European Commission
RESEARCH & INNOVATION
Horizon 2020

prisma cloud

Privacy and Security Maintaining Services in the Cloud

Contract Number: 644962
Call: H2020-ICT-2014-1

Deliverable D4.9

ANALYSIS OF THE STATE OF THE ART OF FPE, OPE AND TOKENIZATION SCHEMES

Deliverable due date: 01.02.2016
Deliverable submission date: 01.02.2016

Document Information	
Title	Analysis of the state of the art of FPE, OPE and tokenization schemes
Creator	Boris Rozenberg
Deliverable no.	4.9
Work Package No.	4
Nature	Report
Dissemination Level	Public
Release Date	01/02/2016
Document description	Review of the state of the art of FPE/OPE techniques as well as analysis of the use case requirements and requirements definition for encryption library

Authors List

Organization	Name	E-mail
IBM	Boris Rozenberg	borisr@il.ibm.com

Reviewers List

Organization	Name	E-mail
TU Graz	Daniel Slamanig	daniel.slamanig@tugraz.at
TUDA	Denise Demirel	ddemirel@cdc.informatik.tu-darmstadt.de

Versioning

Version	Date	Reason/Change	Editor
0.1	7/1/16	Draft document for review	Boris Rozenberg
0.2	14/1/16	Addressing reviewers' comments	Boris Rozenberg

The work described in this document has been conducted within the PRISMACLOUD project. This document reflects only the PRISMACLOUD Consortium view and the European Union is not responsible for any use that may be made of the information it contains.

List of Contents

List of Figures

1. Executive Summary

This deliverable provides an overview of the state of the art on Format Preserving Encryption and Order Preserving Encryption. Based on the requirements and needs of the PRISMACLOUD project use cases, we refine and layout the specific requirements of the FPE/OPE and show how techniques that will be developed during the project could be used in relevant use cases.

2. Abbreviations and acronyms

FPE – Format Preserving Encryption

OPE – Order Preserving Encryption

DFA – Deterministic Finite Automaton

NFA – Non-deterministic Finite Automaton

AES – Advanced Encryption Standard

GPFE – Genereal Format Preserving Encryption

RtE – Rank-then-Encipher

PRP – Pseudo Random Permutation

SPI – Single Point Indistinguishability

MP – Message Privacy

MR – Message Recovery

IND-OCPA – Indistinguishability under Ordered Chosen-Plaintext Attack

POPF-CCA – Pseudorandom Order-Preserving Function against Chosen-Ciphertext Attack

3. Introduction

An encryption scheme is a triplet of probabilistic polynomial-time algorithms $(KeyGen, Enc, Dec)$ such that $KeyGen$ generates a random string k (the *key*); Enc on input a message $m \in M$ (a set of *plaintexts* or messages) and a key k outputs a ciphertext $c \in C$ (a set of *ciphertexts*); and Dec on input a ciphertext c and a key k outputs a message m such that $c = Enc(m, k)$.

Encryption schemes are used to hide information (i.e., the message m) from unauthorized parties, while still allowing the authorized parties to read m. However, standard encryption schemes (such as AES) can significantly alter the data format, causing disruptions both in storing and using the data. Indeed, when storing devices and applications are designed to operate on unencrypted data they may not be able to operate on encrypted data. Consequently, Format-Preserving Encryption (FPE) schemes, namely schemes which encrypt messages into ciphertexts with the same format, have emerged as a most useful tool in applied cryptography. Formally, a Format-Preserving Encryption (FPE) scheme for format *M* is an encryption scheme with the additional property that *M* = *C*. Most FPEs studied in the literature are designed to encrypt only specific formats (e.g., credit-card numbers), while we focus on General-Format Preserving Encryption (GFPE), which can encrypt messages from various formats.

Order-Preserving Encryption (OPE) is a type of deterministic encryption whose encryption function preserves numerical ordering of the plaintexts. That is, a remote untrusted database server is able to index the (sensitive) data it receives, in encrypted form, in a data structure that permits efficient range queries. In fact, as pointed out in [1] OPE not only allows efficient range queries, but allows indexing and query processing to be done exactly and as efficiently as for unencrypted data, since a query just consists of the encryptions of real values (plaintexts) and the server can locate the desired ciphertexts in logarithmic-time via standard tree-based data structures. Formally, an Order-Preserving Encryption (OPE) scheme for format *M* is an encryption scheme with the additional property of maintaining the order between messages. That is, for every key k and every pair $m_1 \leq m_2$ of messages in *M*, $Enc(m_1, k) \leq Enc(m_2, k)$.

4. State of the art

In this section we provide analysis of state of the art in the area of Format Preserving Encryption and Order Preserving Encryption.

4.1. Format Preserving Encryption

First studied in the context of integral domains (namely, when the message domain is $\mathcal{M} = \{0,1,\ldots,m-1\}$ for some $m \in \mathcal{N}$) [2], later works [3] considered more general formats, and two general techniques were suggested for for FPE design. First, the cycle walking strategy of Black and Rogaway [2] constructs an FPE for format $\mathcal{F}$ from any FPE for a format $\mathcal{F}'$ such that $\mathcal{F} \subseteq \mathcal{F}'$. The encryption algorithm for $\mathcal{F}$ repeatedly applies the encryption algorithm of $\mathcal{F}'$, until the ciphertext lies in $\mathcal{F}$. (Decryption is repeated until reaching a valid string in $\mathcal{F}$.) For example, an FPE scheme for credit-card numbers can use cycle-walking on AES (which is an FPE for $\{0,1\}^{128}$).

