Performance Evaluation of Self-Sovereign Identity (SSI)

Emin Adem Buran[1,2],

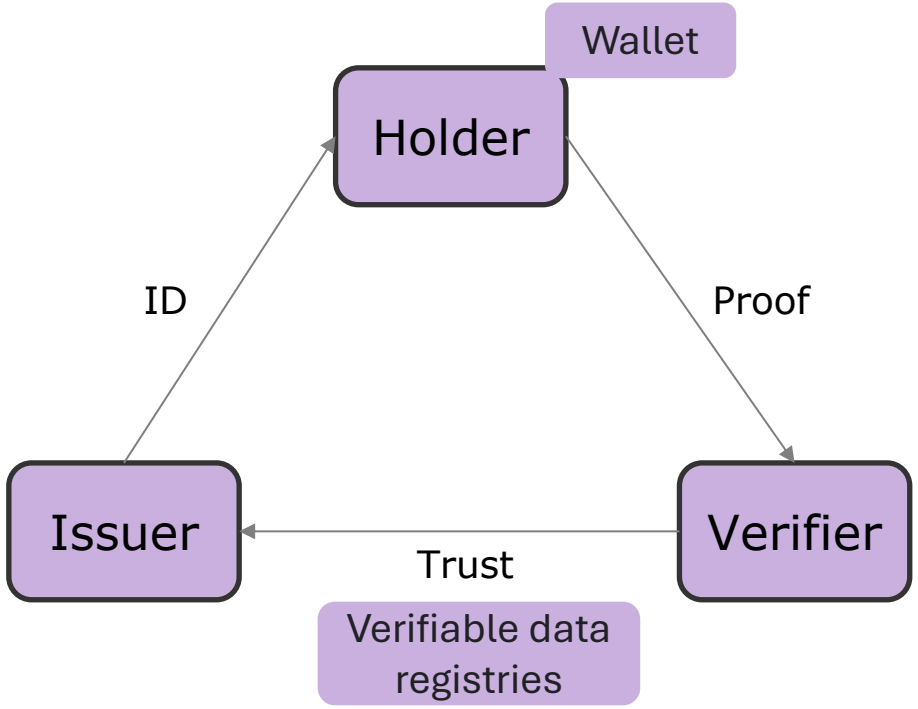
Christian Prehofer[2,1*]

Srivatsav Chenna[2,1]

[1] *Technical University of Munich, Arcisstraße 21, 80333 Munich, Germany*
[2] *DENSO AUTOMOTIVE Deutschland GmbH, Eching, Germany*

Objective

- Self-sovereign, decentral Identity (SSI)
  - W3C standard for DID identity documents
    - Wallets with private keys
  - Separate issuer and verifier(s)
    - Rely on verifiable data registries



- Performance of SSI on Raspb. Pi 4 and x86
  - Latency
  - Memory and CPU usage
  - Scalability

- Considered DID methods:
  - CHEQD: Blockchain based trust anchor
  - KEY, WEB, JWK: local or web-based (TLS certificates) trust anchor

SSI Tool Kit: Walt.id

- An open source identity & wallet infrastructure
- Issuance, verification, and management of verifiable credentials
  - Verifiable credentials (VC) according to W3C Data Model
- Supported DID methods : KEY, JWK, WEB, CHEQD

