# Enhanced Non-Fungible Tokens

Ivan Visconti
University of Salerno

Work in progress based on results achieved with Pierpaolo Della Monica, Andrea Vitaletti and Marco Zecchini

# Possession: Needs vs Exhibitionism

Some people like to own **expensive objects**, often to **show off** their power to enjoy in full the features of those objects, while others can only get a limited access to them, often envying the owner.

Classical examples:

- A soccer player  keeping 5 expensive cars in a garage, showing them to visitors from time to time
- A CEO owning  a long yacht inviting sometimes some other (minor) CEOs for a short trip with her yacht
- A politician having in his house expensive pictures of famous painters to impress her guests

# Possession ⇒ Value ⇒ Business

Another reason pushing towards owning an object and giving only a limited flavor of it to others is **to make a business out of it**, trying to generate interest and high demand to later on trade the ownership of the object and/or some rights to access to it.

Classical examples: real estate, old luxury cars, rare stamps/coins

**Collections** of highly researched objects are therefore created and managed to make a business out of the desire of owning them and/or having a partial (and thus limited) access to them.

# Definition of Collection

A **collec**tion is a group of objects of a single type collected in one place, usually by an individual or an organization.

The collectibles are grouped according to some logic (historical, artistic, scientific) or personal taste.

# Market of physical collectibles

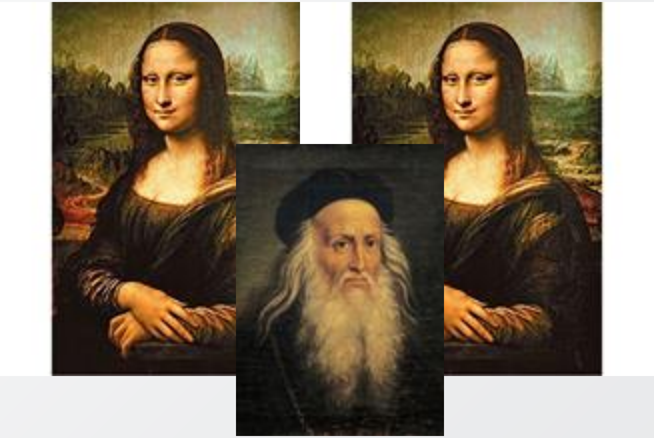
Typically, collectibles in the physical world gain their value from:

- Origin/Author (who has realized it?)
- Scarcity (how many pieces have been realized by the author?)

# Forgeries vs Value in the Physical World

- It is crucial to exploit the **scarcity** of an artwork, therefore forgeries must be hard to realize
- If one can create perfect copies of a unique object then how can we **distinguish** between original and copies?

**The original value of the artwork would be severely reduced!**



# What about the digital world?

No scarcity.
Every file is replicable.
Data replication in many cases is a must (e.g., backups)

# What about the digital world?

**Q1**: How can we  guarantee that a buyer of a digital artwork in a collection will not be penalized by the generation of additional **identical copies** of that artwork in the same collection?

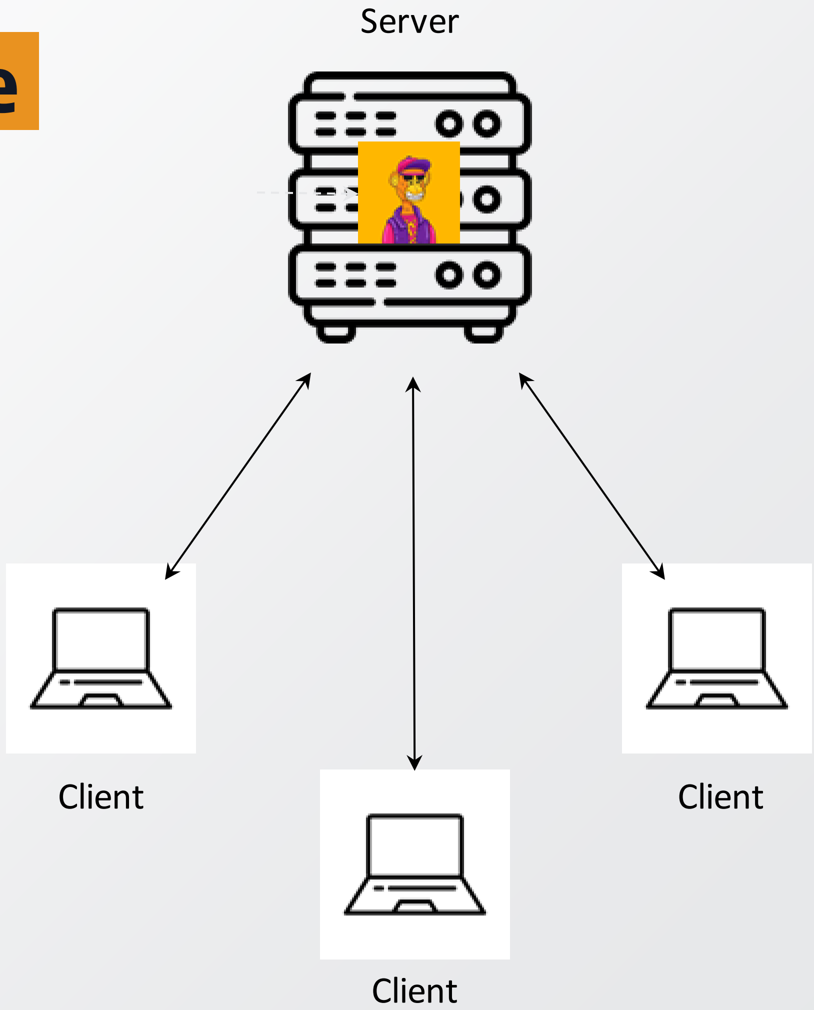
Q2: How can we  allow the owner of a digital artwork to decide **how much** of it will be **visible to others** willing to pay for it?

Q3: How can we build a system where **ownership and full access** to the digital artwork **can jointly be transferred** from seller to buyer, **while others remain excluded**?

# Life is easy as usual if we trust others

We could **trust** an intermediary for maintaining the collection and implementing **access-control** policies to limit the exposed information.

This trusted third party managing the collection will also guarantee that **there will not be clones** (i.e., two assets corresponding exactly to the same digital data).

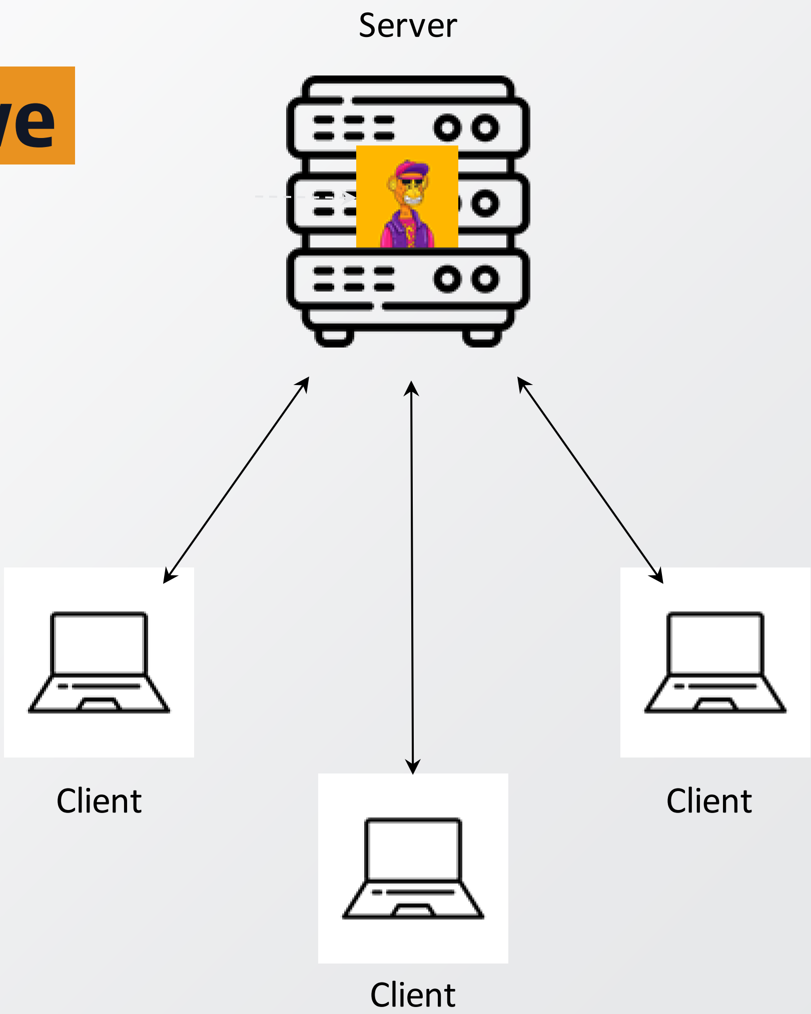


# Life is easy as usual if we trust others

However, the bitter truth is that trusted third parties are **vulnerable to corruption** when there is **high value** in their business.

