1. Genomics Data Management Smart Contract Vulnerabilities

```
Reentrancy in GenomicsDataManagement.listRawGenomicNFT(uint256) (GenomicsManagement.sol#2784-2796):
        External calls:
        - RawNFTSmartContract.transferFrom(msg.sender,address(this),_tokenId) (GenomicsManagement.sol#2788)
        State variables written after the call(s):
        - RawNFTs[RawNFTsCount] = RawGenomicNFT(RawNFTsCount,RawNFTSmartContract,_tokenId,address(msg.sender),SequencingMethodPrice[1],SequencingMethodPrice[2],SequencingMethodPrice[3]) (GenomicsMa
nagement.sol#2790)
        - RawNFTsCount ++ (GenomicsManagement.sol#2789)
        - SecurityDepositValue[RawNFTsCount] = securitydeposit (GenomicsManagement.sol#2792)
Reentrancy in GenomicsDataManagement.listSequencedGenomicNFT(uint256,uint256,uint256) (GenomicsManagement.sol#2955-2961):
        External calls:
        - SequencedNFTSmartContract.transferFrom(msg.sender,address(this),_tokenId) (GenomicsManagement.sol#2956)
        State variables written after the call(s):
        - ActiveChildrenListings[childToParentTokenId[address(SequencedNFTSmartContract)][_tokenId]] += 1 (GenomicsManagement.sol#2959)
        - SequencedNFTs[SequencedNFTsCount] = SequencedGenomicNFT(SequencedNFTsCount,SequencedNFTSmartContract,address(msg.sender),_tokenId,(_fullaccessprice * 10000000000000000),(_limitedaccessp
rice * 10000000000000000)) (GenomicsManagement.sol#2958)
        - SequencedNFTsCount ++ (GenomicsManagement.sol#2957)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-vulnerabilities-2

Reentrancy in GenomicsDataManagement.delistRawGenomicNFT(uint256) (GenomicsManagement.sol#2798-2809):
        External calls:
        - Raw.nft.transferFrom(address(this),msg.sender,Raw.tokenId) (GenomicsManagement.sol#2806)
        External calls sending eth:
        - address(msg.sender).transfer(SecurityDepositValue[Raw.RawNFTId]) (GenomicsManagement.sol#2807)
        Event emitted after the call(s):
        - DelistedRawGenomicNFT(Raw.RawNFTId,address(Raw.nft),Raw.tokenId,msg.sender) (GenomicsManagement.sol#2808)
Reentrancy in GenomicsDataManagement.delistSequencedGenomicNFT(uint256) (GenomicsManagement.sol#2963-2971):
        External calls:
        - Sequenced.nft2.transferFrom(address(this),msg.sender,Sequenced.tokenId) (GenomicsManagement.sol#2969)
        Event emitted after the call(s):
        - DelistedSequencedGenomicNFT(Sequenced.SeqNFTId,address(Sequenced.nft2),Sequenced.tokenId,msg.sender) (GenomicsManagement.sol#2970)
Reentrancy in GenomicsDataManagement.listRawGenomicNFT(uint256) (GenomicsManagement.sol#2784-2796):
        External calls:
        - RawNFTSmartContract.transferFrom(msg.sender,address(this),_tokenId) (GenomicsManagement.sol#2788)
        Event emitted after the call(s):
        - ListedRawGenomicNFT(RawNFTsCount,address(RawNFTSmartContract),_tokenId,msg.sender) (GenomicsManagement.sol#2795)
Reentrancy in GenomicsDataManagement.listSequencedGenomicNFT(uint256,uint256,uint256) (GenomicsManagement.sol#2955-2961):
        External calls:
        - SequencedNFTSmartContract.transferFrom(msg.sender,address(this),_tokenId) (GenomicsManagement.sol#2956)
        Event emitted after the call(s):
        - ListedSequencedGenomicNFT(SequencedNFTsCount,address(SequencedNFTSmartContract),_tokenId,msg.sender,_fullaccessprice,_limitedaccessprice) (GenomicsManagement.sol#2960)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-vulnerabilities-3
```

```
GenomicsDataManagement.refundRawNFTSequencingPayment(uint256,uint256) (GenomicsManagement.sol#2848-2861) uses timestamp for comparisons
        Dangerous comparisons:
        - require(bool,string)(block.timestamp > SequencingRequestStartTime[Raw.RawNFTId][_requestnumber] + SequencingPeriod,The time window for sequencing is still openned) (GenomicsManagement.sol
#2853)
GenomicsDataManagement.refundInitialFullAccessPayment(uint256,uint256) (GenomicsManagement.sol#2864-2874) uses timestamp for comparisons
        Dangerous comparisons:
        - require(bool,string)(block.timestamp > SequencingRequestStartTime[Raw.RawNFTId][_requestnumber] + SequencingPeriod + InitialFullAccessGrantingPeriod,The time window for initial full acces
