International Current Journal of Engineering and Science (ICJES)
Volume 4, Issue 8, pp 25-32, August-2025
ISSN (Online): 2583-7257
Article DOI: https://doi.org/10.47001/ICJES/2025.408005

Protocols for Computing that Ensure Privacy and Safeguard the Inputs of All Involved Participants

[1]Priyanka A. Patil, [2]Mohan Kulkarni

[1,2]M. E. Electronics, Department of Electronics Engg; KBP college of Engg., Satara Dist-SATARA; Maharashtra, India

***Abstract* - This study delves into the examination of cryptographic protocols that permit two or more parties to compute functions based on their shared inputs without disclosing those inputs in their original state. This is a vital aspect of Secure Multi-Party Computation (SMPC). The security attained by safeguarding data during computations enables participants to carry out calculations while ensuring data protection. SMPC is utilized in fields such as finance, healthcare, and data analytics, where sensitive information is at stake. The investigation centers on strategies for embedding security into the examined protocols to confirm their accuracy while maintaining user privacy. This is accomplished through the application of methodologies such as homomorphic encryption, secret sharing, and zero-knowledge proofs. We analyze different security configurations, including semi-honest and malicious security, to evaluate the vulnerability of these protocols to possible attacks or data exposure. Moreover, critical topics such as scalability and computational complexity are tackled, proposing strategies to lessen communication costs and processing time in the realm of Big Data applications. The results indicate that it is achievable to implement a secure and practical SMPC with strong security guarantees, regardless of the performance requirements of various real-world application contexts. This work is significant to the current state of cryptography and suggests new protocols that facilitate sensitive computations for real-world applications while ensuring privacy in the modern digital landscape.**

Keywords: Multi-Party Computation, Homomorphic Encryption, Privacy, Cryptography, MPC, Cryptographic protocols, Secure Protocols.

I. INTRODUCTION

In today's technologically advanced world, there is a need for information security in processing and analysis. Secure Multi-Party Computation (SMPC) has emerged as a revolutionary approach to solving privacy problems in collaborative data analysis. This promising development enables two or more parties to jointly perform computations on each other's data without disclosing it. It is relevant in fields as diverse as finance and medicine, helping companies gain competitive insights from their data while preserving individual privacy. This comprehensive tutorial delves deeper into this topic, introducing the concept of SMPC, its fundamentals, and its evolution. It examines the main elements of SMPC systems and analyzes how they can be implemented in different sectors. It also compares SMPC with other privacy-preserving approaches, such as homomorphic encryption and blockchain technology. Furthermore, it covers the legal and ethical aspects related to SMPC implementation. By the end of the lectures, readers will have a deep understanding of what SMPC is and how it will evolve in the near future, as it shapes the future of data privacy and collaborative computing.[1]

Fundamentals of Secure Multi-Party Computation

Secure Multi-Party Computation (SMPC) is an innovative cryptographic method that allows different parties to jointly process data without compromising the information it contains. This approach has addressed the challenge of computing mathematical functions based on information that cannot be shared but must be combined. SMPC is a set of cryptographic protocols used to solve the problem of maintaining the confidentiality and correctness of computations performed concurrently by multiple parties.

The basic concept of SMPC is to allow parties to perform computations on sensitive data while concealing the input data that each party submits for computation from the other participating parties. Private computation refers to the ability to compute a given function jointly with multiple parties without disclosing their data, using cryptographic methods such as secret sharing, encryption, and zero-knowledge proofs. This opens up enormous opportunities, as all organizations can analyze data before making decisions without necessarily compromising their security (Figure 1).[2-5]

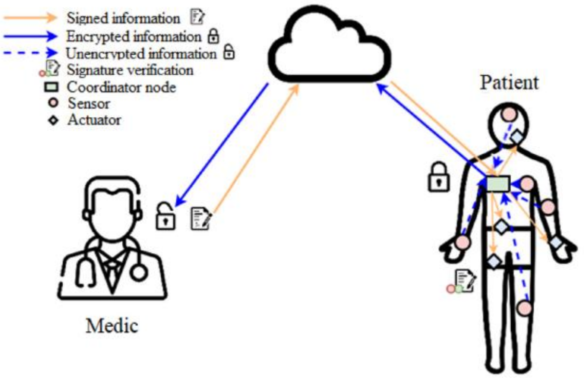

Figure 1: Secure Multi-Party Computation

One of the design goals of SMPC is to protect the identity of an individual input during computation. This is possible through strategies such as secret sharing and encryption. Secret sharing techniques divide the inputs so that neither party can reconstruct the original data with the help of the others. Furthermore, data confidentiality is protected through encryption to protect the data during computations and exchanges between parties.

