|  |  |  |
| --- | --- | --- |
| D:\usr\campos\TSB-Reference\Logos\ITU\sigleITU.gif | INTERNATIONAL TELECOMMUNICATION UNION**TELECOMMUNICATIONSTANDARDIZATION SECTOR**STUDY PERIOD 2017-2020 | SG17-TD2897 |
|  |  | **STUDY GROUP 17** |
|  |  | **Original: English** |
| **Question(s):** | 4/17 | Geneva, 17-26 March 2020 |
| **TD** |
| **Source:** | Rapporteur Q4/17 |
| **Title:** | Proposal for new work item: Security guidelines for FHE-based data collaboration in machine learning |
| **Purpose:** | Proposal |
| **Contact:** | Jihoon ChoSamsung SDSKorea (Republic of) | Tel: +82-10-8254-2040E-mail: jihoon1.cho@samsung.com |
| **Contact:** | Donggeon YheeIMDARCKorea (Republic of) | Tel: +82-10-8931-2945 Email: dgyhee@gmail.com |
| **Contact:** | Jae Hoon NahETRIKorea (Republic of) | Tel: +82-42-860-6749 Email: jhnah@etri.re.kr |

|  |  |
| --- | --- |
| **Keywords:** | Fully Homomorphic Encryption, Machine Learning, Data Analysis, Data Aggregation, Inference |
| **Abstract:** | This temporary document (TD) proposes to set up a new work item on security guidelines for FHE-based data collaboration in machine learning.Annex A: The proposed baseline document.Annex B: Template to describe a new work item to ITU-T Recommendation. |

**Annex A**

#### Scope

This draft TD provides security guidelines for secure inference services and data aggregation in machine learning using fully homomorphic encryption (FHE) technology. This provides the platform, where a data owner uses inference services from a machine learning model provider while each party does not reveal their own data, e.g. an input data to an inference service and machine learning coefficients. This also describes how two parties collaborate to obtain a machine learning model with two data sets as input without data leakage. The draft also describes underlying FHE schemes and secure parameter selection.

#### References

The following ITU-T Recommendation and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation.

[ITU-T Y.XXXX], TBD

#### Terms and Definitions

#### Terms Defined Elsewhere

This Recommendation uses the following terms defined elsewhere:

<TBD>

#### Terms Defined in This Recommendation

This Recommendation defines the following terms:

####  Encryptor

a module which takes as inputs a public encrypting key, and a data in plaintext, and returns an encrypted data as an output

####  Decryptor

a module which takes as inputs a secret decrypting key, and a data in encrypted state, and returns a (partially) decrypted data as an output

####  Evaluator

a module which takes as inputs the public evaluation key and encrypted data, and returns an encrypted data as an output

####  Merge

a module which takes two or more partial decryption results and returns a decrypted message as an output

#### Abbreviations and Acronyms

This Recommendation uses the following abbreviations and acronyms.

AI Artificial Intelligence

FHE Fully Homomorphic Encryption

ML Machine Learning

#### Conventions

This Recommendation uses the following conventions:

The keywords "**is required to**" indicate a requirement which must be strictly followed and from which no deviation is permitted, if conformance to this Recommendation is to be claimed.

The keywords "**is recommended**" indicate a requirement which is recommended but which is not absolutely required. Thus, this requirement need not be present to claim conformance.

The keywords "**is prohibited from**" indicate a requirement which must be strictly followed and from which no deviation is permitted, if conformance to this Recommendation is to be claimed.

The keywords "**can optionally**" indicate an optional requirement which is permissible, without implying any sense of being recommended. This term is not intended to imply that the vendor's implementation must provide the option, and the feature can be optionally enabled by the network operator/service provider. Rather, it means the vendor may optionally provide the feature and still claim conformance with the specification.

#### Overview

Statistical analysis or Artificial Intelligence (AI) increasingly needs to share the data among organizations and individuals to obtain greater accuracy and outsource AI services. Such data, however, is often sensitive, including details about individuals or organizations that could be abused to identify sensitive information of individuals or reveal credentials of organizations.

In particular, data collaboration is increasingly required between computing devices or nodes belonging to, often, multiple organizations. More specifically, a data owner may use inference services from a machine learning model provider while each party does not reveal their own data, e.g. an input to an inference service by a data owner and machine learning coefficients by inference service provider. Two parties would also like to collaborate to obtain a more accurate machine learning model with two data sets as input while not revealing their data sets to each other.

This draft TD analyses data leakage issues for data collaboration in machine learning described above, and provides a platform for secure inference and data aggregation in machine learning using fully homomorphic encryption (FHE) technology. The draft also describes definitions of FHE scheme and secure parameter selection.

