Smart Contract Hacks

This documentation focuses on the most prevalent types of smart contract exploits observed across the Web3 ecosystem—ranging from reentrancy attacks and oracle manipulation to governance takeovers and cross-chain bridge vulnerabilities.
Each incident illustrates a recurring class of design flaw, logic error, or infrastructural oversight.
By systematically categorizing and analyzing these exploit patterns, this document serves as a practical threat modeling reference for developers, auditors, and protocol architects building secure decentralized applications.

Alpha Homora Exploit (2021)

- **Definition**: Integration bug led to nested lending attack.
- **Example**: $37M stolen via Cream Finance.
- **Cause**: Unsecured leveraged loops.
- **Prevention**: Invariant checks and max loop limits.

Uranium Finance Hack (2021)

- **Definition**: Bug in token swap logic during migration.
- **Example**: $50M drained.
- **Cause**: Deployed untested code.
- **Prevention**: Always test and audit migration scripts.

Poly Network Exploit (2021)

- **Definition**: Cross-chain contract flaw allowed fund transfers.
- **Example**: $610M stolen, later returned.
- **Cause**: Insecure owner verification.
- **Prevention**: Validate bridge signatures and restrict critical logic.

Cream Finance Hacks (2021)

- **Definition**: Reentrancy and flash loans abused lending pools.
- **Example**: >$130M lost across multiple incidents.
- **Cause**: Weak permission logic + nested calls.
- **Prevention**: Use ReentrancyGuard; verify loan limits.

BadgerDAO Frontend Attack (2021)

- **Definition**: UI served malicious approval scripts.
- **Example**: $120M drained from users.
- **Cause**: CDN hijack.
- **Prevention**: Harden build processes and use signing gateways.

Wormhole Bridge Hack (2022)

- **Definition**: Minted wrapped ETH without collateral via validation bypass.
- **Example**: $320M stolen before being reimbursed by Jump Crypto.
- **Cause**: Missing guardian signature checks.
- **Prevention**: Use multi-signature and ZKP-based validation.

Ronin Network Hack (2022)

- **Definition**: Validator key compromise enabled massive bridge withdrawal.
- **Example**: $625M stolen from Axie Infinity bridge.
- **Cause**: Centralized validator set.
- **Prevention**: Use threshold signing and rotate keys regularly.

Nomad Bridge Exploit (2022)

- **Definition**: Initialization bug allowed arbitrary withdrawals.
- **Example**: ~$200M drained by multiple attackers.
- **Cause**: Improper Merkle root setup.
- **Prevention**: Validate contract states and restrict initial access.

BNB Chain Hack (2022)

- **Definition**: Forged proofs allowed attacker to mint BNB.
- **Example**: ~$570M in BNB created; chain halted.
- **Cause**: Vulnerable IAVL proof logic.
- **Prevention**: Implement canonical proofs; monitor mint events.

Ankr Protocol Exploit (2022)

- **Definition**: Private key leaked; attacker minted unlimited tokens.
- **Example**: $5M+ drained from aBNBc mint exploit.
- **Cause**: Centralized key management.
- **Prevention**: Use MPC or HSM-based key custody.

TempleDAO Hack (2022)

- **Definition**: Misconfigured role enabled unauthorized withdrawal.
- **Example**: ~$2.3M stolen from staking vault.
- **Cause**: Insecure access control.
- **Prevention**: RBAC and privileged function audit.

Transit Swap Exploit (2022)

- **Definition**: Approval hijack through internal call injection.
- **Example**: ~$21M stolen; partial recovery.
- **Cause**: Poor input validation.
- **Prevention**: Whitelist contract calls; input sanitization.

PancakeBunny Flash Loan Attack (2021)

- **Definition**: Price manipulation using flash loans.
- **Example**: ~$45M drained from BSC protocol.
- **Cause**: No TWAP or price bounding.
- **Prevention**: Use average price oracles and flash loan limits.

