MythX

# REPORT 5F0748FC37ECB700196F584C

| | |
|---|---|
| Created | Thu Jul 09 2020 16:42:36 GMT+0000 (Coordinated Universal Time) |
| Number of analyses | 37 |
| User | sdosch2110@gmail.com |

## REPORT SUMMARY

| Analyses ID | Main source file | Detected vulnerabilities |
|---|---|---|
| 682b6194-cf8b-4950-b878-12369c5b8755 | /security-token/contracts/factories/constraintmodulefactory.sol | 0 |
| cdd9897f-46ad-47b0-910e-a622736b5d98 | /security-token/contracts/token/constrainable.sol | 3 |
| b28ea068-dbf2-4194-bf60-c80ac762f5ac | /security-token/node_modules/@openzeppelin/contracts/utils/reentrancyguard.sol | 2 |
| b667eadc-ca9f-44b9-b9c7-745befcbdad2 | /security-token/contracts/token/administrable.sol | 3 |
| 3e1db6ac-5225-47f8-883d-dd11949c7c62 | /security-token/contracts/token/erc1400erc20.sol | 0 |
| 4738f89e-bd9c-4438-80a8-3a3d61ac7e56 | /security-token/contracts/token/erc1400raw.sol | 0 |
| cfcbbd01-9248-4f5d-b4fa-67210aed0a39 | /security-token/contracts/erc1820/erc1820client.sol | 0 |
| 5db4be78-420e-456d-b640-1c10b0b40b34 | /security-token/contracts/erc1820/erc1820client.sol | 0 |
| c83b46d4-1587-4c5c-aa27-d416ffce8a10 | /security-token/contracts/gsn/gsnmodule.sol | 0 |
| b78c15e8-f78d-4083-ae16-88345b67cc21 | /security-token/node_modules/@openzeppelin/contracts/gsn/irelayrecipient.sol | 1 |
| 61dae313-504d-493a-86b6-48f6237594fa | /security-token/contracts/gsn/gsnrecipient.sol | 0 |
| 72d3c994-a958-4fa9-8350-179f9c06ff84 | /security-token/node_modules/@openzeppelin/contracts/gsn/context.sol | 1 |
| 3723e00d-1ad0-4a10-a0b2-298e5443f71d | /security-token/node_modules/@openzeppelin/contracts/gsn/irelayhub.sol | 1 |
| dee1f83e-4927-4658-8653-d64ab5723cc2 | /security-token/contracts/gsn/gsnable.sol | 3 |
| 76d6da25-c381-471d-9bda-ab9f9a246f4b | /security-token/contracts/interfaces/iadmin.sol | 0 |
| 42e637c9-f5d0-409a-bd5b-243385ccf639 | /security-token/contracts/interfaces/iconstrainable.sol | 0 |
| d9174bfb-6c9a-4173-a1d3-561809613e64 | /security-token/contracts/interfaces/iconstraintmodule.sol | 0 |
| 02ebfc8c-77e6-42f6-9339-c635e26db0fc | /security-token/contracts/interfaces/ierc1400.sol | 1 |
| e14a614e-67ea-4589-84bd-c89a0ebaccb6 | /security-token/contracts/interfaces/ierc1400capped.sol | 0 |

