International Journal of Interdisciplinary Innovative Research &Development (IJIIRD)

ISSN: 2456-236X

Vol. **04** Issue **02** |**2020**

Data Privacy Using Cryptographic Approaches

Astha Jain[1], Ishita Popli[2]

[1]*Master Student, Department of MCA, Jain Deemed-to-be University, Bengaluru, Karnataka, India*
[2]*Master Student, Department of MCA, Jain Deemed-to-be University, Bengaluru, Karnataka, India*

ABSTRACT

Privacy is a major concern that is focused on today's power of sight. If we talk about Machine learning it is a fast-growing field demanding a great deal of attention because of the latest advancements in information and network security. In the past three decades, machine learning techniques, whether supervised or unsupervised, have been applied in cryptographic algorithms. The visible progress in machine learning resulted in extreme curiosityfor data privacy on which it depends, and various methods and newly discovered techniques for preserving privacy. This study directed to the cryptographic approaches for Preserving Privacy in Machine Learning.

Security and privacy are the major concern for all developers and users who are exploring machine learning on a daily basis. The motivation behind doing this project is to create a platform like a shared mind which provides privacy in machine learning using the cryptographic techniques. We have created a platform which will cost less and will serve the same purpose with some features. The private data is used in various machine learning applications to provide the user with a better browsing experience.

Keyword*: Security, Cryptography, Homomorphic, AES, RSA, Secret sharing*

1. INTRODUCTION

Privacy and security are serious issues in the antiquity of data science and machine learning. For example, every day we do transactions, search for queries, watch videos, browse through various social media platforms; a great deal of information is being provided to each of these platforms which are stored and computed on daily basis. This private data is used in various machine learning applications to provide the user with better browsing experience [1].

Machine learning is being utilized for various applications from health care to security. Some applications require individual private data such as browsing history, company data, location, cookies etc. Such private data is uploaded in purescript form for ML algorithms to extract patterns and then build models. The problem is not only with threats which are linked with private data exposed to insider threat at these companies, or outsider threat which means the companies holding these data sets were hacked.

Applications based on Machine learning require private data which is processed by servers. Such private data can be utilized by attackers for malicious purposes such as threatening the company, illegal transactions, unauthorized access etc. [2] So, in order to preserve the privacy of data owners' different cryptographic approaches came into existence by which we have created a platform for data preservation.

2. CRYPTOGRAPHIC APPROACHES

Cryptography is a practice of safeguarding data and message via the utilization of cyphers so that only those individuals for whom the data is proposed can comprehend it and use it. Hence, avoiding illegal entre to important data. In this, the precede "crypt" means "hidden" and suffix "graphy" means "writing".
In this, the methods that are used to guard data are attained from mathematical notions and set of rule-based computations known as algorithms to translate communications in order to make it tough to decrypt it. Such algorithms are used to create cryptographic keys, digital validation, authentication to shield information secrecy, web surfing on internet and to defend, private money transactions like online banking, net banking, debit card and credit card.

Machine learning has widely spread in various fields but it leads to several vulnerabilities and threats to the privacy of data that is being used to setup the ML Model[8]. Hence, certain measures need to be taken to safeguard privacy, cryptography being one of the oldest and popular approaches to date to protect data. There are different cryptographic techniques that have been developed in the last few decades. In some of these approaches attaining improved competence combined having information owners contribute their encoded information to the calculation servers which will diminish the problems to a secure party computation setting. In accumulation to increase competence, these techniques have the advantage of not needing the input parties to be online.

2.1 Homomorphic encryption

During the last few years since the development of a fully homomorphic system in 2010, homomorphic encryption systems are been studied widely and eventually have become important in numerous diverse cryptographic.
In this approach, encryption is performed on the ciphertext using some encryption techniques which at the time of decryption results in the original text as if encryption was never performed on it [9]. Homomorphism is a technique which records two same types of algebraic equations in order to preserve their operations.

$$f(x * y) = f(x) * f(y) \quad (1)$$

This means action on the encoded information such as addition and multiplication might preserve the effect on the original text. There are plentiful partially and fully homomorphic encryption systems in the market. In term of security, Fully homomorphic encryption(FHE) outruns partially homomorphic encryption systems as it provides more security [10].

