PADO | Privacy-preserving Attestation Data Operators

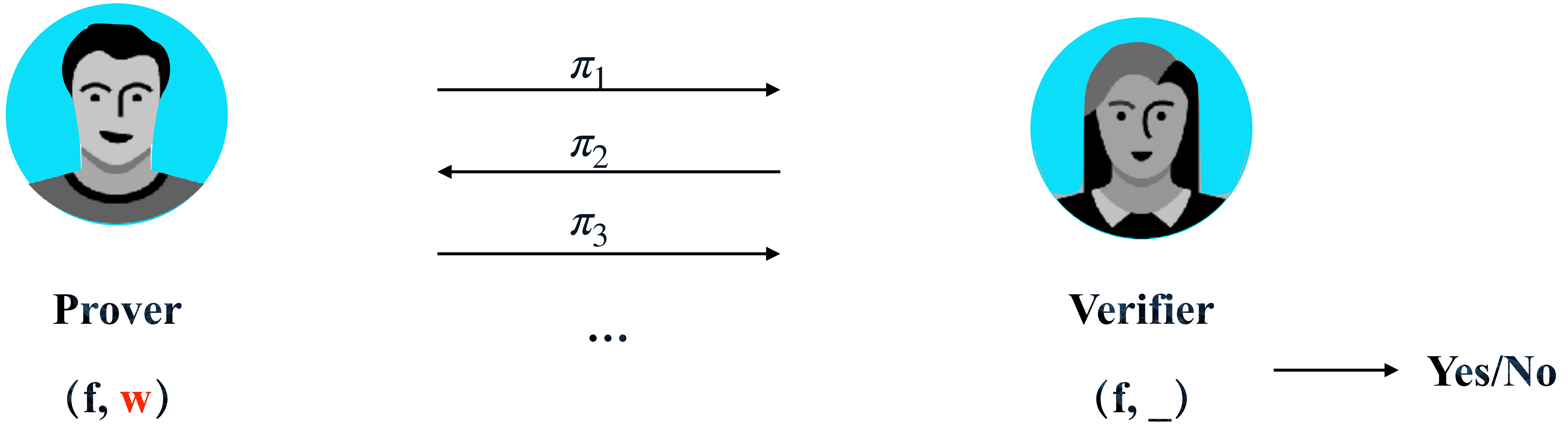
A Survey of Interactive Zero-Knowledge Proof and Its Applications

