Fortified-Edge: PUF based Authentication in Collaborative Edge Computing

Presenter: Seema G. Aarella

Seema G. Aarella[1], Saraju P.Mohanty[2], Elias Kougianos[3] , Deepak Puthal[4]

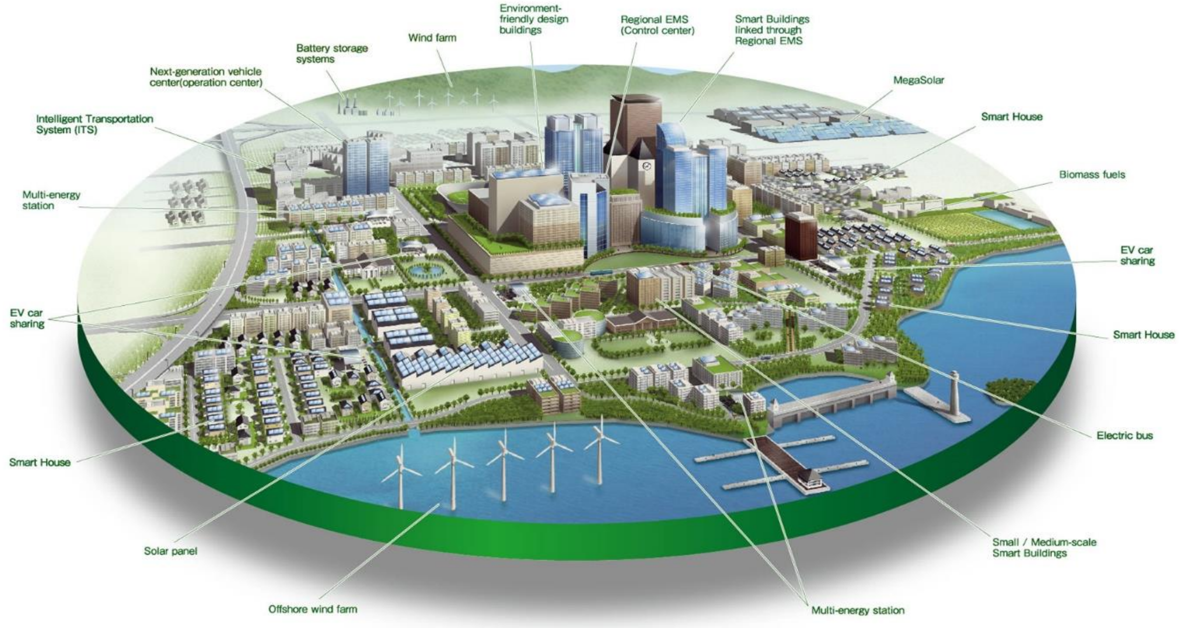
University of North Texas, Denton, TX 76203, USA.[1,2,3]
Khalifa University, Abu Dhabi, UAE.[4]

Email: Seema.Aarella@unt.edu[1], Saraju.Mohanty@unt.edu[2] and Elias.Kougianos@unt.edu[3], deepak.puthal@ku.ac.ae[4]

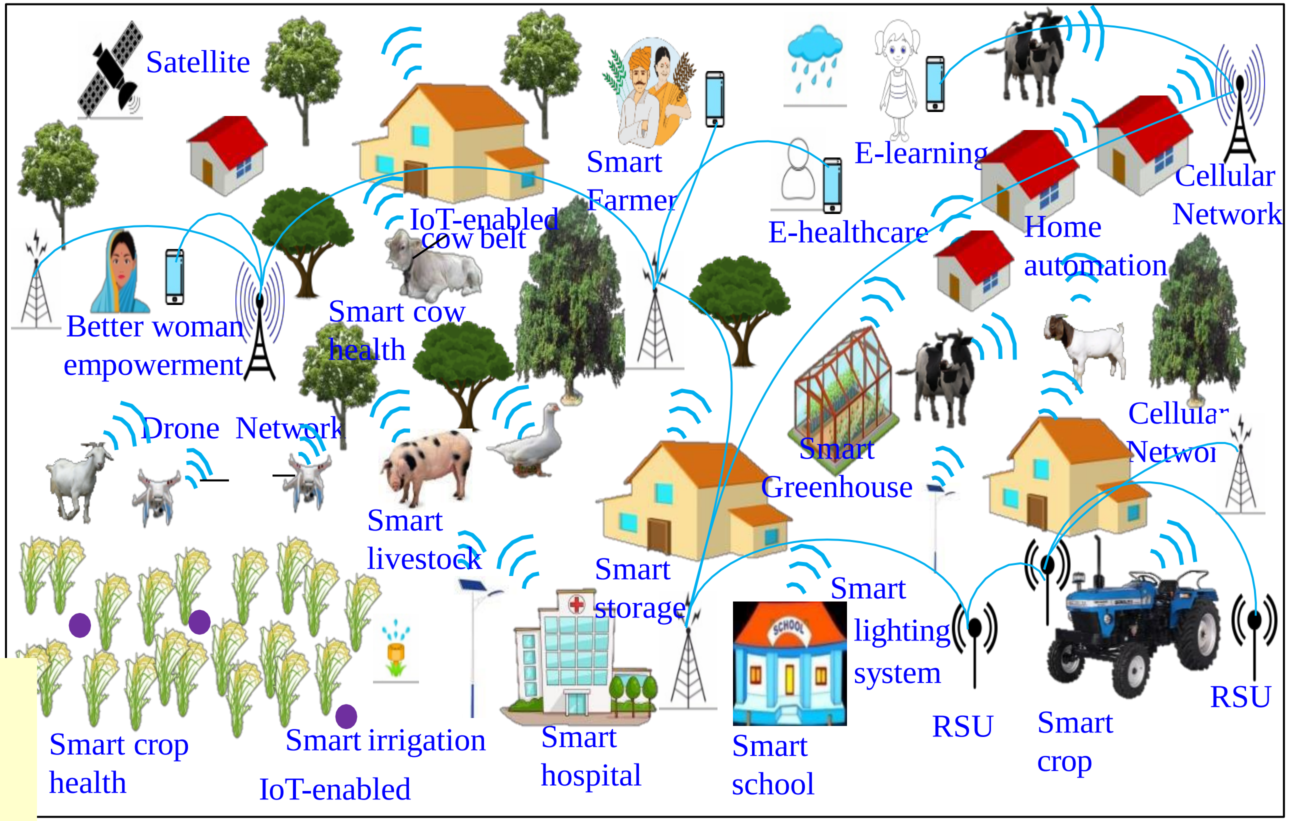
Outline of the Talk

- Introduction
- Smart Cities and Smart Villages
- Related Prior Research
- Collaborative Edge Computing for Smart Village
- Proposed PUF CA Method
- Authentication Algorithms
- Experimental Results
- Conclusion
- Future Research

Smart Cities Vs Smart Villages



Source: http://edwingarcia.info/2014/04/26/principal/



Source; P. Chanak and I. Banerjee, "Internet of Things-enabled Smart Villages: Recent Advances and Challenges," *IEEE Consumer Electronics Magazine*, DOI: 10.1109/MCE.2020.3013244.