Partially Homomorphic Encryption: An encryption system said to be partially homomorphic if it demonstrates only one of the homomorphism properties that are additive and multiplicative but not in cooperation. Some instances of partially homomorphic encryption are El Gamal which is based on multiplicative homomorphism, Paillier which is based on additive homomorphism and RSA has a property of multiplicative homomorphism.
Fully Homomorphic Encryption: An encryption system is said to be fully homomorphic if it demonstrates multiplicative and additive homomorphism properties in cooperation. Till now only one such encryption system exits which are a lattice-based encryption system that was developed in 2010 by Craig Gentry as mentioned before. FHE is thought as the most powerful and secured system around the world to secure third party data in an efficient way.
Protocols were established to permit evaluation of two encrypted values, otherwise to accomplish secure multiplication and decryption actions, generally by "blinding" the encoded text by adding an arbitrary encoded value to the encoded value that needs to be secured [11].
Let's define out symbolization for encoded text(cipher text), original text(message), encryption, and decryption:

Encryption: E(m) = c
Decryption: D(c) = m

Assuming homomorphism, we formerly get:

$$E(m1) + E(m2) = E(m1+m2) \equiv D(E(m1 + m2)) = m1+m2 \quad (2)$$

In this all it is also important to remember that the computation performed on the plain text and the ciphertext are by nature homomorphic as well.

E(2) = c1 and 2 + c1 = c2 (3)
then D(c) = 2 + 2 = 4 (4)

The action of the data was preserved, in spite of the value being encoded [12].
It is vital to fact out that c1 and c2 both are equally cryptographically secure. It points out that though computations can be done on the encoded data, the resultant encrypted values are as protected as they were beforehand when the encryption was performed.
A more genuine instance: If the key that is a pubic key of RSA is exponent **e** and mod **r** and, then the encoding of a message ***x*** is assumed as:

$$E(m) = m^e \bmod r \quad (5)$$

The homomorphic property is then:

$$(m1).(m2)=m^e1m^e2 mod r=(m1m2)^e mod r=E(m1.m2) \quad (6)$$

Its basic examples are:

- RSA(Rivest–Shamir–Adleman)
- AES(Advanced Encryption Standard)

2.1.1 Advanced Encryption Standard (AES): The AES is a symmetric block cypher selected by The National Institute of Standards and Technology (NIST) to protect classified information throughout the world to encrypt sensitive data.It is necessary for computer security, cybersecurity and electronic data protection. The new advanced encryption algorithm is capable of protecting sensitive government information well [12]. It is easy to implement in hardware and software as well as in restricted environments for example, in smart card and defences against various attack techniques.

Steps in the AES Encryption Process:

AES consists of three block cyphers: AES-128, AES-192 and AES-256.The encryption process uses a set of derived keys called round keys. Steps included in encryption of 128 bit- block:

1. Get the set of round keys from the cypher key.
2. Initialize the state array with the block data (plain text).
3. Add the initial round key to the starting state array.
4. Perform nine rounds of state manipulation.
5. After nine rounds, perform the tenth and final round of state manipulation(involves a bit different manipulation from the other rounds).
6. Finally, copy the final state array out as the encrypted data (ciphertext).

AES works with byte quantities, so we will convert the 128 bits into 16 bytes. Each round of the encryption process requires a series of steps to change the state array.

2.1.2 Rivest–Shamir–Adleman(RSA): RSA is a very popular and widely used asymmetric key algorithm used for encoding and decoding of data. As it is asymmetric it makes use of two different keys for encryption and decryption. RSA makes use of a public key and a private key. The data is encrypted using the public key which is openly shared. Under RSA encryption, messages are encrypted with a code called a public key, which can be shared openly. Now if the data is encrypted using a public key it can be decrypted by a private key only.The private key is known to only the receiver that is the one who decrypts the data. Asymmetric key encryption is popular as it solves many security problems and enhances the security level at the same time[14]. In RSA, only the one who has the private key can access the encrypted data and decrypt it.

