MythX

REPORT 640BE3FA31554F001B20EB50

Created	Sat Mar 11 2023 02:14:18 GMT+0000 (Coordinated Universal Time)
Number of analyses	1
User	637bc5fd8288ab39b9a0547b

REPORT SUMMARY

Analyses ID	Main source file	Detected vulnerabilities
d1fefab1-2e3d-4477-b298-6e4b12f08051	trcx.sol	12

Analysis d1fefab1-2e3d-4477-b298-6e4b12f08051 MythX

Started	Sat Mar 11 2023 02:14:23 GMT+0000 (Coordinated Universal Time)
Finished	Sat Mar 11 2023 03:00:08 GMT+0000 (Coordinated Universal Time)
Mode	Deep
Client Tool	Remythx
Main Source File	Trcx.Sol

DETECTED VULNERABILITIES

HIGH	MEDIUM	LOW
0	0	12

ISSUES

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file
trcx.sol
Locations

```
8  // OpenZeppelin Contracts (last updated v4.6.0) (utils/math/SafeMath.sol)
9  //SPDX-License-Identifier: MIT
10 pragma solidity ^0.8.0;
11
12 // CAUTION
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.1"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file
trcx.sol
Locations

```
238 // OpenZeppelin Contracts (last updated v4.8.0) (utils/Address.sol)
239
240 pragma solidity ^0.8.1;
241
242 /**
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

trcx.sol

Locations

```
485 // OpenZeppelin Contracts (last updated v4.8.0) (security/ReentrancyGuard.sol)
486
487 pragma solidity ^0.8.0;
488
489 /**
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

trcx.sol

Locations

```
557 // OpenZeppelin Contracts v4.4.1 (utils/Context.sol)
558
559 pragma solidity ^0.8.0;
560
561 /**
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

trcx.sol

Locations

```
584 // OpenZeppelin Contracts (last updated v4.7.0) (access/Ownable.sol)
585
586 pragma solidity ^0.8.0;
587
588
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

trcx.sol

Locations

```
669 // OpenZeppelin Contracts (last updated v4.7.0) (utils/escrow/Escrow.sol)
670
671 pragma solidity ^0.8.0;
672
673
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

trcx.sol

Locations

```
738 // OpenZeppelin Contracts (last updated v4.8.0) (security/PullPayment.sol)
739
740 pragma solidity ^0.8.0;
741
742
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

trcx.sol

Locations

```
814 // OpenZeppelin Contracts (last updated v4.6.0) (token/ERC20/IERC20.sol)
815
816 pragma solidity ^0.8.0;
817
818 /**
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

trcx.sol

Locations

```
899 // OpenZeppelin Contracts v4.4.1 (token/ERC20/extensions/IERC20Metadata.sol)
900
901 pragma solidity ^0.8.0;
902
903
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

trcx.sol

Locations

```
929 // OpenZeppelin Contracts (last updated v4.8.0) (token/ERC20/ERC20.sol)
930
931 pragma solidity ^0.8.0;
932
933
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.0"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

trcx.sol

Locations

```
1320 // OpenZeppelin Contracts (last updated v4.5.0) (token/ERC20/extensions/ERC20Burnable.sol)
1321
1322 pragma solidity ^0.8.0;
1323
1324
```

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.8.17"". It is recommended to specify a fixed compiler version to ensure that the bytecode produced does not vary between builds. This is especially important if you rely on bytecode-level verification of the code.

Source file

trcx.sol

Locations

```
1359
1360
1361 pragma solidity ^0.8.17;
1362
1363 /**
```