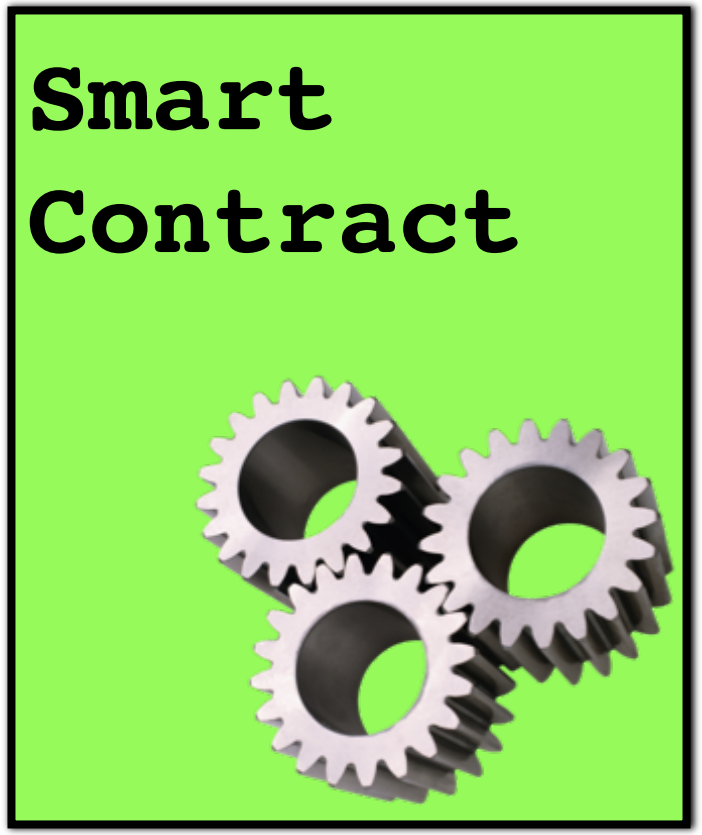
A privacy-preserving oracle for TLS

Fan Zhang, Deepak Maram, Harjasleen Malvai, Steven Goldfeder, Ari Juels

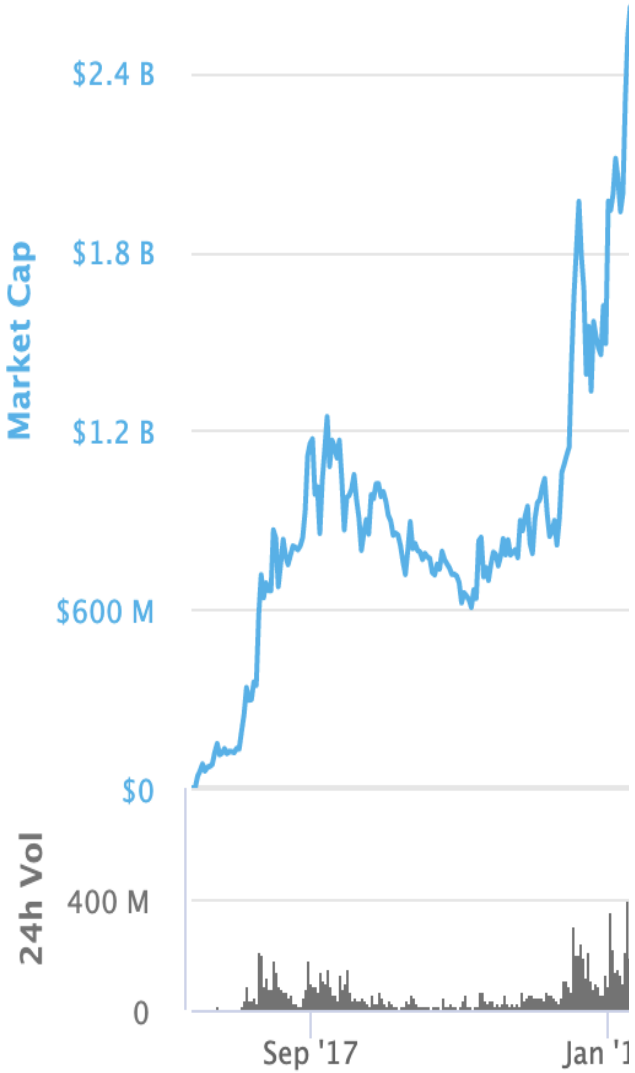
CORNELL TECH
HOME OF THE JACOBS INSTITUTE

IC3 The Initiative For CryptoCurrencies & Contracts

Key application of DECO



Tokens



Floyd 'Crypto' Mayweather promotes an ICO, again

Share on Facebook | Share on Twitter

IMAGE: AP/REX/SHUTTERSTOCK

777.3k likes 23.1k comments

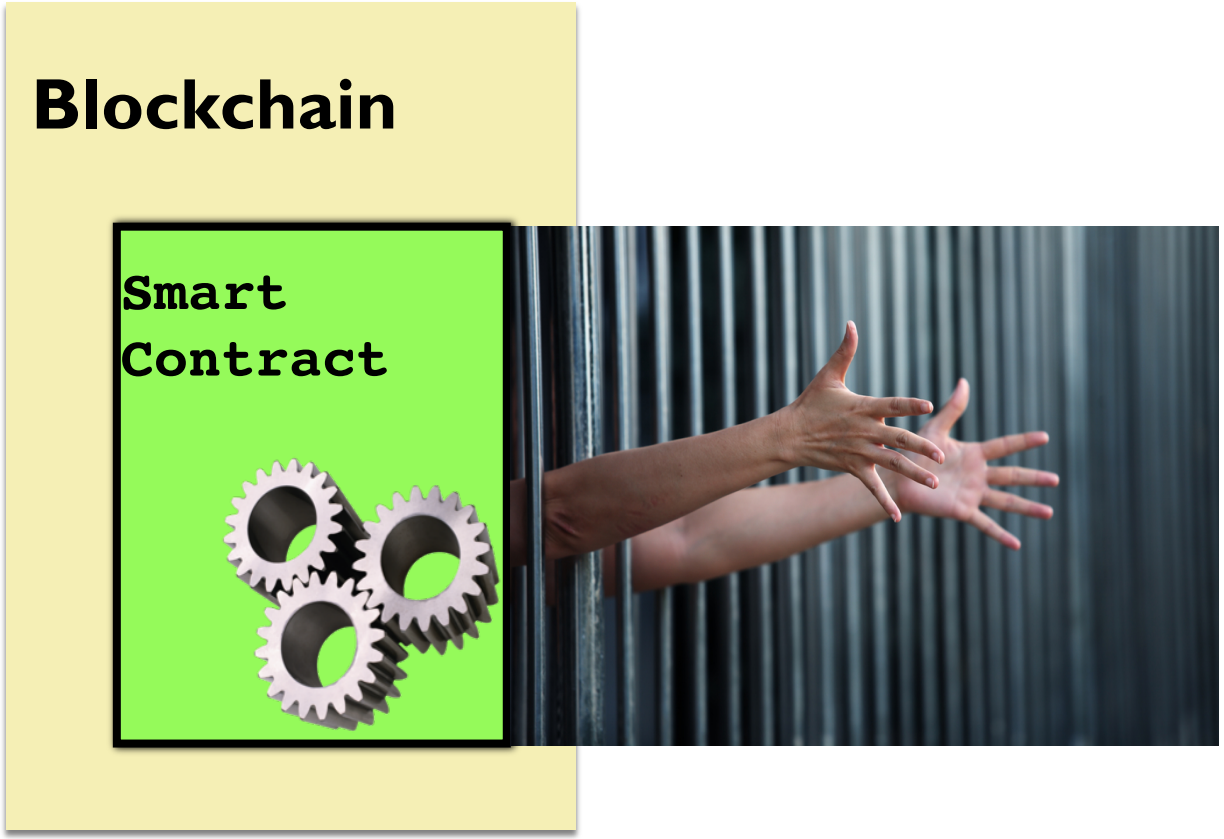
Champion Predictions: I'm gonna make a $hit t$n of money on August 26th. I'm gonna make a $hit t$n of money on August 2nd on the Stox.com ICO. #TMT #STOX #MAYWEATHER #TBE #CRYPTO #CRYPTOCURRENCY #BLOCKCHAIN #ETHEREUM #BITCOIN