Experimental Setup

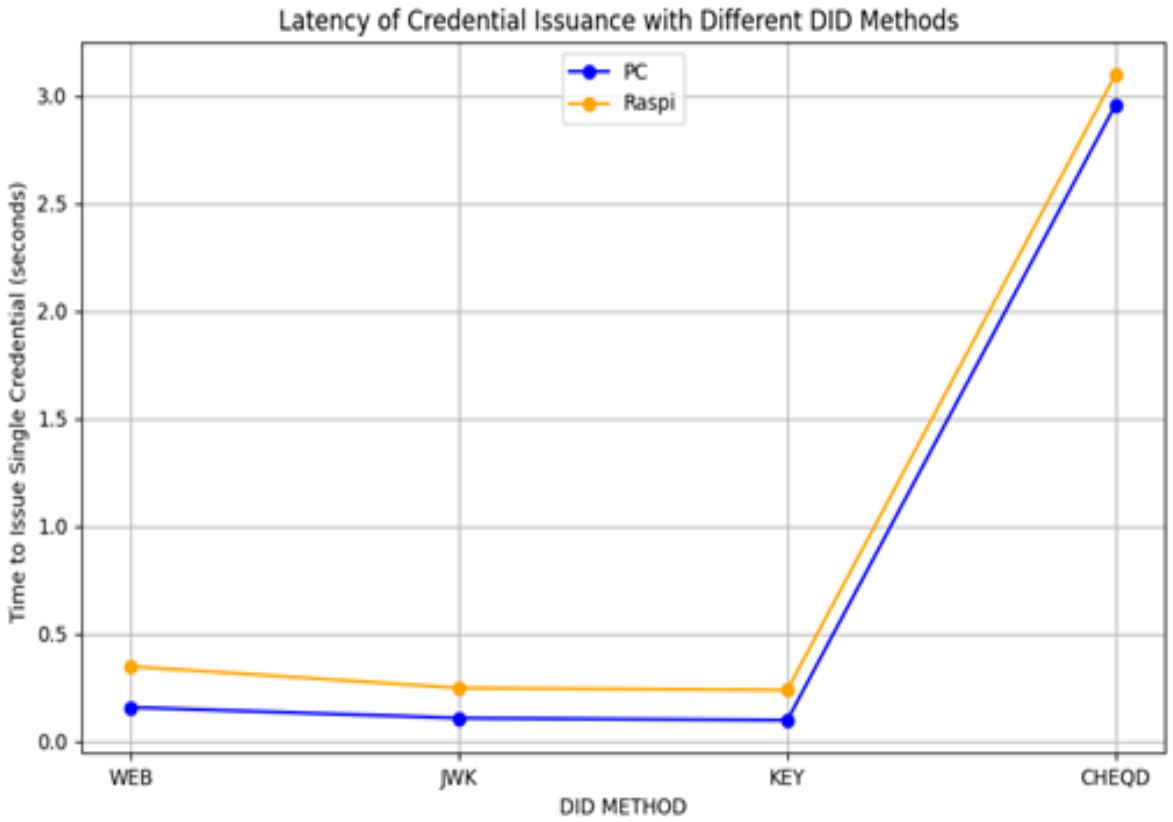
- A Raspberry Pi 4 and a x86 PC

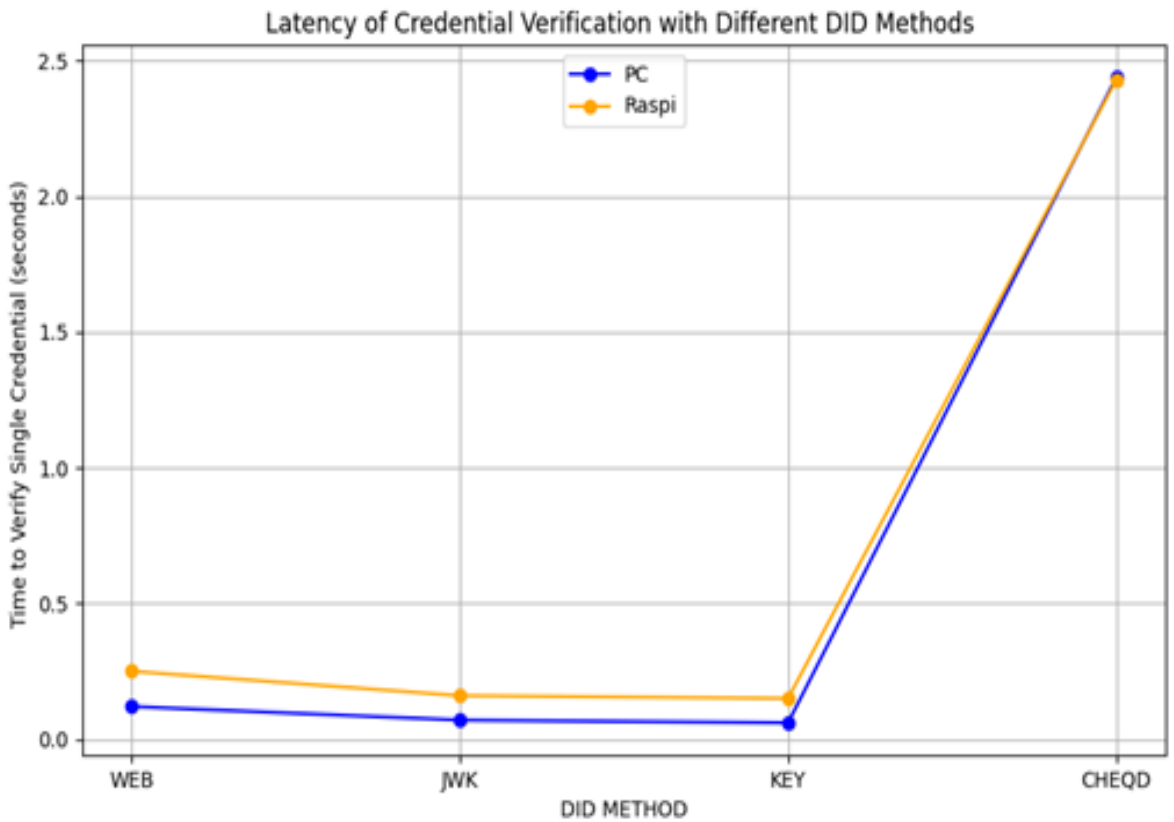
Raspberry Pi 4	X86 PC
Quad-core CPU (1.8 GHz max)	Quad-core CPU (3.4 GHz max)
8GB of memory	16GB of memory
Ubuntu Core 24	Ubuntu 22.04.3 LTS

- Connected to the internet via a DSL connection
- Walt.id identity repository version 0.3.1
- Python scripts to automate interactions with the Issuer, Verifier, and the Wallet services

