ICEAIST-25
M.A.M. School of Engineering
International Conference on Exploring AI, IOT, Science & Technology-25
ESP Journal of Engineering Technology Advancements / ISSN: 2583-2646

Original Article

Attribute-Based Encryption Reliable Outsourcing Decryption

Prof.G.Kanimozhi [1],Dineshwar V [2],Kaviya P [3],Suriya Prakash R [4], keerthana K [5]

[1] *Professor,Department Of Computer Science & Engineering,M.A.M School Of Engineering,Tamilnadu,India,*

[2,3,4,5] *Ug Scholar,M.A.M School Of Engineering, Tiruchirappalli,Tamilnadu,India*

Abstract: *Attribute-Based Encryption (ABE) is a more complex way of encrypting data that lets you restrict who can see it by encrypting it with a set of descriptive properties. Only users whose attributes match those in the access policy that is hidden in the ciphertext can decode the data. This functionality makes ABE a great choice for transferring data securely in cloud computing settings. But one big problem with ABE is that the decryption process takes a lot of computing power, and this gets worse as the quantity and complexity of the attributes involved increases. This means that typical ABE doesn't work well on devices with limited resources, such smartphones, IoT devices, and embedded systems.Researchers have suggested outsourcing the decryption process to cloud services that can't be trusted in order to solve this problem. In this architecture, a user makes a transformation key from their private key and gives it to the server, which does most of the work of decrypting the data. The server sends back a ciphertext that has been partially decrypted. The user can rapidly finish the job with little effort. This method greatly lessens the amount of work that needs to be done on the client side while yet keeping data private.This paper talks about the theory and practice of outsourcing ABE decryption in a way that is dependable. We look into both key-policy and ciphertext-policy ABE schemes, how to make safe transformation keys, and how to include verifiability methods to make sure the outsourced computation is correct. We also talk about more advanced security features like attribute privacy, dual-policy encryption, and zero-knowledge proofs for checking transformations. We talk further about trade-offs in performance and look at how well different schemes function in real-world situations including exchanging healthcare data, controlling access to enterprise systems, and secure communication in decentralized networks. Our research shows that outsourcing decryption to a trusted third party not only speeds things up, but it also makes ABE possible in the real world.*

Keywords: *Attribute-Based Encryption, Outsourced Decryption, Ciphertext-Policy ABE, Verifiable Computation, Lightweight Devices, Cloud Security, Access Control*

I. INTRODUCTION

Cloud computing, mobile technology, and the Internet of Things (IoT) are spreading so quickly that they have changed how data is created, stored, and shared in all kinds of businesses. Cloud systems today routinely store and process huge amounts of sensitive data, from personal communications and health information to workplace collaboration tools and government databases. Cloud computing is the best choice for current IT infrastructures because it is scalable, flexible, and cost-effective. But using third-party cloud providers to store and process data poses serious issues about privacy, security, and control.

One of the most important things to think about in this situation is how to make sure that only authorized users can get to certain bits of data. This is especially true when standard perimeter-based security models don't work in decentralized or multi-user cloud systems. Standard encryption solutions are helpful, but they usually only allow for coarse-grained access control—either full access or none at all. They also depend on safe key distribution and user identity management, which can be hard to scale.

Attribute-Based Encryption (ABE) is a great option. By linking encrypted material to certain sets of properties, this type of public-key encryption lets you regulate who may access it very precisely. An ABE system encrypts data based on an access policy that is set over attributes like role, department, and clearance level. Only people with attribute sets that meet the policy can decode the information. There are two primary types of ABE: Key-Policy ABE (KP-ABE), which puts the policy in the user's secret key, and Ciphertext-Policy ABE (CP-ABE), which puts the policy on the ciphertext. CP-ABE is better for data owner-controlled access, which is why it is so popular for cloud apps.

Even though ABE has some good points, one big problem is that decrypting it takes a lot of computing power. Decryption in ABE schemes usually requires a lot of pairing and exponentiation over elliptic curve groups, which are operations that take a lot of time and get more complicated as the access policy gets bigger and more complicated. This is a big challenge for devices that don't have a lot of resources, such mobile phones, smart sensors, and embedded control units, because they typically don't have enough processing power or battery life to do these kinds of tasks well. As a result, this computational overhead makes it harder to use ABE in real-world settings.

This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/2.0/)

Researchers have suggested getting around this problem by sending the decryption job to outside servers, which are often in the cloud and do most of the work for the user. Outsourced ABE decryption is what this method is called. In these kinds of systems, the user makes a transformation key from their private decryption key and sends it to the cloud server. The server uses this transformation key to make a partially decrypted ciphertext. The user then employs a simple operation to finish decrypting it. This method lets low-power devices take part in secure data sharing networks without sacrificing performance or the ability to restrict who can access the data.