s is still openned) (GenomicsManagement.sol#2868)
GenomicsDataManagement.ConfirmRawNFTSequencing(uint256,uint256) (GenomicsManagement.sol#2894-2909) uses timestamp for comparisons
        Dangerous comparisons:
        - require(bool,string)(block.timestamp <= SequencingRequestStartTime[Raw.RawNFTId][_requestnumber] + SequencingPeriod,The sequencing time window is already closed) (GenomicsManagement.sol#2
898)
GenomicsDataManagement.InitialFullAccessProof(uint256,uint256) (GenomicsManagement.sol#2913-2929) uses timestamp for comparisons
        Dangerous comparisons:
        - require(bool,string)(block.timestamp <= SequencingRequestStartTime[Raw.RawNFTId][_requestnumber] + SequencingPeriod + InitialFullAccessGrantingPeriod,The Initial Full Access Granting Peri
od time window is already closed) (GenomicsManagement.sol#2919)
GenomicsDataManagement.withdrawFullAccessPayment(uint256,uint256) (GenomicsManagement.sol#2993-3003) uses timestamp for comparisons
        Dangerous comparisons:
        - require(bool,string)(FullAccessRequestStartingTime[Sequenced.SeqNFTId][msg.sender] > block.timestamp,The time window for full access request is still openned) (GenomicsManagement.sol#2997
)
GenomicsDataManagement.ProoofofFullAccess(uint256,uint256) (GenomicsManagement.sol#3006-3021) uses timestamp for comparisons
        Dangerous comparisons:
        - require(bool,string)(block.timestamp <= FullAccessRequestStartingTime[Sequenced.SeqNFTId][signature.dataBuyer] + FullAccessGrantingPeriod,The full time access window for this buyer has al
ready closed) (GenomicsManagement.sol#3011)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#block-timestamp
```

```
Reentrancy in GenomicsDataManagement.ConfirmRawNFTSequencing(uint256,uint256) (GenomicsManagement.sol#2894-2909):
        External calls:
        - address(msg.sender).transfer(SequencingMethodPrice[RequestedSequencingMethod[Raw.RawNFTId][_requestnumber]]) (GenomicsManagement.sol#2903)
        State variables written after the call(s):
        - ActiveSequencingRequests[Raw.RawNFTId] -= 1 (GenomicsManagement.sol#2904)
        - SequencingRequestCompleted[Raw.RawNFTId][_requestnumber] = true (GenomicsManagement.sol#2905)
        Event emitted after the call(s):
        - RawNFTisSequenced(Raw.RawNFTId,address(Raw.nft),Raw.tokenId,Raw.dataOwner,SequenceRequesterAddress[Raw.RawNFTId][_requestnumber],msg.sender,signature.sig) (GenomicsManagement.sol#2906)
Reentrancy in GenomicsDataManagement.InitialFullAccessProof(uint256,uint256) (GenomicsManagement.sol#2913-2929):
        External calls:
        - address(msg.sender).transfer(SecurityDepositValue[Raw.RawNFTId]) (GenomicsManagement.sol#2924)
        State variables written after the call(s):
        - SecurityDepositValue[Raw.RawNFTId] = 0 (GenomicsManagement.sol#2925)
        Event emitted after the call(s):
        - InitialFullAccessGranted(Raw.RawNFTId,address(Raw.nft),Raw.tokenId,msg.sender,signature.dataBuyer) (GenomicsManagement.sol#2927)
Reentrancy in GenomicsDataManagement.ProofofFullAccess(uint256,uint256) (GenomicsManagement.sol#3006-3021):
        External calls:
        - address(msg.sender).transfer(Sequenced.fullaccessprice) (GenomicsManagement.sol#3014)
        State variables written after the call(s):
        - ActiveSequencedNFTAccessRequests[Sequenced.SeqNFTId] -= 1 (GenomicsManagement.sol#3015)
        - FullAccessRequestCompleted[Sequenced.SeqNFTId][_requestnumber] = true (GenomicsManagement.sol#3016)
        Event emitted after the call(s):
        - FullAccessGranted(Sequenced.SeqNFTId,address(Sequenced.nft2),Sequenced.tokenId,msg.sender,signature.dataBuyer) (GenomicsManagement.sol#3019)
Reentrancy in GenomicsDataManagement.delistRawGenomicNFT(uint256) (GenomicsManagement.sol#2798-2809):
        External calls:
        - address(msg.sender).transfer(SecurityDepositValue[Raw.RawNFTId]) (GenomicsManagement.sol#2807)
        Event emitted after the call(s):