Xiang Xie   Co-founder@PADO Labs

@_xiang_xie

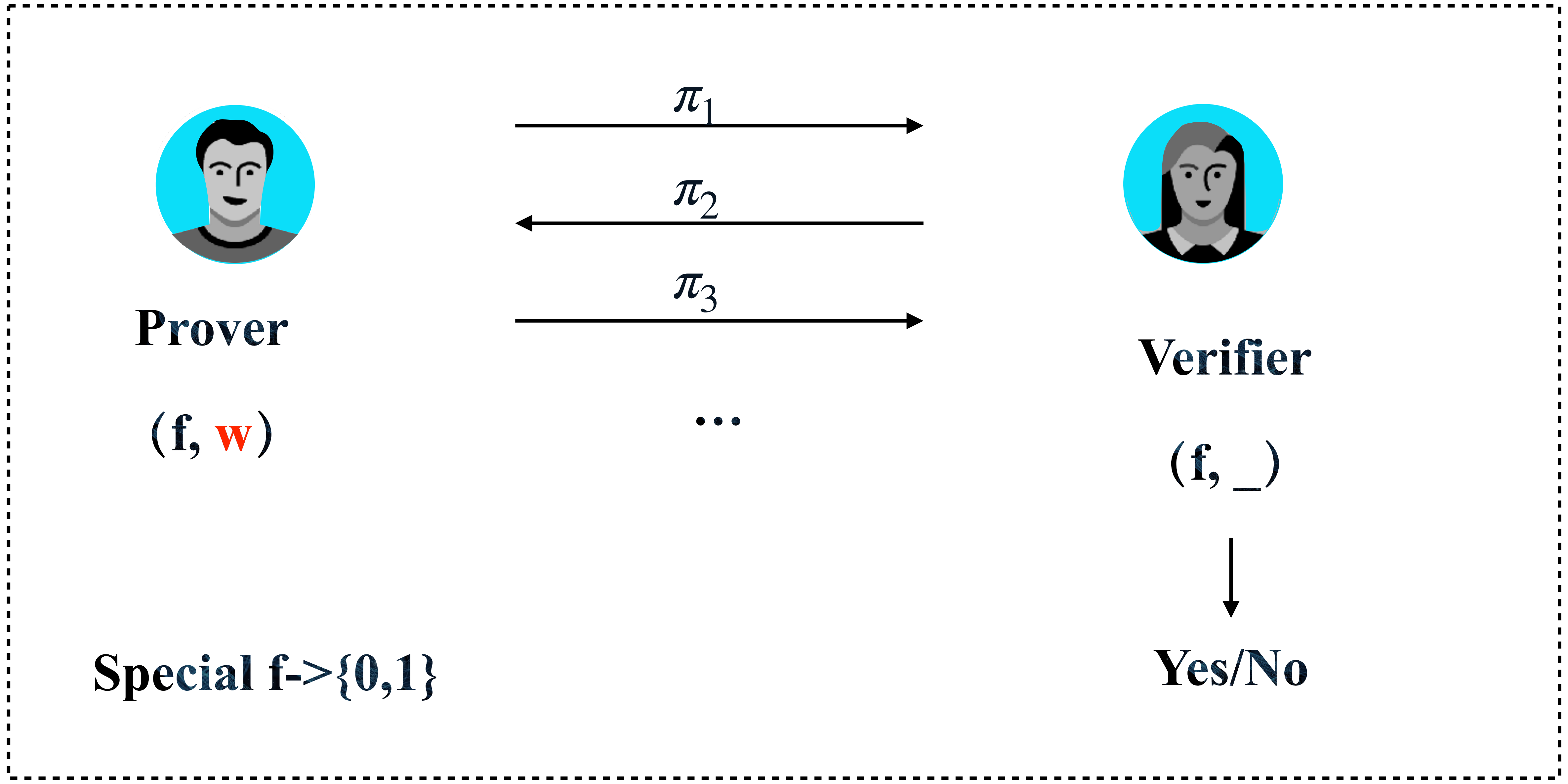
Zero-Knowledge Proofs

f: Instance. The problem/statement.
w: Witness. The secret.
f(w) = 1

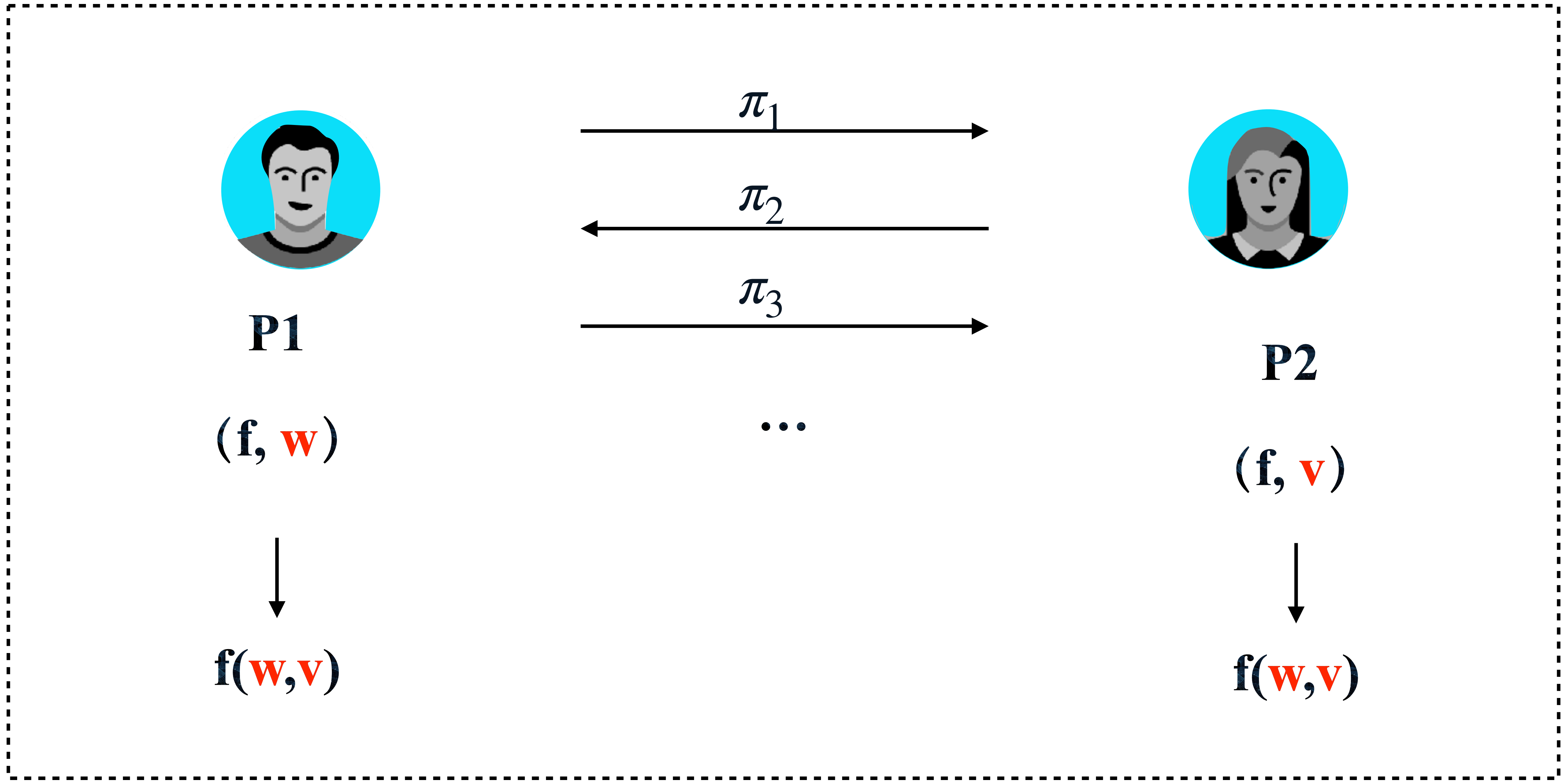


- **Completeness: If Prover has w and generates the proof honestly, Verifier will accept.**
- **Soundness: If Prover does NOT have w, Verifier will always reject.**
- **Zero-Knowledge: Verifier will not learn any other information of w.**