But sending decryption work to servers that you don't trust or just partially trust creates other problems. The most important things are making sure that the server does the decryption transformation correctly (verifiability), keeping the plaintext and attributes private (security and privacy), and stopping unauthorized users or cooperating parties from getting access (collusion resistance). The system must also be able to grow, be light for the client, and be able to handle new risks like adaptive chosen-ciphertext assaults and the leaking of access patterns.

This paper gives a full look at how dependable outsourced decryption works in ABE systems. We look at the cryptographic basis of ABE, look at existing schemes that make outsourcing safe and verifiable, and assess their computing efficiency, security assurances, and applicability for use in modern cloud systems. We also talk about new improvements like dual-policy ABE, attribute-hiding strategies, and verifiable computation techniques that make outsourced decryption even more useful and reliable. We want to give a complete picture of the sector and help create scalable, secure, and efficient attribute-based access control systems for cloud-assisted apps.

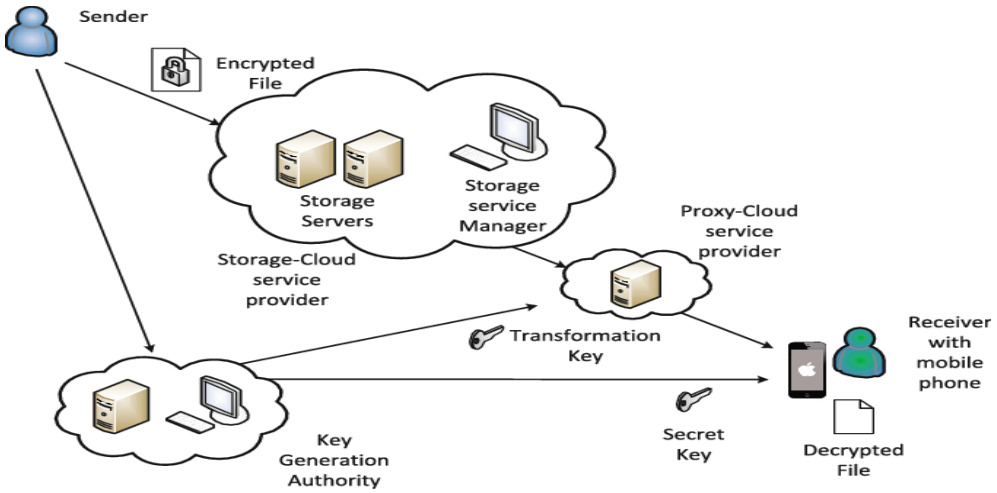

Figure 1. Attribute-Based Encryption Reliable Outsourcing Decryption

II. FUNDAMENTALS OF ATTRIBUTE-BASED ENCRYPTION

Attribute-Based Encryption (ABE) is a complex cryptographic method that builds on classic public-key encryption by allowing access restriction based on descriptive attributes instead of individual user IDs. ABE was created to fix the problems with identity-based and role-based encryption systems, which aren't very flexible or scalable when there are a lot of users with changing access permissions. ABE lets logical policies based on user or data qualities control encryption and decryption. This makes it possible to have fine-grained, scalable, and decentralized access control that works in cloud-based and distributed contexts.

There are two primary types of ABE systems: Key-Policy ABE (KP-ABE) and Ciphertext-Policy ABE (CP-ABE). In KP-ABE, the ciphertext is linked to a group of descriptive attributes, such "role:doctor," "location:clinic," and "department:cardiology." The user's private key has an access policy, like "role:doctor AND department:cardiology." A user can only decrypt a ciphertext if the qualities that go with it meet the access policy that is included into their key. People usually utilize this paradigm when the key issuer is in charge of access control.

CP-ABE, on the other hand, flips this structure around: the access policy is stored in the ciphertext, and the user's private key holds their attribute set. If the user's attributes meet the policy that is encoded in the ciphertext, they can be decrypted. CP-ABE gives data owners more autonomy because they may decide who can access the data when it is encrypted without having to rely on a central authority to implement access controls. This makes CP-ABE a great way to share data safely in open and changing systems like cloud storage and health information exchange.

Complex cryptographic operations, especially bilinear pairings over elliptic curves, are very important to the security and functionality of ABE schemes. These operations make it possible to specify access policies in a flexible way. These procedures are very powerful, but they take a lot of computing resources and can slow down the decryption process a lot.

The more complex the access policy and the more attributes there are, the more pairing operations are needed during decryption. This is a problem for devices that can't do a lot of calculations.

Because of this, ABE has great security and flexibility for access control, but its performance problems—especially in CP-ABE systems—mean that it needs more optimizations or extra methods like outsourced decryption. The goal of these improvements is to make ABE useful for real-world applications that need both security and efficiency, such as mobile computing, the Internet of Things (IoT), and distributed cloud platforms.