By using SMPC, the generated inputs and intermediate results do not fall into the hands of third parties but are encrypted to ensure that the loss of confidential information is prevented.[6] 1.1 Mathematical Foundations

SMPC's computational theory is based on cryptography, the technique used to enable secure computing. One of these is additive secret sharing, in which a value to be protected is divided into several randomly selected parts, called "secret parts." For example, a value of $100,000 can be divided into three secret parts: These variations are incomplete but can range from $20,000, $30,000, to $50,000. One or more secret parts of a given DSM contain no information about the original value, but together they reveal it.

This concept of secret sharing is fundamentally used to ensure privacy even during use. In the SMPC protocol, participants locally add their values and aggregate the result to generate the final output. This process enables secure computation, where the provided information cannot be revealed by the person who provided it.

Security Models

As mentioned above, attackers pose a threat to SMPC protocols, which are designed to ensure the correctness and integrity of computations, due to the involvement of multiple parties who jointly submit false information to the system. By leveraging techniques such as zero-knowledge proofs and secure commitment schemes, SMPC is able to detect various types of attacks, such as result manipulation, fraud, and even attempts to extract excessive information from other people's contributions.

The security requirements of SMPC protocols are strict and can be classified based on the behavior of potential adversaries. There are two main types of security models:

- Semi-honest (passive) security: This model assumes that corrupt parties want to gather information but do not violate an established protocol. It provides a slightly lower level of security; however, this helps prevent the accidental disclosure of information between cooperating parties and lays the foundation for more secure models.
- Malicious (active) security: In this model, the attacker is permitted to act in a manner that is inconsistent with the protocol's execution intent. With this model, very high security can be achieved with protocols that preserve the confidentiality of all honest parties and the accuracy of the result, even when the presence of other malicious parties is assumed.

II. DEVELOPMENT OF SMPC PROTOCOLS

The development of secure SMPC (Multi-Party Computation) protocols has been an exciting process spanning several decades and has seen many milestones and successes. This development has enabled SMPC to move from a theoretical model to a real method for addressing confidentiality issues in collaborative data processing.[7] Initial Approaches

The original ideas behind SMPC emerged with the advent of Mental Poker in the late 1970s, a cryptographic concept designed to mimic remote play without the interference of a trusted third party. This early work laid the foundation for specific protocols better suited to certain functions. The problem, however, is that it was Andrew Yao who first developed ideas for secure two-party computation in the early 1980s and solved the so-called Millionaire's Problem. Yao's contributions include the Scrambled Circuit Protocol, which is still used in the most successful SMPC applications today (Table 1).

Later, other authors such as Oded Goldreich, Silvio Micali, and Avi Wigderson extended Yao's work to multi-party computation. They proposed the GMW paradigm, a transformation of the specification of a scheme for combining multi-party computation protocols secure against passive/semi-honest adversaries into protocols secure against active/malicious adversaries. Although this approach was initially considered expensive due to its high overhead, it paved the way for further improvements to SMPC.

Modern Advances

Recent research on SMPC has focused on optimizing the efficiency of proposed protocols, as proposed since late 2007. These changes have led to more efficient protocols and made SMPC a computationally viable solution for real-world problems such as distributed voting, auctions, and private auctions, the sharing of signature or decryption functions, and the recovery of private information. The Danish sugar beet auction is the practical and successful model of SMPC, which involved a double electronic auction in January 2008. This case demonstrated that SMPC can work under real-world conditions and enable a linear exchange of secrets with minimal communication between stakeholders.

Improvements to the security model have also been made to SMPC protocols. Originally, it was assumed that the best security model was a semi-honest (passive) security model, in which corrupted parties cooperate to obtain more information by consulting the protocol description. However, recent advances in this area address the case of malicious (active) security. In this model, adversaries may mistakenly choose not to execute the protocol and attempt to deceive, creating a need for increased security.

