NIST Internal Report
NIST IR 8472

Non-Fungible Token Security

Peter Mell
Dylan Yaga

This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8472

NATIONAL INSTITUTE OF
STANDARDS AND TECHNOLOGY
U.S. DEPARTMENT OF COMMERCE

NIST Internal Report
NIST IR 8472

Non-Fungible Token Security

Peter Mell
Dylan Yaga
Computer Security Division
Information Technology Laboratory

This publication is available free of charge from:
https://doi.org/10.6028/NIST.IR.8472

March 2024

U.S. Department of Commerce
Gina M. Raimondo, Secretary

National Institute of Standards and Technology
Laurie E. Locascio, NIST Director and Under Secretary of Commerce for Standards and Technology

Certain commercial equipment, instruments, software, or materials, commercial or non-commercial, are identified in this paper in order to specify the experimental procedure adequately. Such identification does not imply recommendation or endorsement of any product or service by NIST, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

There may be references in this publication to other publications currently under development by NIST in accordance with its assigned statutory responsibilities. The information in this publication, including concepts and methodologies, may be used by federal agencies even before the completion of such companion publications. Thus, until each publication is completed, current requirements, guidelines, and procedures, where they exist, remain operative. For planning and transition purposes, federal agencies may wish to closely follow the development of these new publications by NIST.

Organizations are encouraged to review all draft publications during public comment periods and provide feedback to NIST. Many NIST cybersecurity publications, other than the ones noted above, are available at https://csrc.nist.gov/publications.

NIST Technical Series Policies

Publication History
Approved by the NIST Editorial Review Board on 2024-02-02

How to Cite this NIST Technical Series Publication:
Mell P, Yaga D (2024) Non-Fungible Token Security. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Interagency or Internal Report (IR) NIST IR 8472. https://doi.org/10.6028/NIST.IR.8472

Author ORCID iDs
Peter Mell: 0000-0003-2938-897X
Dylan Yaga: 0000-0003-4058-3645

Contact Information
NISTIR8472@nist.gov

National Institute of Standards and Technology
Attn: Computer Security Division, Information Technology Laboratory
100 Bureau Drive (Mail Stop 8930) Gaithersburg, MD 20899-8930

Additional Information
Additional information about this publication is available at https://csrc.nist.gov/pubs/ir/8472/final, including related content, potential updates, and document history.

All comments are subject to release under the Freedom of Information Act (FOIA).

Abstract

Non-fungible token (NFT) technology provides a mechanism to enable real assets (both virtual and physical) to be sold and exchanged on a blockchain. While NFTs are most often used for autographing digital assets (associating one's name with a digital object), they utilize a strong cryptographic foundation that may enable them to regularly support ownership-transferring sales of digital and physical objects. For this, NFT implementations need to address potential security concerns to reduce the risk to purchasers. This publication explains NFT technology and then identifies and discusses a list of 27 potential security issues. All of the identified issues can be addressed through use of a systematic security approach that promotes a secure design and implementation.

Keywords

blockchain; definition; ERC-721; non-fungible token; properties; security; smart contract.

Reports on Computer Systems Technology

The Information Technology Laboratory (ITL) at the National Institute of Standards and Technology (NIST) promotes the U.S. economy and public welfare by providing technical leadership for the Nation's measurement and standards infrastructure. ITL develops tests, test methods, reference data, proof of concept implementations, and technical analyses to advance the development and productive use of information technology. ITL's responsibilities include the development of management, administrative, technical, and physical standards and guidelines for the cost-effective security and privacy of other than national security-related information in federal information systems.

Audience

This publication is intended for readers who want to better understand how NFTs function at a technical level and the associated potential security risks. This includes both purchasers of NFTs and developers of NFT implementations.

Patent Disclosure Notice

NOTICE: ITL has requested that holders of patent claims whose use may be required for compliance with the guidance or requirements of this publication disclose such patent claims to ITL. However, holders of patents are not obligated to respond to ITL calls for patents and ITL has not undertaken a patent search in order to identify which, if any, patents may apply to this publication.

As of the date of publication and following call(s) for the identification of patent claims whose use may be required for compliance with the guidance or requirements of this publication, no such patent claims have been identified to ITL.

No representation is made or implied by ITL that licenses are not required to avoid patent infringement in the use of this publication.

Table of Contents

1. Introduction

Non-fungible token (NFT) technology provides a mechanism to enable real assets (both virtual and physical) to be sold and exchanged on a blockchain. It does this by creating a unique blockchain token to represent each *asset*. A blockchain smart contract manages a group of tokens and enables them to be securely transferred between blockchain accounts. The verification of NFT ownership by an account is straightforward. This architecture provides a strong cryptographic foundation for NFT sales.

NFTs are commonly used for photography, digital art, trading cards, and music [1]. Usually, what is purchased is the right to "autograph" a digital asset with a blockchain ledger entry. In this case, ownership rights are not usually conveyed to the purchaser [6], and the autographing right is not necessarily exclusive. In other cases, sales of the digital tokens are intended by the seller to convey a sale of ownership rights over the linked digital assets. Someday, the technology may broadly support the secure record of physical asset sales (e.g., real estate or cars). NFTs can also be used for more utilitarian purposes, such as voting rights, membership, or benefits [2].

The first NFT was published in 2014 [3]. The market remained nascent for years but then grew dramatically in 2021 and peaked at $18 billion dollars [4]. The most expensive NFT bought by a single person went for $69.3 million in 2021 [5]. The market peaked at that point and has dropped significantly. For example, an NFT of the first tweet was sold in 2021 for $2.9 million; it was put up for auction in April 2023 and received the highest bid of $280 [36].

The purpose of this publication is to evaluate NFT technology and identify potential security concerns. This will promote the secure development of NFT implementations and raise awareness as to possible security concerns. The focus is on the smart contract representation and sales of NFTs and associated blockchain aspects.

A descriptive definition is provided to enable the reader to understand NFTs from a technical perspective. An NFT is not the asset "owned" but, rather, a data record within a smart contract. This definition is used to derive a set of properties inherent to NFTs. Each of these properties is then evaluated to identify 27 potential security concerns that should be addressed by NFT implementations.

