



Quicksilver

Cosmos Security Audit

Prepared by: Halborn

Date of Engagement: July 25th, 2022 - August 29th, 2022

Visit: Halborn.com

DOCUMENT REVISION HISTORY	4
CONTACTS	4
1 EXECUTIVE OVERVIEW	5
1.1 INTRODUCTION	6
1.2 AUDIT SUMMARY	6
1.3 TEST APPROACH & METHODOLOGY	6
RISK METHODOLOGY	7
1.4 SCOPE	9
2 ASSESSMENT SUMMARY & FINDINGS OVERVIEW	10
3 FINDINGS & TECH DETAILS	11
3.1 (HAL-01) DUPLICATED QUERY PROCESSING - MEDIUM	13
Description	13
Code Location	13
Risk Level	13
Recommendation	13
Remediation Plan	13
3.2 (HAL-02) USAGE OF QUO COULD LEAD TO CRASH - INFORMATIONAL	14
Description	14
Code Location	14
Risk Level	16
Recommendation	16
Remediation Plan	16
3.3 (HAL-03) UNHANDLED ERRORS - INFORMATIONAL	17
Description	17

Code Location	17
Risk Level	20
Recommendation	21
Remediation Plan	21
3.4 (HAL-04) DUPLICATED ERROR CHECKS - INFORMATIONAL	22
Description	22
Code Location	22
Risk Level	22
Recommendation	23
Remediation Plan	23
3.5 (HAL-05) COMMENTED OUT CODE / UNUSED CODE - INFORMATIONAL	24
Description	24
Code Location	24
Risk Level	25
Recommendation	25
Remediation Plan	25
3.6 (HAL-06) PANIC IS USED FOR ERROR HANDLING - INFORMATIONAL	26
Description	26
Code Location	26
Risk Level	27
Recommendation	28
Remediation Plan	28
3.7 (HAL-07) OPEN TODOs - INFORMATIONAL	29
Description	29
Code Location	29
Risk Level	31

	Recommendation	31
	Remediation Plan	31
3.8	(HAL-08) UNUSED VARIABLES - INFORMATIONAL	33
	Description	33
	Code Location	33
	Risk Level	33
	Recommendation	33
	Remediation Plan	33
3.9	(HAL-09) INCORRECT COMMENTS IN CODE - INFORMATIONAL	34
	Description	34
	Code Location	34
	Risk Level	34
	Recommendation	34
	Remediation Plan	34
4	AUTOMATED TESTING	35
	Description	36
	Semgrep - Security Analysis Output Sample	36
	Semgrep Results	36
	Gosec - Security Analysis Output Sample	55
	ineffassign - Security Analysis Output Sample	58
	Staticcheck - Security Analysis Output Sample	58

DOCUMENT REVISION HISTORY

VERSION	MODIFICATION	DATE	AUTHOR
0.1	Document Creation	08/15/2022	Chris Meistre
0.2	Draft Review	08/22/2022	Gabi Urrutia
1.0	Remediation Plan	09/02/2022	Chris Meistre
1.1	Remediation Plan Updates	09/02/2022	Gokberk Gulgun
1.2	Remediation Plan Review	09/02/2022	Gabi Urrutia

CONTACTS

CONTACT	COMPANY	EMAIL
Rob Behnke	Halborn	Rob.Behnke@halborn.com
Steven Walbroehl	Halborn	Steven.Walbroehl@halborn.com
Gabi Urrutia	Halborn	Gabi.Urrutia@halborn.com
Chris Meistre	Halborn	Chris.Meistre@halborn.com



EXECUTIVE OVERVIEW



1.1 INTRODUCTION

Quicksilver engaged Halborn to conduct a security audit on their Interchain Staking, Participation Rewards and Interchain Queries modules, beginning on July 25th, 2022 and ending on August 29th, 2022 . The security assessment was scoped to the code base provided to the Halborn team.

1.2 AUDIT SUMMARY

The team at Halborn was provided four weeks for the engagement and assigned two full-time security engineers to audit the security of the modules. The security engineers are blockchain and smart-contract security experts with advanced penetration testing, smart-contract hacking, and deep knowledge of multiple blockchain protocols.

The purpose of this audit to achieve the following:

- Ensure that modules functions are intended.
- Report potential security issues to the Quicksilver Team.

In summary, Halborn identified few security risks that were mostly addressed by the **Quicksilver Team**.

1.3 TEST APPROACH & METHODOLOGY

Halborn performed a combination of manual and automated security testing to balance efficiency, timeliness, practicality, and accuracy in regard to the scope of the Margin module. While manual testing is recommended to uncover flaws in logic, process, and implementation; automated testing techniques help enhance coverage of structures and can quickly identify items that do not follow security best practices. The following phases and associated tools were used throughout the term of the audit:

- Research into architecture and purpose.
- Static Analysis of security for scoped repository, and imported functions. (`staticcheck`, `gosec`, `unconvert`, `LGTM`, `ineffassign` and `semgrep`).
- Manual Assessment for discovering security vulnerabilities on codebase.
- Ensuring correctness of the codebase.
- Dynamic Analysis on modules functions and data types.

RISK METHODOLOGY:

Vulnerabilities or issues observed by Halborn are ranked based on the risk assessment methodology by measuring the **LIKELIHOOD** of a security incident and the **IMPACT** should an incident occur. This framework works for communicating the characteristics and impacts of technology vulnerabilities. The quantitative model ensures repeatable and accurate measurement while enabling users to see the underlying vulnerability characteristics that were used to generate the Risk scores. For every vulnerability, a risk level will be calculated on a scale of 5 to 1 with 5 being the highest likelihood or impact.

RISK SCALE - LIKELIHOOD

- 5 - Almost certain an incident will occur.
- 4 - High probability of an incident occurring.
- 3 - Potential of a security incident in the long term.
- 2 - Low probability of an incident occurring.
- 1 - Very unlikely issue will cause an incident.

RISK SCALE - IMPACT

- 5 - May cause devastating and unrecoverable impact or loss.
- 4 - May cause a significant level of impact or loss.
- 3 - May cause a partial impact or loss to many.
- 2 - May cause temporary impact or loss.
- 1 - May cause minimal or un-noticeable impact.

The risk level is then calculated using a sum of these two values, creating

a value of 10 to 1 with 10 being the highest level of security risk.

CRITICAL	HIGH	MEDIUM	LOW	INFORMATIONAL
----------	------	--------	-----	---------------

10 - CRITICAL

9 - 8 - HIGH

7 - 6 - MEDIUM

5 - 4 - LOW

3 - 1 - VERY LOW AND INFORMATIONAL

1.4 SCOPE

IN-SCOPE:

The security assessment was scoped to `ingenuity-build/quicksilver` repository.

Branch

Commit ID

IN-SCOPE Module :

- `x/interchainquery`
- `x/interchainstaking`
- `x/participationrewards`

REMEDIATION COMMIT PROVIDED:

- Commit ID 1
- Commit ID 2
- Commit ID 3

2. ASSESSMENT SUMMARY & FINDINGS OVERVIEW

CRITICAL	HIGH	MEDIUM	LOW	INFORMATIONAL
0	0	1	0	8

LIKELIHOOD

IMPACT

	(HAL-01)			
(HAL-02) (HAL-03) (HAL-04) (HAL-05) (HAL-06) (HAL-07) (HAL-08) (HAL-09)				

SECURITY ANALYSIS	RISK LEVEL	REMEDIATION DATE
HAL-01 - DUPLICATED QUERY PROCESSING	Medium	SOLVED - 09/02/2022
HAL-02 - USAGE OF QUO COULD LEAD TO PANIC	Low	SOLVED - 09/02/2022
HAL-03 - UNHANDLED ERRORS	Low	SOLVED - 09/02/2022
HAL-04 - DUPLICATED ERROR CHECKS	Informational	SOLVED - 09/02/2022
HAL-05 - COMMENTED OUT CODE / UNUSED CODE	Informational	ACKNOWLEDGED
HAL-06 - PANIC IS USED FOR ERROR HANDLING	Informational	PARTIALLY SOLVED - 09/02/2022
HAL-07 - OPEN TODOs	Informational	PARTIALLY SOLVED - 09/02/2022
HAL-08 - UNUSED VARIABLE	Informational	SOLVED - 09/02/2022
HAL-09 - INCORRECT COMMENTS IN CODE	Informational	ACKNOWLEDGED



FINDINGS & TECH DETAILS



3.1 (HAL-01) DUPLICATED QUERY PROCESSING - MEDIUM

Description:

The same query could be processed more than once, leading to unexpected results. The current check does not ensure that the same block will not be processed again.

Code Location:

[x/interchainquery/keeper/msg_server.go](#), Lines 32-34

Listing 1

```
32     q, found := k.GetQuery(ctx, msg.QueryId)
33     // if found && q.LastHeight.Int64() != ctx.BlockHeader().
    ↳ Height {
34     if found {
```

Risk Level:

Likelihood - 2

Impact - 4

Recommendation:

It is recommended that proper checking be done to ensure that the same query will not be processed again within the same block.

Remediation Plan:

SOLVED: The [Quicksilver Team](#) has solved this by introducing a check to see if the query has been found.

3.2 (HAL-02) USAGE OF QUO COULD LEAD TO CRASH – INFORMATIONAL

Description:

There are some instances where the values used for division are not validated to be zero. In these cases, a **division by zero error** could cause the node to crash.