Moreover they can be **successfully attacked** becoming unreliable against their will.

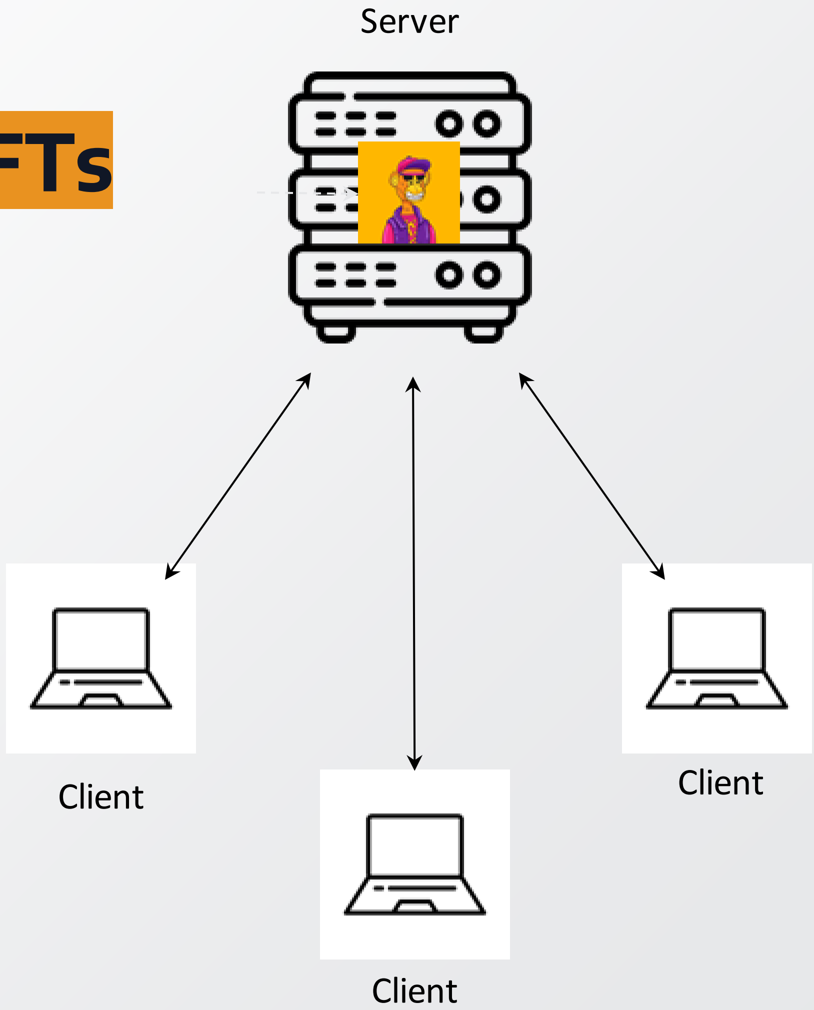
Finally: third parties can be very expensive.



# Decentralization and NFTs

Decentralization helps to avoid the **single-point-of-failure** mitigating the above risks.

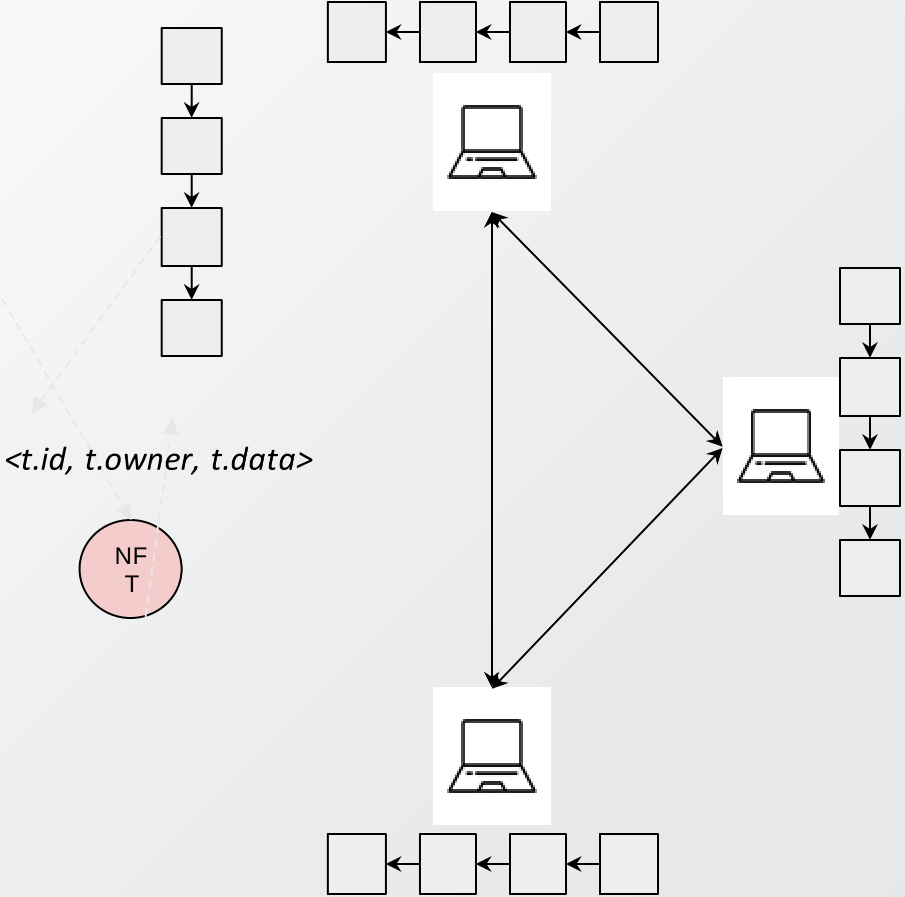
Currently there are decentralized platforms (e.g., decentralized computers like **Ethereum**, decentralized storage like **IPFS**) devoted to managing ownership of and access to digital assets: Non-Fungible Tokens (**NFTs**)



# Decentralized Management of Digital Assets

NFT collections are typically built using a standard **ERC-721** smart contract on Ethereum

The actual digital data corresponding to the asset (e.g., an high-resolution image) is stored using **decentralized storage** (e.g., IPFS) and only a **cryptographic hash** of it appears in the state of the smart contract



# Non-Fungible Tokens (NFTs)

There has been **criticism** on what being **owner** of a **digital artwork** means; this is due to the fact that **everyone** can **download digital data**, therefore enjoying it. Moreover there could be **clones**.

Are such **problems** inherent?
Are **NFTs** for artworks really **useful**?
Is the owner of an asset really protected from future copies in the same collection? And from unauthorized access to data of the asset?