Legal discussion and analysis of NFTs are out of scope for this paper; the focus here is on the technology. However, the legal aspects are just as important as the technical ones. Art Law & More says that

> The creation, distribution, ownership and trading of NFTs are new phenomena which raise a plethora of legal issues, many of which are ambiguous or unresolved... [For example,] there is practically no case law, legislation or regulation addressing smart contracts. This creates questions as to whether smart contracts are actually legally binding. [6]

Another major concern is that the purchase of an NFT does not necessarily convey the copyright (i.e., the purchaser cannot make, sell, or publicly display copies). Rather, the copyright often remains with the original owner, making such NFTs “digital autographs” [6]. For example, the previously cited $69.3 million NFT purchase did not convey the copyright of the art to the purchaser [21]. This is analogous to the physical world where the purchase of a painting or baseball card rarely conveys copywrite; if it does convey then “the transfer must be express and in writing” [37]. In general, the legal issues surrounding NFTs remain legally murky or unresolved. This is a new area undergoing maturation and legal precedent remains to be set. A discussion of the legal issues is available from [6], [2], and [37].

The remainder of this publication is organized as follows. Section 2 provides a short background on blockchains and tokens. Section 3 provides a descriptive NFT definition, a list of NFT properties, and related security considerations for each property. Section 4 is a summary of the 27 potential security concerns identified in Section 3.3. Section 5 reviews notable NFT standards. Section 6 discusses NFT marketplaces. Section 7 is the conclusion.

1.1. Scope

The focus of this paper’s research was on the most common NFT technology used, that based on the Ethereum Request for Comment 721 Non-Fungible Token Standard (ERC-721) and equivalent standards on other blockchains. All non-ERC-721-based NFT systems are out of scope of this paper.

Despite this narrow focus, much of the security analysis may still apply to out-of-scope NFT systems, even non-blockchain and non-smart contract approaches. Readers interested in such systems should consider how alternate approaches implement the NFT properties, enumerated in Section 3.2 and evaluated in Section 3.3. That analysis can then be used to determine which of the property-mapped security considerations listed in Section 4 apply to non-ERC-721 systems of interest.

Security analyses of NFT marketplaces are also out of scope. The focus in this work is on the NFT smart contracts and the services they provide (although this work does cover security concerns with non-blockchain stored assets and asset information). Security analyses of NFT marketplaces are available from [31] and [32].

1.2. Example Non-ERC-271 NFTs

An early example of a non-ERC-721 NFT is Colored Coins on the Bitcoin blockchain. These encode unique information within a coin’s metadata to allow it link to some asset while making it unique from all others. The metadata is encoded onto a Satoshi, which is the smallest unit of transfer for Bitcoin. Such coins are then changed from being fungible (i.e., interchangeable) to non-fungible (unique). Another newer example is Bitcoin ordinals with Bitcoin Request for Comment 20 (BRC-20) [34]. This encodes (JavaScript Object Notation) JSON metadata onto a Satoshi in a manner similar to Colored Coins but utilizing different methods.

2. Background

This section provides definitions for blockchains, smart contracts, and tokens as a foundation for the discussion of NFTs in Section 3.

2.1. Blockchains

According to NIST IR 8202, *Blockchain Technology Overview*, blockchains are “tamper evident and tamper resistant digital ledgers implemented in a distributed fashion (i.e., without a central repository) and usually without a central authority (i.e., a bank, company, or government)” [22]. NIST IR 8202 then provides a more formal definition:

> Blockchains are distributed digital ledgers of cryptographically signed transactions that are grouped into blocks. Each block is cryptographically linked to the previous one (making it tamper evident) after validation and undergoing a consensus decision. As new blocks are added, older blocks become more difficult to modify (creating tamper resistance). New blocks are replicated across copies of the ledger within the network, and any conflicts are resolved automatically using established rules. [22]

2.2. Smart Contracts

NIST IR 8202 defines a smart contract as follows:

> …a collection of code and data (sometimes referred to as functions and state) that is deployed using cryptographically signed transactions on the blockchain network. The smart contract is executed by nodes within the blockchain network; all nodes must derive the same results for the execution, and the results of execution are recorded on the blockchain. [22]

In simpler terms, a smart contract is program that runs on a blockchain. It processes transactions and records state while leveraging the cryptographic security of the blockchain.

2.3. Tokens

In the cryptocurrency community, the term token does not have an agreed upon definition. For the purposes of this publication, a token is a data record that is a digital representation of an asset (physical or virtual), managed by a smart contract, and stored on a blockchain. Tokens are not generally transferable between the smart contracts managing them, meaning that they are tied to a particular blockchain smart contract address. Each token represents some asset (e.g., cryptocurrency, digital artwork). Smart contract tokens usually follow one or more community token standards to enable interoperability with user wallets, exchanges, and other contracts (see Section 5). The transference of a token from one wallet to another involves the updating of the token owner’s address within the managing smart contract.

There are two types of tokens: fungible and non-fungible. An NFT is a non-fungible token, and it is unique and not interchangeable with others (a definition of NFTs is provided in Section 3). Fungible tokens are identical and interchangeable. They represent cryptocurrencies that are not native to a blockchain and are instead managed by smart contracts (e.g., stablecoins). In contrast, a native blockchain cryptocurrency is tied to the blockchain itself and is used to pay for blockchain gas (e.g., Bitcoin and Ethereum). Smart contracts represent fungible tokens by keeping a list of addresses that own tokens and how many tokens each address owns. Fungible tokens are often represented in smart contracts using the Ethereum Request for Comments (ERC) standard ERC-20 or a similar standard on a non-Ethereum blockchain. This is discussed in Section 5.1.

3. Definition, Properties, and Security Evaluations

This section provides a definition for NFTs, related properties, and an evaluation of each property to reveal potential security concerns.

3.1. NFT Definition

The definition provided below is intended to be descriptive and inclusive of all NFTs in use today. It is not intended to define what is and what is not an NFT nor is it intended to limit future NFT technology. The purpose of the definition and resultant properties is to enable the reader to understand current technology and to provide a foundation for an exploration of potential security issues.

> A non-fungible token (NFT) is an owned, transferable, and indivisible data record that is a digital representation of a physical or virtual linked asset. The data record is created and managed by a smart contract on a blockchain.

A common misconception is that an NFT is an asset. A person who buys an NFT might believe that they bought an asset. However, an NFT is not an asset. It is a token — a record maintained by a smart contract on a blockchain. The record may point to an asset or represent some intangible asset. The person buying an NFT is then buying a token. The token may or may not convey to the purchaser any legal rights over the asset.

NFTs are often represented by standard ERC-721 in smart contracts on Ethereum or a similar standard on another blockchain (see Section 5 on token standards for more details). These standards provide minimum functionality to be implemented by NFT implementations. Additional functionality is possible, even expected. For example, NFT smart contracts may have an owner role that can perform management functions (e.g., [28]). Such functionality can include upgrading to a new smart contract (e.g., [29]). Such upgrades can provide the owner arbitrary functionality, including the expiring or delisting of purchased NFTs (e.g., [29]).

