智能合约安全审计报告

[2021]

目录

1 前言

慢雾安全团队于 2021.05.31，收到 Mars 团队对 Mars Ecosystem 智能合约安全审计的申请，慢雾安全团队根据项目特点制定如下审计方案。

慢雾安全团队将采用“白盒为主，黑灰为辅”的策略，以最贴近真实攻击的方式，对项目进行安全审计。

慢雾科技项目测试方法：

测试方法	说明
黑盒测试	站在外部从攻击者角度进行安全测试。
灰盒测试	通过脚本工具对代码模块进行安全测试，观察内部运行状态，挖掘弱点。
白盒测试	基于项目的源代码，进行脆弱性分析和漏洞挖掘。

慢雾科技漏洞风险等级：

漏洞等级	说明
严重漏洞	严重漏洞会对项目的安全造成重大影响，强烈建议修复严重漏洞。
高危漏洞	高危漏洞会影响项目的正常运行，强烈建议修复高危漏洞。
中危漏洞	中危漏洞会影响项目的运行，建议修复中危漏洞。
低危漏洞	低危漏洞可能在特定场景中会影响项目的业务操作，建议项目方自行评估和考虑这些问题是否需要修复。
弱点	理论上存在安全隐患，但工程上极难复现。
增强建议	编码或架构存在更好的实践方法。

2 审计方法

慢雾安全团队智能合约安全审计流程包含两个步骤:

- 使用开源或内部自动化分析的工具对合约代码中常见的安全漏洞进行扫描和测试。
- 人工审计代码的安全问题，通过人工分析合约代码，发现代码中潜在的安全问题。

如下是合约代码审计过程中慢雾安全团队会重点审查的漏洞列表:

（其他未知的安全漏洞及审计项不包含在本次审计责任范围）

- 重入漏洞
- 重放漏洞
- 重排漏洞
- 短地址漏洞
- 拒绝服务漏洞
- 交易顺序依赖漏洞
- 条件竞争漏洞
- 权限控制漏洞
- 整数上溢/下溢漏洞
- 时间戳依赖漏洞
- 未声明的存储指针漏洞
- 算术精度误差漏洞
- tx.origin身份验证漏洞
- 假充值漏洞
- 变量覆盖漏洞
- Gas优化审计
- 恶意 Event 事件审计
- 冗余的回调函数
- 不安全的外部调用审计

- 函数状态变量可见性审计
- 业务逻辑缺陷审计
- 变量声明及作用域审计

3 项目概要

3.1 项目介绍

慢雾科技智能合约安全审计组：Mars Ecosystem，

类型：DeFi

模块：Bondingcurve + core + genesis + oracle + pcv + refs + redemption

代码行数：2206

复杂度：中等

工作量：11 个工作日，智能合约安全审计标准、相关说明及审计后证书查询：https://www.slowmist.com/service-smart-contract-security-audit.html

项目源码：

https://github.com/MarsEcosystem/mars-swap

初始审计commit: 313bf2b4ad943f7c9216aa7c33abfb33cda88551

最终审计commit: c0dad9789a9713ccdf6abfe56d4bddc344616746

由于审计范围内的合约仍然有导入未在范围内的模块，因此仍需对相关模块进行审计。

请确认以上项目地址是否正确。

出具报告前需要项目方在区块浏览器开源验证，并给到相关合约地址。

3.2 漏洞信息

如下是本次审计发现的漏洞及漏洞的修复状态信息：

NO	标题	漏洞类型	漏洞等级	漏洞状态
N1	initialize初始化重要参数未做权限限制可被抢先调用	权限控制攻击	低	已修复
N2	purchase、deposit等方法payable可能导致锁定用户的资产	不安全的外部调用	建议	已修复
N3	redeem方法可能被非预期调用	权限控制攻击	建议	已忽略
N4	launch操作建议增加调用权限	权限控制攻击	建议	已修复
N5	PCVController角色权限过大可通过withdraw取出资产	权限控制攻击	中	已修复
N6	getUSDMAmountGovernance循环过多时可能导致DoS	其它	建议	已忽略