Code Location:

[x/interchainstaking/keeper/ibc_handlers.go](#), Line 734

Listing 2

```
734     portion := rewards.Amount.ToDec().Quo(sdk.NewDec(int64(len(
    ↳ zone.DelegationAddresses)))).TruncateInt()
```

[x/interchainstaking/keeper/intent.go](#), Lines 95-118

Listing 3: (Lines 95,118)

```
95     ordinalizedIntentSum := sdk.ZeroDec()
96     k.IterateIntents(ctx, zone, snapshot, func(_ int64, intent
    ↳ types.DelegatorIntent) (stop bool) {
97         query := bankTypes.QueryBalanceRequest{Address: intent.
    ↳ Delegator, Denom: zone.LocalDenom}
98         balance, err := k.BankKeeper.Balance(sdk.WrapSDKContext(
    ↳ ctx), &query)
99         if err != nil {
100             panic(err)
101         }
102         baseBalance := zone.RedemptionRate.Mul(sdk.NewDecFromInt(
    ↳ balance.Amount)).TruncateInt()
103         for _, vIntent := range intent.Ordinalize(baseBalance).
    ↳ Intents {
104             thisIntent, ok := intents[vIntent.ValoperAddress]
105             ordinalizedIntentSum = ordinalizedIntentSum.Add(
    ↳ vIntent.Weight)
```

```

106         if !ok {
107             intents[vIntent.ValoperAddress] = vIntent
108         } else {
109             thisIntent.Weight = thisIntent.Weight.Add(vIntent.
↳ Weight)
110             intents[vIntent.ValoperAddress] = thisIntent
111         }
112     }
113
114     return false
115 })
116
117     for key, val := range intents {
118         val.Weight = val.Weight.Quo(ordinalizedIntentSum)

```

[x/interchainstaking/keeper/receipt.go](#), Line 107

Listing 4

```

107         outAmount := inCoin.Amount.ToDec().Quo(zone.RedemptionRate
↳ ).TruncateInt()

```

[x/interchainstaking/types/delegation.go](#), Line 300

Listing 5

```

300         percent := current.ToDec().Quo(total.ToDec())
↳                                     // what is this a percent of total
↳ + new

```

[x/interchainstaking/types/validator.go](#), Lines 36-41

Listing 6: (Lines 36,41)

```

36         summedWeight := sdk.ZeroDec()
37         for _, i := range di.Sorted() {
38             summedWeight = summedWeight.Add(i.Weight)
39         }
40         for _, i := range di.Sorted() {
41             i.Weight = i.Weight.QuoTruncate(summedWeight)

```


Risk Level:**Likelihood - 1****Impact - 1****Recommendation:**

It is recommended to implement proper error checking be implemented to avoid any division by zero errors.

Remediation Plan:

SOLVED: The **Quicksilver Team** has solved this by introducing validation before the values are used in calculations.

3.3 (HAL-03) UNHANDLED ERRORS – INFORMATIONAL

Description:

There are some instances where error handling has not been implemented for functions that might return an error.

Code Location:

[x/interchainstaking/keeper/zones.go](#), Lines 411-413

Listing 7: (Line 412)

```
411         // if zero balance, retrigger the query.
412         k.EmitPerformanceBalanceQuery(ctx, &zone)
413         k.Logger(ctx).Info("performance account has a zero balance
    ↳ ; requerying")
```

[x/interchainstaking/keeper/zones.go](#), Lines 594-596

Listing 8: (Line 595)

```
594     k.SetDelegation(ctx, zone, delegation)
595     k.EmitValsetRequery(ctx, zone.ConnectionId, zone.ChainId)
596     k.SetRegisteredZone(ctx, *zone)
```

[x/interchainstaking/keeper/zones.go](#), Lines 549-552

Listing 9: (Line 551)

```
49         data := stakingtypes.GetDelegationKey(delAddr, valAddr)
50
51         k.RemoveDelegation(ctx, zone, existingDelegation)
52         da.DelegatedBalance = da.DelegatedBalance.Sub(
    ↳ existingDelegation.Amount) // remove old delegation from da.
    ↳ DelegatedBalance
```

x/interchainstaking/keeper/zones.go, Lines 354-358

Listing 10: (Line 356)

```

354             }
355
356             k.EmitValsetRequery(ctx, zone.ConnectionId,
↳ zone.ChainId)
357
358             return true

```

x/interchainstaking/keeper/zones.go, Lines 325-327

Listing 11: (Line 326)

```

325     for _, delegationPlan := range toDelete {
326         k.RemoveDelegationPlan(ctx, zone, memo, delegationPlan)
327     }

```

x/interchainstaking/keeper/zones.go, Lines 244-246

Listing 12: (Line 246)

```

244     for _, delegationPlan := range toDelete {
245         k.RemoveDelegationPlan(ctx, zone, memo, delegationPlan
↳ )
246     }

```

x/interchainstaking/keeper/callbacks.go, Lines 158-160

Listing 13: (Line 159)

```

158     if delegation, ok := k.GetDelegation(ctx, &zone,
↳ delegatorAddress, validatorAddress); ok {
159         k.RemoveDelegation(ctx, &zone, delegation)
160         ica, err := zone.GetDelegationAccountByAddress(
↳ delegatorAddress)

```

x/interchainstaking/keeper/callbacks.go, Lines 91-93

Listing 14: (Line 92)

```

91     }
92     SetValidatorForZone(k, ctx, zone, args)
93     return nil

```

x/interchainstaking/keeper/callbacks.go, Lines 81-93

Listing 15: (Line 82)

```

81 }
82 SetValidatorsForZone(k, ctx, zone, args)
83 return nil

```

x/interchainstaking/ibc_module.go, Lines 227-229

Listing 16: (Line 228)

```

227 }
228 msg := distrTypes.MsgSetWithdrawAddress{DelegatorAddress:
↳ address, WithdrawAddress: zone.WithdrawalAddress.String()}
229 im.keeper.SubmitTx(ctx, []sdk.Msg{&msg}, zone.
↳ PerformanceAddress, "")
230 }

```

x/interchainstaking/ibc_module.go, Lines 164-166

Listing 17: (Line 165)

```

164 msg := distrTypes.MsgSetWithdrawAddress{
↳ DelegatorAddress: address, WithdrawAddress: zoneInfo.
↳ WithdrawalAddress.String()}
165 im.keeper.SubmitTx(ctx, []sdk.Msg{&msg}, account, "")
166 }

```

x/interchainstaking/ibc_module.go, Lines 141-143

Listing 18: (Line 142)

```

141         msg := distrTypes.MsgSetWithdrawAddress{
    ↳ DelegatorAddress: da.Address, WithdrawAddress: address}
142         im.keeper.SubmitTx(ctx, []sdk.Msg{&msg}, da, "")
143     }

```

app/app.go, Lines 451-455

Listing 19: (Line 453)

```

451     participationrewardsModule := participationrewards.
    ↳ NewAppModule(appCodec, app.ParticipationRewardsKeeper)
452
453     app.InterchainQueryKeeper.SetCallbackHandler(
    ↳ participationrewardstypes.ModuleName, app.
    ↳ ParticipationRewardsKeeper.CallbackHandler())
454
455     app.AirdropKeeper = airdropkeeper.NewKeeper(

```

app/app.go, Lines 436-440

Listing 20: (Line 438)

```

436     interchainstakingIBCModule := interchainstaking.NewIBCModule(
    ↳ app.InterchainstakingKeeper)
437
438     app.InterchainQueryKeeper.SetCallbackHandler(
    ↳ interchainstakingtypes.ModuleName, app.InterchainstakingKeeper.
    ↳ CallbackHandler())
439
440     app.ParticipationRewardsKeeper = participationrewardskeeper.
    ↳ NewKeeper(

```

Risk Level:

Likelihood - 1

Impact - 1

Recommendation:

We recommend that the appropriate error checking be implemented to avoid unexpected crashes.

Remediation Plan:

SOLVED: The **Quicksilver Team** has solved this by introducing checks for errors from functions that can return errors.

3.4 (HAL-04) DUPLICATED ERROR CHECKS - INFORMATIONAL

Description:

There are two instances where an error check is not required, and the logic can be adjusted to only return the value.

Code Location:

[x/interchainstaking/keeper/receipt.go](#), Lines 211-216

Listing 21: (Line 212)

```
211     _, err = k.ICAControllerKeeper.SendTx(ctx, chanCap,  
    ↳ connectionID, portID, packetData, timeoutTimestamp)  
212     if err != nil {  
213         return err  
214     }  
215  
216     return nil
```

[x/interchainstaking/types/proposals.go](#), Lines 85-90

Listing 22: (Line 86)

```
85     err := govtypes.ValidateAbstract(m)  
86     if err != nil {  
87         return err  
88     }  
89  
90     return nil
```

Risk Level:

Likelihood - 1

Impact - 1

Recommendation:

Since the `err` variable will already be `nil` if the function has not generated any errors. An example of a piece of code that is sufficient:

Listing 23

```
1   err := govtypes.ValidateAbstract(m)
2   return nil
```

Remediation Plan:

SOLVED: The `Quicksilver Team` fixed the issue with the following commit [ID](#).

3.5 (HAL-05) COMMENTED OUT CODE / UNUSED CODE – INFORMATIONAL

Description:

It was found that there is some code that is currently not in use, and has been commented out.

Code Location:

[x/interchainstaking/keeper/receipt.go](#), Line 142-177

Listing 24

```

142 // func (k *Keeper) TransferToDelegateMulti(ctx sdk.Context, zone
    ↳ types.RegisteredZone, plan types.SendPlan, memo string) error {
143 //     eachAmount := sdk.Coins{}
144 //     splits := utils.MinU64(append([]uint64{}, k.GetParam(ctx,
    ↳ types.KeyDelegateAccountCount), uint64(len(zone.
    ↳ GetDelegationAccounts()))))
145
146 //     for _, asset := range inAmount {
147 //         thisAsset := sdk.Coin{Denom: asset.Denom, Amount: asset.
    ↳ Amount.Quo(sdk.NewIntFromUint64(splits))}
148 //         // TODO: maybe set this to some param based threshold?
    ↳ 5000 is an arbitrary figure to avoid distributing dust
    ↳ continuously.
149 //         // 5000 * 100 accounts == 0.5 tokens
150 //         if thisAsset.Amount.GT(sdk.NewInt(5000)) {
151 //             eachAmount = eachAmount.Add(thisAsset)
152 //         }
153 //     }
154
155 //     if eachAmount.Empty() || eachAmount.IsZero() {
156 //         splits = 1
157 //     }
158
159 //     in := []bankTypes.Input{}
160 //     out := []bankTypes.Output{}
161

```

```

162 //  in = append(in, bankTypes.Input{Address: zone.DepositAddress.
    ↳ GetAddress(), Coins: inAmount})
163
164 //  accounts := zone.GetDelegationAccountsByLowestBalance(splits)
165 //  for _, account := range accounts {
166 //      out = append(out, bankTypes.Output{Address: account.
    ↳ GetAddress(), Coins: eachAmount})
167 //      inAmount = inAmount.Sub(eachAmount)
168 //  }
169
170 //  // ensure any remainder gets deposited in the first account (
    ↳ as it will have the lowest balance)
171 //  out[0].Coins = out[0].Coins.Add(inAmount...)
172
173 //  msg := bankTypes.NewMsgMultiSend(in, out)
174 //  // send from deposit to accounts
175
176 //  return k.SubmitTx(ctx, []sdk.Msg{msg}, zone.DepositAddress,
    ↳ memo)
177 // }

```

Risk Level:**Likelihood - 1****Impact - 1****Recommendation:**

It is recommended to delete unused code instead of commenting it out.

