International Journal of Computer Sciences and Engineering
Vol.**12**, Issue.**3**, pp.**11-18**, March **2024**
ISSN: 2347-2693 (Online)
Available online at: www.ijcseonline.org

ISSN: 2347-2693 (E)

Research Article

A Secure Framework based on Sensor Cloud Architecture for Efficient Street Monitoring in Smart Cities using Enhanced Elliptic Curve Encryption

Rajan Kumar Yadav[1*], Munish Saran[2], Upendra Nath Tripathi[3]

[1,2,3]Dept. of Computer Science, Deen Dayal Upadhyaya Gorakhpur University, Gorakhpur, Uttar Pradesh, India

**Corresponding Author: rkyd94@rediffmail.com*

Received: 29/Jan/2024; **Accepted**: 01/Mar/2024; **Published**: 31/Mar/2024. **DOI**: https://doi.org/10.26438/ijcse/v12i3.1118

Abstract: Smart City Surveillance systems play a key role in city development, helping to improve public safety and the efficiency of urban planning. But, there are some challenges regarding the security and integrity of the data transmitted from street cameras, such as privacy protection, legal compliance and protection from data breaches. Traditional methods such as access control lists offers coarse-grained data access and considering this as loophole our suggestion is Attribute-Based Encryption (ABE), which ensures fine-grained access control. We use Enhanced Elliptic Curve Cryptography (EECC) encryption on a sensor-cloud architecture, allowing data owners to define role-based access rules. This approach guarantees confidentiality and integrity and reduces computational overhead for faster encryption, decryption and key generation compared to traditional CP-ABE. This helps improve street camera security and is a step towards more resilient smart cities.

Keyword: Sensor cloud, Data Security, Enhanced Elliptic Curve Cryptography (EECC), Ciphertext Policy-Attribute Based Encryption (CP-ABE)

1. Introduction

Currently, the advent of the Internet of Things (IoT) has contributed to revolutionizing various industries, facilitating data collection, analysis and automation by connecting physical devices to the internet. A significant part of the Internet of Things ecosystem is sensor technology, which allows us to detect and measure only physical phenomena such as temperature, humidity, light, and motion. But, as there is an exponential increase in the number of sensors and data generated by these devices, there is a need for efficient data management and processing. Here comes the concept of Sensor-Cloud [1]

Sensor-Cloud combines sensor networks with cloud computing technologies to improve the efficiency and efficiency of data collection, storage, processing and analysis. It captures real-time information from sensors and uses the scaling, computing and mathematical resources of cloud computing platforms. The basic principle of Sensor-Cloud is to utilize the advantages of cloud computing to overcome the limitations of traditional sensor networks. In a traditional setup, sensors are connected through a local network and have limited storage and math constraints. As a result the data collected by these sensors is limited to spatial data most of the time and cannot be easily shared or obtained by other applications or developers. Sensor-Cloud whether external or manual, allows sensors to be connected to cloud infrastructure, from where they can transmit data widely, provide it remotely and perform deep analytics. By harnessing the power of the cloud, sensor-cloud provides a scalable and cost saving solution to managing large amounts of sensor data. This does not require spatial storage and mathematical resources, because the cloud infrastructure can cope with the mathematical and storage requirements of the sensor network. Sensor cloud is used in various sectors like Smart Agriculture, Industrial IoT, Environmental Monitoring, Smart Cities and Health, etc [2].

What are the benefits of using sensor-cloud ? First, it enables a common data management system, where data coming from different sensors can be gathered, processed and stored in one place. This facilitates real-time monitoring, analysis and decision making allowing businesses and organizations to gain valuable knowledge from the collected data. Secondly, sensor-cloud provides a platform for collaboration and data sharing between different stakeholders and besides this, Sensor-Cloud has expanded and improved its line. As the number of sensors and volume of data increases, cloud infrastructure can handle rapidly increasing requirements without significant demand. In addition to hardware improvements or infrastructure changes, sensor-cloud also enable various types of learning in a sequential manner. Additionally, sensor-cloud provides seamless access to sensor

data, allowing users to monitor and control sensor networks anywhere and at any time. But, along with its benefits, sensor-cloud also has some challenges and considerations. It is important to keep in mind issues like data security, privacy, network connectivity and interoperability so that the Sensor-Cloud solution can be successfully implemented and adopted [2].

2. Preliminaries

This section provides a detailed description of the architecture of Sensor-Cloud and Elliptic Curve Encryption, providing an overview of both.

Architecture of Sensor-Cloud – The architecture of a sensor cloud generally includes the following components :

Sensors – They are equipped with various sensors that capture real-world data such as temperature, humidity, speed or light. These sensors are responsible for collecting data from various sources like environment [3].

Sensor Nodes – Sensor nodes play a supporting role between sensors and cloud infrastructure. They receive data from sensors, process and package it and send it to the cloud for further processing and storage. Sensor nodes often include microcontrollers or microprocessor, communication modules and memory for data buffering [4].