ZKP vs. MPC



Zero-Knowledge Proof



Secure Multi-Party Computation

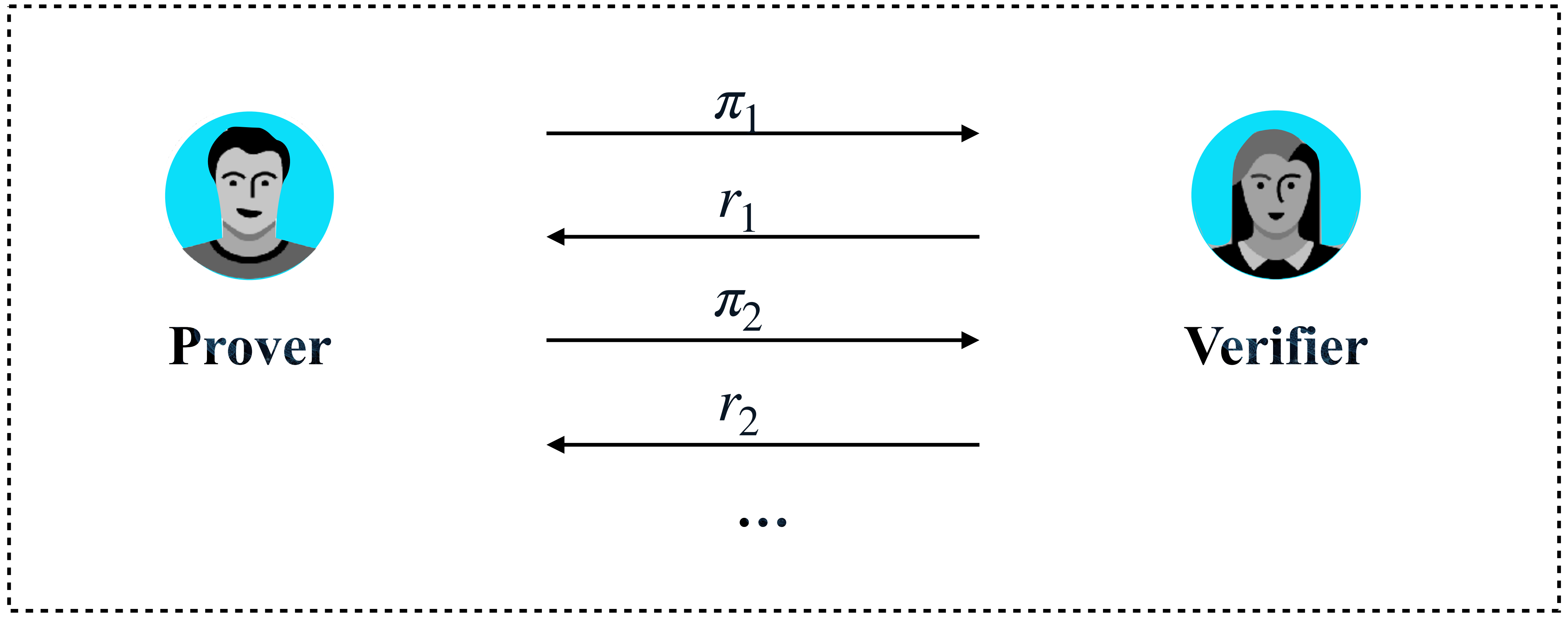
ZKP is a very special case of MPC!

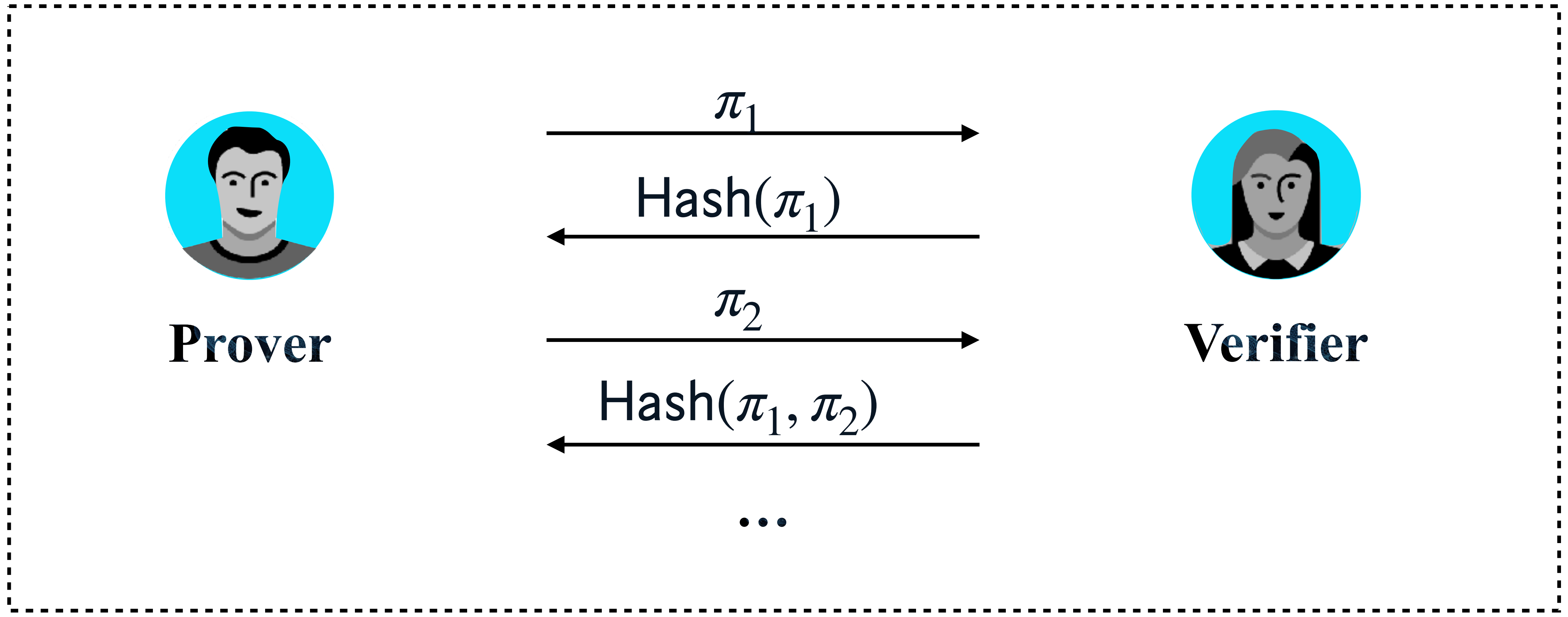
PADO | Privacy-preserving Attestation Data Operators

Interactive Zero-Knowledge Proofs (IZK)

- IZK is a ZKP system in that the Prover and Verifier run a multi-round (interactive) protocol.
- Non-Interactive Zero-Knowledge Proof (NIZK) is a ZKP system with a one-shot protocol.