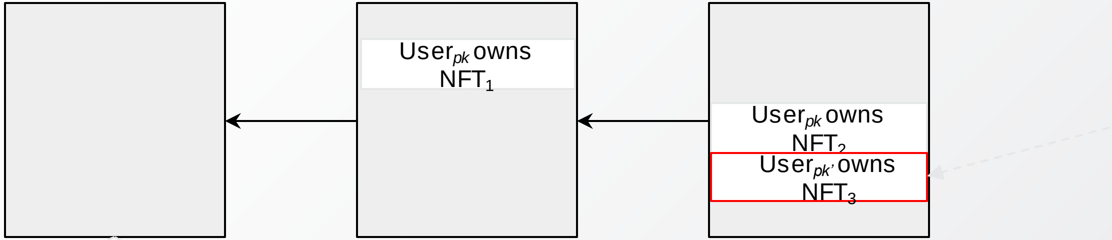
# NFTs

Let’s formalize the problems pragmatically.

In the physical world, there are many **copies** of The Great Wave, of David, of Mona Lisa; still two copies are never identical even though to the eyes of non-experts they look so.

The **owner** of an original copy **enjoys** it in full all the time and **decides** the **accuracy (e.g., the distance)** and the **price** to pay **for others.**
**Can we achieve this in the digital world?**

# NFT Collections with ERC-721: bad news



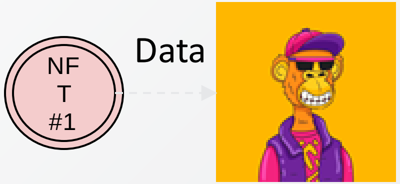
User$_{pk}$ owns NFT$_1$

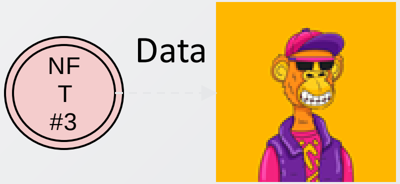
User$_{pk}$ owns NFT$_2$

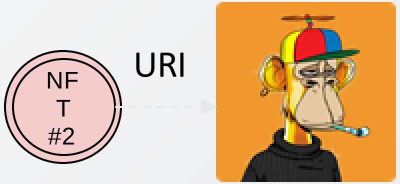
User$_{pk'}$ owns NFT$_3$

ERC-721

```
mint(tokenID, pk, [data/tokenURI]) {
  if (msg.sender is allowed)
    // create the token …
}
transfer(tokenID, pk)
approve(tokenID, pk)
approveAll(tokenID)
```



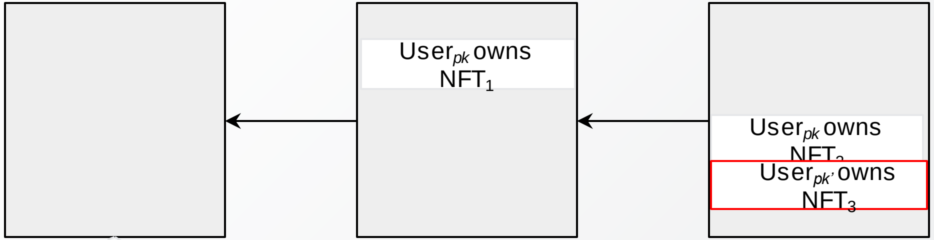




NFTs are identified by an assigned **numerical index** in the smart contract, not by the **content** of the collectible

The minter is a **single-point-of-failure**.

# NFT Collections with ERC-721: bad news



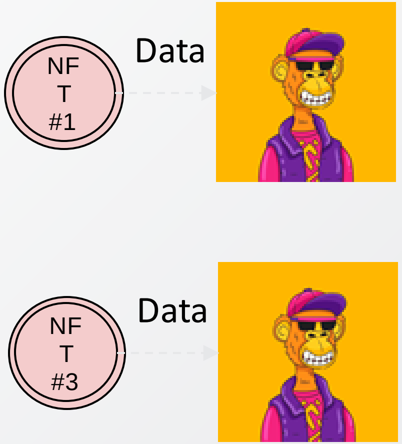
User$_{pk}$ owns NFT$_1$

User$_{pk}$ owns NFT$_2$

User$_{pk'}$ owns NFT$_3$

ERC-721

```
mint(tokenID, pk, [data/tokenURI]) {
  if (msg.sender is allowed)
    // create the token …
}
transfer(tokenID, pk)
approve(tokenID, pk)
approveAll(tokenID)
```



To verify that there are no clones buyers need to **scan the entire collection**. Still, clones could appear in the future.

What is the point of using a **decentralized platform** if there is an obvious **single point of failure**? It can fail, because of corruption or attacks.

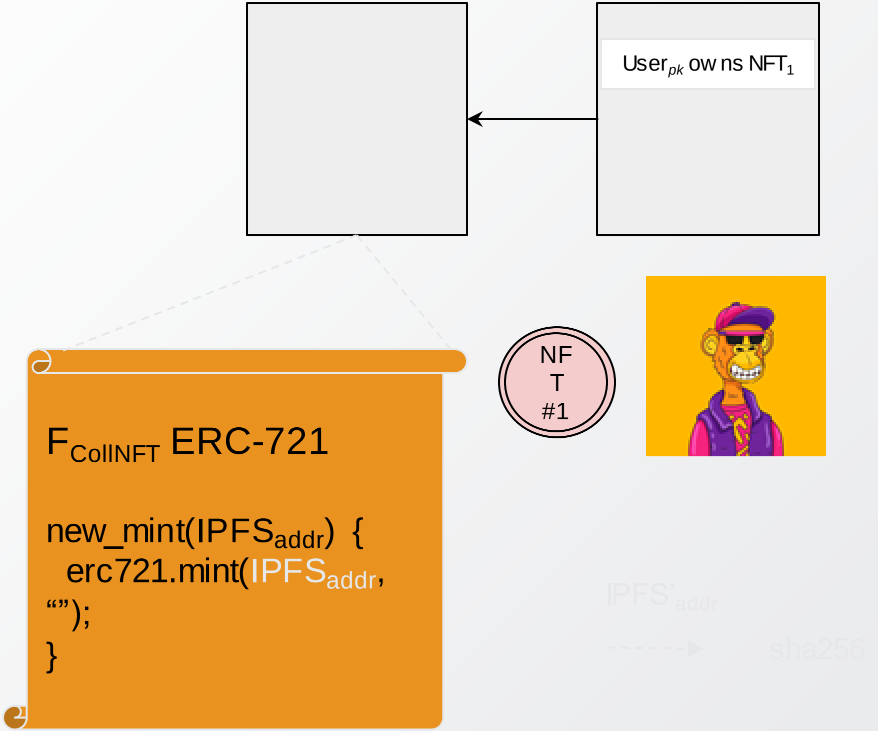
# What about the digital world?

**Q1**: How can we guarantee that a buyer of a digital artwork in a collection will not be penalized by the generation of additional **identical copies** of that artwork in the same collection?

Q2: How can we allow the owner of a digital artwork to decide **how much** of it will be **visible to others** willing to pay for it?

Q3: How can we build a system where **ownership and full access** to the digital artwork **can jointly be transferred** from seller to buyer, **while others remain excluded**?

