From Interaction to Independence: zkSNARKs for Transparent and Non-Interactive Remote Attestation

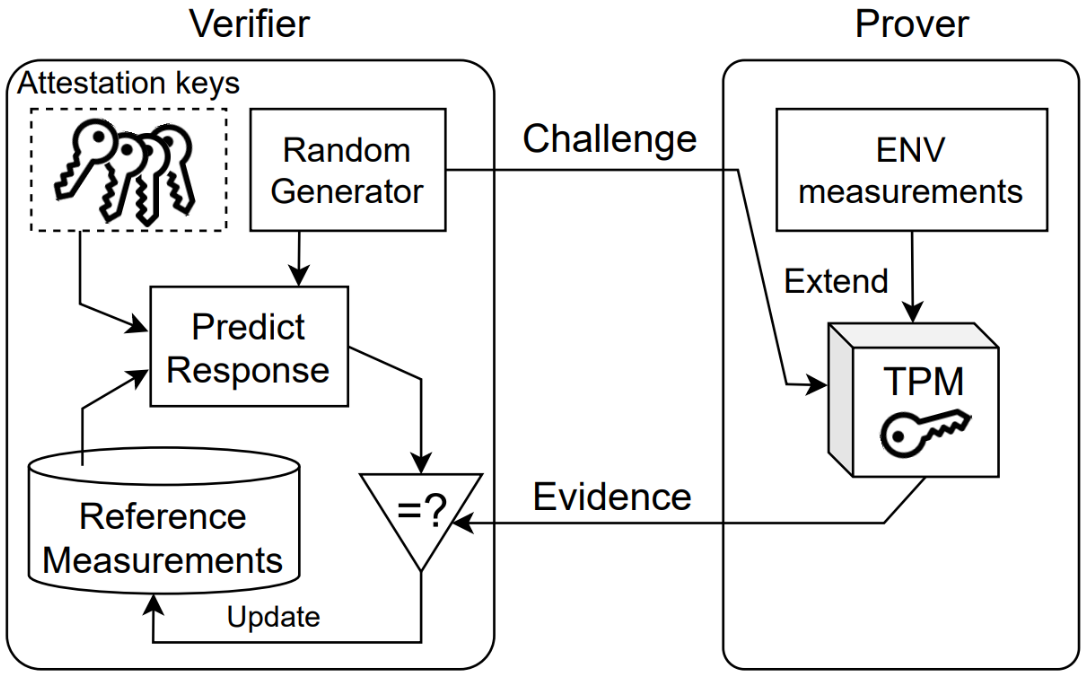
Shahriar Ebrahimi (IDEAS NCBR), Parisa Hassanizadeh (IDEAS NCBR / IPPT PAN)

February 2024

Artifact Evaluated
NDSS SYMPOSIUM
Available
Functional
Reproduced

Remote Attestation

- Verify
  - Authenticity of
    - OS
    - Software
    - Any functionality
  - Remote devices
- Prover: Device
- Verifier: Privileged Owner
- Challenge/Response set



Traditional RA and S-o-t-A

- Trust assumptions
  - Verifier: privileged access to some data
  - Device: trust/authenticate the verifier
  - Users: continuously trust the verifier
- Single point of failure
  - Manufacturer server / Proxy verifier
  - Denial of Service (DoS) attack
- Unique challenge per device
- State-of-the-Art:
  Additional trust assumptions on device or some new entities in protocol
  - Trusted event triggers in device
  - "Secure" smart contracts: usually based on Hyperledger
  - Synchronized secure time clocks
  - . . .

Traditional RA and S-o-t-A

- Trust assumptions
  - Verifier: privileged acce... ...ata
  - Device: trust/a... ...
  - Users: continu... ...the verifier

Unnecessary Trust Assumptions

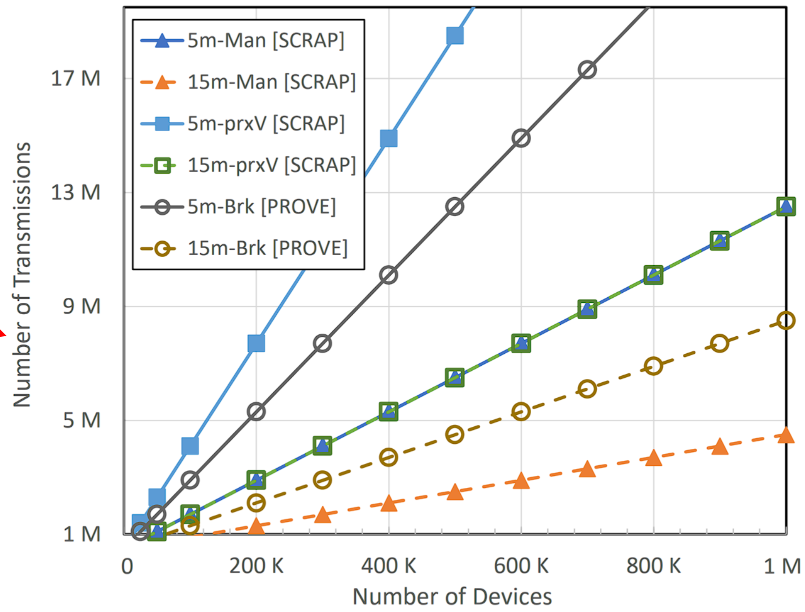
Ambiguous Verification

- Single point of failure
  - Manufac... ...oxy verifier
  - Denial of Service (DoS) attack

Availability

- Unique challenge per de...

Scalability



- State-of-the-Art:

  Additional trust assumptions o... ...or some new entities in protocol
  - Trusted event trig...
  - "Secure" sm... ...ually based on Hyperledger
  - Synchronize... ...e time clocks
  - . . .