Public coin IZK | **Fiat-Shamir transformation** → | **NIZK**





r_1,r_2 … are random strings

We focus on private coin IZK!

Basic Ideas of IZK

Prover

π_1 →

r_1 ←

π_2 →

r_2 ←

…

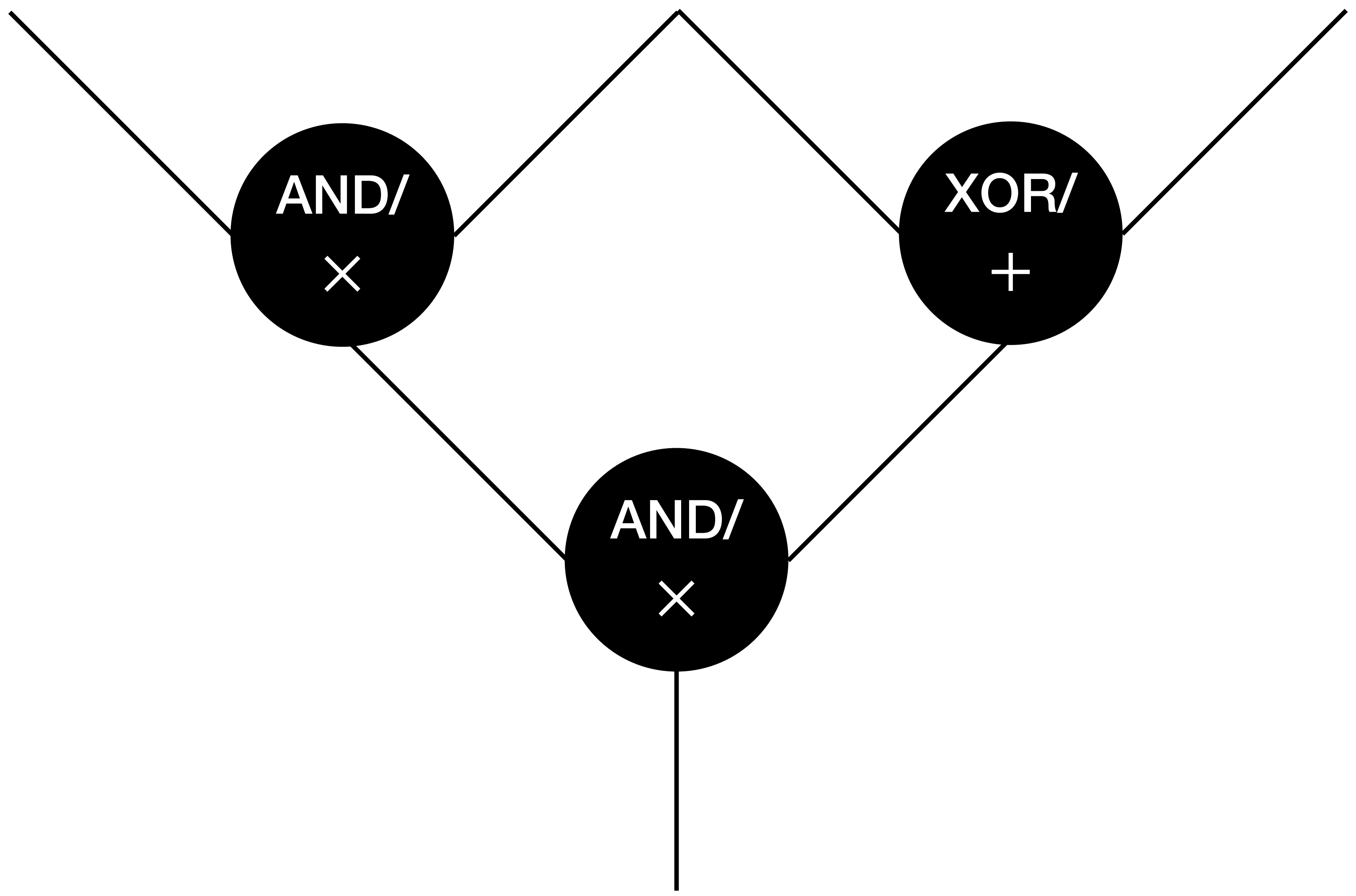
Verifier

Holds the private value b of the wire i.
Hold M_i for the wire i.

Holds a global private Δ.
Hold K_i for the wire i.

$M_i = K_i + b \cdot \Delta$ **This is called an IT-MAC**

Compute the circuit in a gate-by-gate fashion.