| | | |
|---|---|---|
| 72888326-8649-42f1-9907-c7b5f90dec79 | /security-token/contracts/interfaces/ierc1400partition.sol | 0 |
| 664fa8c5-4e25-42fa-b08d-5057be256525 | /security-token/contracts/interfaces/ierc1400raw.sol | 0 |
| 9466b5c1-12a4-4de6-bf59-b6205541d985 | /security-token/contracts/interfaces/igsnable.sol | 0 |
| 5f399725-ed3d-4f1a-a629-c9e02231299e | /security-token/node_modules/@openzeppelin/contracts/gsn/irelayrecipient.sol | 1 |
| 56c3828c-38b0-49dd-883a-43cfe0bcdc5c | /security-token/contracts/interfaces/isecuritytoken.sol | 0 |
| 2fe1f05e-d0dc-4adc-9298-342833830de0 | /security-token/contracts/interfaces/isecuritytokenpartition.sol | 0 |
| 2d5f36eb-d962-48ab-be30-fbb5d8d99399 | /security-token/contracts/constraints/offchainvalidatorconstraintmodule.sol | 0 |
| 81cf89c4-3343-4a06-b120-781289b24bac | /security-token/contracts/constraints/pauseconstraintmodule.sol | 1 |
| dabf0970-2d17-476a-ac87-dde4dde25ba4 | /security-token/contracts/token/securitytoken.sol | 0 |
| 27a054f2-77af-4806-8600-45d75de008da | /security-token/contracts/factories/securitytokenpartitionfactory.sol | 0 |
| 93fd5ace-29a5-4305-bce3-256de00584b4 | /security-token/node_modules/@openzeppelin/contracts/token/erc20/ierc20.sol | 1 |
| 2f8118ed-8240-4dff-8f51-91fe846b6755 | /security-token/contracts/token/securitytokenpartition.sol | 0 |
| 2dcaea90-05ba-4098-ac22-cd918d5e3df6 | /security-token/contracts/constraints/spendinglimitsconstraintmodule.sol | 5 |
| afd38bc0-fd3c-4c30-9c34-42e696bb9465 | /security-token/node_modules/@openzeppelin/contracts/math/safemath.sol | 1 |
| 2ec45903-e00e-4a24-879e-d3eb0cac1b3b | /security-token/contracts/constraints/timelockconstraintmodule.sol | 4 |
| 034bb51a-0c44-4edb-9d3c-08b33565d485 | /security-token/contracts/constraints/vestingperiodconstraintmodule.sol | 8 |
| 85bc8bc5-cdd6-4059-b56c-1b81515bc7e6 | /security-token/node_modules/@openzeppelin/contracts/math/safemath.sol | 1 |
| 57625b92-3a74-44a2-94c7-548a62044f8c | /security-token/contracts/constraints/whitelistconstraintmodule.sol | 1 |

# Analysis 682b6194-cf8b-4950-b878-12369c5b8755

MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:42:47 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:57:57 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Factories/Constraintmodulefactory.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

Analysis cdd9897f-46ad-47b0-910e-a622736b5d98 MythX

|  |  |
|---|---|
| Started | Thu Jul 09 2020 16:42:47 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:57:58 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Token/Constrainable.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 1 | 2 |

## ISSUES

### MEDIUM SWC-110

An assertion violation was triggered.

It is possible to trigger an assertion violation. Note that Solidity assert() statements should only be used to check invariants. Review the transaction trace generated for this issue and either make sure your program logic is correct, or use require() instead of assert() if your goal is to constrain user inputs or enforce preconditions. Remember to validate inputs from both callers (for instance, via passed arguments) and callees (for instance, via return values).

Source file

/security-token/contracts/gsn/gsnable.sol

Locations

```
*/
function setGSNMode(gsnMode mode) public override onlyGSNController {
_gsnMode = gsnMode(mode);
emit GSNModeSet(mode);
}
```

### LOW SWC-107

A call to a user-supplied address is executed.

An external message call to an address specified by the caller is executed. Note that the callee account might contain arbitrary code and could re-enter any function within this contract. Reentering the contract in an intermediate state may lead to unexpected behaviour. Make sure that no state modifications are executed after this call and/or reentrancy guards are in place.

Source file

/security-token/contracts/gsn/gsnrecipient.sol

Locations

```
virtual
{
IRelayHub(_relayHub).withdraw(amount, payee);
}
```

LOW

SWC-123

### Requirement violation.

A requirement was violated in a nested call and the call was reverted as a result. Make sure valid inputs are provided to the nested call (for instance, via passed arguments).

Source file

`/security-token/contracts/gsn/gsnrecipient.sol`

Locations

```
virtual
{
IRelayHub(_relayHub).withdraw(amount, payee);
}
```

# Analysis b28ea068-dbf2-4194-bf60-c80ac762f5ac

MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:42:47 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:42:51 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Node_modules/@Openzeppelin/Contracts/Utils/Reentrancyguard.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 2 |

## ISSUES

LOW

SWC-103

### A floating pragma is set.

The current pragma Solidity directive is ""^0.6.0"". 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

/security-token/node_modules/@openzeppelin/contracts/utils/reentrancyguard.sol

Locations

```
pragma solidity ^0.6.0;

/**
```

LOW

SWC-131

### Unused state variable "_notEntered".

The state variable "_notEntered" is declared within the contract "ReentrancyGuard" but its value does not seem to be used anywhere.

