Name	Description	Ref	Related Patterns
Include fail-safe mechanisms	It is important to have some way to update the contract in the case some bugs will be discovered. Incorporate an emergency stop functionality into the SC that can be triggered by an authenticated party to disable sensitive functions. The fail-safe mechanism, if implemented using theProxy Delegate, could be also exploited for forwarding calls and data to another contract, which is an updated version of the current one (for instance, a version where the bug has been fixed).	43	SB, RL, TE, PD, OW
Never assume that a contract has zero balance	Be aware of coding an invariant that strictly checks the balance of a contract. An attacker can forcibly send ether to any account and this cannot be prevented.	10	CEI, MH, GC
State Channel	In some contexts, transactions either have too high fee compared to their value, or must have low latency. In these cases, rather than performing each blockchain transaction, it is possible to firstly perform the operations outside the blockchain, and then register all the results batching the requests in a unique blockchain transaction.	44, 33	RL
Limit the amount of ether	If the code, the compiler or the platform has a bug, the funds stored inyour smart contract may be lost, so limit the maximum amount. Checkthat all money transfers are performed through explicit withdrawalsmade by the beneficiary.	2	RL, BL, AU

Beware of transaction ordering	Miners have the power to alter the order of transactions arriving inshort times. Inconsistent transactions' orders, with respect to the timeof invocations, can cause race conditions.	35	TC
Be careful with multiple inheritance	Solidity uses the "C3 linearization". This means that when a contract isdeployed, the compiler will linearize the inheritance from right to left.Multiple overrides of a function in complex inheritance hierarchiescould potentially interact in tricky ways.	10	PD, REU
Use trustworthy dependencies	Use audited and trustworthy dependencies to existing SCs and ensurethat newly written code is minimized by using libraries.	10	REU
Withdrawal from Contracts	When you need to send Ethers or tokens to an address, don't sendthem directly. Instead, authorize the address' owner to withdraw thefunds, and let s/he perform the job.	39, 13	CEI
Becareful with external calls	If possible, avoid them. When using low-level call functions makesure to handle the possibility that the call will fail, by checking thereturn value. Also, avoid combining multiple ether transfers in a singletransaction. Mark untrusted interactions: name the variables, methods,and contract interfaces of the functions that call external contracts,in a way that makes it clear that interacting with them is potentiallyunsafe	10	CEI, MU, GC
Beware of re-entrancy	Never write functions that could be called recursively, before thefirst invocations is finished. This may cause destructive consequences.Ensure state committed before an external call.	2, 10	CEI, MU

Embed addresses to grant permissions	Make sure that critical methods can be invoked only by a specific set ofaddresses, which belong to privileged users. For instance, each contracthas an owner and only this address can invoke certain methods, likethe method for updating the address of the owner of the contract.	44	AU, OW
Use hash secrets to grant permissions	Sometimes you need to provide authorizations to some authoritieswhose addresses are not known yet in the developing phase (forinstance, they are unknown authorities). Although theEmbed per-missionspattern can not be applied, hash secrets help providing userpermissions without specifying any address. First, generate a secretkey and, in the contract, provide permissions by requiring its hash.Then, send (off-chain) the secret key to the authorities you want togrant permissions.	26, 44	AU
Use multi-signature	Define a set of entities (or addresses) that can authorize an action andrequire that only a subset of them is required to authorize the action.	26, 44	AU, OW
Avoid using tx.origin for authorizations	tx.originis a global variable that returns the address of the messagesender. Do not use it as an authorization mechanism.	31	AU
Encrypt on-chain data	Encrypt blockchain data for improving confidentiality and privacy.This is particularly important when actors are in competition.	44	PR

Hash objects for tracking off-chain data	Large objects (such as videos) should not be embedded in theblockchain, their hashes can be easily uploaded instead. Hashingobjects can be also applied to hide sensitive data in order to meetspecific legal requirements, such as the European GDPR.	44	PR
Use platform related standards	Use platform related standards, like the ERC (Ethereum Requestfor Comment) standards, which are application-level blueprints andconventions in the Ethereum ecosystem.	44	REU
Prevent overflow and underflow	If a balance reaches the maximum uint value it will circle back tozero; similarly, if a uint is made to be less than zero, it will cause anunderflow and get set to its maximum value. One simple solution tothis issue is to use a library likeSafeMath.solby OpenZeppelin.	35	MH, GC, REU, BL
Beware of rounding errors	All integer divisions round down to the nearest integer. Check thattruncation does not produce unexpected behavior (locked funds, in-correct results).	10	MH, GC, REU
Validate inputs to external and public functions	Make sure the requirements are verified and check for arguments. Useproperlyassert(),require()and revert()to check user inputs, SC state,invariants.	35, 13	GC
Prevent unbounded loops	When executing loops, the gas consumed increases with each iterationuntil it hits the block's gasLimit, stopping the execution. Accordingly,plan the number of iterations you need to perform and establish amaximum number. If you still need more iterations, divide computationamong distinct transactions.	13, 28	RL, BL, TC, TE