State-of-the-art IZK protocols

Prove a single circuit

[YSWW'21] QuickSilver: Efficient and Affordable Zero-Knowledge Proofs for Circuits and Polynomials over Any Field

Threads	Boolean Circuits					Arithmetic Circuits				
	10 Mbps	20 Mbps	30 Mbps	50 Mbps	Local-host	100 Mbps	500 Mbps	1 Gbps	2 Gbps	Local-host
1	4.4 M	6.2 M	7.0 M	7.5 M	7.6 M	1.2 M	3.4 M	4.2 M	4.8 M	4.8 M
2	5.3 M	8.1 M	9.9 M	11.8 M	11.8 M	1.3 M	4.4 M	6.1 M	7.0 M	7.1 M
3	5.7 M	9.1 M	11.4 M	13.9 M	14.3 M	1.4 M	4.9 M	7.2 M	8.4 M	8.4 M
4	5.8 M	9.9 M	12.2 M	14.9 M	15.8 M	1.4 M	5.0 M	7.5 M	8.9 M	8.9 M

Table 2: **Benchmark the performance of our circuit-based ZK protocol.** The benchmark results are the number of AND/MULT gates per second that can be proven using our protocol, where "M" means "million". Benchmark was obtained with different network settings and number of threads.

Instance Information			Boolean Circuits		Arithmetic Circuits	
Type	Price cents/hour	CPU	Speed gates/sec	Cost gates/cent	Speed gates/sec	Cost gates/cent
`c6g.medium`	1.9	ARM	5.3 M	10.0 B	2.2 M	4.1 B
`c5.large`	4.7	Intel	5.9 M	4.5 B	2.9 M	2.2 B
`c5a.large`	4.2	AMD	7.3 M	6.3 B	3.0 M	2.6 B

Table 3: **Performance of stress-testing our ZK protocol on different Amazon EC2 instances.** All instances have 2 vCPUs and 1 GB memory.

State-of-the-art IZK protocols

Prove a batch of circuits

[WYYXW'22] AntMan:Interactive Zero-Knowledge Proofs with Sublinear Communication

B	Running time		Communication
	Setup (ms)	Per gate (μs)	Per gate (field elements)
16	138	0.241	0.82
32	179	0.181	0.41
64	263	0.156	0.205
128	430	0.144	0.1
256	761	0.142	0.051
512	1430	0.142	0.0256
1024	2743	0.141	0.0127
2048	5445	0.141	0.0064
QuickSilver	0	0.107	1

Table 1: **The communication and running time of our ZK protocol.** The running time is benchmarked with 4 threads and 1 Gbps bandwidth. The circuit size $|\mathcal{C}| = 2^{20}$ for the whole table. The setup communication cost for all B in the table is 5.1 MB.

Protocol-thread	Network Bandwidth				
	10 Mbps	50 Mbps	100 Mbps	500 Mbps	1 Gbps
AntMan-1	1.79	2.00	2.05	2.08	2.09
AntMan-2	3.02	3.78	3.91	3.99	4.26
AntMan-4	4.86	6.88	6.69	6.99	7.01
AntMan-8	6.30	10.06	10.79	11.67	11.64
AntMan-16	7.56	14.07	15.86	17.51	17.74
QuickSilver-∞	0.17	0.85	1.7	8.47	16.95

Table 2: **The performance of our ZK protocol subject to the bandwidth and the number of threads.** The benchmark results are the number of million MULT gates per second (mgps). QuickSilver-∞ refers to the theoretical performance of QuickSilver with infinity computational power and thus the running time is solely determined by the communication.

State-of-the-art IZK protocols

Prove machine learning models

[WYXKW'21] Mystique: Efficient Conversions for Zero-Knowledge Proofs with Applications to Machine Learning

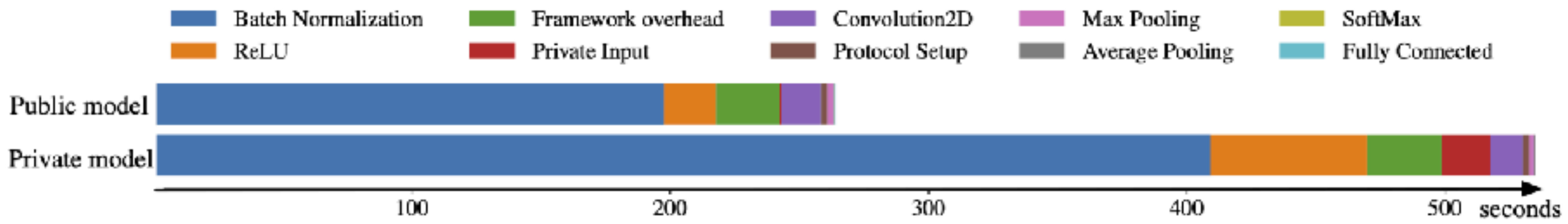


Figure 12: **Execution-time decomposition for ResNet-101 Inference.** The top bar is for public-model private-feature inference; the bottom bar is for private-model private-feature inference. The network bandwidth is throttled to 200 Mbps.

Model	Image	LeNet-5	ResNet-50	ResNet-101
		Communication		
Private	Private	16.5 MB	1.27 GB	1.98 GB
Private	Public	16.5 MB	1.27 GB	1.98 GB
Public	Private	16.4 MB	0.53 GB	0.99 GB
		Execution time (seconds) in a 50 Mbps network		
Private	Private	7.3	465	736
Private	Public	7.5	463	735
Public	Private	6.5	210	369
		Execution time (seconds) in a 200 Mbps network		
Private	Private	5.9	333	535
Private	Public	5.5	336	541
Public	Private	4.9	158	262

Table 3: **Performance of zero-knowledge neural-network inference.** All models are trained using the CIFAR-10 dataset.

