Kathmandu University

Department of Computer Science and Engineering

Dhulikhel, Kavre

A Report

On

"Assignment 4: ERC Standards"

[Course Code: COMP 492]

(For the partial fulfillment of IV year/II Semester in Computer Engineering)

Submitted by:

Sabin Thapa

Roll no.: 54

CE 4th Year

sabint017@gmail.com

Submitted to:

Mr. Suresh Gautam

Department of Computer Science and Engineering

Submission date:

March 1, 2023

1. **Choose any one real-world use case for NFTs and describe how you would implement it using ERC-721 or ERC-1155. What are some considerations you would need to keep in mind when designing the smart contract?**

Ans:

The non-fungible token (NFT) is a cryptographic asset that's created using blockchain technology. NFTs are unique cryptographic tokens that exist on a blockchain and cannot be replicated. They can be traded for cryptocurrencies, money, or other NFTs as well. It all depends on the value the owner and the market have placed on them. Non-fungible tokens contain a digital signature which makes each one unique. They are digital assets and can be photos, videos, audio, etc. Some examples of NFT are artwork, games, sports collectibles, comic books, etc. In the following section, we discuss in detail the real-world use case for NFTs and describe it.

The real-world use cases for NFTs

NFTs can be used to improve transparency and trust within various industries and can be utilized to create unique digital collectibles. They have several real-world use cases. Some of the examples are:

- Voting
- Gaming
- Music
- Sports
- Real Estate
- Digital Art and Collectibles

Let's choose **Voting** as the real-world use case of NFT and discuss further on it.

NFTs in Voting

NFTs have the potential to create a more secure and transparent voting system compared to the traditional voting system by authenticating voters, tracking votes, and creating a tamper-proof decentralized system. It would ensure fair counting of votes as only authenticated users would be eligible to cast their votes. However, there are also risks and challenges associated with it, such as ensuring privacy and preventing hacking, which would need to be addressed before implementing such a system. Therefore, this topic of using NFTs in voting is still a debate and ongoing research. Now, let's discuss how voting can be implemented as NFT using the ERC-721 standards.

Implementing ERC-721 standards for NFTs in Voting

ERC-721 is a standard for creating NFTs on the Ethereum blockchain. The process for implementing voting as NFT using the ERC-721 standards is discussed in the following section:

1. Define the voting process
2. Create the contract (NFT)
3. Issue NFTs to voters
4. Case votes (as NFT transfers)
5. Count the votes
6. Destroy/Burn the NFT

These steps are discussed in detail below:

1. **Define the voting process**
   The voting process including the eligibility criteria and the rules for voters needs to be defined first. We should also define the candidates or options that voters can choose from and cast their votes.

2. **Create the contract (NFT)**
   A smart contract can be created using the ERC-721 standard that will represent the NFTs for each vote. The metadata for the NFTs, including the name, description, and any other relevant information should be included in the contract.

3. **Issue NFTs to voters**
   The next step is to issue the NFT to voters. Each eligible voter should be issued an NFT that represents their right to vote. This NFT should be created and distributed to each voter's wallet address.

4. **Case votes (as NFT transfers)**
   After the eligible voters have an NFT each, they cast their votes. When a voter casts their vote, they basically transfer their NFT to the candidate of their choice. This transfer would then be recorded on the blockchain and would represent the voter's vote.
5. **Count the votes**
   After the voting period ends, the votes can be counted by examining the NFT transfers on the blockchain. The candidate with the most NFT transfer would be announced as the winner.
6. **Destroy/Burn the NFT**
   The NFTs can be destroyed or burned after the voting process is complete to prevent them from being used again in future elections.

In this way, voting can be implemented as NFT using the ERC-721 standards. Implementing voting as NFT using this standard can create a secure and transparent voting system on the blockchain.

To ensure the system is secure, reliable, and transparent, careful **consideration** of several factors is required. Some of the considerations are discussed below:

1. **Transparency**
   To ensure that the voting process is fair and trustworthy, the smart contract should be designed to be transparent which includes making the code of the smart contract open-source, using public blockchains and providing access to the voting results.

2. **Privacy**
   Privacy is one of the key factors that should be considered. It's important to ensure that the voting process is private and confidential i.e. the voters should not be able to see the votes of other voters and the results should be anonymous. Cryptographic techniques such as *zero-knowledge proofs* can be used to ensure that the voting process is private.

3. **Security**
   To prevent hacking and to ensure that votes are not tampered with, the smart contract should be designed with highest level of security. This includes implementing secure authentication mechanisms, using encryption/decryption to protect sensitive data, and following best practices for smart contract development.

4. **Eligibility criteria**
   The smart contract should be designed to ensure that only eligible voters can participate in the voting process.

5. **Double voting prevention**
   The contract should be designed to prevent users from casting more than one vote. This can be achieved by implementing a system that checks for duplicate NFT transfers or by limiting each voter to a single NFT.

6. **Contract upgrades**
   There should be mechanisms for upgrades or improvements to the system, as well as a plan in case of a failure.

