Vender

AMFEIX

Deployment Address

0xe2c43d2c6d6875c8f24855054d77b5664c7e810f

Code Details

https://etherscan.io/address/0xe2c43d2c6d6875c8f24855054d77b5664c7e810f

Vulnerable Code

```
1 function startAuction(uint256 _pepeId, uint256 _beginPrice, uint256 _endPrice, uint64 _duration) public {
2         require(pepeContract.transferFrom(msg.sender, address(this), _pepeId));
3
4         require(now > auctions[_pepeId].auctionEnd);//can only start new auction if no other is active
5
6         PepeAuction memory auction;
7         auction.seller = msg.sender;
8         auction.pepeId = _pepeId;
9
10        auction.auctionBegin = uint64(now);
11
12        auction.auctionEnd = uint64(now) + _duration; // vulnerable code
13        require(auction.auctionEnd > auction.auctionBegin);
14        auction.beginPrice = _beginPrice;
15        auction.endPrice = _endPrice;
16
17        auctions[_pepeId] = auction;
18
19        emit AuctionStarted(_pepeId, msg.sender);
20    }
21 ...
22
23 require(now < auction.auctionEnd); //time check
```

Description

We found this contract has a possible ineffectiveness in time check. In line 23, this contract has a time check (in line 335 for real code). This check relies on the value of the member auctionEnd. However, the auctionEnd is assigned in line 12 (in line 228 for real code) by an argument _duration. This function is feasible for public access and there's no argument check on this _duration variable. If a very large number is assigned to _duration, the time check in line 23 will be easily passed.

Suggestions

Check the _duration value before assign it to auctionEnd.