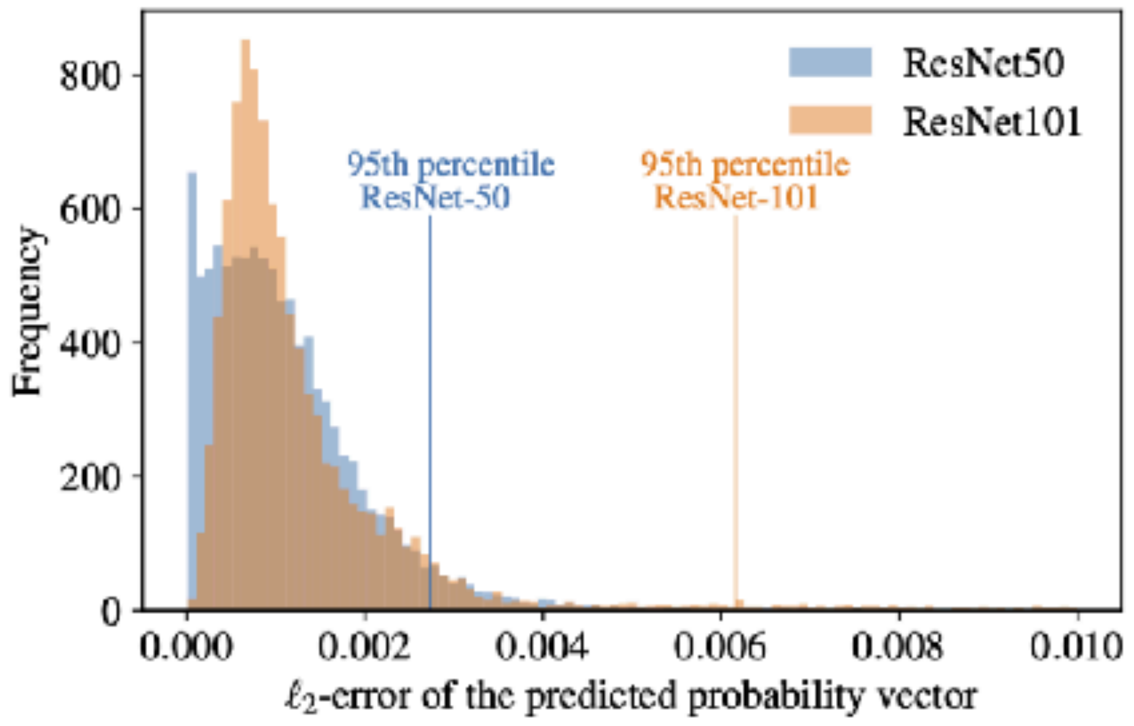


Figure 13: **ℓ_2-norm distance between the plaintext-inference probability vector and the ZK-inference probability vector.** The mean difference is 0.0011 for ResNet-50 and 0.0019 for ResNet-101.

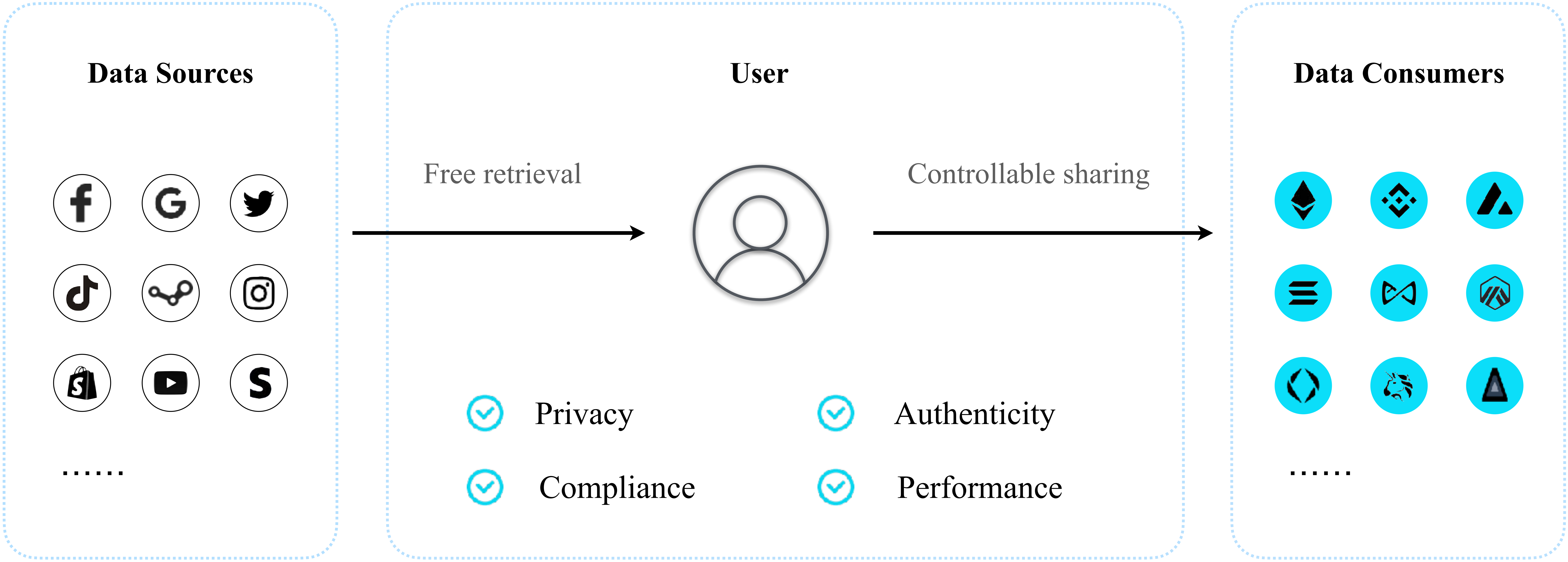
Compared with SNARKs/STARKs

	Prove time	Verification time	Proof size	Memory	Verifier type	Trusted setup
IZK	✓ Fast	✓ Fast	✗ Large	✓ Small	✗ Designed Verifier	✓ No
SNARKs/STARKs	✗ Slow	✓ Super Fast	✓ Small	✗ Large	✓ Any Verifier	✓ ✗ Yes/No