Source file

/security-token/node_modules/@openzeppelin/contracts/utils/reentrancyguard.sol

Locations

```
*/
contract ReentrancyGuard {
bool private _notEntered;

constructor () internal {
```

# Analysis b667eadc-ca9f-44b9-b9c7-745befcbdad2

MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:42:47 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:57:56 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Token/Administrable.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 1 | 2 |

## ISSUES

### MEDIUM SWC-110

**An assertion violation was triggered.**

It is possible to trigger an assertion violation. Note that Solidity assert() statements should only be used to check invariants. Review the transaction trace generated for this issue and either make sure your program logic is correct, or use require() instead of assert() if your goal is to constrain user inputs or enforce preconditions. Remember to validate inputs from both callers (for instance, via passed arguments) and callees (for instance, via return values).

Source file

/security-token/contracts/gsn/gsnable.sol

Locations

```
*/
function setGSNMode(gsnMode mode) public override onlyGSNController {
_gsnMode = gsnMode(mode);
emit GSNModeSet(mode);
}
```

### LOW SWC-107

**A call to a user-supplied address is executed.**

An external message call to an address specified by the caller is executed. Note that the callee account might contain arbitrary code and could re-enter any function within this contract. Reentering the contract in an intermediate state may lead to unexpected behaviour. Make sure that no state modifications are executed after this call and/or reentrancy guards are in place.

Source file

/security-token/contracts/gsn/gsnrecipient.sol

Locations

```
virtual
{
IRelayHub(_relayHub).withdraw(amount, payee);
}
```

LOW

SWC-123

### Requirement violation.

A requirement was violated in a nested call and the call was reverted as a result. Make sure valid inputs are provided to the nested call (for instance, via passed arguments).

Source file

`/security-token/contracts/gsn/gsnrecipient.sol`

Locations

```
virtual
{
IRelayHub(_relayHub).withdraw(amount, payee);
}
```

# Analysis `3e1db6ac-5225-47f8-883d-dd11949c7c62` MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:43:19 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:43:40 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | `/Security-Token/Contracts/Token/Erc1400erc20.Sol` |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

## Analysis `4738f89e-bd9c-4438-80a8-3a3d61ac7e56` MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:43:29 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:58:45 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | `/Security-Token/Contracts/Token/Erc1400raw.Sol` |

### DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

### ISSUES

# Analysis `cfcbbd01-9248-4f5d-b4fa-67210aed0a39` MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:58:01 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:13:12 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | `/Security-Token/Contracts/Erc1820/Erc1820client.Sol` |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

Analysis 5db4be78-420e-456d-b640-1c10b0b40b34 MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:58:11 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:58:12 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Erc1820/Erc1820client.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

# Analysis `c83b46d4-1587-4c5c-aa27-d416ffce8a10` MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:58:11 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:13:22 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | `/Security-Token/Contracts/Gsn/Gsnmodule.Sol` |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

## Analysis b78c15e8-f78d-4083-ae16-88345b67cc21 MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:58:21 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:58:22 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Node_modules/@Openzeppelin/Contracts/Gsn/Irelayrecipient.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 1 |

## ISSUES

LOW

SWC-103

### A floating pragma is set.

The current pragma Solidity directive is ""^0.6.0"". 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

/security-token/node_modules/@openzeppelin/contracts/gsn/irelayrecipient.sol

Locations

```
pragma solidity ^0.6.0;

/**
```

## Analysis 61dae313-504d-493a-86b6-48f6237594fa

MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:58:31 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:58:33 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Gsn/Gsnrecipient.Sol |

### DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

### ISSUES

# Analysis 72d3c994-a958-4fa9-8350-179f9c06ff84

MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:58:41 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:58:43 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Node_modules/@Openzeppelin/Contracts/Gsn/Context.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 1 |