III. THE DECRYPTION BOTTLENECK AND ITS IMPLICATIONS

Attribute-Based Encryption (ABE) is a strong way to regulate access to information in a very detailed way, but it is hard to use in real life because it takes a lot of computing power to decrypt. This problem is even worse in Ciphertext-Policy ABE (CP-ABE) systems, because decrypting requires a lot of pairing and exponentiation operations. These tasks take a lot of computing power and get worse as the number of attributes in the access policy increases. The more complicated the access structures are, such when you use conjunctive or threshold gates, the more it costs to compute them. This generally leads to long delays and high energy use.

This problem is especially important in places where devices don't have a lot of resources, such smartphones, IoT sensors, smart cards, and embedded systems. These gadgets usually don't have a lot of processing power, memory, or battery life. When they have to decrypt CP-ABE ciphertexts, they can get overloaded rapidly, which can cause them to take longer to respond, run out of energy soon, or even fail to decrypt at all. In applications that need to work in real time or are crucial to safety, such emergency healthcare systems, military communications, or financial trading platforms, these kinds of delays are unacceptable and could have serious effects, like loss of life, security breaches, or financial losses.

Also, ABE systems can't scale or be flexible very well because they can't easily decrypt data on mobile or edge devices. If decryption can only happen on powerful servers or workstations, ABE is far less useful in decentralized and ubiquitous computing environments. This fact has made it hard for ABE to be used in real-world, cloud-assisted environments where users want quick, easy, and safe access to private information

Researchers have suggested outsourcing the decryption process to external computers, usually cloud-based ones, that have the processing power to tackle complicated cryptographic tasks. In this technique, a user gives the server a transformation key that is based on their private decryption key. This lets the server do most of the work of decrypting. After that, the server sends back a ciphertext that has been partially decrypted. The user just needs to do a small amount of work to get the original message.

This outsourcing model, on the other hand, raises important security and privacy issues, such making sure that the server can't figure out the plaintext or the user's attribute set and that it does the transformation correctly. Now, a big part of ABE research is figuring out how to make outsourcing protocols that keep things private, correct, and verifiable without putting too much on the client.

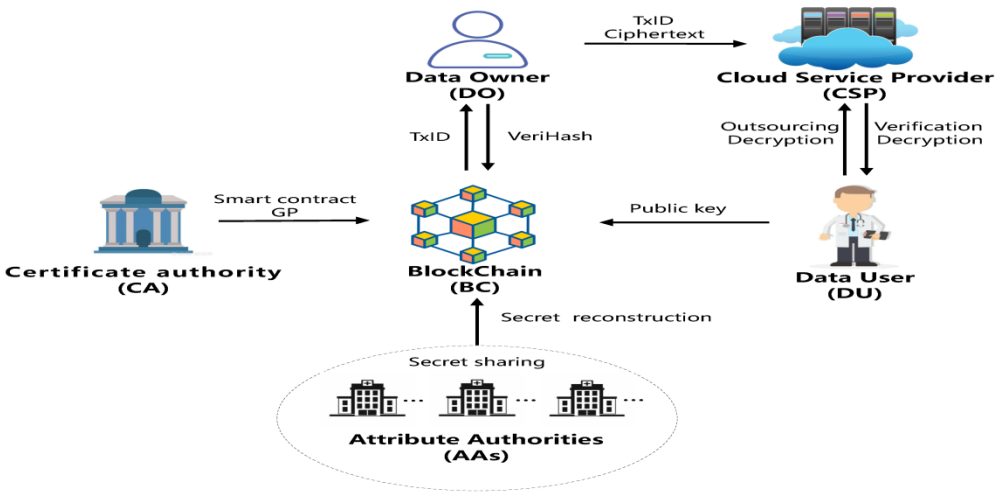

Figure 2.The Decryption Bottleneck and its Implications

IV. OUTSOURCING DECRYPTION IN ABE: AN OVERVIEW

Outsourcing decryption in Attribute-Based Encryption (ABE) is a big step forward that solves the problem of how expensive it is to decode, especially in Ciphertext-Policy ABE (CP-ABE) systems. The main notion behind this approach is

that a user can give an outside computer, usually a cloud server, the hard aspects of decryption without putting the user's private information or the encrypted data at risk. This paradigm, which is sometimes called "transformation-based outsourcing," makes it possible to use cryptographic access control in a way that is efficient and can be scaled up for devices with limited resources.

In transformation-based outsourcing, the user makes a transformation key from their private decryption key first. This transformation key is sent to the server and is made just for the server to change a ciphertext into a partially decrypted form in one direction. The transformation step sends most of the heavy pairing and exponentiation calculations to the server. The converted ciphertext keeps its security features, and the user can only fully decode it through a simple final decryption step, which usually involves a simple hashing or exponentiation operation.

One of the most important things about this method is that the server can't learn the plaintext, figure out the user's attribute set, or figure out how to get the original private key back. The transformation key must be made so that it doesn't give the server any information that it can use to do anything other than what is needed for transformation. This makes sure that the data and the user's identity stay private even if the server is hacked or acts badly.