- **IZK is NOT suitable for on-chain verification, because smart contracts DO NOT interact.**
- **IZK is lightweight that could be run on resource-limited devices, such as browser extensions and mobile apps.**
- **IZK is very suitable for Web3 applications, where a service provider is involved.**

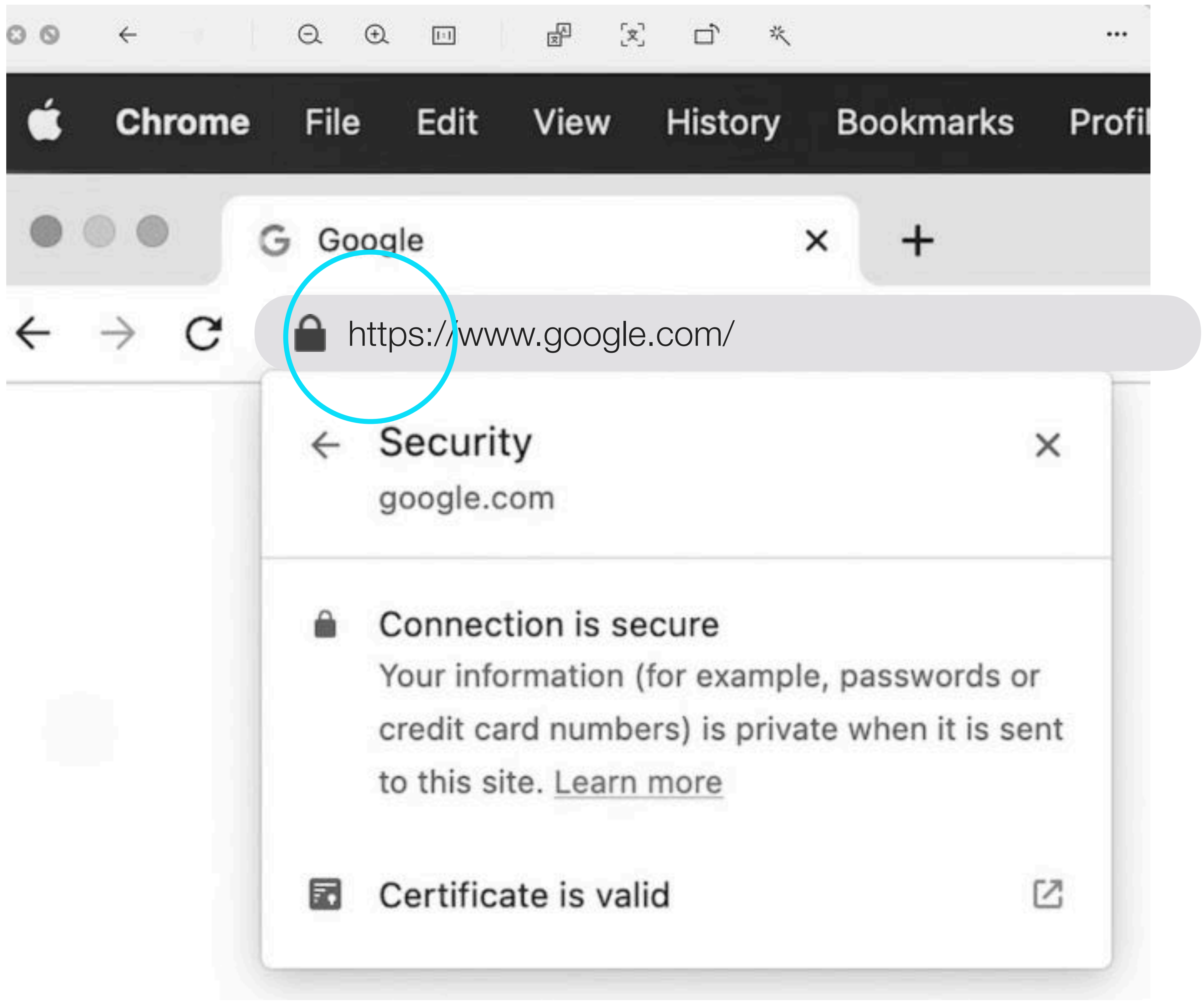
IZK and SNARKs/STRAKs are complements of each other!

