

The Starling Framework

+ Introduction to Cryptographic Primitives

April 4, 2025 // EE292J

Benedict Lau

CTO



Outline

1. Cryptography building blocks

- Cryptographic hashes
- Digital signatures



Outline

2. Relevant concepts

- Integrity data vs. asset data
- Data registration
 - Immutable public ledgers
 - Merkle proofs
 - Selective disclosure
- Decentralized preservation
 - Content addressing
- Verifiable computing
 - Consensus on blockchains
 - ZK SNARK proofs



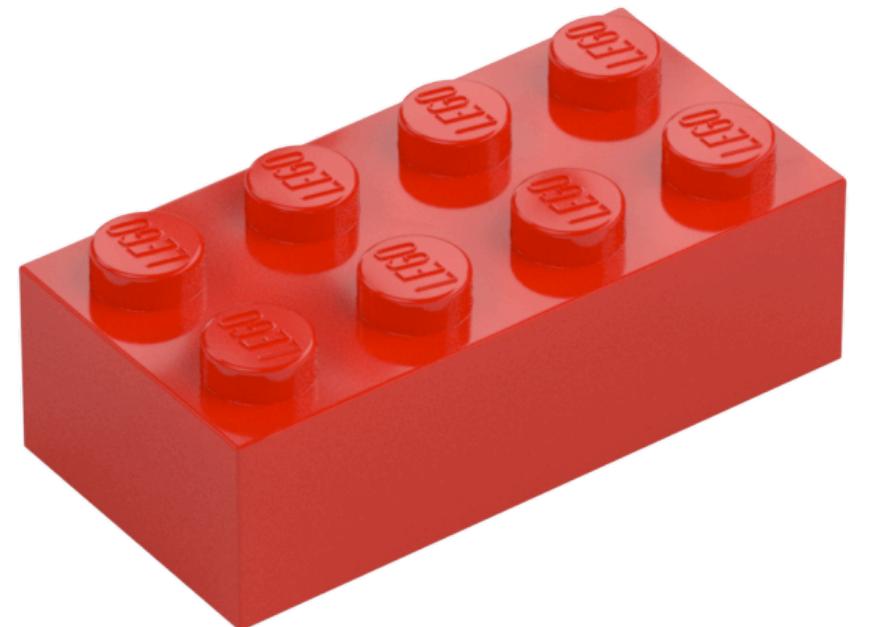
Outline

3. Building towards authenticity

- Starling pipeline: Capture · Store · Verify
- Authenticated attributes



1. Cryptography building blocks



Cryptographic hashes

digests

fingerprints

Input

Fox

cryptographic
hash
function

Digest

DFCD 3454 BBEA 788A 751A
696C 24D9 7009 CA99 2D17

The red fox
jumps over
the blue dog

cryptographic
hash
function

0086 46BB FB7D CBE2 823C
ACC7 6CD1 90B1 EE6E 3ABC

The red fox
jumps ouer
the blue dog

cryptographic
hash
function

8FD8 7558 7851 4F32 D1C6
76B1 79A9 0DA4 AEFE 4819

The red fox
jumps oevr
the blue dog

cryptographic
hash
function

FCD3 7FDB 5AF2 C6FF 915F
D401 C0A9 7D9A 46AF FB45

The red fox
jumps oer
the blue dog

cryptographic
hash
function

8ACA D682 D588 4C75 4BF4
1799 7D88 BCF8 92B9 6A6C

```
● ● ●  ~%1 ~/Dev  
~/Dev  
› echo example | md5  
ddce269a1e3d054cae349621c198dd52  
  
~/Dev  
› echo example2 | md5  
17f0f4ba8a5f1213faca591b58ba52a7  
  
~/Dev  
› echo example | md5  
ddce269a1e3d054cae349621c198dd52  
  
~/Dev  
› 
```

● ● ● 1

~/Dev

~/Dev

› echo example | md5

ddce269a1e3d054cae349621c198dd52

~/Dev

› echo example | openssl sha256

13550350a8681c84c861aac2e5b440161c2b33a3e4f302ac680ca5b686de48de

~/Dev

› echo example | openssl sha512

952de772210118f043a4e2225da5f5943609c653a6736940e0fad4e9c7cd3cfdd

348abebbf28af7b4438c55515e5a351b87cc60c808673f4d23cf12237debf41

~/Dev

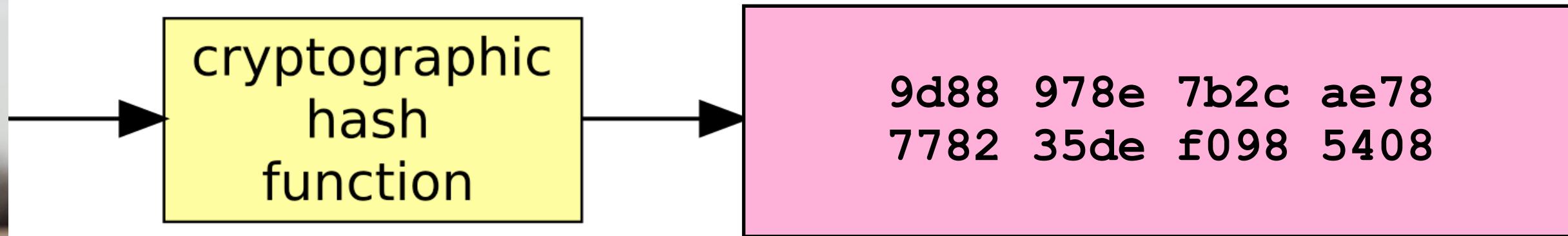
› █

	Digest size	Number of possible digests
MD5	128 bits (16 bytes = 32 hex characters)	2^{128}
SHA-256	256 bits (32 bytes = 64 hex characters)	2^{256}
SHA-512	512 bits (64 bytes = 128 hex characters)	2^{512}

Input



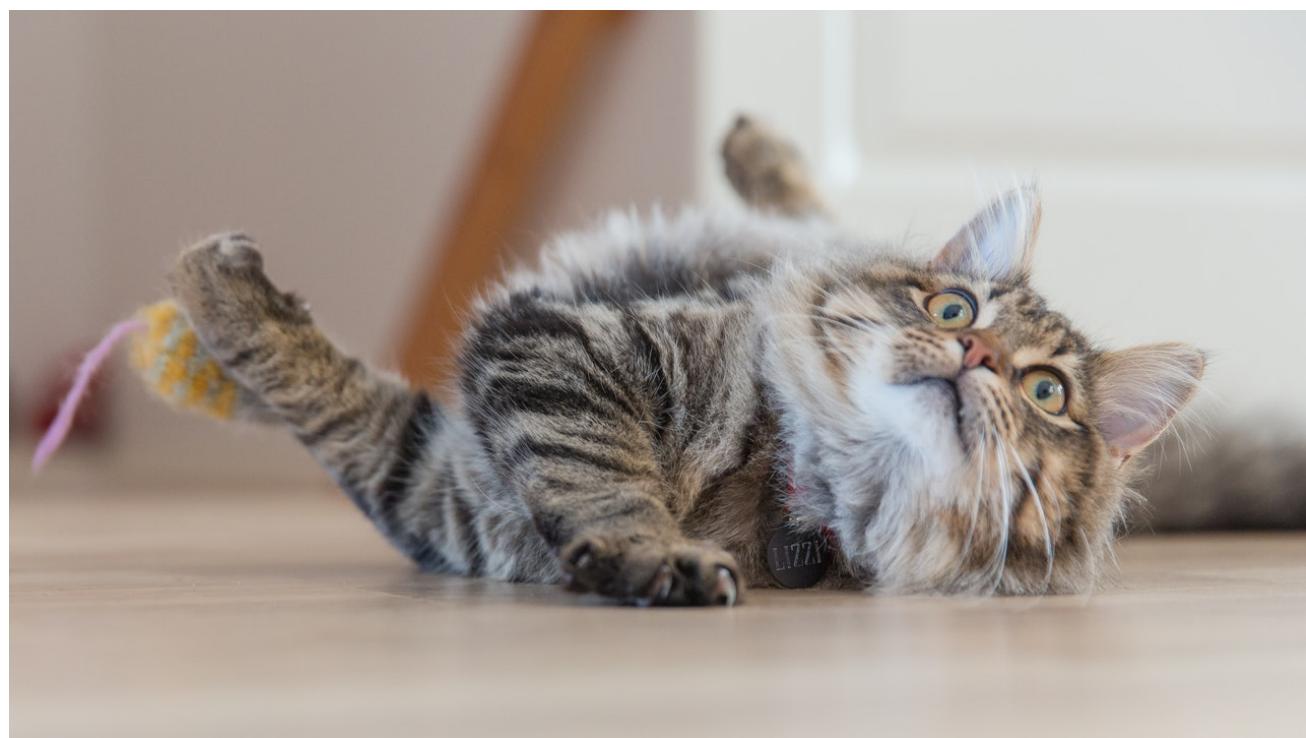
Digest



```
./Documents/cat.jpg
MD5 (/Users/benedict/Documents/cat.jpg) = 9d88978e7b2cae78778235def098540
8


```

Input



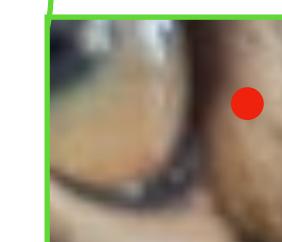
Digest

cryptographic
hash
function

9d88 978e 7b2c ae78
7782 35de f098 5408

cryptographic
hash
function

5e5b 948a de9c 3d41
de66 17b6 8f76 9e55



Input

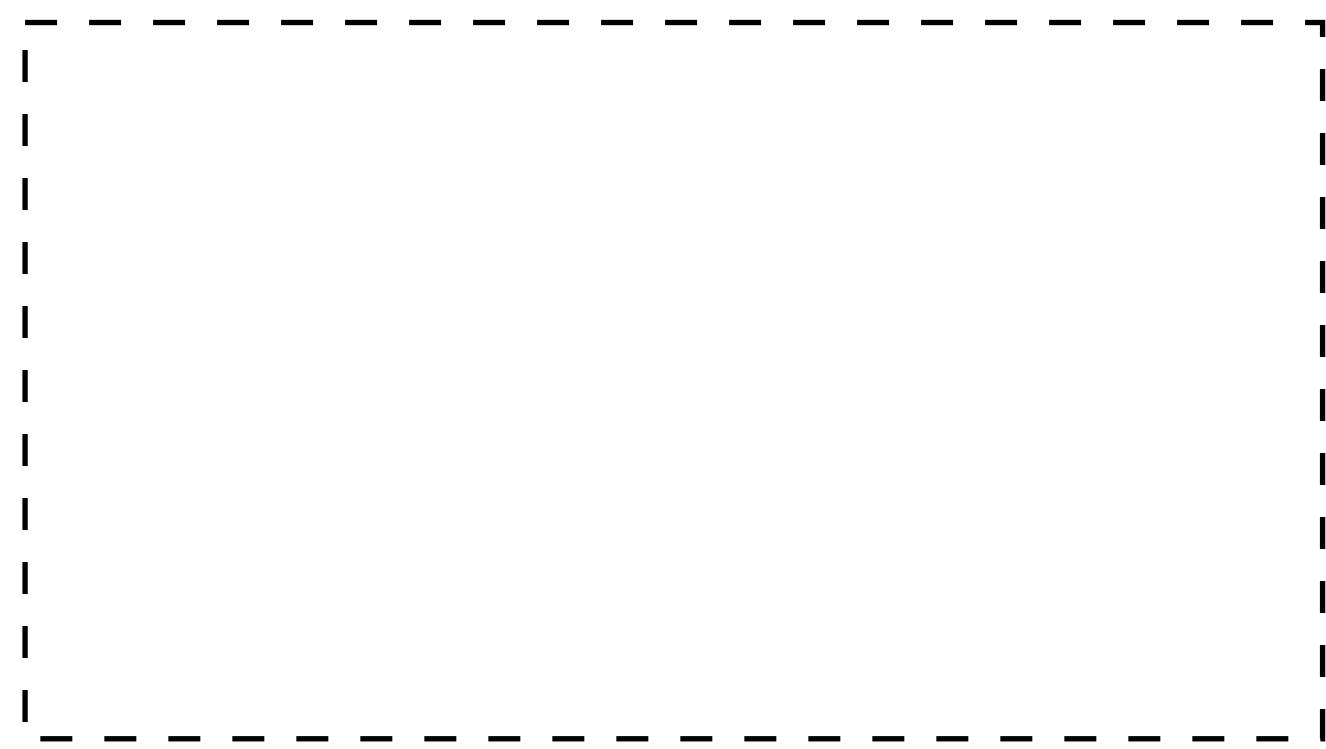


Digest



```
9d88 978e 7b2c ae78  
7782 35de f098 5408
```

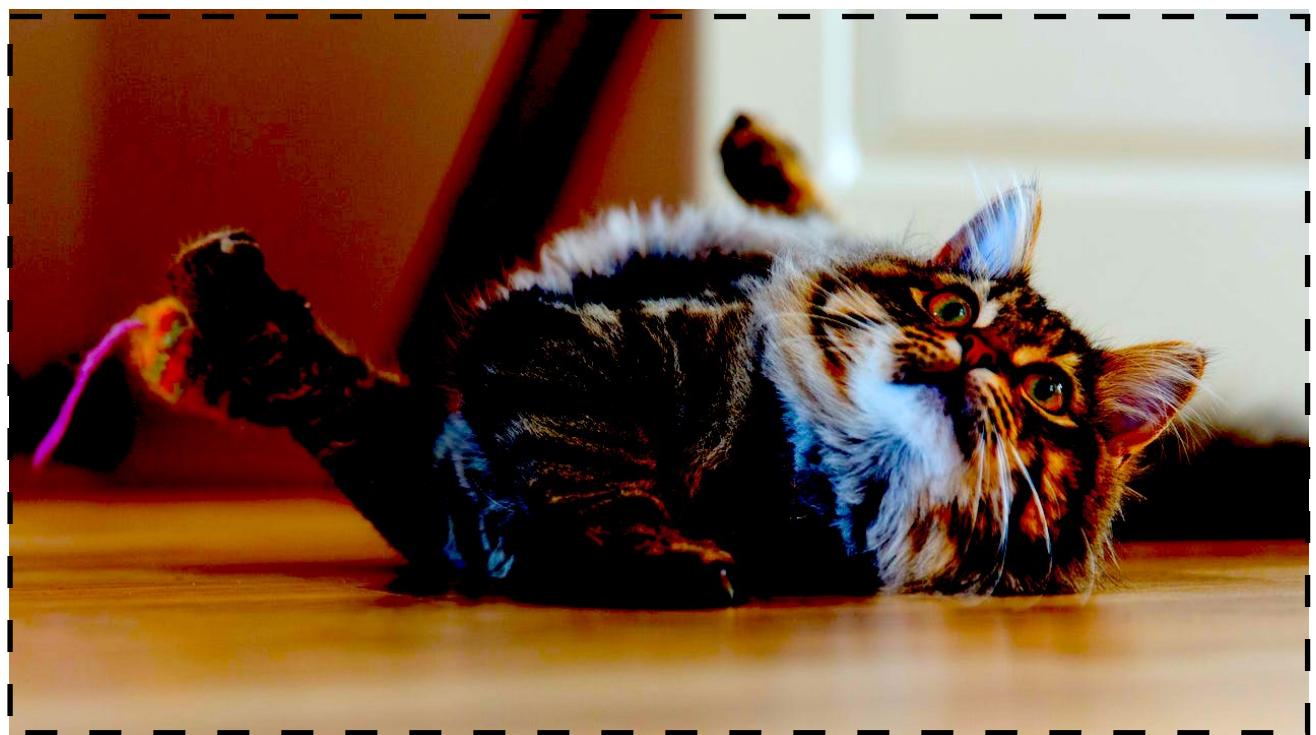
Input



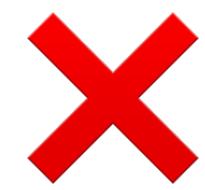
Digest

```
9d88 978e 7b2c ae78  
7782 35de f098 5408
```

Input



```
5e5b 948a de9c 3d41  
de66 17b6 8f76 9e55
```



Digest

```
9d88 978e 7b2c ae78  
7782 35de f098 5408
```

Input



```
9d88 978e 7b2c ae78  
7782 35de f098 5408
```



Digest

```
9d88 978e 7b2c ae78  
7782 35de f098 5408
```

Tamper-evident integrity

```
9d88 978e 7b2c ae78  
7782 35de f098 5408
```

**Can two different inputs have the same
digest?**

Birthday problem

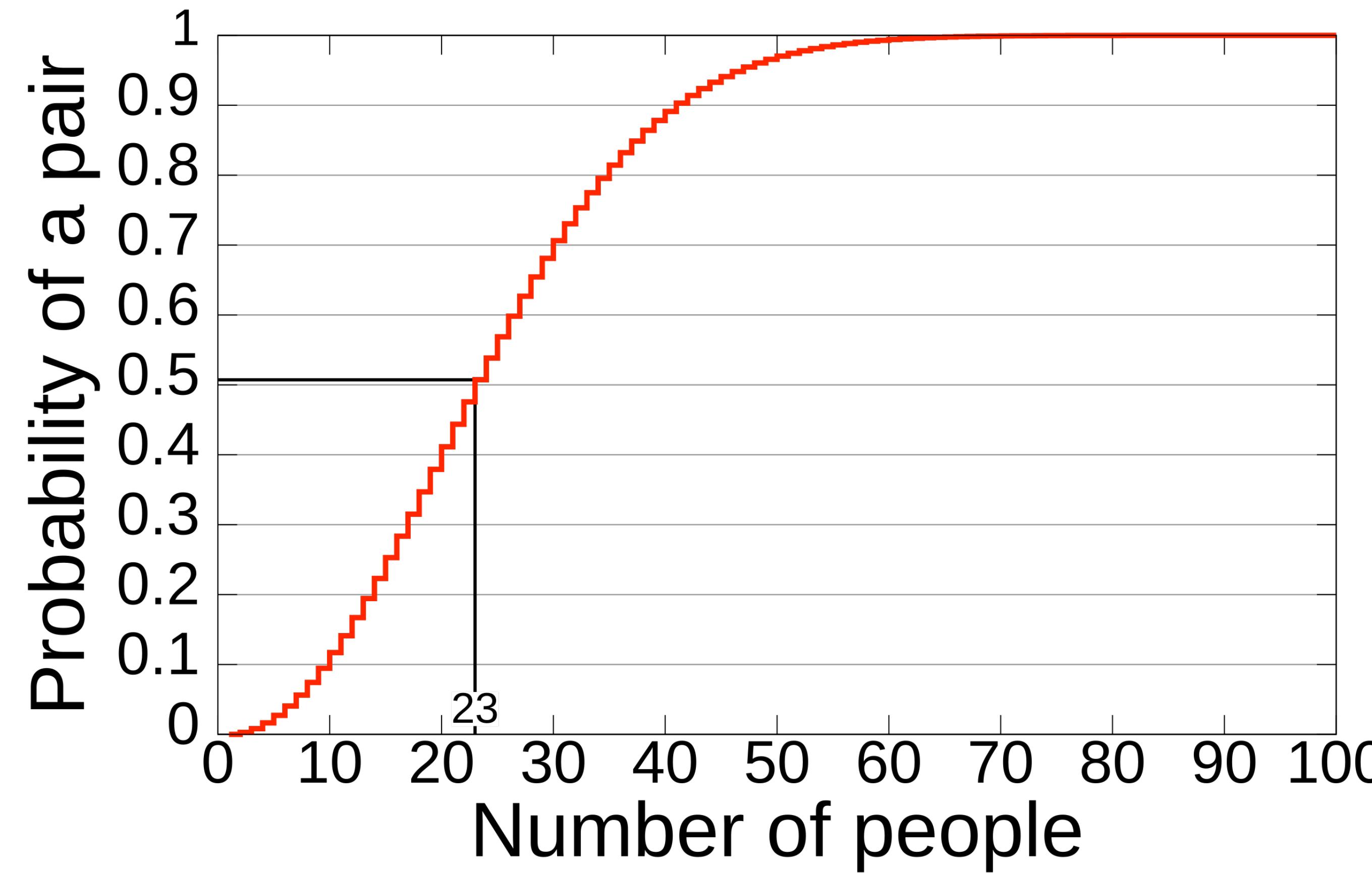
Joey
Rick
Leo
Jack
Max
Sara
Mandy
Geoff
Mike
Nick



January 1
January 2
January 3
:
December 31

365 possible days

Birthday problem



	Digest size	Number of possible digests	Number of digests for probable collision
Birthday problem		$2^{8.5}$ (365 days)	$2^{4.5}$ (23 days)
MD5	128 bits (16 bytes = 32 hex characters)	2^{128}	2^{64}
SHA-256	256 bits (32 bytes = 64 hex characters)	2^{256}	2^{128}
SHA-512	512 bits (64 bytes = 128 hex characters)	2^{512}	2^{256}

It will take a *very, very, very, very*^{very} long time to find one. For comparison, as of January 2015, Bitcoin was computing [300 quadrillion SHA-256 hashes per second](#). That's 300×10^{15} hashes per second.

Let's say you were trying to perform a [collision attack](#) and would "only" need to calculate 2^{128} hashes. At the rate Bitcoin is going, it would take them

$2^{128}/(300 \times 10^{15} \cdot 86400 \cdot 365.25) \approx 3.6 \times 10^{13}$ years. In comparison, our universe is only about 13.7×10^9 years old. Brute-force guessing is not a practical option.

—<https://crypto.stackexchange.com/a/47810>

This
number has
grown ~3,000x
since 2015

It will take a *very, very, very, very*^{very} long time to find one. For comparison, as of January 2015, Bitcoin was computing [300 quadrillion SHA-256 hashes per second](#). That's 300×10^{15} hashes per second.

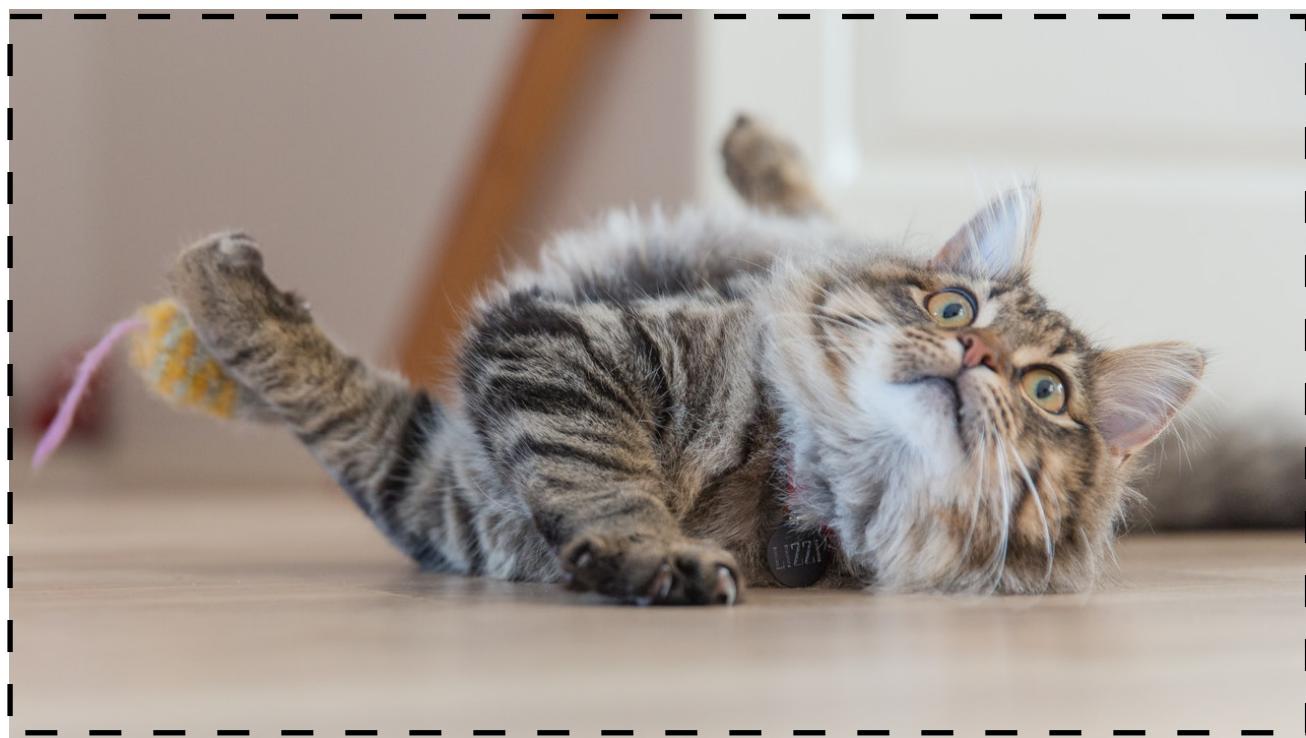
Let's say you were trying to perform a [collision attack](#) and would "only" need to calculate 2^{128} hashes. At the rate Bitcoin is going, it would take them

$2^{128}/(300 \times 10^{15} \cdot 86400 \cdot 365.25) \approx 3.6 \times 10^{13}$ years. In comparison, our universe is only about 13.7×10^9 years old. Brute-force guessing is not a practical option.