, outsourcing decryption comes with significant concerns when it comes to trust and accuracy. A dishonest or broken server could give you the wrong results, which could mean that decryption fails or data corruption goes unnoticed. To lessen this, trustworthy outsourcing plans include ways to check that the work is being done. These tools let users check the accuracy of the modified ciphertext on their own, without having to decrypt the whole message. Cryptographic proofs, tags, or checksum-like validation values that are added throughout the transformation process can make something verifiable.

Also, strong outsourcing plans need to protect attribute privacy and make it hard for people to work together. Keeping the user's attribute set from being seen helps keep them anonymous and stops profiling. Collusion resistance makes sure that several users can't combine their attributes to decipher a ciphertext that they aren't allowed to view on their own. All of these properties make outsourcing decryption a strong and useful way to set up ABE systems in modern cloud environments that are fast, safe, and able to grow

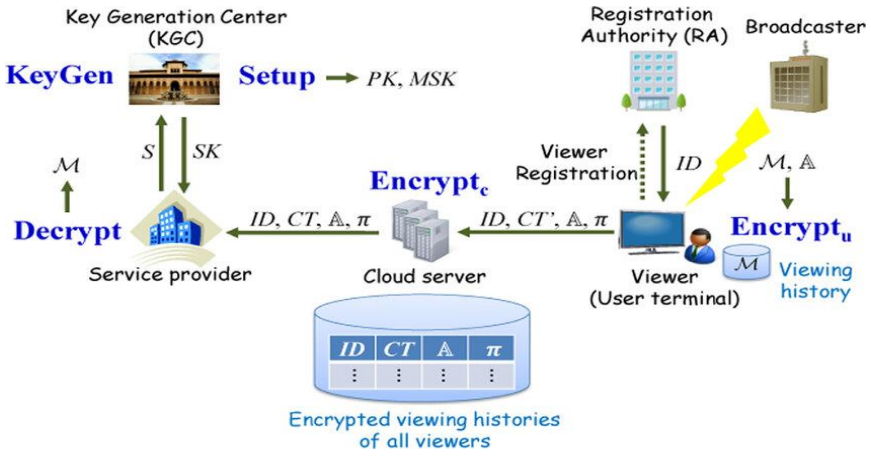

Figure 3. Outsourcing Decryption In Abe: an Overview

V. RELIABLE AND VERIFIABLE OUTSOURCED DECRYPTION

For outsourced ABE decryption to work and be safe, it must have two important features: it must be correct and it must be verifiable. Correctness means that the transformation done by the external server will make a ciphertext that, when the legitimate user processes it, gives them the correct and original plaintext message, as long as the user has a valid attribute set. On the other side, verifiability lets the user check for themselves if the server has correctly carried out the transformation process, even if they can't see the entire plaintext.

It is quite important to be able to verify things when the outsourcing server is not fully trusted. A bad server could give an erroneous or changed transformation without verification procedures, which could lead to invalid or corrupted decryption results that the user might not even notice. To fight this, secure outsourced decryption systems use cryptographic proofs, integrity markers, or validation tests that are built into the ciphertext or transformation result. These let users check the transformation's accuracy and integrity before moving on to the final decryption.

Chase et al. were some of the first to write important papers in this field. They came up with a CP-ABE paradigm that allowed for verifiable outsourced decryption. Their plan included a proof part to the changed ciphertext that the user could

easily check. Green et al. later improved this idea by combining cryptographic methods including zero-knowledge proofs, bilinear aggregate signatures, and safe function evaluation to lower the amount of computing power needed while keeping security.

These verification methods not only make users more likely to trust the system, but they are also necessary for auditability, policy compliance, and legal responsibility in areas of data that are sensitive, like healthcare, finance, and government services. These techniques make it possible to use ABE-based access control in real-world cloud systems while making sure that both efficiency and integrity are maintained by including correctness assurances in the outsourcing protocol.

VI. SECURITY MODELS AND ASSUMPTIONS

Security models and cryptographic assumptions that are very strict support the design and implementation of secure and reliable outsourced decryption in Attribute-Based Encryption (ABE) systems. These models are very important for explicitly proving that the system can withstand different kinds of attacks and attackers, especially when computations are done on servers that are not fully trusted or are only partially trusted.

The Selective Security model is the most frequent security model used in ABE. In this approach, an attacker must first say which attributes they want to attack before the system's public parameters are made public. This model is thought to be weaker than the Adaptive Security model, which lets the enemy choose its target qualities based on what it learns during the attack. However, it is still extensively employed in real-world systems since it is relatively simple and easy to analyze. The selective model makes proofs clearer and is the basis for making more complicated security systems.