Cutting-Edge Techniques

Significant progress has been made in recent years with SMPC techniques to solve problems previously considered intractable or very complex. Among the interesting achievements is the ability to perform secure inference with large language models (LLMs). As a result, researchers demonstrated that it is possible to perform inference operations on an encrypted version of a 13 billion-parameter model in just a few seconds per token using a combination of MPC servers and GPUs. Modern SMPC techniques have also made progress in specific applications, such as convolutional neural networks, cloud deployment, and structure-aware private set intersection. These advances have opened up new application areas for SMPC, making it suitable for various types of computations while maintaining data confidentiality.

In addition, recent advances have included tabular operations for mass use, regular expressions for long text data, and extensions to traditional machine learning algorithms. These solutions help protect training data, queries, and models, depending on the application area and efficiency expectations. However, as SMPC evolves, new problems arise for researchers to solve, such as handling new data types such as image data, free text, and DNA sequences, as well as faster processing. These advances are due to the introduction of new cryptographic protocols, as well as parallelism and hardware acceleration techniques, making SMPC an ideal solution to the data protection problem in the age of big data and collaborative computing.[8]

Key components of the SMPC system: SMPC systems consist of several components that together provide the possibility for private, multi-party computational operations. These components ensure that multiple parties can freely compute a function, while keeping the input information private. Let's examine the essential elements of SMPC systems:

Ticket Exchange

Therefore, the first step in SMPC is to define the function for which the parties wish to perform the joint computation. Each participant has specific details that cannot be disclosed during the computation. To achieve this, SMPC uses secure techniques to exchange secrets and securely distribute inputs between parties. Sharing Shamir's secret is also one of the critical aspects of this approach: The method uses polynomial interpolation to divide the secret so that only a predetermined number of pieces can recover the original data.

For example, through additive secret sharing, a value of $100,000 can be divided into three randomly generated pieces (or "secret pieces"): $20,000, $30,000, and $50,000 must be collected. Each piece individually has no significance compared to the original data, but together they allow the reconstruction of the secret value. Secure Computing

When it comes to data security and encryption, the actual distributed system handles the computation. In this step, the parties work together to calculate the value of the desired function without disclosing any information. During the secure computation phase, various cryptographic methods are applied to ensure confidentiality and protection throughout the process.

For example, if three colleagues wanted to add up their joint hourly wages without disclosing each other's, each participant would create local sums of their secret shares. They would then exchange partial results with other parties so that the final result could be calculated jointly, but no one would know the information entered.

Security models within SMPC protocols can be implemented using two security models: semi-honest (passive) and malicious (active). When corrupt parties collude to obtain information while simultaneously rejecting external information, the semi-honest model is attacked. The adversarial model, on the other hand, assumes that an adversary could change their behavior or decide as they see fit, provided they cheat.

III. OUTPUT RECONSTRUCTION

The final aspect of SMPC systems is the process of computational power reconstruction. At this point, the parties do not receive aggregated values, but rather receive an answer from a calculation performed by all parties without any additional information about the other parties' input values (Figure 2).

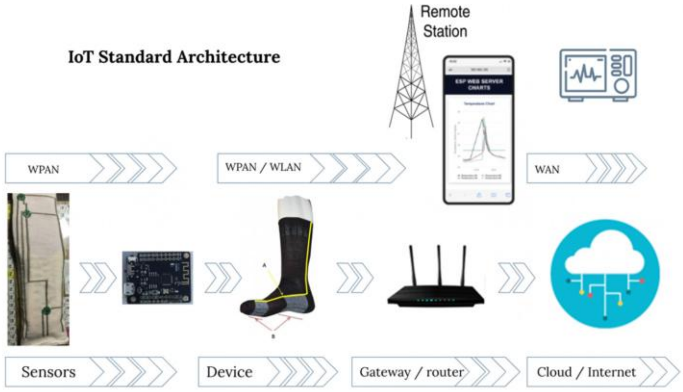

Figure 2: Output of IoT standard architecture

To continue with the salary example: The company's average hourly wage would be disclosed, but the salary of each participant would not. This process ensures accuracy while ensuring that each party's contributions do not reveal the other party's contributions. To securely reconstruct the output, SMPC systems primarily use techniques such as majority filtering. In some cases, a complete binary tree is created from quorums, each consisting of multiple parts, with the root quorum collecting the output of a circuit. This output is processed and passed down the chain using majority filtering, with each party calculating the majority of the messages received from the main quorum to obtain the correct output. These key components collectively contribute to secure multi-party computation, enabling multiple parties to collaborate in computing the results of analyzing sensitive datasets without compromising individual privacy. Thus, SMPC systems offer a promising tool for privacy protection in various domains, from finance and healthcare to machine learning as a service [9].