Performance Measurements

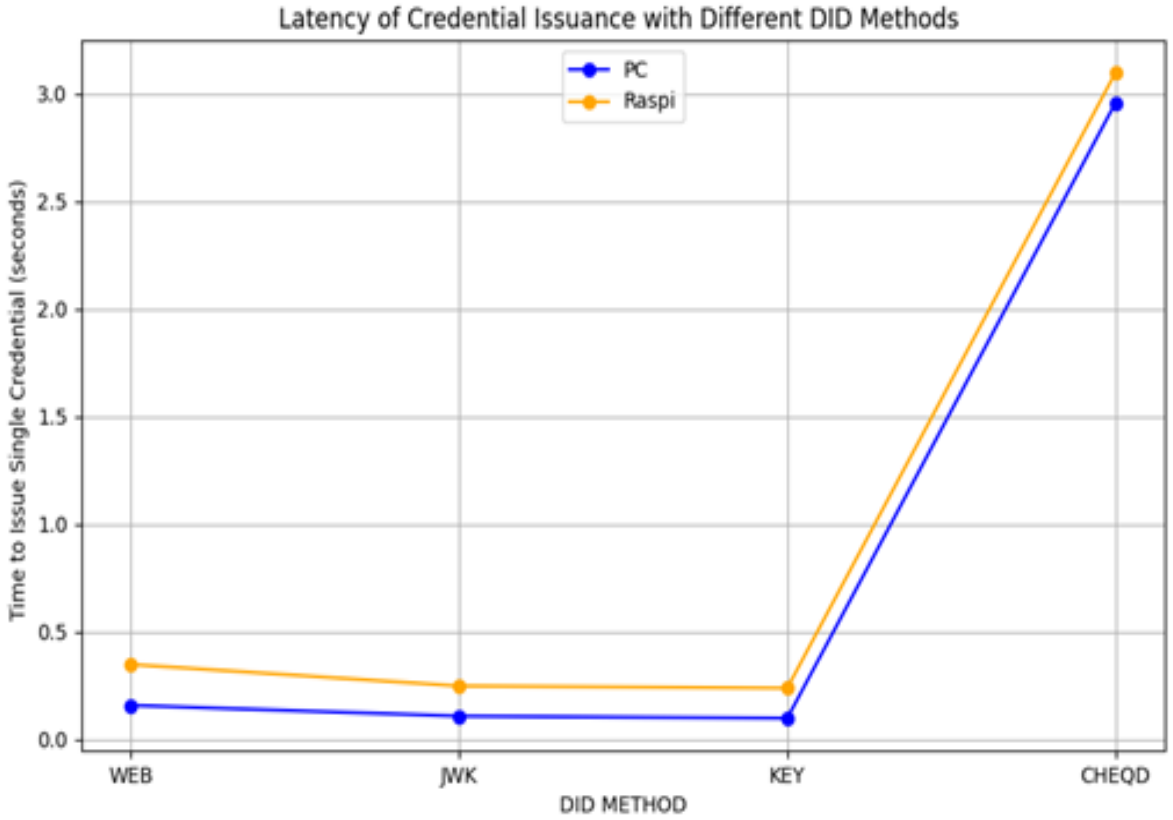
- **Latency Measurements for CHEQD, WEB, KEY, JWK**
  - VC issuance and VC verification separately
  - Repeated 50 times