3.2. NFT Properties

The following non-exhaustive set of NFT properties can be derived from this definition. Most correctly functioning and secured NFT implementations will contain these properties (see Section 3.3 for caveats to this).

1. **Owned:** NFTs designate ownership by recording a blockchain address.
2. **Transferable:** Owners and designated approved entities can transfer the ownership of NFTs to other addresses.
3. **Indivisible:** NFTs cannot be subdivided (although the ownership may be fractionalized).
4. **Linked:** NFTs have references to the asset that they represent.
5. **Recorded:** NFTs are smart contract data records stored on a blockchain.

6. **Provenance:** NFTs have their chain of ownership recorded.
7. **Permanence:** NFTs are normally indestructible (although some are designed to be burned).
8. **Immutable:** The asset that an NFT represents cannot be modified.
9. **Unique:** Each NFT represents a unique asset.
10. **Authentic:** Each NFT asset is what the NFT claims it to be (e.g., artwork from a particular artist).
11. **Authorized:** Each NFT asset has been authorized by an owner to be sold as an NFT.

3.3. Security Evaluation of NFT Properties

This section evaluates potential security issues related to each property presented in Section 3.2. Some of these properties should be provided by the NFT smart contract. Some are inherently provided by the underlying blockchain. Some should be provided by the human management of the NFT smart contract. All of these security issues are addressable through use of a systematic security approach to both design and implementation (such as [35]).

A concise enumeration of the potential security issues revealed in the below analysis is provided in Section 4.

3.3.1. Contract-Provided Properties

A properly constructed NFT smart contract should provide the properties of *owned*, *transferable*, *indivisible*, and *linked*. These properties are described below.

3.3.1.1. Owned

An NFT is often colloquially and incorrectly referred to as the “owned” asset. For example, a person may say that they own an NFT when referring to a piece of digital artwork. However, from a technical point of view, the NFT is a separate entity from the artwork. What an NFT owner definitively owns is a cryptocurrency token (they may or may not also own the linked asset). As defined previously, a cryptocurrency token is a data record managed by a smart contract and stored on a blockchain. The data record contains the metadata (i.e., a collection of data values) necessary to manage the NFT ownership and to link the NFT to a referenced asset.

This distinction is important because ownership of the token does not necessarily legally indicate ownership of the related asset (e.g., digital artwork). This is because a smart contract does not necessarily have the legal authority to designate ownership of a referenced asset (technically, anyone can create an NFT linked to anything). Exploring this legal issue is out of scope for this work. However, seller of NFTs should clearly convey the rights provided to purchasers and buyers should understand the stated rights prior to purchase.

It is tempting to think of these tokens like physical bills that can be handed from one person to another to change ownership. However, NFTs are maintained with the associated smart contract. This is because the tokens are data records of the smart contract that must stay with the smart contract and are, thus, normally locked into a specific smart contract and blockchain. The owner's cryptocurrency wallets then record the smart contract address and a token identifier (they don't hold the token as a physical wallet holds a bill). It is the smart contract that manages the tokens and is the authoritative repository for those tokens.

Smart contracts represent NFT ownership by keeping a list of unique tokens (i.e., data records) along with the owner of each token. The owner is identified only by a blockchain address. The data fields typically recorded for a non-NFT purchase are not present. There are no name, physical address, phone number, or other identifying data. This keeps the owners pseudonymous (identified only by their blockchain address). This is done for privacy concerns because all data stored by the smart contract are public on the blockchain.

If a blockchain account is compromised, then a malicious actor could obtain ownership rights for all NFTs owned by that account. This could happen, for example, through a blockchain wallet being hacked or through a blockchain account private key being stolen. The actor can then submit blockchain transactions to the NFT smart contract to transfer all of the blockchain account's NFTs to an account that they own (i.e., stealing the NFTs). They would likely then quickly sell the NFTs to avoid the (unlikely) possibility that the initial owner could convince the smart contract manager to reverse the transactions that stole the NFTs.

3.3.1.2. Transferable

The NFT smart contract provides functions to enable the transfer of tokens between owners. As previously discussed, the transfer of a token is simply an update to the ownership field in the token's smart contract data record. The owner is allowed to transfer a token to another blockchain address by submitting a blockchain transaction. Typically, the owner is also allowed to approve another address to take possession of the token as well as approve one or more accounts to manage tokens on the owner's behalf. See Section 5.2 for more details.

The smart contract may or may not be designed to allow the contract manager to transfer tokens. If the manager can transfer tokens, then stolen tokens could be restored. However, this becomes challenging if stolen tokens are quickly sold because there would then exist two owners who had spent funds and been granted NFT ownership by the smart contract. It could also be challenging for an owner to prove to the manager that their tokens were stolen, for example when an attacker steals a purchaser's private key and executes an otherwise valid transaction to change ownership of the NFT.

The default for smart contract NFTs following widely adopted standards is for the manager to not be able to transfer tokens. This makes the restoration of stolen tokens impossible, but it also provides owners with assurance that the manager will not confiscate their tokens (either maliciously or because of a legal order). However, NFT smart contracts likely have a mechanism to allow managers to update the code of the smart contract to provide for maintenance and upgrading of the NFT management infrastructure. The updated code could provide managers new privileges (including token transfer abilities) over both existing and to-be-created tokens.

If the smart contract contains coding errors, there may be a vulnerability that enables an attacker to steal tokens. An evaluation of possible vulnerabilities in NFT contracts is available from [30]. The attacker would then likely sell the tokens quickly to obtain cryptocurrency because after launching the attack, their approach would be publicly visible on the blockchain. Others could then launch the same attack, or the contract manager could use the same vulnerability to restore tokens to their owners. If the contract manager can regain control of the smart contract, tokens could be restored. However, the attacker would still have the funds obtained through illegal token sales and the sold tokens would each have two owners (the original owners to whom the tokens are restored and the subsequent purchasers that unwittingly bought the stolen tokens).

3.3.1.3. Indivisible

NFTs have the property that they are indivisible. This distinguishes them from fungible tokens that are divisible. An example of a divisible token would be a stable coin worth $1. This fungible token could be divided into two tokens, each worth $0.50. Since these assets are represented using numbers, it is simple to divide them.

However, an NFT that represents a piece of digital artwork could not be divided in the same manner. Digital assets are typically non-fungible, meaning that they cannot be simply cut in two without damaging the original asset.

On a more technical level, an NFT is a token (explained in Section 2.3). A token is represented in a smart contract by a data record. Indivisibility then refers to the inability to divide the NFT data record into multiple parts. Data records do not naturally divide; they represent values for a fixed set of variables.