IV. SMPC IN PRACTICE

The concept of secure multi-party computation is no longer an abstraction, but a real idea used in various disciplines. This section focuses on the current adoption, applications, case studies, and performance analyses of SMPC systems in practice.

Real-World Implementations

Currently, SMPC integration has expanded into various fields, playing an important role in financial transactions, medical research, and digital assets. In the last two years of the 2010s, digital asset custodians began to rely on SMPC to protect their assets. This application has a profound impact on the blockchain space, and the private key of a Web3 wallet can be split (distributed among multiple parties). A minimum number of participant keys is required to execute a function, which increases the security of the environment against further malicious intrusion. An interesting example of this method can be found in the context of cryptocurrency wallets. MPC-backed Web3 wallets are used by custodians to secure assets and confirm transactions. While traditional multi-signature wallets use multiple private keys to sign a transaction, MPC wallets split the single key into multiple subkeys, each of which is transmitted to the custodian. This approach guarantees both security and the flexibility required for signing transactions.

Practical Cases

It is also important to highlight several examples that demonstrate how SMPC is used in practice to solve specific problems. One example is the Jana system, developed by Galois Inc. in collaboration with several universities and companies. Jana provides a secure MPC database for running a Private Data as a Service (PDaaS) application for relational data. This system is unique among encrypted databases in that it encrypts all data, even during processing. This contrasts with current encrypted databases, which at least make the data available for processing.

Another example is Sharemind, a secure MPC database system developed by Estonia-based Cybernetica. Sharemind aims to address these challenges and, in doing so, solve problems related to data sharing and computation by offering its clients the ability to collaborate on data analysis while maintaining their confidentiality. Partisia, founded in Denmark in 2009, was the first company to use SMPC for purely commercial purposes. They were initially used in auctions; it was also the first large-scale use of the SMPC at the now-famous sugar beet auction. Today, Partisia once again functions as an SMPC-focused commercial marketplace, covering everything from research and development to design marketing.

V. PERFORMANCE ANALYSIS

Recent advances in SMPC protocols have increased their potential effectiveness to the point where they can be considered practically useful. For example, it has been shown that inference operations on an encrypted model with 13 billion parameters can be performed on GPU-based MPC servers in just a few seconds per token (Table 2).

Higher performance levels have also been observed recently in some applications. Researchers implementing GWAS with SMPC found that the overall execution time was reduced under similar network loads. Lower p-values and fewer degrees of freedom justify the expectation of greater efficiency: a study of one million people using an SMPC-based GWAS could be completed in approximately three weeks, whereas previous studies suggested the process would take three months. In a drug-target interaction (DTI) prediction scenario using SMPC, the researchers observed a nearly two- to three-times faster code reduction compared to a four-times faster execution time, and a reduction in network utilization by more than half. These improvements are significant for realistic training sessions, as training time could be reduced to less than one day compared to the current four days.

However, results still depend on the specific application and the increased complexity of certain algorithms. For example, the SMPC implementation took 18.5 hours for classification in the metagenomic binning task, compared to less than 10 seconds for a normal run. This performance difference is primarily due to the overhead of computational aspects specific to the MPC configuration, such as Bloom filters. However, there are still many opportunities for developing SMPC applications in practice and in new areas of secure computing. Therefore, with the accumulation of new knowledge and the optimization of implementations, SMPC has the potential to gradually become a key element of privacy-preserving analyses and collaborations.[10-11]

Privacy-Preserving Data Analysis with SMPC

Secure Multi-Party Computation (SMPC) is a promising technique that has recently received much attention because it allows multiple parties to perform computations on sensitive data without third parties receiving information about it. This has proven very useful in various fields such as statistics, machine learning, and big data management.