Smart Villages
- CPS Types - Less
- Design Cost - Low
- Operation Cost – Low
- Energy Requirement - Low

Smart Cities
- CPS Types - More
- Design Cost - High
- Operation Cost – High
- Energy Requirement - High

Smart Village

Services

- Agriculture
- Irrigation
- Energy
- Livestock
- Healthcare
- Education
- Governance
- Transport

Smart Village

Technologies

- Internet
- Drone Technology
- 5G Technology
- Collaborative Edge Computing
- Green Energy
- Low Power Communication

Smart Village is a paradigm that brings Smart City technologies to the villages but with limitations

Challenges of Smart Village

- Environmental
- Technological
- Demographic
- Political
- Economic

Technological →

- High Energy Requirement
- Security and Privacy
- Latency and Bandwidth

Security-by-Design (SbD)

- Integration of the cybersecurity early in the design phase, not retrofitted
- Device, circuit, and system-level cybersecurity solutions for robust CPS and smart component design

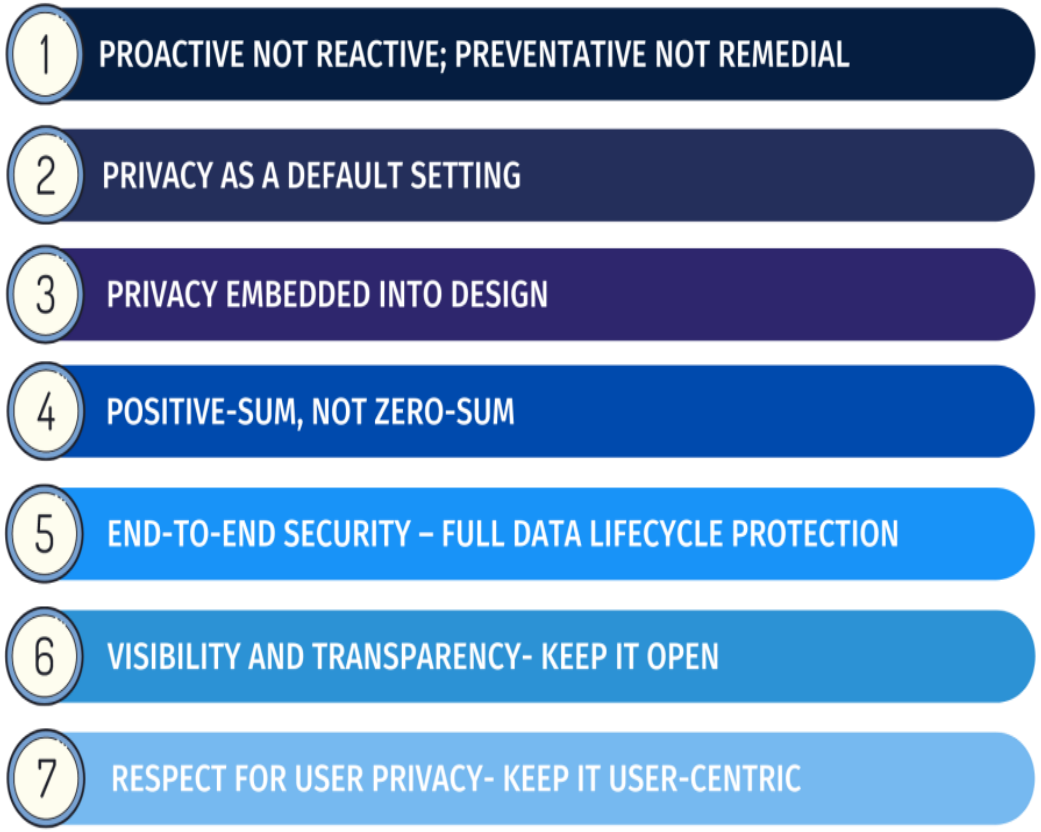
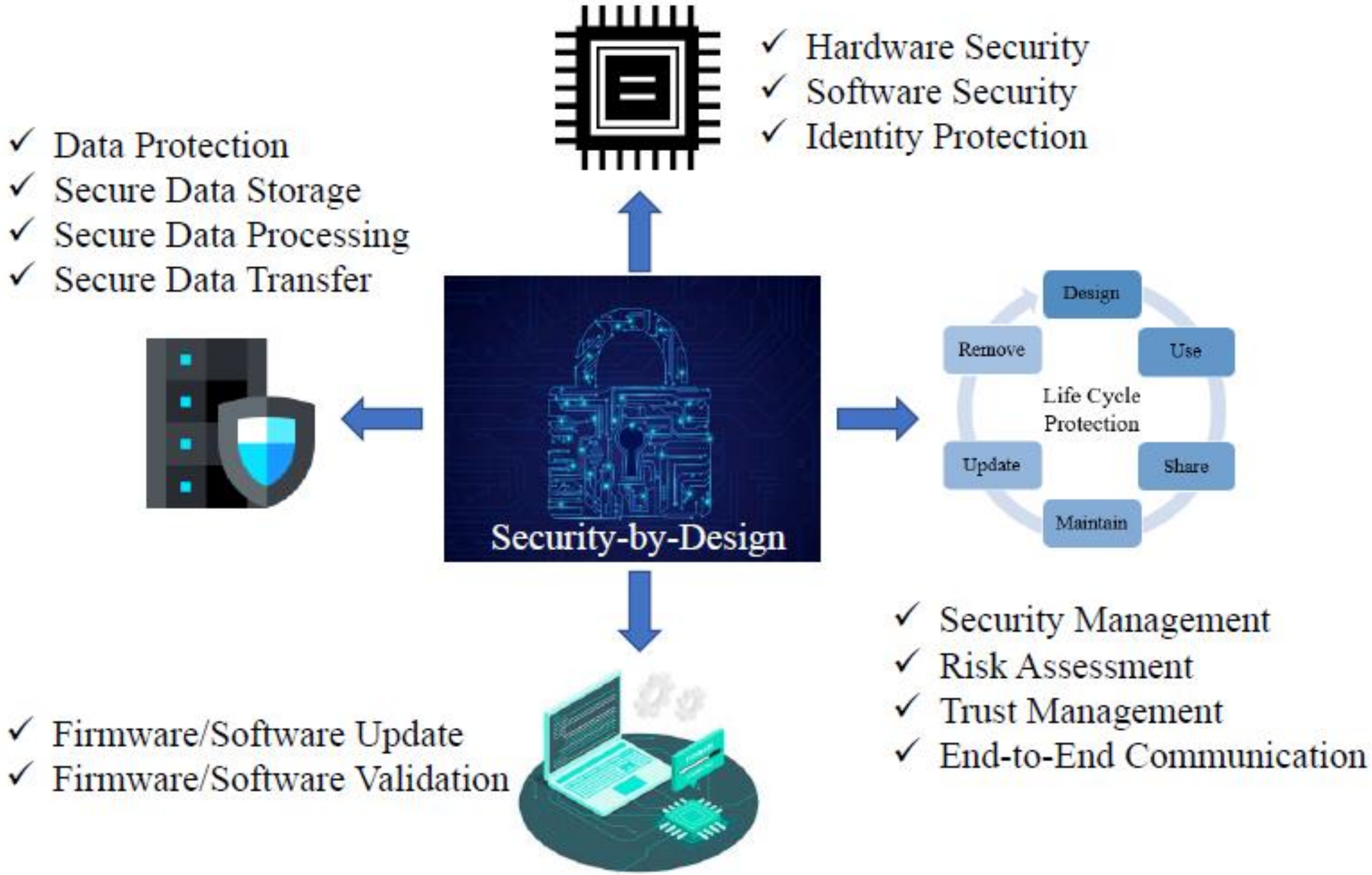


Image Source: https://dataprivacymanager.net/seve-principles-of-privacy-by-design-and-default-what-is-data-protection-by-design-and-default/

Security-by-Design (SbD)



Why SbD?