## ISSUES

LOW

SWC-103

### A floating pragma is set.

The current pragma Solidity directive is ""^0.6.0"". 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

/security-token/node_modules/@openzeppelin/contracts/gsn/context.sol

Locations

```
pragma solidity ^0.6.0;

/*
```

# Analysis 3723e00d-1ad0-4a10-a0b2-298e5443f71d

MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:58:51 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:58:53 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Node_modules/@Openzeppelin/Contracts/Gsn/Irelayhub.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 1 |

## ISSUES

LOW

SWC-103

### A floating pragma is set.

The current pragma Solidity directive is ""^0.6.0"". 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

/security-token/node_modules/@openzeppelin/contracts/gsn/irelayhub.sol

Locations

```
pragma solidity ^0.6.0;

/**
```

Analysis dee1f83e-4927-4658-8653-d64ab5723cc2 MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:58:51 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:14:05 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Gsn/Gsnable.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 1 | 2 |

## ISSUES

### MEDIUM SWC-110 An assertion violation was triggered.

It is possible to trigger an assertion violation. Note that Solidity assert() statements should only be used to check invariants. Review the transaction trace generated for this issue and either make sure your program logic is correct, or use require() instead of assert() if your goal is to constrain user inputs or enforce preconditions. Remember to validate inputs from both callers (for instance, via passed arguments) and callees (for instance, via return values).

Source file

/security-token/contracts/gsn/gsnable.sol

Locations

```
113 */
114 function setGSNMode(gsnMode mode) public override onlyGSNController {
115 _gsnMode = gsnMode(mode);
116 emit GSNModeSet(mode);
117 }
```

### LOW SWC-107 A call to a user-supplied address is executed.

An external message call to an address specified by the caller is executed. Note that the callee account might contain arbitrary code and could re-enter any function within this contract. Reentering the contract in an intermediate state may lead to unexpected behaviour. Make sure that no state modifications are executed after this call and/or reentrancy guards are in place.

Source file

/security-token/contracts/gsn/gsnrecipient.sol

Locations

```
81 virtual
82 {
83 IRelayHub(_relayHub).withdraw(amount, payee);
84 }
```

LOW

SWC-123

### Requirement violation.

A requirement was violated in a nested call and the call was reverted as a result. Make sure valid inputs are provided to the nested call (for instance, via passed arguments).

Source file

`/security-token/contracts/gsn/gsnrecipient.sol`

Locations

```
virtual
{
IRelayHub(_relayHub).withdraw(amount, payee);
}
```

Analysis 76d6da25-c381-471d-9bda-ab9f9a246f4b MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:59:02 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:59:03 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Interfaces/Iadmin.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

Analysis 42e637c9-f5d0-409a-bd5b-243385ccf639 MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:59:12 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:59:12 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Interfaces/Iconstrainable.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

# Analysis `d9174bfb-6c9a-4173-a1d3-561809613e64` MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:59:22 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:59:22 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | `/Security-Token/Contracts/Interfaces/Iconstraintmodule.Sol` |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

## Analysis 02ebfc8c-77e6-42f6-9339-c635e26db0fc

MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:59:32 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:59:33 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Interfaces/Ierc1400.Sol |

### DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 1 |

### ISSUES

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.6.6"". 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

/security-token/contracts/interfaces/ierc1400.sol

Locations

```
pragma solidity ^0.6.6;

```

Analysis `e14a614e-67ea-4589-84bd-c89a0ebaccb6` MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:59:42 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:59:43 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | `/Security-Token/Contracts/Interfaces/Ierc1400capped.Sol` |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

Analysis 72888326-8649-42f1-9907-c7b5f90dec79 MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 16:59:52 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 16:59:53 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Interfaces/Ierc1400partition.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

# Analysis 664fa8c5-4e25-42fa-b08d-5057be256525 MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 17:00:02 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:00:03 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | `/Security-Token/Contracts/Interfaces/Ierc1400raw.Sol` |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

# Analysis 9466b5c1-12a4-4de6-bf59-b6205541d985 MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 17:00:12 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:00:13 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | `/Security-Token/Contracts/Interfaces/Igsnable.Sol` |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