Communication Network – Communication network connects network sensor nodes to cloud infrastructure. This type can be wired or wireless, depending on deployment and requirements. Wireless communication protocols such as Wi-Fi, Bluetooth or cellular networks are often used to transmit sensor data to the cloud [4].

Cloud Infrastructure – Cloud infrastructure includes servers, storage systems and networking components hosted in data centers. This is an example of the computational resources, storage capacity and scalability required to handle the large volumes of data generated by sensors. The type of cloud infrastructure depends on the specific requirements of the application, such as public, private or hybrid [5].

Cloud Services – Under cloud infrastructure, various cloud services are used to process and manage sensor data. These services can include data storage, data processing, analytics, machine learning algorithms and APIs for data access, allowing integration with other applications [5]. Cloud Computing is inherently suspect, which means cloud users have trust their Cloud Service Providers. Those providing cloud services have to take this doubt to the controlled star. In the real world research on cloud data integrity is important not only for privacy but also for obtaining unauthorized data such as crime. The transition from on-premises to cloud based servers should be impractical for users without impacting their experience [6]

Data Processing and Analytics - Sensor data can be processed by cloud services to yield valuable insights using filtering, aggregation, normalization and advanced analytics techniques. After that this processed data can be stored in databases or data warehouses for further analysis and visualization [7].

Applications and Services - Application and services are developed for specific purposes using data processing and analytics. These applications are used for a variety of tasks from real-time monitoring and control systems to predictive maintenance resource optimization or decision support systems. These applications can be delivered to end users through web interfaces, mobile apps or APIs [7].

Virtualization - Virtualization is a key and effective ingredient for sensors cloud architecture. This process improves virtualization and security by creating virtual instances of service, storage and networking components. Each virtual instance is given a separate environment which does not have any impact on other instances. With the help of virtualization resources are fully utilized and the work is made flexible which can be done dynamically and scale up and down. This also reduce the risk of high altitude insecurity. Improves joint management and deployment and facilitates virtual environment testing and development [8].

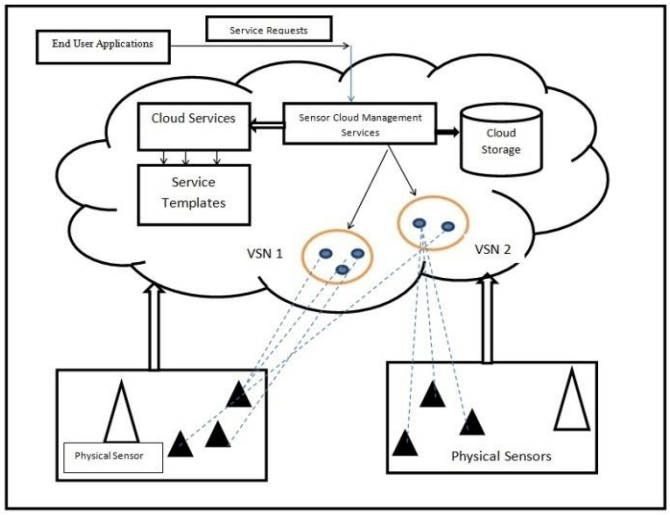

Figure 1. Architecture of "Sensor-Cloud"

Elliptic Curve Encryption

ECC or Elliptic curve cryptography is a new type of encryption techniques used in various cryptographic systems for secure communication and data protection. It is based on the mathematical properties of elliptic curves in finite fields. ECC provides strong security with relatively short key lengths compared to other encryption algorithms such as RSA. The strength of ECC lies in the computational difficulty of reversing the scalar multiplication process without knowledge of the private key. Due to the large size of the elliptic curve and the complexity of the mathematical operations determining the private key from the public key or cipher text become computationally infeasible. There are four phases of ECC [9].

Key Generation - In ECC the first step is key generation. In this a user selects a specific elliptic curve that is define over a

fine field and also selects a base point (or generator point) on that curve. Base point is a fixed point that lies on the curve and occurs with certain characteristics.

Using the base point the user generates his private key, which is random number and varies within a specified range. This private key remains confidential and should not be known to anyone. This private key is used for the generation of the public key which comes in the next step.

Public Key Calculation - After that the mathematical operation used is called scalar multiplication. In scalar multiplication one uses one's private key from the base point on the elliptic curve. This process results in a new point on the curve which is the public key of the user. This public key can be shared with other people and there is no need to keep it secret.

Encryption - So let us assume that user A wants to send an encrypted message to user B. For this user A first obtains user B public key which is used in the encryption process.

a. User A converts his message into numerical value.
b. Then user A generates a random value called an ephemeral key which is used only for this specific encryption.
c. User A produces a new point on the elliptic curve by doing scalar multiplication of the ephemeral key with user B public key.
d. Then user A calculates the x-coordinate of the point we generated and uses it as cipher text. This cipher text is sent to user B.