Yam Finance Bug (2020)

- **Definition**: Integer overflow in rebase logic bricked governance.
- **Example**: $750K lost due to rebasing flaw.
- **Cause**: Faulty math in rebasing implementation.
- **Prevention**: Use SafeMath or Solidity 0.8+ and audit math-heavy logic.

Snowdog DAO Rug Pull (2021)

- **Definition**: Liquidity pulled using a hidden withdrawal script.
- **Example**: $10M lost in a coded rug pull.
- **Cause**: Opaque logic in migration contracts.
- **Prevention**: Transparently publish and audit migration plans.

Acala aUSD Depeg Incident (2022)

- **Definition**: Pool bug allowed excessive minting of aUSD.
- **Example**: 3B aUSD minted; stablecoin depegged.
- **Cause**: Logic flaw in liquidity pool config.
- **Prevention**: Limit mint caps and enforce access controls.

Saddle Finance Exploit (2022)

- **Definition**: Reentrancy bug in swap function.
- **Example**: $10M drained from stablecoin pools.
- **Cause**: No reentrancy guards in swap logic.
- **Prevention**: Apply nonReentrant modifier and update math logic.

Meerkat Finance Rug Pull (2021)

- **Definition**: Developers drained funds shortly after launch.
- **Example**: $31M withdrawn under pretense of a bug.
- **Cause**: Centralized control and malicious devs.
- **Prevention**: Enforce multisig control and perform team due diligence.

Hundred Finance Hack (2023)

- **Definition**: A rounding bug in borrowing logic allowed undercollateralized loans.
- **Example**: ~$7M was drained on Optimism.
- **Cause**: Incorrect borrow amount calculation logic.
- **Prevention**: Implement precise fixed-point math and unit testing.

LendHub Double-Spending Exploit (2023)

- **Definition**: Reentrancy occurred by interacting with old and new lending contracts.
- **Example**: ~$6M was stolen through contract duplication.
- **Cause**: Deprecated contracts were left active.
- **Prevention**: Properly deprecate old deployments and guard against reentrancy.

RAI Reflex Bonding Curve Bug (2022)

- **Definition**: Bonding curve math error allowed price manipulation.
- **Example**: ~$4M was extracted via arbitrage.
- **Cause**: Flawed bonding curve function logic.
- **Prevention**: Thorough test coverage of all bonding curve cases.

Linode Infrastructure Hack (2023)

- **Definition**: Validator node infrastructure provider was compromised.
- **Example**: Validator keys were seized, disrupting staking operations.
- **Cause**: Centralized infrastructure and unencrypted key storage.
- **Prevention**: Use distributed key management and secure node hosting.

CryptoNova Exploit (2023)

- **Definition**: Smart contract allowed minting beyond token cap.
- **Example**: ~$9M worth of excess tokens were minted and dumped.
- **Cause**: Lack of cap enforcement in mint function.
- **Prevention**: Implement hard-coded mint limits and test overflow.

Velodrome Wallet Leak (2023)

- **Definition**: Developer wallet private key was compromised.
- **Example**: ~$350K in funds were stolen.
- **Cause**: Inadequate key storage and protection.
- **Prevention**: Enforce cold storage and restrict wallet usage in production.

StakeWise Withdrawal Bug (2022)

- **Definition**: Logic bug led to excessive reward withdrawals.
- **Example**: Users were overcompensated unintentionally.
- **Cause**: Incorrect math in reward claim calculation.
- **Prevention**: Review all state transitions and math functions.

ParaSwap Contract Upgrade Bug (2022)

- **Definition:** Unsafe fallback logic in new routing contract allowed fund redirection.
- **Example:** ~$2.4M stolen post-upgrade.
- **Cause:** Improper upgrade testing and missing call safeguards.
- **Prevention:** Simulate production upgrades in test environments.