# Analysis 5f399725-ed3d-4f1a-a629-c9e02231299e

MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 17:00:22 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:00:23 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Node_modules/@Openzeppelin/Contracts/Gsn/Irelayrecipient.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 1 |

## ISSUES

LOW

SWC-103

### A floating pragma is set.

The current pragma Solidity directive is ""^0.6.0"". 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

/security-token/node_modules/@openzeppelin/contracts/gsn/irelayrecipient.sol

Locations

```
pragma solidity ^0.6.0;

/**
```

# Analysis 56c3828c-38b0-49dd-883a-43cfe0bcdc5c MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 17:00:32 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:00:33 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Interfaces/Isecuritytoken.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

Analysis 2fe1f05e-d0dc-4adc-9298-342833830de0 MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 17:00:42 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:00:44 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Interfaces/Isecuritytokenpartition.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

Analysis 2d5f36eb-d962-48ab-be30-fbb5d8d99399 MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 17:00:52 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:00:53 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Constraints/Offchainvalidatorconstraintmodule.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

## Analysis 81cf89c4-3343-4a06-b120-781289b24bac

MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 17:01:02 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:16:13 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Constraints/Pauseconstraintmodule.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 1 |

## ISSUES

LOW

SWC-108

### State variable visibility is not set.

It is best practice to set the visibility of state variables explicitly. The default visibility for "_securityToken" is internal. Other possible visibility settings are public and private.

Source file

/security-token/contracts/constraints/pauseconstraintmodule.sol

Locations

```
* @dev Address of securityToken this ConstraintModule is used by
*/
ISecurityToken _securityToken;

/**
```

## Analysis dabf0970-2d17-476a-ac87-dde4dde25ba4 MythX

| | |
|---|---|
| Started | |
| Finished | Thu Jul 09 2020 17:13:19 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | `/Security-Token/Contracts/Token/Securitytoken.Sol` |

### DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

### ISSUES

# Analysis `27a054f2-77af-4806-8600-45d75de008da` MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 17:13:34 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:28:44 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | `/Security-Token/Contracts/Factories/Securitytokenpartitionfactory.Sol` |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

## Analysis 93fd5ace-29a5-4305-bce3-256de00584b4 MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 17:13:44 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:13:47 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Node_modules/@Openzeppelin/Contracts/Token/Erc20/Ierc20.Sol |

### DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 1 |

### ISSUES

LOW

SWC-103

A floating pragma is set.

The current pragma Solidity directive is ""^0.6.0"". 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

/security-token/node_modules/@openzeppelin/contracts/token/erc20/ierc20.sol

Locations

```
pragma solidity ^0.6.0;

/**
```

Analysis 2f8118ed-8240-4dff-8f51-91fe846b6755 MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 17:13:44 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:29:23 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Token/Securitytokenpartition.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 0 |

## ISSUES

## Analysis 2dcaea90-05ba-4098-ac22-cd918d5e3df6 MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 17:13:54 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:29:05 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Constraints/Spendinglimitsconstraintmodule.Sol |

### DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 1 | 0 | 4 |

### ISSUES

HIGH

SWC-101

**The arithmetic operator can overflow.**

It is possible to cause an integer overflow or underflow in the arithmetic operation.

Source file

/security-token/contracts/constraints/spendinglimitsconstraintmodule.sol

Locations

```
function deleteTimelock(uint256 index) public onlySpendingLimitsEditor {
require(_spendinglimits.length > index, "out of bounds");
_spendinglimits[index] = _spendinglimits[_spendinglimits.length - 1];
_spendinglimits.pop();
emit TimelockDeleted(index);
```

LOW

SWC-108

**State variable visibility is not set.**

It is best practice to set the visibility of state variables explicitly. The default visibility for "_securityToken" is internal. Other possible visibility settings are public and private.

