contracts/vendor/compound/ICompoundTimelock.sol#4)
Version constraint ^0.8.1 contains known severe issues
(https://solidity.readthedocs.io/en/latest/bugs.html)
    - FullInlinerNonExpressionSplitArgumentEvaluationOrder
    - MissingSideEffectsOnSelectorAccess
    - AbiReencodingHeadOverflowWithStaticArrayCleanup
    - DirtyBytesArrayToStorage
    - DataLocationChangeInInternalOverride
    - NestedCalldataArrayAbiReencodingSizeValidation
    - SignedImmutables
    - ABIDecodeTwoDimensionalArrayMemory
    - KeccakCaching.
It is used by:
    - ^0.8.1 (lib/openzeppelin-contracts/contracts/utils/Address.sol#4)
Version constraint ^0.8.8 contains known severe issues
(https://solidity.readthedocs.io/en/latest/bugs.html)
    - VerbatimInvalidDeduplication
    - FullInlinerNonExpressionSplitArgumentEvaluationOrder
    - MissingSideEffectsOnSelectorAccess
    - AbiReencodingHeadOverflowWithStaticArrayCleanup
    - DirtyBytesArrayToStorage
    - DataLocationChangeInInternalOverride
    - NestedCalldataArrayAbiReencodingSizeValidation
    - UserDefinedValueTypesBug
    - SignedImmutables.
It is used by:
    - ^0.8.8 (lib/openzeppelin-contracts/contracts/utils/ShortStrings.sol#4)
    - ^0.8.8 (lib/openzeppelin-contracts/contracts/utils/cryptography/EIP712.sol#4)
Version constraint ^0.8.4 contains known severe issues
(https://solidity.readthedocs.io/en/latest/bugs.html)
    - FullInlinerNonExpressionSplitArgumentEvaluationOrder
    - MissingSideEffectsOnSelectorAccess
    - AbiReencodingHeadOverflowWithStaticArrayCleanup
    - DirtyBytesArrayToStorage
    - DataLocationChangeInInternalOverride
    - NestedCalldataArrayAbiReencodingSizeValidation
    - SignedImmutables.
It is used by:
    - ^0.8.4 (lib/openzeppelin-
contracts/contracts/utils/structs/DoubleEndedQueue.sol#3)
Version constraint ^0.8.20 contains known severe issues
(https://solidity.readthedocs.io/en/latest/bugs.html)
    - VerbatimInvalidDeduplication
    - FullInlinerNonExpressionSplitArgumentEvaluationOrder
    - MissingSideEffectsOnSelectorAccess.
It is used by:
    - ^0.8.20 (src/RadworksGovernor.sol#2)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#incorrect-
versions-of-solidity
INFO:Detectors:
Low level call in Governor._execute(uint256,address[],uint256[],bytes[],bytes32)
(lib/openzeppelin-contracts/contracts/governance/Governor.sol#363-375):
    - (success,returndata) = targets[i].call{value: values[i]}(calldatas[i])
(lib/openzeppelin-contracts/contracts/governance/Governor.sol#372)
Low level call in Governor.relay(address,uint256,bytes) (lib/openzeppelin-
contracts/contracts/governance/Governor.sol#593-596):
    - (success,returndata) = target.call{value: value}(data) (lib/openzeppelin-
contracts/contracts/governance/Governor.sol#594)
Low level call in Address.sendValue(address,uint256) (lib/openzeppelin-
contracts/contracts/utils/Address.sol#64-69):
    - (success,None) = recipient.call{value: amount}() (lib/openzeppelin-
contracts/contracts/utils/Address.sol#67)
Low level call in Address.functionCallWithValue(address,bytes,uint256,string)
(lib/openzeppelin-contracts/contracts/utils/Address.sol#128-137):
    - (success,returndata) = target.call{value: value}(data) (lib/openzeppelin-
contracts/contracts/utils/Address.sol#135)
Low level call in Address.functionStaticCall(address,bytes,string) (lib/openzeppelin-
contracts/contracts/utils/Address.sol#155-162):
    - (success,returndata) = target.staticcall(data) (lib/openzeppelin-
contracts/contracts/utils/Address.sol#160)
Low level call in Address.functionDelegateCall(address,bytes,string)
(lib/openzeppelin-contracts/contracts/utils/Address.sol#180-187):
    - (success,returndata) = target.delegatecall(data) (lib/openzeppelin-
contracts/contracts/utils/Address.sol#185)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#low-level-
calls
INFO:Detectors:
Function IGovernor.CLOCK_MODE() (lib/openzeppelin-
contracts/contracts/governance/IGovernor.sol#96) is not in mixedCase
Function IGovernor.COUNTING_MODE() (lib/openzeppelin-
contracts/contracts/governance/IGovernor.sol#121) is not in mixedCase
Function GovernorCompatibilityBravo.COUNTING_MODE() (lib/openzeppelin-
contracts/contracts/governance/compatibility/GovernorCompatibilityBravo.sol#44-46) is
not in mixedCase
Function GovernorTimelockCompound.__acceptAdmin() (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorTimelockCompound.sol#165-167) is not
in mixedCase
Function GovernorVotesComp.CLOCK_MODE() (lib/openzeppelin-
contracts/contracts/governance/extensions/GovernorVotesComp.sol#37-43) is not in
mixedCase
Function IERC6372.CLOCK_MODE() (lib/openzeppelin-
contracts/contracts/interfaces/IERC6372.sol#16) is not in mixedCase
Function ERC20Permit.DOMAIN_SEPARATOR() (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Permit.sol#81-83) is not in mixedCase
Variable ERC20Permit._PERMIT_TYPEHASH_DEPRECATED_SLOT (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Permit.sol#37) is not in mixedCase
Function ERC20Votes.CLOCK_MODE() (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/ERC20Votes.sol#51-55) is not in mixedCase
Function IERC20Permit.DOMAIN_SEPARATOR() (lib/openzeppelin-
contracts/contracts/token/ERC20/extensions/IERC20Permit.sol#59) is not in mixedCase
Function ICompoundTimelock.GRACE_PERIOD() (lib/openzeppelin-
contracts/contracts/vendor/compound/ICompoundTimelock.sol#41) is not in mixedCase
Function ICompoundTimelock.MINIMUM_DELAY() (lib/openzeppelin-
contracts/contracts/vendor/compound/ICompoundTimelock.sol#44) is not in mixedCase
```

```
Function ICompoundTimelock.MAXIMUM_DELAY() (lib/openzeppelin-
contracts/contracts/vendor/compound/ICompoundTimelock.sol#47) is not in mixedCase
Function RadworksGovernor.COUNTING_MODE() (src/RadworksGovernor.sol#89-91) is not in
mixedCase
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#conformance-
to-solidity-naming-conventions
INFO:Detectors:
ShortStrings.slitherConstructorConstantVariables() (lib/openzeppelin-
contracts/contracts/utils/ShortStrings.sol#40-122) uses literals with too many digits:
    - _FALLBACK_SENTINEL =
0x00000000000000000000000000000000000000000000000000000000000000FF (lib/openzeppelin-
contracts/contracts/utils/ShortStrings.sol#42)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#too-many-
digits
INFO:Detectors:
GovernorCompatibilityBravo (lib/openzeppelin-
contracts/contracts/governance/compatibility/GovernorCompatibilityBravo.sol#21-333)
does not implement functions:
    - IGovernor.CLOCK_MODE() (lib/openzeppelin-
contracts/contracts/governance/IGovernor.sol#96)
    - Governor._getVotes(address,uint256,bytes) (lib/openzeppelin-
contracts/contracts/governance/Governor.sol#240)
    - IGovernor.clock() (lib/openzeppelin-
contracts/contracts/governance/IGovernor.sol#89)
    - IGovernorTimelock.proposalEta(uint256) (lib/openzeppelin-
contracts/contracts/governance/extensions/IGovernorTimelock.sol#18)
    - IGovernorTimelock.queue(address[],uint256[],bytes[],bytes32) (lib/openzeppelin-
contracts/contracts/governance/extensions/IGovernorTimelock.sol#20-25)
    - IGovernor.quorum(uint256) (lib/openzeppelin-
contracts/contracts/governance/IGovernor.sol#188)
    - IGovernorTimelock.timelock() (lib/openzeppelin-
contracts/contracts/governance/extensions/IGovernorTimelock.sol#16)
    - IGovernor.votingDelay() (lib/openzeppelin-
contracts/contracts/governance/IGovernor.sol#169)
    - IGovernor.votingPeriod() (lib/openzeppelin-
contracts/contracts/governance/IGovernor.sol#179)
Reference: https://github.com/crytic/slither/wiki/Detector-
Documentation#unimplemented-functions
INFO:Slither:0xD64D01D04498bFc60f04178e0B62a757C5048212 analyzed (37 contracts with 93
detectors), 84 result(s) found
```