—<https://crypto.stackexchange.com/a/47810>

Can I find the input if I only have the digest?

Input



Digest

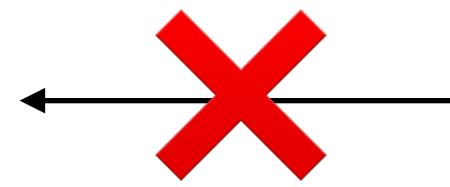


```
9d88 978e 7b2c ae78  
7782 35de f098 5408
```



This is called the **pre-image** of the digest

pre-image



9d88 978e 7b2c ae78
7782 35de f098 5408

Privacy-preserving* identifier

```
9d88 978e 7b2c ae78  
7782 35de f098 5408
```

**Disclaimer: One may gain insight about a pre-image by correlating two pieces of metadata describing the same digest*

Perceptual hashing is the use of a fingerprinting algorithm that produces a snippet, hash, or fingerprint of various forms of multimedia. A perceptual hash is a type of locality-sensitive hash, which is analogous if features of the multimedia are similar. **This is in contrast to cryptographic hashing, which relies on the avalanche effect of a small change in input value creating a drastic change in output value.** Perceptual hash functions are widely used in finding cases of online copyright infringement as well as in digital forensics because of the ability to have a correlation between hashes so similar data can be found (for instance with a differing watermark).

—Wikipedia

Digital signatures

Establishes **verifiable attribution**

Based on **public-private key cryptography**
e.g. RSA (*Rivest-Shamir-Adleman*)

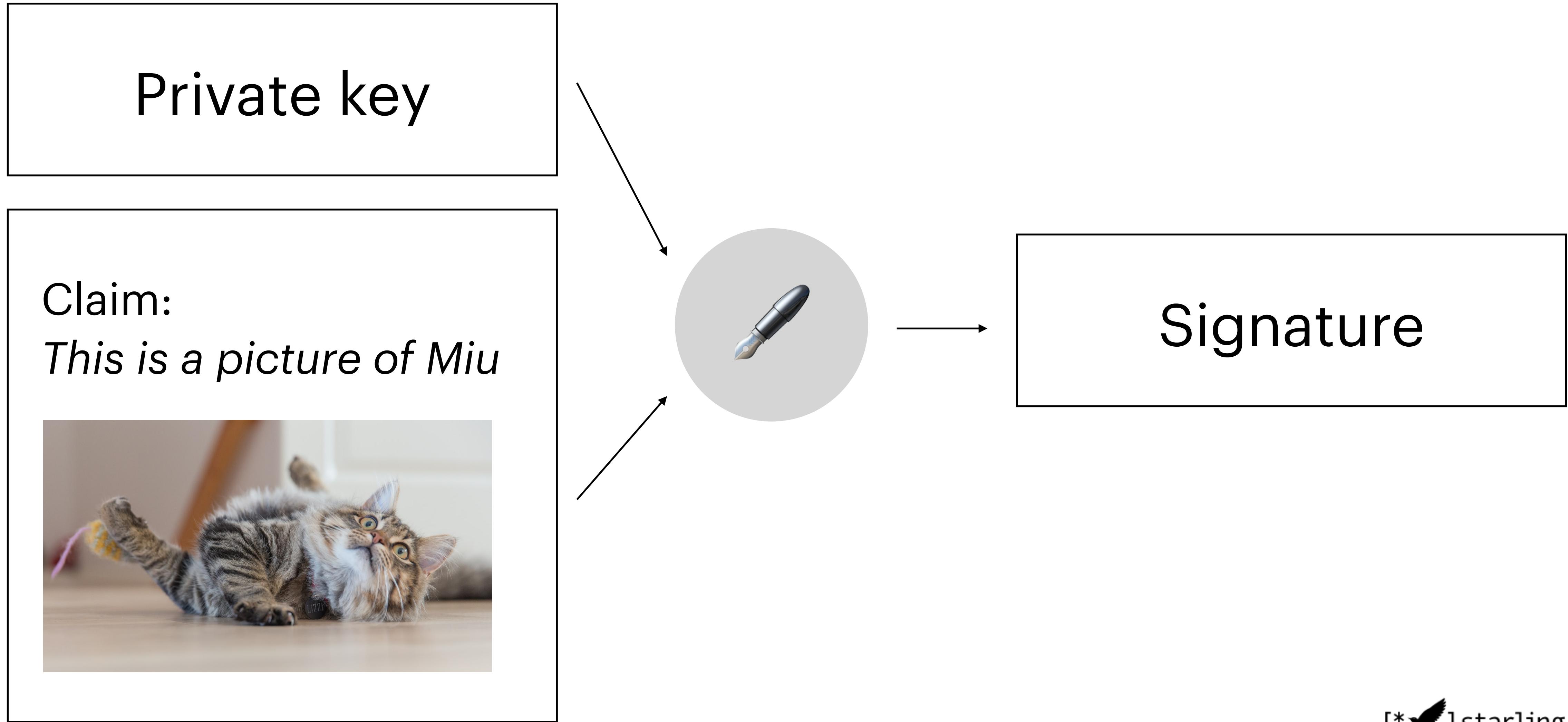
RSA private key

- Keep secret !!
- Use for signing claims

```
● ● ●  ~#1 ~/Dev
> openssl genrsa
Generating RSA private key, 2048 bit long modulus
.....+++
.....+++
e is 65537 (0x10001)
-----BEGIN RSA PRIVATE KEY-----
MIIEpQIBAAKCAQEAlIyj6wcUc/LwPbjmWZGmbXZeR3GLMw4XQhRckxONwiDab057
u5Bn78jW+N7cHFNaNhs4plAvD6Ar5hQcvgDuPIl7Kuz0YRErrc0Ytan1BqVwxtSa
EEMTKjfJC0mc1dhTLTciJeIFNLquYz802o3pYU5DbDGeheAJewPfCJ4UUNziZuiC
2+vgo0M4EIbqUYdZaqQdi26Wj9LBxZHNp04aTA6f6nX0r0A4cLP3quNFRZuwaasG
DC3q59FTIPqRUPRBvnTK+8kiEU8Ts023bmxcGjcxImFYqNQv9d7PcUCm0mHQsyC5
3ReLcvUxSGyNLJQr40kokqe7MqlwPbAN52QSWwIDAQABAoIBAQCS+oTFav/lqZ3i
tCvWu7H75H5sGgAsx+hjTVo7e8FrLq5yKyIp7/um2QU6w0bwb9h2aICorfWidEx7
HbNCnBEcfLsNQd6anC6ngzx5omv543+ue3TkWjmAMCoPrQos3BmQR1jW0sgD4CHj
uQpGpNshmBpF1uK0e51lyfLHyAE575F6eW8X5rJCsg0+x1BFctzP+P/+h19NPU/
+ju+JZ17ALtM93d2P01RG8r5Ljfx2PJRXEIgKpLjPk5Rpur7M+abcgqtK7RvDIH+
iombpyPRmWhKrcbK2b+/VqeivbX+i0qodWSVY7+ttJnouKgE63Ax9406u4EjHva1
dUIYWMPHAoGBAMRc4IXMlMjn9SvugLU3aXHMvY6AVVL6+zWVcq2Tqz6vMYcnVkiG
ZnI02/6nuEec0F6/XA0RrzPQZ/sXIp2fQ0gVQh2BMj+VKLOFrUleYtGZgT0Mx86i
FtgAFYrjP3ADH8i0bybnKL2Ex/izCb4nC8jhqg0g94rl2fjsIlBvYf23AoGBAMgq
Skas/3i3VQ+mvu1EuvahcMlxFteRjDILhLLp0ulheluN+TvTfEXsnLCurWkTFb6u
swXoVvCE0+qN78d4BjZBph9rX17RfBrc2/5sEZC4yEvGjczSrA6ifbVVdwIn+ENj
ff590cpvReexZ/xGR/03YrqJfUrZ90auKBckz1B9AoGBAKAlx5AYPDhmxRgnmQt
nzgK7YZtMCmPPJaFHpRmXUGwjMpX0M0xoBgA+H0GwovVYDQGCbt3c9bkYAmu4rJ/
E9aUT+nUVvD4a8u0eLff30RpN7hc1hC2rb7YwrkVgcWPJrsAPa8S1G1M//Lbw1Rt
b/UT5ybx/jIwKEgV/h5v2RGdAoGBALm73NJddgHLKdwB1STow+Hk0j0mc6Sezc/
zrtb/vfukc+eoKoc0LGrFDdqv0rnj9I5b945jq/LZAYurAjMdoCwg2bcnjGcjP6i
VKZz9ICshwQN89ROMwsUI2JcsiUh8XxdFOPTZVtFEj53rWWnckCVsB5vdz8naBi
5HYVyXxtAoGAJ/njMQH8gB7I6fRU4I0Hso9h7ZoJZZ8Q27IUeBLQGJ5hqGlmQ5xs
kV2vsV/d4QGuH09DQbGDk3uPPqKmQA4oQR2xf/TC SW6ud7nSSB0eEHwYDm8A1Xm1
CDzFKMDRvfBtb/5A3LTfdQebpHK2QST3t3PHQb4ZVAqg78iRjowZajs=
-----END RSA PRIVATE KEY-----

~/Dev
> [ ]
```

Signing a claim



Verifier

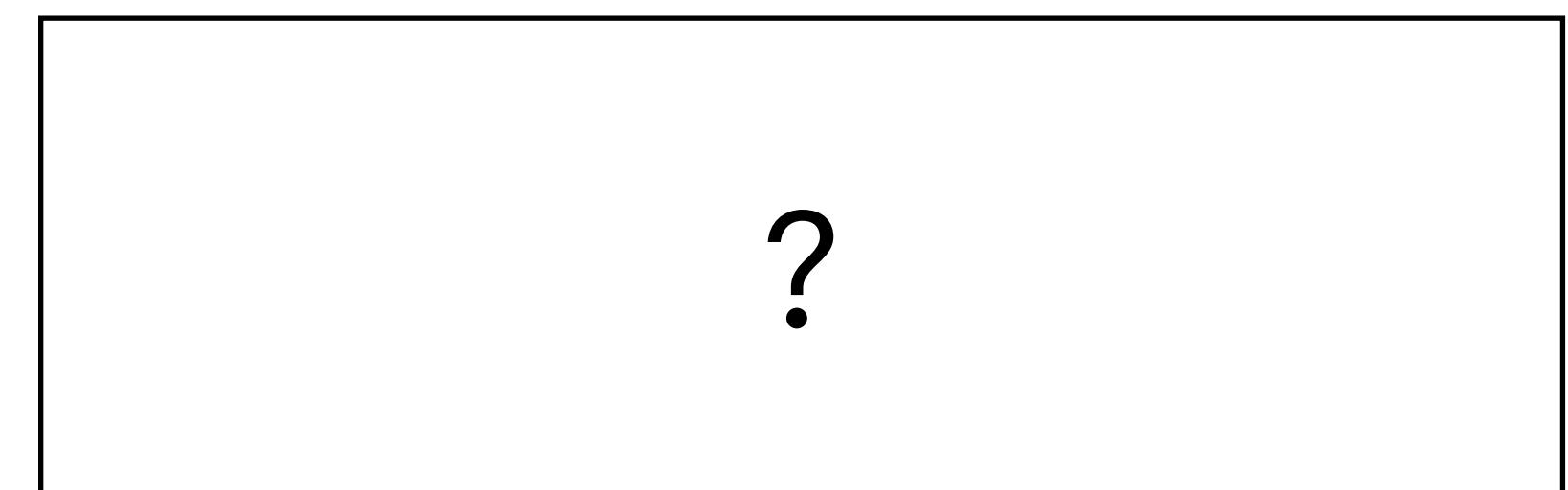
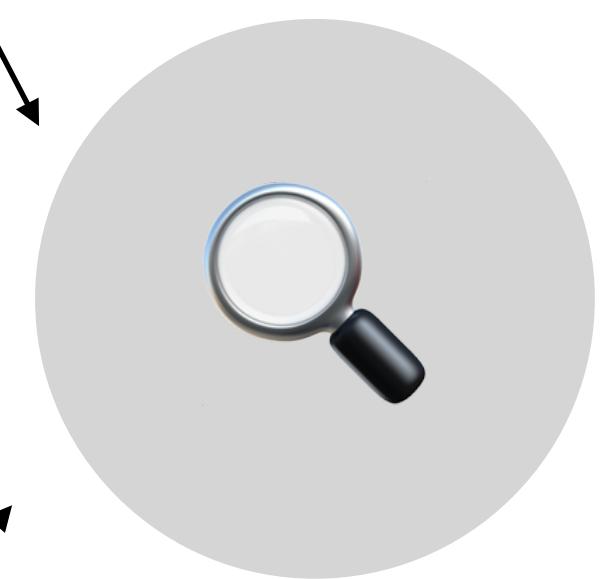
Public key

Signature

Claim:
This is a picture of Miu



Verifying a signature



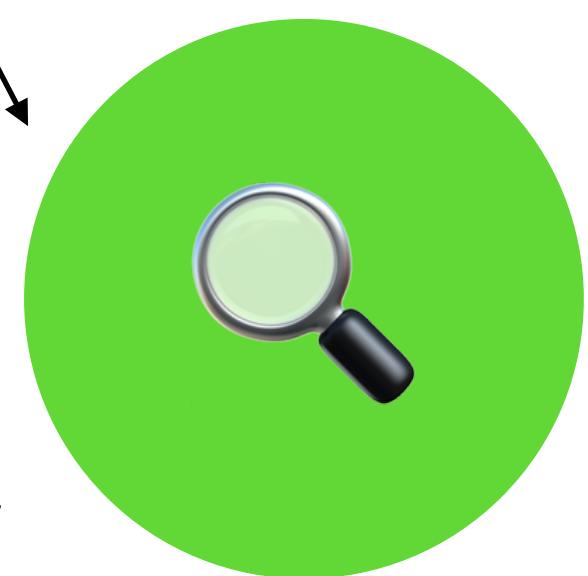
Public key

Signature

Claim:
This is a picture of Miu



Verifying a signature



True

What does this prove?



True

This is a picture of Miu



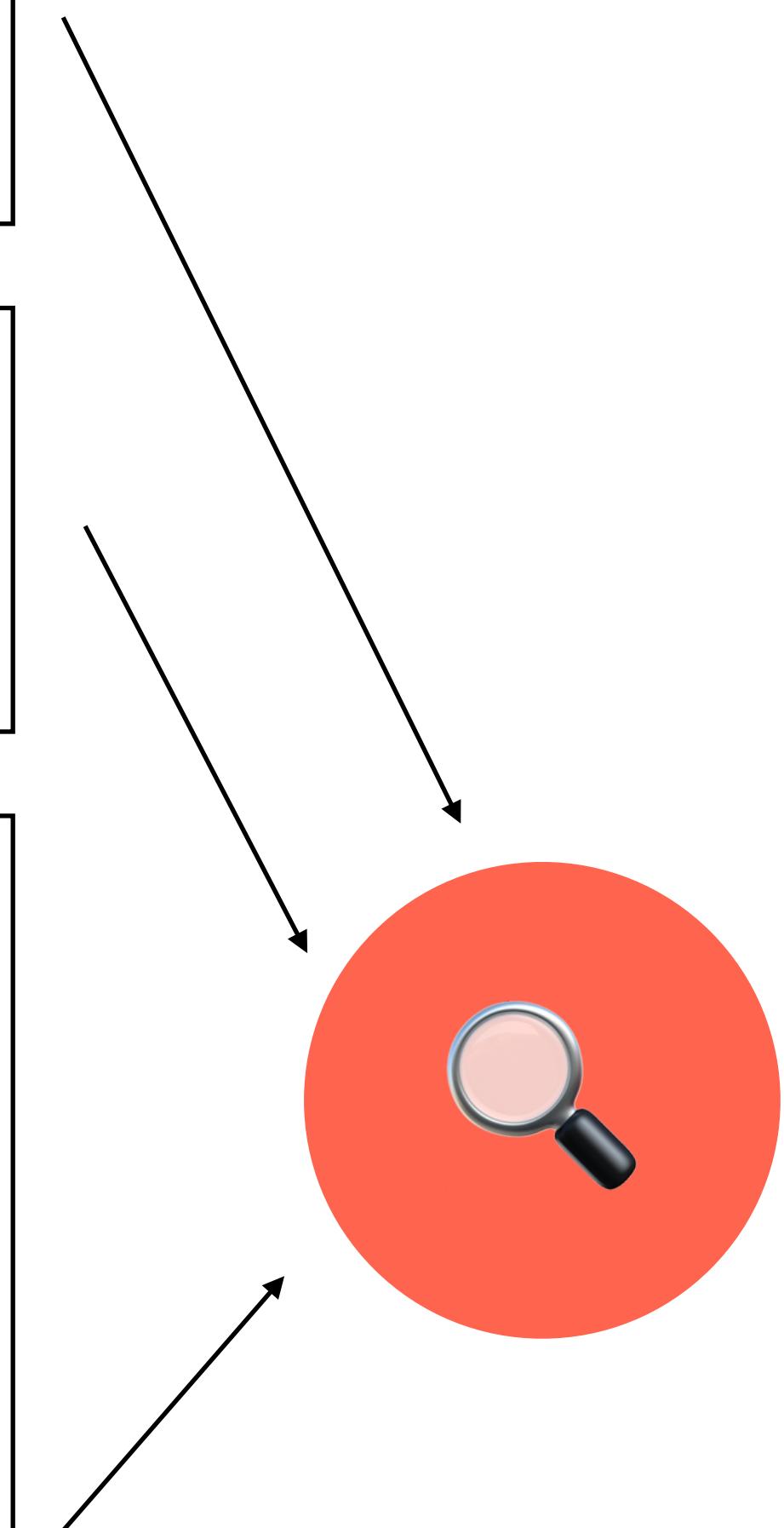
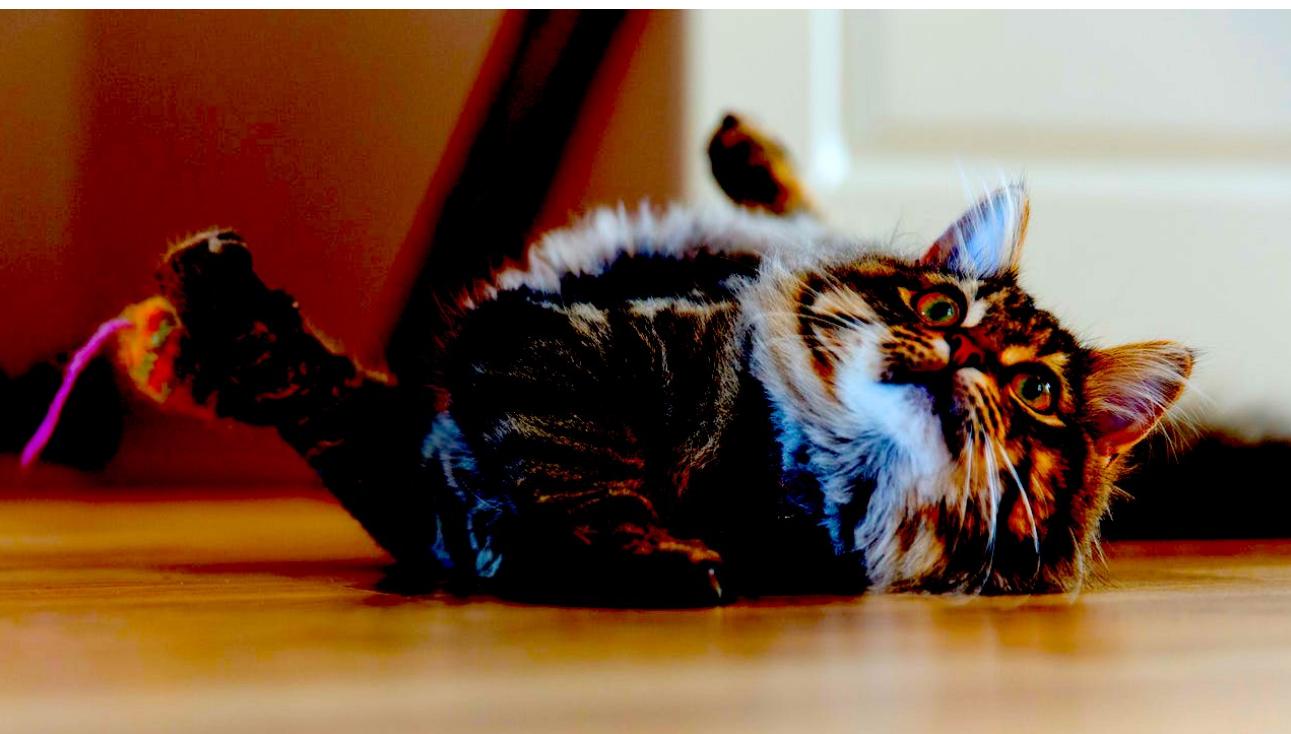
according to the Signer