4 审计详情

4.1 合约基础信息

如下是合约主网地址：

合约模块	合约地址
Core	0x00789Cfb69499c65ac9A3a68fb4917c9b4FcA2a7
XMS	0x7859B01BbF675d67Da8cD128a50D155cd881B576
IMO	0x243DDd2E42CEb93349E726e2367EDeC6339aba75

合约模块	合约地址
MarsSwapFactory	0x6f12482D9869303B998C54D91bCD8bCcba81f3bE
MarsSwapRouter	0xb68825C810E67D4e444ad5B9DeB55BA56A66e72D
MarsStake	0x3b550BBFaC32Ec434F858a8135fa17C40636583B
AirDrop	0x01D152fF991E76b6cb310387c07cAfdFda790a25
Governor (timelock)	0xC35a8BdBB93abFAb362aF6dC3383cD2c6aEA6cBc

4.2 函数可见性分析

在审计过程中，慢雾安全团队对核心合约的函数可见性进行分析，结果如下:

BondingLCurve			
Function Name	Visibility	Mutability	Modifiers
constructor	internal	can modify state	-
setIncentiveAmount	external	can modify state	onlyGovernor
setMaxBasisPointsFromPegLP	external	can modify state	onlyGovernor
setAllocation	external	can modify state	onlyGovernor
setOracle	external	can modify state	onlyGovernor
allocate	external	can modify state	postGenesis,whenNotPaused,validPriceRange
getCurrentPrice	public	-	-
getAmountOut	public	-	-
getTotalPCVHeld	public	-	-

BondingLCurve			
_purchase	internal	can modify state	-
_incentivize	internal	can modify state	-
_isValidPriceRange	internal	-	-
_ignoreUSDMSupplyCap	internal	-	-

BUSDBondingLCurve			
Function Name	Visibility	Mutability	Modifiers
constructor	public	can modify state	-
initialize	external	can modify state	initializer
setOracle	external	can modify state	onlyGovernor
getCurrentPrice	public	-	-
getAmountOut	public	-	-
getAmountIn	public	-	-
setFee	public	can modify state	onlyGovernor
purchase	external	payable	postGenesis,whenNotPaused,ensure
getTotalPCVHeld	public	-	-
_allocateSingle	internal	can modify state	-
_isValidPriceRange	internal	-	-
_ignoreUSDMSupplyCap	internal	-	-
setChainlink	external	can modify state	onlyGovernor

BNBBondingLCurve			
Function Name	Visibility	Mutability	Modifiers
constructor	public	can modify state	-
initialize	external	can modify state	initializer
getCurrentPrice	public	-	-
getAmountOut	public	-	-
getAmountIn	public	-	-
setFee	public	can modify state	onlyGovernor
purchase	external	payable	postGenesis,whenNotPaused,ensure
getTotalPCVHeld	public	-	-
_allocateSingle	internal	can modify state	-
_isValidPriceRange	internal	-	-
_ignoreUSDMSupplyCap	internal	-	-
setChainlink	external	can modify state	onlyGovernor

Permissions			
Function Name	Visibility	Mutability	Modifiers
constructor	public	can modify state	-
createRole	external	can modify state	onlyGovernor
grantMinter	external	can modify state	onlyGovernor
grantBurner	external	can modify state	onlyGovernor

Permissions			
grantPCVController	external	can modify state	onlyGovernor
grantGovernor	external	can modify state	onlyGovernor
grantGuardian	external	can modify state	onlyGovernor
revokeMinter	external	can modify state	onlyGovernor
revokeBurner	external	can modify state	onlyGovernor
revokePCVController	external	can modify state	onlyGovernor
revokeGovernor	external	can modify state	onlyGovernor
revokeGuardian	external	can modify state	onlyGovernor
revokeOverride	external	can modify state	onlyGuardian
isMinter	external	-	-
isBurner	external	-	-
isPCVController	external	-	-
isGovernor	public	-	-
isGuardian	public	-	-
_setupGovernor	internal	can modify state	-
_setupMinter	internal	can modify state	-
_setupBurner	internal	can modify state	-