7. **Testing and auditing**
   Continuous rigorous testing and auditing should be done to identify and fix vulnerabilities issues before deploying the smart contract.

Overall, designing a smart contract as such requires careful consideration of several factors to ensure that the system is secure, reliable and transparent.

2. **ERC-721 allows for unique, non-interchangeable tokens, while ERC-1155 allows for both fungible and non-fungible tokens. Describe a scenario where you would use ERC-721 over ERC-1155 and vice versa.**

Ans:

ERC-721 and ERC-1155 are both Ethereum-based token standards. They have different properties and use cases. ERC-721 can be used torepresent unique assets, such as collectibles, digital art, or real state and can be used to prove ownership of assets, such as tickets or access rights. On the other hand, ERC-1155 can be used to represne collections of assets, such as game items or digital trading cards and can be used for fractional ownershipo of assets, where multiple users own a share of a single asset. The scenario where we would use one over the other is discussed below:

Using ERC-721 over ERC-1155:

Suppose, we want to create a unique, non-fungible asset, such as a collectible or a game item, here, ERC-721 would be the better choice because it allows us to create a single, unique token with a unique ID that cannot be duplicated or divided. Each token can have its own distinct properties and metadata which makes it suitable for one-of-a-kind assets. For example, if we're creating a game where players collect unique jewels, we would use ERC-721 to create a token for each jewel. Each token would have its own name and its image, attributes, and players could collect and trade them like physical trading cards.

Using ERC-1155 ERC-721 :

Consider a scenario, where we want to create a fungible asset that can be used in different contexts - such as current or a reward point. In this case ERC-1155 would be the better choice because it allows us to create a single contract that can represent multiple tokens with the same properties. For example, if we're creating a loyalty program for our business, we could use ERC-721 over ERC-721 to create a token contract that represents reward point where wach token would have the same value and properties, but could be used in different ways by different users, such as for discounts, or rewards. The same contract coule be used by multiple businesses or applications making it more efficient and cost-effective than creating a separate contract for each use case.

In this case, both the ERC-721 and ERC-1155 contracts are equally important in their own contexts. The selection of one over the other solely depends on our use case.

3. **Describe some potential security risks associated with ERC-721 and ERC-1155 tokens. How can these risks be mitigated in the design and implementation of smart contracts?**

Ans:

ERC-721 and ERC-1155 are NFT standards on the Ethereum standard. They have many use cases and along with that there are some potential risks associated with their use. In the following section, we discuss the potential risks and the ways to mitigate them.

ERC-721 token standard:

Risk

1. **Lack of interoperability**

   ERC-721 tokens are not interoperable with other blockchains or token standards, which can pose limitations in terms of usage and liquidity.

2. **Smart contract vulnerabilities**

   Since ERC-721 tokens are implemented through smart contracts, they can be vulnerable to carious security issues such as malicious code injection, reentrancy attacks and overflow/underflow attacks.

3. **Phishing Attacks**

   Phishing attack is an attack where a user is tricked into giving away their private keys or other sensitive information. With ERC-721 tokens, users need to interact with smart contracts to transfer ownership or sell their tokens, which makes them vulnerable to phishing attacks.

4. **Centralized Storage**

   Some ERC-721 tokens require centralized storage of their digital assets, which can pose security risks as the centralized storage could be vulnerable to hacking or data breaches.

Solution

1. The solution to lack of interoperability is to use token bridgingsolutions such as the Polygon Network or cross-chain bridges like Connext, which allow ERC-721 tokens to be transferred across different blockchains.
2. The solution to smart contract vulnerabilities is to follow best oractices for smart contract development and security, such as implementing security checks, regularly updating the contract and using well-audited and tested code.
3. The solution to phishing attacks is to educate users on how to identify and avoid phishing scams, use secure wallets that support ERC-721 tokens, and use two factor authentication (2FA) wherever possible.

ERC-1155 token standard:

Risk

1. **Front-running Attacks:**
   These attacks occur when an attacker can view a transaction before it is executed and manipulate the transaction to their advantage. Since ERC-1155 tokens allow batch transfers, attackers can potentially front-run transactions and manipulate them.
2. **Reentrancy Attacks:**
   These attacks occur when a contract can be called repeatedly bedore its initial invocation has completed which can lead to unexpected behavious and vulnerabilities, such as attacker being able to withdraw more funds than they are entitled to.
3. **Token Swap Attacks:**
   These attacks occur when an attacker cretes a fake token that resembles an existing ERC-1155 token and convinces users to trade their legitimate tokens for the fake ones.
4. **Smart Contract Vulnerabilities:**
   Like with the ERC-791 token, ERC-1155 tokens also rely on smart contracts so they possess the similar vulnerabilities.

Solution

1. Limiting the maximum gas price for transactions can help reduce the impact of front-running attacks.
2. Implementing reentrancy guards can prevent reentrancy attacks by limiting the number of times a contract can be called before it completes its initial invocation.
3. To prevent token swap attacks, users should verify the authenticity of ERC-1155 tokens before trading or transacting with them.
4. Conducting regular code audits can help identify and fix vulnerabilities in the smart contracts that power ERC-1155 tokens.