RSA Algorithm

RSA Comprises of two algorithms:

1. **Generation of keys:**
   a. Select two large prime numbers and q. The larger the number better the safety.
   b. Multiply the two numbers p and q and they're resultant will be considered as modulus n.
   c. Calculate the totient(ϕ) of n by using:
   $$\phi(n)=(p-1)\cdot(q-1) \quad (7)$$
   d. Derive a number e that is larger than 1 and smaller than $\phi(n)$. e should not have a common factor other than 1.
   e. The pairs formed by e and n formulate the public key.
   f. The private key is the multiplicative inverse of the public key which can be calculated using p,q and e.
2. **Function for RSA evaluation:** A function that takes an input and a key and depending on both it takes action whether to encrypt or decrypt data. If k = e than encryption and if k = d then decryption.

$$F(m,k)=m^k \bmod n \quad (8)$$

- **Encryption**:

$$F(m,e)=m^e \bmod n=c \quad (9)$$

- **Decryption**:

$$F(c,d)=c^d \bmod n=m \quad (10)$$

2.2 Secret sharing

Secret sharing is a very old and famous cryptographic approach with already existing everyday applications e.g. password management. For example, secret sharing has to secure computation and used for private machine learning. The idea behind this approach is that the owner wants to divide the secret into a number of portions and then distribute those shares to shareholders without giving them any knowledge about the secret, but if several shareholders recombine their shares then the secret can be restored. Therefore, integrity issue that used to depend on a single party is now depended and distributed to non-cooperated numerous parties. Secret sharing schemes are fascinating from a performance angle, as they depend on possible cryptographic assumptions. Specifically, certain issues such as factoring integers and computing discrete logarithms secret sharing schemes can provide a computational advantage compared to other cryptographic tools such as homomorphic encryption [16].

It is a seamless approach to securely store sensitive information. It can be used for keys used for encryption, code for missile launch, bank account number, etc. These were some information's that are confidential and therefore should be secured because if disclosed it can be very harmful toan individual or an organization and therefore should not be lost or misused. Encryption is not capable to provide such a high level of security and is therefore not very reliable because, at some point, they can be decoded. In case of encryption, we need to store the encryption keys, if we store it at one place it gives maximum secrecy whereas if we store it in multiple places it provides reliability but reduces confidentiality which makes it prone to attack vectors. So with the use of Secret sharing schemes, the confidentiality and reliability both can be maintained[17].

Shamir's Scheme: In additive secret sharing scheme if one shareholder losses its share or is no available then the secret cannot be constructed again that is referred to as R=N constraint. So, to deal with this problem the Shamir Scheme was created. It can remove this constraint by letting R(also T) choose the application.
In this scheme, we take a polynomial function F which fulfils the condition F(0) = sec where sec is the secret and then we calculate F at various non-zero coordinates which refers to various shares of the secret like F(1), F(2), F(3),…,F(n).
So, by changing the values (degree) of F we can set the number of shares that will be required for reconstruction of the secret. In this way, R = N constraint is removed and our secret is safe. Now, if we assume that the value of F is S then the value at S+1 or the coefficient of S+1 can be entirely known using interpolation, so we get that R=S+1 shares are required to reconstruct the secret.

```
def A(secret):
        poly = sample (secret)
        share = [ point(poly,p) for p in Share _points]
        return share
```

Fig 1: Secret Shares Creation

The Secret (S shares) is hidden because some description could be found for a guess (i.e. the value of F at point 0). Therefore, interpolation not only gives us S known shares but it also discovers the polynomial with a precise degree that goes with every value.

```
def reconstruct (shares):
        poly = [ (p,n) for p, n in zip(Share_points,
        shares) if n is not None]
        secret = interpose_at_point(poly,0)
        return secret
```

Fig 2: Secret Reconstruction

Encryption algorithm for building Garbled Circuits should have very similar encryption for multiple messages and precisely verified range. Precisely verifiable range means that given a key kk, it is possible to efficiently (polynomial time) verify if a given ciphertext is in the range of kk.
The encryption scheme should be secure under selected double-encryption. We encrypt the output label with two keys and it is not possible to obtain the plaintext with only one of the keys [18]. This requirement verifies that the evaluator cannot get the labels for other gates without both the corresponding keys. It is said that any scheme that is CPA-secure can be used to satisfy these security constraints.

3. PROPOSED SYSTEM

A platform which uses two approaches that are secret sharing and homomorphic encryption to provide a secure way of sharing and receiving information which will be used for training and testing of the applications that use machine learning.