Core			
Function Name	Visibility	Mutability	Modifiers

Core			
init	external	can modify state	initializer
setXMSSupportRatio	external	can modify state	onlyGovernor
setUSDM	external	can modify state	onlyGovernor
setXMS	external	can modify state	onlyGovernor
setGenesisGroup	external	can modify state	onlyGovernor
allocateXMS	external	can modify state	onlyGovernor
allocateToken	public	can modify state	onlyGovernor
approveXMS	public	can modify state	onlyGovernor
approveToken	public	can modify state	onlyGovernor
completeGenesisGroup	external	can modify state	-
getApprovedPairsLength	public	-	-
getApprovedContractsLength	public	-	-
setApprovedPairAndContract	public	can modify state	onlyGovernor
removeApprovedPairAndContract	public	can modify state	onlyGovernor
_setXMSSupportRatio	internal	can modify state	-
_setUSDM	internal	can modify state	-
_setXMS	internal	can modify state	-

IDO			
Function Name	Visibility	Mutability	Modifiers

IDO			
constructor	public	can modify state	-
deploy	external	can modify state	onlyGenesisGroup
removeLiquidity	external	can modify state	onlyGuardianOrGovernor

BUSDGenesisGroup			
Function Name	Visibility	Mutability	Modifiers
constructor	public	can modify state	-
initGenesis	external	can modify state	onlyGovernor
purchase	external	payable	duringTime
redeem	external	can modify state	-
launch	external	can modify state	nonContract,afterTime
emergencyExit	external	can modify state	-
getAmountsToRedeem	public	-	postGenesis
getAmountOut	public	-	-
_burnFrom	internal	can modify state	-
_usdmXMSExchangeRate	internal	-	-
_getEffectiveMGEN	internal	-	-
setChainlink	external	can modify state	onlyGovernor

CombinationOracle			
Function Name	Visibility	Mutability	Modifiers

CombinationOracle			
constructor	internal	can modify state	-
initialize	external	can modify state	initializer
isStale	public	-	-
canUpdate	public	-	-
update	external	can modify state	-
consult	public	-	-
setRouter	public	can modify state	onlyGovernor
getRouter	public	-	-
_initialize	internal	can modify state	-
_canUpdate	internal	-	-
_isStale	internal	-	-
_update	internal	can modify state	-
_consult	internal	-	-

BNBLastPriceOracle			
Function Name	Visibility	Mutability	Modifiers
constructor	public	can modify state	-
getLatestPrice	public	-	-

BUSDLastPriceOracle			
Function Name	Visibility	Mutability	Modifiers

BUSDLastPriceOracle			
constructor	public	can modify state	-
getLatestPrice	public	-	-

OracleIncentives			
Function Name	Visibility	Mutability	Modifiers
constructor	public	can modify state	-
updateXMSForUSDMMROracle	public	can modify state	whenNotPaused
updateXMSForUSDMSupplyCapOracle	public	can modify state	whenNotPaused
_incentivize	internal	can modify state	-
setIncentiveAmount	external	can modify state	onlyGovernor

SwapMiningOracle			
Function Name	Visibility	Mutability	Modifiers
constructor	public	can modify state	-
getPairsLength	external	-	-
setFactory	external	can modify state	onlyGovernor
setPeriod	external	can modify state	onlyGovernor
addPair	external	can modify state	onlyGovernor
removePair	external	can modify state	onlyGovernor
update	external	can modify state	-
consult	external	-	-