Some NFT owners may wish to divide their NFT by providing fractionalized ownership. The actual NFT itself is not split into multiple parts but, instead, locked into a new fractional NFT smart contract that then creates a specified number of new fungible tokens. These new fungible tokens represent shares of ownership of the NFT and can be traded, purchased, and sold on marketplaces (see Section 6) that specialize in fractional NFT sales (such as [7]). The largest fractional NFT sale to date – “The Merge” digital art – was bought jointly by 28,000 purchasers for $91.8 million [5].

Typically, a fractional NFT smart contract has a function that allows a buyout that can reverse the fractionalization process. This enables the original owner or a fractional investor to reclaim all of the ERC-20 fractional tokens and unlock the ERC-721 NFT from the fractional management smart contract. Unlocking the NFT means transferring ownership away from the fractional smart contract to the new owner. There are multiple methods by which to implement a buyout mechanism.

A common approach is to hold an auction. It requires a buyer to transfer a set amount of a specific ERC-20 fractional token to the smart contract. This then begins a time-limited auction in which all fractionalized owners can bid to keep their fractional shares. If the buyer wins, all ERC-20 tokens are reclaimed by the smart contract, and the buyer becomes the sole owner of the NFT. If the other users outbid the buyer and win the auction, then the buyout was unsuccessful, and the NFT remains fractionalized. If the NFT is successfully bought out, the fractionalized owners are compensated proportionally to the number of fractions that they held. If the buyout is unsuccessful, then the buyer is compensated with the amount that the remainder of fractionalized owners bid, and the fractionalized owners are proportionally compensated with the ERC-20 fractional tokens that the buyer originally transferred to initiate the buyout.

Another approach is to for the smart contract to specify an exit price at which an immediate purchase can be made of all fractional shares.

Appendix B provides an example of fractionalizing an NFT and then someone buying it back at an auction.

3.3.1.4. Linked

Every NFT must be linked to the asset that it represents. More specifically, each NFT data record must have a field or fields that uniquely identify and link to an asset. This collection of information is referred to as the NFT's metadata. The metadata may contain additional descriptive information that is not necessary for identification. An example of such data would be the secure hash of a digital image along with the image's title, creation date, artist name, and a public URL. Metadata can be included in the NFT data record but is often stored publicly and only a link to the metadata is stored on the blockchain. There are multiple approaches to link an NFT to an associated asset using metadata [8].

The metadata can store the asset itself on the blockchain, inside the smart contract. This approach is the most secure as it leverages the integrity of the blockchain itself, but it can be expensive to store data there. This is rarely done for NFTs.

The more common approach is to store on the blockchain a Uniform Resource Locator (URL) or content identifier to an external data source that hosts the digital asset. Non-blockchain public data publishing is much cheaper. Sometimes the identifier will link directly to an asset. This link is usually not to a particular server but, instead, to a file storage service. These storage services can be centralized (but internally distributed with redundancy) or fully decentralized (e.g., with the InterPlanetary File System (IPFS) protocol [9]). Either way, the off blockchain linkage complicates security as an additional attack surface is added to the NFT architecture.

Further complicating the architecture, the linking information is usually not to the asset itself but, instead, to a publicly accessible JSON table of NFT identifiers that provides the URLs for each asset and other metadata [33]. This double linking architecture allows for the asset URLs to be updated by the manager of the table (e.g., NFT marketplace). Note how the owner of the table maintains continued control over where each NFT is linked.

It is critical that the metadata correctly identifies and links to the asset represented by the NFT. A delinked NFT is unlikely to maintain its value. An NFT might be delinked if the original metadata is incorrect, never being linked to any actual asset. NFTs can also be delinked if the public table maintaining the asset URLs fails, is deleted, or is changed. Even for NFT data records with direct URLs to their asset, the server could cease to exist or fails in some way (e.g., corrupted files). One study, with a sample size of 12 353 NFTs, found that 25 % of NFTs were linked to assets that were either lost or inaccessible [33].

If an attacker breaks into the public table mapping NFT identifiers to URLs, the NFT could be delinked, or the links and associated metadata could be changed. This could enable an attacker to swap a cheap NFT asset that they bought for someone else's very expensive one by swapping URLs in the public link table. This could also enable the owner of the table to delink NFT owners from the assets that they purchased. There would be no need to change anything on the blockchain or to access the smart contract. The owner could simply modify the metadata to delink an NFT from its associated asset. Someone who purchased an expensive NFT could be left owning a worthless delinked token on the NFT smart contract.

NFTs for physical objects, often referred to as *physical NFTs*, link to their associated physical asset by including a unique identifier in their metadata. This unique identifier is then materially attached to the physical object [26]. This could be accomplished through the use of a near field communication (NFC) tag, quick response (QR) code, or simply permanently embedding the identifier in the physical asset. For significant assets (e.g., real estate), a linkage would need to be made to the public records to prevent fraud. This is a nascent area around which legal precedents have not been established [26].

3.3.2. Blockchain-Provided Properties

The associated underlying blockchain should provide the properties of *recorded, provenance*, *permanence*, and *immutable*. These properties are described below.

3.3.2.1. Recorded

An NFT is a cryptocurrency token (see Section 2.3). Tokens are data records managed by a smart contract. Smart contract state is *recorded* on a blockchain. This property of NFT state being recorded on a blockchain grants the smart contract and associated NFT the benefits of leveraging a blockchain architecture. These benefits include the properties of *provenance*, *permanence*, and *immutable* (discussed in the following subsections).

The recording of an NFT on a blockchain normally it makes information about the NFT and its ownership (the metadata) public information. Owner accounts are pseudonymous, meaning that the owners are anonymous, but information about their accounts (i.e., which NFTs they own) is public. Accounts may be de-anonymized when an account owner provides personal information (e.g., name and address) when making a purchase using cryptocurrency. This can be mitigated by cryptocurrency users maintaining multiple accounts (separating NFT purchases from other purchases).

3.3.2.2. Provenance

A fundamental property of a blockchain is its ability to track tokens over their entire lifetime. The creation event, every transaction involving it, and the destruction event are all recorded. The blockchain records when these events occurred, as well as the sender and receiver of the transactions. The blockchain provides a complete history of ownership of the token.

This complete history of ownership is beneficial to anyone who wishes to validate the authenticity of a token. It is a simple endeavor to work back from any point of a token's transaction history and determine its origin and where it has been. The ability to validate a token's history can help a user determine whether a token is fraudulent or legitimate.