Solend Governance Takeover Attempt (2022)

- **Definition**: Governance proposal attempted to seize user funds.
- **Example**: Community backlash reversed the decision.
- **Cause**: Overly centralized DAO powers.
- **Prevention**: Limit emergency powers and add veto safeguards.

Beanstalk Flash Loan Governance Exploit (2022)

- **Definition**: Flash loan passed malicious DAO proposal.
- **Example**: ~$182M drained from protocol reserves.
- **Cause**: One-block governance execution.
- **Prevention**: Require time locks on proposal execution.

LaunchZone Token Drain (2023)

- **Definition**: Flash loan manipulated a token burn logic bug.
- **Example**: ~$700K stolen using mint and dump technique.
- **Cause**: Faulty burn/mint logic in liquidity contract.
- **Prevention**: Thoroughly test edge cases for all token functions.

Blur NFT Contract Logic Bug (2023)

- **Definition**: Listing logic bug enabled underpriced NFT purchases.
- **Example**: Snipers acquired rare NFTs at low prices.
- **Cause**: Incorrect price calculation and validation.
- **Prevention**: Add safeguards on listing price limits.

Gala Games Token Print Bug (2022)

- **Definition**: Bridge bug allowed unauthorized token minting.
- **Example**: ~$4.5M laundered before team intervention.
- **Cause**: Improper bridge logic validation.
- **Prevention**: Secure bridges with proofs and rate-limiting.

Harmony NFT Bridge Exploit (2022)

- **Definition**: Cross-chain NFT bridge accepted unsigned messages.
- **Example**: ~$2.2M in NFTs transferred illegitimately.
- **Cause**: No verification of bridge payloads.
- **Prevention**: Add message signature validation.

Multichain Router Vulnerability (2023)

- **Definition:** Admin key compromise led to mass asset drain.
- **Example:** ~$125M in cross-chain tokens were stolen.
- **Cause:** Single-signature access to router keys.
- **Prevention:** Use multisig or HSM-protected admin access.

Raccoon Wallet Exploit (2023)

- **Definition**: RPC injection attack compromised wallet integration.
- **Example**: Funds drained from users of Solana-based wallets.
- **Cause**: Insecure RPC endpoint access.
- **Prevention**: Validate and restrict RPC endpoints and payloads.

Zunami Protocol Exploit (2023)

- **Definition**: Flash loan attack manipulated pool weights in Curve.
- **Example:** ~$2.1M drained via rapid liquidity cycling.
- **Cause:** Lack of TWAP and oracle protection.
- **Prevention:** Use time-weighted average pricing and oracle checks.

Exactly Protocol Price Manipulation (2023)

- **Definition**: Price update delays enabled rate abuse in borrowing logic.
- **Example**: ~$720K lost to interest rate manipulation.
- **Cause**: Slow oracle feed synchronization.
- **Prevention**: Apply update throttling and stale data rejection.

Thena Finance Flash Loan Attack (2023)

- **Definition**: Flash loans used to manipulate LP pricing logic.
- **Example**: ~$400K drained before pools were halted.
- **Cause**: No oracle-based validation of pool prices.
- **Prevention**: Add price deviation guards and oracle checks.

Galxe API Leak & Wallet Drain (2023)

- **Definition**: API keys leaked via CI/CD allowed wallet compromise.
- **Example**: ~$200K lost via phishing-like attack vectors.
- **Cause**: Exposed credentials in development pipeline.
- **Prevention**: Secure CI/CD environments and rotate secrets regularly.

Arcadia Finance Exploit (2023)

- **Definition**: Missing call return checks led to undercollateralized borrowings.
- **Example**: ~$455K drained across L1 and L2 deployments.
- **Cause**: External call return values unchecked.
- **Prevention**: Use safe call wrappers and check return values.

ZyberSwap Router Vulnerability (2023)

- **Definition**: Router logic lacked balance validation.
- **Example**: ~$1M in liquidity drained.
- **Cause**: Improper accounting in transfer functions.
- **Prevention**: Validate transfer amounts and enforce invariant checks.