#### Data leakage analysis

<TBD>

#### Platform structure and interfaces

<TBD>

#### Workflows for FHE-based Machine Learning

Fully homomorphic encryption (FHE) allows arbitrary computations of data in its encrypted form. Since FHE could preserve both addition and multiplication using approximation techniques, it becomes possible to evaluate most functions in machine learning on encrypted data without decrypting them.

When data owners want to make use of well-trained machine learning model, e.g. cancer prediction models with DNA as input, DNA information may be compromised during the inference process. The DNA is now encrypted using FHE, and the data owner transfers the encrypted DNA and its public key to a service provider who assesses risk and prognosis for the cancer. The service provider encrypts the model and computes the result with the encrypted DNA as input. The service provider cannot access to both input and output of prediction services, since there is no decryption key at all within the service provider environment. The encrypted inference result is sent back to the data owner who is only able to decrypt the result with the secret key.

FHE can also be applied to the case that two organizations need to collaborate for data aggregation. Each data set from two organizations could be encrypted using the same public key and computed in untrusted computing environments, e.g. public cloud, and the encrypted result securely decrypted by either threshold manner between them or a trusted third party.

We now describe a platform where two parties can securely share their data without data leakage issue for both inference and training cases in machine learning. The elements below make use of algorithms in FHE schemes in section 10.

#### Secure Inference



The *party A* would like to inference with its data as input using a machine learning model provided by the *party B*. The two parties, however, do not want to disclose its data and the machine learning model, respectively.

1. The *party A* and *party B* generate their partial secret keys and the common public key in a threshold manner. *(Related standards will be provided.)*

2. The *party A* encrypts *data A* which is an input to inference, and the *party B* also encrypts *data B* which is a machine learning model. The two data are encrypted using the same *Encryptor*, and the outputs are sent to a server.

3. Server takes as inputs encrypted *data A* and *data B*, and then returns the inference result in encrypted form using an *Evaluator* and sends it to the *party A* and the *party B.*

4. The *party A* and *party B* decrypt the encrypted output using the *Decryptor* with its own partial secret key as an input.

5. The *party B* sends its partial decryption result to the *party A*. Two partial decryption results are merged to output the inference result using *Merge*.

#### Secure Data Aggregation



The *party A* would like to perform a training with its own *data A* as well as *data B*, and the *party B* does not want to disclose its own data.

1. The *party A* and *party B* generate their partial secret keys and the common public key in a threshold manner. *(Related standards will be provided.)*

2. The *party A* and *party B* encrypts their own data using the same *Encryptor*, and each encrypted data is sent to the server.

3. Server takes as inputs encrypted *data A* and *data B* using to an *Evaluator*, and the output is sent to both the *party A* and the *party B.*

4. The *party A* and *party B* decrypt the encrypted ML model using the *Decryptor* with its own partial secret key as an input.

5. The *party B* sends its partial decryption result to the *party A*. Two partial decryption results are merged to output a trained machine learning model using *Merge*.

A trusted thirty party (TTP) may be involved in secure data aggregation, where a TTP performs the key generation and also provides decryption services for the participating parties.

#### FHE Schemes

Homomorphic encryption is a breakthrough new technology which has been discussed in the literature in many years, and also there exist a list of open source libraries which have implemented FHE schemes as below.

|  |  |  |
| --- | --- | --- |
| **Libraries** | **Organizations** | **Supporting FHE schemes** |
| HEAAN[[1]](#footnote-1) | SNU | CKKS [b-CKKS17] |
| HElib[[2]](#footnote-2) | IBM | BGV [b-BGV11/BGV], CKKS [b-CKKS17] |
| Lattigo[[3]](#footnote-3) | EPFL | BFV [b-B12/BFV], CKKS [b-CKKS17]  |
| PALISADE[[4]](#footnote-4) | Duality | BFV [b-B12/BFV], BGV [b-BGV11/BGV], CKKS [b-CKKS17], etc. |
| SEAL[[5]](#footnote-5) | Microsoft | BFV [b-B12/BFV], CKKS [b-CKKS17] |

#### Definitions and Notations

#### Recommended Security Properties

####  Hard Problems

####  Tables of Recommended Parameters

**Bibliography**

[b-B12/BFV] Brakerski, Zvika. "Fully homomorphic encryption without modulus switching from classical GapSVP." *Annual Cryptology Conference*. Springer, Berlin, Heidelberg, 2012.

[b-BGV11/BGV] Z. Brakerski, C. Gentry, and V. Vaikuntanathan. [Fully Homomorphic Encryption without Bootstrapping](http://eprint.iacr.org/2011/277), In *ITCS 2012*

[b-CKKS17] J. H. Cheon, A. Kim, M. Kim, Y. Song, *Homomorphic Encryption for Arithmetic of Approximate Numbers*. In ASIACRYPT 2017. Pages 409–437.