- The generalization of attacks across all CPS typically ignores the role of Root-of-Trust (RoT) and security perimeter modeling, which are the basis of many SbD approaches

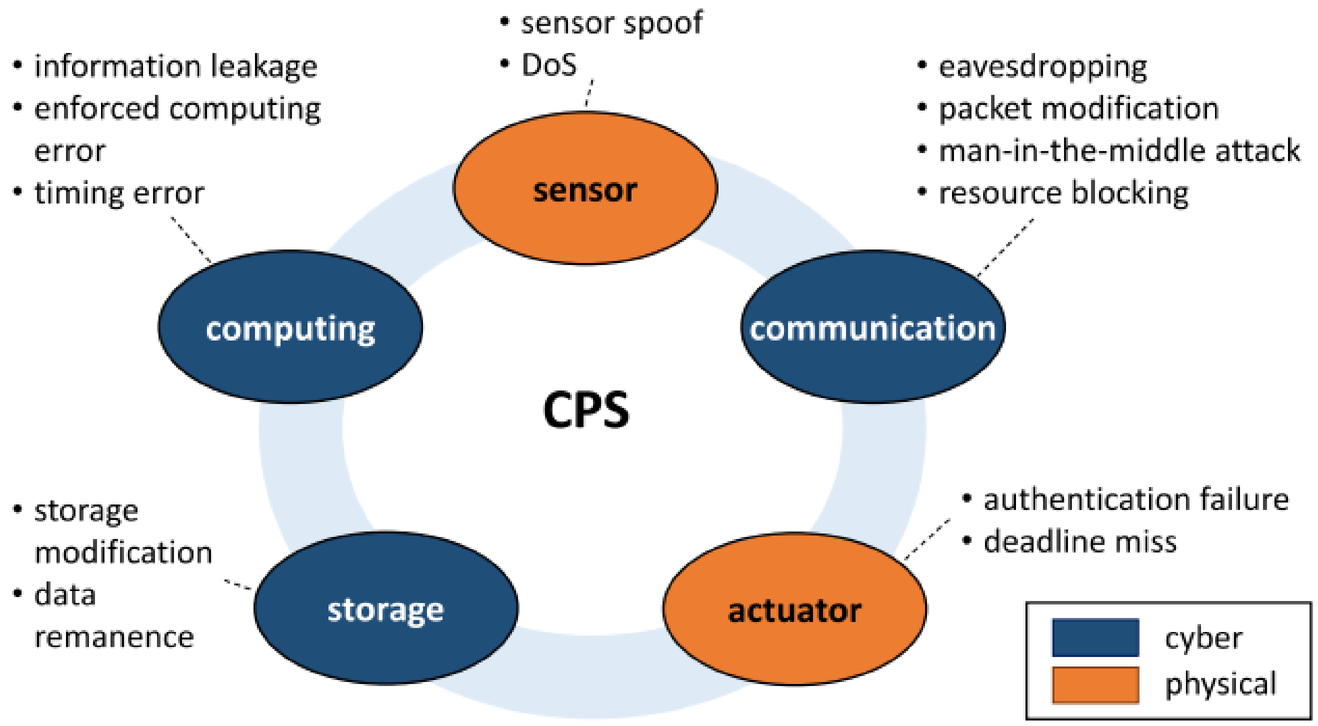
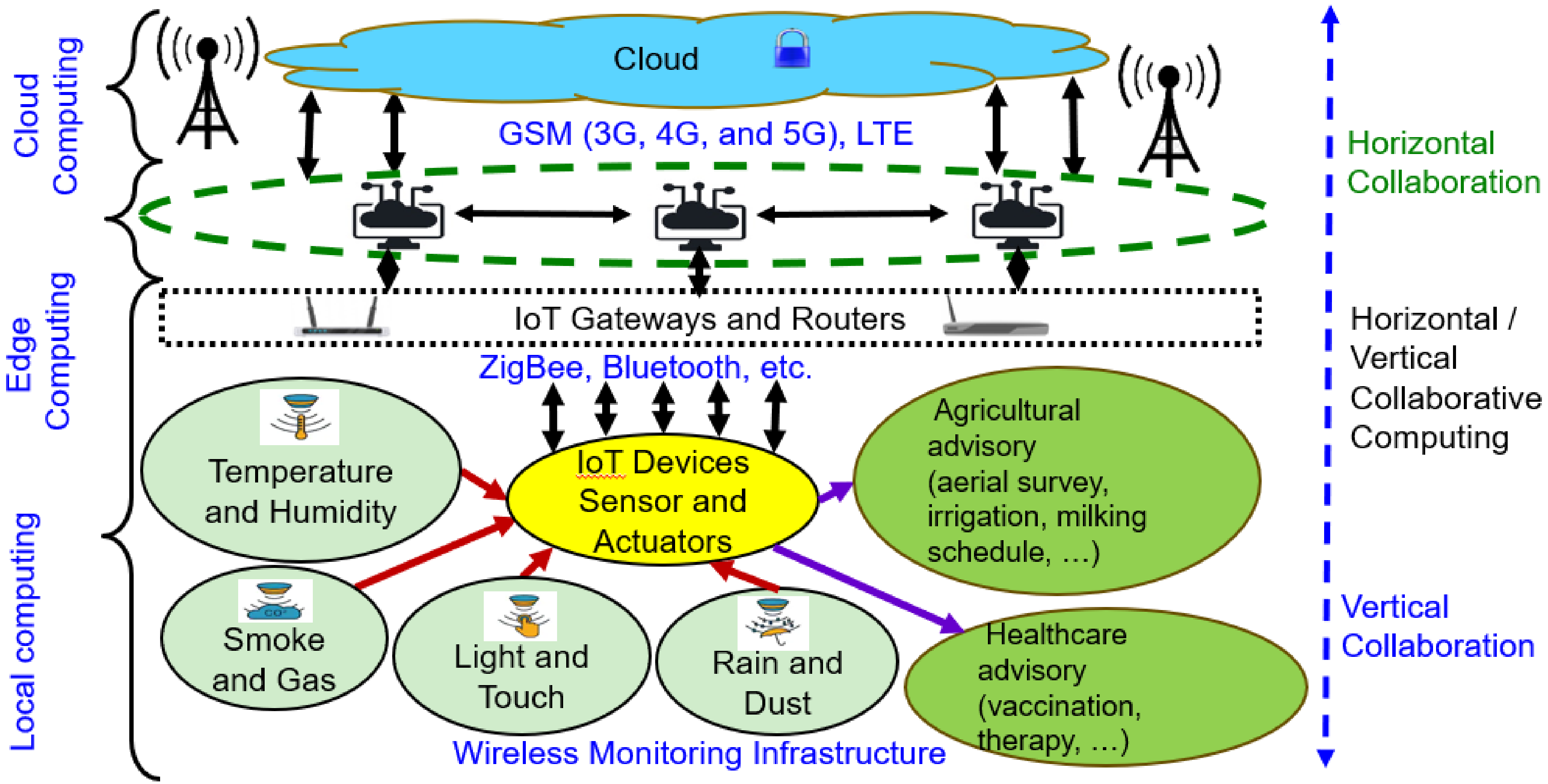