Sturdy Finance Oracle Exploit (2023)

- **Definition**: Bug in LP valuation logic enabled price manipulation.
- **Example**: ~$750K stolen in a single flash loan transaction.
- **Cause**: Vulnerable asset valuation in lending logic.
- **Prevention**: Use resilient price feeds and TWAP-based valuation.

Curve Finance UI Hijack (2023)

- Definition: DNS hijack redirected users to a malicious frontend.
- Example: ~$600K drained via spoofed approval prompts.
- Cause: DNS provider compromise.
- Prevention: Use DNSSEC and frontend integrity verification.

Balancer Frontend Exploit (2023)

- **Definition**: Storage link hijack led users to malicious site.
- **Example**: ~$238K stolen through spoofed transactions.
- **Cause**: Weak storage provider validation.
- **Prevention**: Secure static file delivery with hashing and HTTPS.

Velocore DEX Exploit (2023)

- **Definition**: LP manipulation bug exploited via flash loan.
- **Example**: ~$700K drained on zkSync chain.
- **Cause**: Lack of sync checks during liquidity events.
- **Prevention**: Use oracle guards and commit phases.

RugDoc Clone Scam (2023)

- **Definition**: Fake RugDoc site led users to phishing contracts.
- **Example**: ~$100K in funds were lost.
- **Cause**: Unverified site domain and wallet prompts.
- **Prevention**: Use visual verification and browser extension alerts.

zkSync Name Service Exploit (2023)

- **Definition**: Re-registration bug allowed name squatting.
- **Example**: Valuable top-level domains were hijacked.
- **Cause**: Incomplete ownership checks.
- **Prevention**: Lock names post-registration and verify ownership paths.

Merlin DEX Insider Exploit (2023)

- **Definition**: Insider minted LP tokens post-launch.
- **Example**: ~$1.8M drained by deployer wallet.
- **Cause**: Undisclosed mint function in contract.
- **Prevention**: Full scope auditing and permission visibility.

DEXTools Token Approval Phishing (2023)

- **Definition**: Fake token approval UI mimicked real interface.
- **Example**: ~$2.2M in ERC20 tokens lost.
- **Cause**: Unverified dApp with misleading prompts.
- **Prevention**: Educate users on wallet warnings and use hardware wallets.

Kokomo Finance Exit Scam (2023)

- **Definition**: upgradeToAndCall was used to rug protocol.
- **Example**: ~$4M drained from lending pool.
- **Cause**: Proxy upgrade pattern misused.
- **Prevention**: Lock upgrade functions and use multisig control.

Linea Protocol Bridge Exploit (2023)

- **Definition**: Bridge upgrade introduced signature bypass.
- **Example**: ~$2.5M stolen through unauthorized bridging.
- **Cause**: Unverified message acceptance in bridge.
- **Prevention**: Enforce signer validation and rate limits.

Liquidity Rug on zkFair (2023)

- **Definition**: Project deployed with fake zkSync branding and drained funds.
- **Example**: ~$6M TVL was rugged in minutes.
- **Cause**: Anonymous deployers and misleading marketing.
- **Prevention**: Vet projects before providing liquidity.

Defrost Finance Admin Key Exploit (2023)

- **Definition**: Malicious admin action added fake collateral tokens.
- **Example**: ~$12M drained; some funds later returned.
- **Cause**: Single-signer admin privileges.
- **Prevention**: Move sensitive functions to multisig governance.

Solarbeam DEX LP Drain (2023)

- **Definition**: Migration script lacked access restrictions.
- **Example**: ~$540K in LP tokens redirected.
- **Cause**: Public function for asset migration.
- **Prevention**: Restrict sensitive functions and test migrations.

Earn.Finance NFT Protocol Bug (2023)

- **Definition**: Metadata logic allowed fake claim eligibility.
- **Example**: ~$130K in NFTs claimed fraudulently.
- **Cause**: Unverified user input on claim.
- **Prevention**: Validate claim conditions with Merkle proofs.