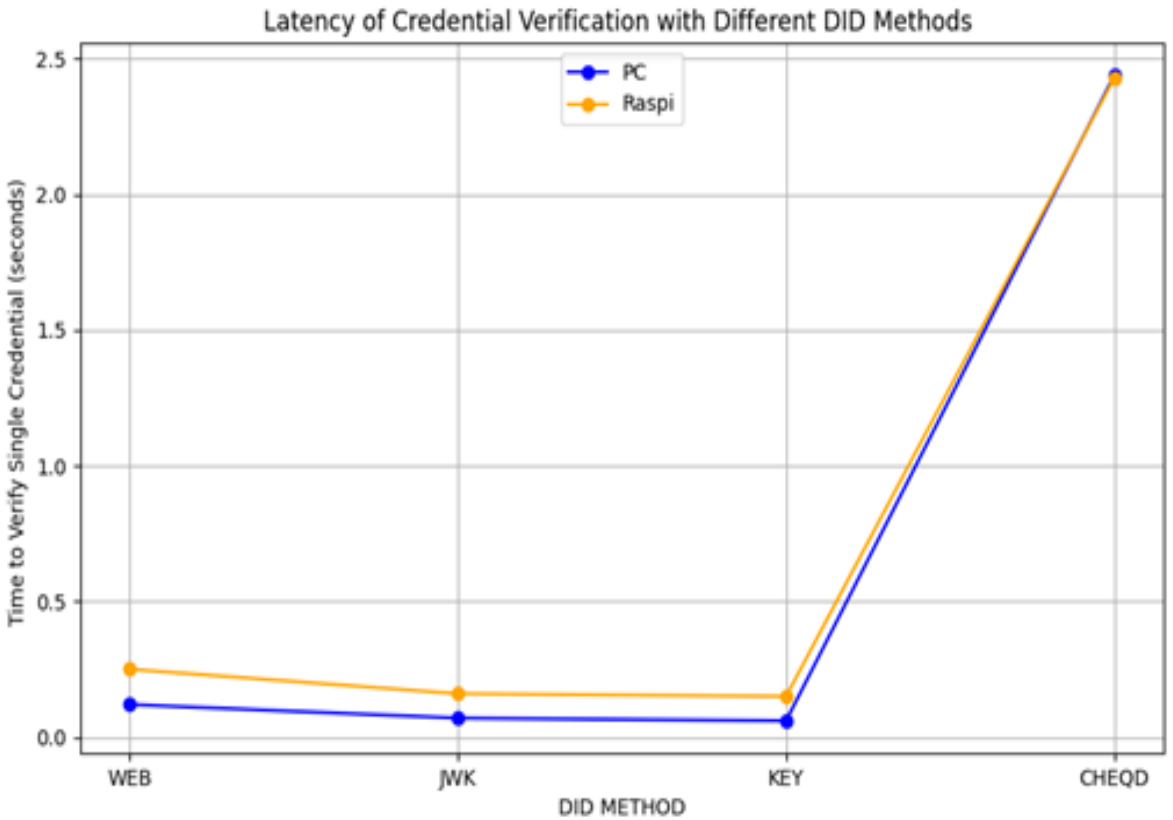




Performance Measurement (1)

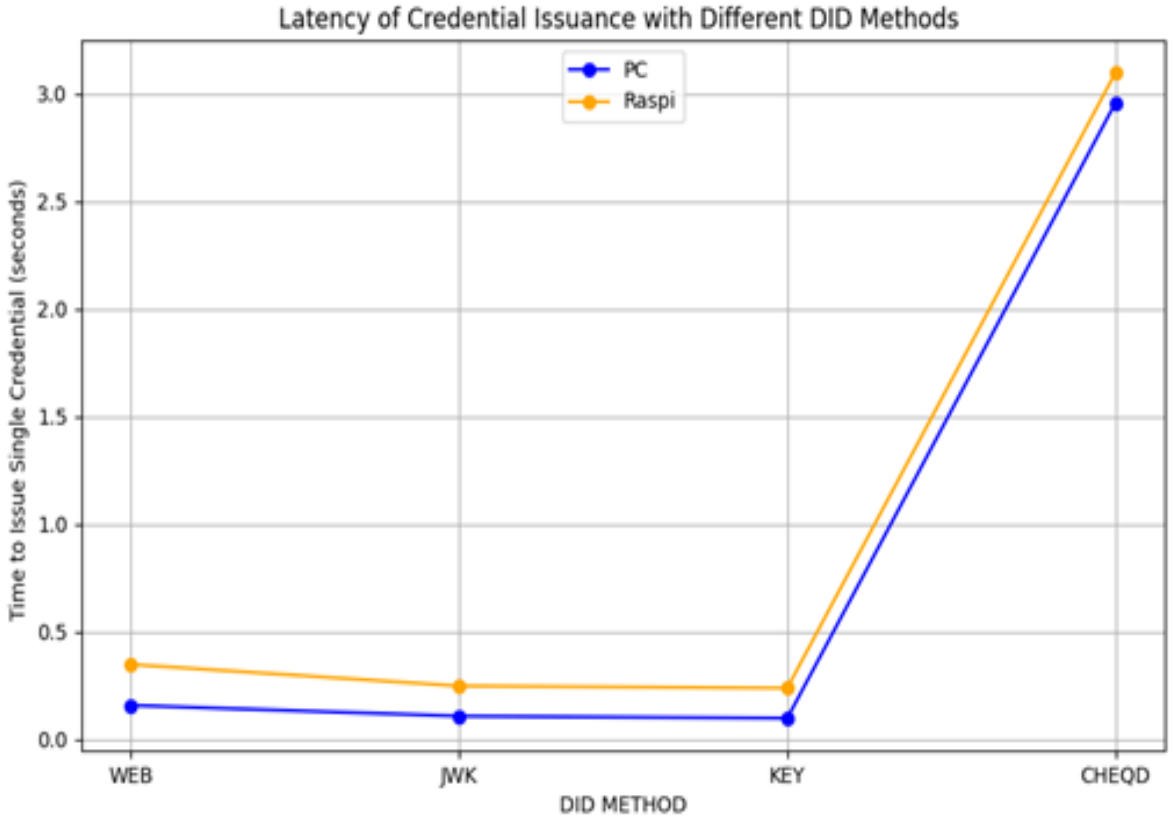
1. **CHEQD DID Method Latency**
   - Issuance and verification times are significantly higher for the CHEQD DID
   - DID document resolution with the Universal DID Resolver
     - Includes public keys in Blockchain