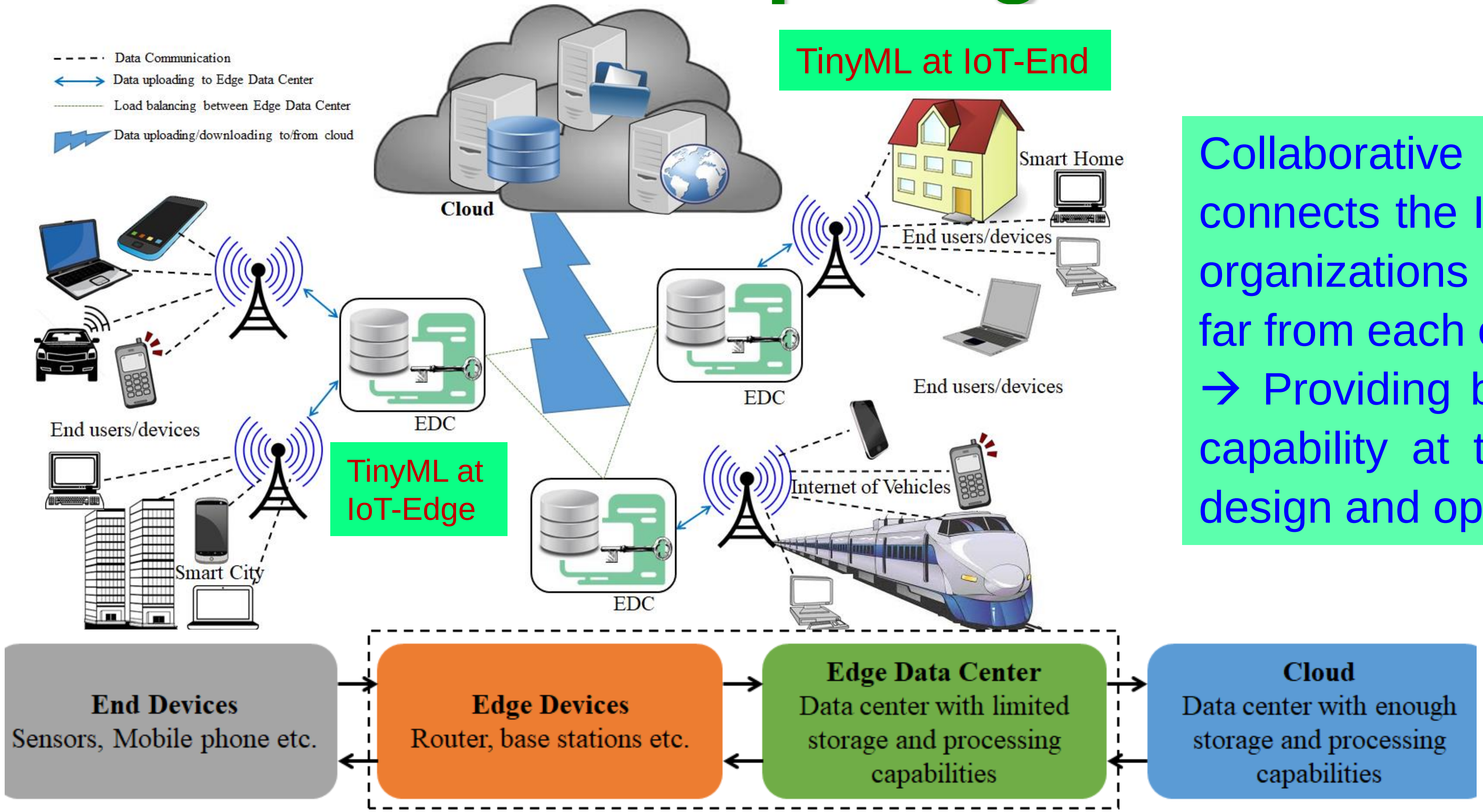
Image Source: A. Chattopadhyay, K. -Y. Lam and Y. Tavva, "Autonomous Vehicle: Security by Design," in IEEE Transactions on Intelligent Transportation Systems, vol. 22, no. 11, pp. 7015-7029, Nov. 2021, doi: 10.1109/TITS.2020.3000797.

Collaborative Edge Computing (CEC)



Source: D. Puthal, S. P. Mohanty, S. Wilson and U. Choppali, "Collaborative Edge Computing for Smart Villages", *IEEE Consumer Electronics Magazine (MCE)*, Vol. 10, No. 03, May 2021, pp. 68-71.

Collaborative Edge Computing is Cost Effective Sustainable Computing for Smart Villages

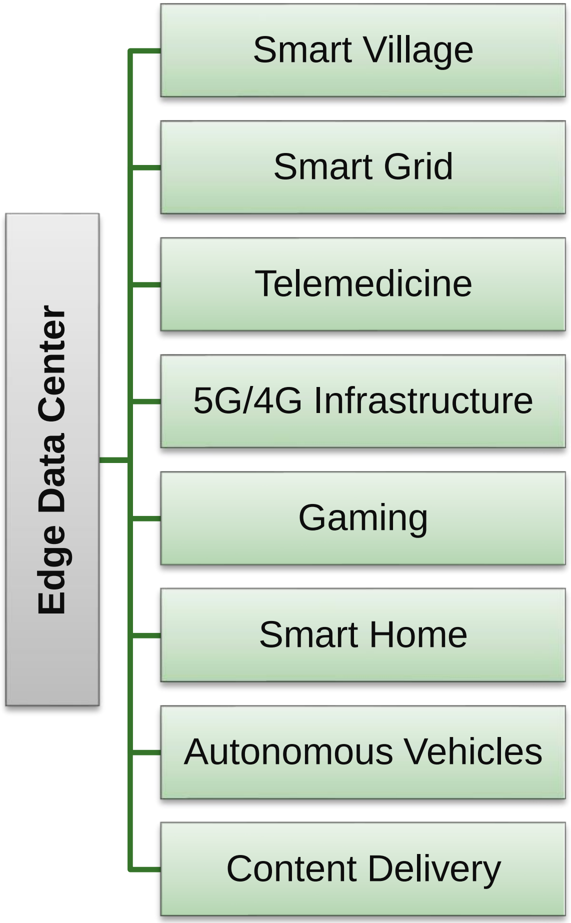


Collaborative edge computing connects the IoT-edges of multiple organizations that can be near or far from each other
→ Providing bigger computational capability at the edge with lower design and operation cost.

Source: D. Puthal, M. S. Obaidat, P. Nanda, M. Prasad, S. P. Mohanty, and A. Y. Zomaya, “Secure and Sustainable Load Balancing of Edge Data Centers in Fog Computing”, *IEEE Communications Mag*, Vol. 56, No 5, May 2018, pp. 60--65.

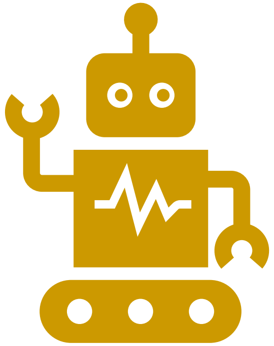
Collaborative Edge Computing (CEC)

- Collaborative Edge Computing is a distributed processing environment
- CEC is a collaboration of distributed edge
- Smart control of heterogenous network
- Reduced Bandwidth and Transmission costs
- CEC enables seamless processing through load balancing



Long-term Vision

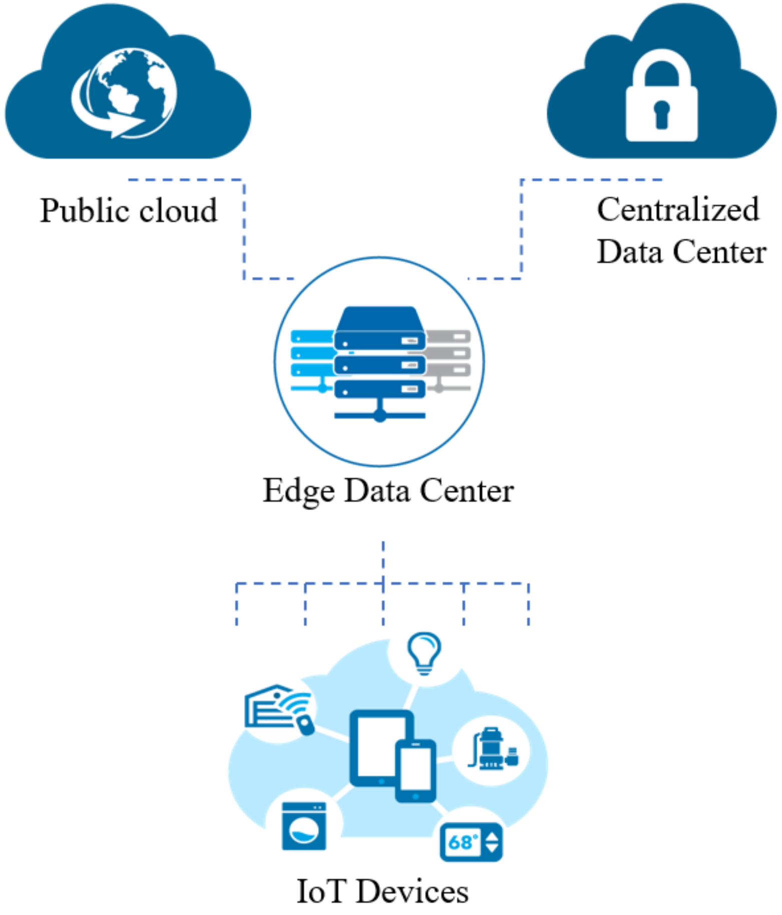
Cybersecurity for smart villages based on the SbD principles for secure resource sharing in the CEC environment