Onyx Protocol Interest Rate Exploit (2023)

- **Definition**: Zero-interest tokens used to overborrow.
- **Example**: ~$2.1M borrowed against manipulated assets.
- **Cause**: Custom token whitelisting flaw.
- **Prevention**: Review all asset parameters and add per-asset caps.

Elk Finance Lockup Exploit (2023)

- **Definition**: Vesting contract logic allowed premature unlocks.
- **Example**: ~$900K claimed before schedule.
- **Cause**: Missing time checks in vesting logic.
- **Prevention**: Add unlock gating and simulate unlock tests.

Shell Protocol Approval Attack (2023)

- **Definition**: UI injected malicious token approval prompts.
- **Example**: Users signed unwanted approvals unknowingly.
- **Cause**: Centralized frontend compromised.
- **Prevention**: Use WalletConnect and verify UI builds cryptographically.

zkBridge Cross-Chain Replay Bug (2024)

- **Definition**: Lack of nonce replay protection in message handler.
- **Example**: ~$3M rerouted between chains.
- **Cause**: Missing message ID uniqueness enforcement.
- **Prevention**: Add replay protection and nonce tracking for all relayed messages.

Radiant Capital Reentrancy Bug (2023)

- **Definition**: Lending logic had a reentrancy vulnerability.
- **Example**: ~$4.5M was drained via callback abuse.
- **Cause**: Nested calls not restricted.
- **Prevention**: Use nonReentrant modifiers and role checks.

Kwenta Trading Contract Leak (2023)

- **Definition**: Internal function exposure allowed manipulation.
- **Example**: Exploit of leverage settings by bots.
- **Cause**: Misconfigured access control.
- **Prevention**: Scope and restrict function visibility.

Sonne Finance Exploit (2024)

- **Definition**: Collateral manipulation enabled overborrowing.
- **Example**: ~$20M was drained.
- **Cause**: Lack of asset cap enforcement.
- **Prevention**: Apply per-asset caps and circuit breakers.

Dolomite Liquidation Exploit (2024)

- **Definition**: Rounding issue abused for liquidation advantage.
- **Example**: ~$2M earned via underpayment during liquidations.
- **Cause**: Incorrect math precision.
- **Prevention**: Use safe math libraries and simulate edge cases.

Kelp DAO Slippage Attack (2024)

- **Definition**: Low liquidity led to price manipulation.
- **Example**: ~$500K drained through misconfigured return logic.
- **Cause**: Missing slippage bounds.
- **Prevention**: Integrate oracle checks and slippage limits.

Chakra Yield Token Printer Bug (2024)

- **Definition**: Infinite mint loop due to faulty exit logic.
- **Example**: ~$1.2M in tokens were minted.
- **Cause**: Loop condition failed to terminate.
- **Prevention**: Add maximum mint limits and loop caps.

Rysk Finance Position Manipulation (2024)

- **Definition**: Trade latency enabled AMM manipulation.
- **Example**: Sandwich trades extracted ~$520K in profits.
- **Cause**: Pricing delay in AMM logic.
- **Prevention**: Use TWAP and commit-reveal mechanics.

Sommelier Vault Withdrawal Bug (2024)

- **Definition**: Same-token withdrawals allowed multiple claims.
- **Example**: ~$300K withdrawn repeatedly.
- **Cause**: State update failure.
- **Prevention**: Track withdrawal state and audit payout logic.

Orbit Bridge Hack (2024)

- **Definition**: Validator compromise enabled forged transfers.
- **Example**: ~$84M stolen across chains.
- **Cause**: Weak validator verification.
- **Prevention**: Use ZK proofs and trusted signer rotation.

LayerZero Proxy Abuse (2024)

- **Definition**: Improper proxy upgrade led to admin takeover.
- **Example**: Funds were stolen from testnet contracts.
- **Cause**: Lack of upgrade controls.
- **Prevention**: Add allowlist and governance on proxy functions.