From a cryptographic point of view, the security of ABE methods, even those that are outsourced, is usually predicated on hard math problems in bilinear groups. The Bilinear Diffie-Hellman (BDH) problem and the Decisional Bilinear Diffie-Hellman Exponent (DBDHE) problem are two important assumptions. These assumptions are the basis of the encryption process. They make sure that even if attackers have certain information, such the public key or transformation key, they can't easily decipher the ciphertext or figure out important information.

When it comes to outsourcing, the threat model usually includes a semi-trusted server that is supposed to follow protocol operations but could also try to eavesdrop, change calculations, or figure out information about the plaintext or the user's attributes. So, under this approach, secure outsourcing plans must be properly thought out to protect the privacy of data, policies, and transformations.

The transformation key given to the server is made using blinding techniques or random masking to protect against these dangers. This makes sure that the server can't figure out the user's full decryption key or attribute set. Some systems also use zero-knowledge proofs or secure function evaluation (SFE) to keep data from leaking during the translation process. More advanced methods provide public verifiability, which means that anyone—not just the data owner—can check the validity of the transformation without having access to the original plaintext. This is quite useful in places where auditing and following the rules are very important.

In the end, these cryptographic models and assumptions are what make people trust outsourced ABE systems. They make sure that security assurances don't get weaker when efficiency goes up.

VII. LIGHTWEIGHT CLIENT-SIDE DECRYPTION

One of the best things about outsourced decryption in Attribute-Based Encryption (ABE) systems is that it makes the client side do a lot less work. In traditional ABE implementations, especially Ciphertext-Policy ABE (CP-ABE), decrypting is hard on computers since it requires things like numerous bilinear pairings, exponentiations, and polynomial evaluations. As the number of attributes in the access policy increases, these procedures become more complicated, making classical decryption problematic in contexts with limited resources.

In outsourced decryption systems that use transformations, most of the work is done by a cloud server that is only somewhat trustworthy. The server uses a transformation key that comes from the user's private key to change the ciphertext into a new ciphertext that keeps the information secret. The client then does a small operation, like a single exponentiation, modular inversion, or hash computation, to get the original plaintext back. This makes decryption easier, which not only speeds things up but also gives devices with limited battery life and processing power more time to work.

This characteristic is very crucial in smart settings, where devices need to be able to respond on their own and with as little delay as possible. For instance, in e-health systems, a wearable sensor could need to securely get and understand encrypted medical instructions in real time. It would be impossible to decrypt directly because of the limited capabilities of the device. When you outsource decryption, though, the sensor can send complicated cryptographic tasks to the cloud and make sure that only people with the right permissions can decrypt the final data.

In the same way, machines can join encrypted command-and-control networks in industrial IoT without having to do a lot of extra work. The end result is a system that can grow and is safe, with strong access control and support for the real-time needs of lightweight clients. So, lightweight client-side decryption is a key step forward for using ABE in real-world, cloud-based settings.

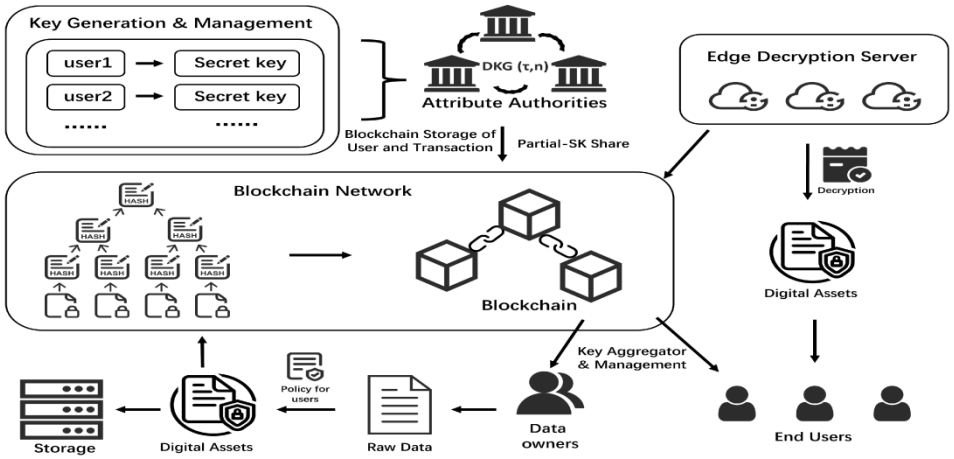

Figure 4.Lightweight Client-Side Decryption: Architecture & Features

VIII. ATTRIBUTE HIDING AND PRIVACY PRESERVATION

In ABE schemes, attributes often hold private information about the user, such as their name, role, or location. When making ciphertext or giving out keys, showing these attributes could put users' privacy at risk. Attribute concealing is an important improvement in outsourced decryption methods that keeps enemies from finding out private information by looking at ciphertext. Some plans do this by either encrypting the properties themselves or hiding the rules for getting to them.