Additional Trust Assumptions

Transparent and Non-Interactive RA

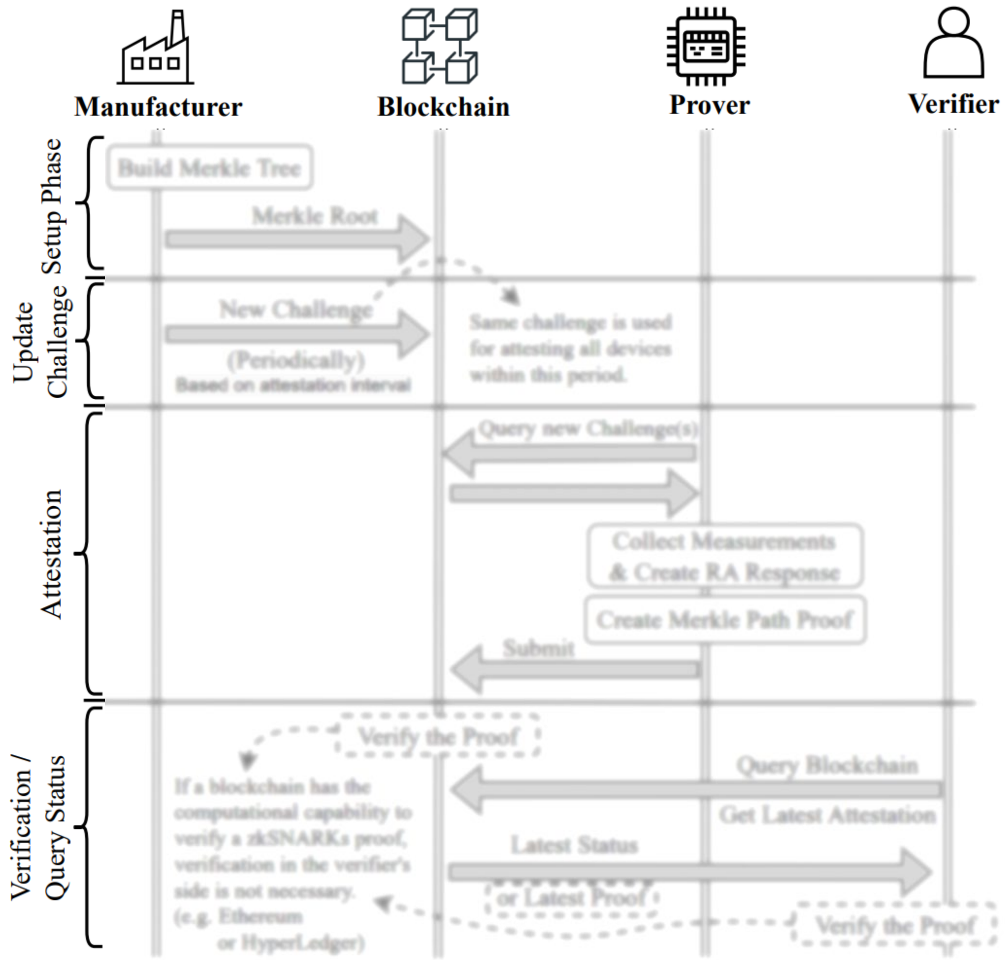
- Transparency
  - Anyone
    - verify the integrity and authenticity of devices
    - without requiring any prior knowledge
      - Platform-Independence
  - New paradigm in the context of public verifiability
    - Trustless public verifiability
- Non-Interactive
  - Zero-trust and server-free
  - Global Challenges
    - Suitable to be built on top of blockchain
    - Resilience to DoS Attacks

Transparent and Non-Interactive RA

- Transparency → No Trust on Manufacturer (After Setup Phase)
  - Anyone
    - verify the integrity and authenticity of devices
    - without requiring any prior knowledge
      - Platform-Independence
  - New paradigm in the context of public verifiability → Zero Trust on Verifier
    - Trustless public verifiability
- Non-Interactive → High Availability
  - Zero-trust and server-free
  - Global Challenges → Linear Network Complexity → High Scalability
    - Suitable to be built on top of blockchain
    - Resilience to DoS Attacks