Decryption - Let us imagine that user B receives the cipher text and wants to decrypt it. For this these steps are taken in the decryption process.

a. User B generates a new point on the elliptic curve by performing scalar multiplication of its private key with the received cipher text (which is the x-coordinate)
b. Then, user B converts the x-coordinate of the generated point into its first numerical value.
c. User B converts this numerical value into the original message that user A originally sent.

3. Literature Review

According to Muhammad Awais Javed at el. [10] we will discuss the algorithms offered by elliptic Curve Cryptography (ECC) for intelligent transportation system (ITS) of smart cities. According to current ITS standards these algorithms provide digital signature, certification and encryption mechanism for ITS messages. We have used an experimental test-based to perform signature and encryption processes using ECC and evaluated their security packet overhead and latency performance. We have also developed an online tool that provides detailed security benchmark results based on our experimental studies. Additionally we have also implemented an ITS application scenario using NS-3 and SUMO simulations. Based on benchmarked security parameter values we analyzed their impact on QoS, security and security of ITS applications.

According to Mauricio A. Ramirez-Moreno at el. [11] in the coming times the population of cities is increasing and smart measures will have to be taken to meet their demand. Smart city initiatives are at the forefront in Europe and Asia. Data driven services using sensors are helping to improve city life. However sensor improvements big data analytics and citizen confidence are still challenges. In smart cities, Social security is important by digitalizing it and keeping in mind the practicality of the individual. Collaboration with government agencies is very important for smart city success. It is also important to understand the relationships between community member and different sectors. Smart communities are taking cities towards a modern, carbon neutral future.

According to Tanweer Alam [12] Cloud based technologies provide experts, business people and common people access to accurate information which helps them make better decisions and improves people's live. In smart cities people use their mobile devices which connect them to home, car and city facilities. Connecting devices and information to city systems reduces costs and improves management. Internet of Things reduces resource management garbage collection, accidents and pollution. In this article the author has introduced the importance of cloud based IoT in smart cities and has also highlighted the research possibilities for the future.

According to Moez Krichen at el. [13] our work is still in its beginning and its human aspect requires a lot of hard work whether it is theoretical or experimental. First of all we have to face the modeling problem. In this case we need to formalize our modeling and identify the specific elements of the IoT experiment. The model should not be bad so that the test population is not expanded. For this purpose and acknowledged star of human abstraction must be created. The second step is that we have to create our test and modify our algorithms keeping the security requirements in mind. It is also necessary to prove new algorithms. Thus humans have to upgrade their tools to implement new algorithms. We also want to test our approach with real examples. Finally we suggest adopting the same methodology presented which combines security and burden testing for IoT applications.

4. Proposed Framework for Security over Sensor Cloud based Street Monitoring:

Security has become a major aspect of society today. For this purpose people are installing smart security cameras in their homes and the government has also installed security cameras on the streets to monitor streets. The government can use public cameras and people's private cameras [12]. This makes street monitoring economical as the government does not need to install cameras near private houses when residents already have cameras installed as shown in the figure 2. There should be a security policy that ensures that only residents of the house can view their own private camera footage and not that of others. But the government should be allowed to view the footage of public and private cameras whenever necessary such as monitoring of entire streets, investigation of theft or crime etc.

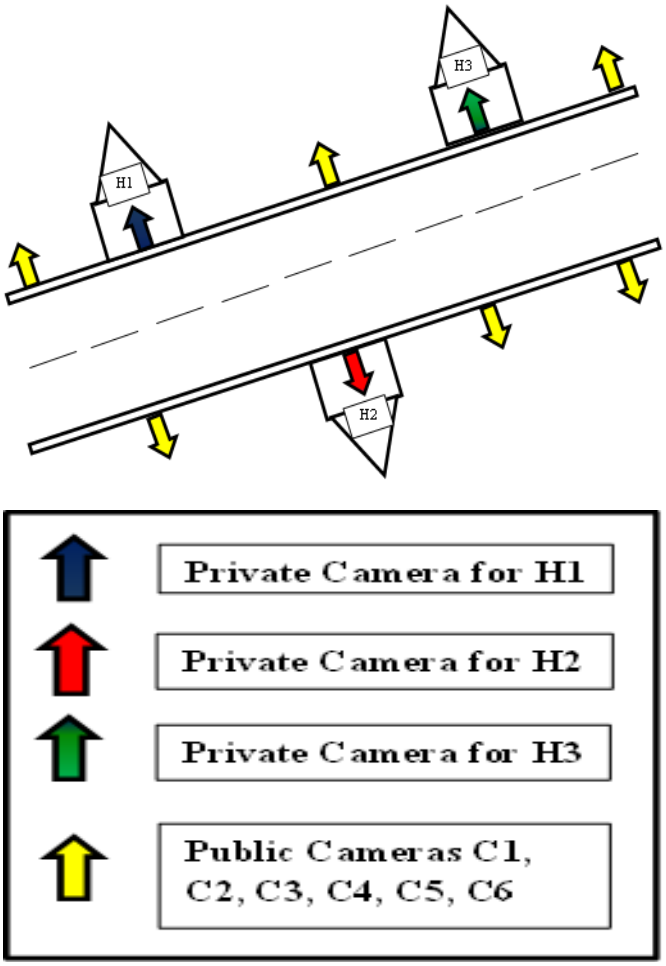

Figure 2. security cameras on the streets monitoring

Our security environment can be better in a virtualized WSN because it allows only authenticated users to see their data.