This platform will contain a database and a server, it will collect the input in encrypted form and then it will be computed and processed using the two techniques to maintain its privacy.

Because of the secret-sharing approach, multi-party participation will be possible and therefore it will allow you to share your data with your partners without providing the whole data to everyone.

4. PROCEDURE

In this project, we have developed a platform that;

- Takes your input in the form of files or direct input.
- In both cases, the system asks you to set the security level.
- Once you choose the security level, it let you choose the key also as per your need in RSA and AES.
- A 128-bit key is used in AES for encryption and decryption.
- 1024 bit key is used in RSA for encryption and decryption.
- The encrypted and decrypted data are stored in different files that are created during the process.
- For secret sharing, Shamir scheme is used which is most effective and protected in terms of secret sharing schemes.
- The secret is divided into 8 different parts and you at least require four to recreate the secret that is distributed.
- Combination of RSA and AES is also an option for high-security level information.
- And at last, we have an option that combines all three approaches to provide security to critical and very high priority information.

5. FUTURE SCOPE

There is vast scope in the field of security as well as machine learning because both are very important and fast-growing fields. In future, there could be models with cryptography as a major component. There can be platforms which will provide security to data at all times not only at the time of training and testing phase of machine learning, that is storage, user interface, etc. With machine learning being new, user-friendly systems could be developed in combination with artificial intelligence, big data analytics, cloud computing, etc. But everything stops at one point that is security and therefore, there is a vast scope of development for new cryptographic techniques and platforms that provides the facility of security using them.

6. CONCLUSION

The field of machine learning is growing at a fast pace and but not much application developers are taking privacy as a serious issue which leads to compromise with private and sensitive data. At the time of training and testing of

ML-based application, these cryptographic approaches should be used to provide security. Finally, we have proposed a system that utilizes these two approaches (homomorphic encryption and secret sharing) to build a secure platform that can be utilized to preserve privacy in machine learning.

7. REFERENCES

[1] Jure Sokoli´c, QiangQiu, Miguel R. D. Rodrigues, and Guillermo Sapiro "Learning to Succeed while Teaching to Fail: Privacy in Closed Machine Learning Systems", 23 May 2017.
[2] Mohammad Al-Rubaie and J. Morris Chang "Privacy-Preserving Machine Learning: Threats and Solutions",2018.
[3] The code blocks team "http://www.codeblocks.org/", 29 March 2020.
[4] The code blocks team "https://en.wikipedia.org/wiki/CodeBlocks", March 2020.
[5] Microsoft Company https://www.techwalla.com/articles/list-of-ms-word-features", 2016.
[6] Richard Brodie "https://www.digitalcitizen.life/beginners-guide-notepad", 1983.
[7] Nicolas Papernot, Patrick McDaniel, Arunesh Sinha, and Michael P. Wellman "Security and Privacy in Machine Learning", 09 July 2018.
[8] Bernard Marr "What Is Homomorphic Encryption? And Why Is It So Transformative? ", 15 November 2019.
[9] V. Chellappan, K.M. Sivalingam "Security and privacy in the Internet of Things", 2016.
[10] Xun Wang, Tao Luo and Jianfeng Li "A MoreEfficient Fully Homomorphic Encryption Scheme", 16 December 2018.
[11] Jaydip Sen "Homomorphic Encryption – Theory and Applications", 3rd May 2012.
[12] Margaret Rouse "https://searchsecurity.techtarget.com/definition/Advanced-Encryption-Standard", 2015.
[13] Vincent Rijmen, Joan Daemen "https://aesencryption.net/" , 1998.
[14] Josh Lake "https://www.comparitech.com/blog/information-security/rsa-encryption/#What_is_RSA_encryption", 10 December 2018.
[15] Shireen Nisha, Mohammed Farik "RSA Public Key Cryptography Algorithm- A Review", July 2017.
[16] Morten Dahl "Secret sharing, Part 1", 04 June 2017.
[17] Hameed, Ali "Simple games with applications to Secret Sharing Schemes", 2013.
[18] Andreas Poyiatzis "Shamir's Secret Sharing - A numeric example walkthrough", 23 September 2018.