MarsSwapPairCombOracle			
Function Name	Visibility	Mutability	Modifiers
constructor	public	can modify state	-
setFactory	external	can modify state	onlyGovernor
setPeriod	external	can modify state	onlyGovernor
setConsultLeniency	external	can modify state	onlyGovernor
setAllowStaleConsults	external	can modify state	onlyGovernor
_initialize	internal	can modify state	-
_canUpdate	internal	-	-
_isStale	internal	-	-
_update	internal	can modify state	-
_consult	internal	-	-

PCVController			
Function Name	Visibility	Mutability	Modifiers
constructor	internal	can modify state	-
forceWithdraw	external	can modify state	onlyGovernor

PCVController			
setPCVDeposit	external	can modify state	onlyGovernor
_withdraw	internal	can modify state	-

BUSDUniswapPCVController			
Function Name	Visibility	Mutability	Modifiers
constructor	public	can modify state	-
depositLpMining	external	can modify state	onlyGovernor
harvest	external	can modify state	onlyGovernor
withdrawLpMining	external	can modify state	onlyGuardianOrGovernor
removeLiquidity	external	can modify state	onlyGuardianOrGovernor
_removeLiquidity	internal	can modify state	-
_harvest	internal	can modify state	-
_depositLpMining	internal	can modify state	-
_withdrawLpMining	internal	can modify state	-
setLpMiningMaster	external	can modify state	onlyGovernor

PCVVenusDeposit			
Function Name	Visibility	Mutability	Modifiers
constructor	internal	can modify state	-
withdraw	external	can modify state	onlyPCVController
leaveSupply	external	can modify state	onlyPCVController

PCVVenusDeposit			
harvest	external	can modify state	onlyPCVController
_supply	internal	can modify state	-
_leaveSupply	internal	can modify state	-
_harvest	internal	can modify state	-
_transferWithdrawn	internal	can modify state	-

BNBVenusPCVDeposit			
Function Name	Visibility	Mutability	Modifiers
constructor	public	can modify state	-
receive	external	payable	-
deposit	external	payable	postGenesis,whenNotPaused
_supply	internal	can modify state	-
_leaveSupply	internal	can modify state	-
_harvest	internal	can modify state	-
_transferWithdrawn	internal	can modify state	-

PCVSplitter			
Function Name	Visibility	Mutability	Modifiers
constructor	internal	can modify state	-
checkAllocation	public	-	-
getAllocation	public	-	-

PCVSplitter			
_allocateSingle	internal	can modify state	-
_setAllocation	internal	can modify state	-
_allocate	internal	can modify state	-

PCVUniswapDeposit			
Function Name	Visibility	Mutability	Modifiers
constructor	internal	can modify state	-
withdraw	external	can modify state	onlyPCVController
removeLiquidity	external	can modify state	onlyPCVController
harvest	external	can modify state	onlyPCVController
depositLpMining	external	can modify state	onlyPCVController
withdrawLpMining	external	can modify state	onlyPCVController
_addLiquidity	internal	can modify state	-
_removeLiquidity	internal	can modify state	-
_harvest	internal	can modify state	-
_depositLpMining	internal	can modify state	-
_withdrawLpMining	internal	can modify state	-
_getLpMiningPid	internal	-	-
_transferWithdrawn	internal	can modify state	-
setLpMiningMaster	external	can modify state	onlyPCVController

BUSDUniswapPCVDeposit			
Function Name	Visibility	Mutability	Modifiers
constructor	public	can modify state	-
deposit	external	payable	postGenesis,whenNotPaused
_getAmountUSDMToDeposit	internal	-	-
_addLiquidity	internal	can modify state	-
_removeLiquidity	internal	can modify state	-
_harvest	internal	can modify state	-
_depositLpMining	internal	can modify state	-
_withdrawLpMining	internal	can modify state	-
_transferWithdrawn	internal	can modify state	-
_getLpMiningPid	internal	-	-
setChainlink	external	can modify state	onlyGovernor