In the above example, let us assume that H1, H2 and H3 are individual private houses which have their own private cameras installed and C1, C2, C3, C4, C5 and C6 are the street cameras installed by the government on the streets as shown in figure 2.

The set of universal properties (U) can be considered as {H1, H2, C1, C2, C3} and the security authority that makes the rules for entry policy (Who can see what). That is only those devices that support their secret key entry policy can access that data. Here too in the example of virtualized WSN (private home owner or government's) secret key supports access policy only that data can be accessed.

In virtualized WSN access control policies are of paramount importance for security which regulated which entities or users can access what types of data and in what ways. This policy includes land, quality, time, place and owner. Through key access rules such as Role-based access control, time based access control, location based access control and ownership based access control. Only authorized users can access their data. It is expected to reach there. Thus this new security mechanism guarantees data confidentiality, data security and fine grained access control [12].

Table 1 . Sensors utilized for street monitoring

S. No.	Camera Sensors	Description
1	Motion Sensors	How to detect movements in the camera's field of view, by using a motion detection sensors.
2	Temperature and Humidity sensors	Monitor the temperature and wetness of the environment.
3	Vibration Sensor	Detects unusual vibrations that may indicate tampering
4	Wi-Fi/Cellular Modules	Enables remote monitoring and data transfer.

Table 2. Street Monitoring data consumers

S. No.	Camera consumers	Description
1	Traffic Management Systems	Analyze camera data for monitoring and optimizing traffic flow
2	Government Authorities	Use data for civic planning, security and city management.
3	Environment Monitoring Agencies	Monitor environment conditions captured by street cameras.
4	Transportation Authorities	Monitor traffic patterns, road conditions and incidents for transportation planning.

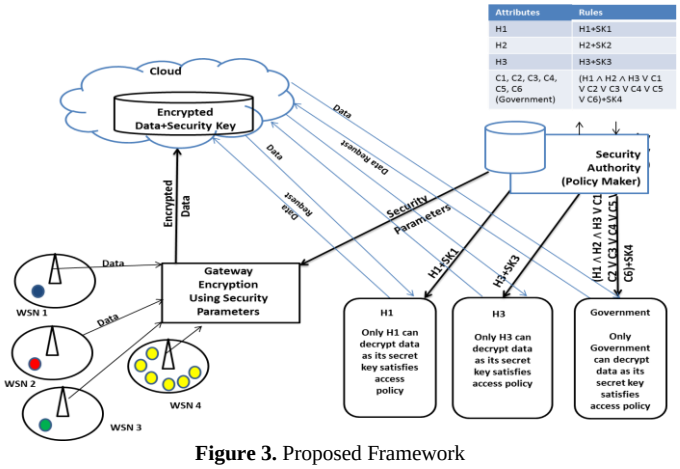

Figure 3. Proposed Framework

Table 3. Modules Involved in the proposed Street Monitoring system

Street Monitoring			
Actors for the system - Private homes cameras (H1, H2, …Hn). Government owned cameras (C1, C2…Cn)			
Modules Involved	**Description**	**Role/Responsibilities**	**Example**
Security Authority	The Security authority for the particular area.	1. Creation of attributes. 2. Generation of Keys (Encryption and Decryption Keys).	Attributes- Camera IDs of Public (government) cameras C1, C2….Cn. Camera IDs of Private (individual home residents) cameras H1, H2…Hn.
Data Owner	Private Home owners, Government.	1. Owns the data. 2. Creation of Security Policy.	1. (H1 V H2 V H3 V C1 V C2 V C3 V C4 V C5 V C6) 2. H1 V C1 V C2 V C3 V C4 V C5 V C6 3. H2 V C1 V C2 V C3 V C4 V C5 V C6
Data Consumer	End users	Need the complete camera footage from both public as well as private cameras for entire street monitoring in case of emergency.	Government monitoring agencies.

A robust framework has been proposed for a street monitoring system. First every authorized user is registered and his attributes are collected. Then an access policy is created that prevents unauthorized access the data. After data upload each data set is encrypted with an ECC encryption key. To control data access unique encryption key (EK) and decryption key (DK) are created for every user [14]. For real time monitoring system will continue to monitor surveillance data and if any suspicious activity or emergency is detected appropriate action will be taken such as generating alerts or informing law enforcement agencies. In this way the steps of user registration, access policy creation, data encryption, access control and real time monitoring can also be followed in the street monitoring framework. So that the street surveillance data can be managed in a secure and efficient manner. Figure 2 and figure 3 describe the architecture and workflow model of the proposed framework.