A blockchain could undergo an attack (e.g., 51 % attack [25]) that enables a malicious entity to change the blockchain history, but this is unlikely for established and widely used blockchains due to the significant resources dedicated to maintenance of those chains (e.g., either mining processing power or large staked holdings).

3.3.2.3. Permanence

A fundamental property of a blockchain is its ability to record data in a near-permanent manner based on its decentralized storage and cryptographic mechanisms. Other than the previously referenced 51% attack [25], there are some exceptions to a blockchain's permanence.

One way to sidestep the property of permanence is to "burn" the NFT. Transferring an NFT to an address that no one can access renders any further use of the NFT impossible. For example, sending any transaction to the Ethereum address "0x0000000000000000000000000000000000000000" will effectively destroy whatever is sent because there is no known private key that resolves to this address (and it is extremely unlikely for someone to find it) so no one can access the account. Other blockchains have specific addresses that the underlying blockchain code will prevent from sending transactions but can still receive transactions. These are hard coded burn addresses, so even if someone were to discover a private key that would resolve to that address, they could not claim any asset associated with it.

There may be legitimate use cases for burning an NFT, such as to provide proof of burning to receive an upgraded NFT in a different smart contract or if the NFT is a consumable object in a blockchain-based video game (i.e., a unique item that provides some benefit for the player). Even though the NFT is burned, it still technically exists in the smart contract on the blockchain.

Another way to sidestep the property of permanence would be for the NFT's smart contract to have the ability to call a method selfdestruct(). In practice, this method is used by many smart contracts to stop its execution and remove the current state from the blockchain (previous states are still recorded in past blocks). While there is nothing to technically prevent an NFT smart contract from using the selfdestruct() or similar method, it is strongly discouraged. The NFT smart contract manages the tokens and records all information about them, including ownership. If a NFT smart contract could call a selfdestruct() method, then all of its associated information would be removed from the blockchain's current state and become effectively lost.

Since all of the NFT information is contained within the smart contract, a user wallet does not reflect that it owned an NFT but simply that it sent funds to an address. Potential buyers of an NFT are strongly encouraged to limit their investment risk by ensuring that the smart contract will provide permanence through either direct inspection or trusting the services of another firm that evaluates smart contracts.

Another issue with permanence is if the NFT content is too large to be stored within the smart contract, and the smart contract instead contains a pointer (e.g., uniform resource locator (URL) or uniform resource identifier (URI)) to an external storage source (e.g., IPFS or some other external data). If the data source should cease to host the NFT itself, then the owner may lose access to the actual NFT content. This is related to the material covered by the linkage property in Section 3.3.1.5.

3.3.2.4. Immutable

An NFT is expected to have the property of being unchanging or immutable. NFT smart contracts enforce this in their code. However, a vulnerability in the smart contract could enable a malicious entity to change NFT data records.

More fundamentally, this is a property provided by the blockchain to ensure that ledger entries are not altered. This normally holds but is not guaranteed. True immutability – to never be changed under any circumstances ever – is not achieved. While blockchains are effectively immutable, there have been cases in which a blockchain has been altered by group consensus (e.g., [27]). True immutability of digital data is very difficult if not impossible to achieve. Under normal operating conditions, a blockchain provides a cryptographically secure ledger that resists alterations of recorded data (a tamper-resistant design), and a participant can detect and discard alterations (a tamper-evident design) if desired. This may lead to a chain split (or cryptocurrency fork) where a portion of the users accept the alterations, while another portion does not, leading to incompatible blockchain records between them. By combining tamper-resistant and tamper-evident designs, a blockchain can provide a near immutable ledger.

3.3.3. Human Management-Provided Properties

The human management of the NFT smart contract should provide the properties of *unique*, *authentic*, and *authorized*. These properties are described below.

3.3.3.1. Unique

The non-fungible aspect of an NFT requires that only one exists. From a technical perspective, this is guaranteed because the NFT smart contract ensures that the data record owned by the purchaser is one-of-a-kind and has a single owner. However, that does not mean that the linked asset is uniquely owned by that data record. The issuer may sell multiple NFTs linked to the same asset (e.g., for digital trading cards). This may be analogous to an artist making a limited run of identical copies of a specific piece of art.

Alternatively, there may be multiple smart contracts with data records linked to that asset. The same virtual object could be sold on multiple NFT marketplaces. To check for this, one could compare the hash values of the virtual object with other virtual objects being sold. However, one could change just a single pixel of a virtual image to obtain a completely different hash value. An artist could also have made many copies of the same artwork, or duplicates of original art are being sold in NFT form.

3.3.3.2. Authentic

In an NFT sale, it is implied that the linked asset is what the seller claims it to be. However, an asset could be a forgery whose origin is misrepresented. The seller may claim to have created something that they simply copied from the internet, or they may attribute the artwork to another artist to increase the sale price. To a large extent, the purchaser must rely on the selling smart contract and the associated NFT marketplace to ensure authenticity.

3.3.3.3. Authorized

The smart contract guarantees that only the current owner of an NFT can sell an NFT data record. However, whether the original seller is in fact authorized to sell an NFT that is linked to a particular asset is a legal question that is out of scope for this publication. From a technical point of view, anybody can sell an NFT linked to anything. This creates a potential for misrepresentation or fraud that must be addressed by non-technical controls (e.g., a legal framework). What is being directly sold is the smart contract data record, and the owner of the linked asset does not need to be involved. Ownership of the data record might convey rights over the linked asset, but that is a legal question. In many cases, the buyer does not obtain any rights whatsoever to the linked asset. For example, one NFT marketplace clearly specifies that "the purchase of an NFT does not give the buyer the right to every copy of the underlying work, nor the right to reproduce, distribute, commercially exploit, publicly perform, or publicly display the NFT or objects included as part of the work" [20]. In such cases, the right being provided the purchaser is the privilege to digitally autograph the asset and to subsequently sell that right to another.

4. List of Potential Security Concerns

This section lists 27 potential security concerns that can exist with NFT ownership and smart contract management of tokens. The identified security concerns are organized by NFT property.

Owned (Section 3.3.1.1)

1. An NFT purchaser may be deceived into thinking that they are purchasing an asset instead of a smart contract data record that contains a reference to the asset (possibly conferring no rights over the asset at all).
2. A smart contract may create a token linked to an asset without the legal authority to do so for that asset since, technically, anyone can create an NFT linked to anything.
3. If a blockchain account is compromised, the malicious actor can transfer all NFTs associated with that address to an address owned by the actor.
4. Stolen tokens will likely be sold immediately by malicious actors for cryptocurrency, preventing easy restoration of the tokens even if a mechanism is available to do so.