JULY 27

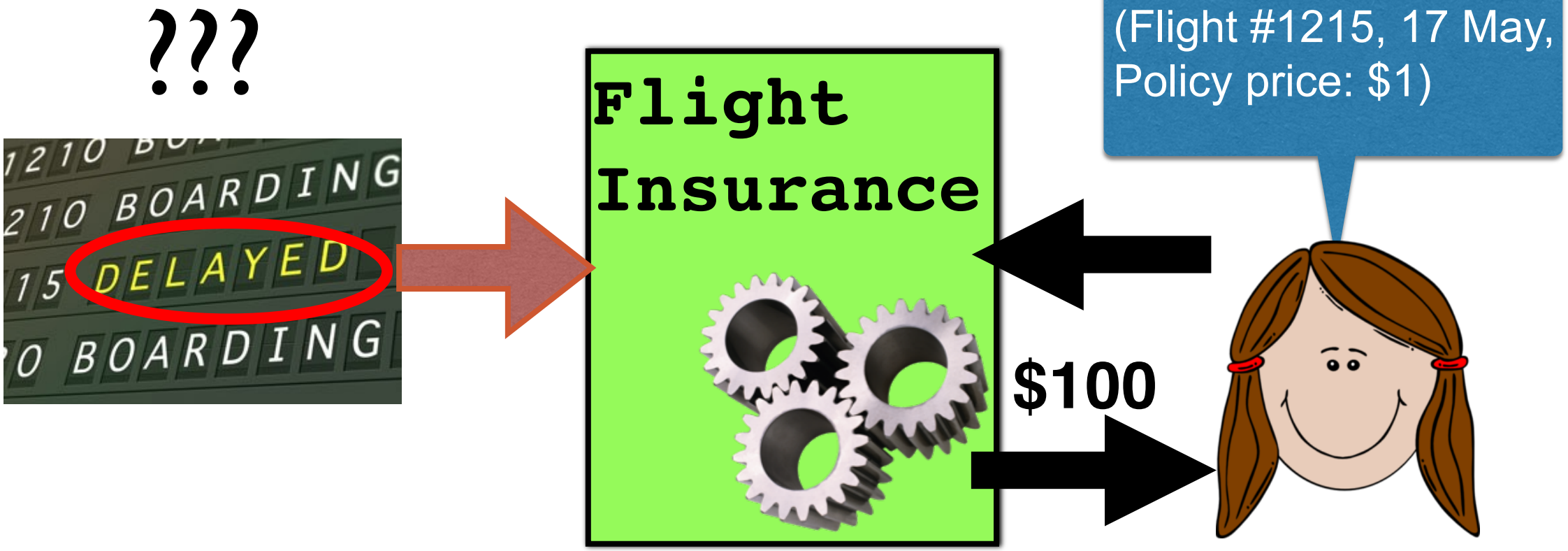
Tokens


$2.4 B
$1.8 B
$1.2 B
$600 M
$0
Market Cap
400 M
0
24h Vol
Sep '17
Jan '18
May '18
Sep '18
Jan '19
May '19
Sep '19
Jan '20

Smart contracts can't fetch real-world data!