5. Proposed EECC Encryption Algorithm

Step 1 : Generate Key

a. Curve selection : a elliptic curve defined over a finite region is selected represented by E(Fp), Where p is a prime number.
b. Selection of base point : A base point G is selected on the curve.
c. Private Key Generation : A private key d is generated which is a random number between [1, n-1], whee n is the order of the base point G.
d. Calculation of the public key : The public key Q associated with the private key d is calculated: Q = d*G.
e. Identification of user attributes : User attributes are recognized and the HKDF (HMAC-based key derivation function) key authentication algorithm is used to identify them.
f. Key extraction : The extracted key (EK) is obtained using the HKDF key authentication algorithm.
g. creating the final key : The final key is created by combining the public key (Q) and the derived key (EK) obtained in step (d) and (f) key = Q +EK.

Step 2 – Encryption

a. Select a random key : A random number k among [1, n-1] is selected, which is suitable for the encryption process.
b. Use HKDF : by combining the public key 'key' and formal key k, the HKDF function is applied to generate a password that can be used to hide the message in a natural way.
c. Calculation of point P : By doing scalar multiplication of the unidimensional key k with the base point G the point P is obtained: P = k*G. This point represents the initial demise of our encryption.
d. Calculation of the shared secret point (S) : To obtain the shared secret point S, we perform scalar multiplication key k : S = k*key = k*(Q+EK). Here Q is the point obtained from the public key and EK is the key extracted from user attributes.
e. Take the x-dimension of the shared secret point (S) : The x-dimension of S is taken and it is represented as cipher text C to be hidden.
f. In this form, the plain text message M is converted to a numerical value and the point P is calculated by selecting a numerical key. Then the shared secret point S is obtained using HKDF whose x-dimension is represented as cipher text C.

ECC Decryption Algorithm

Step 1 : Generate Key

a. Private key generation : The private key d is generated within the same range used for the encryption process.
b. Calculation of the corresponding public key : The corresponding public key Q with the private key d is calculated : Q = d*G.

Step 2 : Decryption

a. Retrieve cipher text C : at the beginning of the decryption process we need to retrieve cipher text C.
b. Calculation of the shared secret point (S) : to convert the ciphertext C to the shared secret point S, we apply scalar multiplication to the private key d with the unidimensional point P (Which is the x-dimension of C) S = d*P = d*(k*G). Here P is the point which we obtained in the second step of encryption.
c. Transformation into original numerical value : The x-dimension of S is taken and it is converted into the original numerical value.
d. Convert the original numeric value back to the original plain text message : Finally we convert the original numerical value back to the original plain text message so that we can get original message.

6. Performance Evaluation

Implementation Setup

Using .Net framework version 5 and Visual studio 2019 we have implemented the proposed framework. The web application has been developed for data owners to register with the Street Monitoring Security Authority and upload street monitoring data as shown in figure 3. Technologies like C#, ASP.NET, ADO.NET and JQuery have been used in this project. Our application uses Microsoft Azure SQL Database for cloud storage which provide high performance, scalable and secure storage for structured data. This gives organizations the flexibility to manage their database while also taking advantage of Azure's robust infrastructure and their unique data management properties.

For home security we use the private home camera and government owned camera dataset which is taken from the "National Home Security Initiative" and this dataset combines information obtained from different cameras installed in private homes. This dataset includes household activity patterns such as entry and exits as well as information obtained from government security cameras. Each camera installed inside or outside a home collects visual and audio data of household activities. Apart from this data from government cameras installed at temples is also included in the dataset. This detailed dataset is helpful in studying home security activities and helps in improving security planning.

The datasets used for evaluating the performance of the proposed framework has multivariate data with 628 total number of frames divided into four vehicles one bike and 11 pedestrians. All the images were divided in four different files namely rouen_gt.sqlite (The annotation of all bike, pedestrians and cars), rouen_gt_cars.sqlite (The annotation of all cars only), rouen_gt_bike.sqlite (The annotation of all bike only) and rouen_gt_pedestrians.sqlite (The annotation of all pedestrians only).