Blast Token Sale Contract Leak (2024)

- **Definition**: Contract had a hidden withdrawal path.
- **Example**: ~$2.1M was extracted pre-sale.
- **Cause**: Malicious developer injection.
- **Prevention**: Audit token sale contracts before deployment.

KyberSwap Elastic Attack (2023)

- **Definition**: Swap and vault callbacks exploited.
- **Example**: ~$48M lost via flash loan and sync bug.
- **Cause**: Improper vault coordination.
- **Prevention**: Limit reentry vectors and vault sync paths.

Chai Protocol Infinite Mint Bug (2023)

- **Definition**: Decimal rounding enabled infinite mint.
- **Example**: ~$110K worth of tokens minted.
- **Cause**: Unsafe math on token amount.
- **Prevention**: Use OZ ERC20 base with SafeMath.

Thorchain Router Logic Bug (2023)

- **Definition**: Fee miscalculation sent duplicate outputs.
- **Example**: ~$1.2M routed to attackers over time.
- **Cause**: Logic flaw in router fee engine.
- **Prevention**: Peer-reviewed code and fee audits.

LeetSwap LP Drain (2023)

- **Definition**: Low liquidity swap reroute drained LPs.
- **Example**: ~$600K lost on zkSync AMM.
- **Cause**: Sync delay on liquidity state.
- **Prevention**: Use slippage and withdrawal throttles.

Libra Incentive Contract Bug (2024)

- **Definition**: Vesting logic allowed instant unlock.
- **Example**: ~$250K worth of tokens claimed early.
- **Cause**: Missing cliff validation.
- **Prevention**: Validate vesting parameters and add delays.

KiloEx Exchange Bug (2024)

- **Definition**: Collateral was underpriced during deposit.
- **Example**: ~$2M in leveraged trades executed unfairly.
- **Cause**: Whitelist logic misapplied.
- **Prevention**: Audit all pricing paths and whitelist filters.

Puffer Finance Bounty Abuse (2024)

- **Definition**: Bounty module misclassified reports as critical.
- **Example**: Entire bounty pool drained.
- **Cause**: Faulty classification logic.
- **Prevention**: Manual validation of critical bug submissions.

El Dorado Exploit (2024)

- **Definition**: Token sale allowed premature withdrawal.
- **Example**: ~$1.3M withdrawn before vesting.
- **Cause**: Early unlock condition exposed.
- **Prevention**: Gate token access by vesting period.

Velodrome Reentrancy Bug (2024)

- **Definition**: Bribe contract allowed recursive execution.
- **Example**: ~$250K stolen via repeated reward claims.
- **Cause**: No guard on callback logic.
- **Prevention**: Add nonReentrant modifiers and call stacks.

KyberSwap Logic Flow Vulnerability (2023)

- **Definition:** Incorrect handling of liquidity and rounding errors in the swap logic allowed unauthorized draining of the pool.
- **Example:** Attacker repeatedly minted and drained liquidity, exploiting a rounding error to set the invariant to zero and steal funds.
- **Cause:** Rounding error in the calculation of the invariant (`k`) and improper logic in liquidity management.
- **Prevention:** Implement strict invariant checks, use fuzz testing, and validate all arithmetic operations to prevent logic errors and rounding exploits

Most Repeated Smart Contract Hacks

Reentrancy Attacks

- **Definition:** External contract re-enters a vulnerable function before state is updated.
- **Occurrences:** 10+ times
- **Causes:** External calls before internal logic completes
- **Example:** DAO Hack (2016), Cream Finance (2021)
- **Prevention:** Use ReentrancyGuard, apply Checks-Effects-Interactions pattern

Oracle Manipulation

- **Definition:** Attackers feed manipulated price data via centralized or poorly sourced oracles.
- **Occurrences:** 12+ times
- **Causes:** Trust in a single price source, flash loan influence
- **Example:** Harvest Finance (2020), bZx Exploits (2020)
- **Prevention:** Use Chainlink or decentralized oracles, apply TWAP, aggregate multiple feeds

