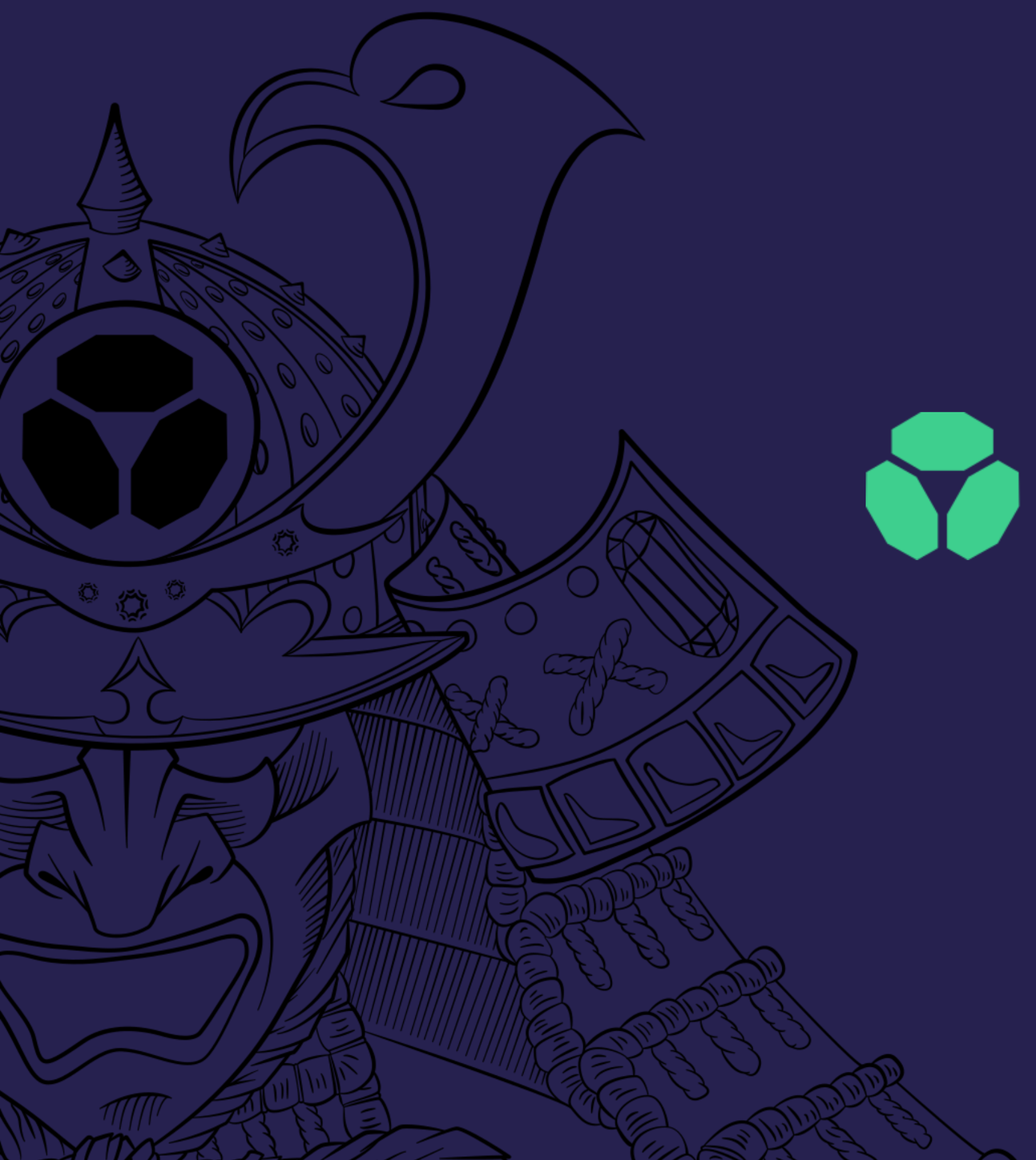


smart state

> Smart
Contract
Audit #

LAUNCH

Feb 08
2022

TABLE OF CONTENTS

METHODOLOGY

MAIN TESTS LIST:

- Best code practices
- ERC20/BEP20 compliance (if applicable)
- FA2 compliance (if applicable)
- Logical bugs
- General Denial Of Service(DOS)
- Locked ether
- Private data leaks
- Using components with known vulns
- Weak PRNG
- Unsed vars
- Uncheked call return method
- Code with no effects
- Pool Asset Security (backdoors in the underlying ERC-20)
- Function visibility
- Use of deprecated functions
- Authorization issues
- Re-entrancy
- Arithmetic Over/Under Flows
- Hidden Malicious Code
- External Contract Referencing
- Short Address/ Parameter Attack
- Race Conditions / Front Running
- Uninitialized Storage Pointers
- Floating Points and Precision
- Signatures Replay

STRUCTURE OF CONTRACT

BRIDGE.SOL

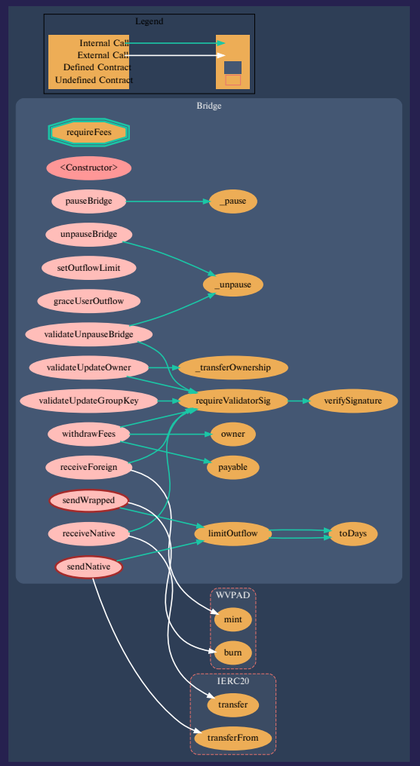
CONTRACT METHODS ANALYSIS:

- toDays(uint256 time)
  Vulnerabilties not detected

WARNING

- limitOutflow(uint256 value)
  lastOutflow is updated every time someone sends tokens, which makes daily limit not actually daily. If user will send in total 10 tokens every 12 hours daily limit would be overcomed only in 12 hours * (daily limit / 10 tokens), however if 10 tokens is actually smaller than daily limit it should never be overcomed.

- pauseBridge()
  Vulnerabilties not detected



Pic. 1.1
Bridge.sol

- unpauseBridge()
  Vulnerabilties not detected

- setOutflowLimit(uint256 amount)
  Function should emit an event

- graceUserOutflow(
  address user,
  uint256 amount
)
  Vulnerabilities not detected

- requireValidatorSig(
  uint256 actionId,
  uint256 sig,
  address proofAddress,
  uint256 actionHash
)
  Vulnerabilities not detected

PAYABLE

- receiveForeign(
  uint256 actionId,
  address to,
  uint256 value,
  uint256 sig,
  address proofAddr
)
  Function should emit an event

Tokens minted

PAYABLE

- receiveNative(
  uint256 actionId,
  address to,
  uint256 value,
  uint256 sig,
  address proofAddr
)
  Vulnerabilities not detected

Tokens out

PAYABLE

- sendNative(
  uint256 value,
  uint256 chainId
)
  Vulnerabilties not detected

Tokens in. ETH in

PAYABLE

- sendWrapped(
  uint256 value,
  uint256 chainId
)
  Vulnerabilties not detected

Tokens burned. ETH in

- validateUnpauseBridge(uint256 actionId, uint256 sig, address proofAddr)
  Function should emit an event

- validateUpdateOwner(uint256 actionId, address newOwner, uint256 sig, address proofAddr)
  Function should emit an event

- validateUpdateGroupKey(uint256 actionId, uint8 _validatorPKX, uint8 _validatorPKYP, uint256 sig, address proofAddr)
  Function should emit an event

PAYABLE

- withdrawFees(uint256 actionId, uint256 sig, address proofAddr)
  Function should emit an event