[b-CHKKS18] J. H. Cheon, K. Han, A. Kim, M. Kim, Y. Song, Bootstrapping for Approximate Homomorphic Encryption. In EUROCRYPT 2018. Pages 360–384.

[b-DD15/FHEW] Ducas, Léo, and Daniele Micciancio. "FHEW: bootstrapping homomorphic encryption in less than a second." *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, Berlin, Heidelberg, 2015.

[b-FV12/BFV] Fan, Junfeng, and Frederik Vercauteren. "Somewhat Practical Fully Homomorphic Encryption." *IACR Cryptology ePrint Archive*2012 (2012): 144.

[b-HES] M. Chase, H. Chen, J. Ding, S. Goldwasser, S. Gorbunov, J. Hoffstein, K. Lauter, S. Lokam, D. Moody, T. Morrison, A. Sahai, and V. Vaikuntanathan. *Security of Homomorphic Encryption*. http://homomorphicencryption.org/white\_papers/security\_homomorphic\_encryption\_white\_paper.pdf

[b-IEEE P3652.1] IEEE P3652.1 – Guide for Architectural Framework and application of Federated Machine Learning

[b-ISO/IEC 18033-6] ISO/IEC 18033-6:2019 Part 6: Homomorphic encryption

[b-SS13] Stehle, D., and R. Steinfeld. "Making ntruencrypt and ntrusign as secure as standard worst-case problems over ideal lattices." *Cryptology ePrint Archive, Report 2013/004* (2013).

**Annex B**

Template to describe a proposed new ITU-T Recommendation

|  |  |  |  |
| --- | --- | --- | --- |
| **Question:** | Q7/17 | **Proposed new ITU-T Recommendation** | 17-26 March 2020 |
| **Reference and title:** | Recommendation ITU-T X.sgfdcml: Security guidelines for FHE-based data collaboration in machine learning |
| **Base text:** | Annex A of this document  | **Timing:** | 09-2021 |
| **Editor(s):** | Jihoon Cho, Samsung SDS, jihoon1.cho@samsung.comDonggeon Yhee, IMDARC, dgyhee@gmail.comJae Hoon Nah, ETRI, jhnah@etri.re.kr | **Approval process:** | AAP |
| **Scope** (defines the intent or object of the Recommendation and the aspects covered, thereby indicating the limits of its applicability): |
| This draft Recommendation analyses data leakage issues for data collaboration in machine learning, i.e. secure inference and data aggregation for training in machine learning, and provides a security guideline for secure inference and data aggregation in machine learning using fully homomorphic encryption (FHE) technology. The draft also describes definitions of FHE scheme and secure parameter selection. |
| **Summary** (provides a brief overview of the purpose and contents of the Recommendation, thus permitting readers to judge its usefulness for their work): |
| Statistical analysis or Artificial Intelligence (AI) increasingly needs to share the data among organizations and individuals to obtain greater accuracy and outsource AI services. Such data, however, is often sensitive, including details about individuals or organizations that could be abused to identify sensitive information of individuals or reveal credentials of organizations. In particular, data collaboration is increasingly required between computing devices or nodes belonging to, often, multiple organizations. More specifically, a data owner may use inference services from a machine learning model provider while each party does not reveal their own data, e.g. an input to an inference service by a data owner and machine learning coefficients by inference service provider. Two parties would also like to collaborate to obtain a more accurate machine learning model with two data sets as input while not revealing their data sets to each other. This draft Recommendation analyses data leakage issues for data collaboration in machine learning described above, and provides a platform for secure inference and data aggregation in machine learning using fully homomorphic encryption (FHE) technology. The draft also describes definitions of FHE scheme and secure parameter selection. |
| **Relations to ITU-T Recommendations or to other standards** (approved or under development): |
| IEEE P3652.1, ISO/IEC 18033-6 |
| **Liaisons with other study groups or with other standards bodies:** |
|  IEEE P3652.1, ISO/IEC JTC 1/SC 27/WG2 |
| **Supporting members that are committing to contributing actively to the work item:** |
| Korea (Rep. of), ETRI, Samsung Electornics Co. Ltd.  |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. <https://github.com/snucrypto/HEAAN> [↑](#footnote-ref-1)
2. <https://github.com/shaih/HElib> [↑](#footnote-ref-2)
3. <https://github.com/ldsec/lattigo> [↑](#footnote-ref-3)
4. <https://gitlab.com/palisade/palisade-release> [↑](#footnote-ref-4)
5. <https://github.com/microsoft/SEAL> [↑](#footnote-ref-5)