Radio Caca

合約審計報告

VER 1.0

2022年6月10日

No. 2022061019080

項目總結

1. 項目介紹

ERC721L項目對ERC721標準進行安全鎖定機制的擴展。NFT所有者通過 setLockApprovalForAll() 或 lockApprove() 準予協力廠商鎖定NFT，而被授權的協力廠商通過lockFrom() 鎖定NFT。在鎖定期結束之前，鎖定的NFT不能夠被轉移。這使得NFT擁有者能夠在不轉移NFT所有權的情況下參與NFT質押項目。

openpfp-contracts項目是由Radio Caca(RACA) 孵化的NFT市場，發行$PFPcoin作為OpenPFP.com的治理代幣，總供應量為10億。不僅如此，OpenPFP.com會向頭部交易團隊提供United States of Mars(USM) 土地。OpenPFP.com支持ETH和穩定幣作為支付貨幣，並且2%的交易費將用於資助運營和（$PFPcoin/USDT）質押LP獎勵。

2. 審計詳情

項目名稱	Radio Caca	平台	N/A
通證名稱	N/A	通證代號	N/A
開始時間	2022年5月31日	語言	Solidity
結束時間	2022年6月8日	官網	N/A
Github	https://github.com/radiocaca/ERC721L/blob/417f4c4bb4bb20bd14aa04865384fb90da250975/contracts/NFTs/MatrixPlusBox.sol https://github.com/radiocaca/ERC721L/blob/417f4c4bb4bb20bd14aa04865384fb90da250975/contracts/EIP5058/ERC5058.sol https://github.com/radiocaca/openpfp-contracts/blob/7423cd516965b188c617be5364bddd01c0d78088/contracts/OpenPFPExchange.sol	白皮書	N/A

2022年6月10日

3. 審計範圍

ID	文件	SHA-256 checksum
ERC721L/contracts/EIP5058	ERC5058.sol	754F975F93F0E6F858ECC1FD868CA42C1D672019EB122EC8A6F1E94DD6F5F673
ERC721L/contracts/EIP5058	IERC5058.sol	F248F94FD4786CB86599741BB30B05611491976230D0D5130A895AD6B5B80D24
ERC721L/contracts/factory	IERC5058Factory.sol	C53ECD67874D5F818898885EF13659F12C2548BDBE1FA40758507568F6EDF706
ERC721L/contracts/factory	IERC721Bound.sol	532FAA8E50205ACB0CFD09834D802B488DCDF7F815FFE79DC4E872C5EAF68959
ERC721L/contracts/EIP5058Upgradeable	ERC721Lockable.sol	926BB23FC1E0ECB2A5F65DAE7BDAC11C12A97E8C106E9A9B49B0328E94D29A2D
ERC721L/contracts/EIP5058Upgradeable	IERC721Lockable.sol	105FBB55549E603176C7152A4F2D6D8EAD4CF68BE5B49A913A14F2D91BC60A96
ERC721L/contracts/EIP5058/extensions	ERC5058Bound.sol	4F923113D2C8CC28FC2F89C77063EAA95BBD50895F6832F5CFDE593578DC487D
ERC721L/contracts/NFTs	MatrixPlusBox.sol	179FA3ECC43AF42CC08F20A8648AFA1A4DBE44A391C0FCDED25A888E6981DF3D
ERC721L/contracts/utils	ERC721Attachable.sol	CEAB4F2F35D3C8B2362159A8713C58D781E0DFC0A09EADA635FD955E64D2AC83
ERC721L/contracts/utils	TokenWithdraw.sol	FA562B879727A3E5E77A19BCC4F46FFA484B27E9165B942E7B4757A5A0935B02
openpfp-contracts/contracts	OpenPFPExchange.sol	A2A52D13F315735122E99CC38305D061A6910E3976A2C6EDDD8BEAD4F5B57CFD
openpfp-contracts/contracts	CurrencyManager.sol	CFEE8CA18CB1F11FEA0D71B74ACE9C0CB533D4501E7566005CB95FBF32F804BC
openpfp-contracts/contracts	ExecutionManager.sol	392360B9A0583677773F940CAFCD37DA85A91B42CA9ACF22EC28DB2174358E0A
openpfp-contracts/contracts	RoyaltyFeeManager.sol	AEDCF4C4ACF9DAA7FE954C501FB630275B04A5577CBB5E0C485D5854A9987857
openpfp-contracts/contracts	TransferSelectorNFT.sol	236E95E82FE35162B2F95122E51E26DA859ED763DD473F5EDC19713023CDC77C
openpfp-contracts/contracts/interfaces	ICurrencyManager.sol	2C56DB1882DF93873F6C0D6FC5E8B9BB060D0F167FA2E8996255E5F6360419D2