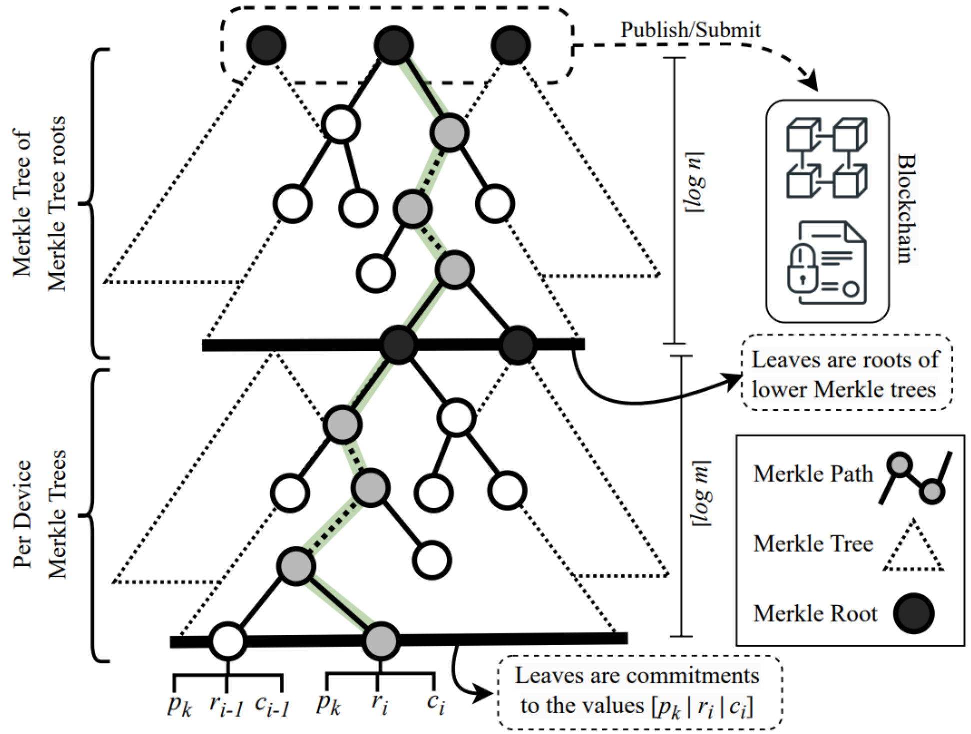
zRA Protocol: Overview

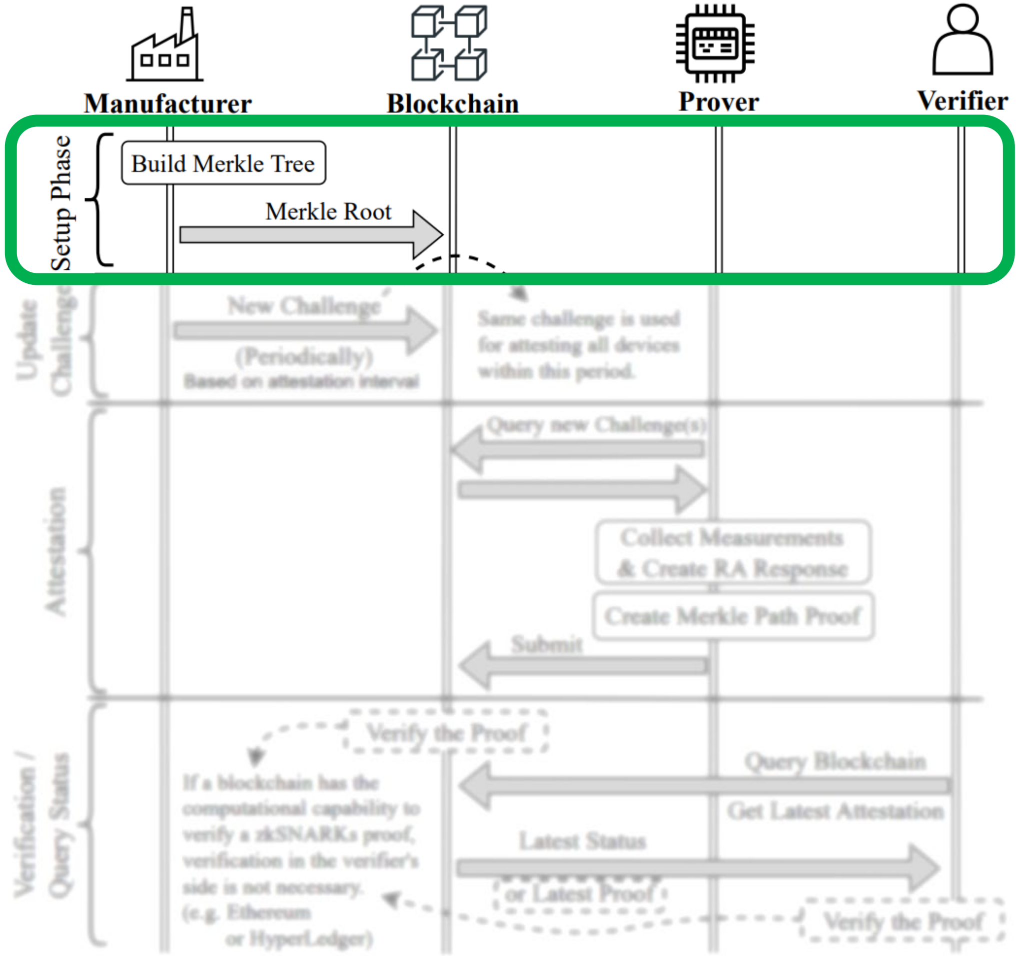
1) Setup phase
   - Done once
   - By Manufacturer
2) Updated challenge
   - Periodically: per attestation interval
   - By Manufacturer
   - Independent from number of devices
3) Attestation
   - By device
   - Asynchronous
4) Verification
   - By anyone



zRA Protocol: Setup

- Merkle Tree of commitments to responses
  - Once deployed, cannot be changed
  - Manufacturer cannot *turn* malicious