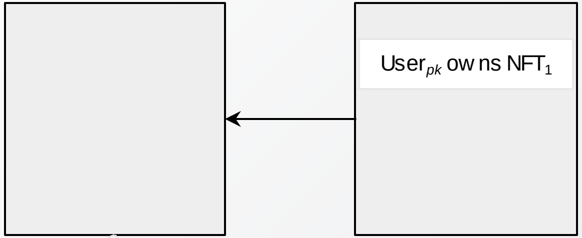
# Addressing Q1 - Trivial (but very partial) Solution



$F_{CollNFT}$ ERC-721

```
new_mint(IPFS_addr) {
  erc721.mint(IPFS_addr,
“”);
}
```

# Addressing Q1 - Trivial (but very partial) Solution



$User_{pk}$ owns $NFT_1$

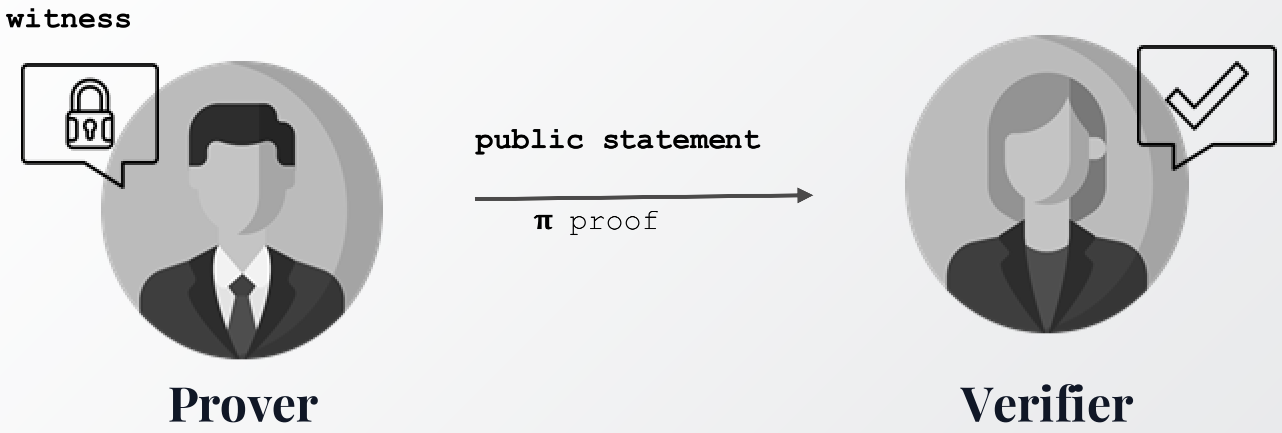
$F_{CollNFT}$ ERC-721

```
new_mint(IPFS_addr) {
  erc721.mint(IPFS_addr,
“”);
}
```

NFT #1

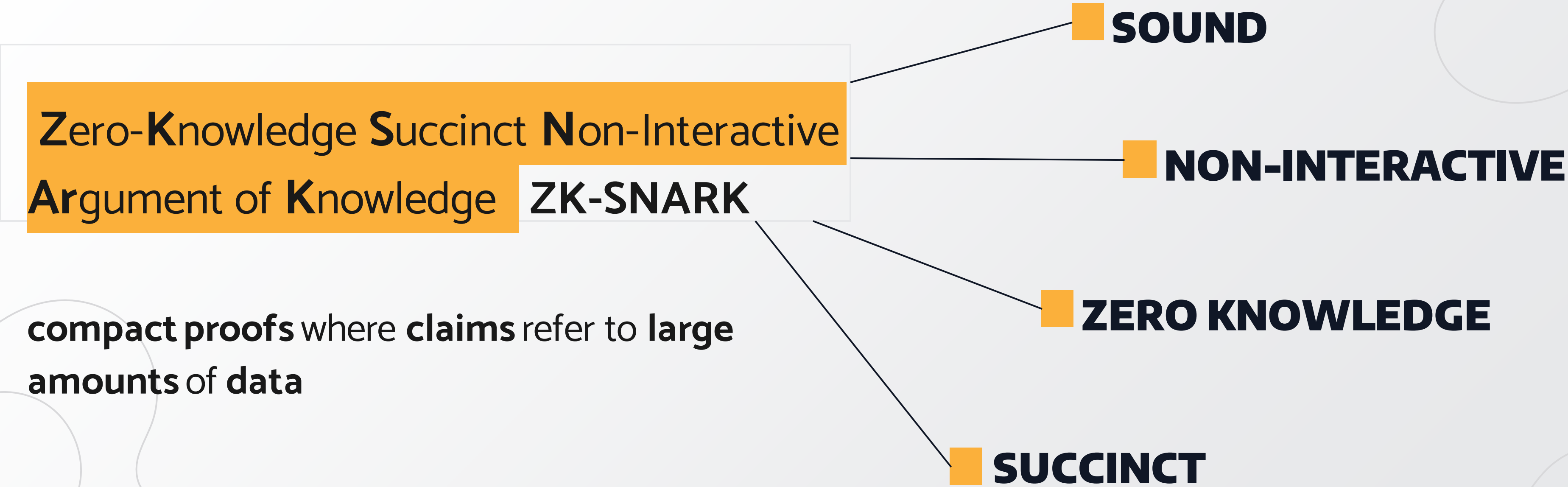
Basically, cryptographic hash is enough but, two identical images in the very same format could simply have different metadata and such data are part of the (to be hashed) file.

# Zero-Knowledge Proofs (ZK Proofs)



Cryptographic tool introduced by *Goldwasser, Micali, and Rackoff* in *1985*, allowing a **prover** to be **convince** a **verifier** about a **claim without revealing any secret information**.

# Zero-Knowledge Proofs (ZK Proofs)



Zero-Knowledge Succinct Non-Interactive Argument of Knowledge ZK-SNARK

SOUND

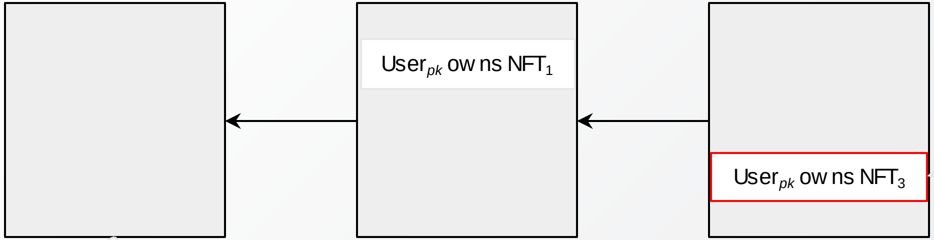
NON-INTERACTIVE

ZERO KNOWLEDGE

SUCCINCT

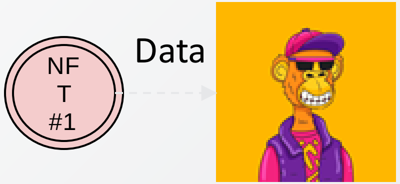
**compact proofs** where **claims** refer to **large amounts** of **data**

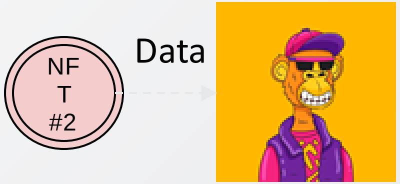
# Addressing Q1 - (Complex) Solution 2



$F_{CollNFT}$ ERC-721

```
new_mint(hash, SNARK) {
  if (SNARK is valid) {
    erc721.mint(hash, “”);
  }
}
```





The snark assesses that the picture (i.e., the sequence of pixels) encoded in the file is different from all other pictures included in the collection.

In theory it works… but obtaining a practical implementation is going to be tough…