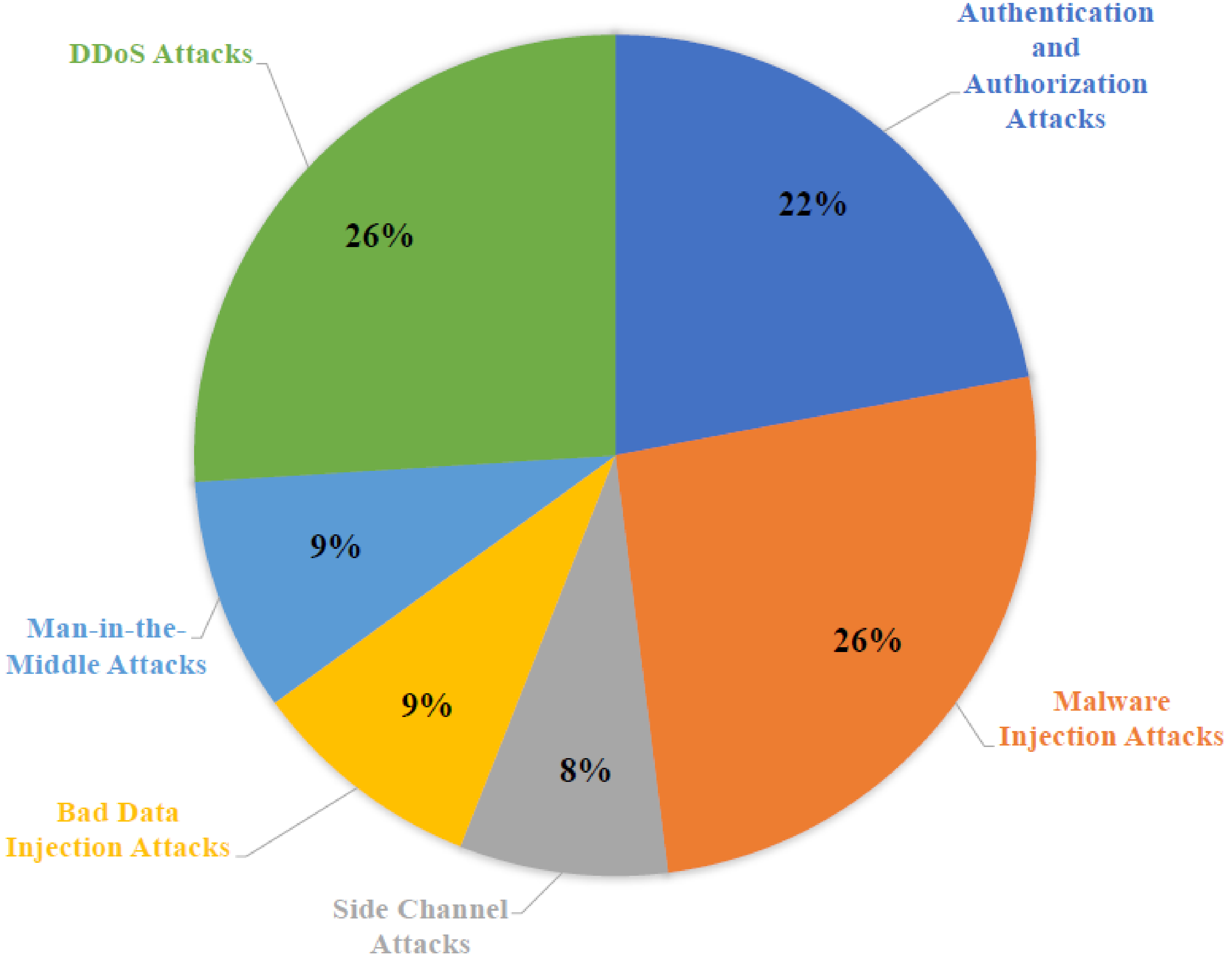
AI/ML for Cybersecurity in Smart Villages

Edge Data Center (EDC) in CEC

- Secure authentication for Load balancing
- Edge Data Centers participate in Load Balancing
- EDCs are deployed at different geographical locations
- Lightweight and secure authentication for EDCs
- Cloud Based Authentication causes latency issues
- Risk of Single-point-of-failure



Need for Secure Authentication of EDC



Existing Solutions

Symmetric and Key Cryptography

- Advanced Encryption Standard(AES)
- Client & server store a secret key

Asymmetric Key Cryptography

- Transport Layer Security(TLS)
- Secure Sockets Layer(SSL)
- Public Key and Private Key Pairs

Device Localization and Environmental data authentication technique

PUF based authentication techniques

Issues and Solutions

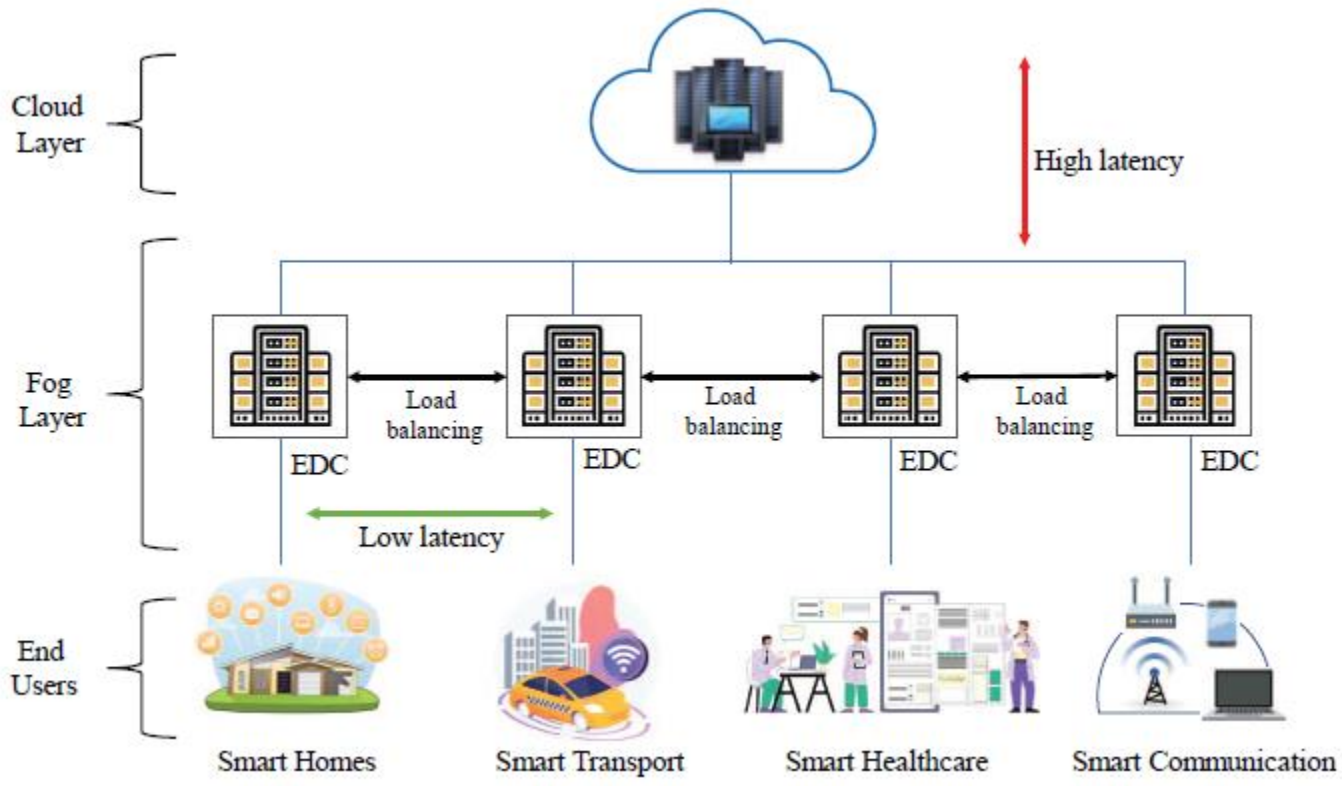
Issues
- High Computation
- High Maintenance Cost
- High Communication Cost
- Cloud Based (High Latency)

Solutions
- Lightweight
- Low Power
- Low Computation
- Edge Based (Low Latency)