        - DelistedRawGenomicNFT(Raw.RawNFTId,address(Raw.nft),Raw.tokenId,msg.sender) (GenomicsManagement.sol#2808)
Reentrancy in GenomicsDataManagement.refundInitialFullAccessPayment(uint256,uint256) (GenomicsManagement.sol#2864-2874):
        External calls:
        - address(msg.sender).transfer(sequencingRequesterPaidAmount[msg.sender][Raw.RawNFTId]) (GenomicsManagement.sol#2871)
        State variables written after the call(s):
        - ActiveInitialFullAccessRequests[Raw.RawNFTId] -= 0 (GenomicsManagement.sol#2873)
Reentrancy in GenomicsDataManagement.refundRawNFTSequencingPayment(uint256,uint256) (GenomicsManagement.sol#2848-2861):
        External calls:
        - address(msg.sender).transfer(sequencingRequesterPaidAmount[msg.sender][Raw.RawNFTId]) (GenomicsManagement.sol#2857)
        State variables written after the call(s):
        - ActiveSequencingRequests[Raw.RawNFTId] -= 1 (GenomicsManagement.sol#2860)
        - sequencingPayerTracker[msg.sender][Raw.RawNFTId] = false (GenomicsManagement.sol#2859)
Reentrancy in GenomicsDataManagement.withdrawFullAccessPayment(uint256,uint256) (GenomicsManagement.sol#2993-3003):
        External calls:
        - address(msg.sender).transfer(Sequenced.fullaccessprice) (GenomicsManagement.sol#3000)
        State variables written after the call(s):
        - ActiveSequencedNFTAccessRequests[Sequenced.SeqNFTId] -= 1 (GenomicsManagement.sol#3002)
        - FullAccessBuyersTracker[_SeqNFTId][msg.sender] = false (GenomicsManagement.sol#3001)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-vulnerabilities-4
GenomicsManagement.sol analyzed (8 contracts with 81 detectors), 19 result(s) found
```

2. Raw Genomic Data NFT Smart Contract

```
0: Reentrancy in RawGenomicDataNFTs.mint(string) (RawGenomicNFTs.sol#1660-1665):
        External calls:
        - _safeMint(msg.sender,tokenCount) (RawGenomicNFTs.sol#1662)
                - IERC721Receiver(to).onERC721Received(_msgSender(),from,tokenId,data) (RawGenomicNFTs.sol#1483-1494)
        State variables written after the call(s):
        - _setTokenURI(tokenCount,_tokenURI) (RawGenomicNFTs.sol#1663)
                - _tokenURIs[tokenId] = _tokenURI (RawGenomicNFTs.sol#1630)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-vulnerabilities-2
```

3. Sequenced Genomic Data NFT Smart Contract

```
0: Reentrancy in SequencedGenomicDataNFTs.mint(string,uint256) (SequencedGenomicNFTs.sol#1684-1694):
        External calls:
        - GenomicsDataManagement.UpdateSequencedNFTMintingBalance(msg.sender,_parentID) (SequencedGenomicNFTs.sol#1686)
        State variables written after the call(s):
        - tokenCount ++ (SequencedGenomicNFTs.sol#1687)
1: Reentrancy in SequencedGenomicDataNFTs.mint(string,uint256) (SequencedGenomicNFTs.sol#1684-1694):
        External calls:
        - GenomicsDataManagement.UpdateSequencedNFTMintingBalance(msg.sender,_parentID) (SequencedGenomicNFTs.sol#1686)
        - _safeMint(msg.sender,tokenCount) (SequencedGenomicNFTs.sol#1688)
                - IERC721Receiver(to).onERC721Received(_msgSender(),from,tokenId,data) (SequencedGenomicNFTs.sol#1483-1494)
        State variables written after the call(s):
        - _setTokenURI(tokenCount,_tokenURI) (SequencedGenomicNFTs.sol#1689)
                - _tokenURIs[tokenId] = _tokenURI (SequencedGenomicNFTs.sol#1630)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-vulnerabilities-2
```

```
0: Reentrancy in SequencedGenomicDataNFTs.mint(string,uint256) (SequencedGenomicNFTs.sol#1684-1694):
        External calls:
        - GenomicsDataManagement.UpdateSequencedNFTMintingBalance(msg.sender,_parentID) (SequencedGenomicNFTs.sol#1686)
        - _safeMint(msg.sender,tokenCount) (SequencedGenomicNFTs.sol#1688)
                - IERC721Receiver(to).onERC721Received(_msgSender(),from,tokenId,data) (SequencedGenomicNFTs.sol#1483-1494)
        Event emitted after the call(s):
        - Transfer(address(0),to,tokenId) (SequencedGenomicNFTs.sol#1352)
                - _safeMint(msg.sender,tokenCount) (SequencedGenomicNFTs.sol#1688)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-vulnerabilities-3
```

4. Raw G