# What about the digital world?

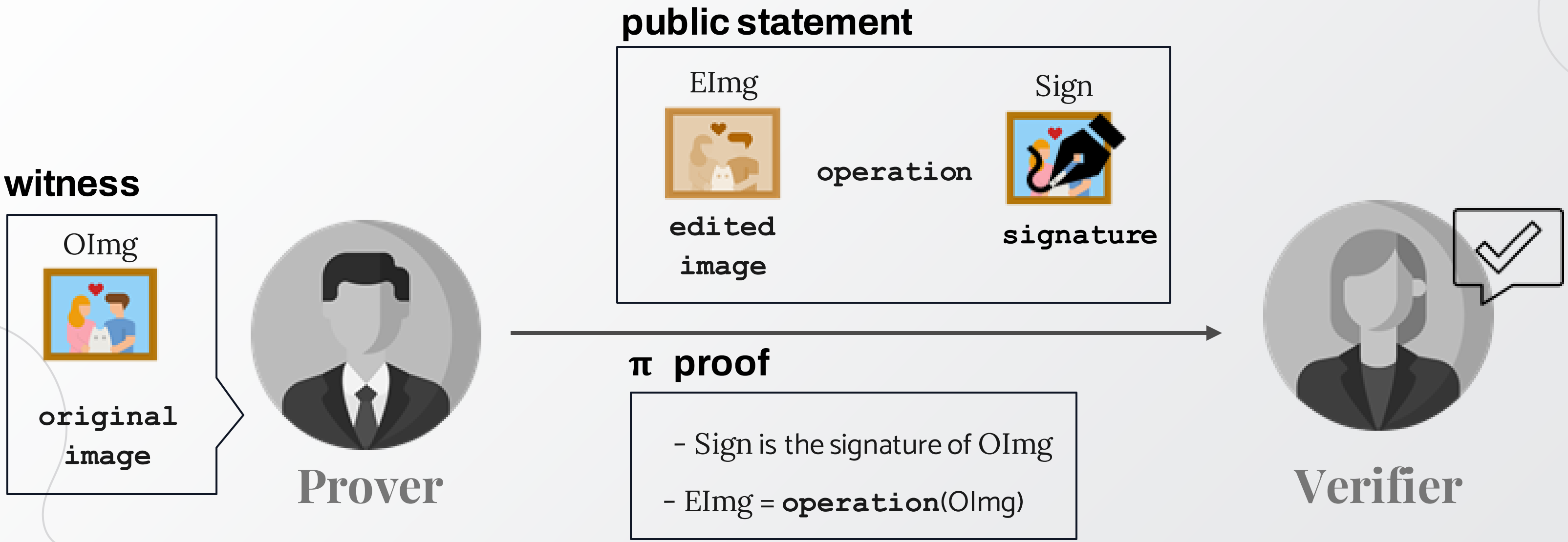
**Q1**: How can we guarantee that a buyer of a digital artwork in a collection will not be penalized by the generation of additional **identical copies** of that artwork in the same collection?

Q2: How can we allow the owner of a digital artwork to decide **how much** of it will be **visible to others** willing to pay for it?

Q3: How can we build a system where **ownership and full access** to the digital artwork **can jointly be transferred** from seller to buyer, **while others remain excluded**?

# Applications of ZK Proofs on Image Transformations

The basic idea consists of **linking** two images, an **original image** and the correspective **modified image** through a **ZK proof**.



**public statement**

EImg

edited image

operation

Sign

signature

**witness**

OImg

original image

Prover

**π proof**

- Sign is the signature of OImg
- EImg = operation(OImg)

Verifier

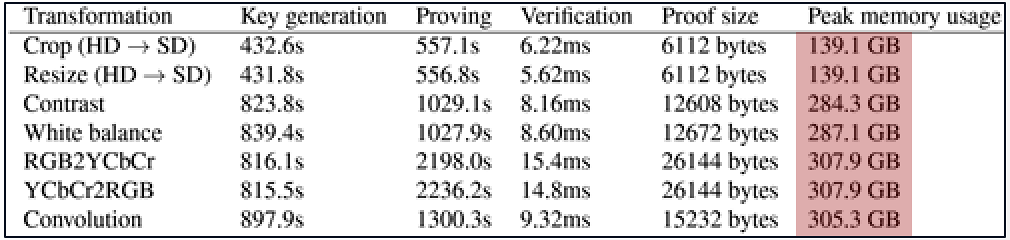
# Applications of ZK Proofs on Image Transformations

## Fingthing Misinformation

Nowadays **unreliable information** can easily be **spread** through popular media, **contributing** to so-called "**fake news**"

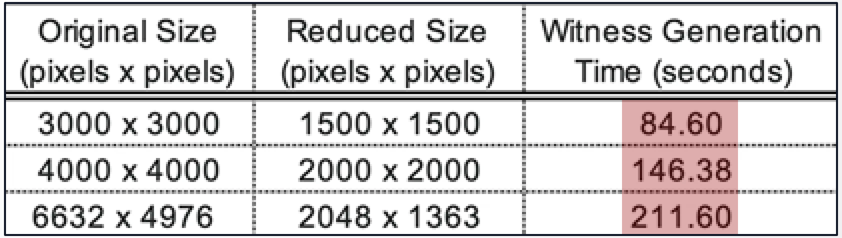
Thanks to this use of ZK proofs it is (at least in theory) possible to **ensure** the **authenticity** of received photos, guaranteeing that **only defined operations** have been performed on the original image produced by a camera capable of signing (keeping the signing key secret)

# Applications of ZK Proofs on Image Transformations



| Transformation | Key generation | Proving | Verification | Proof size | Peak memory usage |
|---|---|---|---|---|---|
| Crop (HD → SD) | 432.6s | 557.1s | 6.22ms | 6112 bytes | 139.1 GB |
| Resize (HD → SD) | 431.8s | 556.8s | 5.62ms | 6112 bytes | 139.1 GB |
| Contrast | 823.8s | 1029.1s | 8.16ms | 12608 bytes | 284.3 GB |
| White balance | 839.4s | 1027.9s | 8.60ms | 12672 bytes | 287.1 GB |
| RGB2YCbCr | 816.1s | 2198.0s | 15.4ms | 26144 bytes | 307.9 GB |
| YCbCr2RGB | 815.5s | 2236.2s | 14.8ms | 26144 bytes | 307.9 GB |
| Convolution | 897.9s | 1300.3s | 9.32ms | 15232 bytes | 305.3 GB |

*D. Kang, T. Hashimoto, I. Stoica, and Y. Sun, "**ZK-IMG: Attested Images via Zero-Knowledge Proofs to Fight Disinformation.**" - arXiv.org - 2022*



| Original Size (pixels x pixels) | Reduced Size (pixels x pixels) | Witness Generation Time (seconds) |
|---|---|---|
| 3000 x 3000 | 1500 x 1500 | 84.60 |
| 4000 x 4000 | 2000 x 2000 | 146.38 |
| 6632 x 4976 | 2048 x 1363 | 211.60 |

*T. Datta and D. Boneh. "**Using zk-proofs to fight disinformation**" - RWC - 2023*

From the **prover**'s point of view, the proof generation becomes **memory** and **time** intensive when the **images** are **HD**.

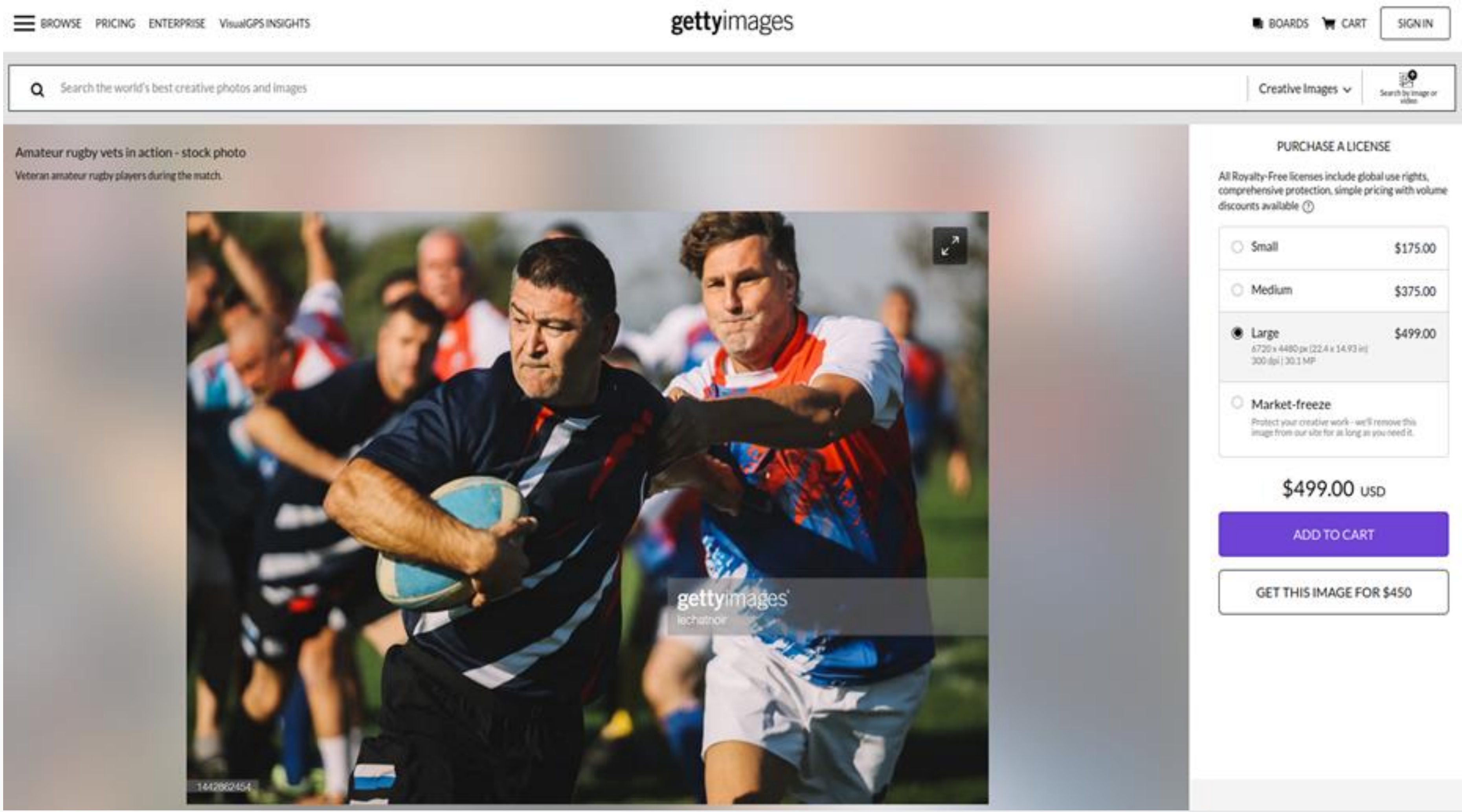
* *Note that these computations were not performed on standard personal computers but on **high-performance cloud architectures**.*