Verifier

Public key

Signature

Claim:
This is a picture of Miu



Tampered photo

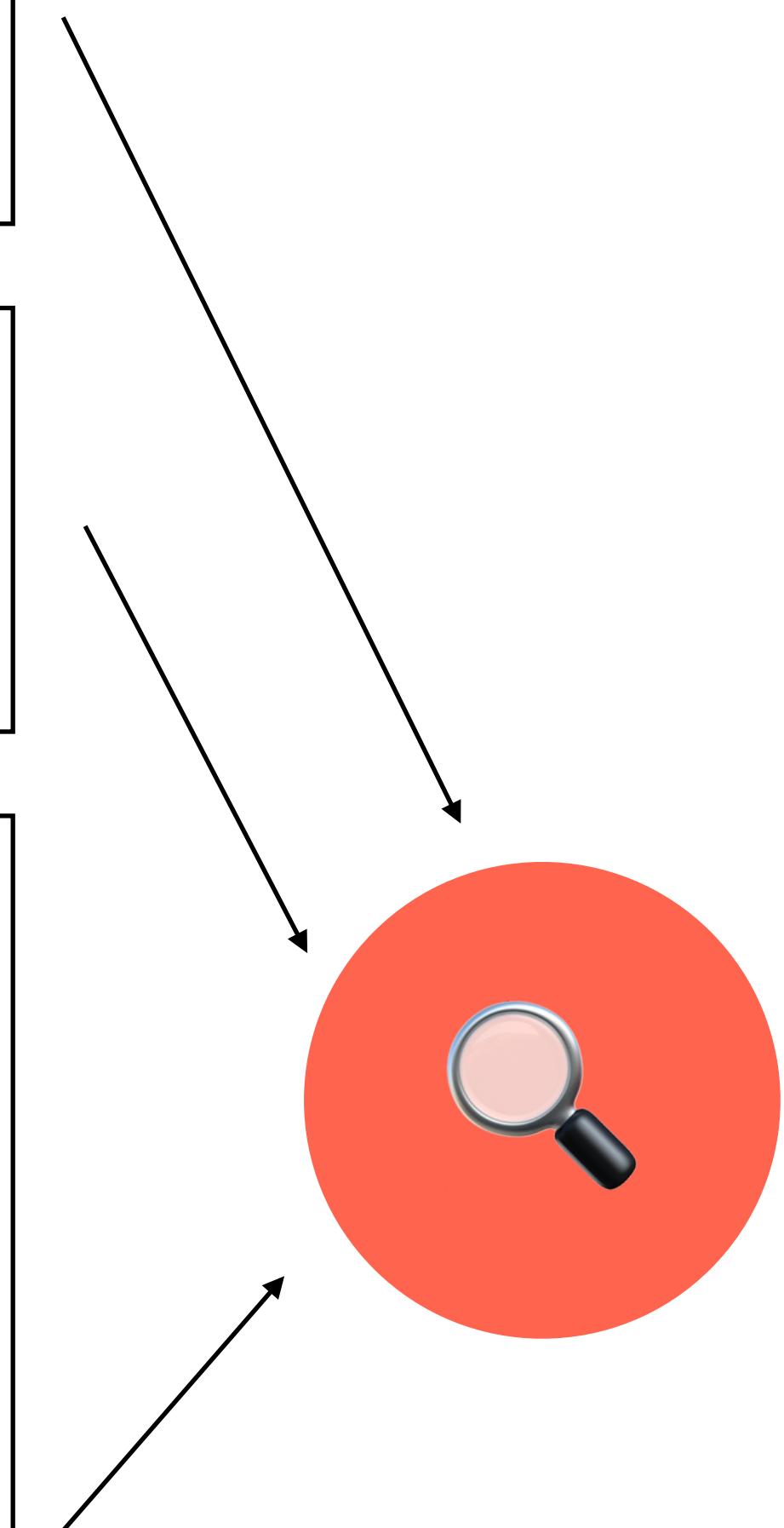
False

Verifier

Public key

Signature

Claim:
This is a picture of Mao



Tampered claim

False

**We can cryptographically verify that the data is
untampered, but do you trust the signer's claim?**

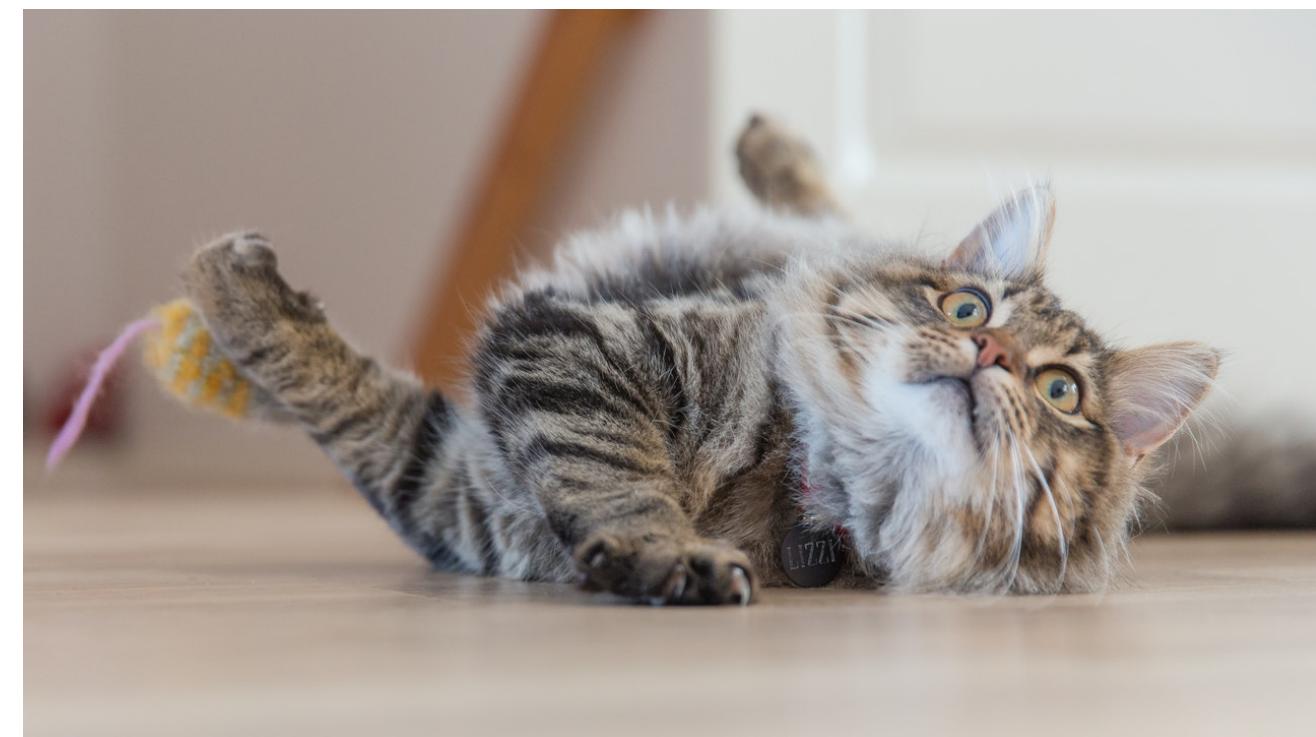
2. Relevant concepts



Integrity data vs. asset data

```
9d88 978e 7b2c ae78  
7782 35de f098 5408
```

~100 bytes



Arbitrary size

Claim:

This is a picture of Miu



367e 62d5 2dc7 7075
a9d0 1079 7f31 82f3

Signer
Private key

Signature

Integrity data vs. asset data

367e 62d5 2dc7 7075
a9d0 1079 7f31 82f3

Signer
Public key

Signature

~1,000 bytes

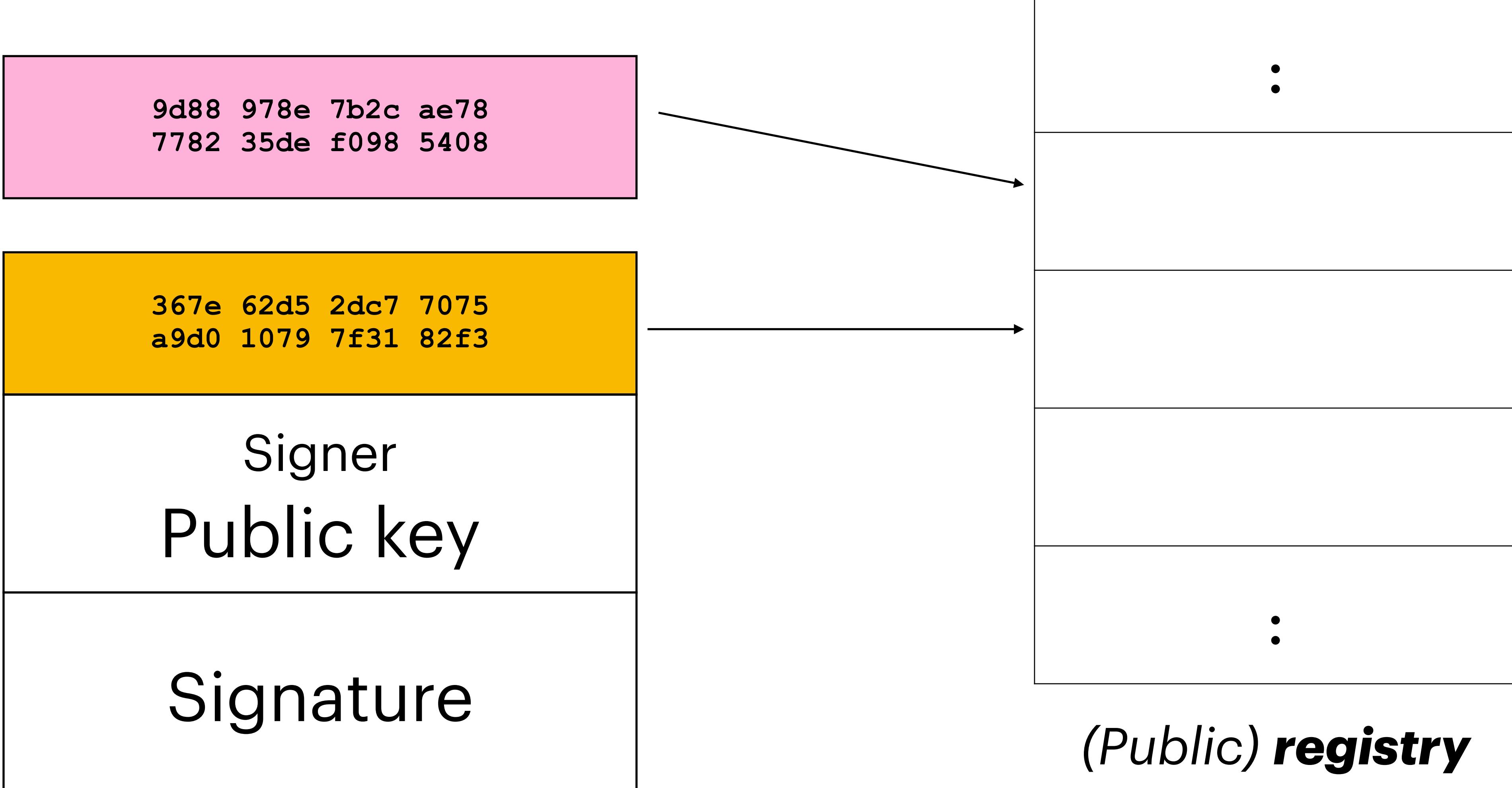
Claim:

This is a picture of Miu



Arbitrary size

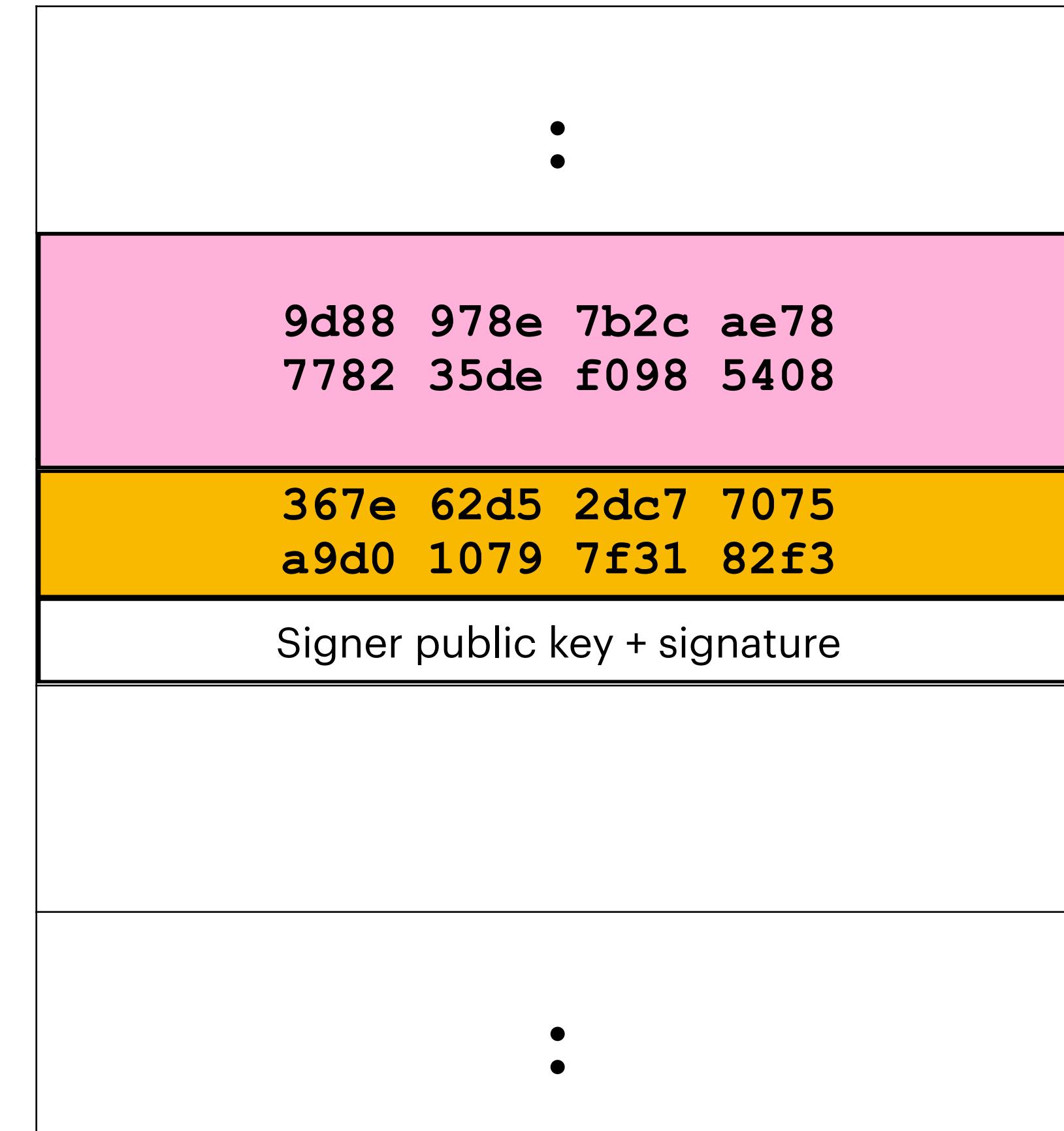
Data registration



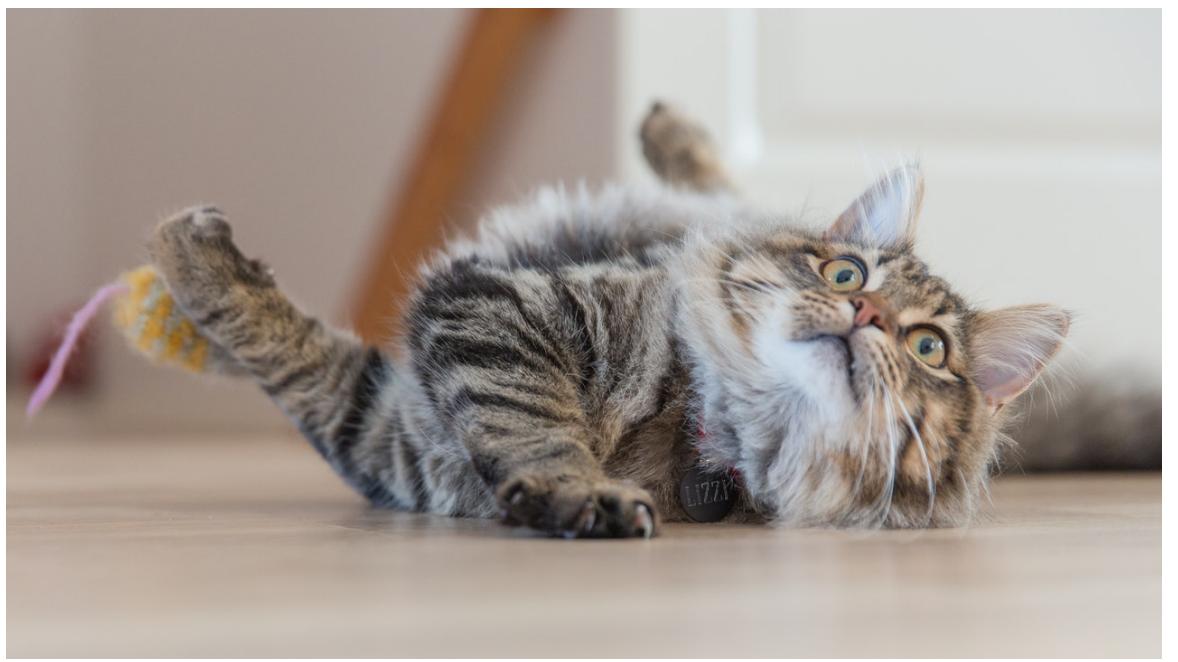
Immutable public ledgers

Since we cannot guess a pre-image, a registered digest is a **proof of existence** of the asset at a particular point in time

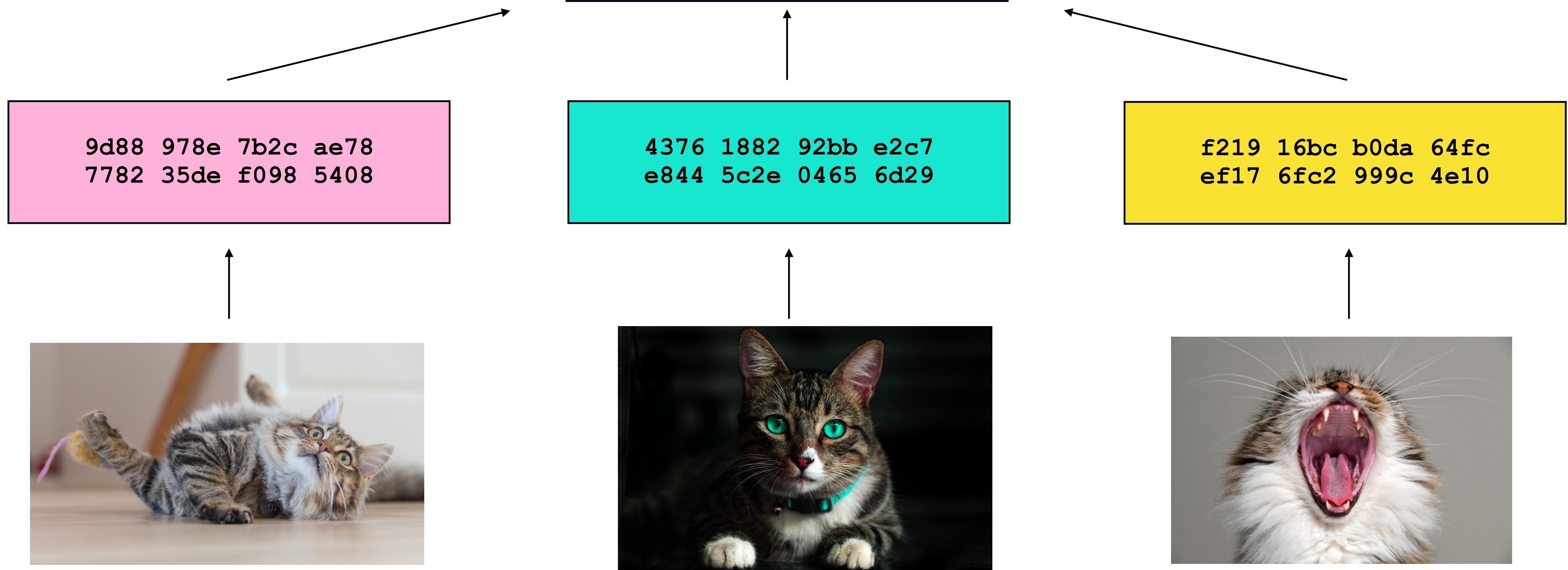
A publicly registered signature supports **verifiable attribution** of claims