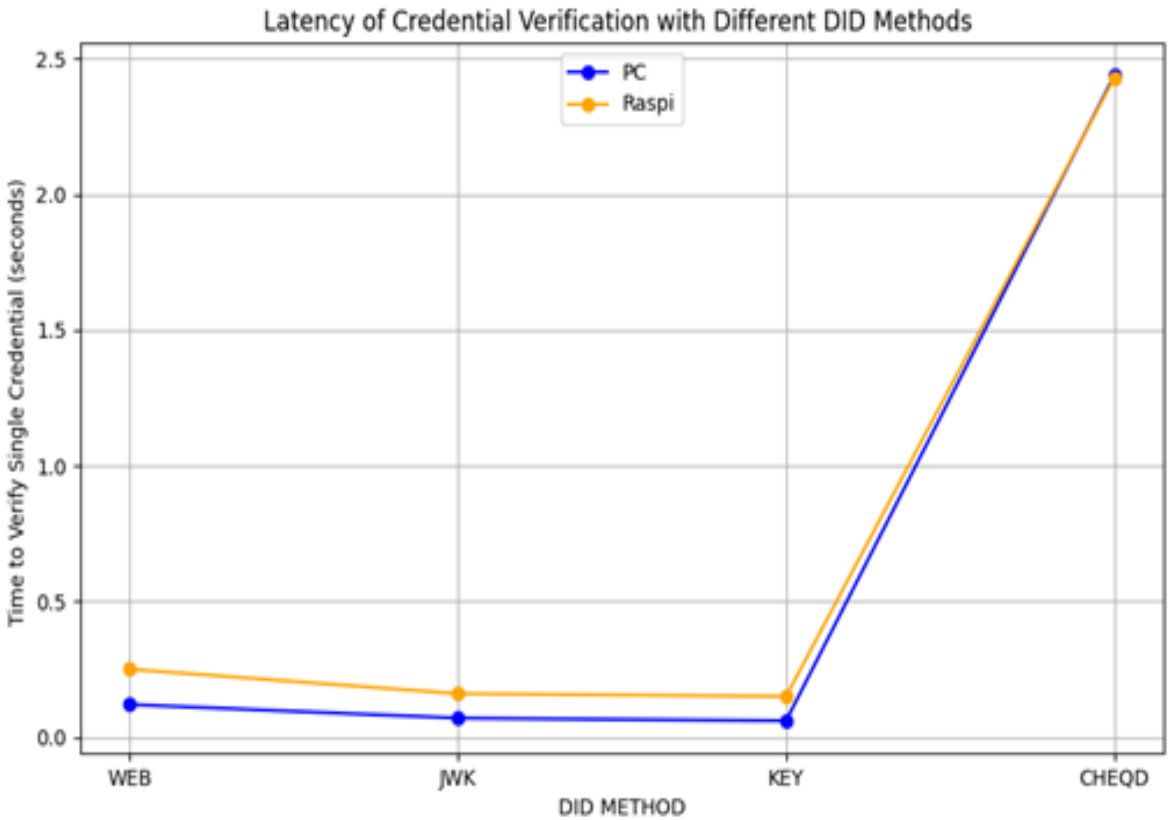




Performance Measurement (2)

2. **Raspberry Pi vs. PC**
   - For other DID methods (WEB, JWK, and KEY), latency on the Raspberry Pi is approximately twice as high as on the PC.
   - Likely due to the CPU performance disparity.