A truly secure outsourced ABE system must make sure that neither the server nor any outside party can figure out the access structure or attribute list from the ciphertext or modified ciphertext. To protect this privacy, methods such as anonymous attribute tokens, dual-policy encryption, and random masking have been used. Also, policy privacy keeps the structure of the access policy that is contained in the ciphertext private, which protects against inference attacks that employ known user roles or data patterns.

IX.VERIFIABLE COMPUTATION IN CLOUD-ENABLED ABE

Verifiable computation is an important idea in safe cloud computing that lets clients check that the results returned by a remote or semi-trusted server are true. Verifiable computation is very important for keeping confidence, consistency, and integrity when delegating decryption chores to an outside party in the context of Attribute-Based Encryption (ABE), especially in outsourced decryption systems.

When you use transformation-based ABE outsourcing, the server takes the ciphertext and applies a transformation key to it. Then, it sends a partially decrypted version back to the client. But there is no built-in guarantee that the transformation was done correctly because the server could be malicious or broken. To fix this, newer ABE systems use verifiable calculation methods that let the client check the server's output without having to see the plaintext.

Zero-knowledge Succinct Non-interactive Arguments of Knowledge (zk-SNARKs), Boneh–Lynn–Shacham (BLS) signatures, and homomorphic authenticators are some of the methods used in ABE frameworks to construct decryption pipelines that can be verified. For instance, cryptographic tags or checksums may be added to ciphertexts while they are being encrypted. When the client gets the changed ciphertext, they check these tags against the expected values to make sure that the transformation was done honestly. The result is thrown away if the verification fails. This keeps the data safe and stops it from being used in the wrong way.

This characteristic is especially important in high-stakes settings like financial systems, e-healthcare, and legal record management, where data authenticity and auditability are necessary for following the rules. Verifiable computation not only makes users more trusting, but it also gives cloud-based ABE services a clear and accountable foundation. These technologies provide a safe and scalable way to share data in untrusted cloud environments by combining ABE's fine-grained access control with strong verifiability.

X. CASE STUDIES AND APPLICATIONS

Attribute-based encryption with reliable outsourced decryption is gaining traction in numerous real-world scenarios. In healthcare systems, sensitive patient data can be encrypted under attributes like disease category, physician credentials, or access clearance. Medical professionals with matching attributes can delegate the decryption burden to a cloud server while ensuring data confidentiality. Similarly, in military communication systems, encrypted tactical data can be accessed by authorized personnel on-the-go, using lightweight decryption enabled by secure outsourcing.

In cloud file-sharing platforms, such as those offered by Dropbox or Google Cloud, ABE with outsourced decryption provides fine-grained access without overburdening the end-user device. Enterprises can securely share financial reports, strategic plans, or legal documents under access policies specifying departments, roles, or project involvement. Only authorized employees can access the information after validating the outsourced transformation. Such mechanisms also play a vital role in decentralized storage systems like IPFS, where data is widely replicated but needs controlled decryption access.

XI. CHALLENGES AND FUTURE DIRECTIONS

Even though a lot of progress has been made, there are still a lot of problems to solve before outsourced ABE decryption techniques can be completely safe and effective. Some of the most important problems are:

- Allowing dynamic attribute revocation without having to re-encrypt
- Making transformation keys smaller for networks with limited resources
- Allowing several ABE authorities to work together across domains
- Post-quantum cryptography primitives make systems more resistant to quantum assaults.
- Creating outsourced ABE protocols that can be verified by anyone

In the future, studies may look into how to use machine learning to predict access trends and make transformation procedures better on the fly. Some people are also interested in using ABE with blockchain for decentralized access auditing and safe key distribution. Another potential field is privacy-preserving federated learning, which uses ABE-secured data to train models on many clients with outsourced decryption for faster local processing.

XII. CONCLUSION

Attribute-Based Encryption (ABE) is one of the most important new developments in cryptographic access control. It lets you keep your data private while still letting you set very specific access rules. ABE is different from typical public-key cryptosystems because it lets you set access permissions based on descriptive features. This makes it possible to share data in decentralized contexts in a way that is scalable, expressive, and dynamic. This makes it great for modern cloud infrastructures, the Internet of Things (IoT), and other types of distributed computing where users need varied levels of access to data according on their jobs, credentials, or other factors.

Even though it has some benefits, the fact that ABE is so hard to compute, especially during the decryption process, makes it very hard to use in real life. Decryption procedures frequently include a lot of pairing and polynomial evaluations, which get bigger and more complicated as access policies get bigger and more complicated. These kinds of requirements are hard on devices with low resources, such smartphones, embedded systems, and edge computing units, which are already widespread in cloud-based ecosystems. Because of this, researchers and developers that want to use ABE in real life are focusing on fixing the performance problem with decryption.