Popular example



Solution: Oracles

Commodity prices

Webpage contents

Weather data

Stock quotes

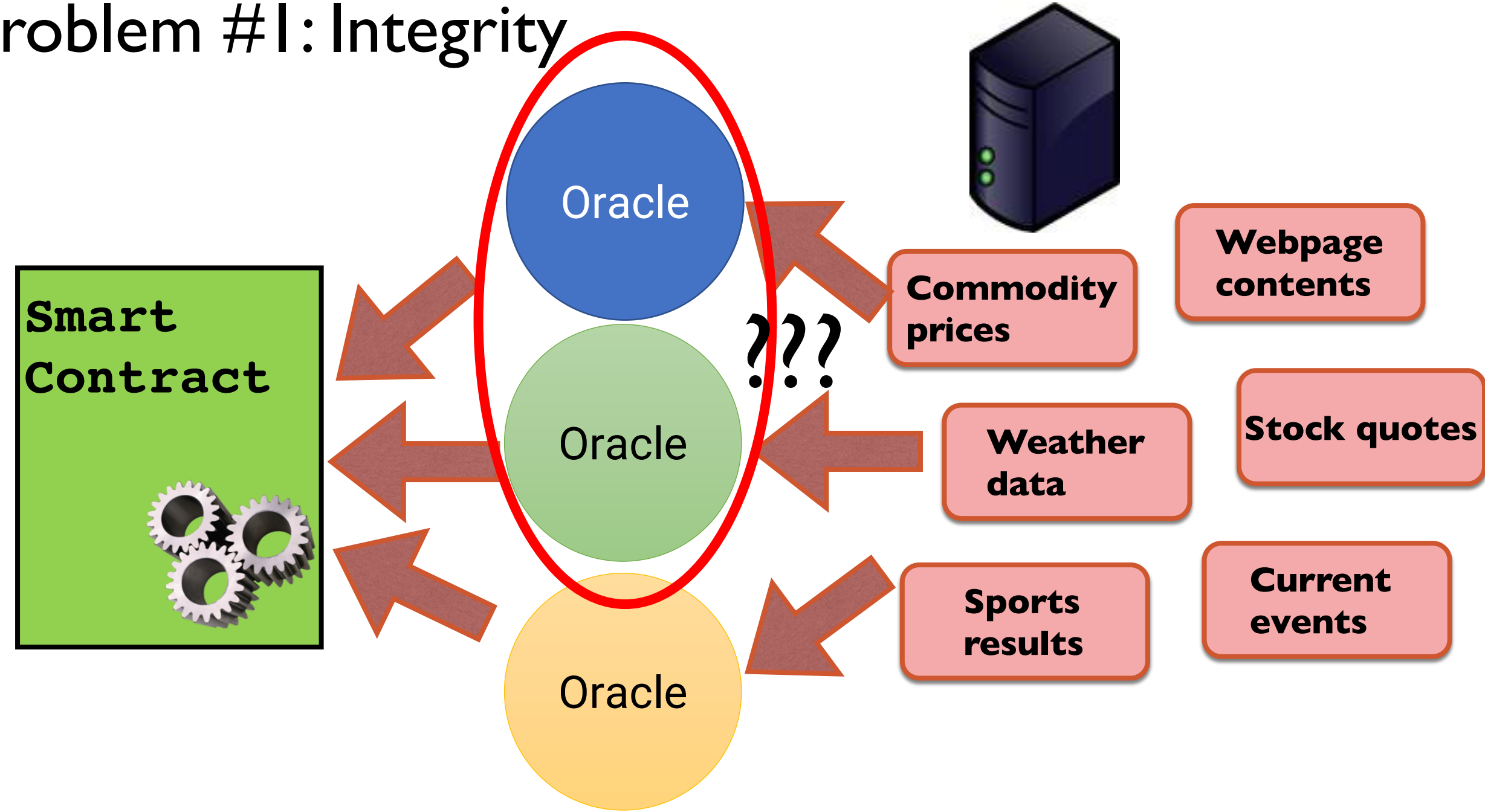
Sports results

Current events

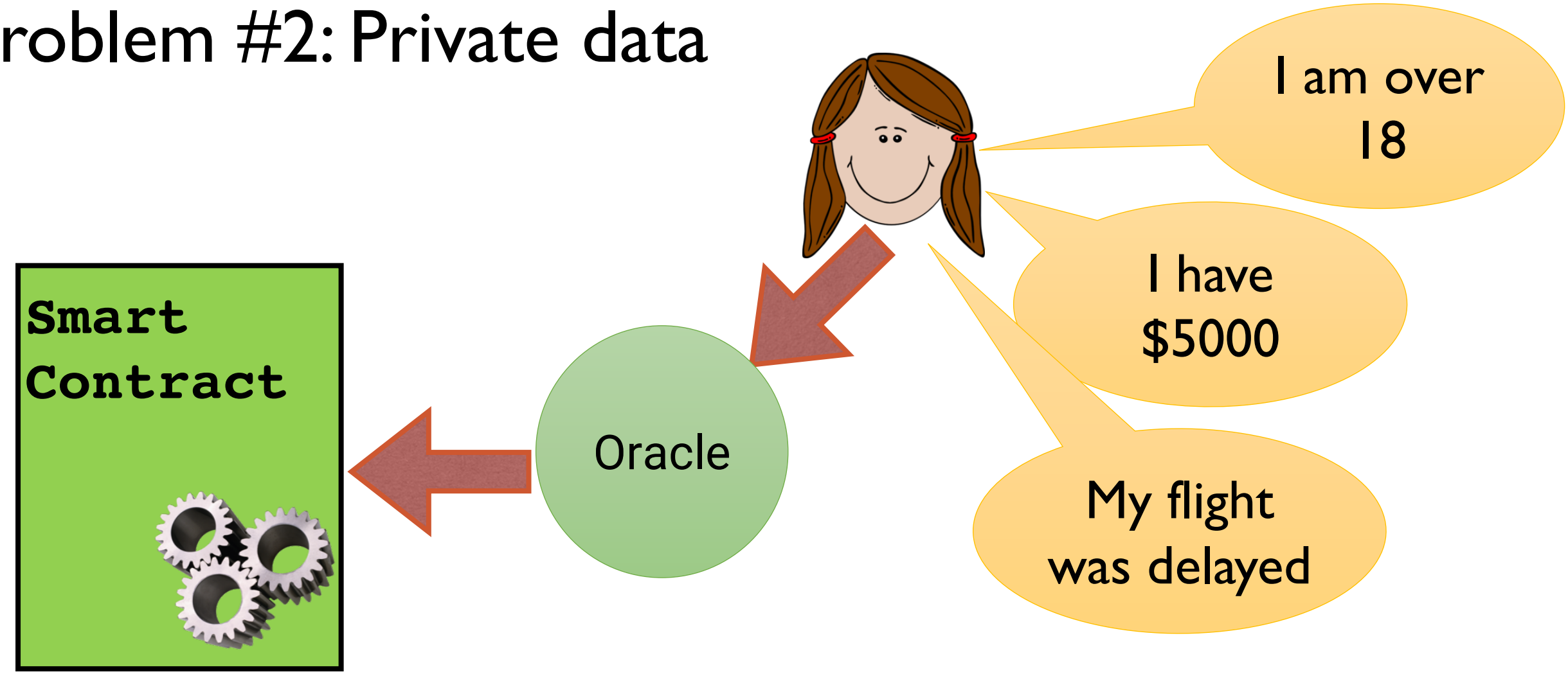
Oracle

Smart Contract

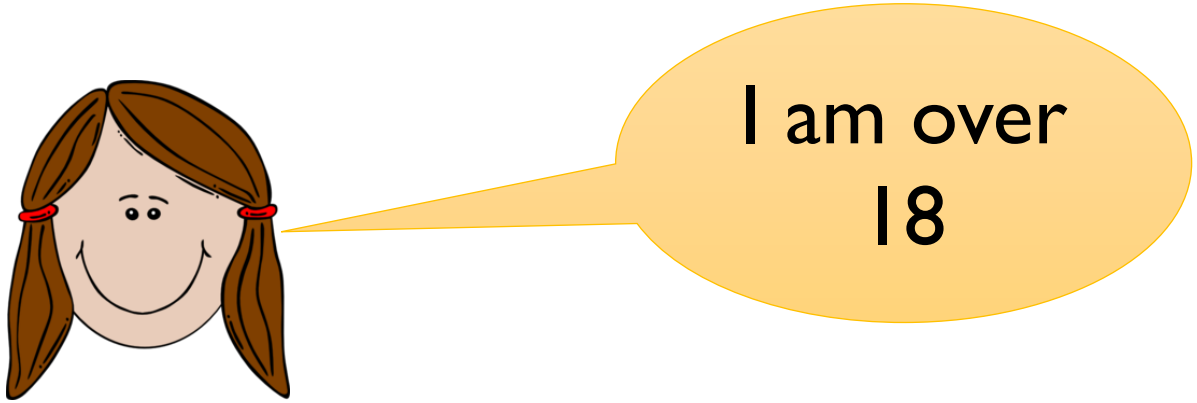
Problem #1: Integrity

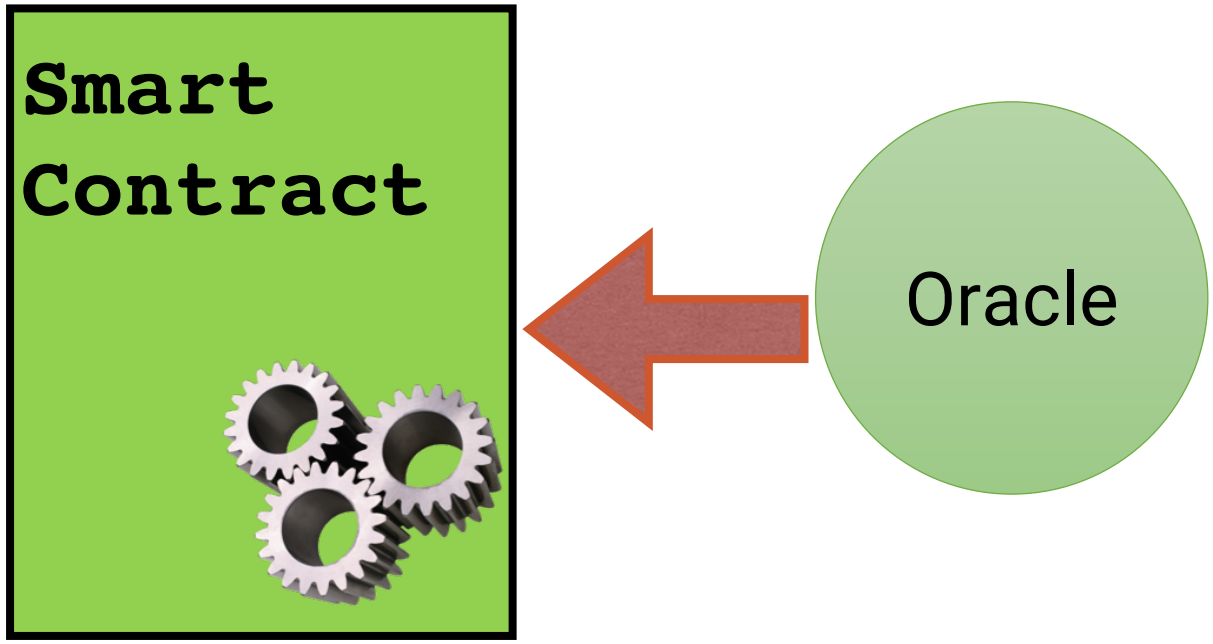


Problem #2: Private data

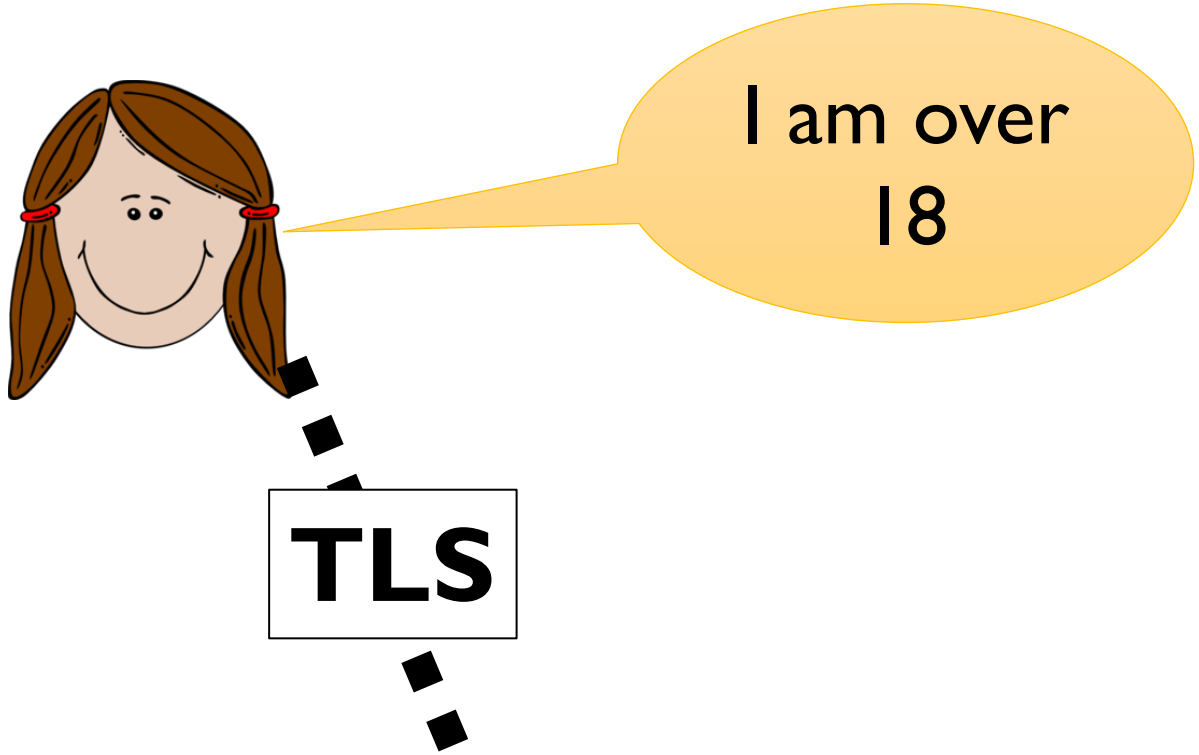


Problem #2: Private data

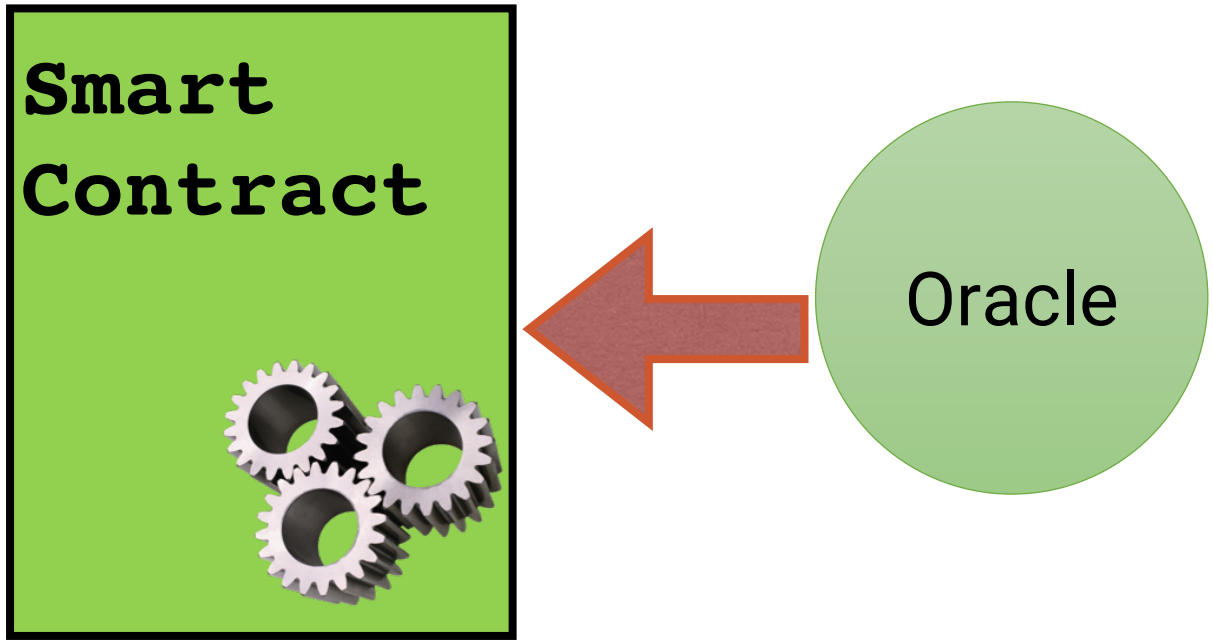




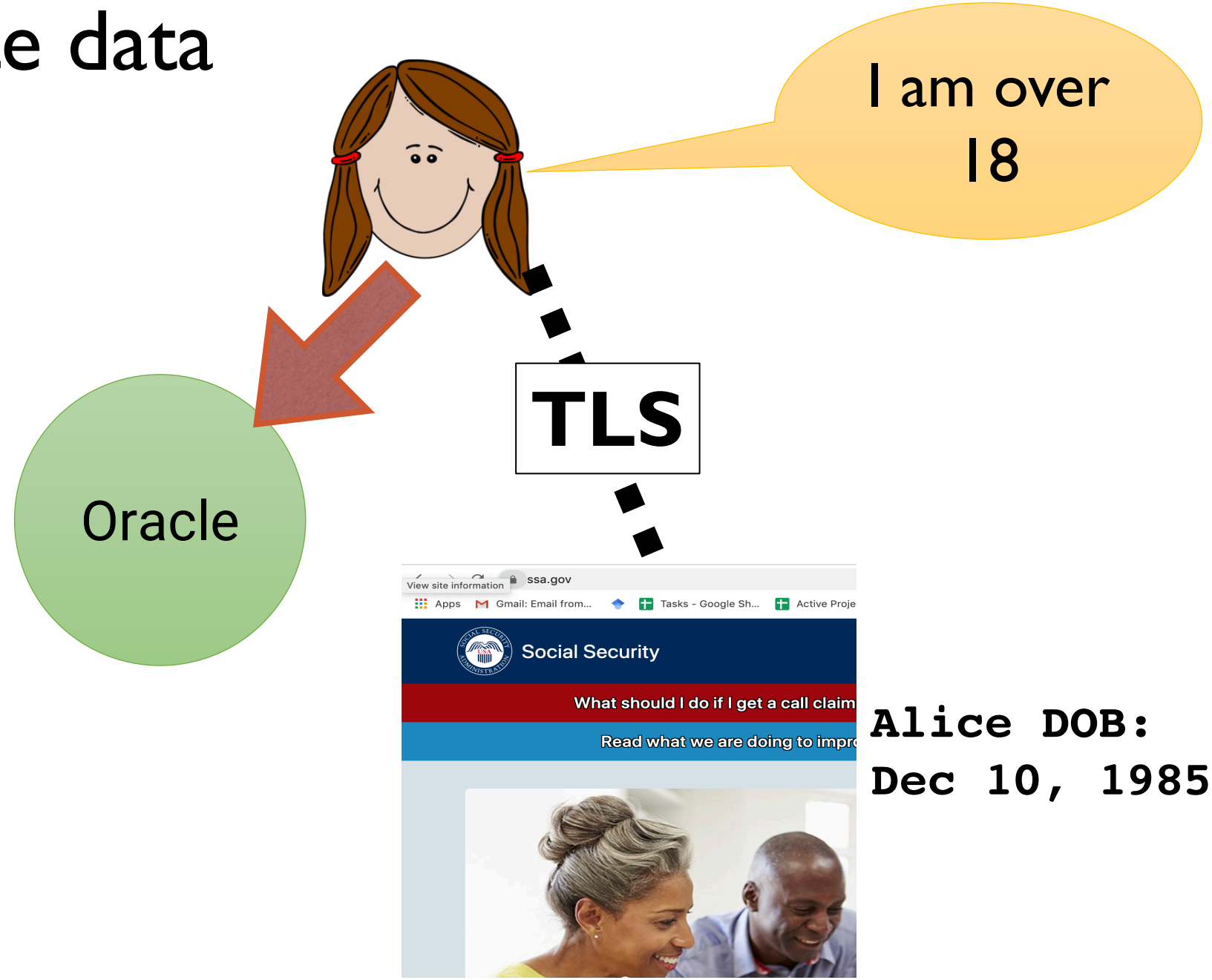
Problem #2: Private data