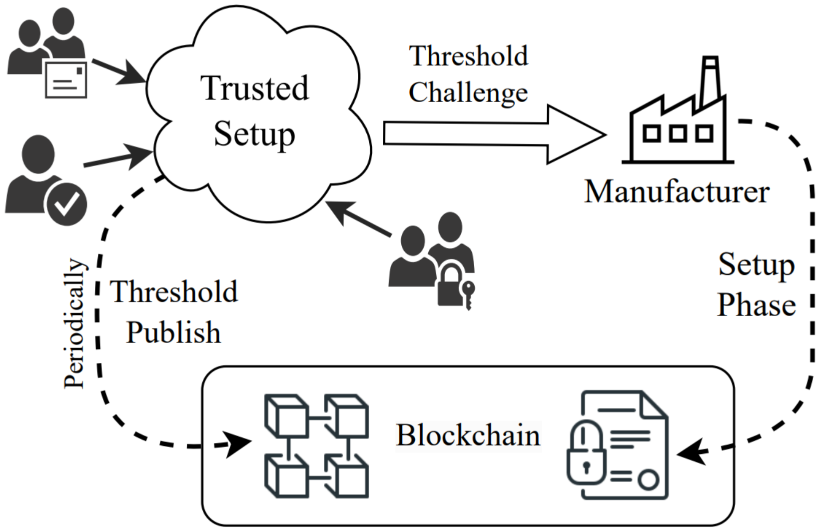


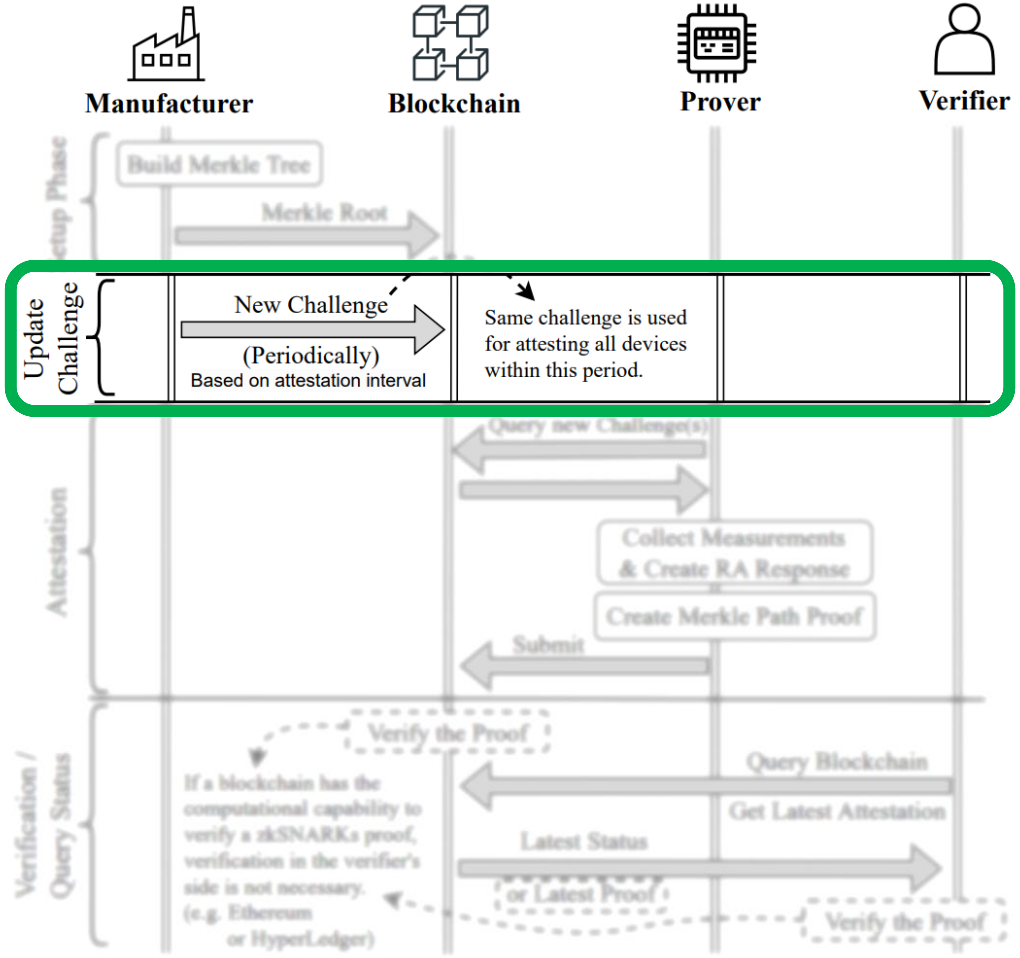


Presented by

zRA Protocol: Update Global Challenge

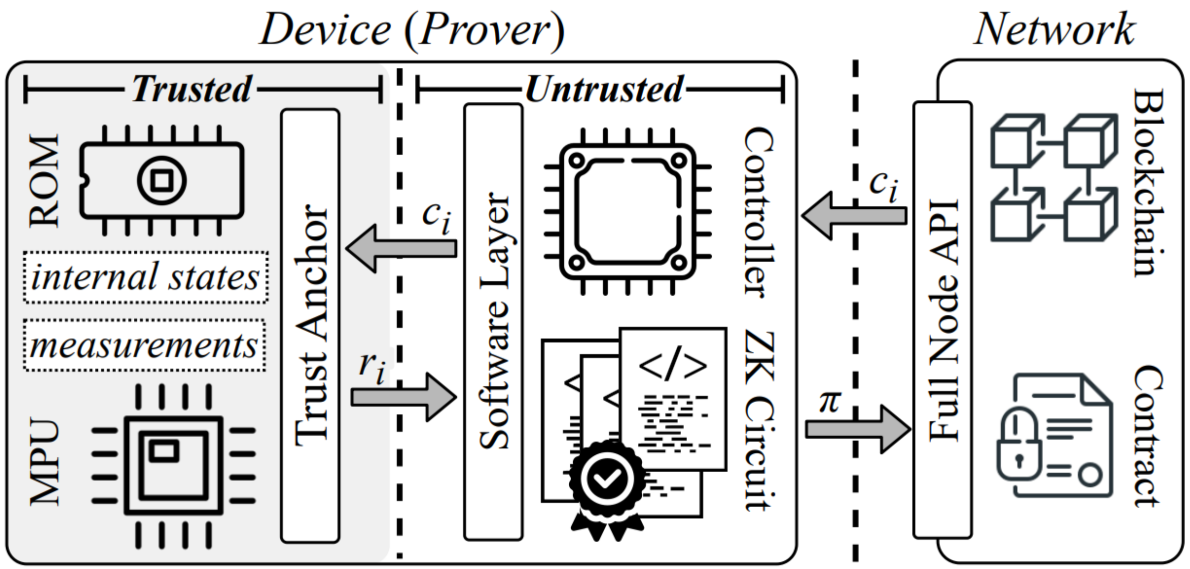
- Done by Manufacturer
  - One C_i for all devices per attestation interval
- Only one (32~64 Bytes) secure storage
  - Pseudo-random sequence
- Potential for being generated in MPC

