Security Audit
Decentraland
ENS Avatars

Monday, 20-Jan-2020

Agustín Aguilar

Introduction

The Decentraland team requested an audit of their ENS Avatars project, the audited repositories follow.

ENS Avatars

Manages the registration and public offer of usernames (avatars) which are represented as ERC721 tokens, linked to an ENS subdomain.

The contracts that constitute the project are:

- DCLRegistrar - Manages the ownership of the usernames through an ERC721 implementation, manages calls to the ENS contracts and allows whitelisted addresses to assign unowned usernames to any address.
- DCLController - Allows any address to register a new username on DCLRegistrar in exchange for 100 MANA, the received funds are burned.

https://github.com/decentraland/avatars-contract (0x744dcf0144987dac3f411e06a3b44f05e02d0fb8)

Issues

Medium severity

M1 - Possible username front-running

The `DCLController` contract fixes a maximum value for `gasPrice` in order to avoid front-running of username registrations, meaning that any address trying to register a username can't register it using a transaction above such limit.

This mechanism's by itself not enough deterrent of such attacks, giving that when two transactions with the same `gasPrice` are included on the same block the order of the two is controlled by the miner, and most of the time is defined randomly.

An attacker could perform a front-running attack to register almost all the usernames that genuine users try to register, such an attack is possible by using multiples accounts to send multiples transactions.

Proposed solutions:

a) Implement a commit and reveal scheme.
b) Implement a prioritized transaction scheme, with a centralized operator that limits front-running.
c) Implement both solutions `a` and `b`.

Low severity

L1 - Inconsistent username ID specification

The `DCLRegistrar` registers the avatar's subdomain ERC721 as the lowercase `subdomainLabelHash` representation. But the "`available`" method of the contract is being called with `subdomainNameHash.`

However, the method available would still work as expected, given the fact that "`registry.owner`" checks for a `subdomainNameHash`, this also leads to false positives.

This issue doesn't pose a direct threat to the project, but it could induce an issue with third party contracts that interact with the avatar's tokens.

Proposed solutions:

a) Define the token ID as `subdomainNameHash` or `subdomainLabelHash.`

Update ad715a7:

The Decentraland team fixed this issue by using `subdomainLabelHash` on the method `available`.

Notes

N1 - Fixed mixed-case representation

The `DCLController` contract defines a fixed mixed-case representation for each avatar, intended for display purposes. This representation is defined when the domain is registered, and there is no mechanism in place to update it after it’s defined.

This issue implies that if a user registers a username with an unintended mixed-case representation, such representation can’t be updated (e.g: nAcho instead of Nacho).

Proposed solution:

a) Add a method to update the mixed-case representation of a registered domain.

N2 - String char count can be simplified

The `DCLController` contract uses the method `_bytes32ToString` during the pre-loading of domains on the contract, each bytes32 representation of the string has no specific length, so the length is inferred searching for the first empty byte.

The code that searches for the length of the string can be simplified, the `j` and `charCount` variables serve the same purpose and one can be removed.

Additionally, the first empty byte is detected by shifting and masking the bytes32 word, it can be safely assumed that if a byte is empty, the following bytes must also be empty, and thus the masking can also be avoided.

N3 - The update of subnode ownership can be automated

The ownership of each avatar is represented using an ERC721 token, this token is transferrable and it’s also linked to a homonym ENS subdomain.

If the owner of the avatar transfers the token, the owner on the ENS Registry is not updated, and the new owner is expected to call the method “`reclaim`” to update such registry.

Consider extending the ERC721 parent contract in order to call `setSubnodeOwner` when a token is transferred, to avoid the UX overhead.

N4 - Constant variables are emitted on events

The `DCLController` contract emits the event `NameBought` after an avatar is sold, this event includes the parameter “`_price`” of the sell, which is a constant.

Consider removing “`_price`” of the emitted event.

Final thoughts

The contracts composing the audited projects are well written. The project makes extensive use of double-checks and redundant mechanics, this is considered good practice with the only downside of the additional gas costs.

The M1 issue can be exploited in the wild, and it’s possible to such an attack to be left unnoticed, for so it’s recommended to set up permanent monitoring of this set of contracts, and a kill switch.

- January 2020 - Agustín Aguilar