Remediation Plan:

ACKNOWLEDGED: The [Quicksilver Team](#) has acknowledged this finding.

3.6 (HAL-06) PANIC IS USED FOR ERROR HANDLING – INFORMATIONAL

Description:

Several instances of the `panic` function were identified in the codebase. They appear to be used to handle errors. This can cause potential issues, as invoking a panic can cause the program to halt execution and crash in some cases. This in turn can negatively impact the availability of the software for users.

Code Location:

Listing 25

```

1 ./x/epochs/module.go:80:          panic(err)
2 ./x/epochs/keeper/keeper.go:31:    panic("cannot set epochs
↳ hooks twice")
3 ./x/epochs/simulation/genesis.go:41:    panic(err)
4 ./x/participationrewards/module.go:75:    panic(err)
5 ./x/participationrewards/keeper/keeper.go:45:    panic(fmt.
↳ Sprintf("%s module account has not been set", types.ModuleName))
6 ./x/participationrewards/keeper/rewards_validatorSelection.go:138:
↳    panic("this should never happen!")
7 ./x/participationrewards/keeper/distribution.go:213:
↳ panic("unable to obtain zone proportion on second zone pass")
8 ./x/interchainstaking/module.go:92:    panic(err)
9 ./x/interchainstaking/keeper/keeper.go:45:    panic(fmt.
↳ Sprintf("%s module account has not been set", types.ModuleName))
10 ./x/interchainstaking/keeper/zones.go:392:    panic("
↳ redemption with remaining amount:" + remainingTokens.String())
11 ./x/interchainstaking/keeper/receipt.go:114:    panic(err)
12 ./x/interchainstaking/keeper/receipt.go:119:    panic(err)
13 ./x/interchainstaking/keeper/msg_server.go:156:    panic(
↳ err) // panic here because something is terribly wrong if we cann'
↳ t find the delegation bucket here!!!
14 ./x/interchainstaking/keeper/intent.go:83:    panic(err)
15 ./x/interchainstaking/keeper/intent.go:100:    panic(err)
16 ./x/interchainstaking/keeper/intent.go:135:    panic(err)

```

```

17 ./x/interchainstaking/keeper/ibc_handlers.go:351:
    ↳ panic(err)
18 ./x/interchainstaking/keeper/ibc_handlers.go:412: panic("not
    ↳ implemented")
19 ./x/interchainstaking/types/delegation.go:35: panic(err)
20 ./x/interchainstaking/types/delegation.go:50: panic(err)
21 ./x/interchainstaking/types/delegation.go:58: panic(err)
22 ./x/interchainstaking/types/delegation.go:136: panic(err)
23 ./x/interchainstaking/types/delegation.go:151: panic(err)
24 ./x/interchainstaking/types/delegation.go:159: panic(err)
25 ./x/interchainstaking/types/params.go:44: panic(err)
26 ./x/mint/keeper/keeper.go:41: panic("the mint module
    ↳ account has not been set")
27 ./x/mint/keeper/keeper.go:71: panic("cannot set mint hooks
    ↳ twice")
28 ./x/mint/keeper/keeper.go:101: panic("stored minter should
    ↳ not have been nil")
29 ./x/mint/keeper/hooks.go:48: panic(err)
30 ./x/mint/keeper/hooks.go:54: panic(err)
31 ./x/interchainquery/module.go:84: panic(err)
32 ./x/interchainquery/keeper/keeper.go:132: panic(err)
    ↳ )
33 ./x/interchainquery/keeper/keeper.go:137: panic(err)
    ↳ )
34 ./x/airdrop/module.go:75: panic(err)
35 ./x/airdrop/genesis.go:25: panic(err)
36 ./x/airdrop/genesis.go:31: panic("insufficient airdrop
    ↳ module account balance")
37 ./x/airdrop/genesis.go:39: panic("zone sum not found")
38 ./x/airdrop/genesis.go:43: panic("zone sum does not
    ↳ match zone allocation")
39 ./x/airdrop/genesis.go:59: panic(err)
40 ./x/airdrop/keeper/keeper.go:37: panic(fmt.Sprintf("%s
    ↳ module account has not been set", types.ModuleName))
41 ./x/airdrop/keeper/abci.go:15: panic(err)
42

```

Risk Level:

Likelihood - 1

Impact - 1

Recommendation:

Instead of using panics, custom errors should be defined and handled according to the [Cosmos best practices](#).

Remediation Plan:

PARTIALLY SOLVED: The [Quicksilver Team](#) partially solved with the removing following [panics](#).

Listing 26

```

1 ./x/participationrewards/keeper/rewards_validatorSelection.go:138:
↳   panic("this should never happen!")
2 ./x/interchainstaking/keeper/zones.go:392:                panic("
↳ redemption with remaining amount:" + remainingTokens.String())
3 ./x/interchainstaking/keeper/receipt.go:114:             panic(err)
4 ./x/interchainstaking/keeper/receipt.go:119:             panic(err)
5 ./x/interchainstaking/keeper/intent.go:83:                panic(err)
6 ./x/interchainstaking/keeper/intent.go:100:               panic(err)
7 ./x/interchainstaking/keeper/intent.go:135:               panic(err)
8 ./x/interchainstaking/keeper/ibc_handlers.go:412:         panic("not
↳ implemented")
9 ./x/interchainstaking/types/delegation.go:35:             panic(err)
10 ./x/interchainstaking/types/delegation.go:50:            panic(err)
11 ./x/interchainstaking/types/delegation.go:58:            panic(err)
12 ./x/interchainstaking/types/delegation.go:136:           panic(err)
13 ./x/interchainstaking/types/delegation.go:151:           panic(err)
14 ./x/interchainstaking/types/delegation.go:159:           panic(err)
15 ./x/airdrop/keeper/abci.go:15:                            panic(err)

```

3.7 (HAL-07) OPEN TODOs - INFORMATIONAL

Description:

Open TODOs can point to architecture or programming issues that still need to be resolved. Often these kinds of comments indicate areas of complexity or confusion for developers. This provides value and insight to an attacker who aims to cause damage to the protocol.

Code Location:

Listing 27

```

1 ./x/epochs/module.go:193:      return nil // TODO
2 ./x/epochs/spec/07_future_improvements.md:15:TODO for postlaunch:
↳ We should see if we can architect things such that the receiver
↳ doesn't have to do this filtering, and the epochs module would pre
↳ -filter for them.
3 ./x/epochs/simulation/genesis.go:44:      // TODO: Do some
↳ randomization later
4 ./x/participationrewards/keeper/distribution.go:42:      // TODO
↳ : this needs to be verified as it currently does not trigger
↳ anymore
5 ./x/participationrewards/keeper/distribution.go:149:// TODO:
↳ remove when above is properly implemented
6 ./x/participationrewards/keeper/distribution.go:166:      //
↳ TODO: remove once allocateHoldingsRewards is implemented: >>>
7 ./x/participationrewards/keeper/hooks.go:18:      // TODO: implement
↳ and use this
8 ./x/participationrewards/keeper/hooks.go:23:      // TODO: remove
↳ this when the above is implemented
9 ./x/participationrewards/keeper/msg_server.go:13:      // TODO:
↳ implement
10 ./x/participationrewards/types/expected_keepers.go:13:      // TODO
↳ remove with genesis 2-phases refactor https://github.com/cosmos/cosmos-sdk/issues/2862
11 ./x/participationrewards/types/messages.go:36:      // TODO: check for
↳ valid zone (chain_id)

```

```

12 ./x/participationrewards/types/msgs.go:38:    // TODO: check for
    ↳ valid asset type (sdk.Coin)
13 ./x/interchainstaking/module.go:208:    return nil // TODO
14 ./x/interchainstaking/genesis.go:17:        // TODO: instantiate
    ↳ connections to ICAs based upon genesis values
15 ./x/interchainstaking/keeper/zones.go:166:    // TODO: figure
    ↳ out how this impacts delegations in progress / race conditions (in
    ↳ most cases, the duplicate delegation will just fail)
16 ./x/interchainstaking/keeper/delegation.go:171:
    ↳ delPlan = types.DelegationPlanFromUserIntent(zone, coin, valPlan)
    ↳ // TODO: does it make sense to do this? why don't we just use the
    ↳ global intent?
17 ./x/interchainstaking/keeper/callbacks.go:207:    // TODO: use
    ↳ pagination.GetTotal() to dispatch the correct number of requests
    ↳ now; rather than iteratively.
18 ./x/interchainstaking/keeper/receipt.go:148://        // TODO:
    ↳ maybe set this to some param based threshold? 5000 is an arbitrary
    ↳ figure to avoid distributing dust continuously.
19 ./x/interchainstaking/keeper/ibc_handlers.go:465:// TODO: this
    ↳ should be part of Keeper, but part of zone. Refactor me.
20 ./x/interchainstaking/types/genesis.go:16:    // TODO: validate
    ↳ genesis state.
21 ./x/interchainstaking/types/msgs.go:33:    // TODO: check from
    ↳ address
22 ./x/interchainstaking/types/msgs.go:119:    // TODO: check for
    ↳ valid chain_id
23 ./x/mint/spec/03_end_epoch.md:25:Calculate the provisions
    ↳ generated for each epoch based on current epoch provisions. The
    ↳ provisions are then minted by the `mint` module's `
    ↳ ModuleMinterAccount`. These rewards are transferred to a `
    ↳ FeeCollector`, which handles distributing the rewards per the
    ↳ chains needs. (See TODO.md for details) This fee collector is
    ↳ specified as the `auth` module's `FeeCollector` `ModuleAccount`.
24 ./x/mint/spec/02_state.md:38:**TODO:**
25 ./x/mint/types/expected_keepers.go:15:    // TODO remove with
    ↳ genesis 2-phases refactor https://github.com/cosmos/cosmos-sdk/
    ↳ issues/2862
26 ./x/mint/types/params.go:186:    // TODO: Maybe we should allow
    ↳ this :joy:, lets you burn osmo from community pool
27 ./x/interchainquery/module.go:197:    return nil // TODO
28 ./x/interchainquery/keeper/keeper.go:105:    if val.LocalHeight.LT
    ↳ (sdk.NewInt(ctx.BlockHeight() - int64(max_age))) { // this is
    ↳ somewhat arbitrary; TODO: make this better