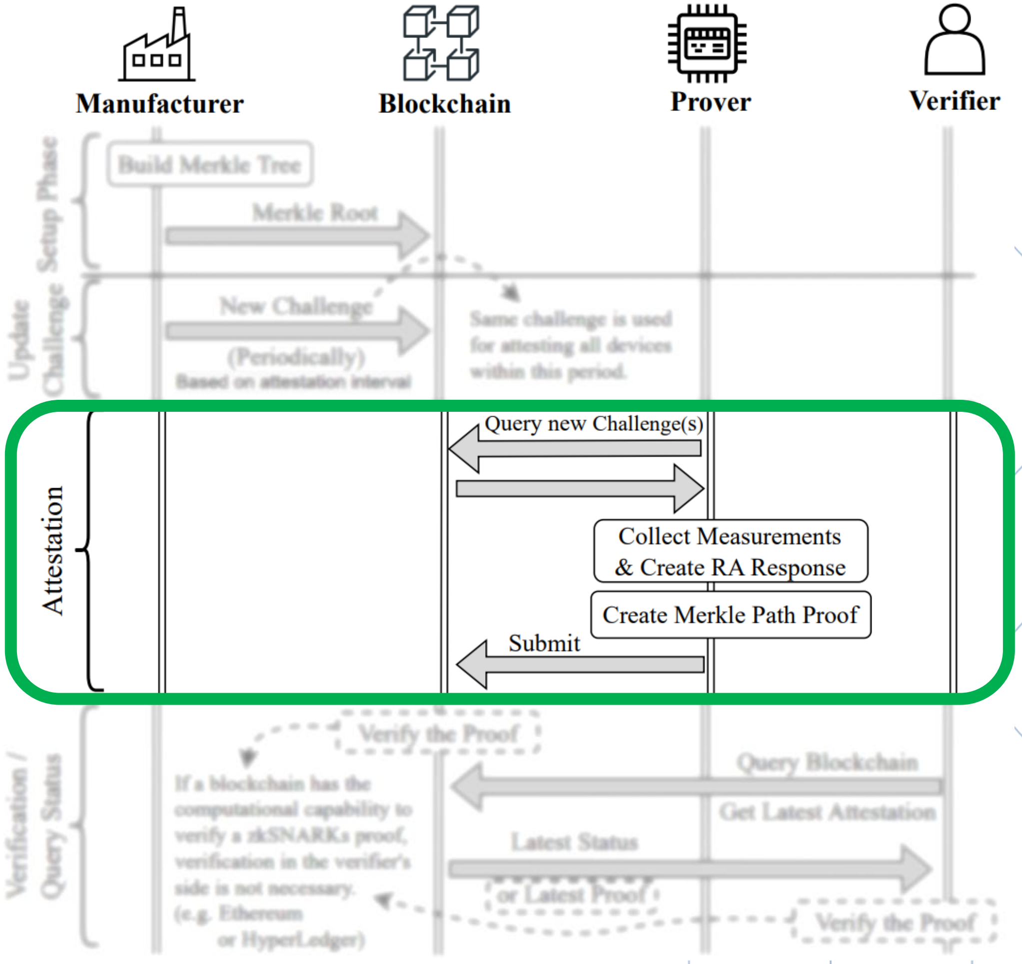




zRA Protocol: Attestation

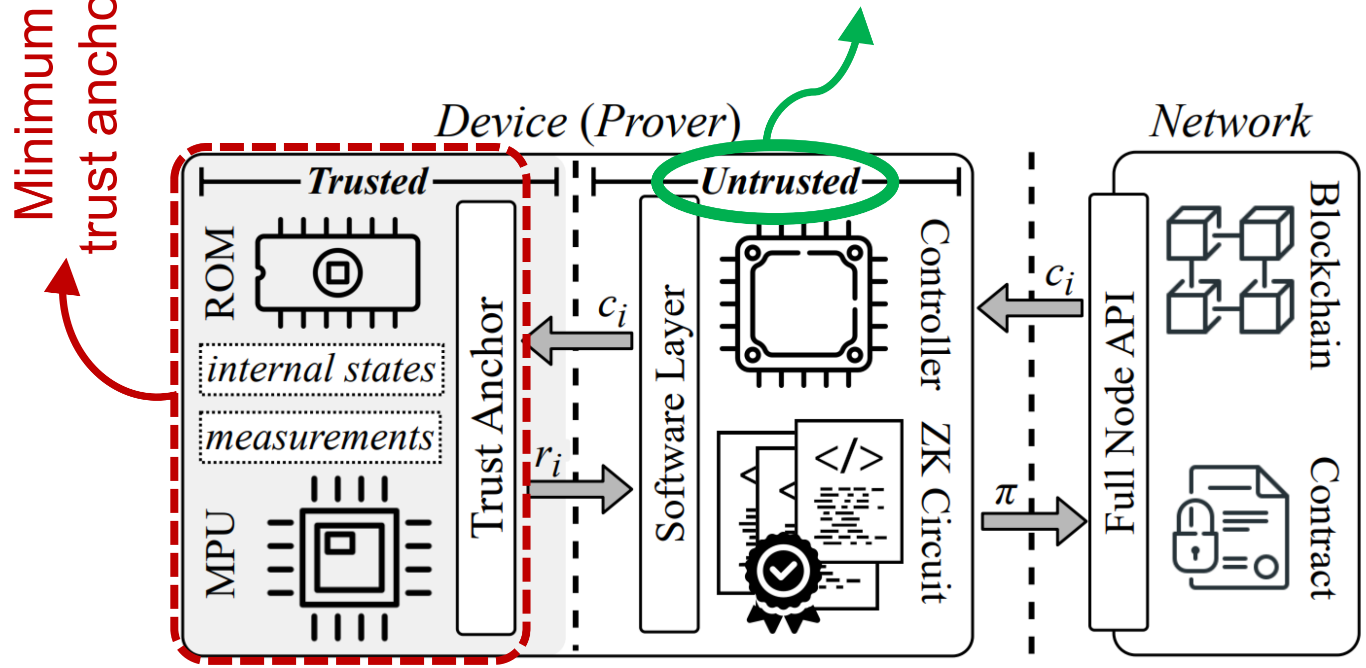
$$S[\mathcal{R}_{\mathcal{T}}, l, c_i, i, p_k] = \{ \text{ I know } r_i \in \mathbb{B}^{248},\ i \in \mathbb{B}^{h}, \text{ such that } l = H^3_{pos}(p_k | r_i | c_i) \text{ and } O(\mathcal{T}, i) \text{ is the opening (path) of } l \text{ at position } i \text{ to the root } \mathcal{R}_{\mathcal{T}} \}$$