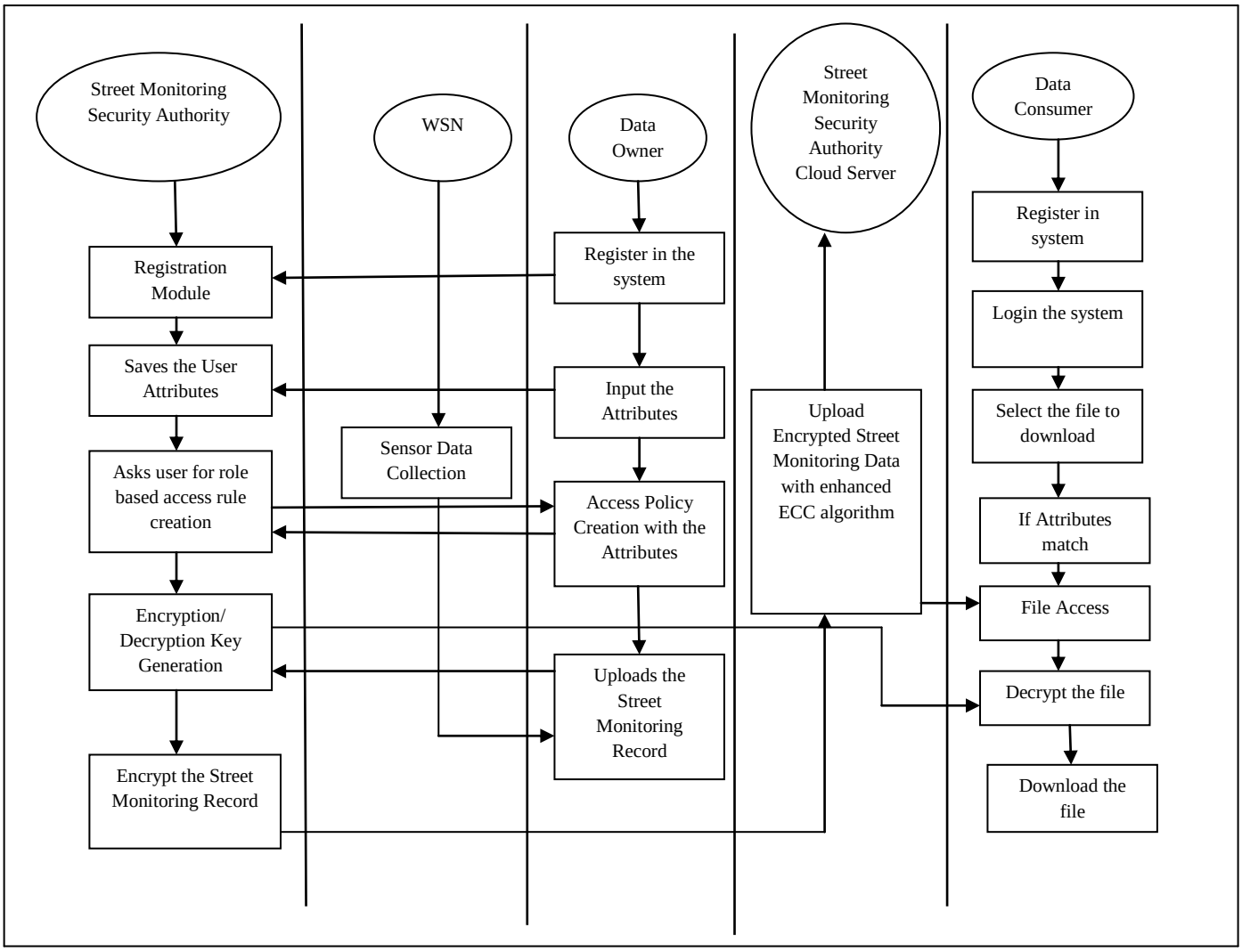

Figure 4. Workflow of the Proposed Framework

Encryption, Decryption, Key-Generation Time Analysis

Using files of different sizes 4.70 MB, 9.38 MB, 14 MB, 19.1MB, 24.5 MB and 30 MB which contain data from different camera sensors, the existing CP-ABE scheme is proposed by us. To evaluate the time for encryption, decryption and key generation compared to the scheme. It is observed that our proposed scheme produces better results than the existing scheme in every three device. These results are clearly presented in Figure 5, 6 and 7.