```

```

29 ./x/interchainquery/keeper/abci.go:34:           // TODO: add
    ↳ height to request type
30 ./x/interchainquery/types/genesis.go:16:         // TODO: validate
    ↳ genesis state.
31 ./x/interchainquery/types/msgs.go:34:           // TODO: is there a chain
    ↳ validation spec in ICS?
32 ./x/interchainstaking/keeper/ibc_handlers.go:729: // todo: use
    ↳ multisend.
33 ./x/interchainstaking/keeper/ibc_handlers.go:730: // todo: this
    ↳ will probably not want to be an equal distribution. we want to
    ↳ use this to even out the distribution between accounts.
34 ./x/interchainstaking/keeper/ibc_handlers.go:729: // todo:
    ↳ use multisend.
35 ./x/interchainstaking/keeper/ibc_handlers.go:730: // todo:
    ↳ this will probably not want to be an equal distribution. we want
    ↳ to use this to even out the distribution between accounts

```

Risk Level:

Likelihood - 1

Impact - 1

Recommendation:

Consider resolving the To-dos before deploying code to a production context. Use an independent issue tracker or other project management software to track development tasks.

Remediation Plan:

PARTIALLY SOLVED: The [Quicksilver Team](#) partially solved with the resolving todos [panics](#).

Listing 28

```

1 ./x/participationrewards/keeper/distribution.go:42:           // TODO
  ↳ : this needs to be verified as it currently does not trigger
  ↳ anymore

```



```
2 ./x/participationrewards/keeper/distribution.go:149:// TODO:
↳ remove when above is properly implemented
3 ./x/participationrewards/keeper/msg_server.go:13:    // TODO:
↳ implement
4 ./x/participationrewards/types/msgs.go:36:    // TODO: check for
↳ valid zone (chain_id)
5 ./x/participationrewards/types/msgs.go:38:    // TODO: check for
↳ valid asset type (sdk.Coin)
6 ./x/interchainstaking/keeper/ibc_handlers.go:465:// TODO: this
↳ should be part of Keeper, but part of zone. Refactor me.
7 ./x/interchainstaking/types/msgs.go:33:    // TODO: check from
↳ address
8 ./x/interchainstaking/types/msgs.go:119:    // TODO: check for
↳ valid chain_id
9 ./x/mint/types/params.go:186:    // TODO: Maybe we should allow
↳ this :joy:, lets you burn osmo from community pool
```

3.8 (HAL-08) UNUSED VARIABLES - INFORMATIONAL

Description:

In the `x/interchainstaking/keeper/delegation.go` there is a variable `valPlan` that is declared on line 161, but never used.

Code Location:

`x/interchainstaking/keeper/delegation.go`, Line 160-162

Listing 29: (Line 161)

```
160         if coin.Denom == zone.BaseDenom {  
161             valPlan := make(types.ValidatorIntents)  
162             plan, found := k.GetIntent(ctx, zone, delegator, false  
    ↪ )
```

Risk Level:

Likelihood - 1

Impact - 1

Recommendation:

It is recommended that unused code be removed from the code base.

Remediation Plan:

SOLVED: The `Quicksilver Team` solved the issue with the following `commit`.

3.9 (HAL-09) INCORRECT COMMENTS IN CODE – INFORMATIONAL

Description:

In the `x/interchainstaking/client/cli/tx.go` there is a comment on line 71 that does not correspond to the function that follows it. The comment refers to `GetRegisterZoneTxCmd` and the function's name is `GetRequestRedemptionTxCmd`.

Code Location:

[`x/interchainstaking/client/cli/tx.go`, Line 70 <https://github.com/ingenuity-build/quicksilver/blob/v0.5.0/x/interchainstaking/client/cli/tx.go#L70>]

Listing 30: (Line 70)

```
70 // GetRegisterZoneTxCmd returns a CLI command handler for creating
   ↳ a MsgSend transaction.
71 func GetRequestRedemptionTxCmd() *cobra.Command {
```

Risk Level:

Likelihood - 1

Impact - 1

Recommendation:

It is recommended that the comments in the code correctly corresponds to the code.

Remediation Plan:

ACKNOWLEDGED: The `Quicksilver Team` has acknowledged this finding.



AUTOMATED TESTING



Description:

Halborn used automated testing techniques to enhance coverage of certain areas of the scoped component. Among the tools used were staticcheck, gosec, semgrep, unconvert, LGTM and Nancy. After Halborn verified all the contracts and scoped structures in the repository and was able to compile them correctly, these tools were leveraged on scoped structures. With these tools, Halborn can statically verify security related issues across the entire codebase.

Semgrep - Security Analysis Output Sample:

Listing 31: Rule Set

```

1 semgrep --config "p/dgryski.semgrep-go" ./ --exclude='*_test.go'
↳ --max-lines-per-finding 1000 --no-git-ignore -o dgryski.semgrep
2 semgrep --config "p/owasp-top-ten" ./ --exclude='*_test.go'
↳ --max-lines-per-finding 1000 --no-git-ignore -o owasp-top-ten.
↳ semgrep
3 semgrep --config "p/r2c-security-audit" ./ --exclude='*_test.go'
↳ --max-lines-per-finding 1000 --no-git-ignore -o r2c-security-audit
↳ .semgrep
4 semgrep --config "p/r2c-ci" ./ --exclude='*_test.go'
↳ --max-lines-per-finding 1000 --no-git-ignore -o r2c-ci.semgrep
5 semgrep --config "p/ci" ./ --exclude='*_test.go'
↳ --max-lines-per-finding 1000 --no-git-ignore -o ci.semgrep
6 semgrep --config "p/golang" ./ --exclude='*_test.go'
↳ --max-lines-per-finding 1000 --no-git-ignore -o golang.semgrep
7 semgrep --config "p/trailofbits" ./ --exclude='*_test.go'
↳ --max-lines-per-finding 1000 --no-git-ignore -o trailofbits.
↳ semgrep

```

Semgrep Results:

Listing 32

```

1 quicksilver/client/docs/swagger-ui/swagger-ui-standalone-preset.
↳ js
2 javascript.browser.security.wildcard-postmessage-
↳ configuration.wildcard-postmessage-

```

```

3      configuration
4      The target origin of the window.postMessage() API is set
↳ to "*". This could allow for
5      information disclosure due to the possibility of any
↳ origin allowed to receive the message.
6      Details: https://sg.run/PJ4p
7
8      2701 f.postMessage(t + "", "*")
9      -----
10     10847 }, t.postMessage("", "*"), t.onmessage = n, e
11     -----
12     10863 t.postMessage(u + e, "*")
13     -----
14     javascript.lang.correctness.useless-eqeq.eqeq-is-bad
15     Detected a useless comparison operation `t == t` or `t !=
↳ t`. This operation is always true.
16     If testing for floating point NaN, use `math.isnan`, or `
↳ cmath.isnan` if the number is
17     complex.
18     Details: https://sg.run/Kl6n
19
20     1634 return t === e || t != t && e != e
21     -----
22     javascript.lang.correctness.useless-eqeq.eqeq-is-bad
23     Detected a useless comparison operation `e == e` or `e !=
↳ e`. This operation is always true.
24     If testing for floating point NaN, use `math.isnan`, or `
↳ cmath.isnan` if the number is
25     complex.
26     Details: https://sg.run/Kl6n
27
28     1634 return t === e || t != t && e != e
29     -----
30     javascript.lang.correctness.useless-eqeq.eqeq-is-bad
31     Detected a useless comparison operation `t == t` or `t !=
↳ t`. This operation is always true.
32     If testing for floating point NaN, use `math.isnan`, or `
↳ cmath.isnan` if the number is
33     complex.
34     Details: https://sg.run/Kl6n
35
36     3600 if (t === e || t != t && e != e) return !0;
37     -----
38     javascript.lang.correctness.useless-eqeq.eqeq-is-bad

```

```

39         Detected a useless comparison operation `e == e` or `e !=
↳ e`. This operation is always true.
40         If testing for floating point NaN, use `math.isnan`, or `
↳ cmath.isnan` if the number is
41         complex.
42         Details: https://sg.run/Kl6n
43
44         3600 if (t === e || t != t && e != e) return !0;
45         -----
46         javascript.lang.correctness.useless-eqeq.eqeq-is-bad
47         Detected a useless comparison operation `t == t` or `t !=
↳ t`. This operation is always true.
48         If testing for floating point NaN, use `math.isnan`, or `
↳ cmath.isnan` if the number is
49         complex.
50         Details: https://sg.run/Kl6n
51
52         3603 if ((t = t.valueOf()) === (e = e.valueOf()) || t != t
↳ && e != e) return !0;
53         -----
54         javascript.lang.correctness.useless-eqeq.eqeq-is-bad
55         Detected a useless comparison operation `e == e` or `e !=
↳ e`. This operation is always true.
56         If testing for floating point NaN, use `math.isnan`, or `
↳ cmath.isnan` if the number is
57         complex.
58         Details: https://sg.run/Kl6n
59
60         3603 if ((t = t.valueOf()) === (e = e.valueOf()) || t != t
↳ && e != e) return !0;
61         -----
62         javascript.lang.correctness.useless-eqeq.eqeq-is-bad
63         Detected a useless comparison operation `t == t` or `t !=
↳ t`. This operation is always true.
64         If testing for floating point NaN, use `math.isnan`, or `
↳ cmath.isnan` if the number is
65         complex.
66         Details: https://sg.run/Kl6n
67
68         3826 if (t != t || t === 1 / 0) return 0;
69         -----
70         javascript.lang.correctness.useless-eqeq.eqeq-is-bad
71         Detected a useless comparison operation `o == o` or `o !=
↳ o`. This operation is always true.