# What about the digital world?

**Q1**: How can we guarantee that a buyer of a digital artwork in a collection will not be penalized by the generation of additional **identical copies** of that artwork in the same collection?

Q2: How can we allow the owner of a digital artwork to decide **how much** of it will be **visible to others** willing to pay for it?

Q3: How can we build a system where **ownership and full access** to the digital artwork **can jointly be transferred** from seller to buyer, **while others remain excluded**?



Amateur rugby vets in action - stock photo

Veteran amateur rugby players during the match.

PURCHASE A LICENSE

All Royalty-Free licenses include global use rights, comprehensive protection, simple pricing with volume discounts available ⓘ

| | |
|---|---|
| ○ Small | $175.00 |
| ○ Medium | $375.00 |
| ● Large<br>6720 x 4480 px (22.4 x 14.93 in)<br>300 dpi \| 30.1 MP | $499.00 |
| ○ Market-freeze<br>Protect your creative work - we'll remove this image from our site for as long as you need it. | |

$499.00 USD

# Applications of ZK Proofs on Image Transformations

Fingthing Misinformation

Adult Contents

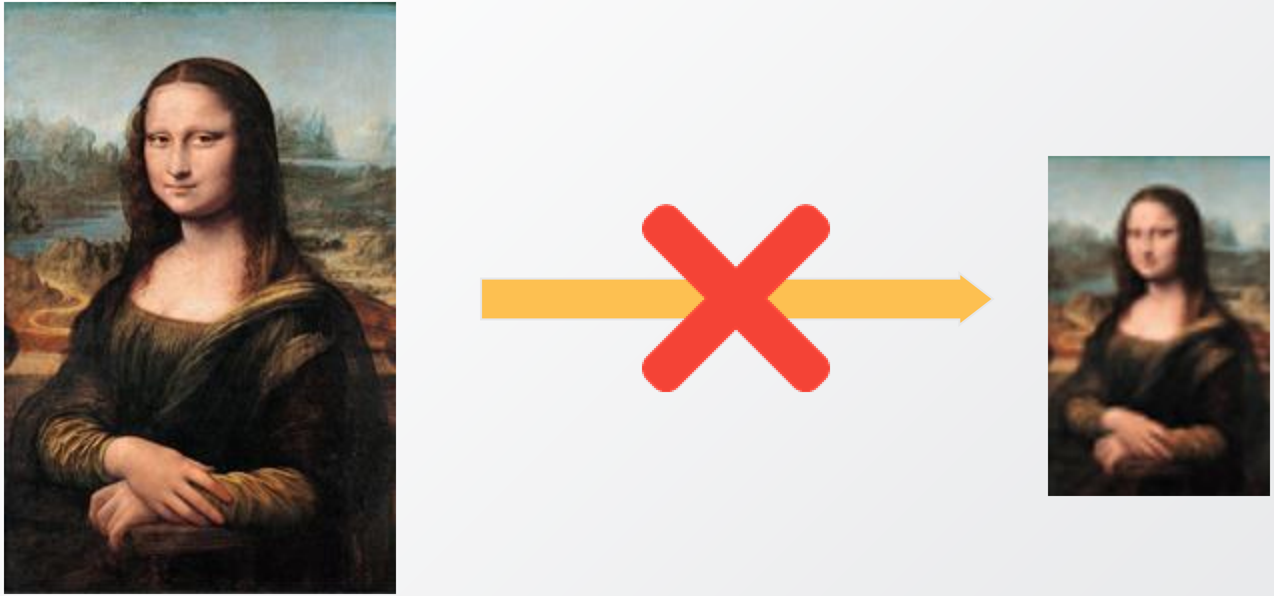
# Applications of ZK Proofs on Image Transformations

## Adult Contents

The **paid adult entertainment** industry is an evergreen. Nowadays platforms such as **OnlyFans** manage a gigantic business.

Through decentralized platforms and ZK Proofs on transformable images the **creator** could **selectively display censored content** while end-users will be saved from frauds, without third parties involved (and charging) to guarantee the reliability of the transactions.

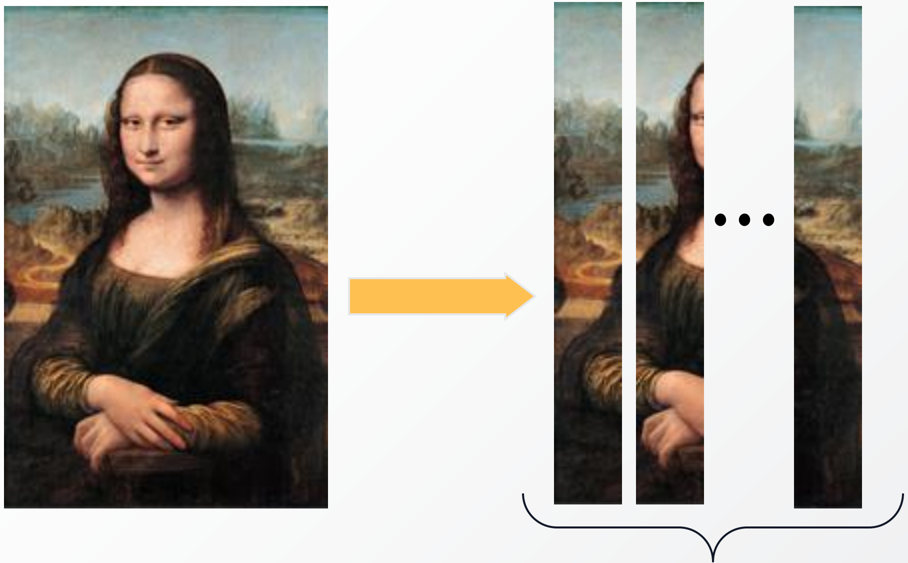
# Applications of ZK Proofs on Image Transformations



Given a large image, **FullHD** (1920x1080), a corresponding resized image **RI** and a cryptographic hash **H** of **FullHD** it is tough **(with the above techniques)** to generate a ZKP proving knowledge of a correct FullHD matching H and RI.