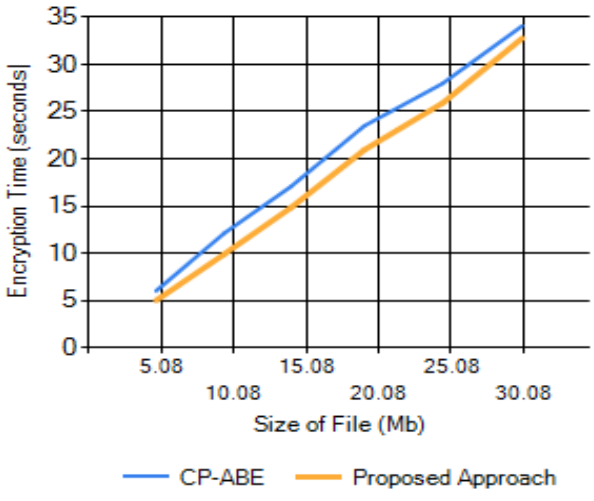

Figure 5. Encryption Time Analysis

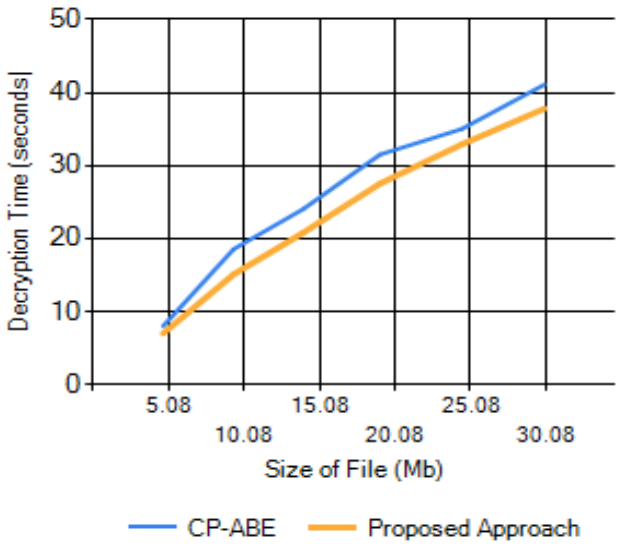

Figure 6. Decryption Time Analysis

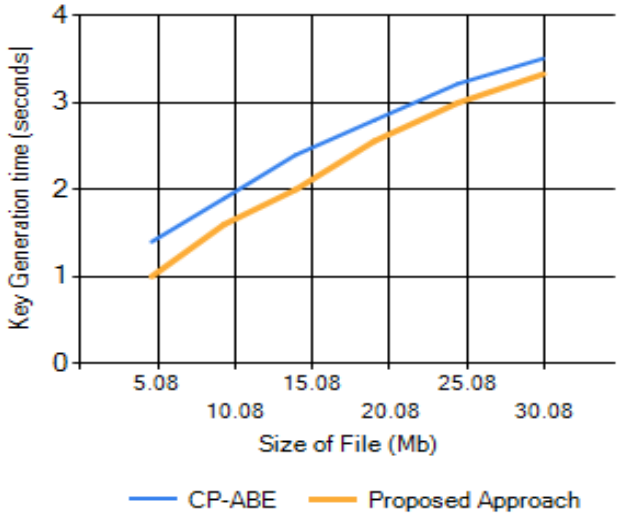

Figure 7. Key Generation Time Analysis

Table 4. Encryption, Decryption,Key-GenerationTime Analysis

Application	Size of File (MB)	Encryption Time (seconds)		Decryption Time (seconds)		Key Generation Time (seconds)	
		CP-ABE	Proposed Approach	CP-ABE	Proposed Approach	CP-ABE	Proposed Approach
Street Monitoring	4.7	6.01	5	8.01	7	1.4	1
	9.38	12.1	9.88	18.56	15.1	1.9	1.6
	14	17.05	14.81	24.05	20.81	2.4	2
	19.1	23.52	20.97	31.52	27.54	2.8	2.56
	24.5	28	25.92	35	32.9	3.22	3
	30	34.1	32.79	41.1	37.8	3.51	3.33

7. Conclusion

In today's time data security of street monitoring is very important in smart cities so that data privacy can be protected rules can be followed data security can prevent congestion so that continuity of care can be secured on the street and keep trust in the monitoring system. Prioritizing data security in private home cameras and government organizations can protect sensitive information, reduce mitigate risks and contribute to the achievement of high level street monitoring services. Individual private cameras and street cameras installed by the government have to face cyber attacks which include some common attacks like ransomware attack, data compression, phasing attack, insider threat, DoS/DDoS attacks, credential theft etc. To protect against these threats a security authority should be used which can provide an effective and transparent security mechanism. Our proposed approach uses an Enhanced ECC encryption algorithm based on sensor cloud architecture which provides an effective and transparent security system through which data owners can maintain complete control over their data using what is termed role based access control rules. The proposed framework develops security and trust and security monitoring in smart cities systems that are used to harness the benefits of sensor data. The results clearly show that the proposed approach achieves faster encryption, decryption and key generation compared to the legacy CP-ABE, even with more complex ECC keys.

Conflict of Interest

The Author's declare that there is no conflict of Interest to report.

Funding Source

The research was entirely Self funded by the Author's.

Author's Contributions

Rajan Kumar Yadav, as the main author of this research paper. Munish Saran and Upendra Nath Tripathi has provided necessary support to every phases on this research paper as co-authors.

Acknowledgements

This paper and the research behind it would not have been possible without the exceptional support of my supervisor Dr. Upendra Nath Tripathi. I would also like to acknowledge the invaluable contributions of my research colleagues and advisors who provided guidance and support throughout the research process.

References

[1] S. A. Chaudhry, K. Yahaya, F. Al-Turjman and Ming-Hour Yang, "A Secure and Reliable Device Access Control Scheme for IoT Based Sensor Cloud Systems", Special Section on Reliability in Sensor-Cloud Systems and Applications (SCSA), IEEE Access **2020**, Doi : 10.1109/ACCESS.2020.3012121.

[2] R.K. Dwivedi, M. Saran and R. Kumar, "A Survey on Security over Sensor- Cloud", 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), **2019.**

[3] K. Hasseb, A. Almogren, I. Ud Din, N. Islam and A. Altameem, " SASC : Secure and Authentication-Based Sensor Cloud Architecture for Intelligent Internet of Things", Sensors **2020**, 2468, Doi : 10.3390/s20092468, MDPI.