The idea of outsourcing ABE decryption in a reliable way has come up as a good way to move expensive calculations to other servers without putting data security or privacy at risk. With transformation-based outsourcing, users can send the job of decrypting to servers that they don't fully trust or just partially trust by using specially made transformation keys. These keys let servers change encrypted data into a form that the client can readily finish with little local resources. One important thing about strong outsourcing schemes is that they include verifiability mechanisms. This means that users may cryptographically check to see whether the transformation was done correctly, which keeps data safe and secure even when there are untrusted third parties involved.

This research study gave a full look at the ABE framework, with a focus on the problems and solutions that come up when you outsource the decryption process. We talked about important types of ABE, like CP-ABE and KP-ABE, and looked at what using them might mean for computers. The article also looked at improvements to the architecture that make it possible to decrypt data on the client side without using a lot of resources, as well as the use of verifiable computation methods like zk-SNARKs and BLS signatures, and other security models that protect policy confidentiality and attribute concealing.

In fields like healthcare, where devices need to make decisions based on encrypted patient data, finance, where privacy and compliance with regulations are important, and industrial automation, where IoT devices work in environments with limited bandwidth and power, the need for secure and efficient outsourced decryption is very clear. In all of these

situations, ABE with trustworthy outsourced decryption keeps data safe, useable, and verifiable even when it is stored on infrastructure that isn't trusted.

As the need for safe and cooperative data sharing grows, future research should focus on making verifiability better, adding support for dynamic attributes, and combining ABE with new technologies like blockchain and federated learning. In conclusion, dependable outsourced ABE decryption is a key part of creating cloud-enabled systems that are scalable, privacy-aware, and trustworthy, and that allow for safe information flow across the digital ecosystem.

XIII.REFERENCES

[1] Sahai and B. Waters (2005). Encryption based on fuzzy identity. EUROCRYPT 2005.

[2] Goyal, V., Pandey, O., Sahai, A., and Waters, B. (2006). Attribute-based encryption lets you regulate who can access encrypted data in a very specific way. ACM CCS.

[3] Bethencourt, J., Sahai, A., and Waters, B. (2007). Ciphertext-policy encryption based on attributes. IEEE S&P.

[4] Chase, M. (2007). Encryption based on multiple authorities. TCC.

[5] Green, M., Hohenberger, S., and Waters, B. (2011). Giving someone else the job of decrypting ABE ciphertexts. USENIX Security.

[6] Yu, S., Wang, C., Ren, K., and Lou, W. (2010). Getting safe, scalable, and fine-grained access control to data in the cloud. IEEE INFOCOM.

[7] Franklin, M., and Boneh, D. (2001). The Weil pairing lets you encrypt depending on identity. CRYPTO.

[8] Xing, X., Liu, X., Cao, Z., and Liang, K. (2011). Encryption based on attributes with outsourced decryption that can be verified. ESORICS.

[9] Liang, K., Liu, X., and Zhang, J. (2012). Sharing data in the cloud quickly and safely with access control. IEEE Transactions on Cloud Computing.

[10] Rouselakis, Y. and Waters, B. (2013). New proof methods and useful constructs for large universe attribute-based encryption. ACM CCS.

[11] Jiang, J., Yu, S., Ren, K., Lou, W., and Hou, W. (2013). DAC-MACS: a way to manage who can access data in cloud storage systems with more than one authority. IEEE INFOCOM.

[12] Hur, J. and Noh, D. K. (2011). Attribute-based access control with easy revocation in systems that outsource data. IEEE Transactions on Systems That Are Parallel and Distributed

A. Lewko, T. Okamoto, A. Sahai, K. Takashima, and B. Waters (2010). Attribute-based encryption and (hierarchical) inner product encryption are two types of fully secure functional encryption. EUROCRYPT.

[13] Nishide, T., Yoneyama, K., and Ohta, K. (2008). Attribute-based encryption with access mechanisms that are only partially masked by the encryptor. ACISP.

[14] Li, M., Yu, S., Zheng, Y., Ren, K., and Lou, W. (2013). Using attribute-based encryption, cloud computing makes it possible to share personal health records in a way that is both safe and scalable. IEEE Transactions on Parallel and Distributed Systems.

[15] Lai, J., Deng, R. H., and Li, Y. (2012). CP-ABE that is expressive and has access structures that are only partially veiled. ESORICS.

[16] Zhang, Y., Chen, X., Li, J., Xiang, Y., Hassan, M. M., and Alelaiwi, A. (2016). A dual-policy ABE system for sharing data in the cloud. Computer Systems for the Next Generation

[17] Han, Q., Susilo, W., Mu, Y., and Zhou, J. (2015). A way to manage access to and search for keywords in encrypted cloud data that keeps your privacy safe. The Journal of Network and Computer Applications.

A. Sahai and B. Waters (2014). The theory and practice of attribute-based encryption. IACR Cryptology ePrint Archive.

[18] M. Chase and S. S. Chow (2009). Making multi-authority attribute-based encryption safer and more private. ACM CCS.