Source file

/security-token/contracts/constraints/spendinglimitsconstraintmodule.sol

Locations

```
* @dev Address of securityToken this ConstraintModule is used by
*/
ISecurityToken _securityToken;

/**
```

LOW

SWC-116

## A control flow decision is made based on The block.timestamp environment variable.

The block.timestamp environment variable is used to determine a control flow decision. Note that the values of variables like coinbase, gaslimit, block number and timestamp are predictable and can be manipulated by a malicious miner. Also keep in mind that attackers know hashes of earlier blocks. Don't use any of those environment variables as sources of randomness and be aware that use of these variables introduces a certain level of trust into miners.

Source file

/security-token/contracts/constraints/spendinglimitsconstraintmodule.sol

Locations

```

// period has not ended => there has been at least 1 tx
if (now <= user.periodEnd) {
// accumulated amount plus the amount to be transferred exceeds the allowed amount
if (user.amount.add(value) > _spendinglimits[i].amountAllowed) {
invalid = true;
reason = "spending limit for this period reached";
code = hex"A8";
} else {
// accumulated amount plus the amount to be transferred does not exceed the allowed amount
// increase accumulated amount and leave periodEnd

// INFO no record keeping for view
// user.amount = user.amount.add(value);
this;
}
} else {
// period ended => no tx in the relevant timeperiod
if (value > _spendinglimits[i].amountAllowed) {
invalid = true;
reason = "spending limit for this period reached";
code = hex"A8";
} else {
// INFO no record keeping for view
// user.amount = value;
// user.periodEnd = _spendinglimits[i].periodLength.add(now);
this;
}
}
}
```

LOW

SWC-116

### A control flow decision is made based on The block.timestamp environment variable.

The block.timestamp environment variable is used to determine a control flow decision. Note that the values of variables like coinbase, gaslimit, block number and timestamp are predictable and can be manipulated by a malicious miner. Also keep in mind that attackers know hashes of earlier blocks. Don't use any of those environment variables as sources of randomness and be aware that use of these variables introduces a certain level of trust into miners.

Source file

`/security-token/contracts/constraints/spendinglimitsconstraintmodule.sol`

Locations

```

// period has not ended, there has been at least 1 tx
if (now <= user.periodEnd) {
// accumulated amount plus the amount to be transferred exceeds the allowed amount
if (user.amount.add(value) > _spendinglimits[i].amountAllowed) {
invalid = true;
reason = "A8 - spending limit for this period reached";
} else {
// accumulated amount plus the amount to be transferred does not exceed the allowed amount
// increase accumulated amount and leave periodEnd
user.amount = user.amount.add(value);
}
} else {
// period ended, no tx in the relevant timeperiod
if (value > _spendinglimits[i].amountAllowed) {
invalid = true;
reason = "A8 - spending limit for this period reached";
} else {
user.amount = value;
user.periodEnd = _spendinglimits[i].periodLength.add(now);
}
}
}
```

LOW

SWC-116

### A control flow decision is made based on The block.timestamp environment variable.

The block.timestamp environment variable is used to determine a control flow decision. Note that the values of variables like coinbase, gaslimit, block number and timestamp are predictable and can be manipulated by a malicious miner. Also keep in mind that attackers know hashes of earlier blocks. Don't use any of those environment variables as sources of randomness and be aware that use of these variables introduces a certain level of trust into miners.

Source file

`/security-token/node_modules/@openzeppelin/contracts/math/safemath.sol`

Locations

```
function add(uint256 a, uint256 b) internal pure returns (uint256) {
uint256 c = a + b;
require(c >= a, "SafeMath: addition overflow");

return c;
```

# Analysis afd38bc0-fd3c-4c30-9c34-42e696bb9465 MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 17:14:14 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:29:25 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Node_modules/@Openzeppelin/Contracts/Math/Safemath.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 1 |

## ISSUES

LOW

SWC-103

### A floating pragma is set.

The current pragma Solidity directive is ""^0.6.0"". 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

/security-token/node_modules/@openzeppelin/contracts/math/safemath.sol

Locations

```
pragma solidity ^0.6.0;

/**
```

# Analysis 2ec45903-e00e-4a24-879e-d3eb0cac1b3b MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 17:16:24 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:31:37 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Constraints/Timelockconstraintmodule.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 4 |

## ISSUES

LOW

SWC-108

### State variable visibility is not set.

It is best practice to set the visibility of state variables explicitly. The default visibility for "_securityToken" is internal. Other possible visibility settings are public and private.