ID	文件	SHA-256 checksum
openpfp-contracts/contracts/interfaces	IExecutionManager.sol	4F16AED94A69248306D307B0EBA26E355279C5367AB4E1B75F464F50B6B6F2FC
openpfp-contracts/contracts/interfaces	IExecutionStrategy.sol	3C331C764DCF6A2CB683CCAAB679D0669547D32502CAA0A16446BBB3824DC90A
openpfp-contracts/contracts/interfaces	IOpenPFPExchange.sol	237C4A31AE2E2AA93A116C30FEF3E29FEEF74A32A910D06717D86B263A6FBA2C
openpfp-contracts/contracts/interfaces	IRoyaltyFeeManager.sol	B9799D9FA53FD9A43EE819D3B37CEDBA93466D485F55816A707375635F7F8357
openpfp-contracts/contracts/interfaces	IRoyaltyFeeRegistry.sol	854CB3632E17DA1AD14EF32C344AFDCA4E9BF1A35541EF85907DF02883C45A9B
openpfp-contracts/contracts/interfaces	ITransferManagerNFT.sol	7A6A94CC3BA8ACF57E6BBA0EA1115E2B40327A8B8CFF799788AB3E37604F6509
openpfp-contracts/contracts/interfaces	ITransferSelectorNFT.sol	DC645A8C0CB6F3DE6597E88822026044CC1C973AA69900A4C1E09F3D686158EC
openpfp-contracts/contracts/interfaces	IWETH.sol	69007C90A776A8EFE8F5890BBAE1A6F9F8567A743603A99AF32B6087EC92E73D
openpfp-contracts/contracts/libraries	OrderTypes.sol	A2038E91D121922C2DA0A7B2F2EA104E6E64C662F944068F55AC731541C79F49
openpfp-contracts/contracts/libraries	SignatureChecker.sol	82A3FE05B58130C843CD2B4A8138D9E45694321BD118456595294B12AAB7BEFF
openpfp-contracts/contracts/royalty	RoyaltyFeeRegistry.sol	31DE9FAD57CDF308AE1EFD1691D140D69079EE82A055CBA05AB3F448598C6A09
openpfp-contracts/contracts/strategys	StrategyAnyItemFromCollectionForFixedPrice.sol	E730E946C7A680E0666FE5ACDF1A42DB77368E5127BD037ED853562E426BC969
openpfp-contracts/contracts/strategys	StrategyPrivateSale.sol	51D36BEEB4DE48383F43B57694D0FEF1C5FB43896193909F2F238F00ED121267
openpfp-contracts/contracts/strategys	StrategyStandardSaleForFixedPrice.sol	7172C3A854A570617814085B30C46C58083C1D98E0D97DA24E3AE03DFE46C157
openpfp-contracts/contracts/transfers	TransferManagerERC1155.sol	B0F2BEE1F35906307412C77E89E02AACE0CD11AD575E3EE02FC7283A79D59DFA
openpfp-contracts/contracts/transfers	TransferManagerERC721.sol	239F07AE00E5D58F64DE9522D381440F911E675D8D1D3C5C39A4F76188752422
openpfp-contracts/contracts/transfers	TransferManagerNonCompliantERC721.sol	B1B127F7C5CBAA723837F60F82C0397F6619FF34D29E5BE62F86AA36FC29EEF0

2022年6月10日

4. 代碼結構

```
# REC721L
 ├──EIP5058
 |    |   ERC5058.sol                          #可鎖定NFT合約
 |    |   IERC5058.sol                         #介面檔案
 |    ├──extensions
 |    |       ERC5058Bound.sol                 #可鎖定NFT擴展Bound代幣
 |    └──factory
 |            IERC5058Factory.sol              #介面檔案
 |            IERC721Bound.sol                 #介面檔案
 ├──EIP5058Upgradeable
 |      ERC721Lockable.sol                     #可鎖定NFT合約
 |      IERC721Lockable.sol                    #介面檔案
 ├──NFTs
 |      MatrixPlusBox.sol                      #可鎖定NFT的擴展代幣
 └──utils
       ERC721Attachable.sol                    #ERC721代幣擴展
       TokenWithdraw.sol                       #代幣交易
```