Transferable (Section 3.3.1.2)

5. There is likely no smart contract mechanism to restore stolen tokens to their rightful owner.
6. If a smart contract enables the contract manager to restore stolen tokens, this feature could be used by the manager to confiscate, freeze, or unilaterally transfer tokens.
7. A smart contract may not allow a manager to restore stolen tokens, but the smart contract may have a manager-controlled update mechanism whereby this feature could be added in the future (enabling the previously mentioned security concern).
8. Coding errors in the smart contract could enable attackers to steal tokens and transfer them to their accounts.

Indivisible (Section 3.3.1.3)

9. Fractional ownership increases the NFT attack surface by involving an additional third-party smart contract that handles the fractional ownership.
10. Owners of fractional shares may not be aware that they could lose their shares through a forced buyout.

Linked (Section 3.3.1.4)

11. Inaccurately stored metadata (either done maliciously or accidentally) can delink an NFT from the asset it represents and make it worthless.
12. Server errors that make a digital asset unavailable (e.g., corrupted file, server failure, or storage service discontinuation) could effectively delink an NFT from the asset it represents and make it worthless.

13. If the off-blockchain table linking NFT identifiers to URLs is compromised, an attacker could delink NFTs from their assets and/or change which NFTs represent which assets.
14. If off-blockchain tables are used to link NFT identifiers to URLs, the owner of the table could use their access to delink NFTs and/or change which NFTs represent which assets.

Recorded (Section 3.3.2.1)

15. An NFT owner may not realize that their account and information on the NFTs that their account owns are public information on the associated blockchain.
16. While blockchain accounts are anonymous, they can be de-anonymized through account owner purchases that include personally identifying information (e.g., name and address).

Provenance (Section 3.3.2.2)

17. A blockchain could undergo an attack enabling changes to blockchain history (this is unlikely with established blockchains).

Permanence (Section 3.3.2.3)

18. An NFT may be burned (accidentally or maliciously) by sending it to an address no one has access to.
19. An NFT smart contract could self-destruct, destroying the managed NFTs.

Immutable (Section 3.3.2.4)

20. If the smart contract code contains a vulnerability, the data records could be changed by a malicious actor.
21. Blockchains are occasionally changed through participant consensus or have their chains split into distinct and different versions when consensus is not reached on resolving a major issue.
22. A blockchain split will result in the duplication of NFT contracts, which in turn results in NFT owners having the same NFTs on two blockchains. They could sell one and keep the other, causing significant issues for NFTs that convey ownership rights over their linked asset.

Unique (Section 3.3.3.1)

23. Buyers may not be aware that an exchange is selling the same NFT multiple times (e.g., permitting a limited number of autographs for video clips).
24. The same asset (or copies with unperceivable changes to humans) could be sold simultaneously by multiple NFT exchanges or smart contracts.

Authentic (Section 3.3.3.2)

25. An asset linked to an NFT may be a forgery or an authentic original artwork whose origin is misrepresented or attributed to a different creator (e.g., to increase its perceived value).

Authorized (Section 3.3.3.3)

26. The seller may not be authorized to sell an NFT linked to a particular asset.
27. The buyer may be deceived into not receiving the rights over the linked asset that they think they are obtaining by purchasing an NFT.

5. Token Standards

NFT standards build upon the work done in fungible token standards and modify token definitions so that each token is unique. Standards are critical for all types of cryptocurrency tokens so that cryptocurrency exchanges can easily adopt them, smart contracts can accept and manage them, and user wallets can buy and sell new token types. Such standards define the services and interfaces for token smart contracts. The standards are typically represented in the form of code that has mandatory inheritable functions. They are often created and managed within cryptographic communities and are, thus, community standards that are not associated with traditional formal standards bodies.

Many token standards [10] are in the form of an Ethereum Request for Comment (ERC) [11] because Ethereum was the first blockchain platform to provide tokens. ERCs are standards for Ethereum, and they provide requirements for smart contract interface design. As additional blockchains have incorporated token management into their platforms, ERCs have been ported and/or used as the basis for developing equivalent standards on the other platforms.

5.1. ERC-20: Fungible Token Standard

ERC-20 was the first fungible token standard [13]; it defines a minimum interface for smart contracts that provide interchangeable and identical tokens. Compliant contracts provide functions that return the following state information:

1. The name of the token,
2. The symbol,
3. The total token supply,
4. The balance for each owner, and
5. The amount that an “approved” spender is allowed to transfer from an owner’s account.

Additional required functions manage the token transfers:

1. The owner transfers a specified number of their tokens to an address.
2. The owner approves an address to transfer a certain number of their tokens.
3. An approved spender transfers a specified number of tokens from one address to another address (limited by the amount specified by the owner).

An ERC-20-compliant smart contract must emit an “event” for every transfer and address approval. An event is an entry in a blockchain log and is, thus, publicly viewable by all blockchain users.

5.2. ERC-721: Non-Fungible Token Standard

ERC-721 functions, similar to ERC-20, define a minimum interface for smart contracts that provide unique tokens. Compliant contracts provide functions that return the following state information:

1. The owner of an NFT,
2. The number of NFTs assigned to each owner,
3. The address "approved" to transfer an NFT, and
4. Whether or not an address is an "authorized operator" for another address.

Additional required functions manage the token transfers:

1. The owner, an approved address, or an authorized operator transfers tokens from one address to another.
2. The owner, an approved address, or an authorized operator "safely" transfers tokens from one address to another (checking that the recipient smart contract can receive NFTs).
3. The owner or an authorized operator sets the address that is "approved" to transfer an NFT.
4. The owner updates the status of an address relative to being an "authorized operator" to manage all their NFTs.

Like ERC-20, a compliant smart contract must emit an "event" for every transfer, address approval, and "authorized operator" change of status.

The transfer "safely" function is based on ERC-165 [14]. The NFT contract checks to see whether the recipient of an NFT is a smart contract or a user by checking the code size of the recipient address. If the recipient is a contract, the NFT contract calls the "onERC721Received" function in the recipient contract to determine if the contract is prepared to accept an ERC 721 token. It checks for a return value of the Keccak-256 hash for a specified string (comprising the function call and its parameters). If the correct return value is not supplied (often because the "onERC721Received" function does not exist since the contract isn't set up to receive tokens), then the transfer is reverted.

An example ERC-721 smart contract is available at [15].

5.3. Other NFT Standards

Other related NFT standards are provided below in Table 1.