Source file

/security-token/contracts/constraints/timelockconstraintmodule.sol

Locations

```
* @dev Address of securityToken this ConstraintModule is used by
*/
ISecurityToken _securityToken;

/**
```

LOW

SWC-108

### State variable visibility is not set.

It is best practice to set the visibility of state variables explicitly. The default visibility for "_timeLock" is internal. Other possible visibility settings are public and private.

Source file

/security-token/contracts/constraints/timelockconstraintmodule.sol

Locations

```
* @dev Until when the whole token is locked
*/
uint256 _timeLock;

/**
```

LOW

SWC-116

### A control flow decision is made based on The block.timestamp environment variable.

The block.timestamp environment variable is used to determine a control flow decision. Note that the values of variables like coinbase, gaslimit, block number and timestamp are predictable and can be manipulated by a malicious miner. Also keep in mind that attackers know hashes of earlier blocks. Don't use any of those environment variables as sources of randomness and be aware that use of these variables introduces a certain level of trust into miners.

Source file

/security-token/contracts/constraints/timelockconstraintmodule.sol

Locations

```
)
{
if (_timeLock > now) {
return (false, hex"A8", "", "A8 - partition is still locked");
} else if (_accountTimeLock[msg_sender] > now) {
return (false, hex"A8", "", "A8 - account is still locked");
} else {
return (true, code, extradata, "");
}
}
```

LOW

SWC-116

### A control flow decision is made based on The block.timestamp environment variable.

The block.timestamp environment variable is used to determine a control flow decision. Note that the values of variables like coinbase, gaslimit, block number and timestamp are predictable and can be manipulated by a malicious miner. Also keep in mind that attackers know hashes of earlier blocks. Don't use any of those environment variables as sources of randomness and be aware that use of these variables introduces a certain level of trust into miners.

Source file

/security-token/contracts/constraints/timelockconstraintmodule.sol

Locations

```
if (_timeLock > now) {
return (false, hex"A8", "", "A8 - partition is still locked");
} else if (_accountTimeLock[msg_sender] > now) {
return (false, hex"A8", "", "A8 - account is still locked");
} else {
return (true, code, extradata, "");
}
}
```

# Analysis 034bb51a-0c44-4edb-9d3c-08b33565d485 MythX

Started Thu Jul 09 2020 17:28:56 GMT+0000 (Coordinated Universal Time)

Finished Thu Jul 09 2020 17:44:09 GMT+0000 (Coordinated Universal Time)

Mode Standard

Client Tool Truffle

Main Source File /Security-Token/Contracts/Constraints/Vestingperiodconstraintmodule.Sol

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 8 |

## ISSUES

### LOW SWC-108

State variable visibility is not set.

It is best practice to set the visibility of state variables explicitly. The default visibility for "_securityToken" is internal. Other possible visibility settings are public and private.

Source file
/security-token/contracts/constraints/vestingperiodconstraintmodule.sol

Locations

```
* @dev Address of securityToken this ConstraintModule is used by
*/
ISecurityToken _securityToken;

/**
```

### LOW SWC-108

State variable visibility is not set.

It is best practice to set the visibility of state variables explicitly. The default visibility for "_vestingStart" is internal. Other possible visibility settings are public and private.

Source file
/security-token/contracts/constraints/vestingperiodconstraintmodule.sol

Locations

```
* @dev Time until vesting starts
*/
uint256 _vestingStart;

/**
```

LOW

SWC-108

### State variable visibility is not set.

It is best practice to set the visibility of state variables explicitly. The default visibility for "_vestedFraction" is internal. Other possible visibility settings are public and private.

Source file

/security-token/contracts/constraints/vestingperiodconstraintmodule.sol

Locations

```
* @dev Fraction vested after starting
*/
uint256 _vestedFraction;

/**
```

LOW

SWC-108

### State variable visibility is not set.

It is best practice to set the visibility of state variables explicitly. The default visibility for "_vestingRatio" is internal. Other possible visibility settings are public and private.