2022年6月10日

```
# openpfp-contracts
|   CurrencyManager.sol                                          #貨幣管理
|   ExecutionManager.sol                                         #策略管理
|   OpenPFPExchange.sol                                          #訂單簿訂單配對
|   RoyaltyFeeManager.sol                                        #特權費用管理
|   TransferSelectorNFT.sol                                      #特權費用管理
|
├──interfaces                                                    #介面檔案
|       ICurrencyManager.sol
|       IExecutionManager.sol
|       IExecutionStrategy.sol
|       IOpenPFPExchange.sol
|       IRoyaltyFeeManager.sol
|       IRoyaltyFeeRegistry.sol
|       ITransferManagerNFT.sol
|       ITransferSelectorNFT.sol
|       IWETH.sol
|
├──royalty
|       RoyaltyFeeRegistry.sol                                   #特權費用注册
|
├──strategys
|       StrategyAnyItemFromCollectionForFixedPrice.sol           #固定價格任意訂單策略
|       StrategyPrivateSale.sol                                  #特定地址策略
|       StrategyStandardSaleForFixedPrice.sol                    #固定價格任意買賣方策略
|
├──transfers
|       TransferManagerERC1155.sol                               #ERC1155轉帳函數
|       TransferManagerERC721.sol                                #ERC721轉帳函數
|       TransferManagerNonCompliantERC721.sol                    #ERC721非安全轉帳函數
|
└──libraries
        OrderTypes.sol                                           #訂單格式
        SignatureChecker.sol                                     #簽名檢查
```

2022年6月10日

審計報告匯總

1. 審計方式

通過清晰地理解該項目的設計目的、運行原理和實現管道，稽核團隊對合約程式碼進行了深入的研究和分析。 在分清各個合約及其函數的調用關係的基礎上，對合約可能存在的漏洞進行了定位及分析。 最終產生問題描述和給出相應的修改意見。

審計方法	Static analysis, Manual Review

2. 審計流程

步驟	操作	詳細內容
1	背景研究	閱讀項目介紹、白皮書、合約源碼等項目方團隊提供的相關信息，確保正確理解項目功能
2	自動化檢測	主要用自動化工具掃描源碼，找到常見的潛在漏洞
3	人工審閱	工程師逐行閱讀代碼，找到潛在漏洞
4	邏輯校對	工程師將對代碼的理解和項目方提供的信息比較，檢查代碼實現是否符合項目白皮書信息
5	測試用例檢測	包括測試用例設計，測試範圍分析、符號執行等
6	優化審查	根據應用場景、調用方式及最新的研究成果從可維護性、安全性及可操作性等方面審查項目

2022年6月10日

3. 風險分級

風險級別	風險描述
致命	存在致命風險及隱患，需要立即解決
高危	存在高危風險及隱患，將引發相同問題，必須解決
中度	存在中度風險及隱患，可能導致潛在風險，最終仍然需要解決
低危	存在低風險及隱患，指各類處理不當或會引發警告信息的細節，這類問題可暫時擱置
建議	存在可優化的部分，這類問題可以擱置，但建議最終解決

4. 審計結果

編號	審計項目	風險級別	狀態
1	重入	無	
2	注入	無	
3	權限繞過	無	
4	Mempool搶跑	無	
5	回滾	無	
6	條件競爭	無	
7	循環耗盡gas	無	
8	閃電貸高影響	無	
9	經濟模型不合理	無	
10	可預見的隨機數	無	
11	投票權管理混亂	無	

2022年6月10日

編號	審計項目	風險級別	狀態
12	數據隱私洩露	無	
13	鏈上時間使用不當	無	
14	Fallback函數編碼不當	無	
15	鑒權不當	無	
16	Opcode使用不當	無	
17	內聯匯編使用不當	無	
18	構造函數不規範	無	
19	返回值不規範	無	
20	Event不規範	無	
21	關鍵字使用不規範	無	
22	未遵循ERC標準	中	已告知
23	條件判斷不規範	無	
24	流動性枯竭風險	無	
25	中心化風險	無	
26	邏輯變更風險	無	
27	整數溢出	無	
28	函數可見性不當	無	
29	變量初始化不當	無	
30	合約間調用不當	無	
31	變量不規範	無	
32	重放	無	
33	隨機存儲位置寫入	無	
34	蜜罐邏輯	無	
35	哈希碰撞	無	
36	外部函數調用不當	高	已告知