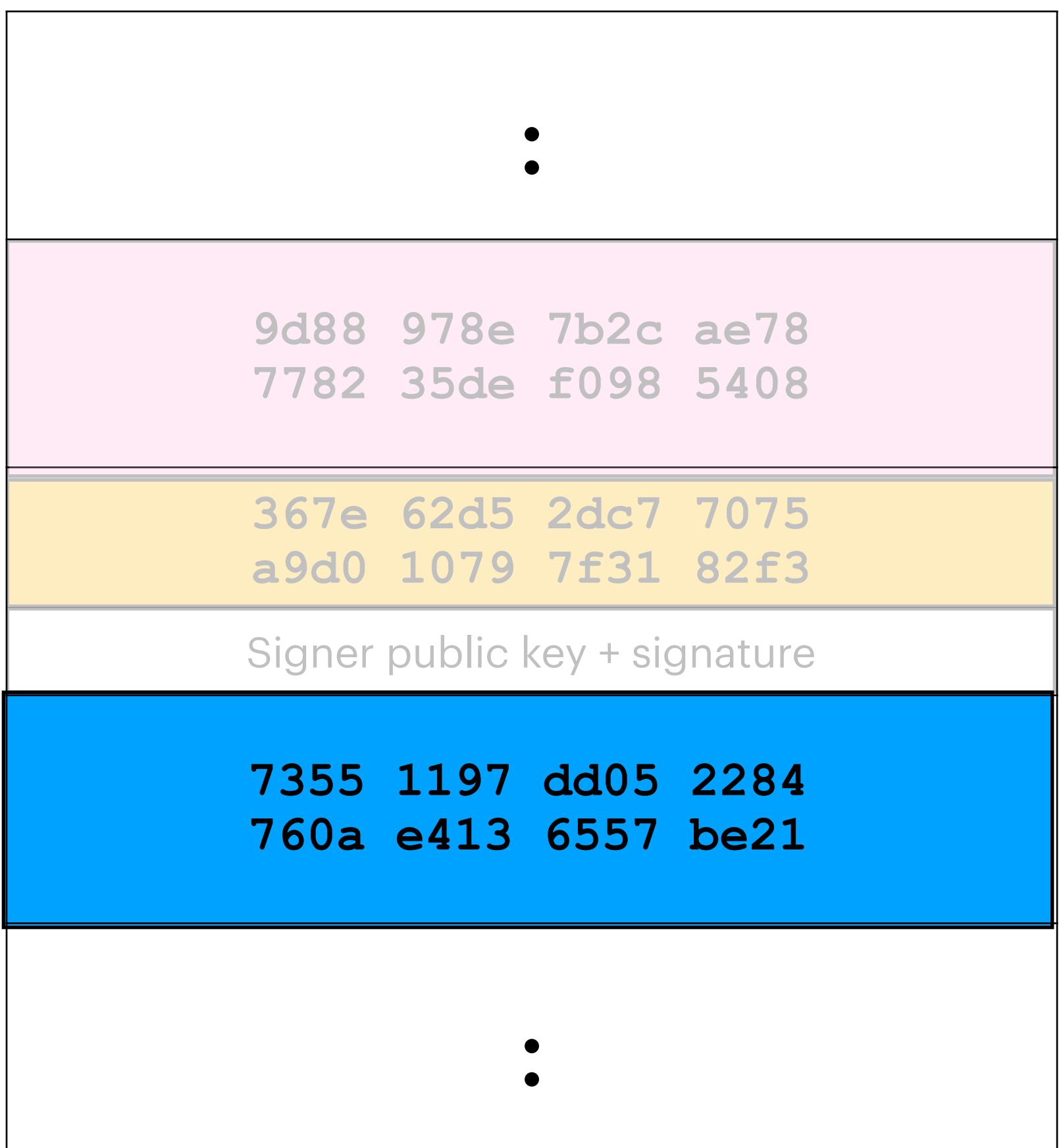
Merkle proofs



Merkle proofs



Merkle proofs



Merkle proofs

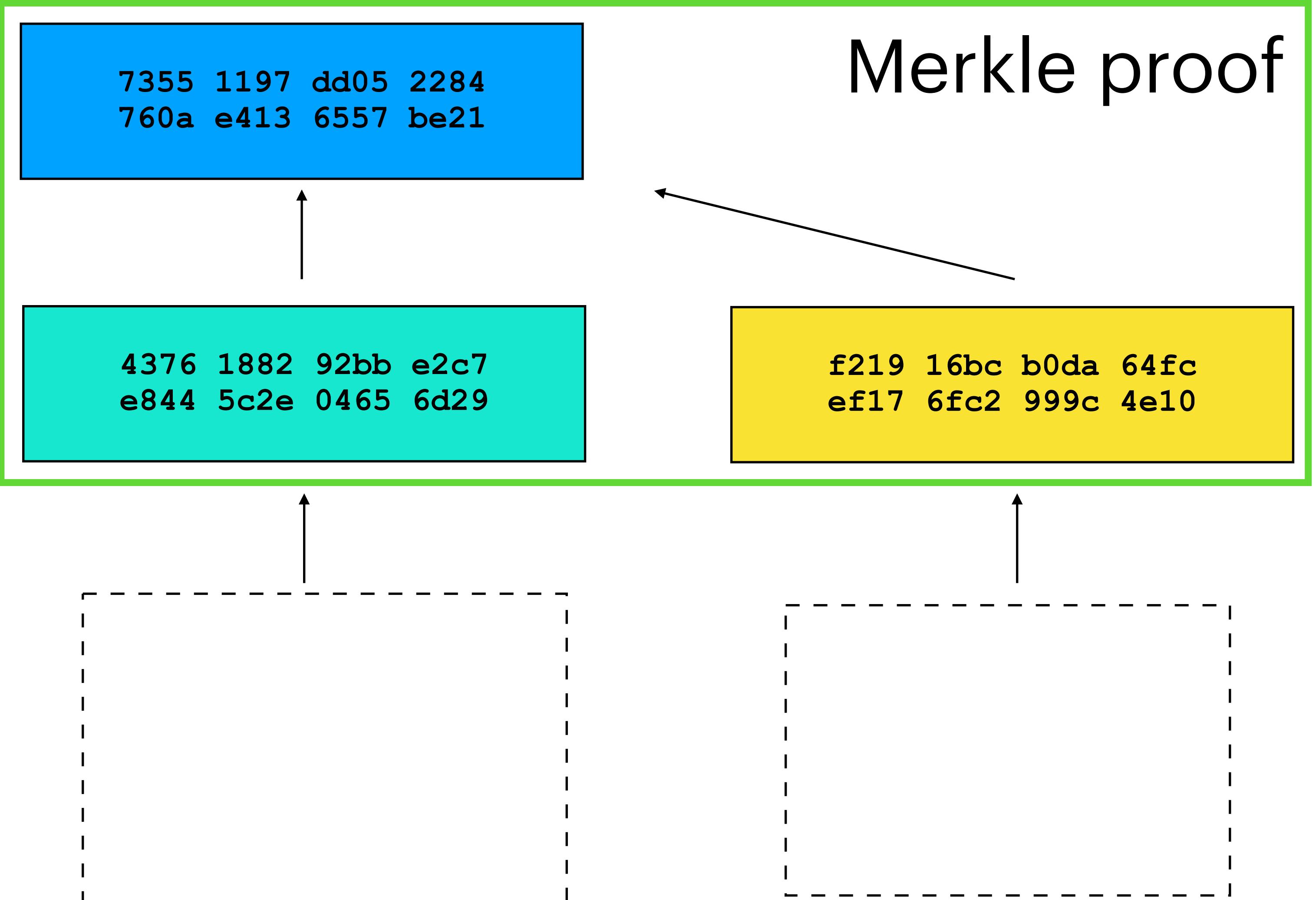
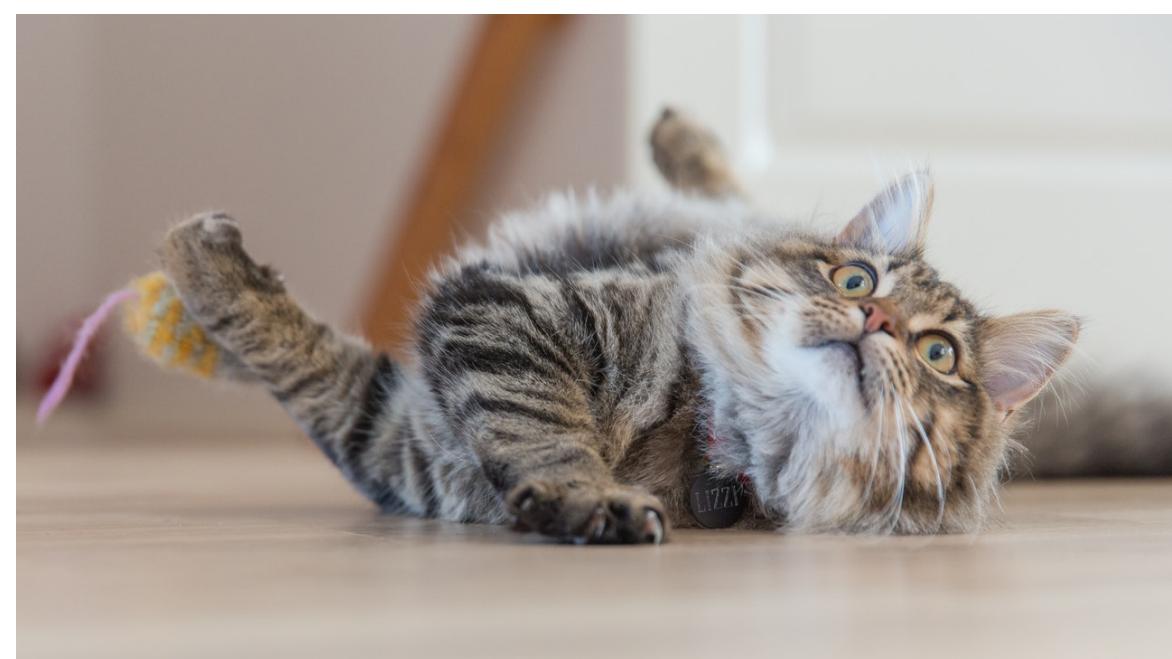


How do we prove that



is included in this digest?

Merkle proof



Selective disclosure





```
7355 1197 dd05 2284  
760a e413 6557 be21
```

```
4376 1882 92bb e2c7  
e844 5c2e 0465 6d29
```

```
f219 16bc b0da 64fc  
ef17 6fc2 999c 4e10
```

7355 1197 dd05 2284
760a e413 6557 be21

9d88 978e 7b2c ae78
7782 35de f098 5408

f219 16bc b0da 64fc
ef17 6fc2 999c 4e10



9d88 978e 7b2c ae78
7782 35de f098 5408

7355 1197 dd05 2284
760a e413 6557 be21

4376 1882 92bb e2c7
e844 5c2e 0465 6d29



Where do we store the actual asset?

Arbitrary size





~70 years

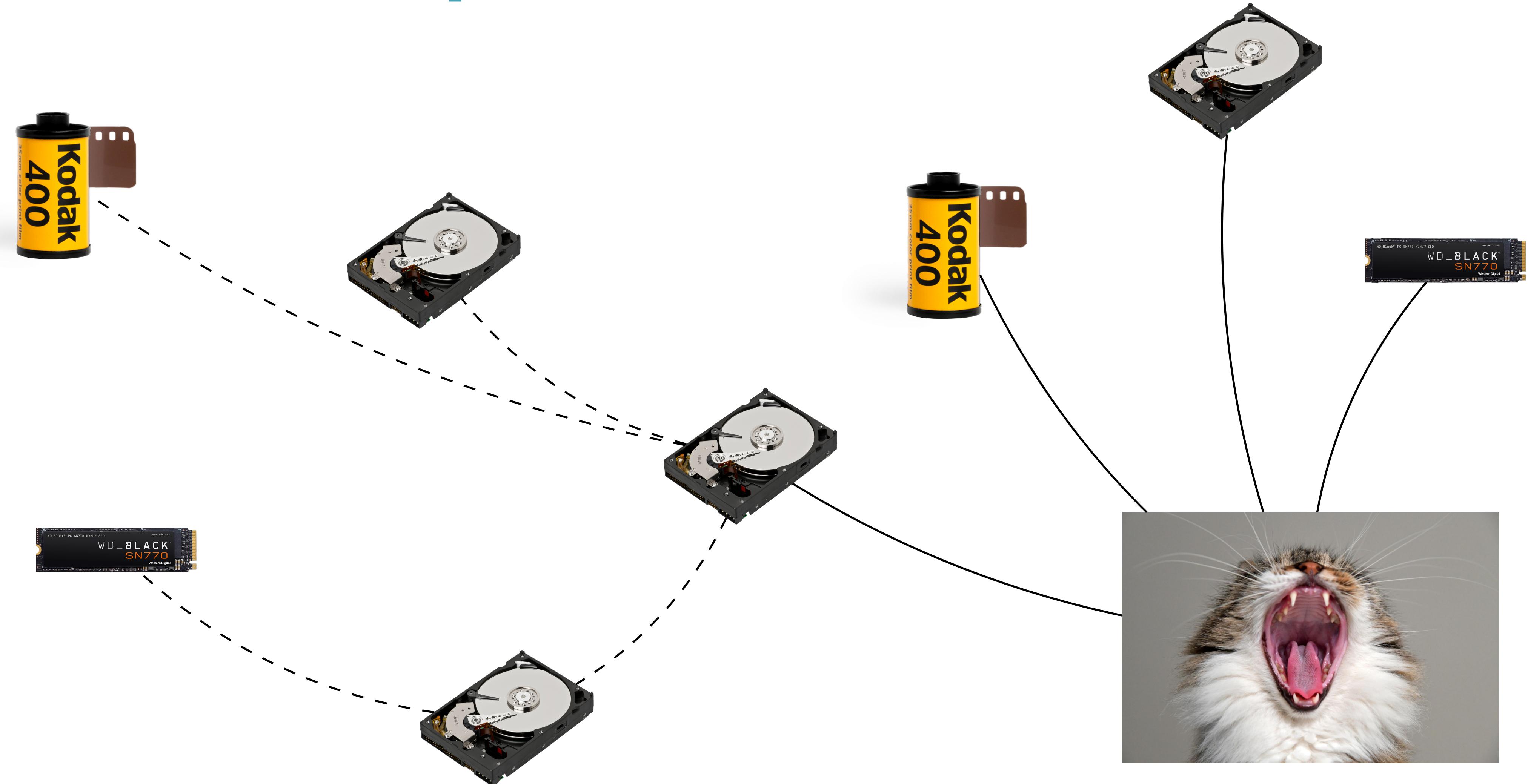


~10 years

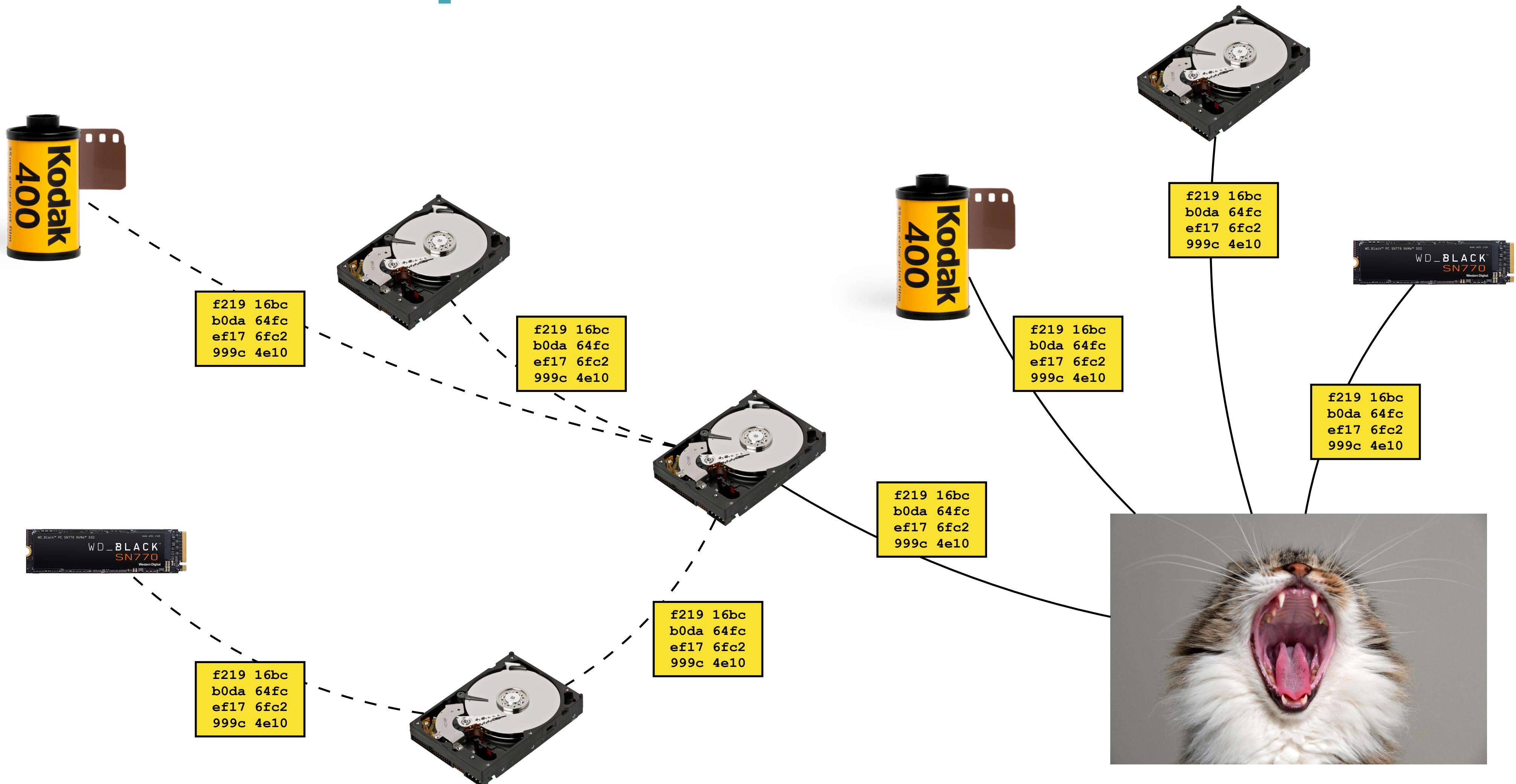


~5 years

Decentralized preservation



Decentralized preservation



Content addressing

Host-address: find asset using host name + host-specific identifier

<https://youtu.be/0X95Nybu2g8>

Content-address: find asset using a digest-based content identifier (CID)

<ipfs://QmcCXAbjDwnkhzL8qfrbJLH6DqEbrkBvaF8AiXsdXNLiDL>

[CID](#)[Docs](#) [Spec](#) [Tutorial](#)**QmcCXAbjDwnkhzL8qfrbJLH6DqEbrkBvaF8AiXsdXNLiDL**

HUMAN READABLE CID

base58btc - cidv0 - dag-pb - (sha2-256 : 256 : CDEE70FBF12D9CD962C11AB94ECCA32B7C9CEA7254FDD0EB322B7F85EBEDB97)

MULTIBASE - VERSION - MULTICODEC - MULTIHASH (NAME : SIZE : DIGEST IN HEX)

MULTIBASE

PREFIX:

implicit

NAME:

base58btc

MULTICODEC

CODE:

0x70

NAME:

dag-pb

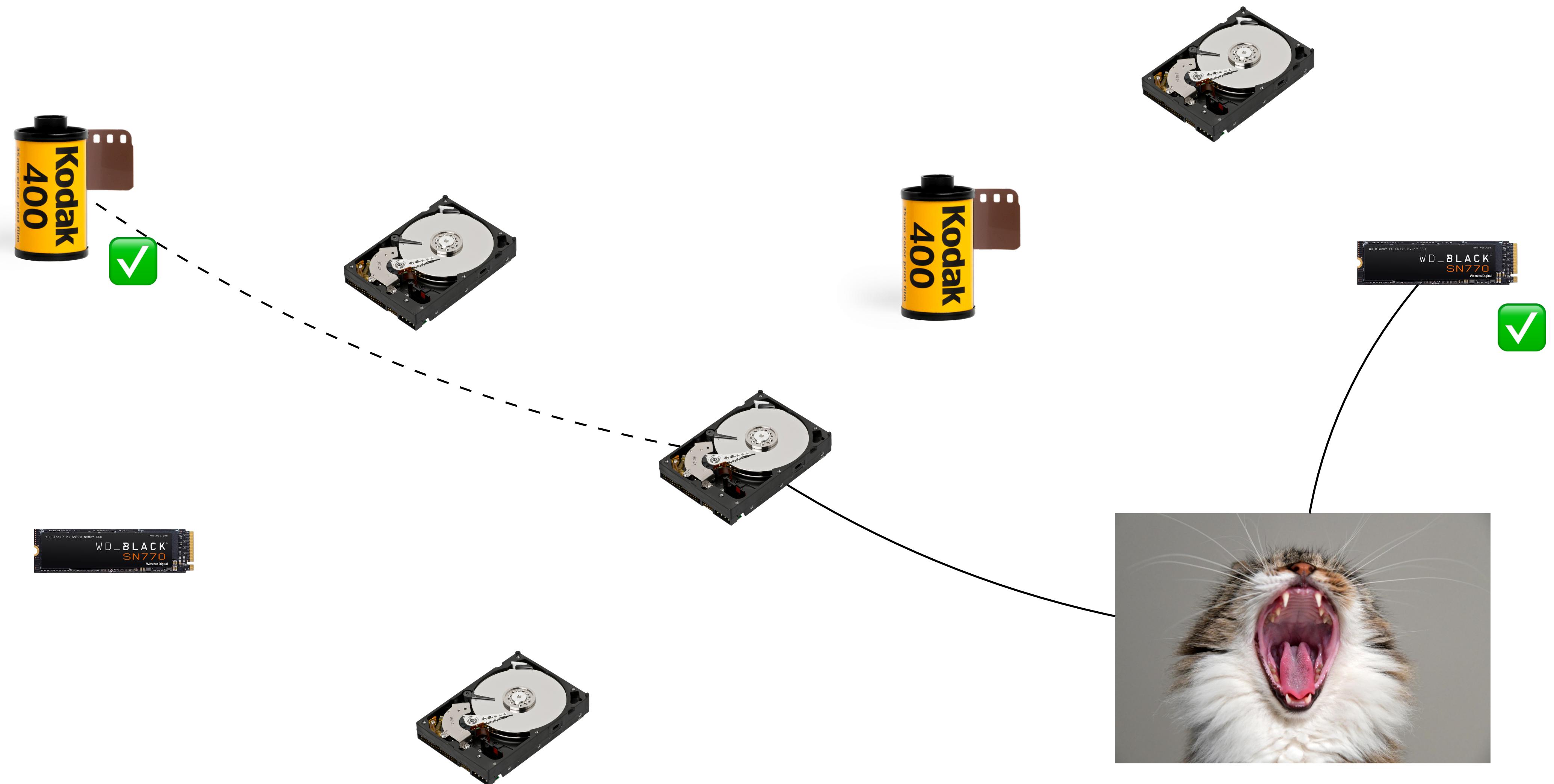
DESCRIPTION:

MerkleDAG protobuf

MULTIHASH



Who has
QmcC...LiDL



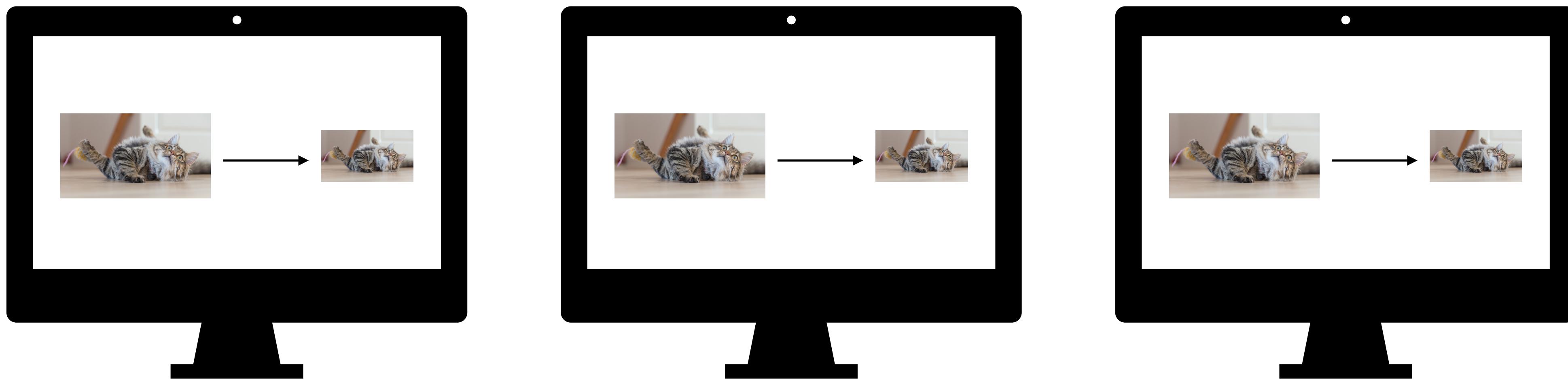
Verifiable computing



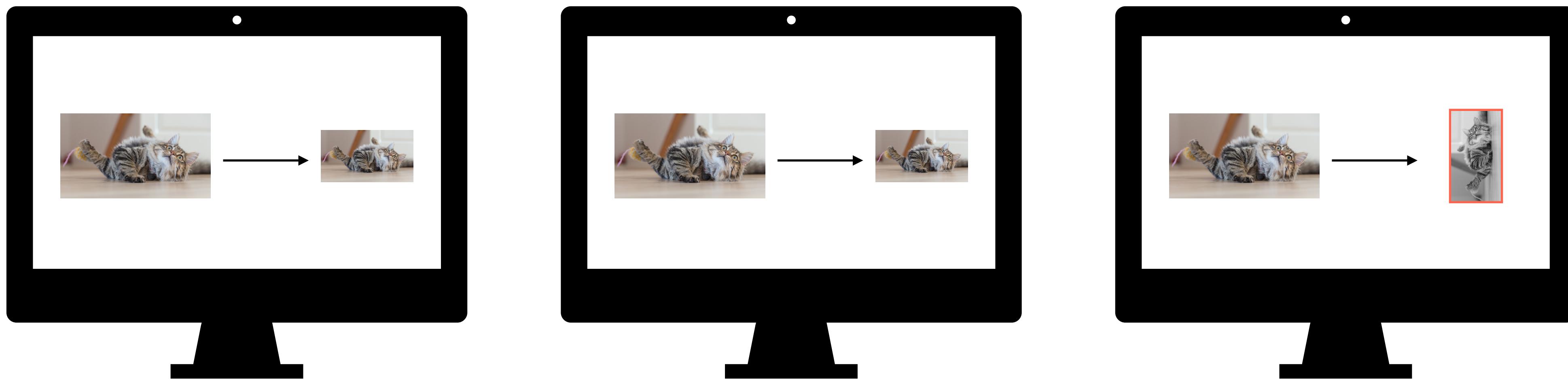
Transformation →



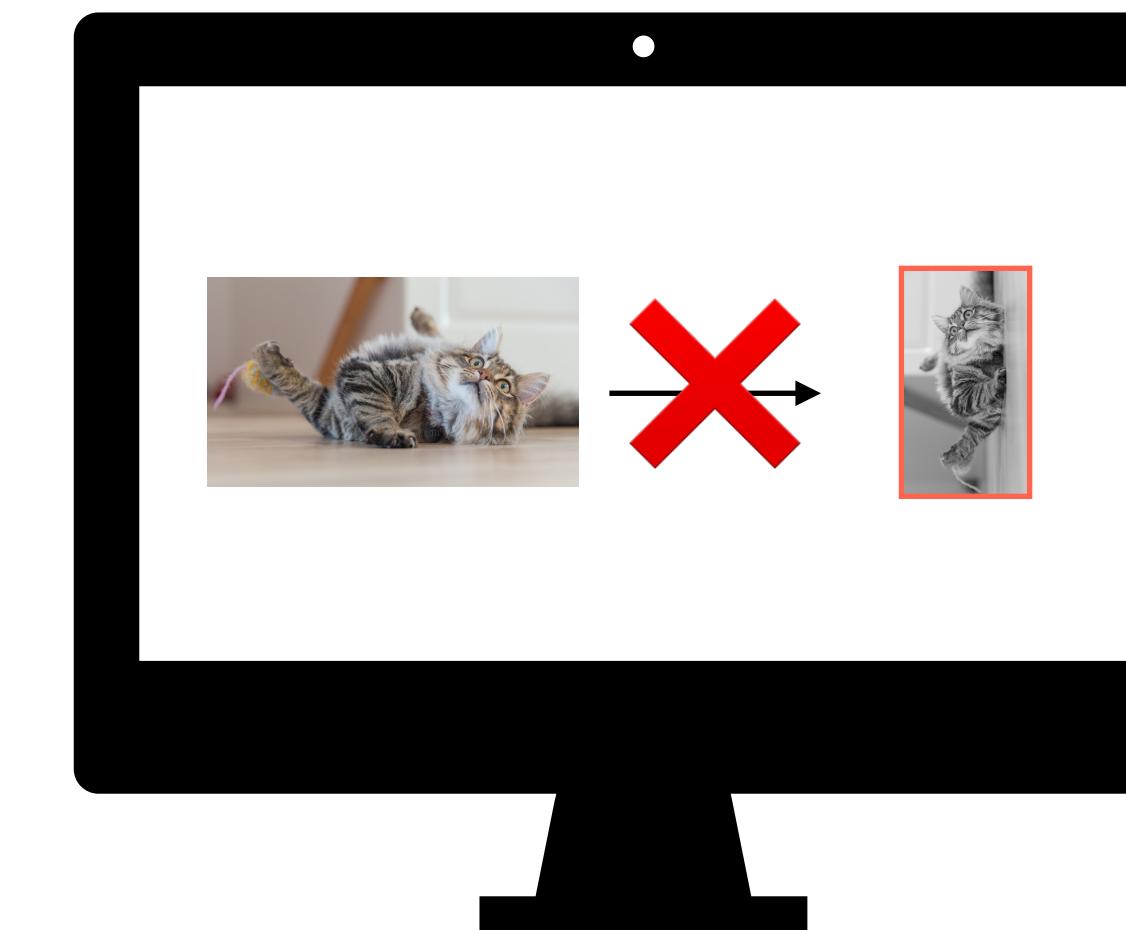
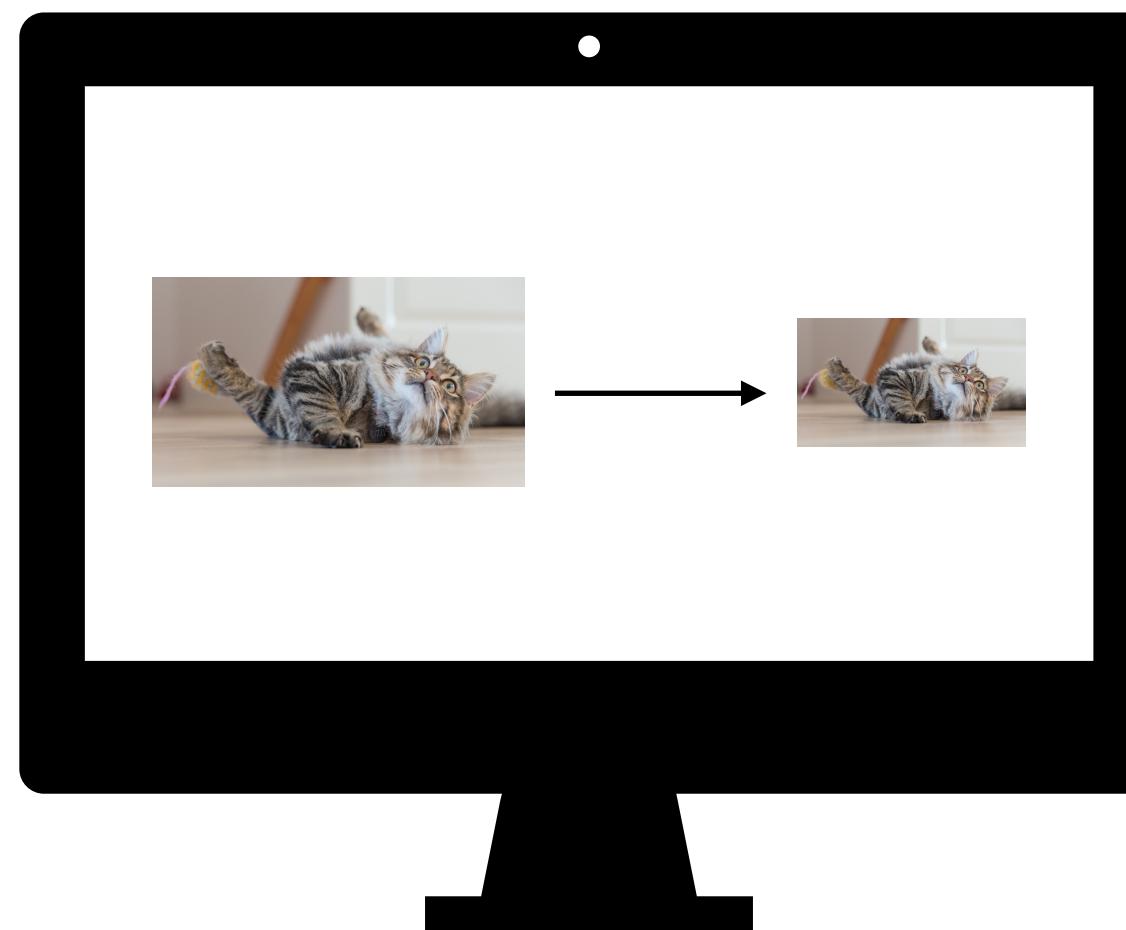
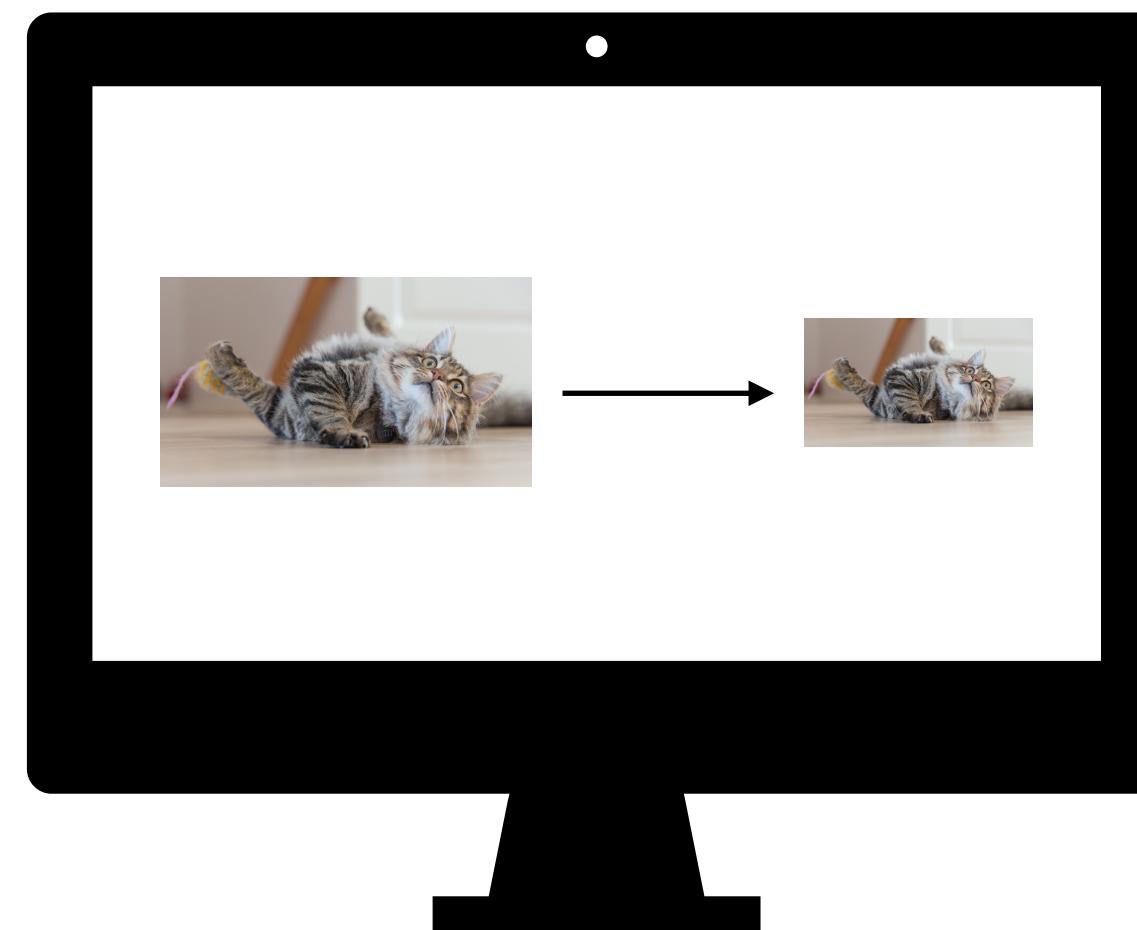
Consensus on blockchains



Consensus on blockchains



Consensus on blockchains



— Consensus algorithm —

ZK SNARK proofs

Zero-Knowledge

Succinct

Non-interactive

Argument of Knowledge



Argument of knowledge

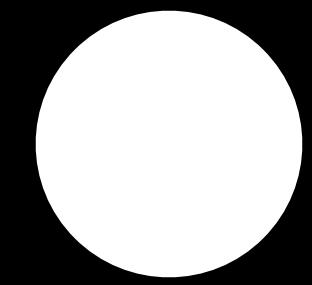
I found Waldo

Zero-Knowledge: I do not want to reveal where Waldo is

Succinct: The proof is compact and easy to verify

Non-interactive: You can verify the proof as is without interaction with me (the prover)

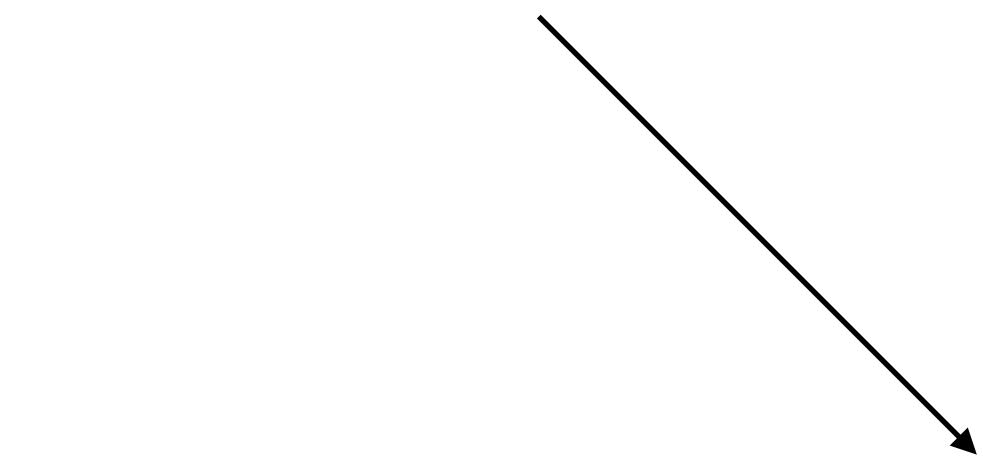
ARgument of Knowledge: I found Waldo







Verifiable
Transformation



ZK SNARK

BACK TO ARCHIVE

BACK TO THE STORY

BACK TO ARCHIVE

THE DOCUMENT LISTS HUNDREDS OF NAMES, BUT NOTABLY INCLUDES:

SRDJAN GOLUBOVIC

In our investigation, we found the name **ГОЛУБОВИЋ СРЂАН** in Cyrillic, which is often spelled Srđan Golubović in Serbian or Srdjan Golubovic in English.

The same individual in the payroll records is mentioned in ICTY testimony and has been accused by members of the Serbian press and by local citizens of being the soldier in Haviv's famous photo.

According to the payroll records, Golubović was deployed and paid in September 1994, and January, April, May, June, July, August, September, October, November, and December 1995.

РЕПУБЛИКА СРБИЈА
МИНИСТАРСТВО УНУТРАШЊИХ ПОСЛОВА
РЕСПОРД РДБ-ЈАТД
Бр. 120-5/95
13.01. 1995 год.
Београд, Кнеза Милоша 103

06332618

СПИСАК ПРИПАДНИКА ЈЕДИНИЦЕ ЗА А Т Д РАДИ ИСПЛАТЕ
дневница за период од 1-15.01.95. ГОДИНУ

БРОЈ ПРИПАДНИКА 225
ИЗНОС ПО ПРИПАДНИКУ 299,50
УКУПАН ИЗНОС 80.437,50

РЕПУБЛИКА СРБИЈА
МУП - РДБ - ЈАТД
БРОЈ: 121-163
ДАНА: 17.04.1995. ГОДИНЕ
Б Е О Г Р А Д

06332409

СПИСАК ПРИПАДНИКА РАДИ ИСПЛАТЕ ДНЕВНИЦА

Р.Б.	ПРЕЗИМЕ И ИМЕ	ВРЕМЕ АНГАЖАВА.	БР ДНЕ	ИЗНОС ДНЕВНИ.	УКУПНО	ПОТПИС
1		1-15.1.	15	19,50	292,50	
2		1-15.1.	15	19,50	292,50	
3	Голубовић Срђан	1-15.1.	15	19,50	292,50	
5		1-15.1.	15	19,50	292,50	
6		1-15.1.	15	19,50	292,50	
7		1-15.1.	15	19,50	292,50	
8		1-15.1.	15	19,50	292,50	
9		1-15.1.	15	19,50	292,50	
10		1-15.1.	15	19,50	292,50	
11		1-15.1.	15	19,50	292,50	
12		1-15.1.	15	19,50	292,50	
13		1-15.1.	15	19,50	292,50	
14		1-15.1.	15	19,50	292,50	
15		1-15.1.	15	19,50	292,50	

Закључно са редним бројем 15.

06332619

СПИСАК ПРИПАДНИКА РАДИ ИСПЛАТЕ ДНЕВНИЦА

Р.Б.	ПРЕЗИМЕ И ИМЕ	ВРЕМЕ АНГАЖ.	БР ДНЕ	ИЗНОС ДНЕВНИ	УКУПНО
1		1-15.4.	15	21,40	321,00
2		1-15.4.	15	21,40	321,00
3.	GOLUBOVIC SRDJAN	1-15.4.	15	21,40	321,00
4.		1-15.4.	15	21,40	321,00
5.		1-15.4.	15	21,40	321,00
6.		1-15.4.	15	21,40	321,00
7.		1-15.4.	15	21,40	321,00
8.		1-15.4.	15	21,40	321,00
9.		1-15.4.	15	21,40	321,00
10.		1-15.4.	15	21,40	321,00
11.		1-15.4.	15	21,40	321,00
12.		1-15.4.	15	21,40	321,00
13.		1-15.4.	15	21,40	321,00

06332422

DATE

2022-11-02T11:17:20.231Z

CAPTURED USING

WEBRECORDER - ARCHIVEWEB.PAGE / AUTHENTICATED WEB ARCHIVE

LOCATION

LOS ANGELES, CA, UNITED STATES

REGISTERED ON

OPENTIMESTAMPS

E071739EEC16594207BE3BD255CE083C87C83935821BD131CFCD0A2ABE4E3C9C

NUMBERS

AVALANCHE

ISCN

2 STORE

PRESERVED ON

FILECOIN

3 VERIFY

VERIFIED BY

C2PA CLAIM 1: AUTH
ARCHIVE AND SCREEN
11/03/2022

C2PA CLAIM 2: REDA
PROOF

ATTESSTATIONS

WEB ARCHIVE

BAFYBEIHQITR4UYFT4CXYW74VZK5TNPTOSOHL7H64I2EKESZHJCUFL0FCQ

REDACTION: ZERO KNOWLEDGE PROOF

BAFYBEIDE5QAVI57RNZC3CHT64WDUJVMKUD0F5GB5NUWB24X4U5IDNS33UA

CAPTION

BAFKREIG6ZL3AXGFB3D3P7NUQCIMIA2ARAEV64C73UDC3XEBA2UNQ436GI

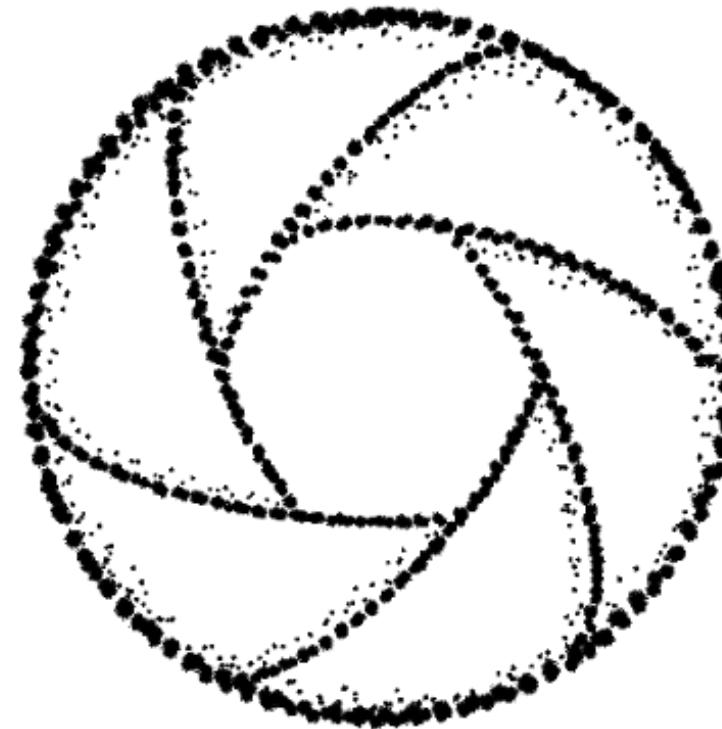
The screenshot shows a web browser window for ipfs.io. The title bar says "ipfs.io". The main content area displays an "Index of /ipfs/bafybeide5qavi57rnzc3cht64wdujvmkudof5gb5nuwb24x4u5idns33ua" page. The page has a size of 181 kB. It lists five files:

File	Content Hash	Size
C049-1641_coords.txt	bafk...jsua	43 B
C049-1641_hash.txt	bafk...c4wi	10 kB
C049-1641_proof.txt	bafk...vizm	548 B
C049-1641_red.png	bafk...n5wy	169 kB
README.txt	bafk..y7ha	217 B

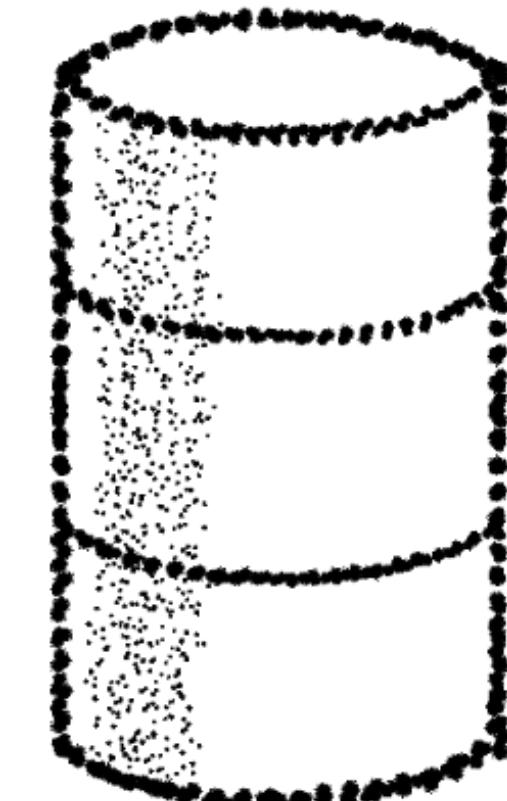
3. Building towards authenticity



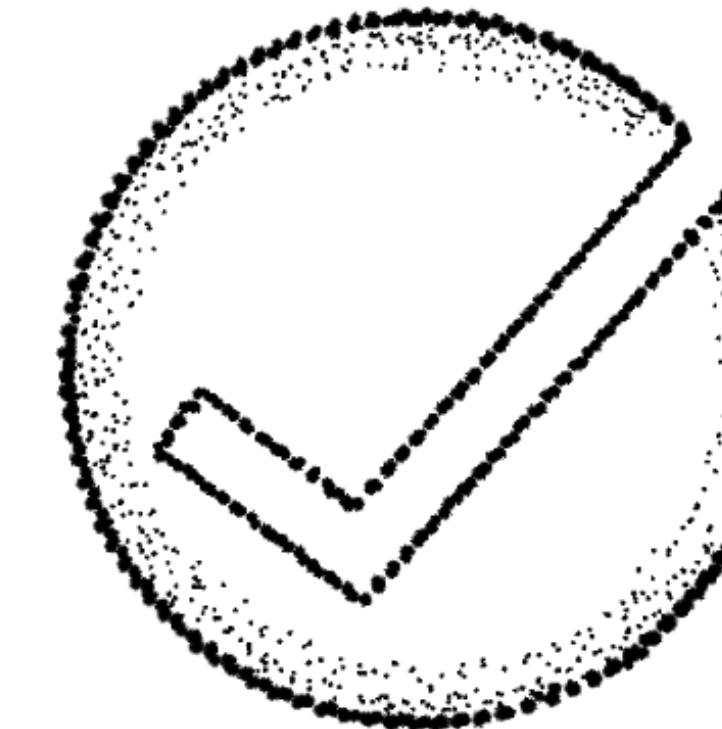
Starling pipeline: Capture · Store · Verify



Gather the image or file, taking note of when and how you obtained it and other contextual metadata

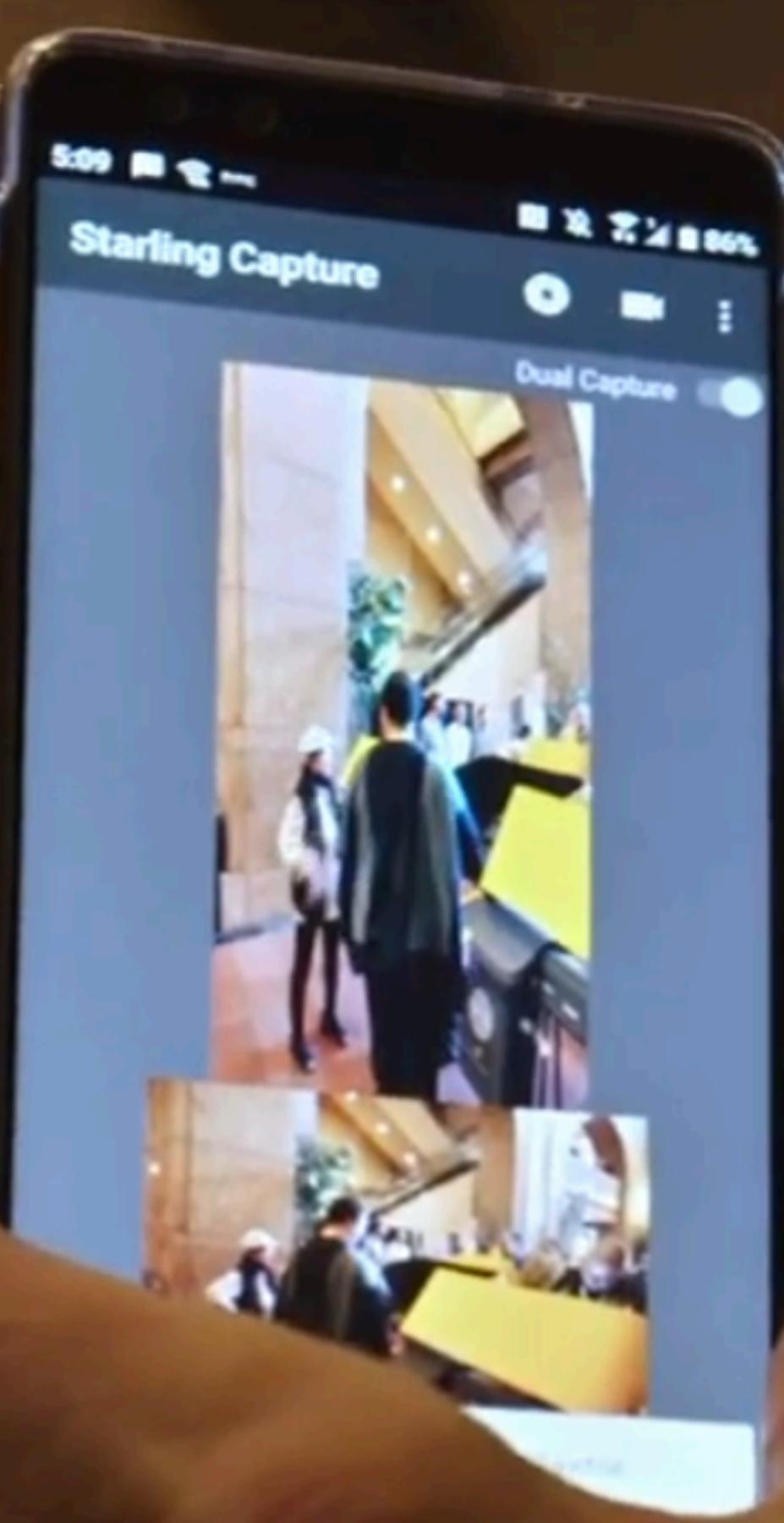


Store the file and its metadata in a way that is available and persistent



Investigate the claims made by the image and metadata

Capture





WANT TO ACCE

Reuters and Canon demonstrate end-to-end content authenticity system in the field

As the leading global news agency, Reuters works relentlessly to bring news from the source and provide readers and viewers with unbiased and reliable information from the world around us. Every day, our journalists capture the first draft of history, without bias or agenda, to inform billions of people.