```

```

72         If testing for floating point NaN, use `math.isnan`, or `
↳ cmath.isnan` if the number is
73         complex.
74         Details: https://sg.run/Kl6n
75
76         4692 if (o != o || u != u) return qe(t.toSeq()).cacheResult
↳ (), e, n, r);
77         -----
78         javascript.lang.correctness.useless-eqeq.eqeq-is-bad
79         Detected a useless comparison operation `u == u` or `u !=
↳ u`. This operation is always true.
80         If testing for floating point NaN, use `math.isnan`, or `
↳ cmath.isnan` if the number is
81         complex.
82         Details: https://sg.run/Kl6n
83
84         4692 if (o != o || u != u) return qe(t.toSeq()).cacheResult
↳ (), e, n, r);
85         -----
86         javascript.lang.correctness.useless-eqeq.eqeq-is-bad
87         Detected a useless comparison operation `s == s` or `s !=
↳ s`. This operation is always true.
88         If testing for floating point NaN, use `math.isnan`, or `
↳ cmath.isnan` if the number is
89         complex.
90         Details: https://sg.run/Kl6n
91
92         4694 s == s && (a = s < 0 ? 0 : s);
93         -----
94         javascript.lang.correctness.useless-eqeq.eqeq-is-bad
95         Detected a useless comparison operation `n == n` or `n !=
↳ n`. This operation is always true.
96         If testing for floating point NaN, use `math.isnan`, or `
↳ cmath.isnan` if the number is
97         complex.
98         Details: https://sg.run/Kl6n
99
100        4817 return 0 === r && n !== e && (void 0 === n || null ===
↳ n || n != n) || r > 0
101        -----
102        javascript.lang.correctness.useless-eqeq.eqeq-is-bad
103        Detected a useless comparison operation `e == e` or `e !=
↳ e`. This operation is always true.

```



```

104         If testing for floating point NaN, use `math.isnan`, or `
↳ cmath.isnan` if the number is
105         complex.
106         Details: https://sg.run/Kl6n
107
108         5916 return e === n || (null == e || null == n || !i(e) &&
↳ !i(n) ? e !== e && n !== n : r(e, n, o, u, t, a))
109         -----
110         javascript.lang.correctness.useless-eqeq.eqeq-is-bad
111         Detected a useless comparison operation `n == n` or `n !=
↳ n`. This operation is always true.
112         If testing for floating point NaN, use `math.isnan`, or `
↳ cmath.isnan` if the number is
113         complex.
114         Details: https://sg.run/Kl6n
115
116         5916 return e === n || (null == e || null == n || !i(e) &&
↳ !i(n) ? e !== e && n !== n : r(e, n, o, u, t, a))
117         -----
118         javascript.lang.correctness.useless-eqeq.eqeq-is-bad
119         Detected a useless comparison operation `t == t` or `t !=
↳ t`. This operation is always true.
120         If testing for floating point NaN, use `math.isnan`, or `
↳ cmath.isnan` if the number is
121         complex.
122         Details: https://sg.run/Kl6n
123
124         6009 return t == t && !r(t)
125         -----
126         6118 ... t != t ? e.flowing && e.length ? e.buffer.head.
↳ data.length : e.length : (t > e.highWaterMark && (e.highWaterMark
↳ = function(t) { ...
127         [shortened a long line from output, adjust with --max-
↳ chars-per-line]
128         -----
129         javascript.lang.correctness.useless-eqeq.eqeq-is-bad
130         Detected a useless comparison operation `n == n` or `n !=
↳ n`. This operation is always true.
131         If testing for floating point NaN, use `math.isnan`, or `
↳ cmath.isnan` if the number is
132         complex.
133         Details: https://sg.run/Kl6n
134
135         6831 if (t && n !== n) {

```

```

136         -----
137         6973 if (t && n != n) {
138         -----
139         javascript.lang.correctness.useless-eqeq.eqeq-is-bad
140         Detected a useless comparison operation `t == t` or `t !=
        ↳ t`. This operation is always true.
141         If testing for floating point NaN, use `math.isnan`, or `
        ↳ cmath.isnan` if the number is
142         complex.
143         Details: https://sg.run/Kl6n
144
145         8621 return t === e ? 0 !== t || 1 / t == 1 / e : t != t &&
        ↳ e != e
146         -----
147         javascript.lang.correctness.useless-eqeq.eqeq-is-bad
148         Detected a useless comparison operation `e == e` or `e !=
        ↳ e`. This operation is always true.
149         If testing for floating point NaN, use `math.isnan`, or `
        ↳ cmath.isnan` if the number is
150         complex.
151         Details: https://sg.run/Kl6n
152
153         8621 return t === e ? 0 !== t || 1 / t == 1 / e : t != t &&
        ↳ e != e
154         -----
155         10356 return e == e ? n ? e - n : e : 0
156         -----
157         javascript.lang.correctness.useless-eqeq.eqeq-is-bad
158         Detected a useless comparison operation `t == t` or `t !=
        ↳ t`. This operation is always true.
159         If testing for floating point NaN, use `math.isnan`, or `
        ↳ cmath.isnan` if the number is
160         complex.
161         Details: https://sg.run/Kl6n
162
163         10363 return t ? (t = r(t)) === i || t === -i ? (t < 0 ? -1
        ↳ : 1) * o : t == t ? t : 0 : 0 === t ? t : 0
164         -----
165         11547 return t != t
166         -----
167         javascript.lang.security.audit.detect-non-literal-regexp.
        ↳ detect-non-literal-regexp
168         RegExp() called with a variable, this might allow an
        ↳ attacker to DOS your application with a

```

```

169         long-running regular expression.
170         Details: https://sg.run/gr65
171
172         3132 if (!new RegExp(e).test(t)) return "Value must follow
↳ pattern " + e
173         -----
174         7477 s = RegExp("^" + a + a + "*"),
175         -----
176         7478 c = RegExp(a + a + "*$"),
177         -----
178         7656 return void 0 !== i ? i.call(n, r) : new RegExp(n)[e](
↳ String(r))
179         -----
180         7674 return void 0 !== i ? i.call(n, r) : new RegExp(n)[e](
↳ String(r))
181         -----
182         7693 y = new RegExp(t.source, h + "g");
183         -----
184         7694 for (a || (r = new RegExp("^" + y.source + "$(?:!\\s)",
↳ h));
185         -----
186         9383 l = RegExp(i + "(?=" + i + ")|" + f + c, "g");
187         -----
188         9645 ... RegExp([f + "?" + u + "+(?:['](?:d|l|m|r|e|s|t|v|e
↳ ))?(?=" + [r, f, "$"].join("|") + ")", p + "+(?:['](?:D|L|L|M|RE|S|I
↳ T|VE))?(?=" + [r, f + l, "$"].join ...
189         [shortened a long line from output, adjust with --max-
↳ chars-per-line]
190         -----
191         9691 ... RegExp("^" + f.call(l).replace(/[\^\$].*+?()
↳ [\]\{\}|\]/g, "\\$&").replace(/hasOwnProperty|(\function).*?(?=\\()|
↳ for .+?(?=\\)]/g, "$1.*?") + "$"); ...
192         [shortened a long line from output, adjust with --max-
↳ chars-per-line]
193         -----
194         12810 return "/" === e[0] && (n && (r = n[1]), e = e.slice
↳ (1, e.length - r.length - 1)), new RegExp(e, r)
195         -----
196         javascript.lang.security.audit.prototype-pollution.prototype-
↳ pollution-loop.prototype-pollution-
197         loop
198         Possibility of prototype polluting function detected. By
↳ adding or modifying attributes of

```

```

199         an object prototype, it is possible to create attributes
    ↳ that exist on every object, or
200         replace critical attributes with malicious ones. This can
    ↳ be problematic if the software
201         depends on existence or non-existence of certain
    ↳ attributes, or uses pre-defined attributes
202         of object prototype (such as hasOwnProperty, toString or
    ↳ valueOf). Possible mitigations
203         might be: freezing the object prototype, using an object
    ↳ without prototypes (via
204         Object.create(null) ), blocking modifications of
    ↳ attributes that resolve to object
205         prototype, using Map instead of object.
206         Details: https://sg.run/w1DB
207
208         10305 t = t[p]
209
210
211     quicksilver/docker-compose.yml
212     yaml.docker-compose.security.no-new-privileges.no-new-
    ↳ privileges
213         Service 'quicksilver' allows for privilege escalation via
    ↳ setuid or setgid binaries. Add
214         'no-new-privileges:true' in 'security_opt' to prevent this
    ↳ .
215         Details: https://sg.run/0n8q
216
217         3 quicksilver:
218         -----
219         yaml.docker-compose.security.no-new-privileges.no-new-
    ↳ privileges
220         Service 'quicksilver2' allows for privilege escalation via
    ↳ setuid or setgid binaries. Add
221         'no-new-privileges:true' in 'security_opt' to prevent this
    ↳ .
222         Details: https://sg.run/0n8q
223
224         14 quicksilver2:
225         -----
226         yaml.docker-compose.security.no-new-privileges.no-new-
    ↳ privileges
227         Service 'quicksilver3' allows for privilege escalation via
    ↳ setuid or setgid binaries. Add

```



```

259         54 testzone1-4:
260             -----
261         yaml.docker-compose.security.no-new-privileges.no-new-
262         ↳ privileges
263         Service 'testzone2-1' allows for privilege escalation via
264         ↳ setuid or setgid binaries. Add
265         'no-new-privileges:true' in 'security_opt' to prevent this
266         ↳ .
267         Details: https://sg.run/0n8q
268
269         62 testzone2-1:
270             -----
271         yaml.docker-compose.security.no-new-privileges.no-new-
272         ↳ privileges
273         Service 'testzone2-2' allows for privilege escalation via
274         ↳ setuid or setgid binaries. Add
275         'no-new-privileges:true' in 'security_opt' to prevent this
276         ↳ .
277         Details: https://sg.run/0n8q
278
279         73 testzone2-2:
280             -----
281         yaml.docker-compose.security.no-new-privileges.no-new-
282         ↳ privileges
283         Service 'testzone2-3' allows for privilege escalation via
284         ↳ setuid or setgid binaries. Add
285         'no-new-privileges:true' in 'security_opt' to prevent this
286         ↳ .
287         Details: https://sg.run/0n8q
288
289         81 testzone2-3:
290             -----
291         yaml.docker-compose.security.no-new-privileges.no-new-
292         ↳ privileges
293         Service 'testzone2-4' allows for privilege escalation via
294         ↳ setuid or setgid binaries. Add
295         'no-new-privileges:true' in 'security_opt' to prevent this
296         ↳ .
297         Details: https://sg.run/0n8q
298
299         89 testzone2-4:
300             -----
301         yaml.docker-compose.security.no-new-privileges.no-new-
302         ↳ privileges