Flash Loan Exploits

- **Definition:** Borrowing assets without collateral and exploiting protocol logic within a single transaction.
- **Occurrences:** 15+ times
- **Causes:** Logic reliant on unprotected asset states
- **Example:** Alpha Homora (2021), Pickle Finance (2020)
- **Prevention:** Flash loan guards, limit actions in single tx, invariant checks

Access Control Failures

- **Definition:** Exploiting improperly protected admin functions.
- **Occurrences:** 10+ times
- **Causes:** Missing/misused onlyOwner, hardcoded roles
- **Example:** Ankr (2022), Zunami Protocol (2024)
- **Prevention:** RBAC, multisig for sensitive ops, formal audits of admin logic

Logic Errors / Business Logic Flaws

- **Definition:** Flaws in economic assumptions or logic implementations.
- **Occurrences**: 10–12 times
- **Causes:** Incomplete testing, complex interactions, rebasing bugs
- **Example:** Yam Finance (2020), Iron Finance (2021)
- **Prevention:** Formal verification, unit + economic testing, threat modeling

Improper Upgradeability

- **Definition:** Proxy contracts with insecure logic or initializations.
- **Occurrences:** 6+ times
- **Causes:** Unlocked implementations, no upgrade checks
- **Example:** Parity Wallet (2017), LayerZero Proxy Abuse (2024)
- **Prevention:** OZ patterns, lock implementations, use multisig upgrade governance

Unchecked External Calls

- **Definition:** Not verifying the result of call, delegatecall, etc.
- **Occurrences:** 8+ times
- **Causes:** Assuming external contract returns true
- **Example:** Arcadia Finance (2023), Chai Protocol
- **Prevention:** Always check return values, use safe wrappers, avoid raw calls

Integer Overflow/Underflow

- **Definition:** Math overflows (e.g., x + y > uint256) or underflows (x - y < 0)
- **Occurrences:** 6+ times (mostly pre-Solidity 0.8)
- **Causes:** Lack of input validation or unchecked math
- **Example:** BatchOverflow (2018)
- **Prevention:** Use Solidity 0.8+, use SafeMath for older versions, validate input/output math

Denial of Service (DoS)

- **Definition:** Preventing a contract from operating due to gas limits or revert conditions.
- **Occurrences:** 5–7 times
- **Causes:** Expensive loops, failed transfers, griefing
- **Example:** BadgerDAO (2021), Balancer Pool (2020)
- **Prevention:** Use pull payments, restrict loops, set gas limits

Rug Pulls / Insider Abuses

- **Definition:** Malicious withdrawal of funds by project insiders/devs.
- **Occurrences:** 20+ incidents
- **Causes:** Centralized ownership, backdoors, anonymous devs
- **Example:** Jetfuel Finance, Meerkat
- **Prevention:** Require renounced ownership, audits, multisig access

Frontend Injection Attacks

- **Definition:** Attackers inject malicious code into dApp UI to hijack user approvals.
- **Occurrences:** 4+ times
- **Causes:** Lack of frontend integrity, centralized hosting
- **Example:** BadgerDAO (2021), Curve Finance DNS exploit (2022)
- **Prevention:** Enable CSP headers, verify builds, separate signing UI

Cross-Chain Bridge Exploits

- **Definition:** Exploits in cross-chain transfer logic or validator signature verification.
- **Occurrences:** 8+ times
- **Causes:** Poor multi-sig setup, no zk/multichain proof validation
- **Example:** Wormhole (2022), Ronin (2022), Harmony (2022)
- **Prevention:** Use threshold signing, zk proofs, third-party monitoring

Insecure Randomness

- **Definition:** Using block data (timestamp, blockhash) to generate randomness.
- **Occurrences:** 5+ incidents
- **Causes:** Predictable entropy
- **Example:** Fomo3D game, random airdrops
- **Prevention:** Use Chainlink VRF or commit-reveal schemes