Provide fallback functions	The "fallback function" is called whenever a contract receive amessage which does not match any of the available functions, orwhenever it receives Ethers without any other data associated with thetransaction. Remember to mark it aspayable, be sure it does not haveany arguments, has external visibility and does not return anything.Moreover, keep it simple and if the fallback function is intended tobe used only for the purpose of logging received Ether, check that thedata is empty (i.e. require(msg.data.length == 0)).	10, 35	CEI, MU, GC
Check if referral code is written OnChain	Use Solidity to check etherscan, verify the block of the input transaction and compare the exchange front-end information with the blockchain. CEX example: Kucoin. DEX example: GMX.	Kucoin referral code	GMX referral code
Check if built-in variables or functions were overridden	It is possible to override built-in globals in Solidity. This allows SCsto override the functionality of built-ins such asmsgandrevert().Although this is intended, it can mislead users of a SC, so the wholecode of every SC called from the SC you are writing must be checked.	10	GC
Use interface type instead of the address for type safety	When a function takes a contract address as an argument, it is betterto pass an interface or contract type rather than a rawaddress. If thefunction is called elsewhere within the source code, the compiler willprovide additional type safety guarantees.	10	GC

Be careful with randomness	Random number generation in a deterministic system is very difficult.Do not rely on pseudo-randomness for important mechanisms. Currentbest solutions include hash-commit-reveal schemes (ie. one partygenerates a number, publishes its hash to "commit" to the value, andthen reveals the value later), querying oracles, and RANDAO.	4, 21	OR, REU
Be careful with Timestamp	Be aware that the timestamp of a block can be manipulated by aminer; all direct and indirect uses of timestamp should be analyzedand verified. If the scale of your time-dependent event can vary by30 seconds and maintain integrity, it is safe to use a timestamp. Thisincludes thing like ending of auctions, registration periods, etc. Do notuse theblock.numberproperty as a timestamp.	35	TC
Fix compiler warnings	Take warnings seriously and fix them. Always use the latest versionof the compiler to be notified about all recently introduced warnings.	39	
Lock programs to specific compiler version	Contracts should be deployed with the same compiler version and flagsthat they have been tested with, so locking the version helps avoid therisk of undiscovered bugs.	10	
Enforce invariants with assert	An assert guard triggers when an assertion fails - for instance aninvariant property changing. You can verify it with a call toassert().Assert guards should be combined with other techniques, such aspausing the contract and allowing upgrades. (Otherwise, you may endup stuck, with an assertion that is always failing.)	10	

Develop unit testing	Be sure to have a 100% text coverage and cover all critical edge caseswith unit tests. Do not deploy recently written code, especially if itwas written under tight deadline.	10	
Use frameworks for testing	When approaching smart contract testing, do not start from scratchbut use existing framework for contract testing.	10	
Use test networks	Before deploying the smart contract in the main network, try it in apublic test network or use a software for configuring a private localnetwork.	10	
Check Effect Interaction	When performing a function in a SC: first, check all the preconditions, then apply the effects to the contract's state, and finally interact with other contracts. Never alter this sequence.	43	CEI
Proxy Delegate / Decorator	Proxy patterns are a set of SCs working together to facilitate upgrading of SCs, despite their intrinsic immutability. A Proxy is used to refer to another SC, whose address can be changed. This approach also ensures that blockchain resources are used sparingly, thus saving GAS.	36,13,47,26,28	PD
Authorization	Restrict the execution of critical methods to specific users. This is accomplished using mappings of addresses, and is typically checked using modifiers.	4	AU
Ownership	Specify the contract owner, which is responsible for contract management and has special permissions, e.g. it is the only address authorized to call critical methods. This patter can be seen as a special instance of the authorization pattern.	4	OW

Oracle	An oracle is a SC providing data from outside the blockchain, which are in turn fed to the oracle by a trusted source. Here the security risk lies in how actually the source can be trusted.	4,44	OR
Reverse Oracle	A reverse oracle is a SC providing data to be read by off-chain components for checking specific conditions.	44	RO
Rate Limit	Regulate how often a task can be executed within a period of time, to limit the number of messages sent to a SC, and thus its computational load.	43	RL
Balance Limit	Limit the maximum amount of funds held within a SC.	43	BL
Guard Check	Ensure that all requirements on a SC state and on function inputs are met.	13	GC
Time Constraint	A time constraint specifies when an action is permitted, depending on the time registered in the block holding the transaction. It could be also used in Speed Bump and Rate Limit patterns.	4	TC
Termination	Used when the life of a SC has come to an end. This can be done by inserting ad-hoc code in the contract, or calling the selfdestruct function. Usually, only the contract owner is authorized to terminate a contract.	4	TE
Math	A logic which computes some critical operations, protecting from overflows, underflows or other undesired characteristics of finite arithmetic.	4	MH
Privacy	Encrypt on-chain critical data improving confidentiality and meeting legal requirements, such as the European GDPR.	44	PR
Reusability	Use contract libraries and templates as a factory for creating multiple instances.	44	REU

Mutex	A mutex is a mechanism to restrict concurrent access to a resource. Utilize it to hinder an external call from re-entering its caller function again.	43	MU
Speed Bump	Slow down contract sensitive tasks, so when malicious actions occur, the damage is limited and more time to counteract is available. For instance, limit the amount of money a user can withdraw per day, or impose a delay before withdrawals.	43	SB