Related Prior Research

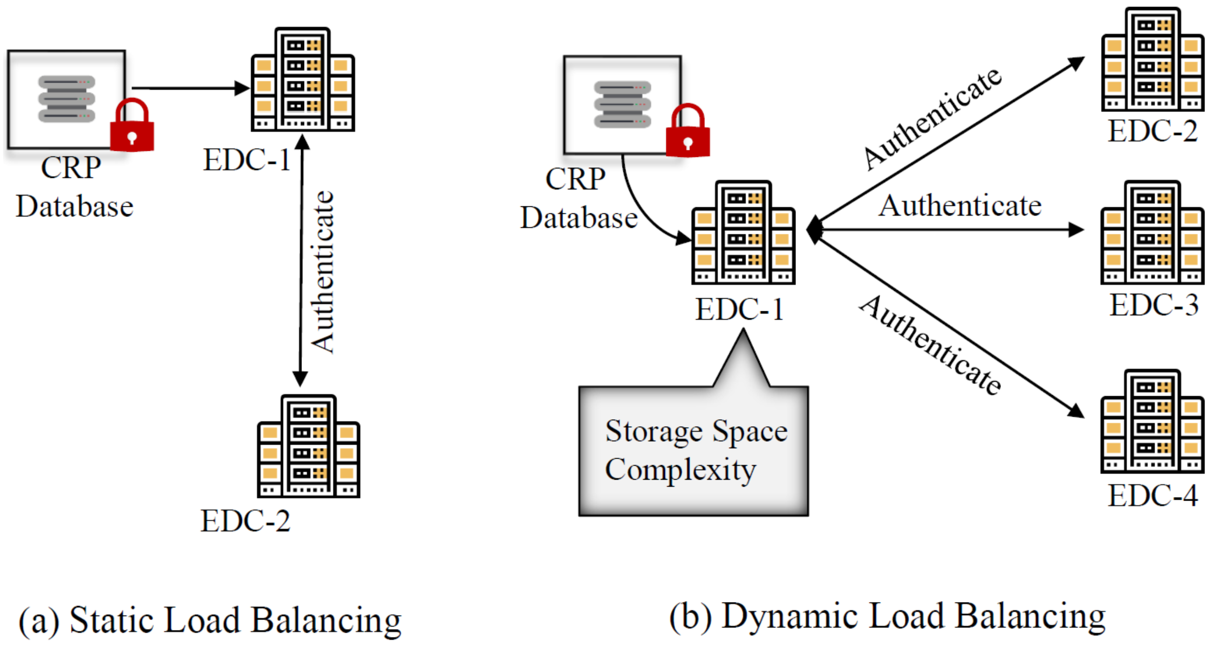
Research	Algorithm	Application
Puthal, et al. [17]	AES-based Symmetric Encryption	Authentication and Load Balancing of EDCs
Barbareschi, et al. [3]	PUF based PHEMAP	Fog-IoT Systems
Hathal, et al. [8]	TA, TESLA	Vehicular Communication Systems
Li, et al. [13]	p-KNN	SND-based Edge Computing for Healthcare Systems
Zhang, et al. [25]	SRAM PUF and Blockchain	Multiserver Authentication in Cloud-Edge IoT
Puthal, et al. [15]	Decision Tree	Data aggregation and PoAh for Blockchain in IoT Edge
Fortified-Edge (Current Paper)	SRAM PUF based CA	Edge Data Center Authentication in CEC

Load Balancing in Edge Data Centers



PUF based Authentication

- Storage Space Complexity
- Data Security against data breaches and attacks
- Need for Root-of-Trust
- Faster Authentication Protocols



Certificate Authority

- A Certificate Authority (CA) is a trusted resource that issues Secure Socket Layer (SSL) digital certificates which are a part of the Public Key Infrastructure (PKI)
- CAs help maintain trust between communicating entities over the internet
- The CA helps build a Root-of-Trust between the connected devices in the environment
- A centralized CA will be prone to single-point failure whereas a more distributed CA will provide flexible effective security management

Problems Addressed

Need for robust, secure and lightweight authentication scheme with low computational power

Authentication without Cloud Server to address latency issues

Lightweight and Low latency protocol for mutual authentication of EDCs

Solving the storage space complexity when storing CRP databases that are involved in PUF-based authentication schemes

Novel Contributions of Current Research

CA-based authentication to overcome the need for storing CRP databases in the EDCs

Reducing the storage space requirement at EDC, enhancing data security

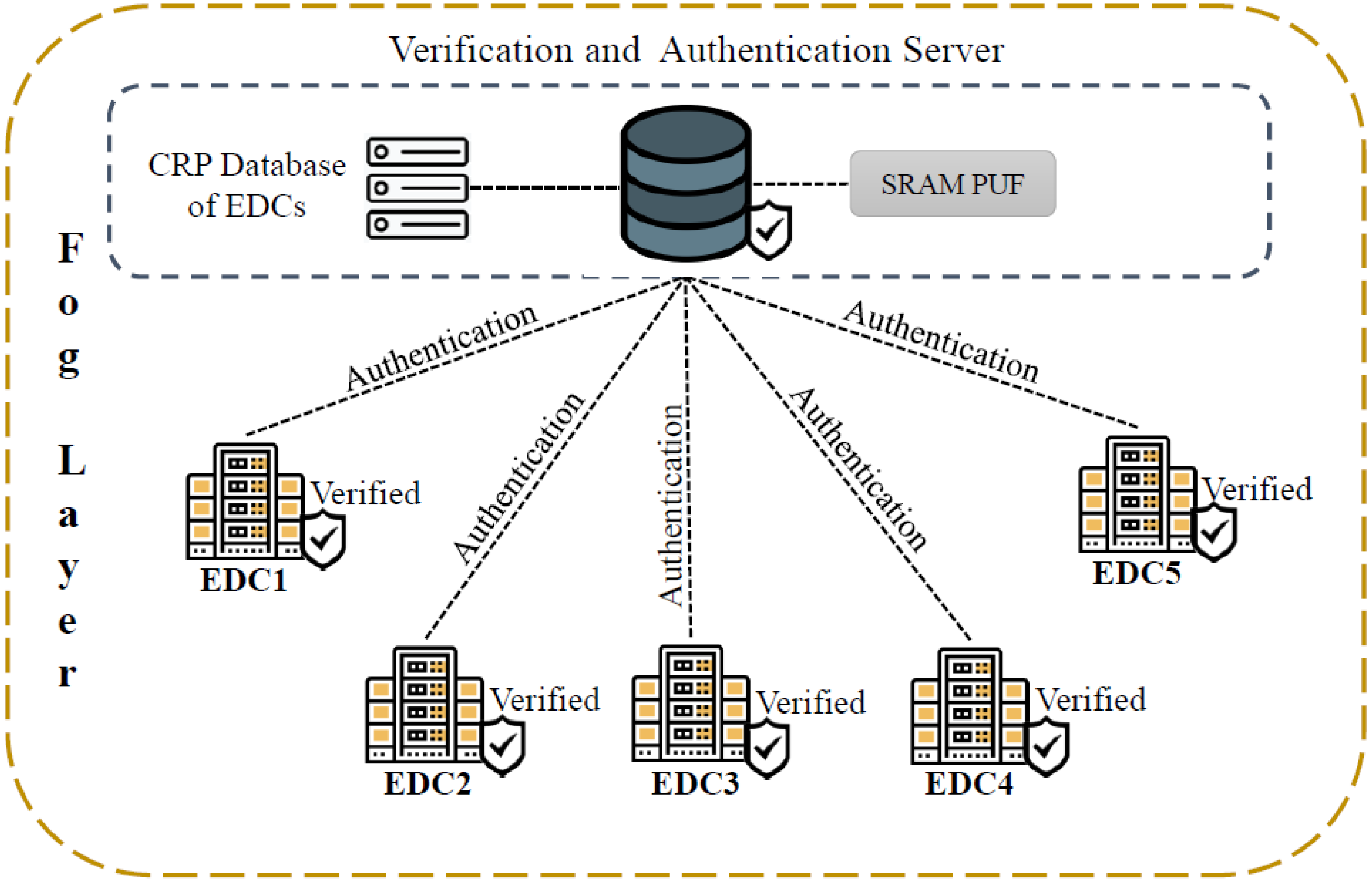
PUF as the lightweight, robust and secure mode of key generation