Timestamp Manipulation

- **Definition:** Miners manipulate block.timestamp to impact time-based logic.
- **Occurrences:** 4–5 incidents
- **Causes:** Relies on miner-controlled variables
- **Example:** Time-based lotteries, farming protocols
- **Prevention:** Use block.number or off-chain time services

Signature Replay Attacks

- **Definition:** Reusing a valid signature on a different contract or context.
- **Occurrences:** 4–6 incidents
- **Causes:** No nonce validation, signature binding flaws
- **Example:** Permit attacks in ERC20s
- **Prevention:** Use EIP-712, unique nonces, domain separators

Insecure Fallback/Receive Functions

- **Definition:** Exploitable fallback/receive functions receiving Ether unexpectedly.
- **Occurrences:** 5+ cases
- **Causes:** Implicit logic inside fallback
- **Example:** Force-feeding attacks or DoS on fallback
- **Prevention:** Add revert() in unneeded fallback, handle ETH explicitly

Event Emission Flaws

- **Definition:** Missing or incorrect event logs that mislead observers or UIs.
- **Occurrences:** 4–5+ incidents
- **Causes:** Incomplete logging, incorrect event parameters
- **Example:** Swaps not properly emitted, governance proposals
- **Prevention:** Ensure complete and correct event emission; test UI integration

Improper Initialization of Contracts

- **Definition**: Contracts deployed without calling the initialize() function or constructor-equivalent.
- **Occurrences**: 5–6 times
- **Causes**: Uninitialized proxy patterns or libraries
- **Example**: Parity Wallet (2017), Nomad (2022)
- **Prevention**: Lock implementations, validate initializer state.

Insecure Delegatecall Usage

- **Definition**: delegatecall executes code from another contract in the caller's context, enabling storage corruption.
- **Occurrences**: 5+
- **Causes**: Blindly trusting external libraries or upgrade modules
- **Example**: Multichain exploit (2023), Synthetix (prelaunch bug)
- **Prevention**: Avoid untrusted delegatecalls, isolate logic from state.

Improper Fallback Functions

- **Definition**: Fallbacks executing logic unexpectedly upon receiving ETH or calls.
- **Occurrences**: 4+
- **Causes**: Missing receive() guard logic, assumptions about gas or msg.sender
- **Example**: DoS via fallback (various early contracts)
- **Prevention**: Make fallback revert() if unused, write explicit logic.

Upgrade Function Misuse (upgradeToAndCall)

- **Definition**: Protocol upgrade functions like upgradeToAndCall() are used maliciously to inject new logic.
- **Occurrences**: 5+
- **Causes**: Exposed upgrade logic or single-owner control
- **Example**: Kokomo Finance (2023), Tornado Cash Governance Takeover (2023)
- **Prevention**: Wrap upgrades in multisig or DAO governance.

Incorrect Event Emission

- **Definition**: Misleading, missing, or fake event logs deceive off-chain systems.
- **Occurrences**: 4–5+
- **Causes**: Incomplete logging or event spoofing
- **Example**: Fake swap or transfer logs
- **Prevention**: Emit all critical state changes, validate on frontend indexers.

Incorrect Math Precision / Rounding

- **Definition**: Rounding errors in token calculations can lead to economic leakage.
- **Occurrences**: 5+
- **Causes**: Division before multiplication, low-precision math
- **Example**: Hundred Finance (2023), Dolomite (2024)
- **Prevention**: Use fixed-point libraries, test rounding edge cases.

Slippage Misconfiguration

- **Definition**: Protocols process trades with no or excessive slippage protection.
- **Occurrences**: 6+
- **Causes**: Missing min/max expected output checks
- **Example**: Kelp DAO (2024), Blur (2023)
- **Prevention**: Add slippage bounds to all swap or minting logic.