Alice DOB: Dec 10, 1985

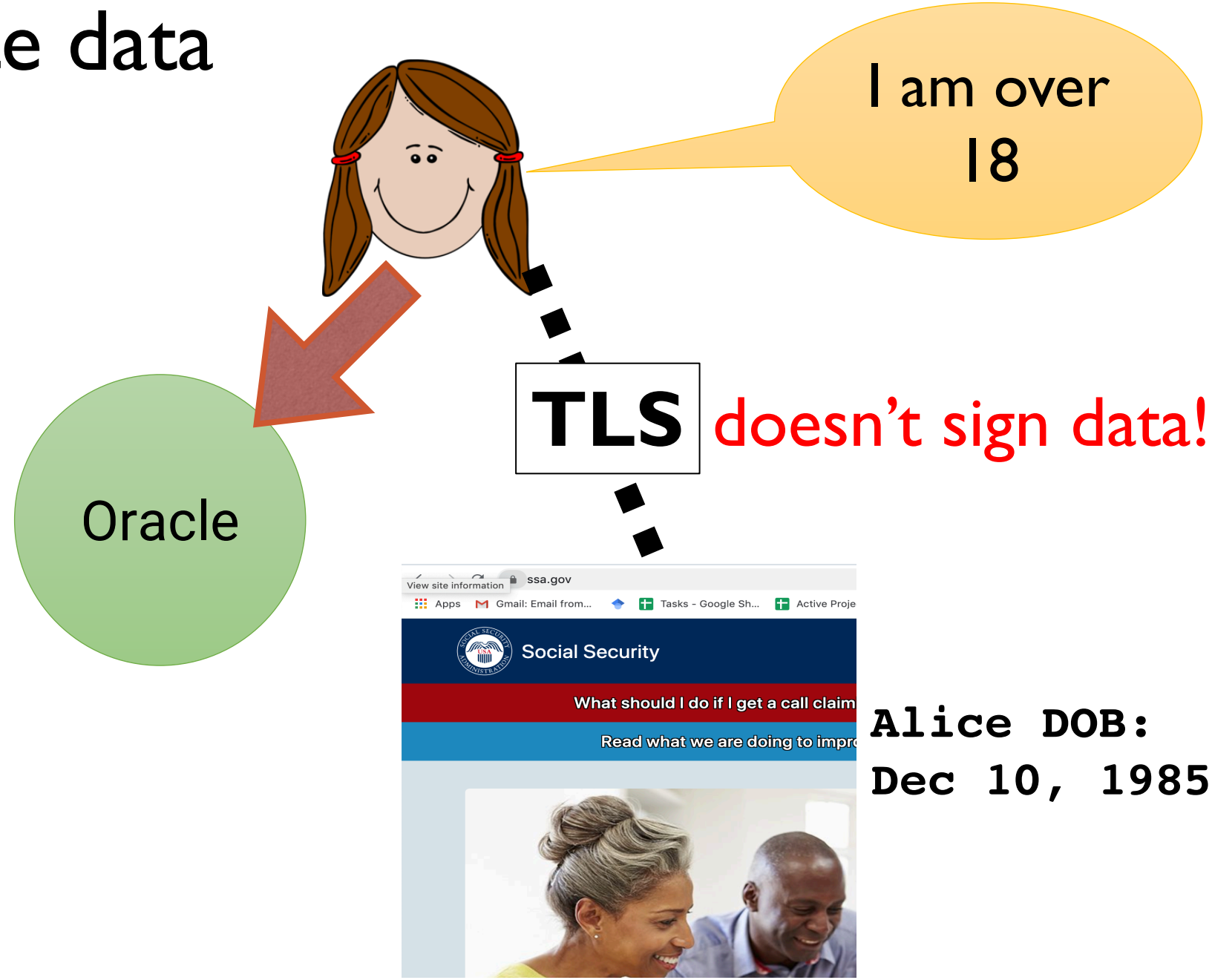


Problem #2: Private data



Alice DOB: Dec 10, 1985

Problem #2: Private data



Current approaches

- Change TLS to sign data
  - Requires adoption…
- Use Trusted Execution Environment
  - Extra trust assumption
  - Not always available

Ritzdorf, Hubert, et al. "TLS-N: Non-repudiation over TLS Enabling Ubiquitous Content Signing." In *NDSS*, 2018.

Zhang, Fan, et al. "Town Crier: An authenticated data feed for smart contracts." In *CCS*, 2016.

Introducing the DECO protocol

- Facilitates privacy-preserving proofs about TLS data to oracles
  - And thus to smart contracts
- Requires ***no*** *trusted hardware*
- Requires ***no*** *server-side modifications*
  - i.e., "transparent" to HTTPS-enabled servers
- Works with ***modern*** *TLS versions (1.2 & 1.3)*

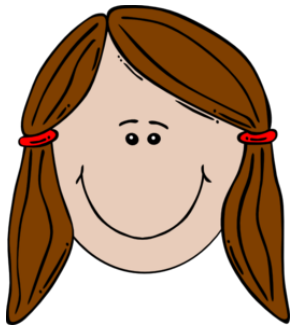
Goal and adversarial model

- Prove the provenance of TLS ciphertexts
- Decrypt or proving statements about the plaintext in ZK (e.g., bal > $5,000)

Running unmodified TLS

What's my balance?