RedemptionUnit			
Function Name	Visibility	Mutability	Modifiers
constructor	internal	can modify state	-
fee	external	-	-
feePrecision	external	-	-
_purchase	internal	can modify state	-
purchase	external	payable	postGenesis,whenNotPaused,ensure

XMSRedemptionUnit			
Function Name	Visibility	Mutability	Modifiers
constructor	public	can modify state	-
getCurrentPrice	public	-	-
getAmountOut	public	-	-
getAmountIn	public	-	-
_purchase	internal	can modify state	-
getTotalAssetHeld	public	-	-
setFee	public	can modify state	onlyGovernor

UniAndOracleRef			
Function Name	Visibility	Mutability	Modifiers
constructor	internal	can modify state	-
setPair	external	can modify state	onlyGovernor
setRouter	external	can modify state	onlyGovernor
setFactory	external	can modify state	onlyGovernor
token	public	-	-
getReserves	public	-	-
liquidityOwned	public	-	-
_approveToken	internal	can modify state	-
_setupPair	internal	can modify state	-

UniAndOracleRef			
_setupRouter	internal	can modify state	-
_setupFactory	internal	can modify state	-
_isPair	internal	-	-
_getUniswapPrice	internal	-	-

CoreRef			
Function Name	Visibility	Mutability	Modifiers
constructor	internal	can modify state	-
setCore	external	can modify state	onlyGovernor
pause	public	can modify state	onlyGuardianOrGovernor
unpause	public	can modify state	onlyGuardianOrGovernor
core	public	-	-
usdm	public	-	-
xms	public	-	-
usdmBalance	public	-	-
xmsBalance	public	-	-
getUSDMAmountGovernance	public	-	-
_burnUSDMHeld	internal	can modify state	-
_mintUSDM	internal	can modify state	-

OracleRef

OracleRef			
Function Name	Visibility	Mutability	Modifiers
constructor	internal	can modify state	-
setXMSForUSDMMROracle	external	can modify state	onlyGovernor
setXMSForUSDMSupplyCapOracle	external	can modify state	onlyGovernor
invert	public	-	-
getXMSPrice	public	-	-
getUSDMSupplyCap	public	-	-
_setXMSForUSDMMROracle	internal	can modify state	-
_setXMSForUSDMSupplyCapOracle	internal	can modify state	-

UniRef			
Function Name	Visibility	Mutability	Modifiers
constructor	internal	can modify state	-
setPair	external	can modify state	onlyGovernor
setRouter	external	can modify state	onlyGovernor
setFactory	external	can modify state	onlyGovernor
token	public	-	-
getReserves	public	-	-
liquidityOwned	public	-	-
_approveToken	internal	can modify state	-

UniRef			
_setupPair	internal	can modify state	-
_setupRouter	internal	can modify state	-
_setupFactory	internal	can modify state	-
_isPair	internal	-	-
_getUniswapPrice	internal	-	-

4.3 漏洞详情

[N1] [低] initialize初始化重要参数未做权限限制可被抢先调用

漏洞类型: 权限控制攻击

详细内容

contracts/bondingcurve/BNBBondingLCurve.sol#53-59,

```
function initialize(address _wbnb, address _chainlink)
    external
    initializer
{
    wbnb = IERC20(_wbnb);
    chainlink = IChainlinkLastPriceOracle(_chainlink);
}
```

contracts/bondingcurve/BUSDBondingLCurve.sol#55-61,

```
function initialize(address _busd, address _chainlink)
    external
    initializer
{
    busd = IERC20(_busd);
    chainlink = IChainlinkLastPriceOracle(_chainlink);
}
```

此方法仅有一个initializer修饰，而initializer只会判断是否已初始化过，所以此方法可被任意人调用。

相似的问题点：

contracts/core/Core.sol#40-42，

```
function init() external override initializer {
    _setupGovernor(msg.sender);
}
```

contracts/oracle/CombinationOracle.sol#40-45，

```
function initialize(address _oracle, address[] memory _path)
    external
    initializer
{
    setRouter(_oracle, _path);
}
```