上述表格中，狀態欄內容若為「已告知」，則表示審計團隊已告知項目方項目存在的漏洞，但項目方未對漏洞進行修改，或未告知審計團隊漏洞的修改進度。若狀態欄中填寫「已修改」則表示項目方已進行對漏洞的修改，並通過審計團隊確認。

2022年6月10日

5. 風險項與修改方案

以下部分為審計後得知的風險項相關詳細信息，其中內容包括風險類型、風險級別、問題位置、問題描述、修改建議及項目方反饋。

1. 未遵循ERC標準

位置	文件	風險状态	風險級別
Line 86-95	MatrixPlusBox.sol	⚠ 已告知	中风险

① 风险描述

MatrixPlusBox 作为可锁定 NFT，在 NFT 锁定期间，代币的拥有者不能对代币进行转账操作。但是销毁代币时并没有对 NFT 是否处于锁定状态进行检查。

② 修改建议

在对 NFT 代币进行销毁之前添加检查环节，确保代币处于解锁状态下才可进行销毁

2022年6月10日

③ 关联代码

```JavaScript
function burn(uint256 tokenId) external {
    //@OKLink Audit Description: 缺失对 NFT 是否处于锁定状态进行检查
    //@OKLink Audit Solution: 添加检查环节，确保解锁状态下的代币才可进行销毁
    require(
        _isApprovedOrOwner(_msgSender(), tokenId) ||
            hasRole(BURNER_ROLE, _msgSender()) ||
            masterOf(tokenId) == _msgSender(),
        "ERC721: caller is not owner nor approved"
    );

    _burn(tokenId);
}
```

2022年6月10日

2. 未遵循ERC標準

位置	文件	風險状态	風險級別
Line 32-53	ERC5080Bound.sol	⚠️ 已告知	中风险

① 风险描述

#1：Bound 代币在调用 _lock 函数（进行 NFT 代币锁定操作）的时候铸造，在调用 unlockFrom 函数（进行 NFT 代币解锁操作）的时候销毁。而 Bound 代币的销毁操作只存在于 _afterTokenLock 函数，该函数在 ERC5080 中的 unlockFrom，_lock 以及 _burn 函数中被调用到。但是即使不调用 unlockFrom 函数，在锁定时间过后 NFT 代币也会恢复到解锁状态，而此时 NFT 代币对应的 Bound 代币依然未被销毁。

#2：代币锁定状态判断函数 isLocked：在锁定时间过后代币也会恢复到解锁状态。

② 修改建议

建议开发团队重新考虑 NFT 代币锁定状态的判断逻辑或者 Bound 代币的销毁逻辑，从而保证 Bound 代币只在 NFT 代币锁定的期间存在，在 NFT 代币解锁的时候销毁。

③ 关联代码#1

```JavaScript
function _afterTokenLock(
    address operator,
    address from,
    uint256 tokenId,
    uint256 expired
) internal virtual override {
    super._afterTokenLock(operator, from, tokenId, expired);

    if (bound != address(0)) {
        if (expired != 0) {
            // lock mint
            if (operator != address(0)) {
                IERC721Bound(bound).safeMint(msg.sender, tokenId, "");
            }
        } else {
            // unlock
            if (IERC721Bound(bound).exists(tokenId)) {
                IERC721Bound(bound).burn(tokenId);
            }
        }
    }
}
```

③ 关联代码#2

```C++
function isLocked(uint256 tokenId) public view virtual override returns (bool)
{
    return lockedTokens[tokenId] > block.timestamp;
}
```

2022年6月10日

3. 外部函数调用不当

位置	文件	風險状态	風險級别
Line 204、275、348	OpenPFPExchange.sol	⚠ 已告知	高风险

① 风险描述

matchAskWithTakerBidUsingETHAndWETH，matchAskWithTakerBid，matchBidWithTakerAsk三个函数都接受自定义的 MakerOrder 作为参数，并且没有对其内容进行安全性和可用性检测，用户可以随意构造结构体内容，在后续的执行中存在调用外部恶意合约的风险。