```

```

290     Service 'hermes' allows for privilege escalation via
    ↳ setuid or setgid binaries. Add 'no-new-
291     privileges:true' in 'security_opt' to prevent this.
292     Details: https://sg.run/0n8q
293
294     97 hermes:
295         -----
296     yaml.docker-compose.security.no-new-privileges.no-new-
    ↳ privileges
297     Service 'icq' allows for privilege escalation via setuid
    ↳ or setgid binaries. Add 'no-new-
298     privileges:true' in 'security_opt' to prevent this.
299     Details: https://sg.run/0n8q
300
301     112 icq:
302         -----
303     yaml.docker-compose.security.no-new-privileges.no-new-
    ↳ privileges
304     Service 'icq2' allows for privilege escalation via setuid
    ↳ or setgid binaries. Add 'no-new-
305     privileges:true' in 'security_opt' to prevent this.
306     Details: https://sg.run/0n8q
307
308     120 icq2:
309         -----
310     yaml.docker-compose.security.writable-filesystem-service.
    ↳ writable-filesystem-service
311     Service 'quicksilver' is running with a writable root
    ↳ filesystem. This may allow malicious
312     applications to download and run additional payloads, or
    ↳ modify container files. If an
313     application inside a container has to save something
    ↳ temporarily consider using a tmpfs. Add
314     'read_only: true' to this service to prevent this.
315     Details: https://sg.run/e4JE
316
317     3 quicksilver:
318         -----
319     yaml.docker-compose.security.writable-filesystem-service.
    ↳ writable-filesystem-service
320     Service 'quicksilver2' is running with a writable root
    ↳ filesystem. This may allow malicious
321     applications to download and run additional payloads, or
    ↳ modify container files. If an

```

```

322         application inside a container has to save something
    ↳ temporarily consider using a tmpfs. Add
323         'read_only: true' to this service to prevent this.
324         Details: https://sg.run/e4JE
325
326         14 quicksilver2:
327         -----
328         yaml.docker-compose.security.writable-filesystem-service.
    ↳ writable-filesystem-service
329         Service 'quicksilver3' is running with a writable root
    ↳ filesystem. This may allow malicious
330         applications to download and run additional payloads, or
    ↳ modify container files. If an
331         application inside a container has to save something
    ↳ temporarily consider using a tmpfs. Add
332         'read_only: true' to this service to prevent this.
333         Details: https://sg.run/e4JE
334
335         22 quicksilver3:
336         -----
337         yaml.docker-compose.security.writable-filesystem-service.
    ↳ writable-filesystem-service
338         Service 'testzone1-1' is running with a writable root
    ↳ filesystem. This may allow malicious
339         applications to download and run additional payloads, or
    ↳ modify container files. If an
340         application inside a container has to save something
    ↳ temporarily consider using a tmpfs. Add
341         'read_only: true' to this service to prevent this.
342         Details: https://sg.run/e4JE
343
344         30 testzone1-1:
345         -----
346         yaml.docker-compose.security.writable-filesystem-service.
    ↳ writable-filesystem-service
347         Service 'testzone1-2' is running with a writable root
    ↳ filesystem. This may allow malicious
348         applications to download and run additional payloads, or
    ↳ modify container files. If an
349         application inside a container has to save something
    ↳ temporarily consider using a tmpfs. Add
350         'read_only: true' to this service to prevent this.
351         Details: https://sg.run/e4JE
352

```



```

353         38 testzone1-2:
354             -----
355         yaml.docker-compose.security.writable-filesystem-service.
356             ↳ writable-filesystem-service
357         Service 'testzone1-3' is running with a writable root
358             ↳ filesystem. This may allow malicious
359             ↳ applications to download and run additional payloads, or
360             ↳ modify container files. If an
361             ↳ application inside a container has to save something
362             ↳ temporarily consider using a tmpfs. Add
363             ↳ 'read_only: true' to this service to prevent this.
364             ↳ Details: https://sg.run/e4JE
365
366         46 testzone1-3:
367             -----
368         yaml.docker-compose.security.writable-filesystem-service.
369             ↳ writable-filesystem-service
370         Service 'testzone1-4' is running with a writable root
371             ↳ filesystem. This may allow malicious
372             ↳ applications to download and run additional payloads, or
373             ↳ modify container files. If an
374             ↳ application inside a container has to save something
375             ↳ temporarily consider using a tmpfs. Add
376             ↳ 'read_only: true' to this service to prevent this.
377             ↳ Details: https://sg.run/e4JE
378
379         54 testzone1-4:
380             -----
381         yaml.docker-compose.security.writable-filesystem-service.
382             ↳ writable-filesystem-service

```

```

383         Service 'testzone2-2' is running with a writable root
    ↳ filesystem. This may allow malicious
384         applications to download and run additional payloads, or
    ↳ modify container files. If an
385         application inside a container has to save something
    ↳ temporarily consider using a tmpfs. Add
386         'read_only: true' to this service to prevent this.
387         Details: https://sg.run/e4JE
388
389         73 testzone2-2:
390         -----
391         yaml.docker-compose.security.writable-filesystem-service.
    ↳ writable-filesystem-service
392         Service 'testzone2-3' is running with a writable root
    ↳ filesystem. This may allow malicious
393         applications to download and run additional payloads, or
    ↳ modify container files. If an
394         application inside a container has to save something
    ↳ temporarily consider using a tmpfs. Add
395         'read_only: true' to this service to prevent this.
396         Details: https://sg.run/e4JE
397
398         81 testzone2-3:
399         -----
400         yaml.docker-compose.security.writable-filesystem-service.
    ↳ writable-filesystem-service
401         Service 'testzone2-4' is running with a writable root
    ↳ filesystem. This may allow malicious
402         applications to download and run additional payloads, or
    ↳ modify container files. If an
403         application inside a container has to save something
    ↳ temporarily consider using a tmpfs. Add
404         'read_only: true' to this service to prevent this.
405         Details: https://sg.run/e4JE
406
407         89 testzone2-4:
408         -----
409         yaml.docker-compose.security.writable-filesystem-service.
    ↳ writable-filesystem-service
410         Service 'hermes' is running with a writable root
    ↳ filesystem. This may allow malicious
411         applications to download and run additional payloads, or
    ↳ modify container files. If an

```

```

412         application inside a container has to save something
413         ↳ temporarily consider using a tmpfs. Add
414         'read_only: true' to this service to prevent this.
415         Details: https://sg.run/e4JE
416
417         97 hermes:
418         -----
419         yaml.docker-compose.security.writable-filesystem-service.
420         ↳ writable-filesystem-service
421         Service 'icq' is running with a writable root filesystem.
422         ↳ This may allow malicious
423         applications to download and run additional payloads, or
424         ↳ modify container files. If an
425         application inside a container has to save something
426         ↳ temporarily consider using a tmpfs. Add
427         'read_only: true' to this service to prevent this.
428         Details: https://sg.run/e4JE
429
430         112 icq:
431         -----
432         yaml.docker-compose.security.writable-filesystem-service.
433         ↳ writable-filesystem-service
434         Service 'icq2' is running with a writable root filesystem.
435         ↳ This may allow malicious
436         applications to download and run additional payloads, or
437         ↳ modify container files. If an
438         application inside a container has to save something
439         ↳ temporarily consider using a tmpfs. Add
440         'read_only: true' to this service to prevent this.
441         Details: https://sg.run/e4JE
442
443         120 icq2:
444         -----
445         quicksilver/scripts/config/hermes.toml
446         javascript.lang.security.detect-insecure-websocket.detect-
447         ↳ insecure-websocket
448         Insecure WebSocket Detected. WebSocket Secure (wss) should
449         ↳ be used for all WebSocket
450         connections.
451         Details: https://sg.run/GWyz
452
453         32 websocket_addr = 'ws://quicksilver:26657/websocket'
454         -----

```

```

445         53 websocket_addr = 'ws://testzone1-1:26657/websocket'
446         -----
447         74 websocket_addr = 'ws://testzone2-1:26657/websocket'
448
449
450     quicksilver/scripts/setup.sh
451         bash.lang.correctness.unquoted-expansion.unquoted-command-
452         ↳ substitution-in-command
453         The result of command substitution $(...) or `...`, if
454         ↳ unquoted, is split on whitespace or
455         other separators specified by the IFS variable. You should
456         ↳ surround it with double quotes to
457         avoid splitting the result.
458         Details: https://sg.run/gYK5
459
460         171 sed -i 's/stake/uqck/g' $(pwd)/${CHAIN_DIR}/${
461         ↳ CHAINID_0}/config/genesis.json
462         -----
463         172 sed -i 's/stake/uqck/g' $(pwd)/${CHAIN_DIR}/${
464         ↳ CHAINID_0}a/config/genesis.json
465         -----
466         173 sed -i 's/stake/uqck/g' $(pwd)/${CHAIN_DIR}/${
467         ↳ CHAINID_0}b/config/genesis.json
468         -----
469         175 sed -i 's/stake/uatom/g' $(pwd)/${CHAIN_DIR}/${
470         ↳ CHAINID_1}/config/genesis.json
471         -----
472         176 sed -i 's/stake/uatom/g' $(pwd)/${CHAIN_DIR}/${
473         ↳ CHAINID_1}a/config/genesis.json
474         -----
475         177 sed -i 's/stake/uatom/g' $(pwd)/${CHAIN_DIR}/${
476         ↳ CHAINID_1}b/config/genesis.json
477         -----
478         178 sed -i 's/stake/uatom/g' $(pwd)/${CHAIN_DIR}/${
479         ↳ CHAINID_1}c/config/genesis.json
480         -----
481         181 sed -i 's/stake/uosmo/g' $(pwd)/${CHAIN_DIR}/${
482         ↳ CHAINID_2}/config/genesis.json
483         -----
484         182 sed -i 's/stake/uosmo/g' $(pwd)/${CHAIN_DIR}/${
485         ↳ CHAINID_2}a/config/genesis.json
486         -----
487         183 sed -i 's/stake/uosmo/g' $(pwd)/${CHAIN_DIR}/${
488         ↳ CHAINID_2}b/config/genesis.json