Statistical Computations

In particular, SMPC enables statistical computations to be performed on multiple distributed datasets while maintaining the confidentiality of the original datasets. A well-known misconception occurs when two millionaires try to determine who is richer than the other without disclosing their wealth. This concept also applies to more complex cases, such as the sum, mean, median, or any other form of average value of different bids from two or more parties (Figure 3).

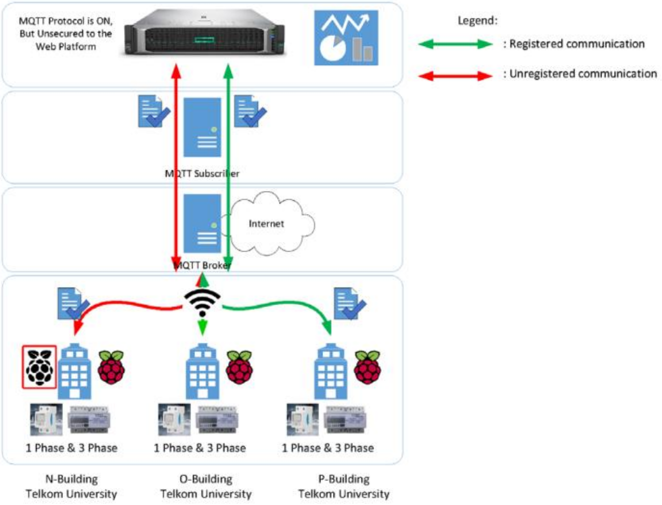
MQTT Protocol is ON, But Unsecured to the Web Platform
Legend:
: Registered communication
: Unregistered communication
MQTT Subscriber
Internet
MQTT Broker
1 Phase & 3 Phase
1 Phase & 3 Phase
1 Phase & 3 Phase
N-Building Telkom University
O-Building Telkom University
P-Building Telkom University

Figure 3: Statistical computations

In one practical application, SMPC helped Boston employers process confidential salary information by collaborating with them. This enabled them to generate summary statistics without exposing either party's individual salary information to the outside world. These implementations demonstrate the applicability of SMPC for managing privacy concerns in collaborative data analysis.

VI. MACHINE LEARNING APPLICATIONS

The combination of SMPC and machine learning has created potential opportunities for analyzing private data. Most machine learning methods require the collection of large amounts of data to make predictions with a certain degree of confidence, which often represents an invasion of privacy. This SMPC solves this problem by enabling different parties to collaboratively and securely train and run machine learning models without any of them disclosing their data to the others. One example of this application is a collaborative study conducted with contributions from the University of Pennsylvania and nine other institutions. Using a method called "federated learning" based on SMPC, a machine learning model was trained for MRI diagnosis in patients with brain tumors. This approach made it possible to distinguish healthy brain tissue from malignant brain tissue without exchanging patient records between institutions. However, it is important to note that the effectiveness and, in particular, the feasibility of practical implementation of SMPC with machine learning (ML) still require further research. The latter means that the choice of protocol and security model can significantly influence computation time, and the number of participants in the protocol can significantly influence this when increasing from 2 to 3, and even more so when transitioning from semi-honest security to malicious security.

Big Data Processing

Due to the increasing amount of data, the use of SMPC in processing large data sets is certainly worthwhile. However, current SMPC algorithms generally exhibit poor scalability in terms of data set, making their application to big data challenging. To address this problem, researchers have proposed several solutions that complement SMPC with other methods to improve scalability. One example of such an approach is a query compiler called Conclave, which accelerates data analysis by transforming queries into a combination of parallel data, local plaintext, and small SMPC steps. This mode of operation enables the system as a whole to process datasets that can be three to six orders of magnitude larger than those that existing SMPC frameworks can handle independently. Conclave offers the most advantages for relational analytic queries. It is designed to maintain the overall security of SMPC without relying on cryptographic SMPC for all operations. Indeed, Conclave can perform other actions outside of SMPC when parties entrust parts of the data to third parties, thus improving scalability by implementing new hybrid MPC-plaintext protocols.