以上三个函数出现问题的地方类似，下面采用 matchAskWithTakerBid 函数进行举例说明。函数在 IExecutionStrategy(makerAsk.strategy).canExecuteTakerBid() 处调用 makerAsk.strategy 参数传入的地址，而该参数可以由调用者恶意构造，从而自定义调用的逻辑与返回值。

② 修改建议

对 makerAsk.strategy 添加白名单判断，确保目标地址是安全可信的地址。

2022年6月10日

③ 关联代码 1

JavaScript

```
function matchAskWithTakerBidUsingETHAndWETH(
        OrderTypes.TakerOrder calldata takerBid,
        OrderTypes.MakerOrder calldata makerAsk
    ) external payable override nonReentrant

function matchAskWithTakerBid(
        OrderTypes.TakerOrder calldata takerBid,
        OrderTypes.MakerOrder calldata makerAsk
    ) external override nonReentrant

function matchBidWithTakerAsk(
        OrderTypes.TakerOrder calldata takerAsk,
        OrderTypes.MakerOrder calldata makerBid
    ) external override nonReentrant
```

2022年6月10日

③ 关联代码 2

```JavaScript
function matchAskWithTakerBid(
    OrderTypes.TakerOrder calldata takerBid,
    OrderTypes.MakerOrder calldata makerAsk
) external override nonReentrant {
    require(
        (makerAsk.isOrderAsk) && (!takerBid.isOrderAsk),
        "Order: Wrong sides"
    );
    require(
        msg.sender == takerBid.taker,
        "Order: Taker must be the sender"
    );

    // Check the maker ask order
    bytes32 askHash = makerAsk.hash();
    //@OKLink Audit: 只检查签名
    _validateOrder(makerAsk, askHash);

    //@OKLink Audit Description: 未检查 makerAsk.strategy 地址的安全性即对其进行调
用。
    //@OKLink Audit Solution: 对 makerAsk.strategy 添加白名单判断
    (
        bool isExecutionValid,
        uint256 tokenId,
        uint256 amount
    ) = IExecutionStrategy(makerAsk.strategy).canExecuteTakerBid(
        takerBid,
        makerAsk
    );
```

2022年6月10日

```
require(isExecutionValid, "Strategy: Execution invalid");

// Update maker ask order status to true (prevents replay)
_isUserOrderNonceExecutedOrCancelled[makerAsk.signer][
    makerAsk.nonce
] = true;

// Execution part 1/2
_transferFeesAndFunds(
    makerAsk.strategy,
    makerAsk.collection,
    tokenId,
    makerAsk.currency,
    msg.sender,
    makerAsk.signer,
    takerBid.price,
    makerAsk.minPercentageToAsk
);

// Execution part 2/2
_transferNonFungibleToken(
    makerAsk.collection,
    makerAsk.signer,
    takerBid.taker,
    tokenId,
    amount
);

emit TakerBid(
    askHash,
    makerAsk.nonce,
    takerBid.taker,
    makerAsk.signer,
    makerAsk.strategy,
    makerAsk.currency,
    makerAsk.collection,
    tokenId,
    amount,
    takerBid.price
);
```

2022年6月10日

免責聲明

i. 本審計報告僅針對最終出具報告中載明的審計類型進行審計，其他未知安全漏洞不在本次審計責任範圍之內，我方無需為此承擔責任。

ii. 我方僅應根據審計報告發布之前存在或發生的攻擊或漏洞發布審計報告。對於將來存在或發生的新攻擊或漏洞，我方無法確定對其項目安全狀態的可能影響，對此概不負責。

iii. 我方發布的審計報告中的安全審計分析及其他內容應僅基於項目方在發布審計報告之前向我方提供的文件和材料（包括但不限於合約代碼），並且上述文件和資料不應該存在缺乏信息、被篡改、刪除或隱藏的情況，如果項目方提供的文件和資料存在不真實、不準確、缺乏信息、被篡改、刪除或隱藏的情況，或者對上述文件和資料的改動是在發布審計報告之後作出的，我方不承擔因反映情況與實際情況不一致引起的損失和不利影響 。

iv. 項目方知曉我方出具的審計報告系根據項目方提供的文件和資料、依靠我方現掌握的技術而作出的。但由於任何機構均存在技術的局限性，我方作出的審計報告仍存在無法完整檢測出全部風險的可能性。我方審計團隊鼓勵項目的開發團隊以及任何相關利益方對項目進行後續的測試及審計。

v. 項目方保證其委托我方提供審計或測試服務的項目合法、合規，且不違反適用法律。審計報告僅用於項目方參考，審計報告的內容、獲取方式、使用以及任何其所涉及的服務或資源都不能作為任何形式的投資、稅務、法律、監管及建議等的依據，我方不因此承擔相關責任。在未經我方書面同意之前，項目方不得將審計報告的全部或部分內容以任何形式提及、引用、展示或發送給任何第三方，否則由此產生的任何損失和責任由項目方自行承擔。我方對任何人依賴審計報告或將之用於任何目的概不承擔責任 。