EDC mutual authentication scheme during load balancing using SRAM PUF for CA

Proposed Solutions

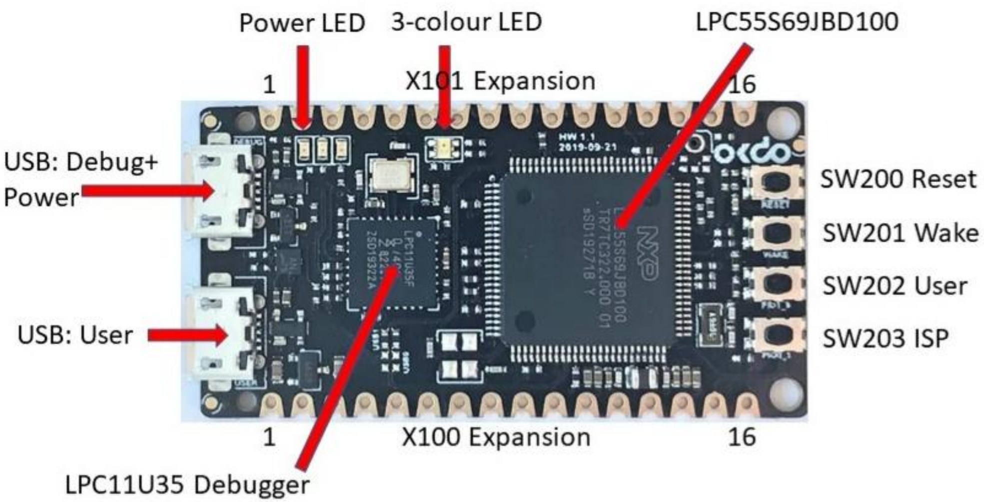
- Edge server-based architecture for EDC verification and authentication
- A mutual authentication scheme for client-client authentication
- Removing the need for storing a CRP database locally at the EDC
- SRAM PUF-based certificate generation to establish the root of trust between EDCs
- Mutual authentication scheme based on certificates

Architecture of the proposed PUF based CA scheme



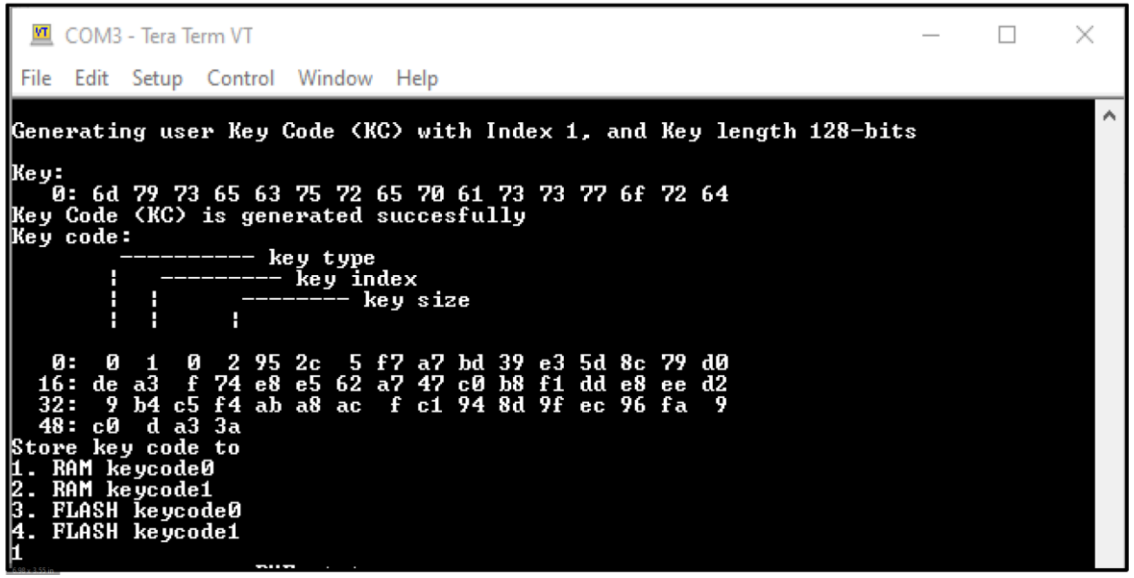
SRAM PUF

- Okdo E1 Development board used as SRAM PUF Module
- MCUXpresso IDE
- LCPXpresso55S9 SDK
- Tera Term – SSH Terminal
- Digital fingerprint is generated when the SRAM PUF is powered up
- The 256-bit key is the root key used for the encryption/decryption of user keys



Certificate Generation

- Digital fingerprint of the SRAM PUF are the responses, stored in the CRP database of the Verification and Authentication Server
- Keycode:
  - 32-bit Key Header
  - Key Index ranging from 0-15
  - Key size ranging from 64-bits to 4096 bits
- SRAM PUF start up data along with activation code generates PUF key



Parameters

- Digital Certificate is generated which includes the following information:
  - Certificate Version - C_v
  - Certificate Serial Number - C_s
  - Issuer ID - C_i
  - Validity Period with Timestamp - C_d
  - Edge Data Center ID - E_i
  - Digital Signature - D_s

Algorithm for PUF Certificate

Algorithm 1: Algorithm for Server Verifying EDC and Sending Certificate.

Input: Recieve EDC ceritification request with payload
Output: Verify EDC and send Certificate from Authentication Server

Client request recieved ;
get MacID ;
if $MacID_c = MacID_s$ **then**
    EDC is Identified;
else
    EDC is NOT Identified ;
    Registration NOt Successful ;
Send random challenge C_r based on EDC ID ;
Get PUF response R_p ;
if $R_p \neq R_s$ **then**
    EDC is NOT Authenticated ;
    Registration NOt Successful ;
else
    Registration Successful ;
    Generate Certificate ;
    Create hashString = (C_v,C_s, C_i,C_d,E_i, D_s) ;
    Compute hash (hashString$'$);
    Generate Private Key P_k ;
    Create Digital Signature = (hashString$'$ + P_k) ;
Send Digitally signed Certificate to EDC ;
/* The Certificate Authority module will generate the authentication certificate and send it to the EDC to store. */

Algorithm for Mutual Authentication

Algorithm 2: Algorithm for EDCs Mutual Authentication during load Balancing.