[4] R. Alturki, H. J. Alyamani, M. A. Ikram, MD Arafatur Rahman, M. D. Alshehri, F. Khan and M. Haleem, "Sensor-Cloud Architecture : A Taxonomy of Security Issues in Cloud-assisted Sensor Networks", DOI : 10.1109/Access.2021.3088225, IEEE Access.

[5] R. K. Yadav, M. Saran and U. N. Tripathi, "A Comprehensive Review of data Security in Cloud Computing Environment Using Cryptographic Algorithms", International Journal for Research Trends and Innovation, Vol.7, Issue.11, ISSN : 2456-3315.
[6] R.K. Yadav, M. Saran, P. Maurya, S. Devi and U.N. Tripathi, "Hybrid DES-RSA Model for the Security of Data over Cloud Storage", Internation Journal of Computer Sciences and Engineering, October , Vol.**11**, Issue.**10**, pp.**1-7, 2023.**
[7] S. K. Dash, J. P. Sahoo, S. Mohapatra and S. P. Pati, " Sensor-Cloud Assimilation of Wireless Sensor Network and the Cloud", CCSIT 2012, Part I, LNICDT 84, pp.**455-464, 2012.**
[8] S. Bose, A. Gupta and S. Adhikary, "Towards a Sensor-Cloud Infrastructure with Sensor Virtualization", Proceedings of the senond Workshop on Mobile sensing, Computing and Communication, pp.**25-30, 2015**, doi : 10.1145/2757743.2757748.
[9] Y. Yan, "The Overview of Elliptic Curve Cryptography (ECC)", Journal of Physics : Conference series, Volume 2386, the International Conference on Computing Innovation and Applied Physics (CONF-CIAP 2022), 20 August **2022.**
[10] M. A. Javed, E. B. Hamida and W. Znaidi, " Security in Intelligent Transport Systems for Smart Cities : From Theory to Practice", Sensors **2016**, 16, 879; doi : 10.3390/s16060879.
[11] M. A. Ramirez- Moreno, S. Keshtkar, Diego A. Padilla-Reyes, Edrick Ramos-Lpez, Moises Garcia-Martinez, Monica C. Hernandez-Luna, Antonio E. Mogro, Jurgen Mahlknecht, Jose Ignacio Huertas, Rodrigo E. Peimbert-Garcia, Ricardo A. Ramirez-Mendoza, Agostino M. Mangini, Michele Roccotelli, Blas L. Perez-Henriquez, Subhas C. Muhkopadhyay and Jorge de Jesus Lozoya-Santos, "Sensors for Sustainable Smart Cities : A Review", Appl. Sci. **2021**, 11, 8198. https://doi.org/10.3390/app11178198.
[12] T. Alam, " Cloud – Based IoT Applications and Their Roles in Smart Cities", Smart Cities, 4, pp.**1196-1219, 2021.** https://doi.org/10.3390/smartcities4030064.
[13] M. Krichen, M. Lahami, O. Cheikhrouhou, R. Alroobaea and A. J. Maalej, "Security testing of Internet of Things for Smart City Applications : A Formal Approach", Smart Infrastructure and Applications, EAI/Springer Innovations in Communication and Computing, https://doi.org/10.1007/978-3-030-13705-2_26.
[14] S. Alshehri, S. P. Radziszowski and R. K. Raj, "Secure Access for Healthcare Data in the Cloud Using Ciphertext-policy Attribute-Based Encryption", **2012** IEEE 28th International Conference on Data Engineering Workshops, doi : 10.1109/ICDEW.2012.86.

AUTHORS PROFILE

Rajan Kumar Yadav Earned his Bachelor of Science (B.Sc.) in Computer Science from Deen Dayal Upadhyaya Gorakhpur University, Gorakhpur (Uttar Pradesh, India) and Master of Computer Application (MCA) from Madan Mohan Malaviya University of Technology (MMMUT, Gorakhpur, Uttar Pradesh, India) He is currently Ph.D. Research Scholar in the Department of Computer Science, Deen Dayal Upadhyaya Gorakhpur University, Gorakhpur, Uttar Pradesh, India. His Research Interest includes Cloud Computing, Machine Learning, Deep Learning and IoT.

Munish Saran – Earned his Bachelor of Technology (B.Tech) in Computer Science Engineering (CSE) from Babu Banarasi Das National Institute of Technology & Management and Master of Technology (M.Tech) in Computer Science Engineering (CSE) from Madan Mohan Malaviya University of Technology. He is Currently Ph.D. research scholar in the Department of Computer Science Deen Dayal Upadhyaya Gorakhpur University, Gorakhpur, Uttar Pradesh, India. His research interest includes Cloud Computing, IoT, Machine Learning and Deep Learning. He was previously working in Infosys as senior system Engineer for 4 years.

Dr. Upendra Nath Tripathi - Currently Associate Professor in the Department of Computer Science, Deen Dayal Upadhyaya Gorakhpur University, Gorakhpur, Uttar Pradesh, India. He has 22 Yeays of teaching and research experience. His area of interest are Database, IoT, Machine Learning, Deep Learning, Cloud Computing and Data Science.