Photojournalism plays a crucial role in documenting that history. Yet, with generative AI technology now available to anyone with internet access, it has become much easier to create visuals that can deceive or misinform. To make sure we can protect the trust we have built up over decades, we're always exploring innovative technologies. This is one big step forward.

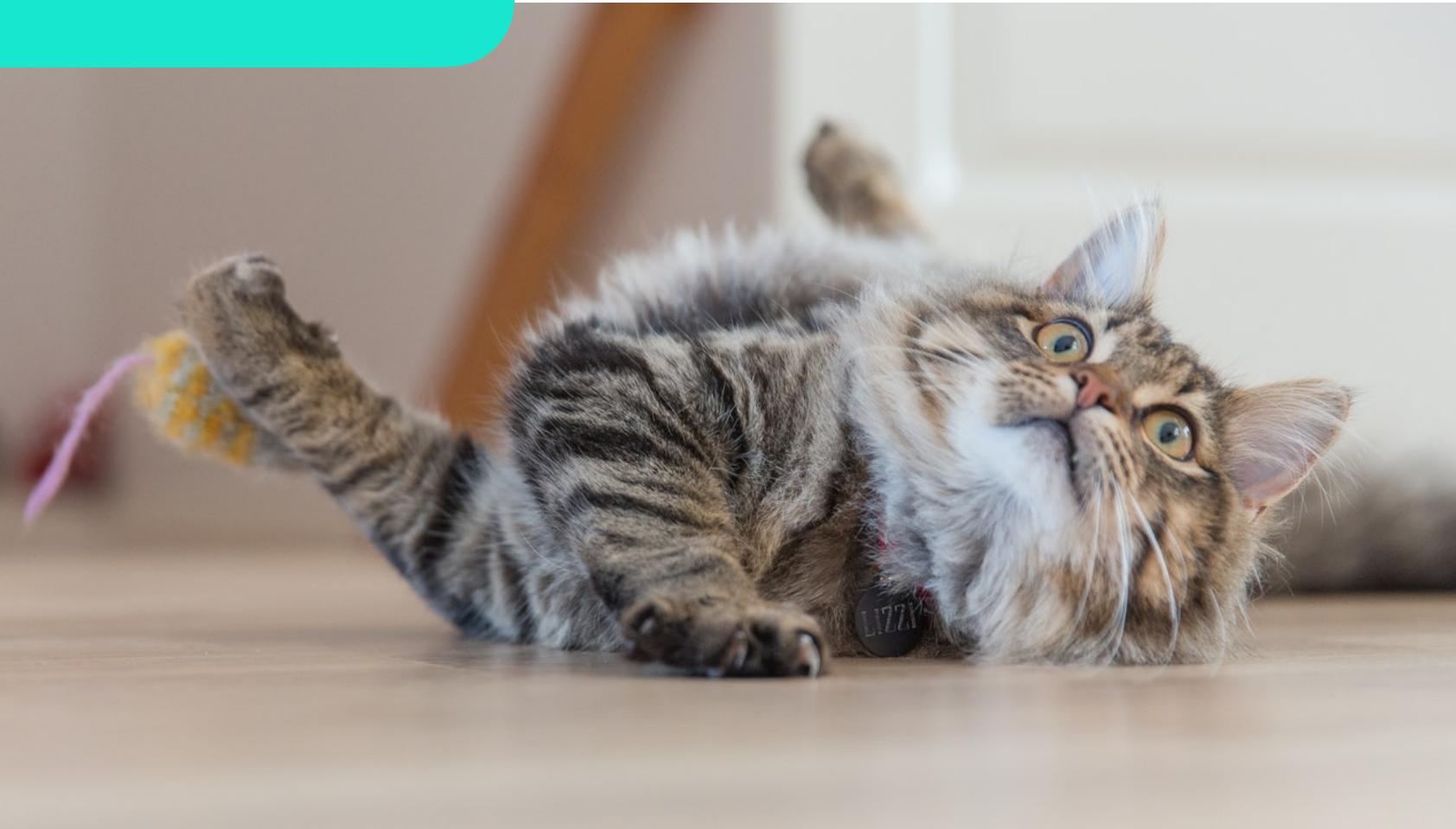
Many photojournalists worldwide rely on Canon technologies. As a leading global camera manufacturer, Canon understands the role images play in society and recognises the importance of preserving image authenticity. Working as part of the [Content Authenticity Initiative \(CAI\)](#) alliance, Canon wants to take meaningful measures to protect image outputs that serve the news community worldwide.



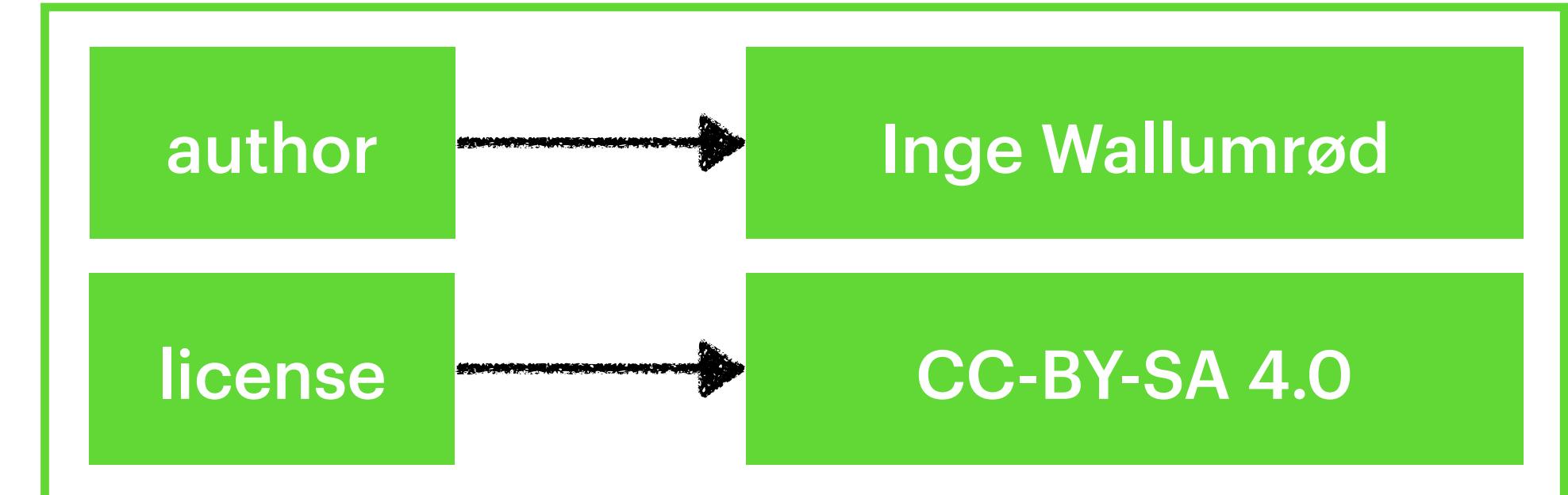
Data



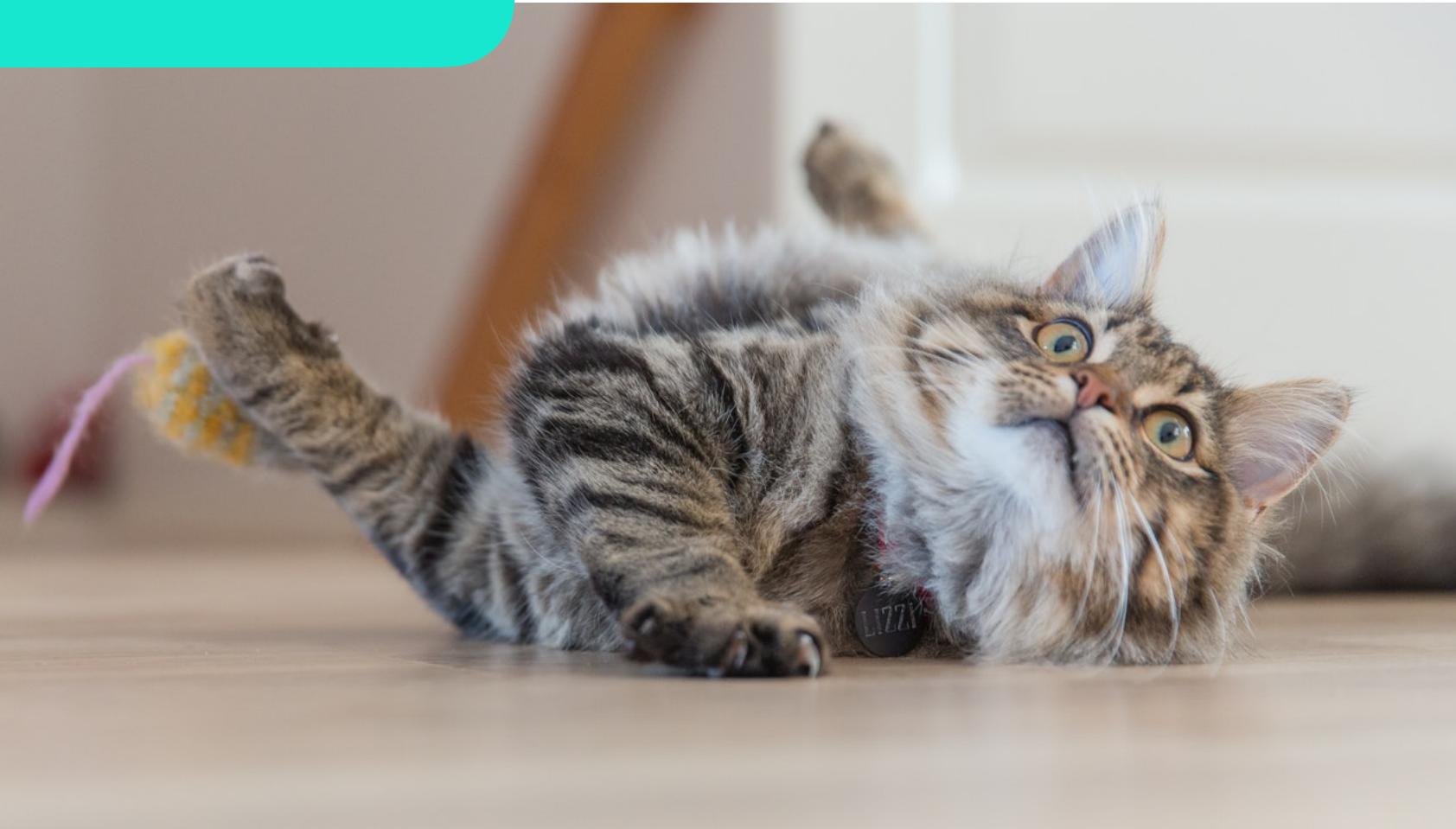
Data



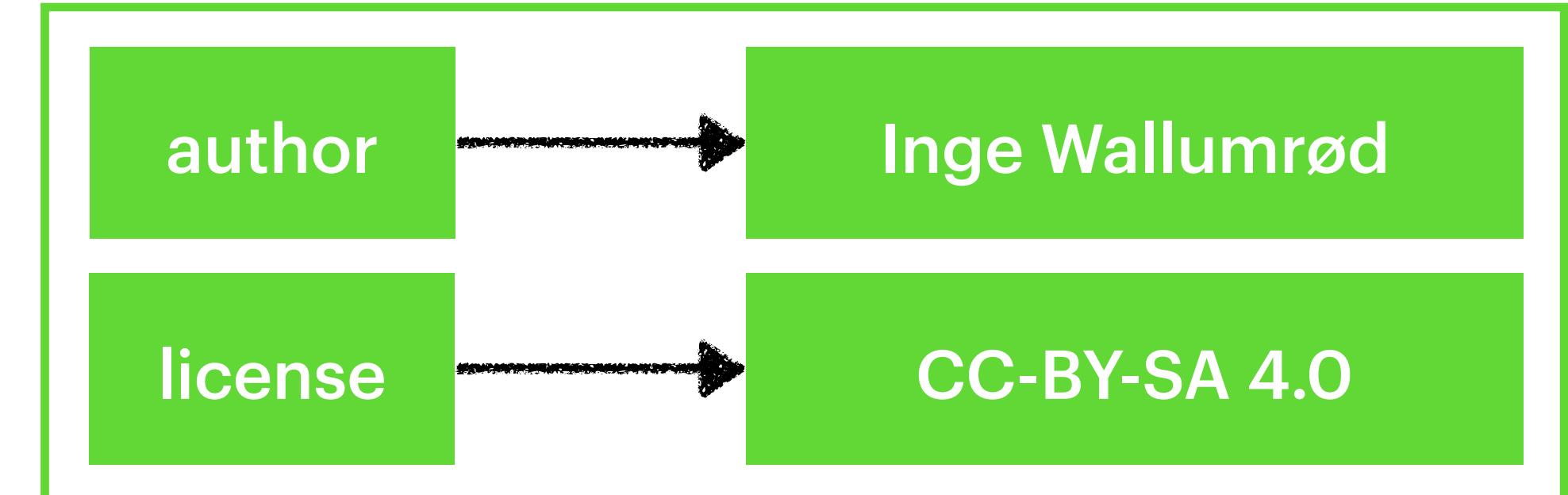
Metadata



Data



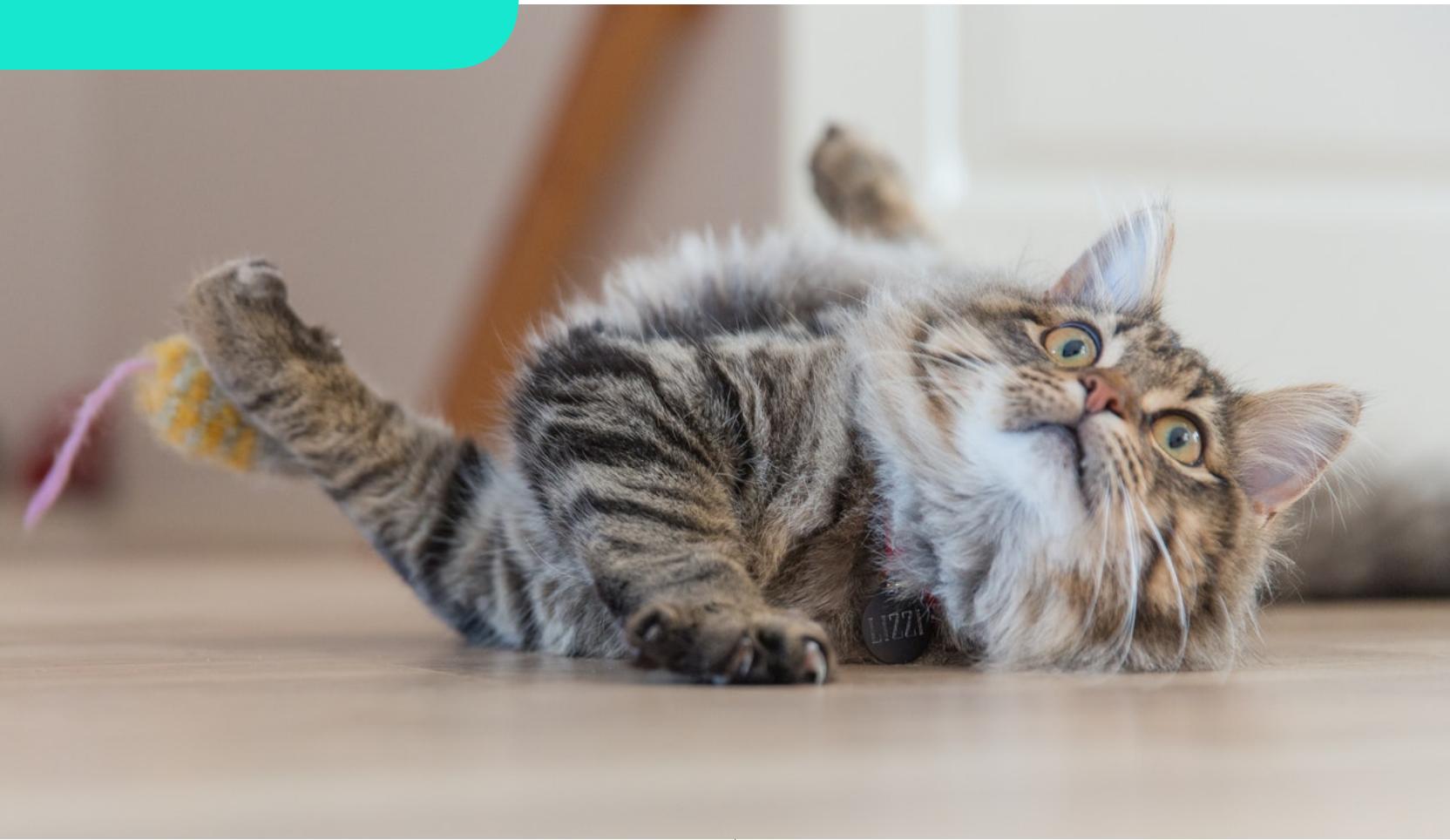
Metadata



↓ SHA 256

600b244925ffc9665c8544083b9cc002
48530f6c7b0ecdd5c89c859e3c5818cf

Data



Metadata

author

Inge Wallumrød

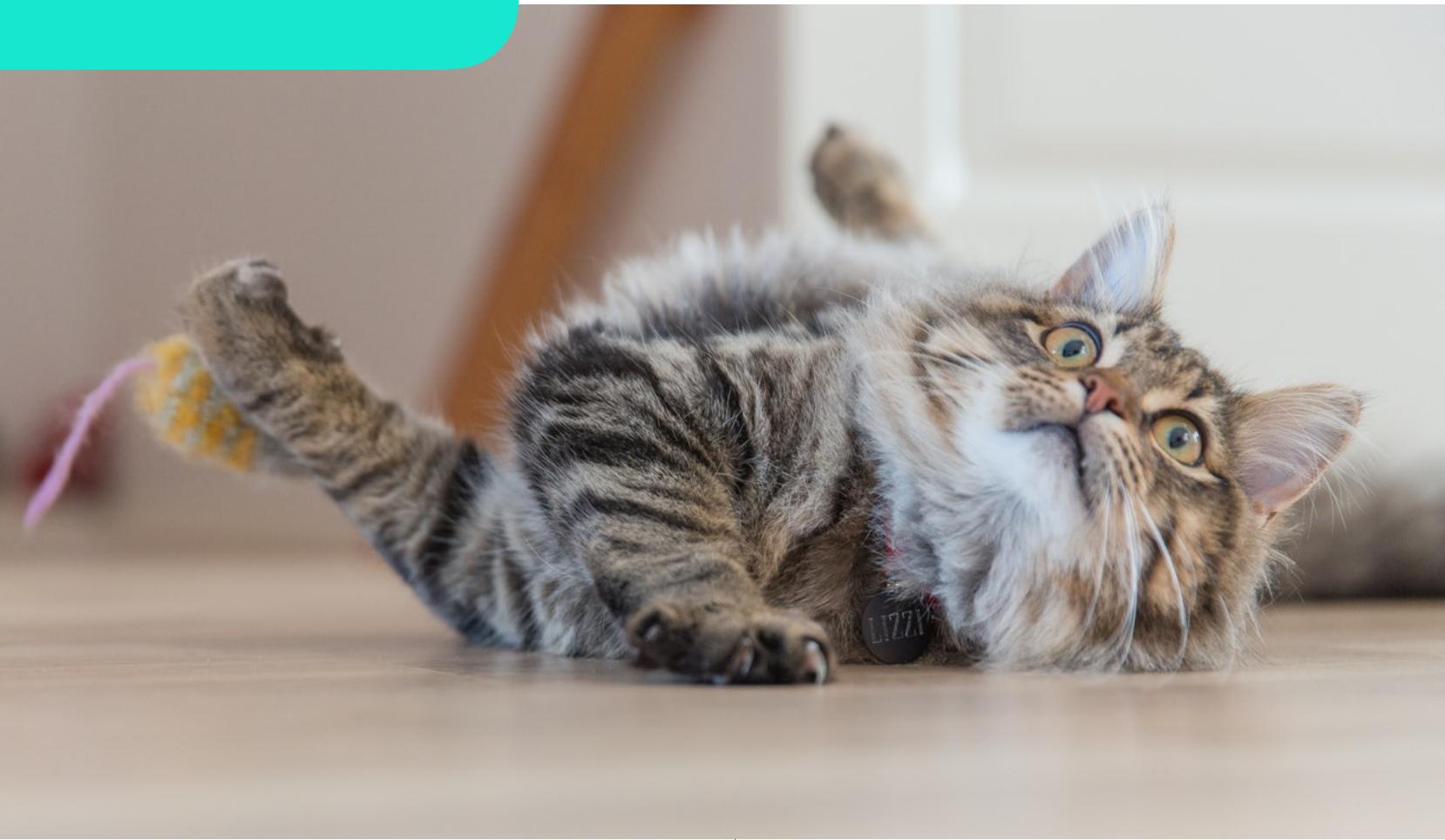
license

CC-BY-SA 4.0

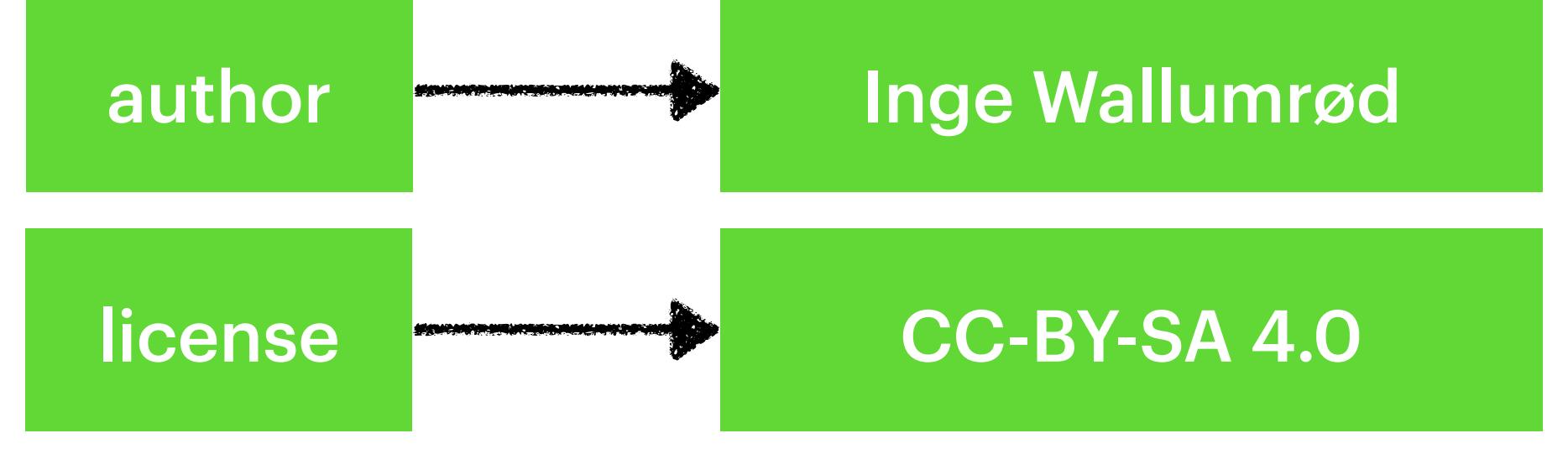
↓ SHA 256
**600b244925ffc9665c8544083b9cc002
48530f6c7b0ecdd5c89c859e3c5818cf**