**Computational** and **memory** requirements are extremely large and unpractical for **common computers**.

# Our Trick: Image Tiling



*The image is too large to calculate its ZKP*

*Each tile has a reduced dimension and it is possible to split the computational effort*

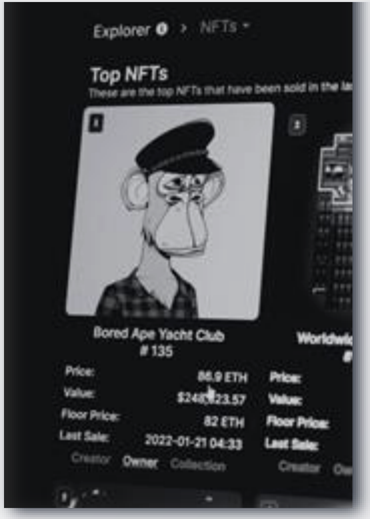
This methodology consists of **splitting** the image into many **tiles.**
**For each tile**, a **ZKP** can be defined, enabling **hashing** for a **limited set of values** and producing multiple hashes that represent different image components.

⚠ It is important that the transformation of the full images can be computed working locally tile by tile. Some “resize” operations follow this approach.

# Applications of ZK Proofs on Image Transformations

## Fingthing Misinformation

## Privacy Preserving NFTs



Explorer › NFTs

Top NFTs

These are the top NFTs that have been sold in the la

Bored Ape Yacht Club
# 135

Price: 86.9 ETH
Value: $248,[illegible]23.57
Floor Price: 82 ETH
Last Sale: 2022-01-21 04:33

Creator Owner Collection

Worldwic

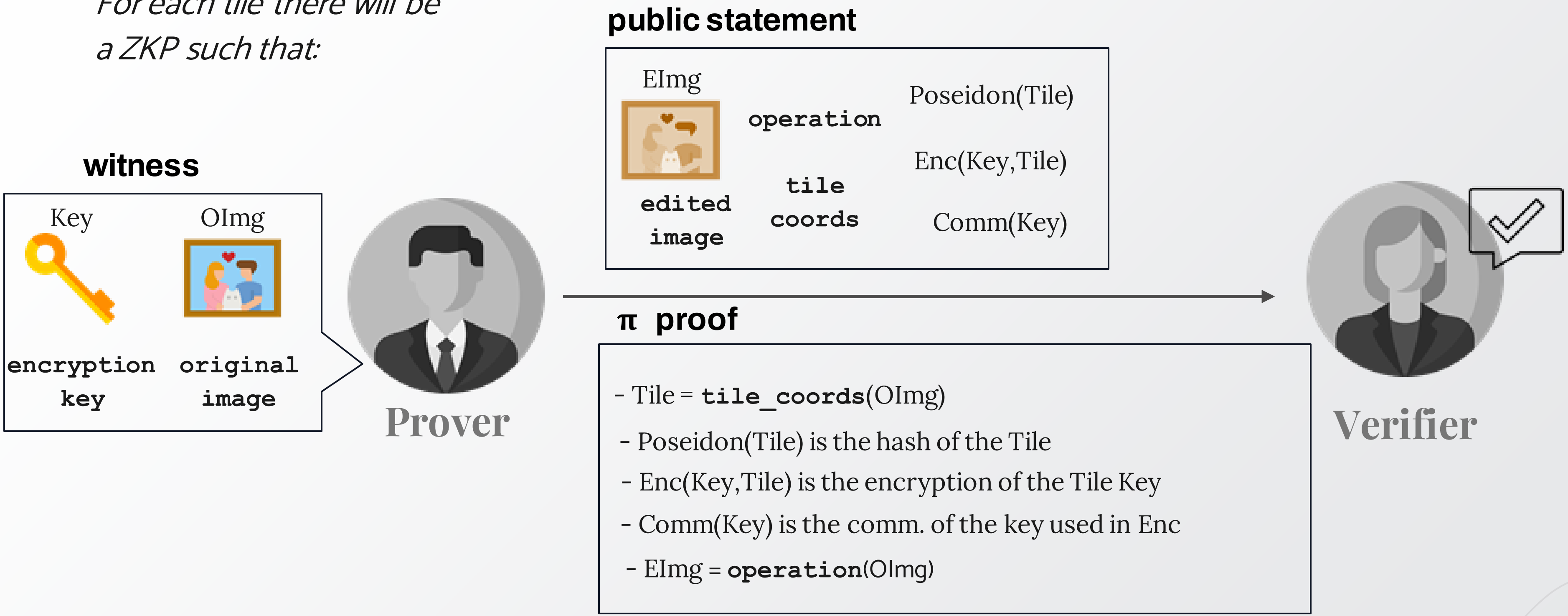
Price:
Value:
Floor Price:
Last Sale:

Creator

## Adult Contents

# Statement for ***NFTs*** protocol

*For each tile there will be a ZKP such that:*



**public statement**

EImg

**`edited image`**

**`operation`**

**`tile coords`**

Poseidon(Tile)

Enc(Key,Tile)

Comm(Key)

**witness**

Key

**`encryption key`**

OImg

**`original image`**

Prover

Verifier

**π proof**

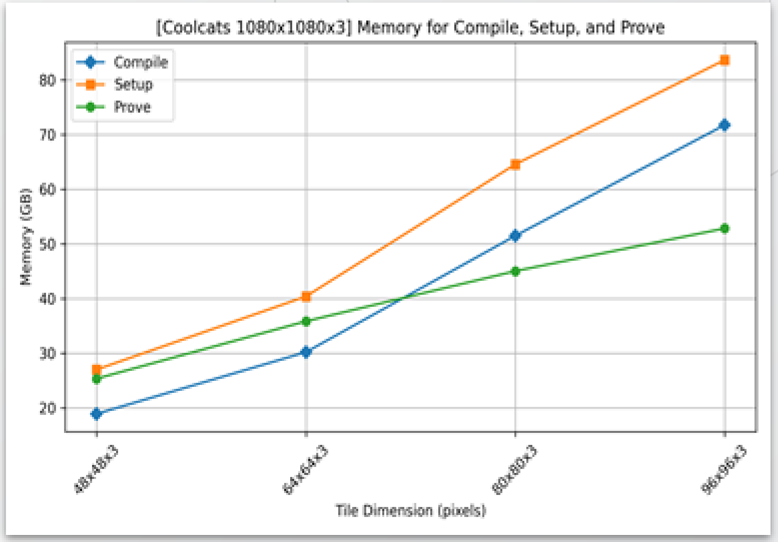
- Tile = **`tile_coords`**(OImg)
- Poseidon(Tile) is the hash of the Tile
- Enc(Key,Tile) is the encryption of the Tile Key
- Comm(Key) is the comm. of the key used in Enc
- EImg = **`operation`**(OImg)

* *Note that the* **`operation`** *used in the evaluation phase that we selected to produce the* **`edited image`** *was a complex resize operation, in particular* ***Bilinear Image Resize*** *algorithm; this one does now work operating locally tile by tile!*

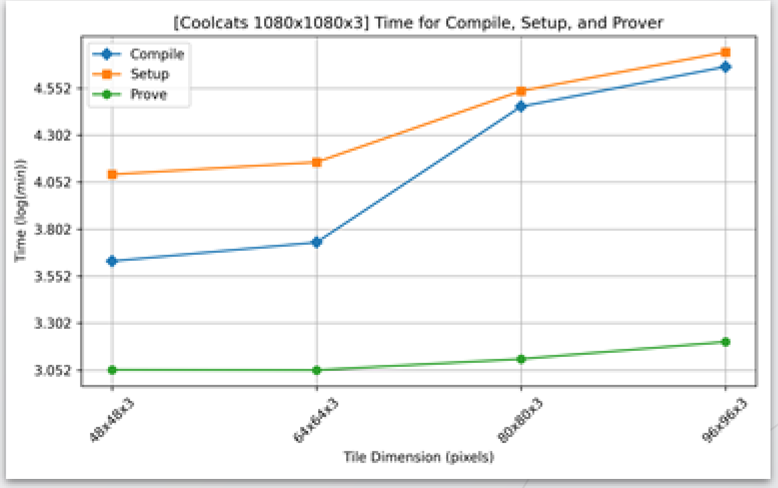
# Our Trick: Image Tiling

**Experiments** were conducted on an **image** (**1080x1080 pixels**) from the **CoolCats** collection.

* *Note that **Compute** and **Setup** operations must be performed **only once** for each fixed **dimension**.*