ETH out

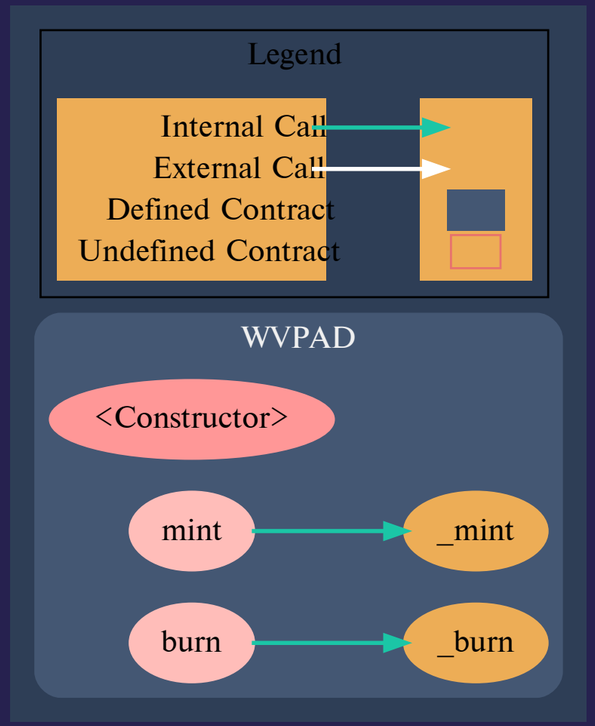
STRUCTURE OF CONTRACT

WVPAD.SOL

CONTRACT METHODS ANALYSIS:

- mint(address to, uint256 amount)
  Vulnerabilties not detected

- burn(address from, uint256 amount)
  Vulnerabilties not detected



Pic. 1.2
WVPAD.sol

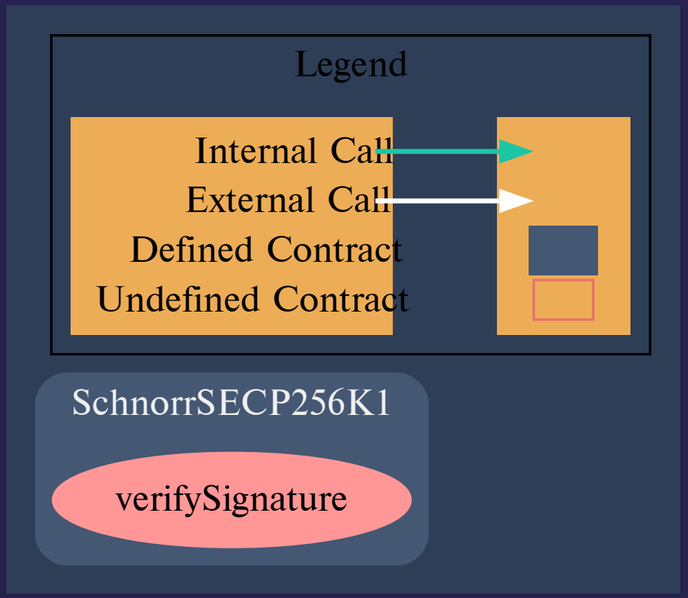
STRUCTURE OF CONTRACT

SCHNORRSECP256K1.SOL

CONTRACT METHODS ANALYSIS:

- verifySignature(
  uint256 signingPubKeyX,
  uint8 pubKeyYParity,
  uint256 signature,
  uint256 msgHash,
  address nonceTimesGeneratorAddress)

  Vulnerabilties not detected



Pic. 1.3
SchnorrSECP256K1.sol

VERIFICATION CHECK SUMS

Contract Name	Solc version	Optimisation	Bytecode hash (SHA 256)
Bridge.sol	0.8.0	200	ecaa6854270df75374d4b8d021c9ad5d09ad3cd27904c84187cfb8ea34225ab6
WVPAD.sol	0.8.0	200	eebde25c0348d8242fa81d8c3d3980b835aaed04356d507b22c15dce054e419

Get In Touch

info@smartstate.tech

smartstate.tech