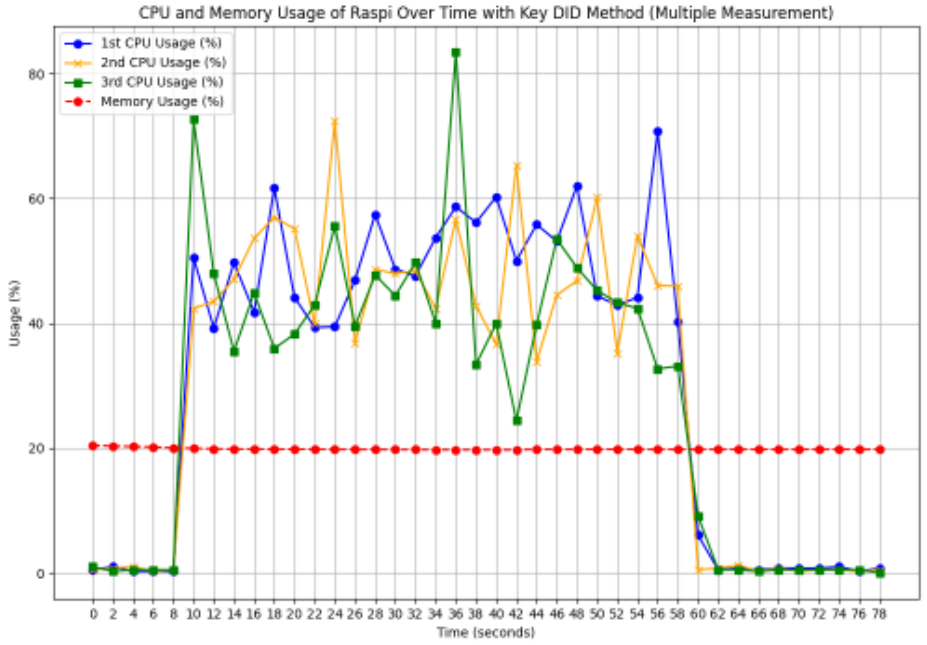
CPU and Memory Usage Measurement

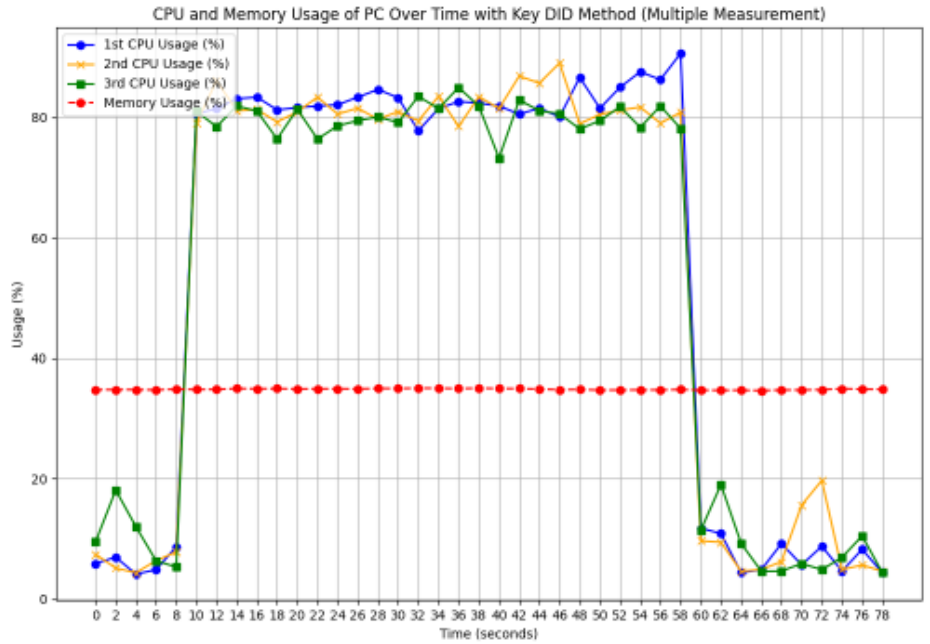
- Two DID methods: KEY and CHEQD
- Measurements were taken over an 80-second interval
  - Load phase is 60 seconds
- Repeated three times

Blue – the first measurement,
orange – the second measurement
green – the third measurement.
The memory is same, single red line.

DID KEY,

Ras Pi





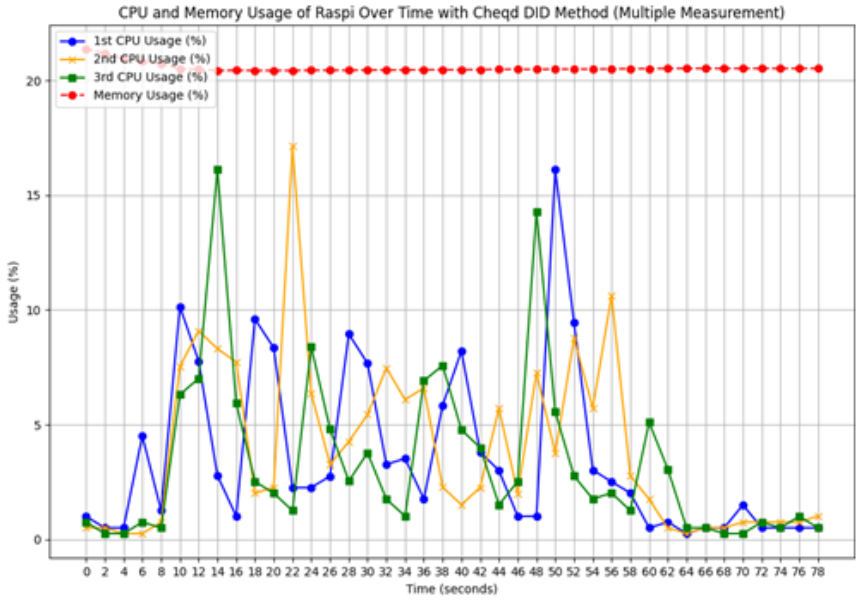
DID KEY,

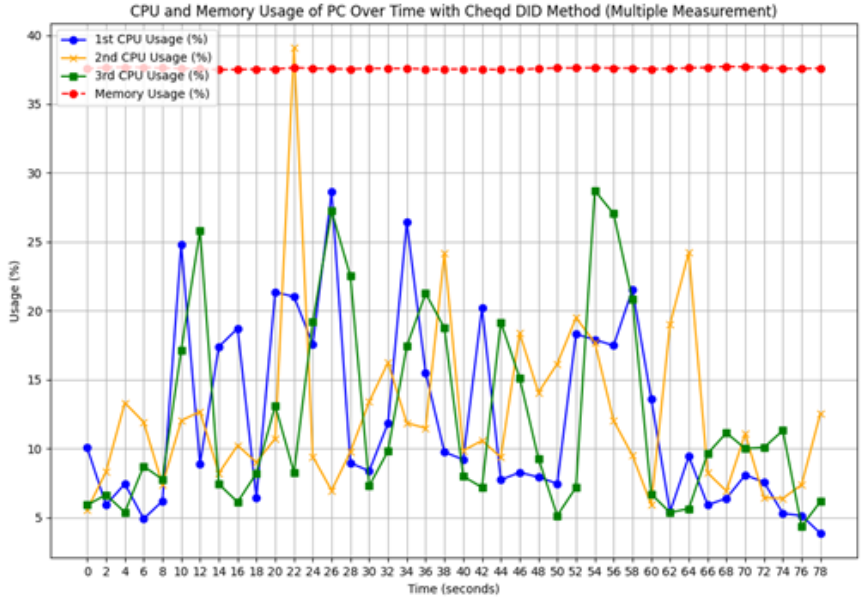
PC

CPU and Memory Usage Measurement (continued)