Source file

/security-token/contracts/constraints/vestingperiodconstraintmodule.sol

Locations

```
* @dev Fraction of tokens vested in 1 month
*/
uint256 _vestingRatio;

/**
```

LOW

SWC-108

### State variable visibility is not set.

It is best practice to set the visibility of state variables explicitly. The default visibility for "_amountSpentByUser" is internal. Other possible visibility settings are public and private.

Source file

/security-token/contracts/constraints/vestingperiodconstraintmodule.sol

Locations

```
* @dev Tracks amount already spent by users
*/
mapping(address => uint256) _amountSpentByUser;

/**
```

LOW

SWC-116

### A control flow decision is made based on The block.timestamp environment variable.

The block.timestamp environment variable is used to determine a control flow decision. Note that the values of variables like coinbase, gaslimit, block number and timestamp are predictable and can be manipulated by a malicious miner. Also keep in mind that attackers know hashes of earlier blocks. Don't use any of those environment variables as sources of randomness and be aware that use of these variables introduces a certain level of trust into miners.

Source file

`/security-token/contracts/constraints/vestingperiodconstraintmodule.sol`

Locations

```
{
// dormant Period not over
if (now < _vestingStart) {
return (false, hex"A8", "", "A8 - vesting has not started yet");
}
// dormant period is over
```

LOW

SWC-116

### A control flow decision is made based on The block.timestamp environment variable.

The block.timestamp environment variable is used to determine a control flow decision. Note that the values of variables like coinbase, gaslimit, block number and timestamp are predictable and can be manipulated by a malicious miner. Also keep in mind that attackers know hashes of earlier blocks. Don't use any of those environment variables as sources of randomness and be aware that use of these variables introduces a certain level of trust into miners.

Source file

`/security-token/node_modules/@openzeppelin/contracts/math/safemath.sol`

Locations

```
*/
function sub(uint256 a, uint256 b, string memory errorMessage) internal pure returns (uint256) {
require(b <= a, errorMessage);
uint256 c = a - b;
```

LOW

SWC-116

### A control flow decision is made based on The block.timestamp environment variable.

The block.timestamp environment variable is used to determine a control flow decision. Note that the values of variables like coinbase, gaslimit, block number and timestamp are predictable and can be manipulated by a malicious miner. Also keep in mind that attackers know hashes of earlier blocks. Don't use any of those environment variables as sources of randomness and be aware that use of these variables introduces a certain level of trust into miners.

Source file

`/security-token/node_modules/@openzeppelin/contracts/math/safemath.sol`

Locations

```

uint256 c = a * b;
require(c / a == b, "SafeMath: multiplication overflow");

return c;
```

# Analysis 85bc8bc5-cdd6-4059-b56c-1b81515bc7e6 MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 17:29:16 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:44:27 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Node_modules/@Openzeppelin/Contracts/Math/Safemath.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 1 |

## ISSUES

LOW

SWC-103

### A floating pragma is set.

The current pragma Solidity directive is ""^0.6.0"". 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

/security-token/node_modules/@openzeppelin/contracts/math/safemath.sol

Locations

```
pragma solidity ^0.6.0;

/**
```

# Analysis `57625b92-3a74-44a2-94c7-548a62044f8c` MythX

| | |
|---|---|
| Started | Thu Jul 09 2020 17:29:36 GMT+0000 (Coordinated Universal Time) |
| Finished | Thu Jul 09 2020 17:44:47 GMT+0000 (Coordinated Universal Time) |
| Mode | Standard |
| Client Tool | Truffle |
| Main Source File | /Security-Token/Contracts/Constraints/Whitelistconstraintmodule.Sol |

## DETECTED VULNERABILITIES

| HIGH | MEDIUM | LOW |
|---|---|---|
| 0 | 0 | 1 |

## ISSUES

LOW

SWC-108

### State variable visibility is not set.

It is best practice to set the visibility of state variables explicitly. The default visibility for "_securityToken" is internal. Other possible visibility settings are public and private.

Source file

/security-token/contracts/constraints/whitelistconstraintmodule.sol

Locations

```
* @dev Address of securityToken this ConstraintModule is used by
*/
ISecurityToken _securityToken;

/**
```