Rareskill's Checklist (Extracted)

Summary

Author: Jeffrey Scholz

Source: Rareskill's Checklist

The list below is a modified version extracted by Nisedo

Checklist

- Excessive function restriction
  - Getting the balance right
  - Security often boils down to managing the way money exits the contract
- Double voting or msg.sender spoofing
- Flashloan Governance Attacks
- Flashloan Price Attacks
- Bypassing the contract check
- tx.origin
- Gas Griefing or Denial of Service
  - Deleting arrays that others can add to is also an denial of service vector
  - ERC777, ERC721, and ERC1155 can also be griefing vectors
- Insecure Randomness
- Using the Chainlink Randomness Oracle Wrong
- Getting stale data from a price Oracle
- Relying on only one oracle
- Oracles in general are hard to get right
- Mixed accounting
- Treating cryptographic proofs like passwords
- Solidity does not upcast to the final uint size
- Solidity sneakily makes some literals uint8
- Solidity downcasting does not revert on overflow
- Writes to storage pointers don't save new data.
- Deleting structs that contain dynamic datatypes does not delete the dynamic data
- ERC20 token issues
  - ERC20: Fee on transfer
  - ERC20: rebasing tokens
  - ERC20: ERC777 in ERC20 clothing
  - ERC20: Not all ERC20 tokens return true
  - ERC20: Address Poisoning
  - ERC20: Just flat out rugged
- Logic bugs in lending protocols
  - Ways lenders lose out
  - Ways borrowers lose out
- Vulnerabilities in staking protocols

- Unchecked return values
- msg.value in a loop
- Private Variables
- Try Catch is hard to get right
- Insecure Delegate Call
- Upgrade bugs related to proxies
- Overpowered Admins
- Use Ownable2Step instead of Ownable
- Rounding Errors
- Frontrunning
    - Frontrunning: Unprotected withdraw
    - Frontrunning: ERC4626 Inflation attack, a combination of frontrunning and rounding errors
    - Frontrunning: ERC20 approval
    - Frontrunning: Sandwich attacks
    - Learn more about frontrunning
- Signature Related
    - Anatomy of a signature
    - Signatures: ecrecover returns address(0) and doesn't revert when the address is invalid
    - Signature replay
    - Signature malleability
    - Secure signatures
    - Signatures can be forged or crafted without proper safeguards
    - Signatures as identifiers
- Some Solidity compiler versions have bugs
- Assuming smart contracts are immutable
- Transfer() and send() can break with multi-signature wallets
- Is Arithmetic overflow still relevant?
- What about block.timestamp?
- Corner Cases, Edge Cases, and Off By One Errors
    - Corner Case: Example 1
    - Off-By-One: Example 2
    - Corner Case Example 3: Compound Finance Reward Miscalculation
- Real World Hacks
    - Pairity Wallet Freeze (November 2017)
    - Badger DAO Hack (December 2021)
- Attack vectors for wallets
    - Private keys with insufficient randomness
    - Reused nonces or insufficiently random nonces.
- Most vulnerabilities are application specific
- Many vulnerabilities could have been caught with unit tests