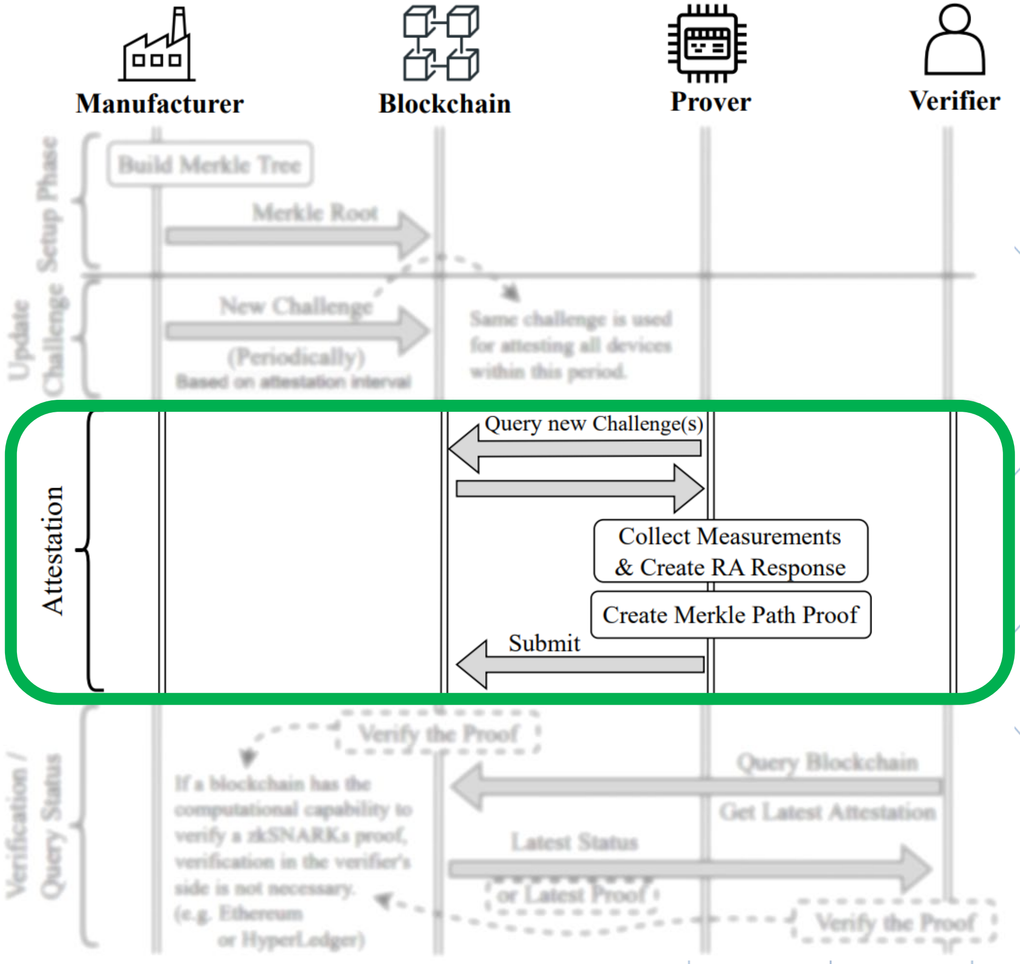




zRA Protocol: Attestation

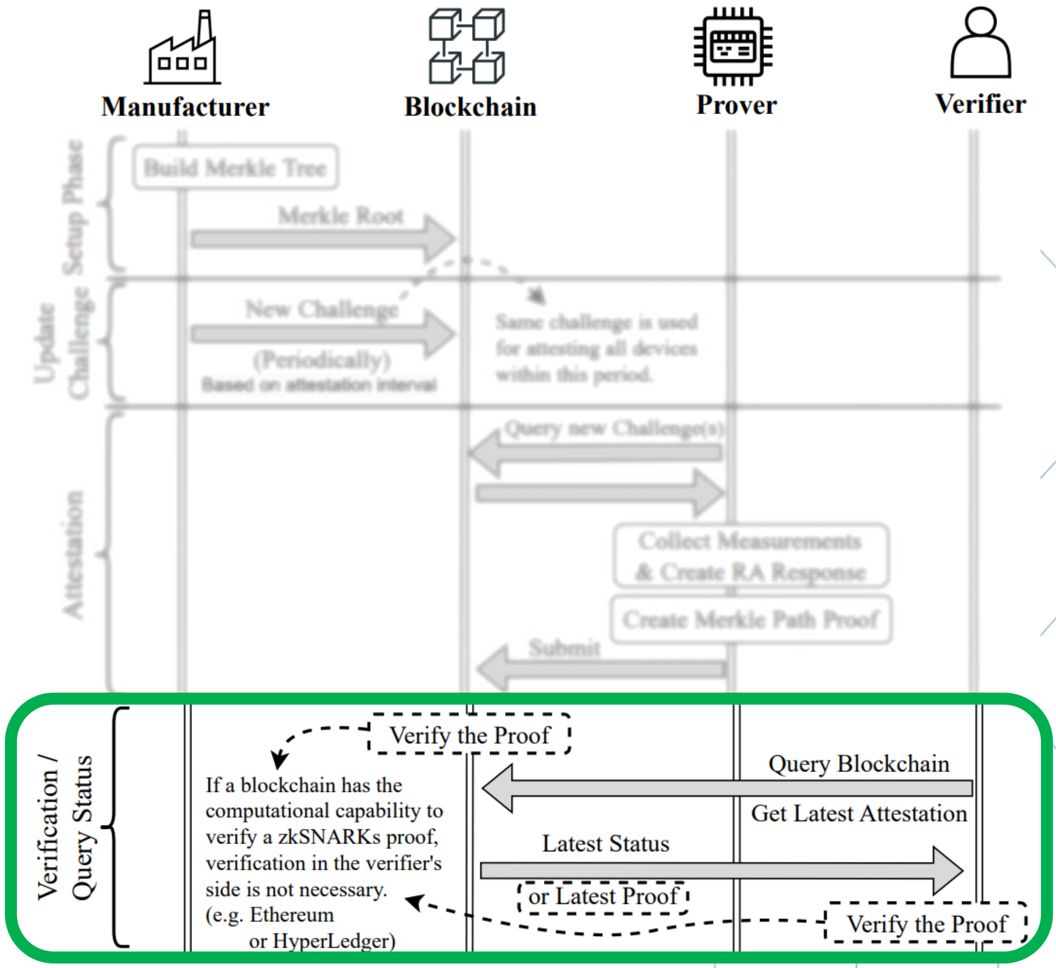
$$S[\mathcal{R}_{\mathcal{T}}, l, c_i, i, p_k] =$$
$$\{ \text{ I know } r_i \in \mathbb{B}^{248},\ i \in \mathbb{B}^{h},$$
$$\text{such that } l = H^3_{pos}(p_k|r_i|c_i) \text{ and } O(\mathcal{T}, i) \text{ is the}$$
$$\text{opening (path) of } l \text{ at position } i \text{ to the root } \mathcal{R}_{\mathcal{T}} \}$$