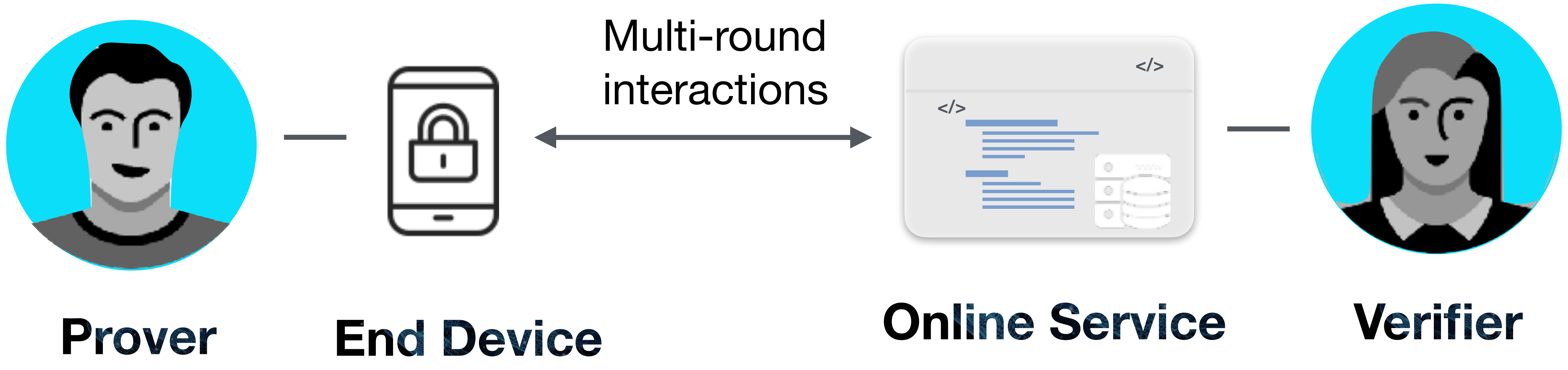
Applications: Bring Web2 data to Web3



Solution insight

Retrieve data via standard Transport Layer Security (TLS) protocol



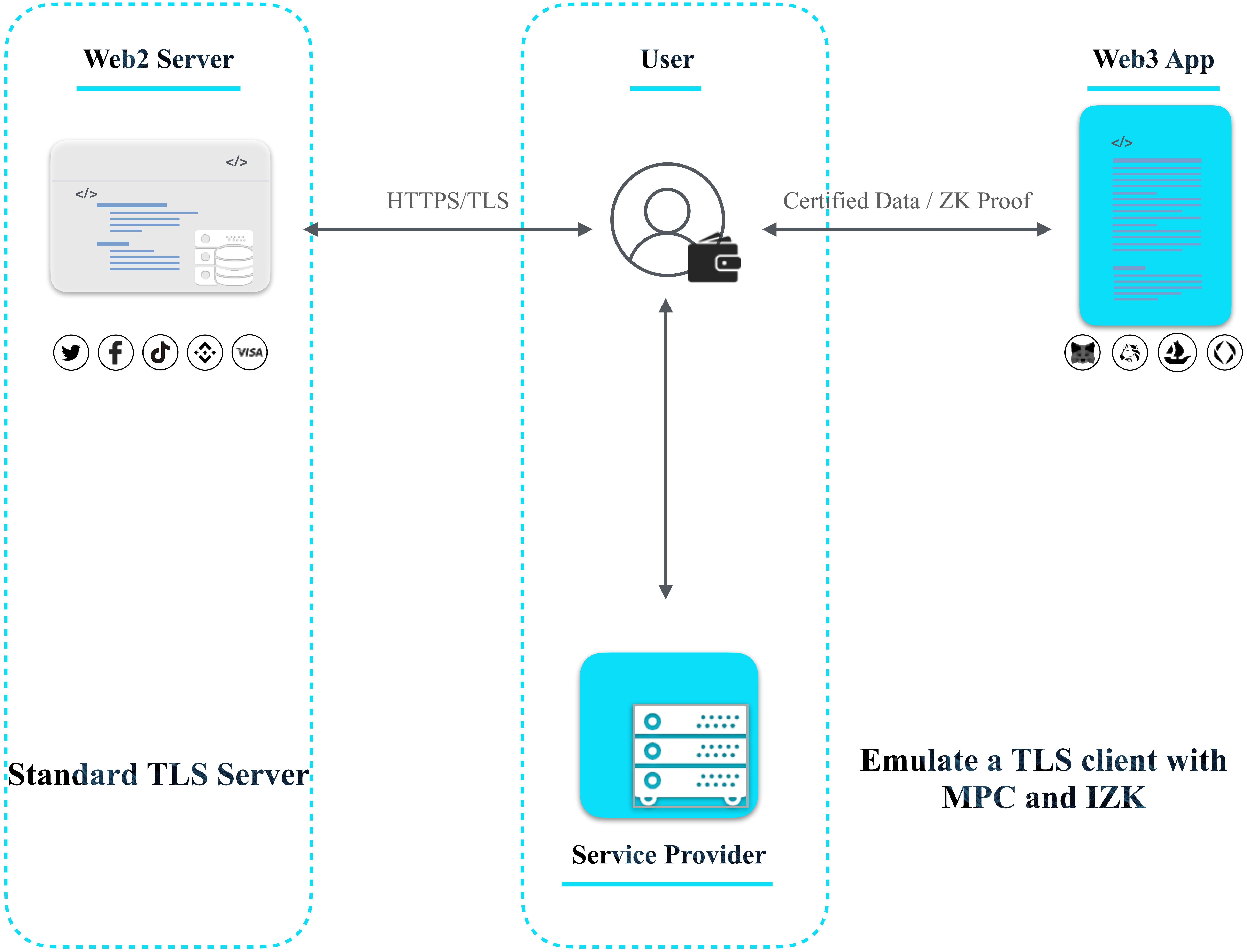


Interactive zero-knowledge proofs (IZK)

- The device proves that the (encrypted) data is indeed derived via TLS.
- Easy to handle zk-unfriendly primitives like AES and SHA256.

PADO | Privacy-preserving Attestation Data Operators

zkDAS—Data Attestation Service



- Computation time: < 1s (10x improvement)
- Communication size: 30MB (14x improvement)
- Memory: < 150MB (>20x improvement)

References

github.com/pado-labs

@padolabs

medium.com/@padolabs

padolabs.org