解决方案

增加onlyOwner / onlyGovernor等判断

漏洞状态

已修复

[N2] [建议] purchase、deposit等方法payable可能导致锁定用户的资产

漏洞类型: 不安全的外部调用

详细内容

contracts/genesis/BUSDGenesisGroup.sol#76-87，

```
function purchase(address to, uint256 value)
    external
    payable
    override
    duringTime
{
    require(value != 0, "BUSDGenesisGroup::purchase: No value sent");
```

```
    busd.transferFrom(msg.sender, address(this), value);
    _mint(to, value);

    emit Purchase(to, value);
}
```

支持payable但并没有具体使用。

contracts/pcv/BNBVenusPCVDeposit.sol#39，

```
receive() external payable {}
```

contracts/pcv/BUSDUniswapPCVDeposit.sol#40-54，

```
function deposit(uint256 busdAmount)
    external
    payable
    override
    postGenesis
    whenNotPaused
{
    busdAmount = busd.balanceOf(address(this)); // Include any BUSD dust from prior
LP

    _addLiquidity(busdAmount);

    _burnUSDMHeld(); // Burn any USDM dust from LP

    emit Deposit(msg.sender, busdAmount);
}
```

也是类似的问题。

解决方案

去掉payable。

漏洞状态

已修复；

[N3] [建议] redeem方法可能被非预期调用

漏洞类型: 权限控制攻击

详细内容

contracts/genesis/BUSDGenesisGroup.sol#91-132,

```
function redeem(address to) external override {
    (uint256 usdmAmount, uint256 genesisXMS, uint256 busdAmount) =
        getAmountsToRedeem(to);
    require(
        block.number > launchBlock,
        "BUSDGenesisGroup::redeem: No redeeming in launch block"
    );

    // Burn MGEN
    uint256 amountIn = balanceOf(to);
    _burnFrom(to, amountIn);

    // Send USDM and XMS and BUSD
    if (usdmAmount != 0) {
        usdm().transfer(to, usdmAmount);
    }
    if (genesisXMS != 0) {
        uint256 genesis20Percent = genesisXMS.mul(2).div(10);
        xms().transfer(to, genesis20Percent);
        address tokenTimelockDelegator =
            address(
                new StraightTokenTimelockDelegator(
                    address(xms()),
                    to,
                    3600 * 24 * 30 * 12
                )
            );
        tokenTimelockDelegators[to] = tokenTimelockDelegator;
        xms().transfer(
            tokenTimelockDelegator,
            genesisXMS.sub(genesis20Percent)
        );
        ILinearTokenTimelock(tokenTimelockDelegator).initialize(
            launchTimestamp
        );
    }
    if (busdAmount != 0) {
```

```
        busd.transfer(to, busdAmount);
    }

    emit Redeem(to, amountIn, usdmAmount, genesisXMS);
}
```

此redeem是在赎回to地址的资产，然而却可由其它人调起，可能非to用户自身的意愿。

解决方案

判断发起方是否与to地址一致

漏洞状态

已忽略；项目方：这个机制已经被其他知名的协议测试过了。

[N4] [建议] launch操作建议增加调用权限

漏洞类型: 权限控制攻击

详细内容

contracts/genesis/BUSDGenesisGroup.sol#135-180,

```
function launch() external override nonContract afterTime {
    // Complete Genesis
    core().completeGenesisGroup();
    launchBlock = block.number;
    launchTimestamp = block.timestamp;

    (totalEffectiveMGEN, supersuper) = _getEffectiveMGEN(totalSupply());

    uint256 endOfTime = uint256(-1);
    // Bonding curve purchase and PCV allocation
    bondingCurve.purchase(address(this), totalEffectiveMGEN, 0, endOfTime);
    bondingCurve.allocate();

    ido.deploy(_usdmXMSExchangeRate());

    // solhint-disable-next-line not-rely-on-time
    emit Launch(block.timestamp);
}
```