zRA Protocol: Verification

- No need to
  - Know device's public key
  - Know calculations behind $r_i = f(c_i)$
  - Interact with device
  - Keep track of previous responses
  - Care about replay attacks
  - Trust the manufacturer
    - After the setup phase, manufacturer and all devices are committed to correct challenges and responses pairs.

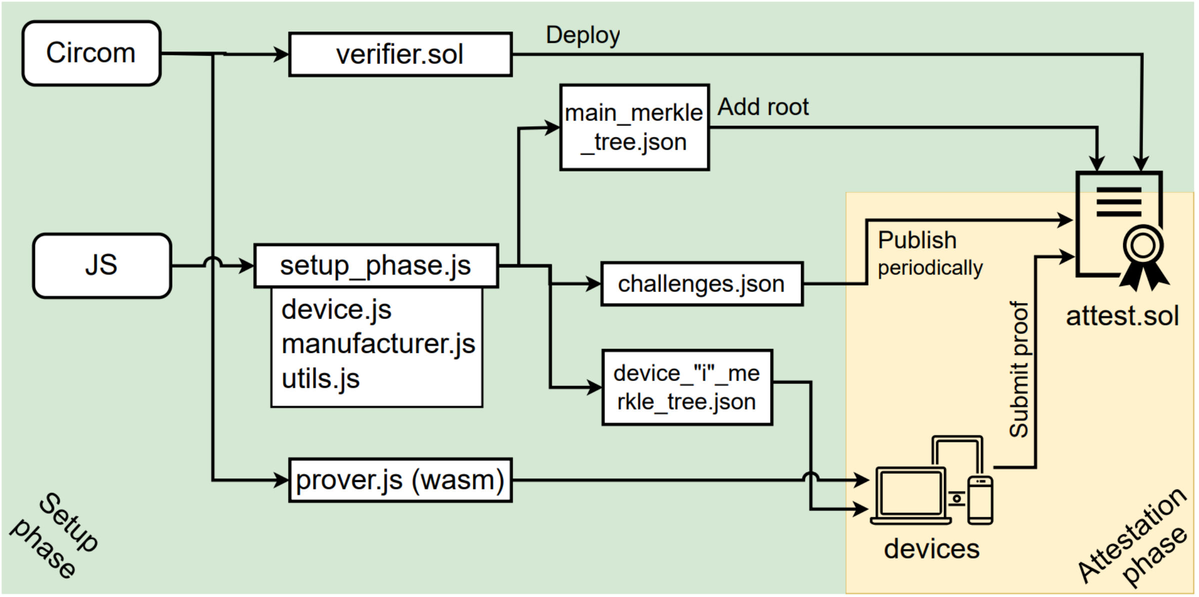


Security Analysis

- Denial of service (DoS)
  - Public permissionless blockchain
- False attestation and Replay attacks
  - Soundness of the proving system + Public inputs of the proof: $pk|r_i|c_i$
- Message manipulation or access to the private key s_k
  - Soundness of the proving system + security of hash function: H^3_{pos}
- Manipulate ZK circuit execution
  - Soundness of the proving system
- Blockchain update delay (Block-time)
  - limited to the block-time of the blockchain. e.g., Ethereum 11 seconds
- Software updates and rollback attacks
  - Easily solved by updating the root in contract

Implementation

- Manufacturer
  - JS
- Automated Verifier
  - Solidity
  - Deployed contract
    - On Sepolia testnet
    - Links in the paper
- Device
  - SnarkJS, Circom



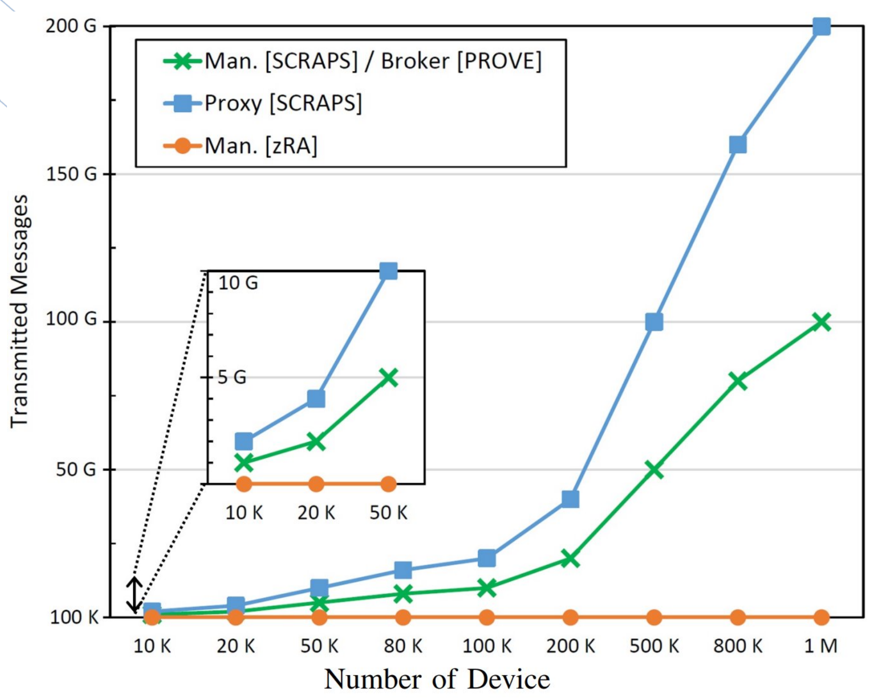
- Github: https://github.com/zero-savvy/zk-remote-attestation