DID CHEQD,

Ras Pi





DID CHEQD,

PC

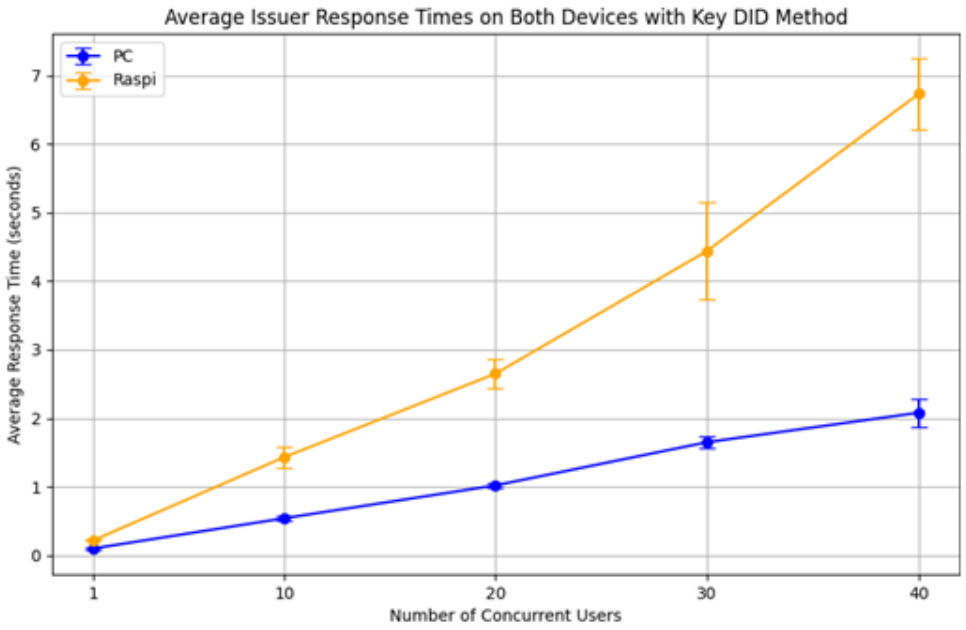
1. **MEMEORY USAGE**
   - No significant memory usage increase during the 60-second load phase
2. **KEY DID METHOD (last slide)**
   - The PC maintained stable CPU usage with minimal fluctuations
3. **CHEQD DID Method**
   - Both devices showed noticeable CPU usage spikes
   - Spikes are linked to the system's interaction with the external Universal Resolver for DID resolution

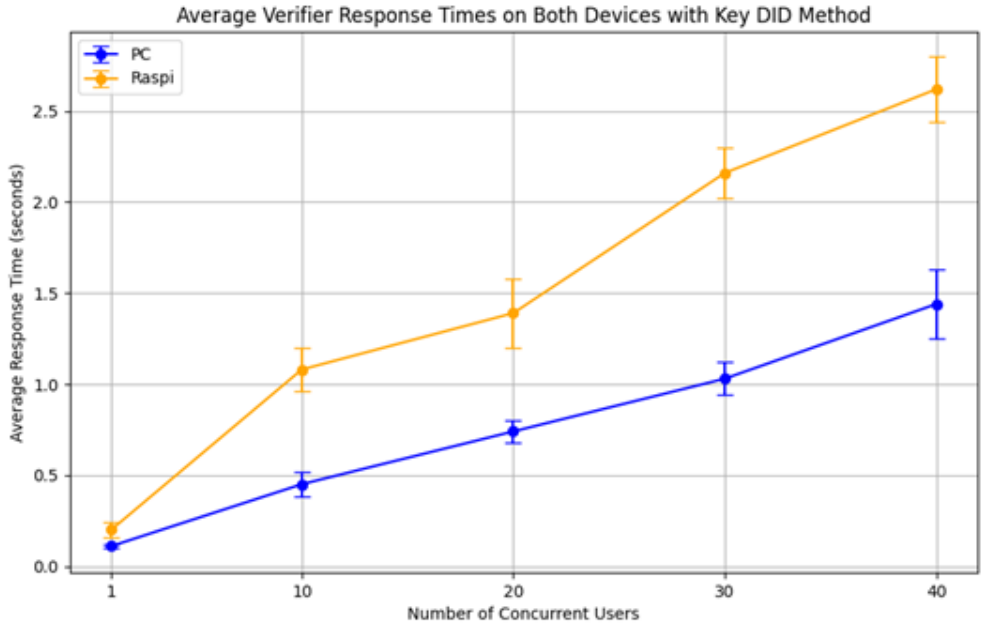
Scalability Measurements: DID Key

- Raspberry Pi experiences a larger proportional increase in issuer response time
- The proportional increase in verification response time is similar between them
  - Two different DID methods: CHEQD and KEY
  - Number of concurrent users — from 1 to 40
  - Repeated 10 times