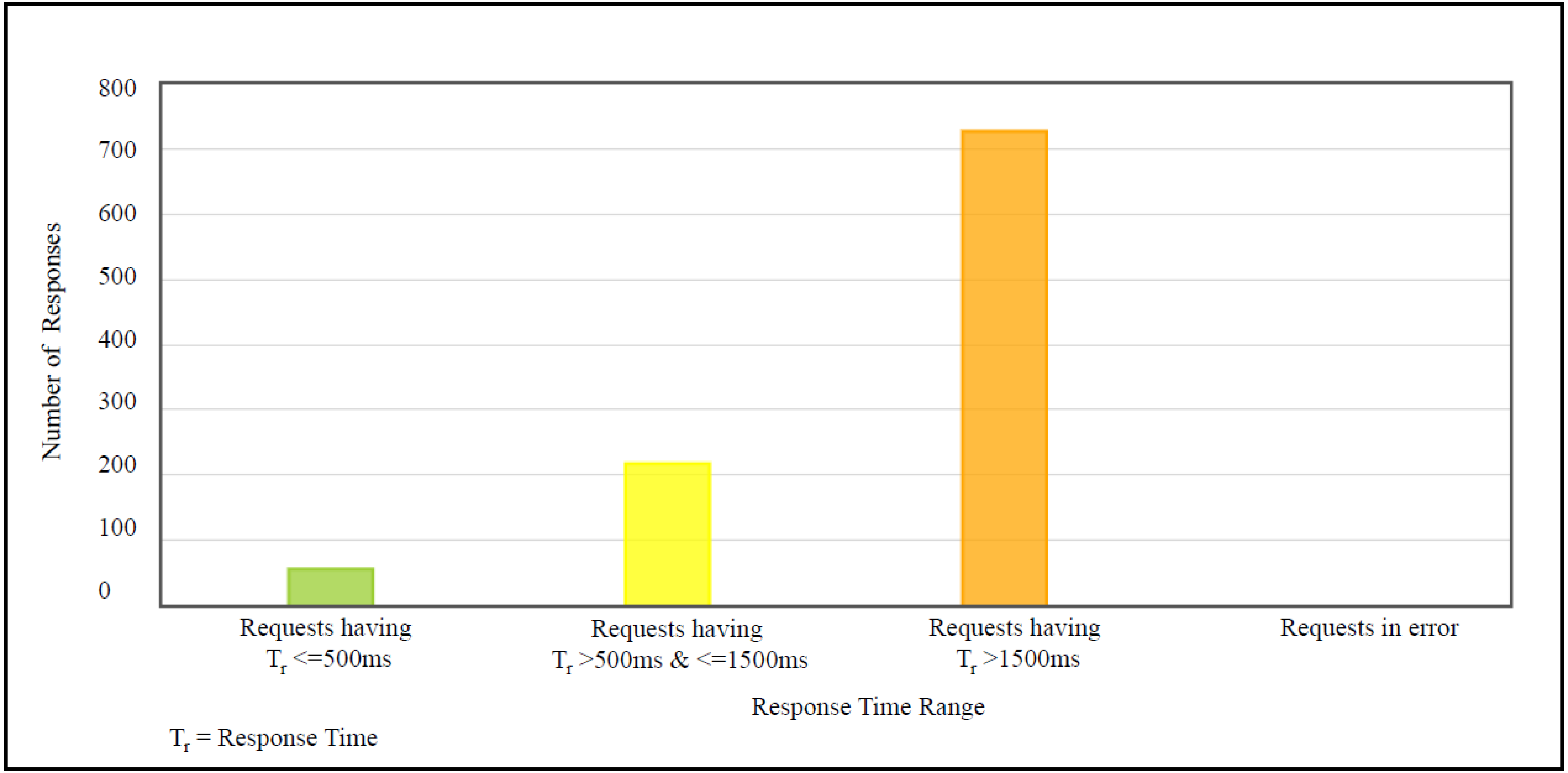
Input: Recieve Authentication request from EDC with payload
Output: Authenticate the EDC based on Certificate

```
Authentication request recieved ;
Send Certificate ;
Get Certificate ;
Check validity Period ;
if Valid then
    Get Public Key Pu ;
    Verify Digital Signature = (hashString' + Pu);
if Verified then
    Successfully Verified ;
/* The EDCs will exchange the cerificates verify the
   validity period and digital signature, if it is
   valid they participate in load balancing        */
```

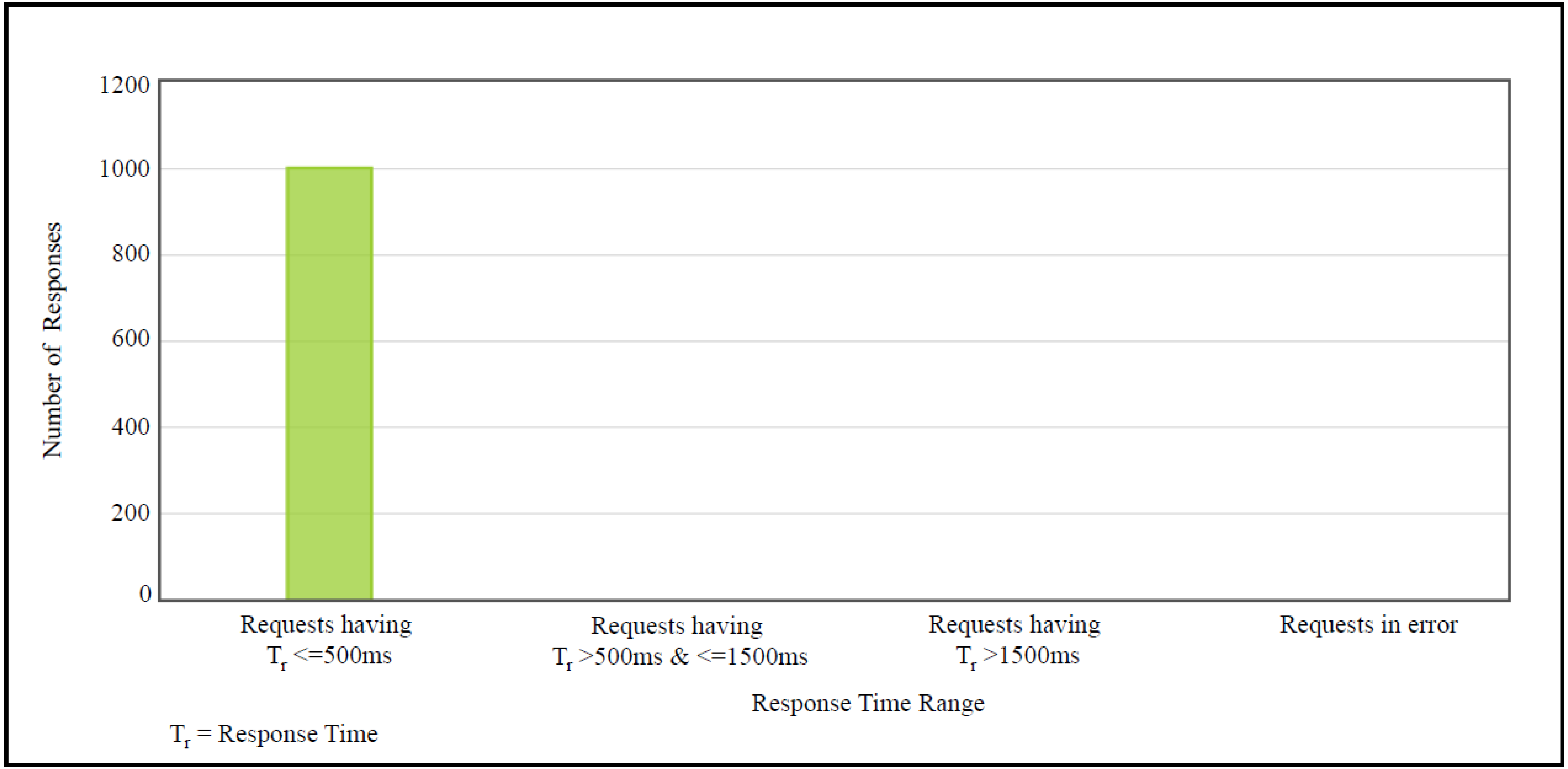
Experimental Results

Research	Algorithm	Server Authentication Time	Mutual Authentication Time
Barbareschi, et al.[3]	PUF based PHEMAP	NA	38.58ms
Hathal, et al.[8]	TA, TESLA	NA	8600ms
Zhang, et al. [25]	SRAM PUF and Blockchain	3302.9ms	991.8ms
Puthal, et al. [15]	Decision Tree(DT)	NA	0.6s to 0.803s
Aarella, et al.	XORArbiter PUF	0.5s -1.5s	500ms
Fortified-Edge	PUF based CA	<1500ms	500ms

Experimental Results – Server Response Times



Experimental Results – Mutual Authentication Time



Conclusion

- PUF-based authentication systems are proven to be a secure and lightweight scheme in IoT applications
- Mutual authentication of EDCs during load balancing takes less than 500 ms, hence reducing the latency
- The use of SRAM PUFs to generate certificates ensures the security of the authentication system
- The certificate-based authentication scheme discussed in this research removes the need for storing the CRP database at the client end, making it safe from external attackers accessing the database

Future Research

- For future research and development of the developed scheme, we propose extensive security analysis against external attacks like man-in-the-middle, spoofing attacks, machine learning attacks, and so on.

- Another objective is to design a PUF-based Security-by-Design (SbD) model for developing secure IoT applications for Smart Villages

Thank you!