Your bal is $8,000.



This is from my bank:

Your bal is $8,000.

Not signed by S!

Oracle

TLS server ***S***

TLS Client aka Prover

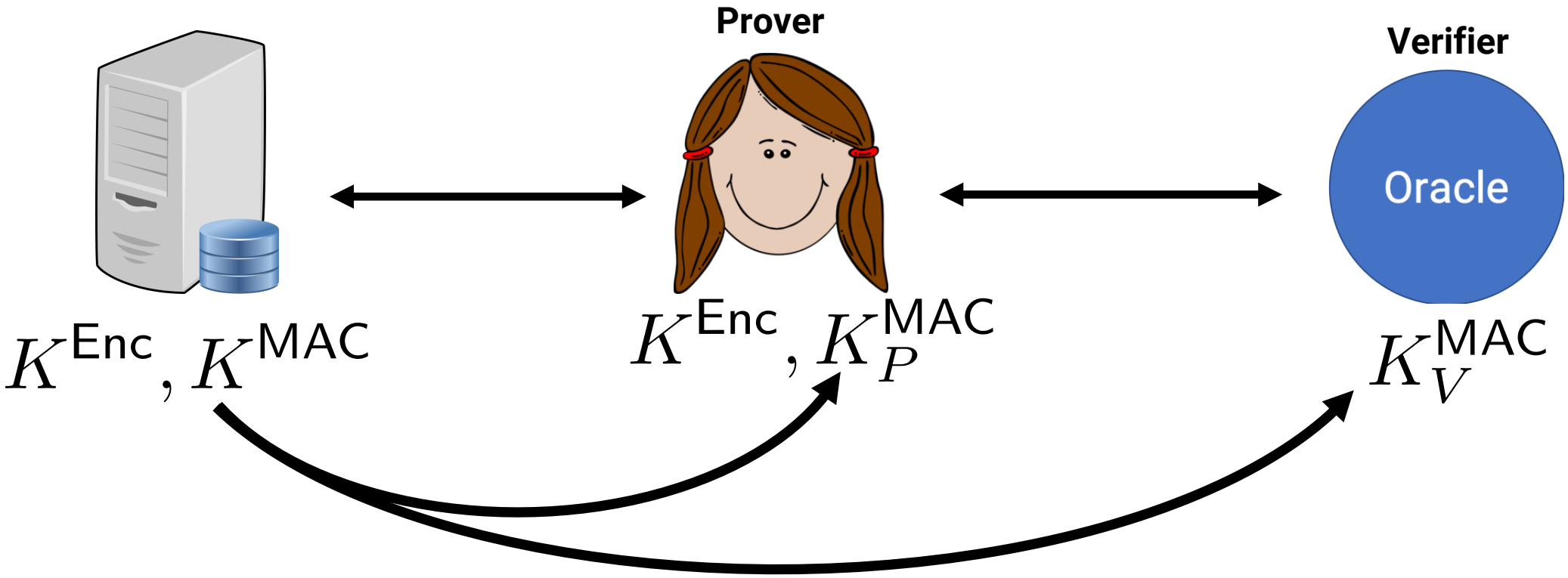
Verifier

This denotes TLS ciphertext.

Main idea: Three-party handshake

DECO logo

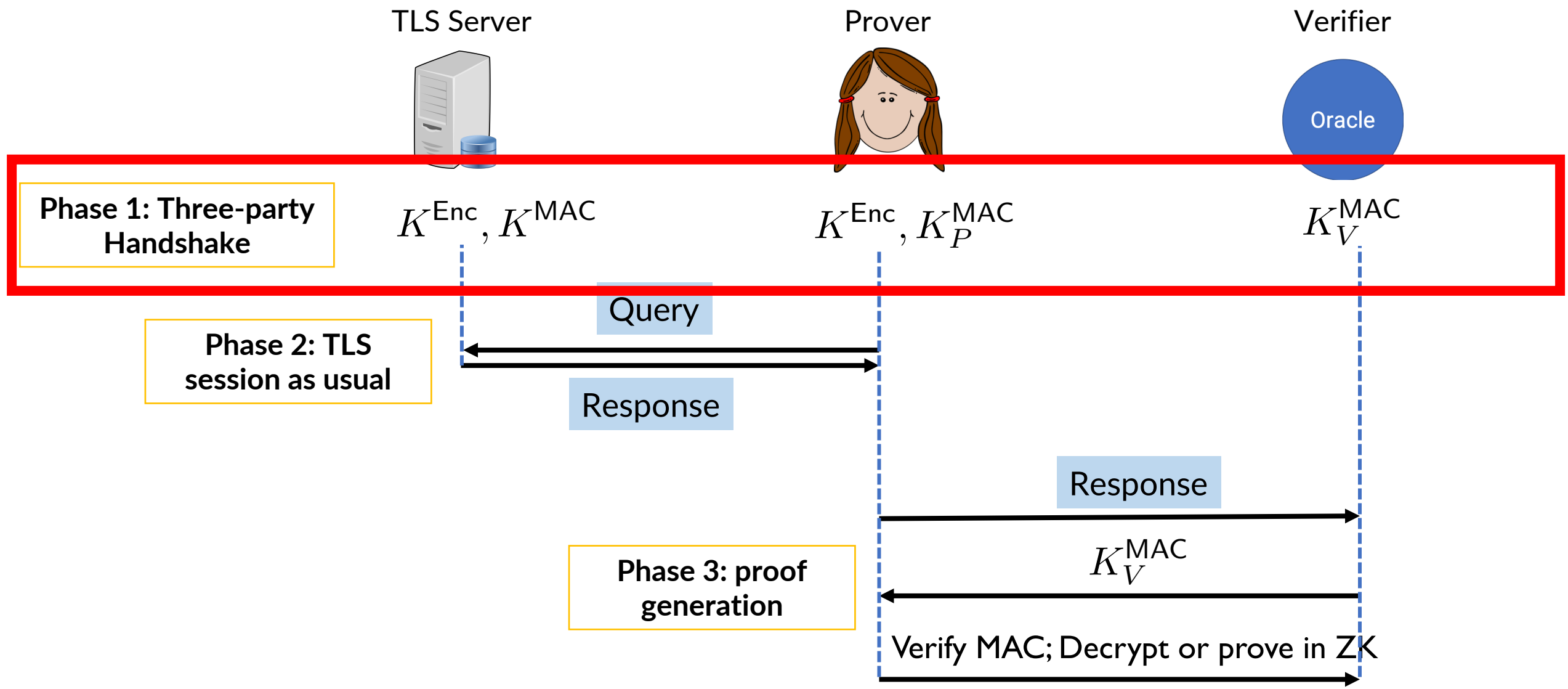
- Idea: Hide the MAC key from the prover until she commits.
- Assuming CBC-HMAC for now (GCM later)



$$K_V^{\text{MAC}} \oplus K_P^{\text{MAC}} = K^{\text{MAC}}$$

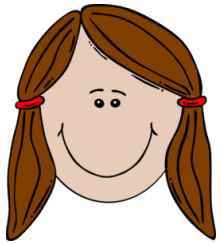
DECO Overview

This denotes a TLS ciphertext.

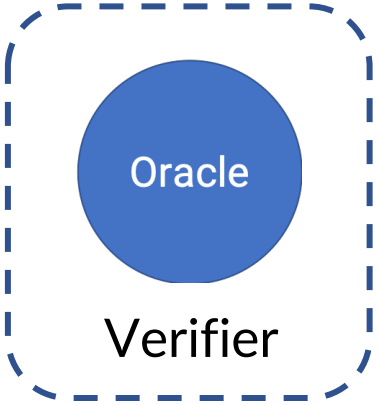


Standard TLS handshake

TLS Server



TLS Client



- Key exchange (e.g. ECDHE)
- Key derivation

- Leverage the homomorphic properties of ECDHE.
- Perform secure Two-party computation (2PC).