Overall, it is important for developers and users to be aware of the potential security risks associated with ERC-791 and ERC-1155 tokens and take appropriate measures to mitigate them.

4. **Compare and contrast ERC-721 and ERC-1155 in terms of gas costs, scalability, and ease of use. What are some factors that developers should consider when deciding which standard to use for their project?**

Ans:

The comparison of ERC-721 and ERC-1155 tokens in terms of gas costs, scalability, and ease of use is as follows:

1. **Gas costs**

   ERC-721 tokens generally have higher gas costs than ERC-1155 tokens. This is because each ERC-721 token is a unique asset with its own smart contract whereas ERC-1155 tokens can represent multiple assets within a single smart contract, which can reduce gas costs for transactions.

2. **Scalability**

   ERC-1155 tokens can represent multiple assets within a single smart contract, which can improve scalability and reduce network congestion. On the other hand, ERC-721 tokens can be less scalable than ERC-1155 tokens because each ERC-721 token requires its own smart contract, which can lead to network congestion and slower transaction times.

3. **Ease of use**

   For non-technical users, ERC-721 tokens are typically easier to use because they represent unique assets, such as collectibles or game items, that are easy to understand. In contrast, ERC-1155 tokens can represent multiple assets within a single smart contract which can be more complex to understand.

Some factors that developers should consider when deciding which standard to use for their project are as follows:

1. **Token type**

   ERC-721 is suitable for non-fungible tokens, while ERC-1155 can handle both fungible and non-fungible tokens.

2. **Ease of use**

   Developers should determine which standard will be more user-friendly to their target audience.

3. **Scalability**

   Developers should consider the scalability of each standard and determine which standard will be more scalable for their project.

4. **Gas costs**

   ERC-721 is generally more expensive than ERC-1155 due to the uniqueness of each token. So, developers should consider the gas costs of each standard and determine which standard will be more effective for their project.

5. **Complexity and development resources**

   ERC-721 is simpler to implement since it only handles non-fungible tokens, while ERC-1155 is more complex due to its ability to handle both fungible and non-fungible tokens. So, developers should consider the resources they have available for development and determine which standard will be easier and more efficient to implement for their project.

In conclusion, developers should carefully consider the specific requirements of their project and weigh the trade-odds of each standard in terms of gas costs, scalability, and ease of use before deciding which standard to use.

5. **What are some potential applications of ERC-721 and ERC-1155 tokens? How could these standards be used to create new types of digital assets and marketplaces?**

Ans:

ERC-721 and ERC-1155 tokens have wide ranges of applications. Some of them are discussed below:

1. **Gaming**

   ERC-721 tokens can be used to represent unique game items, such as collectibles or rare weapons, weapon skins while ERC-1155 tokens can be used to represent fungible assets, such as game currency or resources.

2. **Real Estate**

   ERC-721 tokens can be used to represent ownership of a rental property or a piece of real estate while ERC-1155 tokens can be used to represent multiple shares or fractional ownership of a property.

3. **Digital Art**

   ERC-721 tokens can be used to represent unique pieces of digital art, providing proof of ownership, while ERC-1155 tokens can be used to represent editions or versions of a piece of art.

4. **Supply Chain**

   ERC-721 tokens can be used to track the ownership and movement of physical goods throughout the supply chain, while ERC-1155 tokens can be used to represent multiple units or batches of goods.

5. **Identity**

   ERC-721 tokens can be used to represent a person's identity, such as a digital passport or driver's license, while ERC-1155 tokens can be used to represent multiple pieces of identity information, such as proof of address or age.

ERC-721 and ERC-1155 tokens provide a lot of opportunities for creating new types of digital assets and marketplaces. By using these standards, developers can create unique and valuable digital assets that can be traded in a decentralized and transparent way, without the need for centralized platforms. Some of the examples are discussed below:

1. **Unique digital asset**
   ERC-721 tokens are NFT that represent unique and indivisible digital assets. These tokens can be used to represent any unique digital asset, such as digital art, collectibles, or virtual real estate.

2. **Decentralized marketplaces**
   Using these tokens, developers can create decentralized marketplaces where users can buy, sell, and trade digital assets. These marketplaces are transparent and decentralized, which means that users don't have to rely on centralized platforms to buy and sell digital assets.

3. **In-game assets**
   ERC-1155 tokens can be used to represent in-game assets such as weapons, armor, weapon skins and other collectibles. This creates a new way for players to buy, sell, and trade items in the game's ecosystem.

4. **Royalties and licensing**
   These tokens can be programmed to include royalty and licensing agreements. This means that creators of digital assets can earn royalties every time their asset is bought or sold on the marketplace.

Overall, ERC-721 and ERC-1155 standards provide developers with a lot of flexibility to create new types of digital assets and marketplaces. By using these standards, developers can create unique and valuable digital assets that can be traded in a decentralized and transparent way, without the need for intermediaries or centralized platforms.