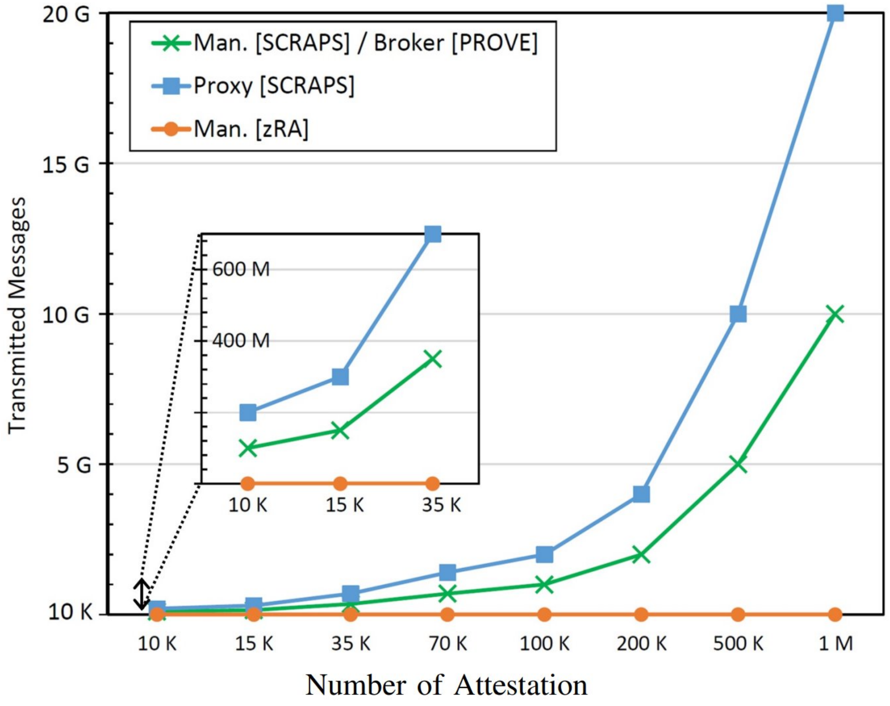
Experiments: Setup

	Dell Latitude 5531	Raspberry Pi Zero 2W	ASUS Tinker board
Memory	16.0 GiB	512MB SDRAM	2.0 GiB LPDDR3
Processor	12th Gen Intel® Core™ i5-12500H	1GHz quad-core Arm Cortex-A53	1.8GHz Quad-core ARM Cortex-A17
Storage	512 GB	16 GB SanDisk SD Card	
Operating System	Ubuntu 22.04.2 LTS	Raspberry Pi OS Lite (64-bit)	Tinker Board Debian Stretch V2.2.9
Power Source	USB-C Thunderbolt: 45W	Micro USB power: 12W (5V)	Micro USB power: 15W (5V)
IoT Compatible	✗	✓♦	✓⁎

♦ Dimensions: 65mm×30mm. ⁎ Dimensions: 85mm×54mm.

Experiments: Scalability & Communication cost





[SCRAPS] Petzi et al. "SCRAPS: Scalable Collective Remote Attestation for Pub-Sub IoT Networks with Untrusted Proxy Verifier." 31st USENIX Security Symposium (2022)
[PROVE] Dushku et al. "PROVE: Provable remote attestation for public verifiability." Journal of Information Security and Applications 75 (2023)

Experiments: Attestation Performance

	Prover			Proxy Verifier [1] / Broker [2]	Verifier
	Device	Time	Energy		
SCARAPS [1]	Cortex M-33	1.07 s	N/A	55.4 ms	-
PROVE [2]	Virtex-7	4.6 ms	N/A	~7 ms	-
zRA	Core-i5	0.6 s	479 mJ	-	<1 ms
	Cortex-A53	21.8 s	14.46 J		
	Cortex-A17	11.9 s	53.08 J		

[1] Petzi et al. "SCRAPS: Scalable Collective Remote Attestation for Pub-Sub IoT Networks with Untrusted Proxy Verifier." 31st USENIX Security Symposium (2022)
[2] Dushku et al. "PROVE: Provable remote attestation for public verifiability." Journal of Information Security and Applications 75 (2023)

Experiments: Attestation Performance

	Prover			Proxy Verifier [1] / Broker [2]	Verifier
	Device	Time	Energy		
SCARAPS [1]	Cortex M-33	1.07 s	N/A	55.4 ms	-
PROVE [2]	Virtex-7	4.6 ms	N/A	~7 ms	-
zRA	Core-i5	0.6 s	479 mJ	-	<1 ms
	Cortex-A53	21.8 s	14.46 J		
	Cortex-A17	11.9 s	53.08 J		

Acceptable

Room for improvement

Direct effect on Scalability

[1] Petzi et al. "SCRAPS: Scalable Collective Remote Attestation for Pub-Sub IoT Networks with Untrusted Proxy Verifier." 31st USENIX Security Symposium (2022)
[2] Dushku et al. "PROVE: Provable remote attestation for public verifiability." Journal of Information Security and Applications 75 (2023)

Conclusion

💡 zkSNARKs for Cyber-Physical Systems (CPS)

🚀 High potential to increase scalability

- Remove interactions
- Ideal for building on top of distributed infrastructures, e.g., blockchains
- Global challenges

🤝 Possibility to resolve trust issues in different protocols

🚧 Concern

- Computational complexity in prover (usually devices) side

🔭 Future work

- More efficient implementations
- Hardware acceleration
- More efficient proving schemes, e.g., Spartan
- Trade-offs: proving complexity, proof size, and verification complexity