[Coolcats 1080x1080x3] Memory for Compile, Setup, and Prove



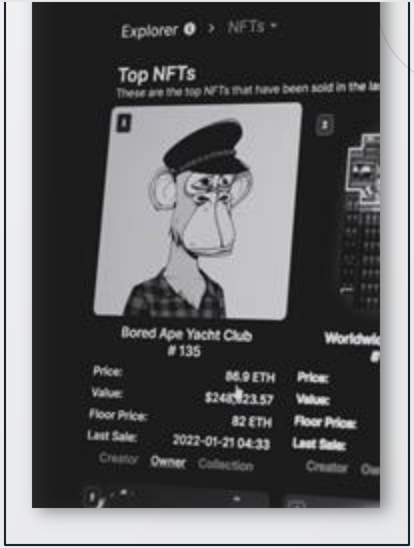
[Coolcats 1080x1080x3] Time for Compile, Setup, and Prover

# Applications of ZK Proofs on Image Transformations

**ERC-721** smart contracts commonly give **all data** describing the **digital asset** associated to a token

Through ZK Proofs on image transformations we can have a **low-resolution** version of the **asset** that is **shown publicly** still leaving the asset **appealing** to potential **buyers**, while **ensuring** that the **original asset** remains **accessible in full only** to the **owner**.

## Privacy Preserving NFTs



Explorer › NFTs

Top NFTs

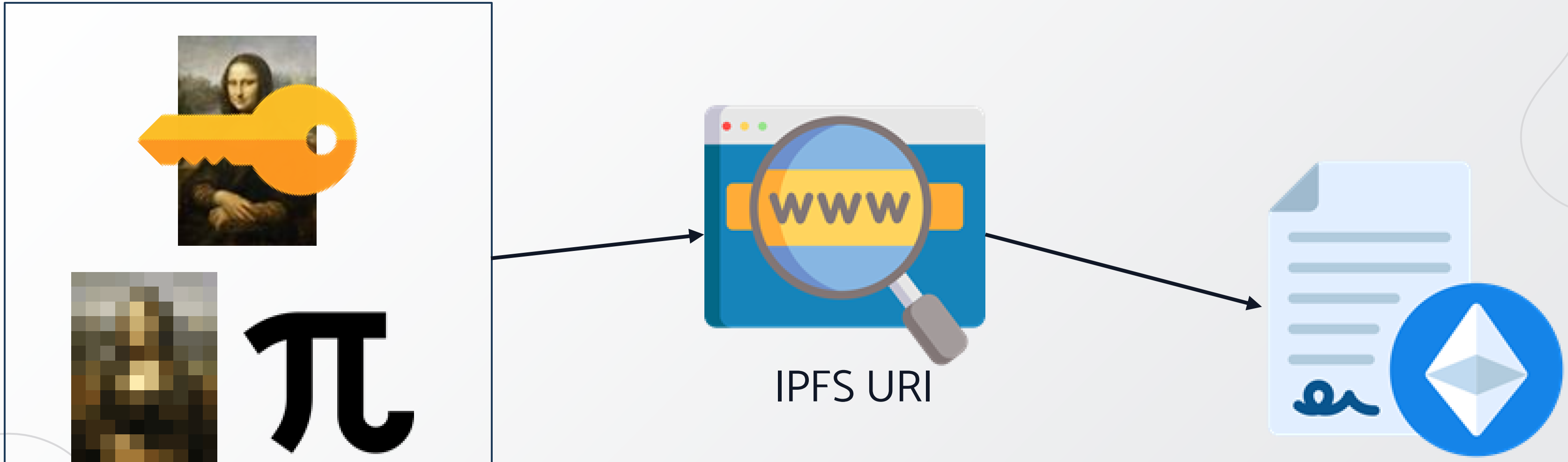
These are the top NFTs that have been sold in the la

Bored Ape Yacht Club
# 135

Price: 86.9 ETH
Value: $248,[illegible]23.57
Floor Price: 82 ETH
Last Sale: 2022-01-21 04:33

Creator Owner Collection

# Applications of ZK Proofs on Image Transformations



Proof, Low-Res Image and Encrypted Full Image

IPFS URI

Smart Contract
for the asset management

# Smart Contract Logic

## 1ST

### Expensive Solution

The buyer deposits the money. The seller could transfer the encryption key to the buyer through the state of the smart contract, proving with a snark that this was done correctly. The **smart contract** verifies the snark and transfers the money.
It requires smart contract supporting snarks and transactions can be expensive.

## 2ND

### Cheap Solution ✓

In this case, the transfer optimistically does not require any work on the side of the smart contract. If the buyer complains for having received a bad key, the smart contract will perform only a few simple computations.

# Smart Contract Logic

## Seller

Mint a **PP-NFT** with:

I. Its associated **URI** (containing the **ZKP** and the **low-quality image**, and an encryption of the image)

II. The **encryption key commitment** used to encrypt the image $comm_{key}$

III. The **public key** of the seller $Pk_s$

## Buyer

Compute his own **secret key** $sk_b$

Place a **bid** on the **PP-NFT** and also deposit its **public key**. $Pk_b$

Bid is **stored** on the **smart contract**, seller can view it

# Smart Contract Logic

## Seller

**If Seller accept the bid:**

I. Compute **Diffie-Hellman key**.
$K_{dh} = Pk_b^{sk_s}$

II. Compute the **shared key**.
$K_{sb} = Ro(K_{dh})$

**III. Encrypt** the **key** that can decrypt the image.
$enckey = Enc(K_{sb}, key)$

**IV. Store** the **encrypted key** into the **smart contract**

**else:** trade ends

## Buyer

I. Compute **Diffie-Hellman key**.
$K_{dh} = Pk_s^{sk_b}$

II. Compute the **shared key**.
$K_{sb} = Ro(K_{dh})$

**Decrypt** the **key.**
$key = Dec(K_{sb}, enckey)$

# Smart Contract Logic

## Smart Contract

## Buyer

**If** `key` **can decrypt the image:**

The **protocol ends**, the **buyer becomes** the new **owner**, and the **seller receives** the **funds**.

**else:**

Complaining by **sending** his **secret key** $sk_b$ to the **smart contract**.

I. Compute **Diffie-Hellman key**.

$K_{dh} = Pk_s^{sk_b}$

II. Compute the **shared key**.

$K_{sb} = Ro(K_{dh})$

**III. Decrypt** the **key.**

$key = Dec(K_{sb}, enckey)$

**IV. Check** .

$comm_{key} == comm(key)$

`if so,` **buyer becomes** the new **owner**, and the **seller receives** the **funds**

otherwise, the **NFT** is **burned** and the **bid** is **refunded** to the **buyer**

# The Power of Enhanced NFTs

Q1: How can we guarantee that a buyer of a digital artwork in a collection will not be penalized by the generation of additional identical copies of that artwork in the same collection?

Q2: How can we allow the owner of a digital artwork to decide how much of it will be visible to others willing to pay for it?

Q3: How can we build a system where ownership and full access to the digital artwork can jointly be transferred from seller to buyer, while others remain excluded?

# Conclusions

- There has been a lot of SCAM around NFTs for artworks. The huge criticism against them makes somewhat sense when considering the current way assets are managed on decentralized platforms through commonly used ERC-721 smart contracts.
- However, decentralized platforms are still in their infancy and current implementations of NFTs are still very crude.
- This talk presented techniques to mitigate the popular weaknesses of NFT for artworks due to clonability and non-authorized access to resources, paving the way to a more appealing use of NFTs.
- ZK snarks are a very powerful tool to design DAPPs allowing simultaneously transparency and robustness. They are already very beneficial in many applications and we have shown how to leverage ZK snarks for enhanced NFTs.
- Our techniques work only for certain transformations and do not guarantee the buyer of an NFT or of NFT-related services more than what is expressed by the claim proved by the ZK snarks.

# THANKS !