Table 1. Other NFT Standards

Standard	Description
ERC-1155: Multi-Token Standard	ERC-1155 provides for both fungible and non-fungible tokens in the same smart contract [16]. With ERC-1155, a single smart contract can simultaneously support ERC-721 and ERC-20 functionality while managing multiple token types.
ERC-2309: ERC-721 Consecutive Transfer Extension	ERC-2309: ERC-721 provides “a standardized event emitted when creating/transferring one, or many non-fungible tokens using consecutive token identifiers” [17].
ERC-4400: EIP-721 Consumable Extension	ERC-4400 enables a “consumer” role for NFTs [18]. Consumers can perform limited operations upon NFTs without owning them. For example, if an NFT represents a parcel of digital land in a virtual universe, a consumer of the NFT might be allowed to modify the property (as if they were renting it) but would not be the owner (could not transfer ownership).
ERC-4907: Rental NFT	ERC-4907 enables a “user” role for NFTs [19]. Users can use the NFT for a specified period of time, but they cannot transfer ownership of the NFT or enable other users. An example would be a virtual tool in a game that allows a user to build virtual objects but only during their specified time limit.

6. Marketplaces and Exchanges

NFT marketplaces (also called exchanges) enable users to buy, sell, and mint (i.e., create) NFTs [23]. The marketplaces should provide some level of verification for the posted NFTs. The oldest was launched in 2017, making both NFTs and their exchanges relatively new technology.

These marketplaces have an attack surface separate from the associated NFT smart contracts and may be the target of hacking activity. As mentioned in Section 1.1, a security analysis of NFT marketplaces is out of scope of this publication (but see [31]and [32]). Here, a brief overview of NFT marketplace security models is provided.

NFTs can be bought through direct purchase, by participating in an auction, or by making an offer. For exchanges that use a decentralized finance (DeFi) approach, customers need their own cryptocurrency wallet (either software or hardware). Alternatively, exchanges may use a centralized finance (CeFi) approach in which customers use custodial wallets provided by the exchanges. In the CeFi model, the exchange is the custodian of the cryptographic keys and holds the NFTs on behalf of their customers (analogous to an investment firm acting as a custodian and holding stock for its clients). In the DeFi mode, the purchaser uses a wallet to hold the cryptographic keys that grant them ownership of the NFTs.

In both approaches, a malicious entity could compromise the user-owned wallet (for DeFi NFT approaches) or the custodial wallet system (for CeFi NFT approaches). The former requires the user to secure their own wallet; many cryptocurrency wallet users have had their cryptographic keys stolen. The latter requires the user to trust the CeFi custodian to secure their NFTs in custodial wallets; cryptocurrency custodial systems have been hacked, resulting in the loss of user assets. There is no guaranteed security for crypto assets. Guidance for the security of cryptographic wallets is out of scope for this publication, though many resources are available online (e.g., [24]).

While some take credit cards and other forms of traditional payment (usually with an additional processing fee), marketplaces may only accept cryptocurrency, which is the preferred form of payment. This is because NFT data records are managed by smart contracts, and smart contracts only accept cryptocurrency. Also, marketplaces may not be able to handle fiat currencies due to associated regulatory requirements. Also, accepting fiat currency requires additional centralization of the marketplace architecture and many strive to maintain a decentralized model.

7. Conclusion

Currently, most NFT sales are of the digital autographing type. This makes most NFTs prestige purchases where the buyer obtains the right from the linked asset's copyright holder to uniquely link their name to the asset in a smart contract data record on a blockchain. However, NFTs are also used for actual sales of assets (both digital and physical) as well as for utilitarian purposes such as voting rights, membership, and benefits. These latter use cases necessitate a robust and secure design and implementation of NFTs.

NFT reliance on blockchains and smart contracts provides secure cryptographic methods for establishing and publicly recording ownership. The NFT smart contracts provide the NFT properties of *recorded*, *owned*, *transferable*, *indivisible*, and *linked*. The blockchain ensures *provenance*, *permanence*, and *immutable*. Human NFT management provides the properties of *unique*, *authentic*, and *authorized*.

Despite a solid cryptographic foundation, there are potential security concerns related to these NFT properties, this work identified 27 by evaluating the 11 NFT properties. Each of these can be addressed through considering security upfront and creating a secure design and implementation. Adoption of a systematic security approach, such as the NIST Risk Management Framework [35], can help address these potential concerns. While further research should be conducted in this area, the security analysis in this work did not reveal any non-addressable weaknesses that would undermine the overall approach and technology.

References

[1] Investopedia (2023) *Non-Fungible Token (NFT): What It Means and How It Works.* Available at https://www.investopedia.com/non-fungible-tokens-nft-5115211

[2] Holbein (2022) *Evolving Legal Issues for NFTs.* Available at https://www.jdsupra.com/legalnews/evolving-legal-issues-for-nfts-5461995

[3] Mettei (2023) *Code is Not Law: Case on Who Owns the First NFT Dismissed by Judge.* Available at https://www.artnews.com/art-news/news/kevin-mccoy-quantum-case-dismissed-free-holdings-sothebys-1234662076

[4] BCC Publishing (2022) *Non-Fungible Tokens (NFTs): Global Market.* Available at https://www.bccresearch.com/market-research/information-technology/nft-market.html

[5] Shewale (2023) *12 Most Expensive NFTs Ever Sold.* Available at https://www.demandsage.com/most-expensive-nfts

[6] Clark, Aujla, Gould (2021) *What are the Legal Issues Concerning Non-Fungible Tokens (NFTs)?* Available at https://artlawandmore.com/2021/07/08/what-are-the-legal-issues-concerning-non-fungible-tokens-nfts

[7] Fractional (2023) *Buy and Sell Fractions of NFTs.* Available at [https:// docs dot fractional dot art / fractional]

[8] Nico (2021) *How to Make an NFT in 14 Lines of Code.* Available at https://www.freecodecamp.org/news/how-to-make-an-nft

[9] IPFS (2023) *What is IPFS.* Available at https://docs.ipfs.tech/concepts/what-is-ipfs

[10] Ethereum (2022) *Ethereum Development Standards.* Available at https://ethereum.org/en/developers/docs/standards

[11] Crypto.com (2022) *What are Token Standards? An Overview.* Available at https://crypto.com/university/what-are-token-standards

[12] Crypto.com (2023) *What is the BRC-20 Token Standard for Bitcoin?* Available at https://crypto.com/university/brc-20-token-standard-bitcoin

[13] Ethereum (2023) *ERC-20 Token Standard.* Available at https://ethereum.org/en/developers/docs/standards/tokens/erc-20

[14] Zhang (2019) *Ethereum Standard ERC165 Explained.* Available at https://medium.com/@chiqing/ethereum-standard-erc165-explained-63b54ca0d273