This extension of SMPC for big data analytics makes it easier to perform privacy-preserving data analytics in various domains, including healthcare, finance, and research. As research in this area advances, SMPC will be used to analyze even larger datasets while maintaining a higher level of privacy.

SMPC compared to other privacy-preserving technologies. SMPC is one of several privacy-preserving techniques developed to preserve data confidentiality while enabling simultaneous analysis. In this section, the author compares SMPC with other common methods and discusses their respective advantages and disadvantages.

Differential Privacy

Differential privacy is a mathematical approach that adds a small amount of randomness, or noise, to the data to obscure an individual's contribution. This technique has gained widespread acceptance in the market, and major companies such as Google and Apple have integrated it into their systems. While at SMPC, we value the input data and ensure that it remains confidential during computation, DP ensures that the result of a data analysis does not contain information about specific individuals. At the same time, however, differential privacy is one of those techniques that provides a certain level of privacy. Furthermore, it clearly defines how much information about a specific individual can be derived from the analysis results. This makes it particularly useful for publishing summary statistics or machine learning models. However, there is a problem with differential privacy: additional noise can cause problems if there is not enough data available to balance the actual signal and noise. Furthermore, deciding how much noise to add (privacy budget) is equally difficult and may require practical experience. Federated Learning

Federated learning is defined as a distributed machine learning approach in which multiple parties collaborate to train a model toward a central goal while maintaining the confidentiality of their raw data. Instead, each party builds its local model from its own data and only transmits the gradients to a central server for accumulation. One advantage of this technique is the inherent scalability of SMPC computations, as it minimizes information sharing between the participating parties. Federated learning is particularly beneficial in situations where data can only remain at its source due to various constraints. However, federated learning alone does not provide strong data protection. Although the raw data remains local, model parameters shared between models can lead to information leakage. It has therefore been proposed to implement federated learning in conjunction with other data protection techniques such as differential privacy or SMPC.

Homomorphic Encryption

Homomorphic encryption (HE) is an encryption technique that allows computations on encrypted data without first decrypting it. Its most important property makes it usable for such computations when outsourced to untrusted providers, thus maintaining data security. Like SMPC, HE shares some similarities in that it also allows for the secure computation of private data. However, HE focuses on situations where one party has an interest in protecting data from an untrusted computing provider, whereas SMPC primarily targets multiple parties sharing data and wishing to compute a common function. This is a major advantage of HE because the data is constantly encrypted, thus providing good security guarantees. However, the computational overhead of HE is very high, making it unrealistic for many applications, especially those requiring complex computations or large datasets. SMPC, on the other hand, is more flexible and can efficiently solve a wider range of computations, albeit at the cost of some communication complexity. SMPC also provides greater security against collusion between network participants, provided that honest participants only make up a fraction of the total number.

Legal and Ethical Considerations for SMPC

Future applications of SMPC will present a number of legal and ethical challenges that organizations must address. As noted in the previous sections, there are few areas where SMPC is not relevant and does not have the potential to have a significant impact. These applications will raise various legal and ethical questions that organizations implementing or impacted by them will need to address. The results also demonstrate that SMPC enablers and inhibitors have a significant impact on compliance regulations, data protection laws, and ethics.

Regulatory Compliance

SMPC offers organizations a useful solution for complying with current legislation while ensuring the security of their information. By using SMPC-based key generation and management, CISOs benefit from improved data protection, regulatory compliance, and risk management. This approach is already gaining popularity among large enterprises such as financial, pharmaceutical, and automotive groups. SMPC offers many advantages in helping organizations comply with data protection laws, including GDPR and HIPAA. These regulations sometimes prevent the sharing of datasets even with authorized authorities that require them. The decentralized analysis capabilities provided by SMPC enable combined analysis of many datasets while maintaining the security of each individual input, thus overcoming the challenges of current regulations.

Data Protection Laws

The relevant law passed by the European Union is the GDPR, which contributes significantly to the protection of personal data. It aims to strengthen the protection of the human rights of individuals and citizens and facilitate business by providing a legal framework for organizations and government authorities in digitalized markets. The GDPR created a uniform law to eliminate legal differences between different national frameworks and avoid unnecessary bureaucracy.