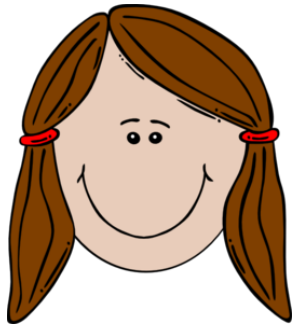
Three-party handshake: key exchange

$y_{\text{server}} = g^{x_s}$

Prover

Verifier

$y_{\text{client}} = g^{x_p} \cdot y_v$



$y_v = g^{x_v}$

Oracle

$z = y_{\text{client}}^{x_s}$

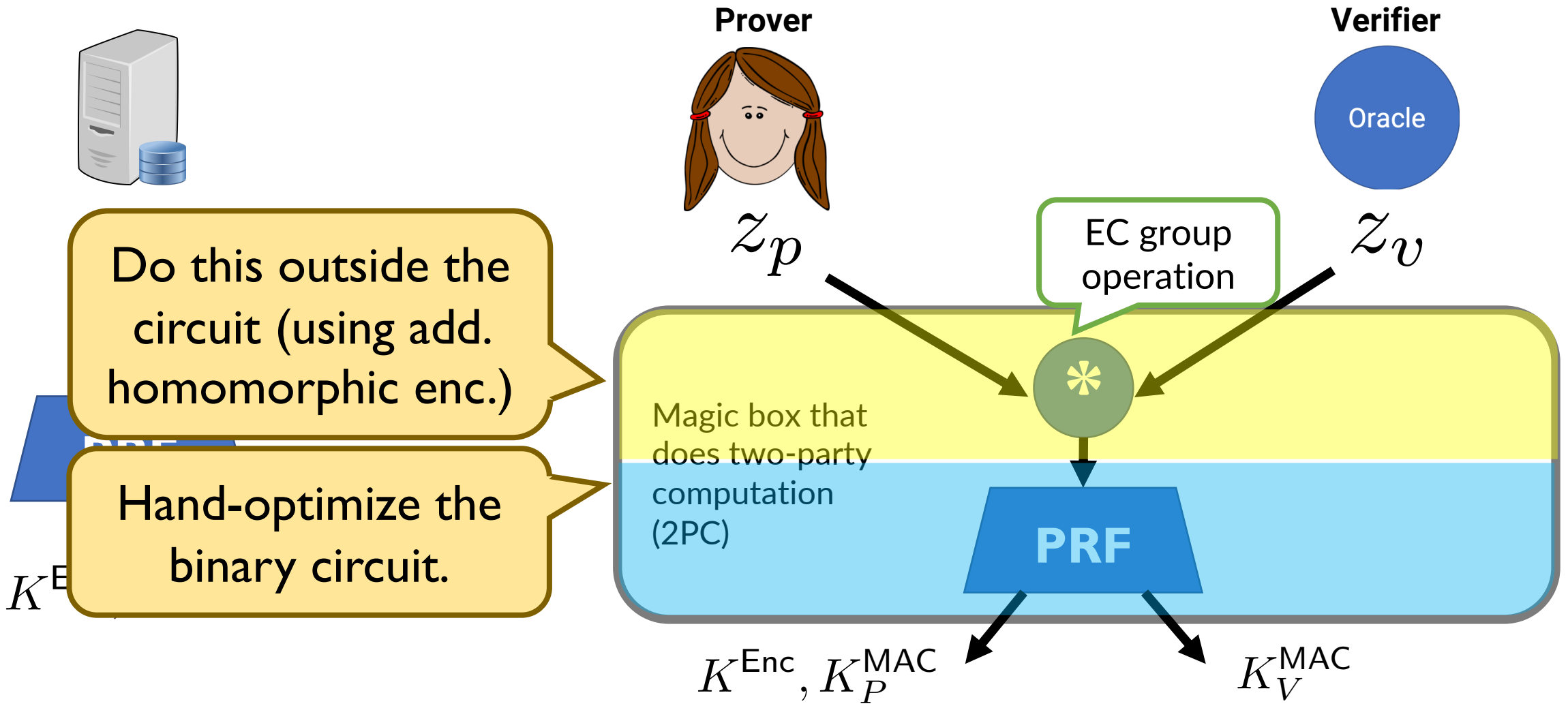
$z_p = y_{\text{server}}^{x_p}$

$z_v = y_{\text{server}}^{x_v}$

EC group operation

$$z = z_P \star z_V$$

Three-party handshake: key derivation



Three-party handshake Performance

- AND complexity of ~770k
- Runtime: 1.40s in LAN, 5.70s in WAN
- Not blazingly fast, but sufficient for DECO applications.