600b244925ffc9665c8544083b9cc0024853
0f6c7b0ecdd5c89c859e3c5818cf
author:Inge Wallumrød
license:CC-BY-SA 4.0

Data



Metadata



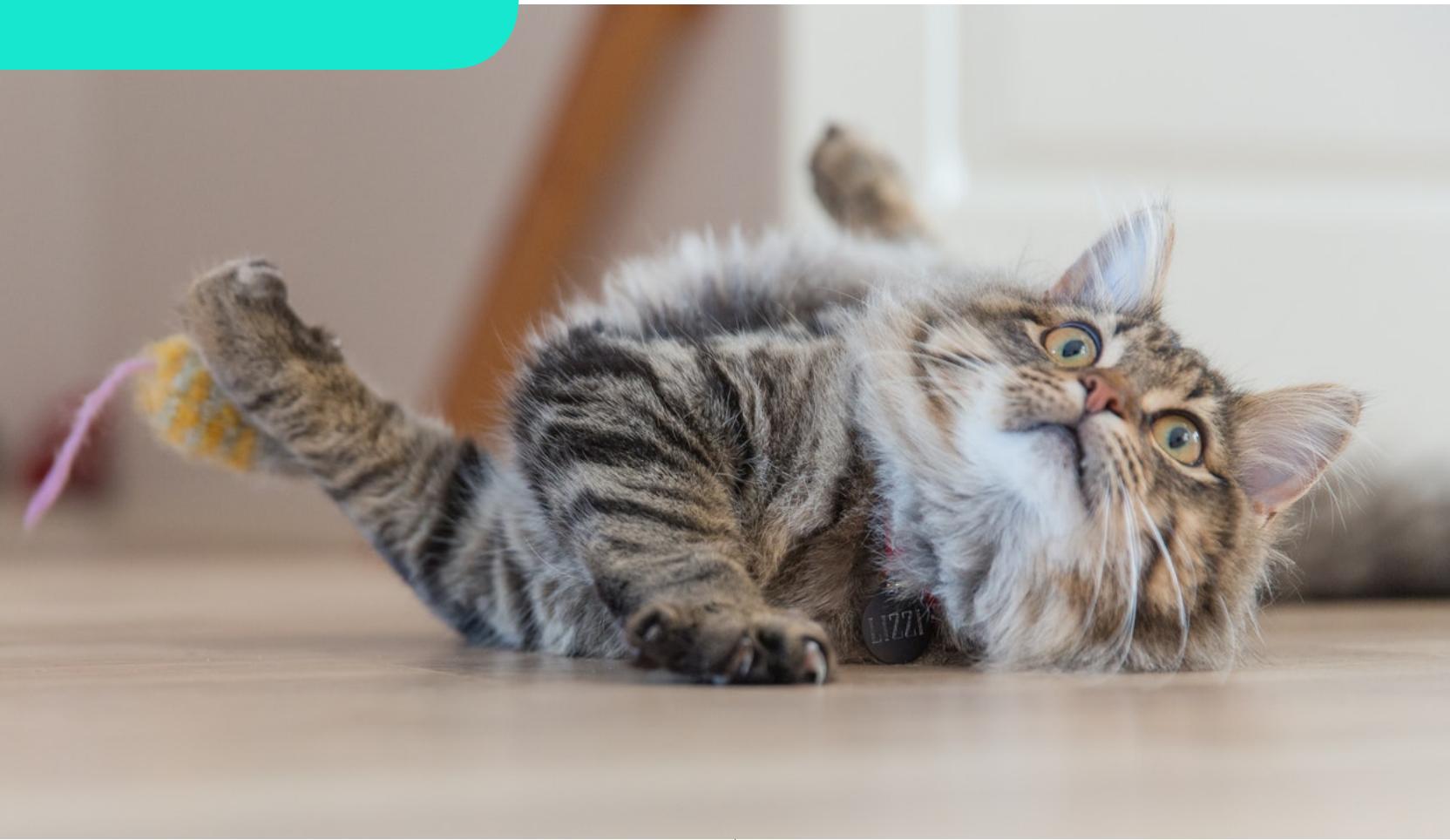
↓ SHA 256
**600b244925ffc9665c8544083b9cc002
48530f6c7b0ecdd5c89c859e3c5818cf**



**600b244925ffc9665c8544083b9cc0024853
0f6c7b0ecdd5c89c859e3c5818cf**
author:Inge Wallumrød
license:CC-BY-SA 4.0

↓ SHA 256
**72bf3d242a06d5831f83dcdba8079a7ef17
09f36e993d583504fc3e7026d5aa9**

Data



Metadata

author

license

Inge Wallumrød

CC-BY-SA 4.0

↓ SHA 256
**600b244925ffc9665c8544083b9cc002
48530f6c7b0ecdd5c89c859e3c5818cf**



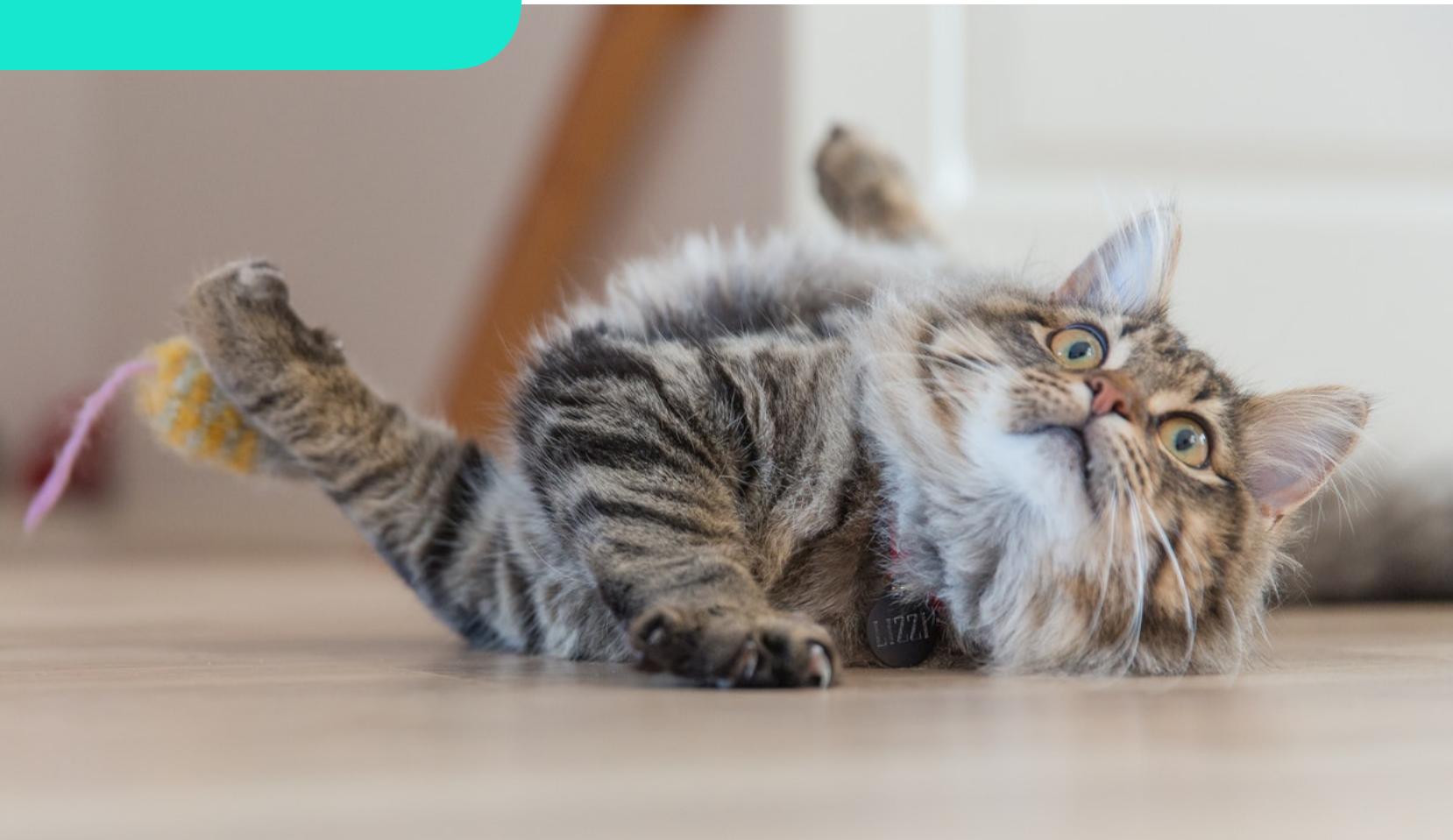
**600b244925ffc9665c8544083b9cc0024853
0f6c7b0ecdd5c89c859e3c5818cf
author:Inge Wallumrød
license:CC-BY-SA 4.0**



**72bf3d242a06d5831f83dcdba8079a7ef17
09f36e993d583504fc3e7026d5aa9**



Data



Metadata

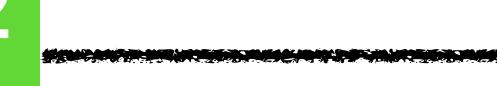
author

license

Inge Wallumrød

CC-BY-SA 4.0

↓ SHA 256
**600b244925ffc9665c8544083b9cc002
48530f6c7b0ecdd5c89c859e3c5818cf**

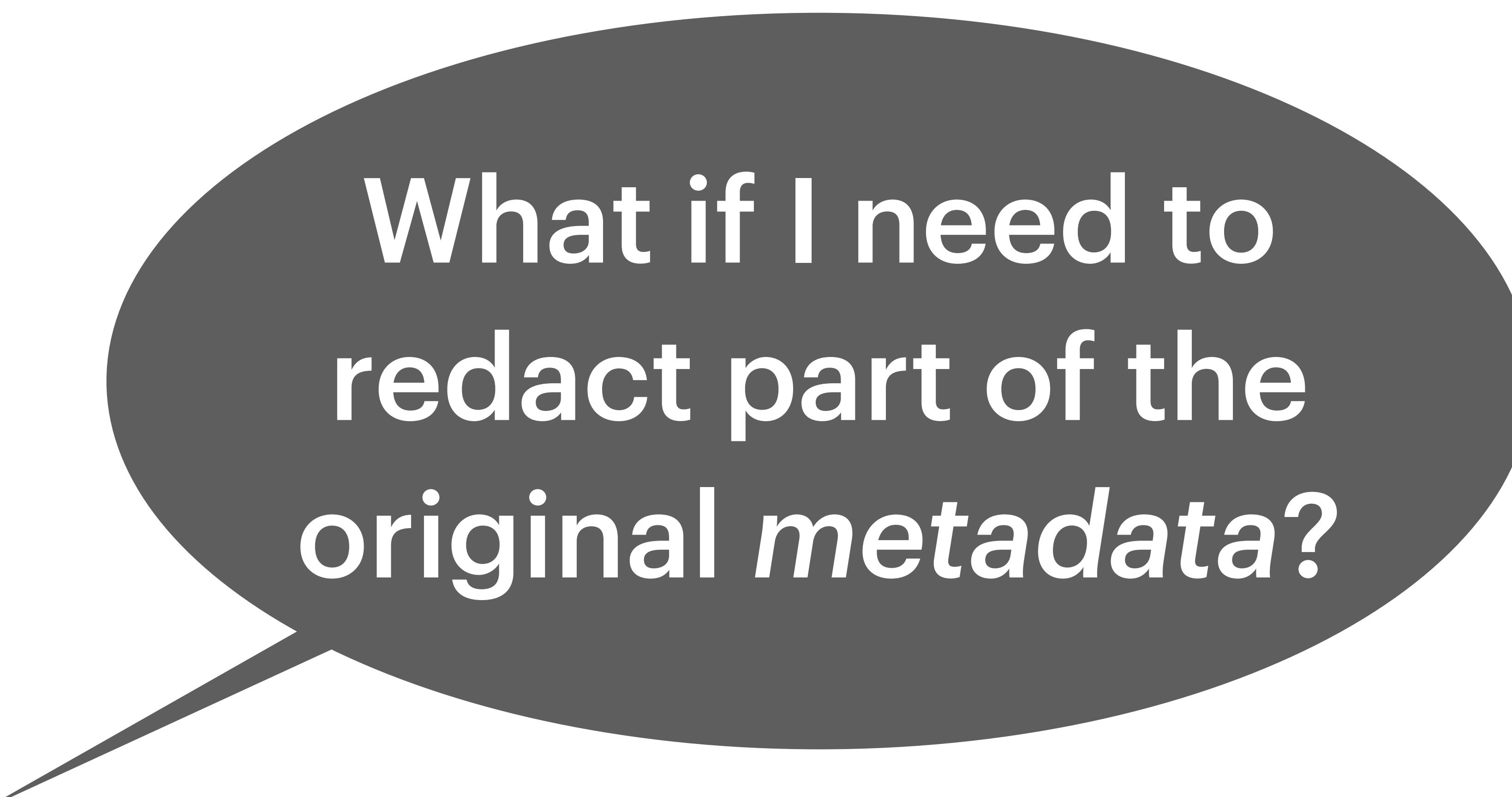


**600b244925ffc9665c8544083b9cc0024853
0f6c7b0ecdd5c89c859e3c5818cf**
author:Inge Wallumrød
license:CC-BY-SA 4.0

↓ SHA 256
**72bf3d242a06d5831f83dcdba8079a7ef17
09f36e993d583504fc3e7026d5aa9**

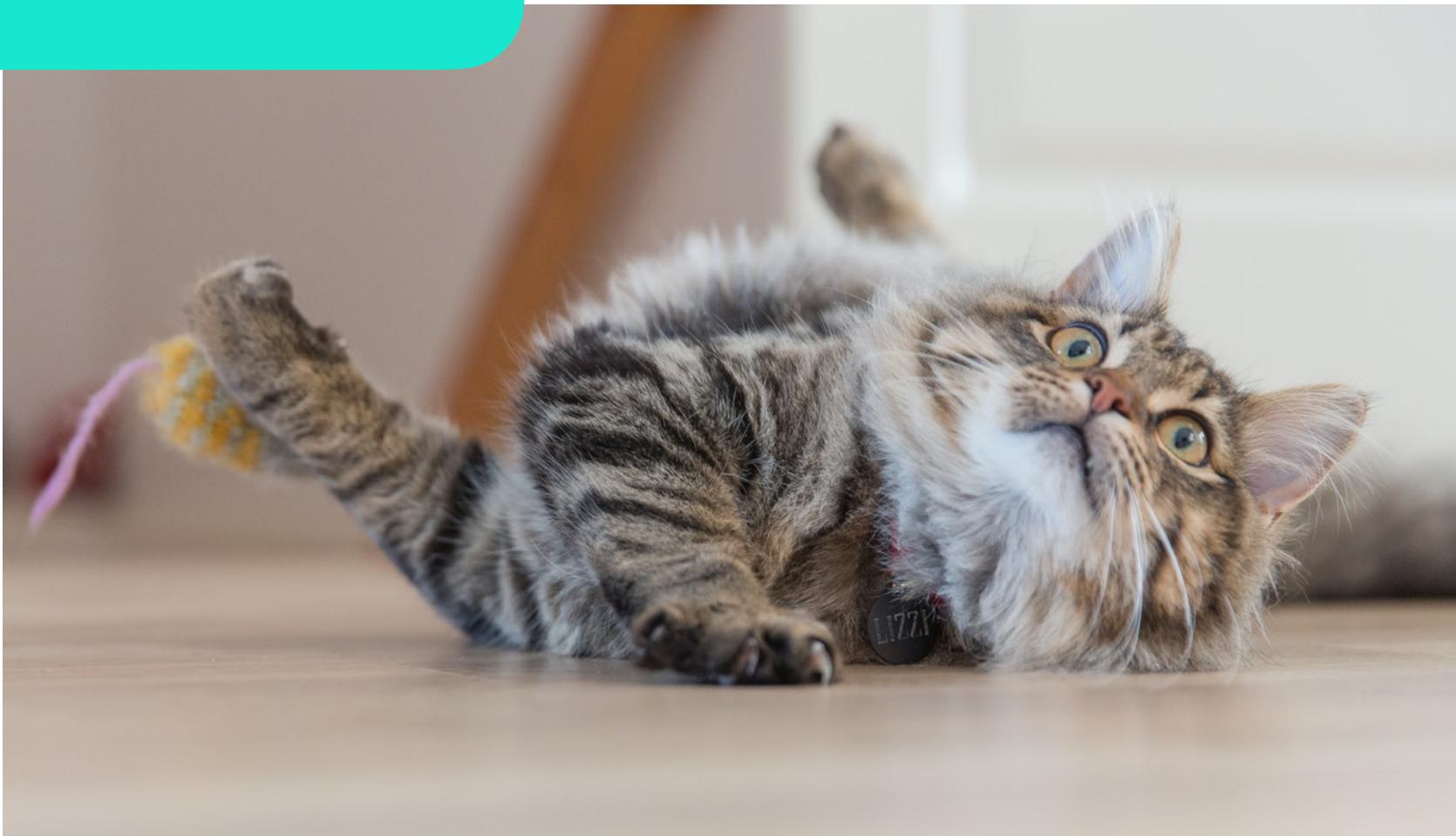
Integrity





What if I need to
redact part of the
original *metadata*?