```

```

476      -----
477      184 sed -i 's/stake/uosmo/g' $(pwd)/${CHAIN_DIR}/${
↳ CHAINID_2}c/config/genesis.json
478
479
480      quicksilver/x/interchainquery/types/query.pb.gw.go
481      trailofbits.go.questionable-assignment.questionable-
↳ assignment
482      Should `protoReq` be modified when an error could be
↳ returned?
483      Details: https://sg.run/qq6y
484
485      56 protoReq.ConnectionId, err = runtime.String(val)
486      -----
487      90 protoReq.ConnectionId, err = runtime.String(val)
488
489
490      quicksilver/x/interchainstaking/keeper/receipt.go
491      dgryski.semgrep-go.errnilcheck.err-nil-check
492      superfluous nil err check before return
493      Details: https://sg.run/5Qd6
494
495      220 if err != nil {
496          return err
497      }
498      223
499      224 return nil
500
501
502      quicksilver/x/interchainstaking/types/proposals.go
503      dgryski.semgrep-go.errnilcheck.err-nil-check
504      superfluous nil err check before return
505      Details: https://sg.run/5Qd6
506
507      84 if err != nil {
508          return err
509      }
510      87
511      88 return nil
512
513
514      quicksilver/x/interchainstaking/types/query.pb.gw.go
515      trailofbits.go.questionable-assignment.questionable-
↳ assignment

```

```

516         Should `protoReq` be modified when an error could be
    ↪ returned?
517         Details: https://sg.run/qq6y
518
519         88 protoReq.ChainId, err = runtime.String(val)
520         -----
521         115 protoReq.ChainId, err = runtime.String(val)
522         -----
523         142 protoReq.ChainId, err = runtime.String(val)
524         -----
525         153 protoReq.DelegatorAddress, err = runtime.String(val)
526         -----
527         180 protoReq.ChainId, err = runtime.String(val)
528         -----
529         191 protoReq.DelegatorAddress, err = runtime.String(val)
530         -----
531         222 protoReq.ChainId, err = runtime.String(val)
532         -----
533         256 protoReq.ChainId, err = runtime.String(val)
534         -----
535         294 protoReq.ChainId, err = runtime.String(val)
536         -----
537         305 protoReq.DelegatorAddress, err = runtime.String(val)
538         -----
539         339 protoReq.ChainId, err = runtime.String(val)
540         -----
541         350 protoReq.DelegatorAddress, err = runtime.String(val)
542         -----
543         388 protoReq.ChainId, err = runtime.String(val)
544         -----
545         399 protoReq.ValidatorAddress, err = runtime.String(val)
546         -----
547         433 protoReq.ChainId, err = runtime.String(val)
548         -----
549         444 protoReq.ValidatorAddress, err = runtime.String(val)
550         -----
551         482 protoReq.ChainId, err = runtime.String(val)
552         -----
553         516 protoReq.ChainId, err = runtime.String(val)
554
555
556     quicksilver/x/participationrewards/keeper/distribution.go
557     dgryski.semgrep-go.errtodo.err-todo
558     TODO in error handling code

```

```
559         Details: https://sg.run/GeNB
560
561         165 // TODO: remove once allocateHoldingsRewards is
    ↪ implemented: >>>
```

Gosec - Security Analysis Output Sample:

Listing 33

```

1 [x/interchainstaking/client/cli/tx.go:233] - G304 (CWE-22):
↳ Potential file inclusion via variable (Confidence: HIGH, Severity:
↳ MEDIUM)
2     232:
3   > 233:     contents, err := os.ReadFile(proposalFile)
4     234:     if err != nil {
5
6
7
8 [x/interchainstaking/client/cli/tx.go:164] - G304 (CWE-22):
↳ Potential file inclusion via variable (Confidence: HIGH, Severity:
↳ MEDIUM)
9     163:
10  > 164:     contents, err := os.ReadFile(proposalFile)
11     165:     if err != nil {
12
13
14
15 [x/interchainstaking/keeper/zones.go:412] - G104 (CWE-703): Errors
↳ unhandled. (Confidence: HIGH, Severity: LOW)
16     411:         // if zero balance, retrigger the query.
17   > 412:         k.EmitPerformanceBalanceQuery(ctx, &zone)
18     413:         k.Logger(ctx).Info("performance account has a zero
↳ balance; requerying")
19
20
21
22 [x/interchainstaking/keeper/ibc_handlers.go:595] - G104 (CWE-703):
↳ Errors unhandled. (Confidence: HIGH, Severity: LOW)
23     594:         k.SetDelegation(ctx, zone, delegation)
24   > 595:         k.EmitValsetRequery(ctx, zone.ConnectionId, zone.
↳ ChainId)
25     596:         k.SetRegisteredZone(ctx, *zone)
26
27
28
29 [x/interchainstaking/keeper/ibc_handlers.go:551] - G104 (CWE-703):
↳ Errors unhandled. (Confidence: HIGH, Severity: LOW)
30     550:
31   > 551:         k.RemoveDelegation(ctx, zone, existingDelegation)

```



```

32     552:         da.DelegatedBalance = da.DelegatedBalance.Sub(
    ↳ existingDelegation.Amount) // remove old delegation from da.
    ↳ DelegatedBalance
33
34
35
36 [x/interchainstaking/keeper/ibc_handlers.go:356] - G104 (CWE-703):
    ↳ Errors unhandled. (Confidence: HIGH, Severity: LOW)
37     355:
38     > 356:             k.EmitValsetRequery(ctx, zone.
    ↳ ConnectionId, zone.ChainId)
39     357:
40
41
42
43 [x/interchainstaking/keeper/ibc_handlers.go:326] - G104 (CWE-703):
    ↳ Errors unhandled. (Confidence: HIGH, Severity: LOW)
44     325:     for _, delegationPlan := range toDelete {
45     > 326:         k.RemoveDelegationPlan(ctx, zone, memo,
    ↳ delegationPlan)
46     327:     }
47
48
49
50 [x/interchainstaking/keeper/ibc_handlers.go:245] - G104 (CWE-703):
    ↳ Errors unhandled. (Confidence: HIGH, Severity: LOW)
51     244:         for _, delegationPlan := range toDelete {
52     > 245:             k.RemoveDelegationPlan(ctx, zone, memo,
    ↳ delegationPlan)
53     246:     }
54
55
56
57 [x/interchainstaking/keeper/callbacks.go:159] - G104 (CWE-703):
    ↳ Errors unhandled. (Confidence: HIGH, Severity: LOW)
58     158:         if delegation, ok := k.GetDelegation(ctx, &zone,
    ↳ delegatorAddress, validatorAddress); ok {
59     > 159:             k.RemoveDelegation(ctx, &zone, delegation)
60     160:             ica, err := zone.GetDelegationAccountByAddress
    ↳ (delegatorAddress)
61
62
63

```

```

64 [x/interchainstaking/keeper/callbacks.go:92] - G104 (CWE-703):
   ↳ Errors unhandled. (Confidence: HIGH, Severity: LOW)
65     91:    }
66   > 92:    SetValidatorForZone(k, ctx, zone, args)
67     93:    return nil
68
69
70
71 [x/interchainstaking/keeper/callbacks.go:82] - G104 (CWE-703):
   ↳ Errors unhandled. (Confidence: HIGH, Severity: LOW)
72     81:    }
73   > 82:    SetValidatorsForZone(k, ctx, zone, args)
74     83:    return nil
75
76
77
78 [x/interchainstaking/ibc_module.go:228] - G104 (CWE-703): Errors
   ↳ unhandled. (Confidence: HIGH, Severity: LOW)
79     227:    msg := distrTypes.MsgSetWithdrawAddress{
   ↳ DelegatorAddress: address, WithdrawAddress: zone.WithdrawalAddress
   ↳ .String()}
80   > 228:    im.keeper.SubmitTx(ctx, []sdk.Msg{&msg}, zone.
   ↳ PerformanceAddress, "")
81     229:    }
82
83
84
85 [x/interchainstaking/ibc_module.go:165] - G104 (CWE-703): Errors
   ↳ unhandled. (Confidence: HIGH, Severity: LOW)
86     164:    msg := distrTypes.MsgSetWithdrawAddress{
   ↳ DelegatorAddress: address, WithdrawAddress: zoneInfo.
   ↳ WithdrawalAddress.String()}
87   > 165:    im.keeper.SubmitTx(ctx, []sdk.Msg{&msg},
   ↳ account, "")
88     166:    }
89
90
91
92 [x/interchainstaking/ibc_module.go:142] - G104 (CWE-703): Errors
   ↳ unhandled. (Confidence: HIGH, Severity: LOW)
93     141:    msg := distrTypes.MsgSetWithdrawAddress{
   ↳ DelegatorAddress: da.Address, WithdrawAddress: address}
94   > 142:    im.keeper.SubmitTx(ctx, []sdk.Msg{&msg}, da, "
   ↳ ")

```

```

95     143:         }
96
97
98
99 [app/app.go:453] - G104 (CWE-703): Errors unchecked. (Confidence:
   ↳ HIGH, Severity: LOW)
100     452:
101   > 453:     app.InterchainQueryKeeper.SetCallbackHandler(
   ↳ participationrewardstypes.ModuleName, app.
   ↳ ParticipationRewardsKeeper.CallbackHandler())
102     454:
103
104
105
106 [app/app.go:438] - G104 (CWE-703): Errors unchecked. (Confidence:
   ↳ HIGH, Severity: LOW)
107     437:
108   > 438:     app.InterchainQueryKeeper.SetCallbackHandler(
   ↳ interchainstakingtypes.ModuleName, app.InterchainstakingKeeper.
   ↳ CallbackHandler())
109     439:

```

ineffassign - Security Analysis Output Sample:

Listing 34

```

1 x/interchainstaking/keeper/delegation.go:161 ineffectual
   ↳ assignment to valPlan

```

Staticcheck - Security Analysis Output Sample:

Listing 35

```

1 x/airdrop/keeper/keeper.go
2   (24, 2)  U1000  field feeCollectorName is unused
3
4 x/airdrop/keeper/zonedrop.go
5   (132, 2) S1008  should use 'return !bt.After(zd.StartTime)'
   ↳ instead of 'if bt.After(zd.StartTime) { return false }; return
   ↳ true'

```

```

6   (156, 2)  S1008  should use 'return !bt.Before(zd.StartTime.Add(
↳ zd.Duration).Add(zd.Decay))' instead of 'if bt.Before(zd.StartTime
↳ .Add(zd.Duration).Add(zd.Decay)) { return false }; return true'
7
8 x/airdrop/types/query.pb.gw.go
9   (16, 2)  SA1019  "github.com/golang/protobuf/descriptor" is
↳ deprecated: See the "google.golang.org/protobuf/reflect/
↳ protorelect" package for how to obtain an EnumDescriptor or
↳ MessageDescriptor in order to programatically interact with the
↳ protobuf type system.
10  (17, 2)  SA1019  "github.com/golang/protobuf/proto" is
↳ deprecated: Use the "google.golang.org/protobuf/proto" package
↳ instead.
11  (33, 9)  SA1019  descriptor.ForMessage is deprecated: Not all
↳ concrete message types satisfy the Message interface. Use
↳ MessageDescriptorProto instead. If possible, the calling code
↳ should be rewritten to use protobuf reflection instead. See
↳ package "google.golang.org/protobuf/reflect/protorelect" for
↳ details.
12
13 x/epochs/types/query.pb.gw.go
14  (16, 2)  SA1019  "github.com/golang/protobuf/descriptor" is
↳ deprecated: See the "google.golang.org/protobuf/reflect/
↳ protorelect" package for how to obtain an EnumDescriptor or
↳ MessageDescriptor in order to programatically interact with the
↳ protobuf type system.
15  (17, 2)  SA1019  "github.com/golang/protobuf/proto" is
↳ deprecated: Use the "google.golang.org/protobuf/proto" package
↳ instead.
16  (33, 9)  SA1019  descriptor.ForMessage is deprecated: Not all
↳ concrete message types satisfy the Message interface. Use
↳ MessageDescriptorProto instead. If possible, the calling code
↳ should be rewritten to use protobuf reflection instead. See
↳ package "google.golang.org/protobuf/reflect/protorelect" for
↳ details.
17
18 x/interchainquery/types/messages.pb.gw.go
19  (16, 2)  SA1019  "github.com/golang/protobuf/descriptor" is
↳ deprecated: See the "google.golang.org/protobuf/reflect/
↳ protorelect" package for how to obtain an EnumDescriptor or
↳ MessageDescriptor in order to programatically interact with the
↳ protobuf type system.
20  (17, 2)  SA1019  "github.com/golang/protobuf/proto" is
↳ deprecated: Use the "google.golang.org/protobuf/proto" package

```

```

↳ instead.
21 (33, 9) SA1019 descriptor.ForMessage is deprecated: Not all
↳ concrete message types satisfy the Message interface. Use
↳ MessageDescriptorProto instead. If possible, the calling code
↳ should be rewritten to use protobuf reflection instead. See
↳ package "google.golang.org/protobuf/reflect/protoreflect" for
↳ details.
22
23 x/interchainquery/types/query.pb.gw.go
24 (16, 2) SA1019 "github.com/golang/protobuf/descriptor" is
↳ deprecated: See the "google.golang.org/protobuf/reflect/
↳ protoreflect" package for how to obtain an EnumDescriptor or
↳ MessageDescriptor in order to programatically interact with the
↳ protobuf type system.
25 (17, 2) SA1019 "github.com/golang/protobuf/proto" is
↳ deprecated: Use the "google.golang.org/protobuf/proto" package
↳ instead.
26 (33, 9) SA1019 descriptor.ForMessage is deprecated: Not all
↳ concrete message types satisfy the Message interface. Use
↳ MessageDescriptorProto instead. If possible, the calling code
↳ should be rewritten to use protobuf reflection instead. See
↳ package "google.golang.org/protobuf/reflect/protoreflect" for
↳ details.
27
28 x/interchainstaking/ibc_module.go
29 (285, 57) SA1029 should not use built-in type string as key
↳ for value; define your own type to avoid collisions
30
31 x/interchainstaking/keeper/callbacks.go
32 (12, 2) ST1019 package "github.com/cosmos/cosmos-sdk/x/bank/
↳ types" is being imported more than once
33 (13, 2) other import of "github.com/cosmos/cosmos-
↳ sdk/x/bank/types"
34
35 x/interchainstaking/keeper/delegation.go
36 (161, 4) SA4006 this value of valPlan is never used
37
38 x/interchainstaking/keeper/keeper_test.go
39 (72, 57) SA1029 should not use built-in type string as key for
↳ value; define your own type to avoid collisions
40
41 x/interchainstaking/keeper/proposal_handler.go
42 (101, 10) ST1005 error strings should not be capitalized
43

```

```

44 x/interchainstaking/types/messages.pb.gw.go
45 (16, 2) SA1019 "github.com/golang/protobuf/descriptor" is
↳ deprecated: See the "google.golang.org/protobuf/reflect/
↳ protoreflect" package for how to obtain an EnumDescriptor or
↳ MessageDescriptor in order to programatically interact with the
↳ protobuf type system.
46 (17, 2) SA1019 "github.com/golang/protobuf/proto" is
↳ deprecated: Use the "google.golang.org/protobuf/proto" package
↳ instead.
47 (33, 9) SA1019 descriptor.ForMessage is deprecated: Not all
↳ concrete message types satisfy the Message interface. Use
↳ MessageDescriptorProto instead. If possible, the calling code
↳ should be rewritten to use protobuf reflection instead. See
↳ package "google.golang.org/protobuf/reflect/protoreflect" for
↳ details.
48
49 x/interchainstaking/types/query.pb.gw.go
50 (16, 2) SA1019 "github.com/golang/protobuf/descriptor" is
↳ deprecated: See the "google.golang.org/protobuf/reflect/
↳ protoreflect" package for how to obtain an EnumDescriptor or
↳ MessageDescriptor in order to programatically interact with the
↳ protobuf type system.
51 (17, 2) SA1019 "github.com/golang/protobuf/proto" is
↳ deprecated: Use the "google.golang.org/protobuf/proto" package
↳ instead.
52 (33, 9) SA1019 descriptor.ForMessage is deprecated: Not all
↳ concrete message types satisfy the Message interface. Use
↳ MessageDescriptorProto instead. If possible, the calling code
↳ should be rewritten to use protobuf reflection instead. See
↳ package "google.golang.org/protobuf/reflect/protoreflect" for
↳ details.
53
54 x/mint/types/query.pb.gw.go
55 (16, 2) SA1019 "github.com/golang/protobuf/descriptor" is
↳ deprecated: See the "google.golang.org/protobuf/reflect/
↳ protoreflect" package for how to obtain an EnumDescriptor or
↳ MessageDescriptor in order to programatically interact with the
↳ protobuf type system.
56 (17, 2) SA1019 "github.com/golang/protobuf/proto" is
↳ deprecated: Use the "google.golang.org/protobuf/proto" package
↳ instead.
57 (33, 9) SA1019 descriptor.ForMessage is deprecated: Not all
↳ concrete message types satisfy the Message interface. Use
↳ MessageDescriptorProto instead. If possible, the calling code

```

```

↳ should be rewritten to use protobuf reflection instead. See
↳ package "google.golang.org/protobuf/reflect/protorelect" for
↳ details.
58
59 x/participationrewards/keeper/callbacks.go
60 (9, 2) ST1019 package "github.com/ingenuity-build/
↳ quicksilver/x/interchainquery/types" is being imported more than
↳ once
61 (10, 2) other import of "github.com/ingenuity-build
↳ /quicksilver/x/interchainquery/types"
62
63 x/participationrewards/keeper/keeper_test.go
64 (72, 57) SA1029 should not use built-in type string as key for
↳ value; define your type to avoid collisions
65
66 x/participationrewards/types/messages.pb.gw.go
67 (16, 2) SA1019 "github.com/golang/protobuf/descriptor" is
↳ deprecated: See the "google.golang.org/protobuf/reflect/
↳ protorelect" package for how to obtain an EnumDescriptor or
↳ MessageDescriptor in order to programatically interact with the
↳ protobuf type system.
68 (17, 2) SA1019 "github.com/golang/protobuf/proto" is
↳ deprecated: Use the "google.golang.org/protobuf/proto" package
↳ instead.
69 (33, 9) SA1019 descriptor.ForMessage is deprecated: Not all
↳ concrete message types satisfy the Message interface. Use
↳ MessageDescriptorProto instead. If possible, the calling code
↳ should be rewritten to use protobuf reflection instead. See
↳ package "google.golang.org/protobuf/reflect/protorelect" for
↳ details.
70
71 x/participationrewards/types/query.pb.gw.go
72 (16, 2) SA1019 "github.com/golang/protobuf/descriptor" is
↳ deprecated: See the "google.golang.org/protobuf/reflect/
↳ protorelect" package for how to obtain an EnumDescriptor or
↳ MessageDescriptor in order to programatically interact with the
↳ protobuf type system.
73 (17, 2) SA1019 "github.com/golang/protobuf/proto" is
↳ deprecated: Use the "google.golang.org/protobuf/proto" package
↳ instead.
74 (33, 9) SA1019 descriptor.ForMessage is deprecated: Not all
↳ concrete message types satisfy the Message interface. Use
↳ MessageDescriptorProto instead. If possible, the calling code
↳ should be rewritten to use protobuf reflection instead. See

```

```
↳ package "google.golang.org/protobuf/reflect/protorelect" for  
↳ details.  
75
```




THANK YOU FOR CHOOSING

// HALBORN