Issuer DID KEY,

Ras Pi, PC





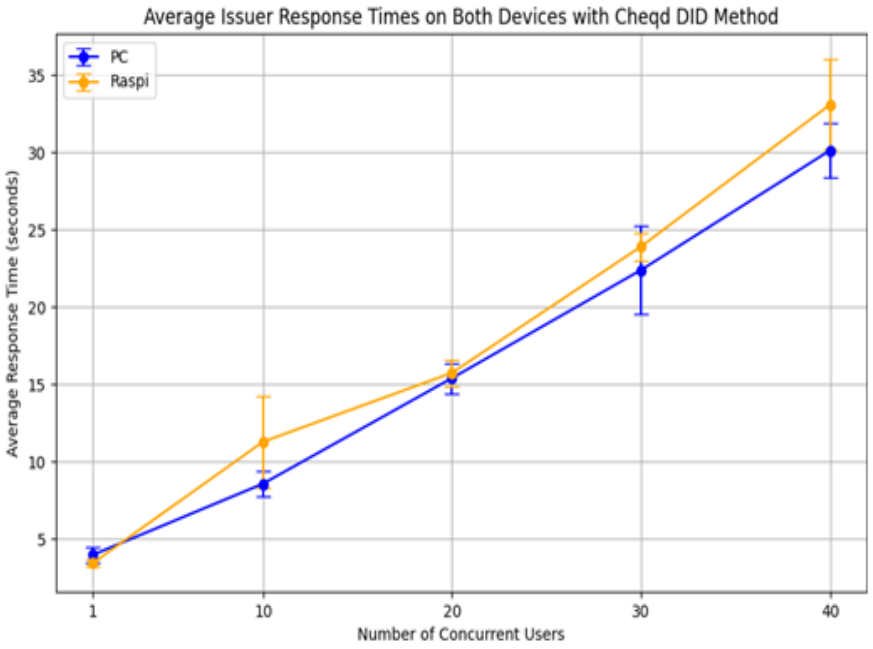
Verification DID KEY,

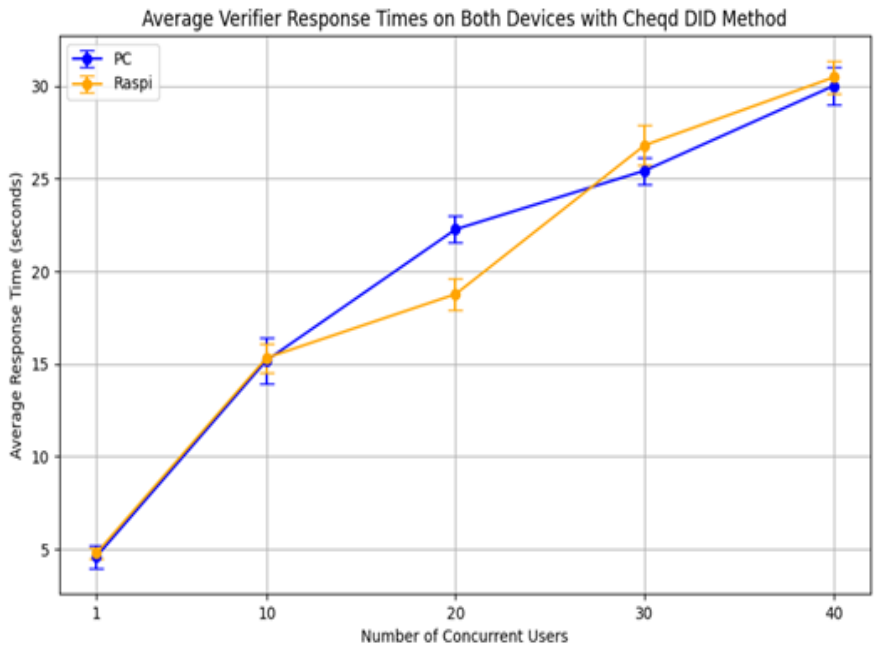
Ras Pi, PC

Scalability Measurements: DID CHEQD

Issuance DID CHEQD,

Ras Pi, PC





Verification DID CHEQD,

Ras Pi, PC

- CHEQD DID method similar scaling behavior for both devices.
- External DID resolution process dominates latency
  - diminishing the relative impact of hardware differences on performance.

Conclusions

- Issuance and verification significantly slower for CHEQD
  - Due to universal resolver and blockchain access
  - Related work similarly shows 1-3s latency
- PC and Raspberry Pi perform well for SSI
  - PC performance more stable in some cases
  - Higher scalability of PC for load scenarios (factor 2)
- Little difference from Ubuntu vs Ubuntu Core

Thank you for listening!