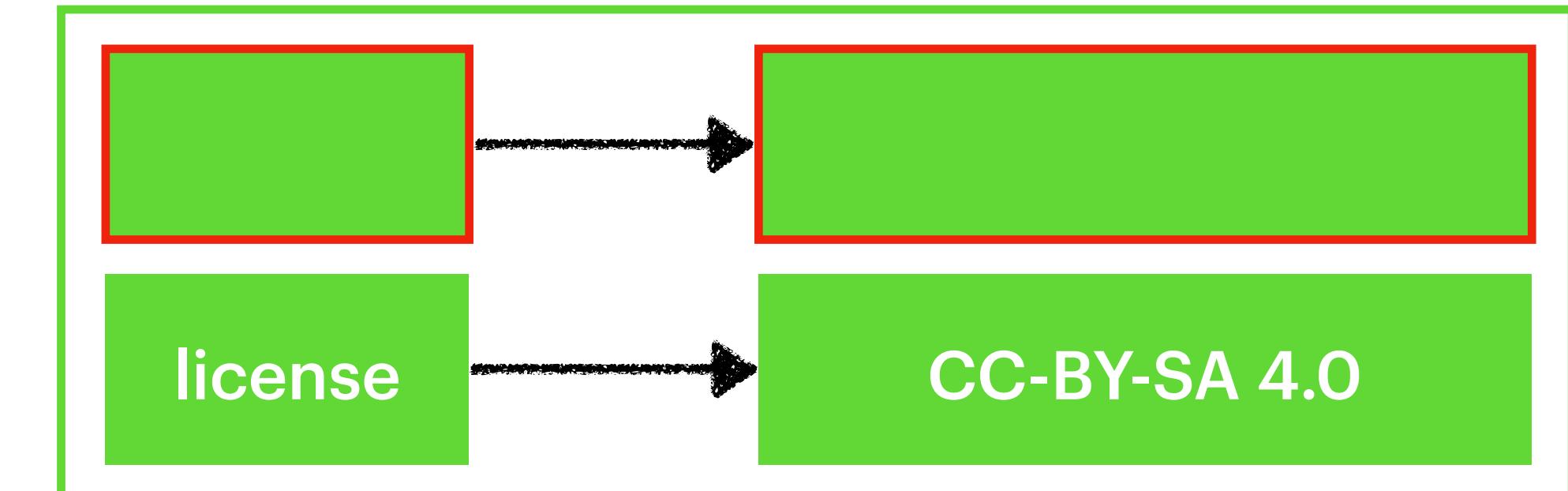
Data



 SHA 256

600b244925ffc9665c8544083b9cc002
48530f6c7b0ecdd5c89c859e3c5818cf

Metadat



600b244925ffc9665c8544083b9cc0024853
Of6c7b0ecdd5c89c859e3c5818cf
license:CC-BY-SA 4.0

Integrity

SHA 256

- 72bf3d242a06d5831f83dcdba8079a7ef17
- 09f36e993d583504fc3e7026d5aa9

ECDSA + Public Key



Metadata

author

Inge Wallumrød

license

CC-BY-SA 4.0

600b244925ffc9665c8544083b9cc0024853
0f6c7b0ecdd5c89c859e3c5818cf
author:Inge Wallumrød

600b244925ffc9665c8544083b9cc0024853
0f6c7b0ecdd5c89c859e3c5818cf
license:CC-BY-SA 4.0

Integrity

SHA 256

SHA 256

0abf3d242a0cdba8079a7ef1709f36e993d
583504fc3e7026d5aadf6d5831f83

45a7ef172bf3d242a0e55831f83dcdb807
09f36e993d583504fc3e7026d5ab9

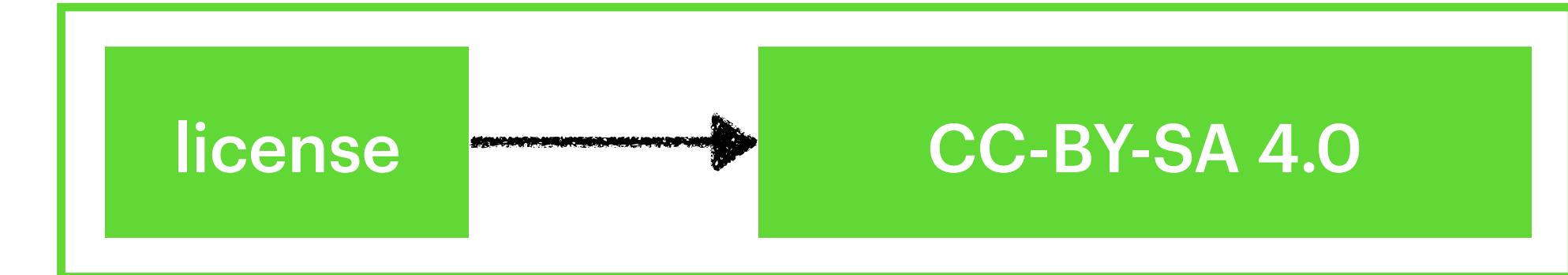
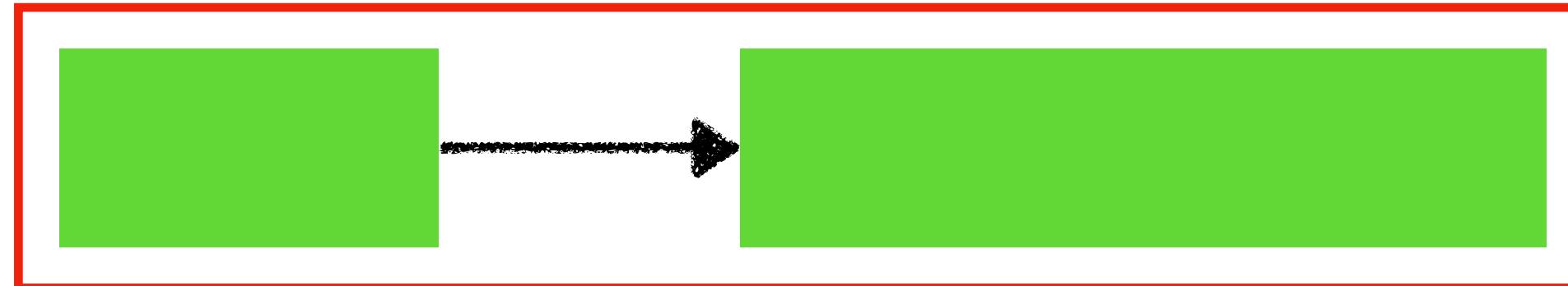
ECDSA + Public Key



ECDSA + Public Key



Metadata



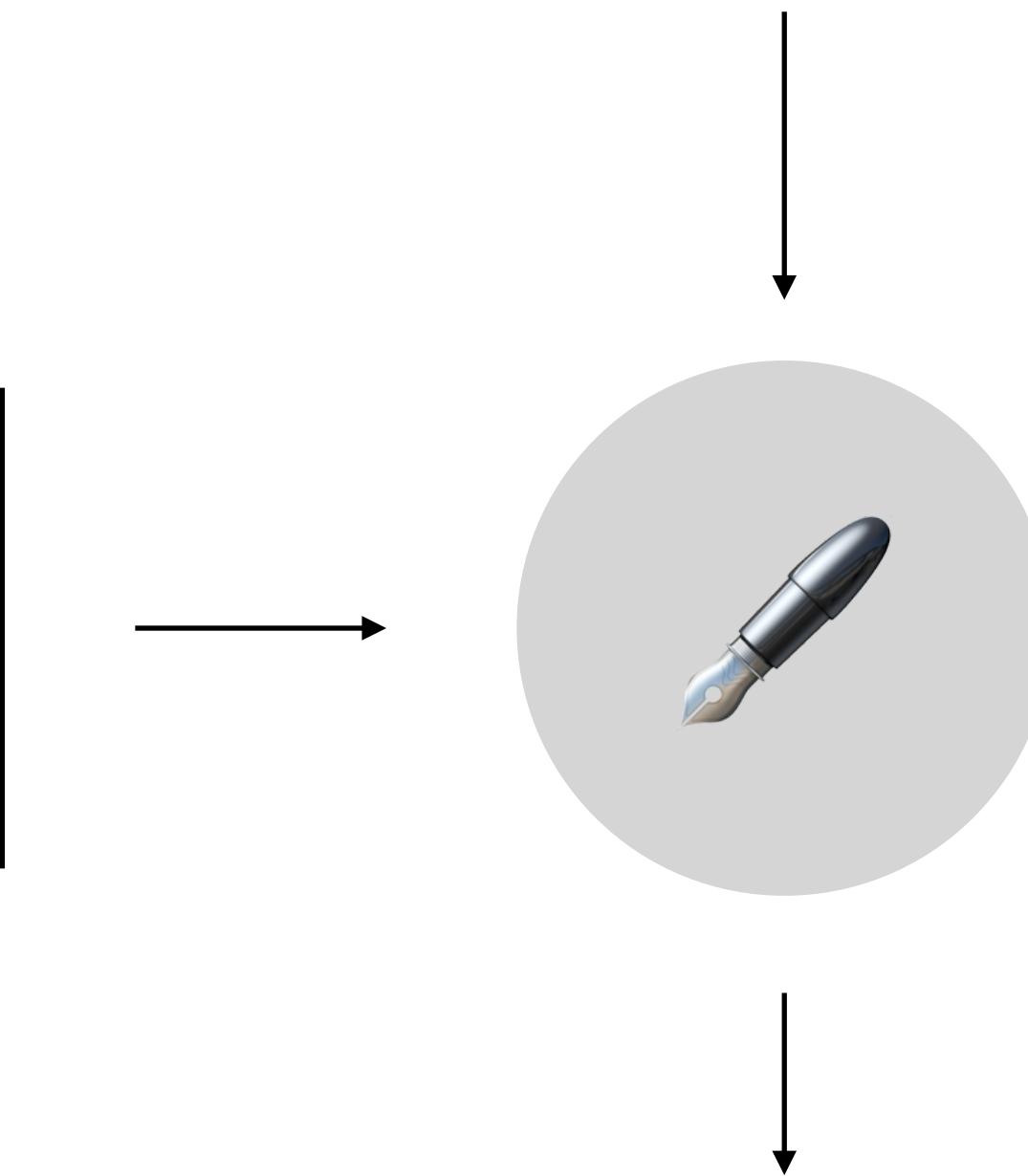
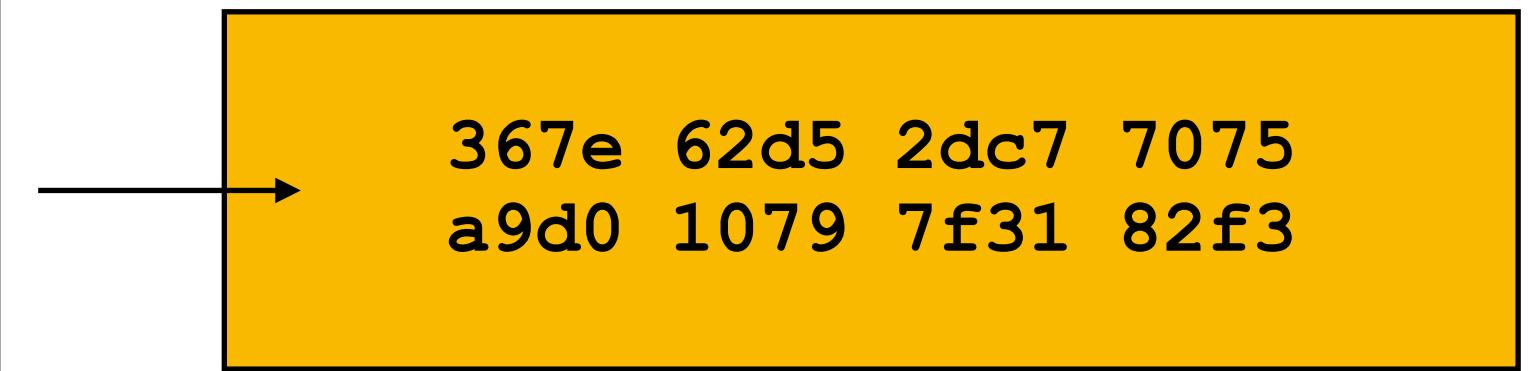
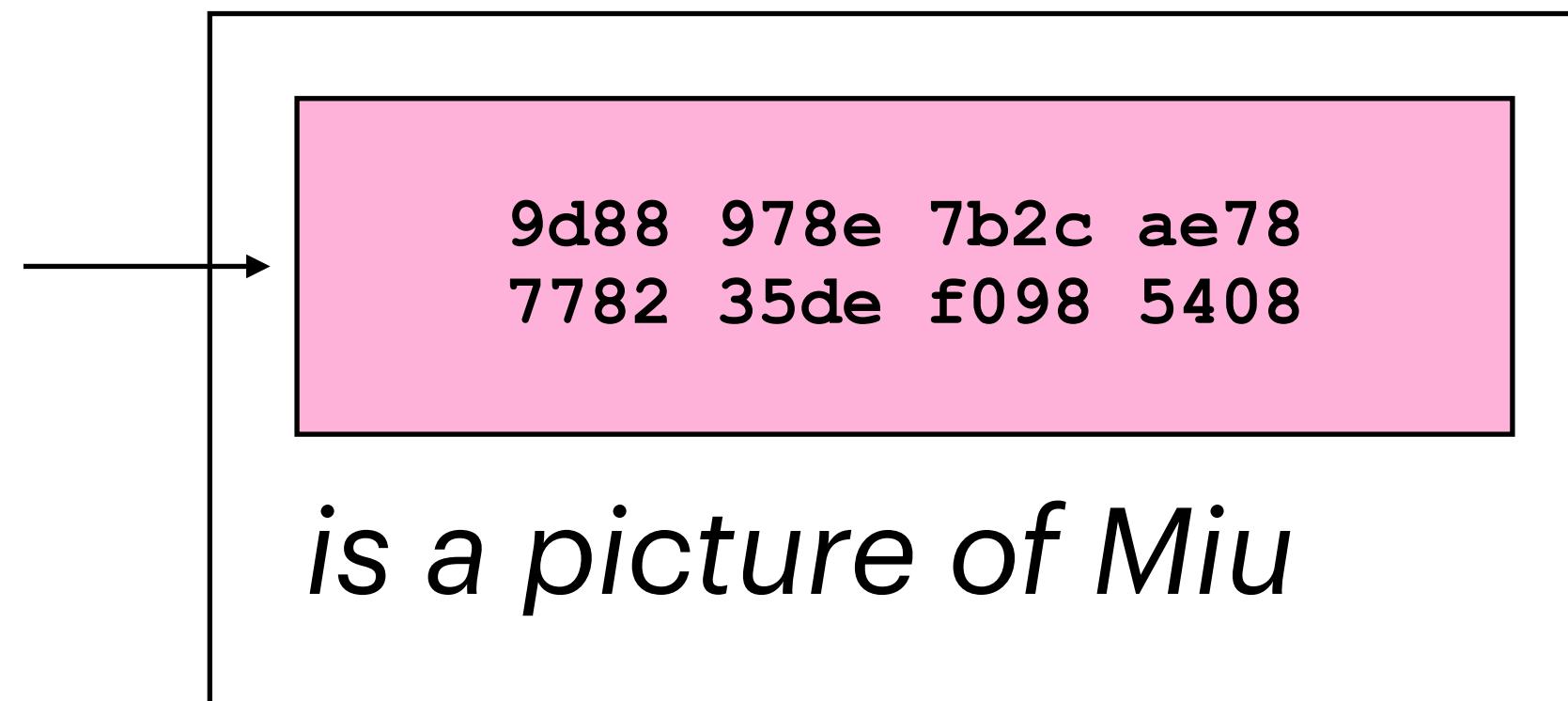
600b244925ffc9665c8544083b9cc0024853
0f6c7b0ecdd5c89c859e3c5818cf
license:CC-BY-SA 4.0

45a7ef172bf3d242a0e55831f83dcdba807
09f36e993d583504fc3e7026d5ab9

ECDSA + Public Key



Integrity



9d88 978e 7b2c ae78
7782 35de f098 5408

is a picture of Miu

 *Josh*

9d88 978e 7b2c ae78
7782 35de f098 5408

depicts a cat

 *Kate*

9d88 978e 7b2c ae78
7782 35de f098 5408

*is captured on
August 5*

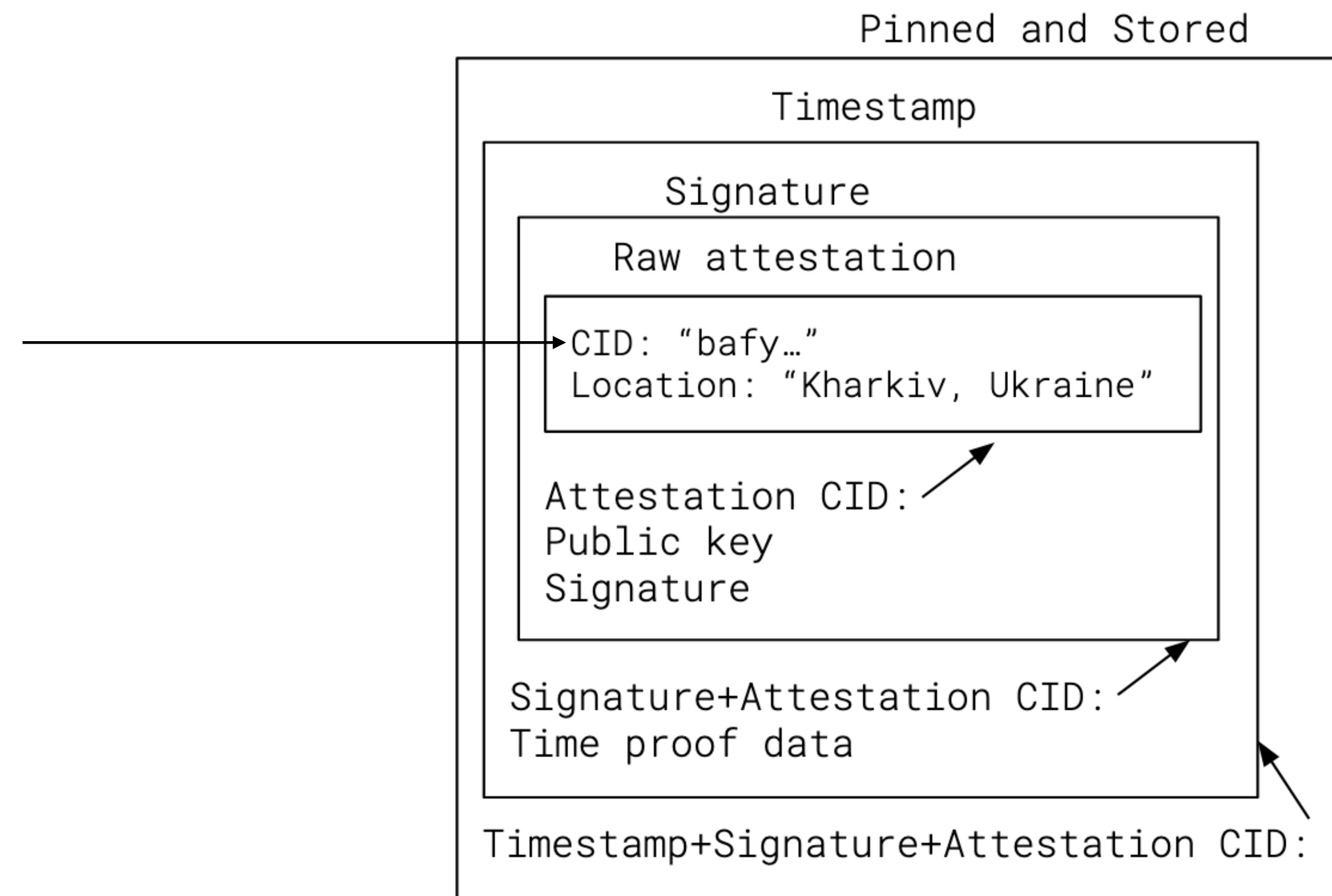
 *Lexa*

9d88 978e 7b2c ae78
7782 35de f098 5408

was born in Paris

 *Josh*

Authenticated attributes



FRAMEWORK

1 CAPTURE

▲ DATE

2022-11-02T11:17:20.231Z

▲ CAPTURED USING

WEBRECODER - ARCHIVEWEB.PAGE / AUTHENTICATED WEB ARCHIVE

▲ LOCATION

LOS ANGELES, CA, UNITED STATES

▲ REGISTERED ON

▲ OPENTIMESTAMPS

C71E97B66F9C148C2EA36117ECCD97F57D976B50942D6B7DA55CC1D26F7DD42A ↗

▲ NUMBERS

0X014AC7FA821E3C6AB124953F6724AFBE00DC03F7CF0C1EBB174D186F1512DEE7 ↗

▲ AVALANCHE

0XD7C191494B5726D784CF75019B8A08B9349FE4A26667419A71C65DDA184EB9D5 ↗

▲ ISCN

E408581741959151FA857D2D2D61D615E593A65BDDB34A92197EAD52872ECC1C ↗

LEDGERS

OPENTIMESTAMPS

NUMBERS

AVALANCHE

ISCN

IPFS

FILECOIN

STORY

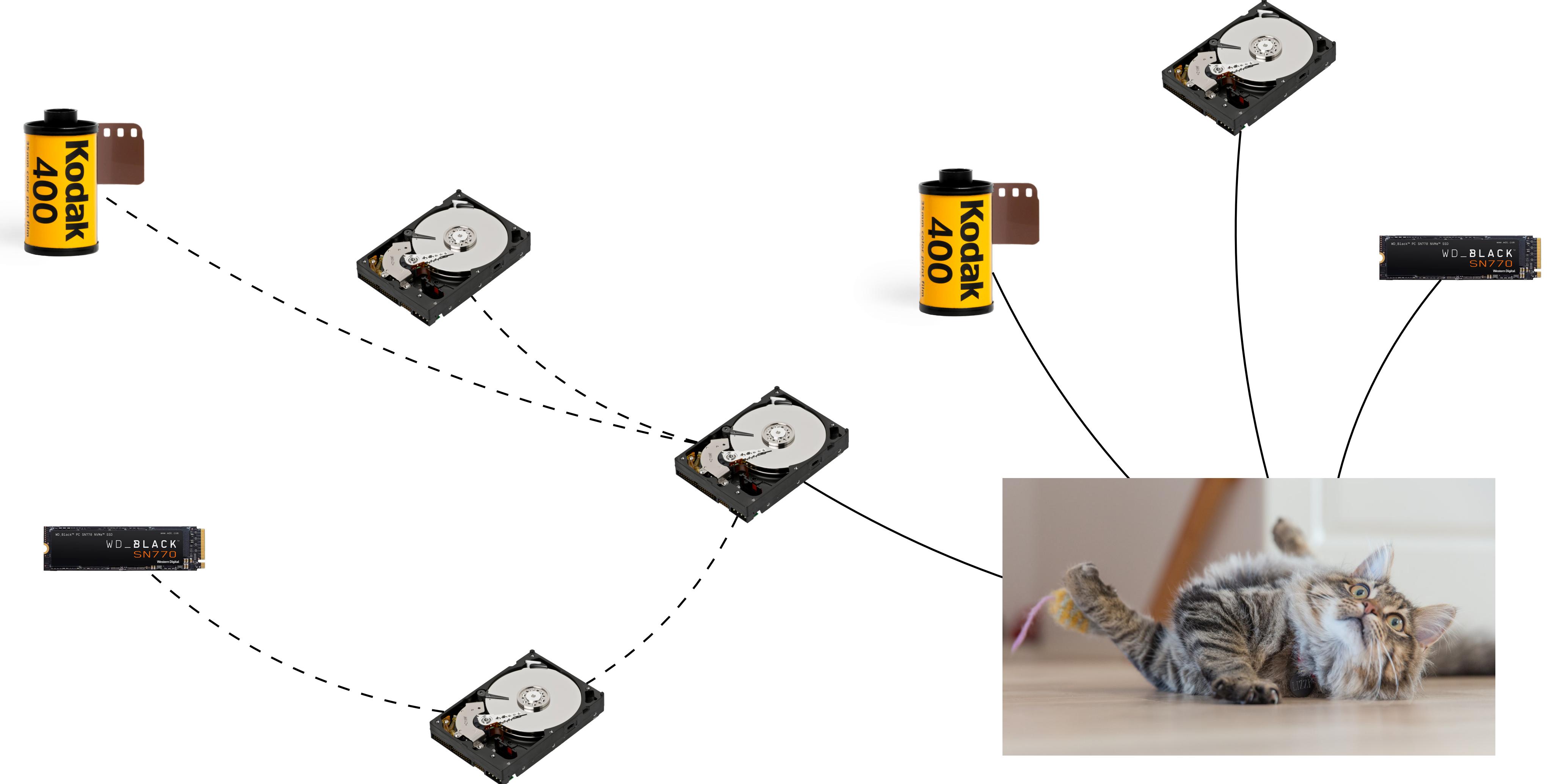
9d88 978e 7b2c ae78
7782 35de f098 5408

*is registered on the
Ethereum blockchain*



Starling Lab

Store



9d88 978e 7b2c ae78
7782 35de f098 5408

is stored on IPFS



Starling Lab

Verify

Investigate
the claims
made by the
image and
metadata

Uwazi English ⚙️

Demo wacz

WACZ

2.68 MiB

← → C search://view=pages

Search by Page URL, Title or Text

Date▲	Page Title
6/9/2022 2:21:09 PM	(2) FrustratedWithTime on Twitter: "A small rundown of schools getting shot, bombed, on fire... 7 days of war, at least 7 schools.. Kharkiv "invasion" or "Massacre"... Kharkiv School №17, №42, №75, №96, №108, №118, №134.. #KharkivMassacre #OSINT https://t.co/5ZazMkt6Ui " / Twitter https://mobile.twitter.com/FrustratedTime/status/150003105222769459

1 Pages Found

CID bafybeifgkpgb7yqqjnovszaio7tzetmdfmigylr24hg6a76wnjxchnhx54

EDIT VIEW ADDITIONAL METADATA GRAPH DELETE

mime application/wacz

name Web archive

asset bafybeifgkpgb7yqqjnovszaio7tzetmdfmigylr24hg6a76wnjxchnhx54

pages {"XrSVY9": "https://mobile.twitter.com/FrustratedTime/status/150003105222769459"}

author {"name": "FrustratedWithTime", "url": "https://twitter.com/FrustratedTime"}
Timestamping Proof: [View](#)

zipcid bafybeia... 652li5msjgfohl6a

Parents Verified
bafybeihue7rxlqvnyxd2xxrioa2mgwk6vn2ep...

zipname 3b545bf8c25d1a2881987845806ebf67a44959a8
7be3cc45c3362b3d8763e680.zip

filename 5170a615c4add2bc999101d944c1fc94fd14a
ad7c0a0f0a92ca627344128c.wacz

software

UWAZI | Uwazi | Library | Settings

**How does data integrity and reliable
attribution support the work of verification?**