[19] Wang, S., Zhou, Q., Yu, S., Lou, W., and Hou, W. (2016). PD-ABE: A way to encrypt data that can be done in parallel and across multiple servers for scalable access control in the cloud. IEEE Transactions on Systems That Are Parallel and Distributed.

[20] Zhang, K., Liang, X., Lu, R., and Shen, X. (2013). Synergetic ways to protect data privacy in the cloud. IEEE Transactions on Parallel and Distributed Systems.

[21] Xu, J. and Wang, W. (2013). Cloud computing allows for flexible and detailed access management. Computer systems for the next generation.

[22] Liang, X., Cao, Z., Lin, H., and Shao, J. (2009). Attribute-based proxy re-encryption that can delegate tasks. ASIACCS.

[23] Wang, C., Wang, Q., Ren, K., and Lou, W. (2012). Public auditing that protects privacy for data storage security in cloud computing. IEEE INFOCOM.

[24] Canard, S., and Devigne, J. (2015). Delegation of massive polynomials and matrix computations that can be checked by anyone, with examples. IACR.

[25] Gennaro, R., Gentry, C., and Parno, B. (2010). Non-interactive verifiable computing means hiring people you don't trust to do your work. CRYPTO.

[26] Parno, J. Howell, C. Gentry, and M. Raykova (2013). Pinocchio: Almost a practical way to check computations. IEEE S&P.

[27] Benabbas, S., Gennaro, R., and Vahlis, Y. (2011). Delegation of computation over big datasets that can be verified. CRYPTO.

[28] Boneh, B. Lynn, and H. Shacham (2004). Short signatures from the Weil pairing. The Cryptology Journal.

[29] Yu, S., Wang, C., Ren, K., and Lou, W. (2010). Sharing data based on attributes and taking them away. ASIACCS.

[30] Wang, Q., Wang, C., Ren, K., Lou, W., and Li, J. (2011). Allowing public verification and data changes to keep cloud computing data safe. ESORICS.

Sahai and H. Seyalioglu (2012). Attribute-based encryption with small ciphertexts that is completely safe. TCC.

[31] Freeman (2010). Changing pairing-based cryptosystems from groups of composite order to groups of prime order. EUROCRYPT.
[32] Zheng, H. Zhang, and J. Zhou (2017). An effective CP-ABE technique that allows for hiding access policies and checking them. The Journal of Network and Computer Applications
[33] Yu, J., Ren, K., Lou, W., and Li, Y. (2012). Getting cloud computing to have safe, scalable, and fine-grained data access control. IEEE INFOCOM.
[34] Wang, S., Zhou, Q., Yu, S., Lou, W., and Hou, W. (2016). An effective CP-ABE technique for transferring cloud data with policy updates. IEEE TDSC.
[35] Delerablée, C. (2007). Identity-based broadcast encryption that keeps the size of the ciphertexts and private keys the same. ASIACRYPT.
[36] Emura, K., Miyaji, A., Nomura, A., Omote, K., and Soshi, M. (2009). A ciphertext-policy attribute-based encryption approach that keeps the size of the ciphertext the same. ASIACCS.
[37] Zhang, Y., Chen, X., and Li, J. (2015). A hybrid ABE technique for mobile cloud computing that lets you verify outsourced decryption. IEEE Transactions on Computing Services.
[38] Lewko, A. and Waters, B. (2011). Making attribute-based encryption less centralized. EUROCRYPT.
[39] Han, W. and Huang, X. (2013). A verified outsourced CP-ABE decryption system with access control that is very detailed. Network Security: An International Journal.
[40] Jiang, Y. and Zhang, M. (2015). Secure and verifiable outsourced decryption of attribute-based encryption in cloud computing. Networks for security and communication.
[41] Liu, Z., Huang, X., and Zhang, J. (2016). Access control for cloud-based e-healthcare systems that keeps your information private and can be verified. IEEE Transactions on Cloud Computing
[42] Liu, X., Cao, Z., and Au, M. H. (2013). Attribute-based encryption that works well, has ciphertext of a fixed size, and decrypts quickly. ESORICS.
[43] Deng, M. and Li, J. (2015). For safe cloud storage, attribute-hiding and verifiable outsourced decryption ABE. Science of Information.
[44] Yang, K., Jia, X., Ren, K., Zhang, B., and Xie, R. (2013). DAC: In cloud computing, deduplication with access control. IEEE Transactions on Distributed and Parallel Systems.
[45] Guo, F., Chi, Y., and Liu, J. K. (2017). Attribute-based signature with easy revocation. TDSC.
[46] Yu, S., Ren, K., Wang, C., and Lou, W. (2009). Using attribute-based encryption to control who can access cloud storage data in small amounts. ACM CCS.
[47] Zhang, J. and Mao, J. (2013). An outsourced ABE decryption technique that is flexible and can be checked, with attribute concealing. Networks for security and communication.