[15] OpenZeppelin (2023) *ERC721.sol.* Available at openzeppelin-contracts/ERC721.sol at master · OpenZeppelin/openzeppelin-contracts · GitHub

[16] OpenZeppelin (2023) *ERC-1155 Multi-Token Standard.* Available at openzeppelin-contracts/ERC721.sol at master · OpenZeppelin/openzeppelin-contracts · GitHub

[17] Papanikolas (2019) *ERC-2309: ERC-721 Consecutive Transfer Extension.* Available at https://eips.ethereum.org/EIPS/eip-2309

[18] Ivanov (2021) *ERC-4400: EIP-721 Consumable Extension.* Available at https://eips.ethereum.org/EIPS/eip-4400

[19] Anders, Lance, Shrug (2022) *ERC-4907: Rental NFT, an Extension of EIP-721.* Available at https://eips.ethereum.org/EIPS/eip-4907

[20] Verisart (2023) *What is an NFT?* Available at https://help.verisart.com/en/articles/5647641-what-is-an-nft

[21] Wikipedia (2023) *Everydays: the First 5000 Days.* Available at https://en.wikipedia.org/wiki/Everydays:_the_First_5000_Days#:~:text=Sundaresan%20receives%20rights%20to%20display,view%20through%20a%20web%20browser

[22] Yaga D, Mell P, Roby N, Scarfone K (2018), Blockchain Technology Overview. (National Institute of Standards and Technology, Gaithersburg, MD), NIST Interagency or Internal Report (IR) NIST IR 8202. https://doi.org/10.6028/NIST.IR.8202

[23] Rodeck (2023) Top *NFT Marketplaces of June 2023*. Available at https://www.forbes.com/advisor/investing/cryptocurrency/best-nft-marketplaces

[24] Hojjati (2022) *How to Secure Your Crypto Wallet Against Hacks.* Available at https://www.digicert.com/blog/how-to-secure-your-crypto-wallet-against-hacks

[25] Frankenfield (2022) *51% Attack: Definition, Who is at Risk, Example, and Cost*. Available at https://www.investopedia.com/terms/1/51-attack.asp

[26] Binance (2022) *Physical NFTs: Bridging the Gap Between Digital and Physical Worlds*. Available at https://www.binance.com/en/blog/nft/physical-nfts-bridging-the-gap-between-digital-and-physical-worlds-7460772280213595786

[27] Wikipedia (2023) *The DAO (organization)*. Available at https://en.wikipedia.org/wiki/The_DAO_(organization)

[28] Bored Ape Yacht Club (2023). *Contract 0xBC4CA0EdA7647A8aB7C2061c2E118A18a936f13D*. Available at https://etherscan.io/address/0xbc4ca0eda7647a8ab7c2061c2e118a18a936f13d#code.

[29] OpenSea (2022). *Announcing a contract upgrade*. Available at https://opensea.io/blog/articles/announcing-a-contract-upgrade

[30] Yang, Shuo, Jiachi Chen, and Zibin Zheng. "Definition and Detection of Defects in NFT Smart Contracts." arXiv preprint arXiv:2305.15829 (2023). Available at https://arxiv.org/pdf/2305.15829.pdf

[31] Stöger, Felix, et al. "Demystifying Web3 Centralization: The Case of Off-Chain NFT Hijacking." Available at https://fc23.ifca.ai/preproceedings/156.pdf

[32] Das, Dipanjan, et al. "Understanding security issues in the NFT ecosystem." Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security. 2022. Available at https://arxiv.org/pdf/2111.08893.pdf

[33] Wang, Ziwei, Jiashi Gao, and Xuetao Wei. "Do NFTs' Owners Really Possess their Assets? A First Look at the NFT-to-Asset Connection Fragility." Proceedings of the ACM Web Conference 2023. 2023. Available at https://arxiv.org/pdf/2212.11181.pdf

[34] Sharma (2023). *BRC-20 Tokens: A Primer*. Available at https://research.binance.com/static/pdf/BRC-20%20Tokens%20-%20A%20Primer.pdf

[35] Joint Task Force (2018). Risk Management Framework for Information Systems and Organizations: A System Life Cycle Approach for Security and Privacy. NIST SP 800-37 Rev. 2. https://doi.org/10.6028/NIST.SP.800-37r2

[36] Thubron (2023). *Auction for the $2.9 million Jack Dorsey tweet NFT has a high bid of $1,871*. Available at https://www.techspot.com/news/99510-auction-29-million-jack-dorsey-tweet-nft-has.html

[37] Dreben, Pennington (2021). Nonfungible Tokens and Copyright: Diligence Issues to Consider. Available at https://www.morganlewis.com/pubs/2021/04/nonfungible-tokens-and-copyright-diligence-issues-to-consider

Appendix A. List of Symbols, Abbreviations, and Acronyms

BRC

Bitcoin Request for Comment

ERC

Ethereum Request for Comment

F-NFT

Fractionalized non-Fungible Token

IR

Interagency or Internal Report

JSON

JavaScript Object Notation

NFT

Non-Fungible Token

IPFS

InterPlanetary File System

QR

Quick Response

URI

Uniform Resource Identifier

URL

Uniform Resource Locator

Appendix B. Fractional Token Example

Consider a person buying an image NFT from a marketplace. The NFT smart contract records the owner's blockchain address in the NFT's data record. To fractionalize, the owner transfers the NFT to a fractionalized NFT (F-NFT) smart contract. The original NFT smart contract records the F-NFT contract as the owner. The NFT is now "locked" in the F-NFT contract. The F-NFT contract then sells 10 ERC-20 tokens for 1 Ether (ETH) each and gives the proceeds to the original owner, minus a fee. Five users buy the tokens:

- Alice: 4 $JPEG
- Bob: 1 $JPEG
- Carol: 2 $JPEG
- Dave: 1 $JPEG
- Erin: 2 $JPEG

The F-NFT contract specifies a buyout function that requires at least four of the tokens be deposited to start the auction. Eventually, Alice decides that she wants the whole NFT to herself and deposits her four tokens to initiate the buyout.

Alice bids 1.1 ETH per token. If she wins, she will need to pay 6.6 ETH to purchase the remaining six tokens and claim the original NFT for herself. The other fractional owners would then split the 6.6 ETH proportionally according to the number of ERC-20 tokens that they hold.

If Bob, Carol, Dave, and Erin collectively bid 1.2 ETH per token and outbid Alice, they would then pay 4.8 ETH to Alice and receive a fraction of the four ERC-20 tokens that Alice had deposited, proportional to the amount that each owner contributed. If each of them paid 1.2 ETH, then they would each gain one additional token (representing fractional ownership) after outbidding Alice.