Therefore, organizations subject to the GDPR can use SMPC to achieve the objectives of data protection regulations while performing the necessary calculations without data loss. This harmonization has created a new source of control for the SMPC and established a new form of trust for the exchange of data across organizations. Ethical Implications

Like any new practice, SMPC offers many benefits but also raises new ethical questions. This creates new types of requirements: the need to trust basic calculations and new, insecure forms of data misuse. Companies must grapple with various ethical issues related to SMPC, particularly in high-risk areas such as medicine and finance.

Another ethical issue is the ability to process data sets sufficiently to identify individuals. While de-identification and anonymization techniques have been introduced, these are currently considered insufficient to deter those attempting to circumvent them. Combined with differential privacy, SMPC offers a better approach to preventing information leakage, particularly with respect to individual records or data sets, except for intended and authorized information sharing.

VII. CONCLUSION

Secure multi-party computing (SMPC) represents a significant advancement that influences how businesses handle sensitive information and engage in complex collaborative computations. It has opened new avenues for addressing challenges in sectors such as finance, healthcare, and machine learning by enabling data analysis without compromising privacy. As SMPC technology evolves, the associated computational costs are decreasing, making its practical application more attainable across various industries. These developments provide a solid groundwork for broader adoption in multiple fields. The future of SMPC appears bright, with ongoing research aimed at enhancing its capabilities and resolving current limitations. In an era where privacy protection is increasingly critical, SMPC holds the promise of facilitating data analysis while safeguarding individual privacy. To fully realize the potential of SMPC, it is essential to consider the legal and ethical ramifications involved. This approach ensures that this transformative technology is utilized responsibly, fostering innovation while maintaining the security of sensitive data.

REFERENCES

[1] Peter, A., Tews, E. and Katzenbeisser, S., 2013. Efficiently outsourcing multiparty computation under multiple keys. *IEEE transactions on information forensics and security*, 8(12), pp.2046-2058.

[2] Goldwasser, S., 1997, August. Multi party computations: past and present. *In Proceedings of the sixteenth annual ACM symposium on Principles of distributed computing* (pp. 1-6).

[3] Brandt, F., 2005, December. Efficient cryptographic protocol design based on distributed El Gamal encryption. *In International Conference on Information Security and Cryptology* (pp. 32-47). Berlin, Heidelberg: Springer Berlin Heidelberg.

[4] Smart, N.P., 2003. Cryptography: an introduction (Vol. 3, p. 433). *New York: McGraw-Hill.*

[5] Canetti, R., 2000. Security and composition of multiparty cryptographic protocols. *Journal of CRYPTOLOGY*, 13, pp.143-202.

[6] Goldreich, O., 2003. Cryptography and cryptographic protocols. *Distributed Computing*, 16, pp.177-199.

[7] Demmler, D., Schneider, T. and Zohner, M., 2015, February. ABY-A framework for efficient mixed-protocol secure two-party computation. *In NDSS*.

[8] Chaum, D., Damgård, I.B. and Van de Graaf, J., 1988. Multiparty computations ensuring privacy of each party's input and correctness of the result. *In Advances in Cryptology—CRYPTO'87: Proceedings* 7 (pp. 87-119)..

[9] Naor, M. and Nissim, K., 2001, July. Communication preserving protocols for secure function *Springer Berlin Heidelberg* evaluation. *In Proceedings of the thirty-third annual ACM symposium on Theory of computing* (pp. 590-599).

[10] Canetti, R., Lindell, Y., Ostrovsky, R. and Sahai, A., 2002, May. Universally composable two-party and multi-party secure computation. *In Proceedings of the thiry-fourth annual ACM symposium on Theory of computing* (pp.494-503).

[11] Lindell, Yehuda, and Benny Pinkas. "An efficient protocolfor secure two-party computation in the presence of malicious adversaries." *In Advances in Cryptology-EUROCRYPT 2007: 26th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Barcelona, Spain*, May 20-24, 2007. Proceedings 26, pp. 52-78. *Springer Berlin Heidelberg*, 2007.

Citation of this Article:

Priyanka A. Patil, & Mohan Kulkarni. (2025). Protocols for Computing that Ensure Privacy and Safeguard the Inputs of All Involved Participants. *International Current Journal of Engineering and Science (ICJES)*, 4(8), 25-32. Article DOI: https://doi.org/10.47001/ICJES/2025.408005

****