方法可被任意调用，可能出现非预期的行为，尤其launch有ido.deploy操作。

解决方案

如无必要建议增加控制权限

漏洞状态

已修复；

[N5] [中] PCVController角色权限过大可通过withdraw取出资产

漏洞类型: 权限控制攻击

详细内容

contracts/pcv/PCVVenusDeposit.sol#27-34，

```
function withdraw(address to, uint256 amountUnderlying)
    external
    override
    onlyPCVController
{
    _transferWithdrawn(to, amountUnderlying);
    emit Withdrawal(msg.sender, to, amountUnderlying);
}
```

contracts/pcv/BNBVenusPCVDeposit.sol#87-90，

```
function _transferWithdrawn(address to, uint256 amount) internal override {
    (bool success, ) = to.call{value: amount}("");
    require(success, "BNBVenusPCVDeposit::_transferWithdrawn: Transfer failed");
}
```

有onlyPCVController的权限限制，仍存在权限过大，存在私钥丢失后被攻击或相关人员作恶的可能。

解决方案

分散权限，移交给社区治理，增加timelock等。

漏洞状态

已修复；默认PCVController为空，添加或更改PCVController时为OnlyGovernor，已经设置Timelock

[N6] [建议] getUSDMAmountGovernance循环过多时可能导致DoS

漏洞类型: 其它

详细内容

contracts/refs/CoreRef.sol#157-176,

```
function getUSDMAmountGovernance()
    public
    view
    override
    returns (uint256 usdmAmount)
{
    address pair;
    address _contract;
    for (uint256 i; i < core().getApprovedPairsLength(); i++) {
        pair = core().approvedPairs(i);
        for (uint256 j; j < core().getApprovedContractsLength(pair); j++) {
            _contract = core().approvedContracts(pair, j);
            usdmAmount += core()
                .usdm()
                .balanceOf(pair)
                .mul(IERC20(pair).balanceOf(_contract))
                .div(IERC20(pair).totalSupply());
        }
    }
}
```

方法中有循环调用外部合约，当core().getApprovedPairsLength()过大时可能导致DoS，从而方法调用失败。

解决方案

控制getApprovedPairsLength()在合理范围内，或者分批执行。

漏洞状态

已忽略；项目方：由协议管理的在DEX上添加流动性的USDM资产种类有限，目前只有XMS/USDM和BUSD/USDM两个流动性对。

5 审计结果

审计编号	审计团队	审计日期	审计结果
0X002106110004	SlowMist Security Team	2021.05.31 - 2021.06.14	通过

总结：

慢雾安全团队采用人工结合内部工具对代码进行分析，审计期间发现了 1 个中危漏洞， 1 个低危漏洞， 4 个增强建议。其中 2 个增强建议暂时被忽略；其它所有漏洞均已修复。

6 声明

厦门慢雾科技有限公司(下文简称 ”慢雾”) 仅就本报告出具前项目方已经发生或存在的事实出具本报告，并就此承担相应责任。对于出具以后项目方发生或存在的未知漏洞及安全事件，慢雾无法判断其安全状况，亦不对此承担责任。本报告所作的安全审计分析及其他内容，仅基于信息提供者截至本报告出具时向慢雾提供的文件和资料(简称“已提供资料”)。慢雾假设: 已提供资料不存在缺失、被篡改、删减或隐瞒的情形。如已提供资料信息缺失、被篡改、删减、隐瞒或反映的情况与实际情况不符的，慢雾对由此而导致的损失和不利影响不承担任何责任，慢雾仅对该项目的安全情况进行约定内的安全审计并出具了本报告，慢雾不对该项目背景及其他情况进行负责。

官方网址

www.slowmist.com

电子邮箱

team@slowmist.com

微信公众号