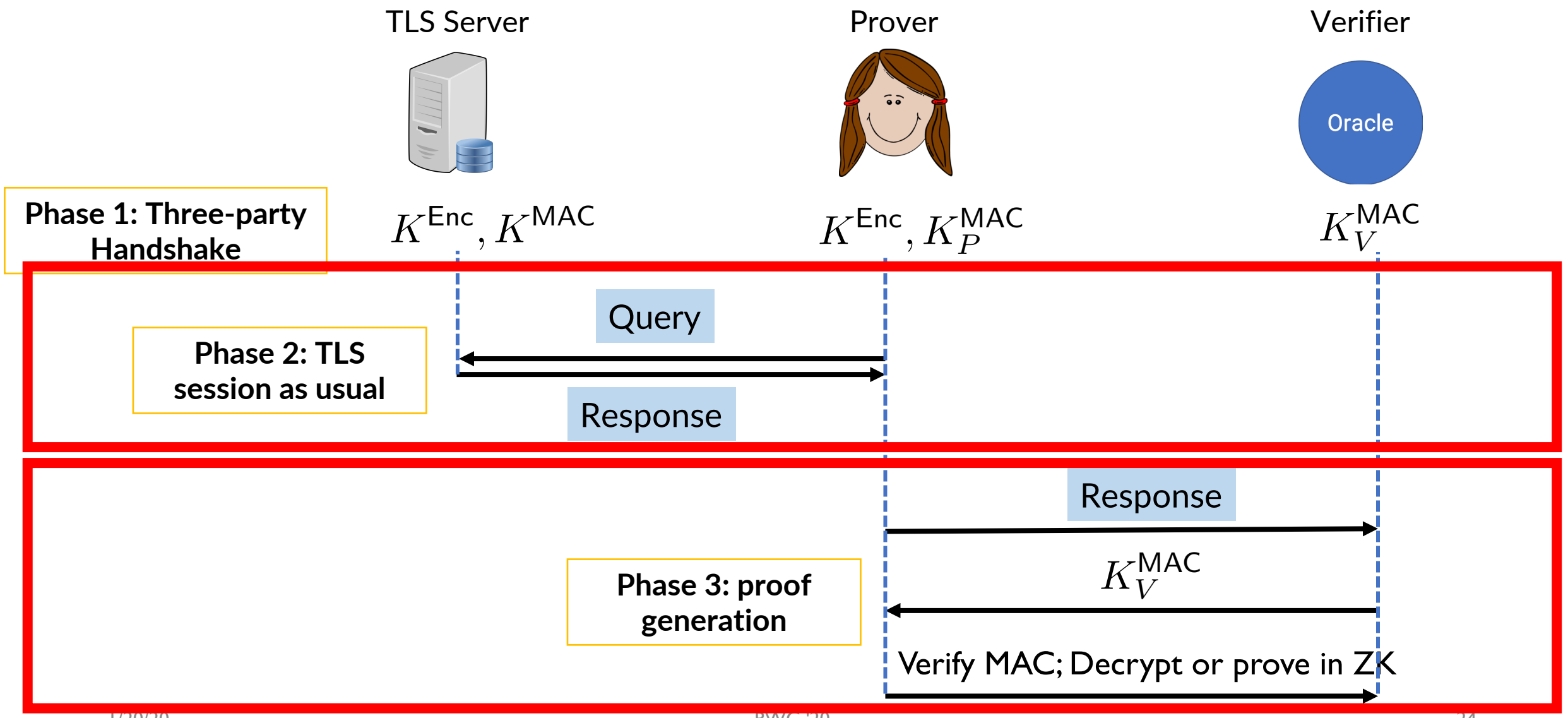
GCM and TLS 1.3

- Handshake for GCM
  - Essentially the same as CBC-HMAC
  - Need a key commitment step (GCM ciphertext is not committing)
  - Overall: small impact on the performance

- DECO supports modern TLS versions
  - TLS 1.2: CBC-HMAC & GCM
  - TLS 1.3: GCM

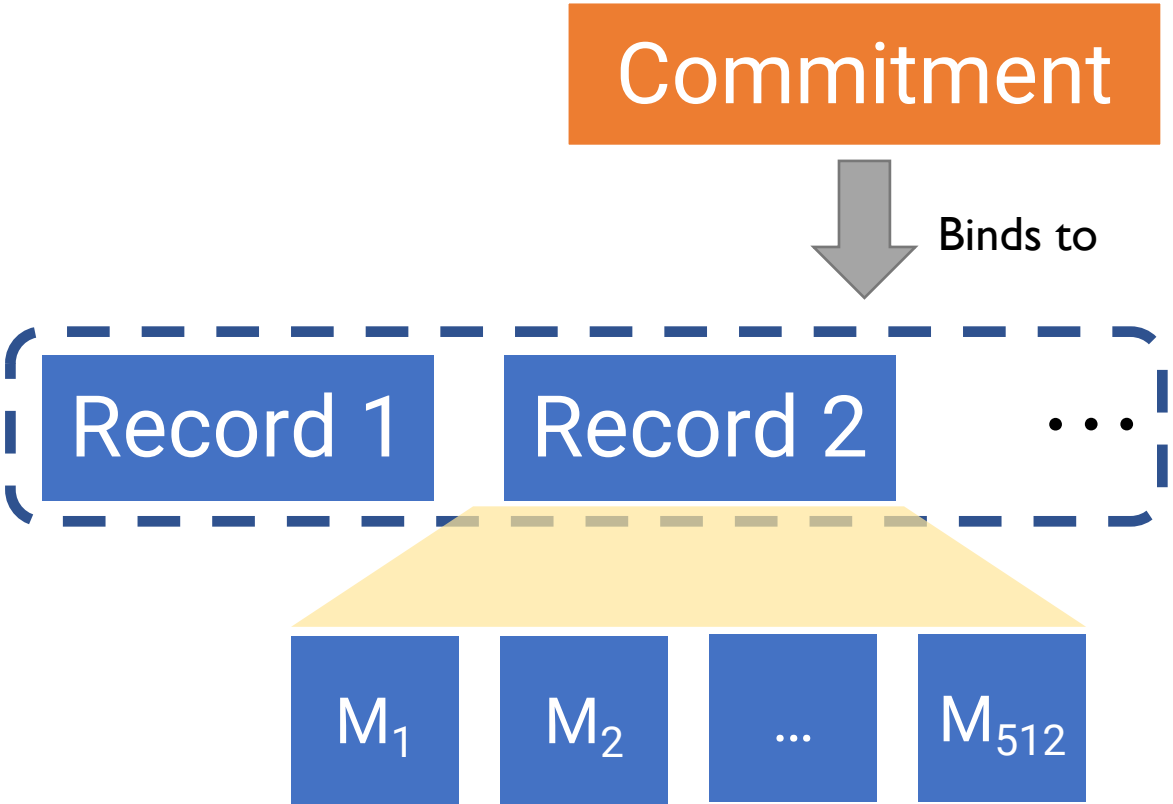
DECO Overview

This denotes a TLS ciphertext.



Now that we can prove provenance…

- Ciphertexts are commitments.
- Open the whole thing (forgoing privacy)
- Selective opening: decrypt partially
  - Record (16KB) and block (128bit) level
- Selective opening + ZKP
  - E.g., age > 18 or bal > $5,000.

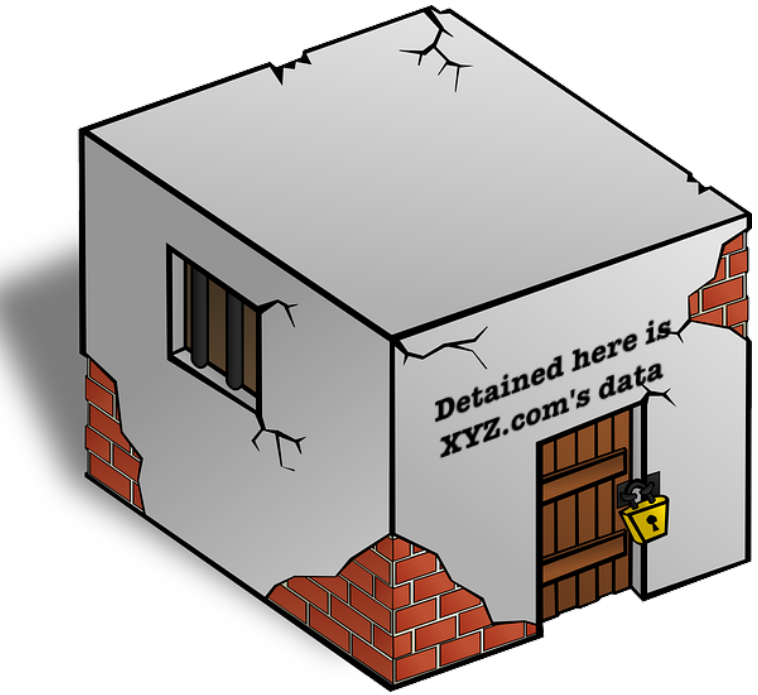


Proof Generation Performance

- Application-specific
- E.g., Age proof: prove age > 18 according to University Registrar website

	Binary Option	Anon. cred.: Age Proof	Price Discrimination
prover time	9.917s	3.677s	8.249s
verifier time	0.011s	0.007s	0.012s
proof size	0.860KB	0.574KB	0.860KB
# constraints	511k	164k	405k

DECO Applications



- Blockchain applications
  - Decentralized identity (DID)
  - Decentralized finance (DeFi)
- Non-blockchain applications too!
  - Age proof
  - Anonymous proofs of ownership of accounts
  - Privacy-preserving personal data marketplace
- Allow users to export private data w/ integrity guarantees *without server's help*.

Take home

- DECO is a privacy-preserving oracle protocol
  - Works with ***modern*** *TLS versions (1.2 & 1.3)*
  - Requires ***no*** *trusted hardware*
  - Requires ***no*** *server-side modifications*
- Visit https://deco.works for our blog post and paper.

Fan Zhang
PhD Candidate, Cornell
https://fanzhang.me