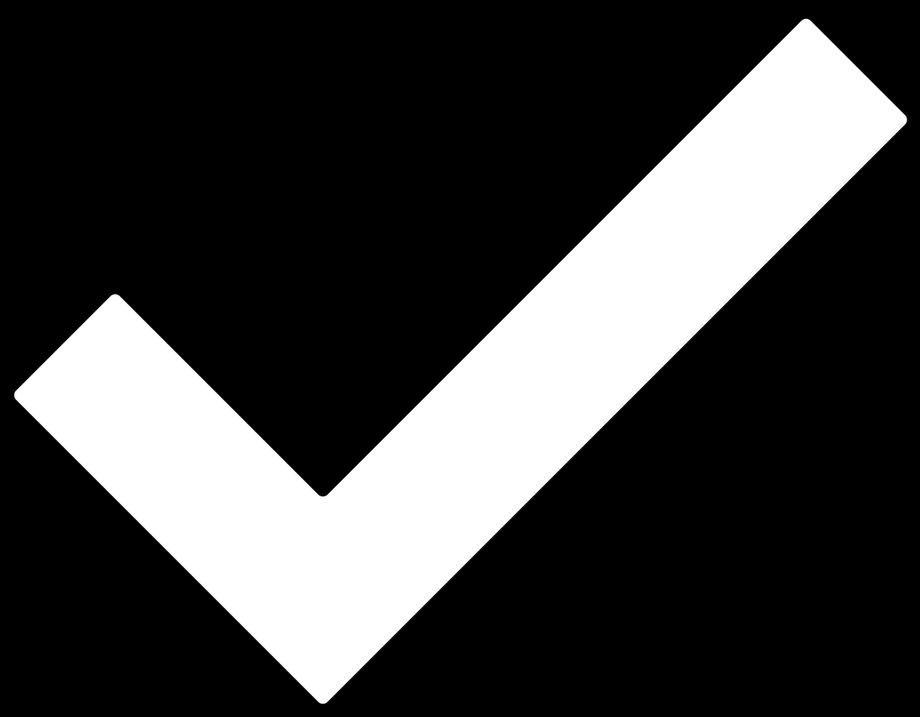
vi. 本審計報告不涉及合約的編譯器及任何超出智能合約編程語言的領域，所審計的智能合約因引用鏈下信息或資源所導致的風險及責任，由項目方自行承擔 。

vii. 不可抗力。不可抗力是指雙方在訂立合同時不能預見、對其發生和後果不能避免且不能克服的事件，包括但不限於戰爭、臺風、水災、火災、地震、潮汐、雷電、天災、罷工、核爆炸、流行病等自然災害和法律、法規和政策變更及政府行為等其它不可預見，對其發生和後果不能防止或避免的事件，且該事件妨礙、影響或延誤任何一方根據合同履行其全部或部分義務。

viii. 如果有一方認為不可抗力發生影響履行本協議義務，應迅速通知另一方，按事件對履約影響的程度，由雙方協商決定是否終止合同或部分免除履約的責任，或者延期履約。

ix. 當不可抗力發生時，任何一方都不能被視作違約或不履行本協議義務。在事件前存在的經濟上的責任，不應受到影響，項目方應對我方已完成工作做出支付。

2022年6月10日



審計通過.

日期 2022年6月10日

審計 歐科雲鏈

本次审计的目标是 Radiocaca 下基于Solidity语言编写的 ERC721L 和 openpfp-contracts 两个项目。关注 ERC721L 项目中代币协议的设计、代币锁定机制以及代币转移前检测机制，发现潜在的安全隐患。关注 openpfp-contracts 项目中订单簿形式下的订单配对、取消以及下单，代币转移策略以及费率设置等方面，发现潜在的安全隐患。