Second, the *Rank-then-Encipher* (RtE) method suggested by Bellare et al. [3] reduces the task of designing an FPE for format $\mathcal{F}$ to the task of designing and FPE for an integral domain. (In particular, the RtE framework allows one to apply the same encryption logic to all formats, thus eliminating the need to design specially-tailored encryption schemes for every format.) More specifically, a format $\mathcal{F}$ of size N is arbitrary ordered as $\mathcal{F} = \{s_0,\ldots,s_{N-1}\}$, and encryption (decryption) is based on an integer-FPE (i.e., for an integral domain), where a string $s \in \mathcal{F}$ is encrypted in three steps, called ranking, integer-encryption, and unranking. First the index i such that $s = s_i$ is found; then i is encrypted into an index j, using the integer-FPE encryption algorithm; finally, the encryption of s is the message s_j. (Decryption is performed in the same manner by replacing the integer-FPE encryption with the decryption algorithm). If $\mathcal{F}$ has a deterministic finitie automaton (DFA), then $\mathcal{F}$ has efficiently computable ranking and unranking algorithm [4]. The scheme inherits its security from the integer-FPE, while ranking and unranking do not contribute to security. Efficiency of the scheme relies heavily on the efficiency of ranking and unranking. The combination of cycle-walking and RtE yield an FPE scheme for any "rankable" format, which raises the question of designing efficient ranking and unranking methods for general formats. Although ranking can use translation-tables, such tables cannot be constructed efficiently, require expensive storage, and do not admit efficient searching algorithms. Moreover, designing a single encryption scheme for several formats raises the question of efficiently representing formats, since a representation of the format will be given as input to the encryption algorithm.

Several works suggested FPEs for specific formats, such as fixed-base, fixed length vectors (i.e., $\{0,1,\ldots,m\}^n$ for $n, m \in \mathbb{N}$) [5], [6]; and more practical message-domains such as social-security numbers [7], credit-card numbers, and dates [8]. These schemes are tailored for specific formats, and it is not clear whether, and how, they can be generalized. The ranking strategy suggested for more general formats (e.g., names, addresses, etc.) [9], [10] partitions the format into many sub-formats, where the messages in each sub-format share additional characteristics (e.g., length), and therefore raises both efficiency and security concerns.

Regarding security, the schemes maintain "cosmetic" characteristics of the message which are not part of the properties defining the format, thus allowing an attacker to deduce (from the ciphertext) many message-specific characteristics, which do not follow from m having format $\mathcal{F}$.

This renders the schemes completely insecure in theory and practice. Regarding efficiency, the scheme of [9], [10] is inefficient in practice. First, they do not suggest a method of efficiently representing formats, and partitioning the format into sub-formats (which must be done before encryption) is too costly to be performed in practice, since it depends on the number or plaintexts, rather than their lengths as in non-format-preserving encryption schemes. Second, all formats (even when $|\mathcal{F}| \gg 2^{128}$, which is the case for many practical formats) are encrypted using the same methods. As these schemes rely on integer-FPEs, they are inefficient in practice. Thirdly, as the scheme admits no method of representing complex format properties, encryption can be inefficient even for medium-sized format due to cycle-walking, which repeatedly applies the "heavy" operations of integer-FPE encryption and decryption. Therefore, the average cycle length may be long, and more importantly, there is no worst-case bound on the actual cycle length. This motivates eliminating the use of cycle-walking.

To overcome these problems, Luchaup et al. [11] developed libFTE – a unfying format-preserving and format-transforming encryption scheme (in format-transforming encryption, all ciphertexts are guaranteed to have format $\mathcal{F}_0$, which may differ from the message format). libFTE also employs the RtE method, where regular expressions (regexes) represent formats, that are ranked using either a corresponding deterministic finite automaton (DFA) or non-deterministic finite automaton (NFA). More specifically, a DFA can be obtained through a general regex-to-DFA transformation, which is not always efficient. To allow the use of the (more efficient) regex-to-NFA transformation, the authors relax the ranking method, such that it can also be based on an NFA.

In [12], we proposed an efficient FPE scheme with optimal security. Our scheme includes an efficient method of representing general (complex) formats, and provides efficient encryption and decryption algorithms that do not require an expensive set-up. During encryption, only format-specific properties are preserved, while all message-specific properties remain hidden, thus guaranteeing data privacy.

The main shortcoming of existing schemes is their inflexibility in format representation: they offer a single, very specific method of representing general formats, focusing on a specific set of properties (length and location specific character-sets), while ignoring all other format properties. As we have shown in [12], this results in a scheme which is insecure and achieves nonoptimal efficiency. Our scheme is also based on the RtE framework, but by providing a flexible framework of representing general formats, we improve security and efficiency. We cannot possibly predict all formats to which our scheme may be applied, so we supply several format "building-blocks" (primitives), from which compound formats are constructed by applying "composition operations" which we define [12]. Next, we provide efficient ranking and unranking methods for all formats representable in our framework. Concretely, we provide ranking and unranking algorithms for all primitives and composition operations. Thus, primitives are ranked directly; and compound fields can be ranked using the ranking method of the composition operations with which they were constructed.

As experimental results show that in many cases large format domains cannot be encrypted efficiently, we extend our scheme to support large formats, by imposing a user-defined bound on the maximal format size, thus obtaining a flexible security-efficiency tradeoff and the best possible security (under the size limitation). The proposed scheme was also filed for US patent [13].

Though presenting a general FPE scheme, our goals, focus, and solutions, are very different than introduced in [11]. First, libFTE is designed for developers, and as such provides the developer with several possible schemes, out of which she chooses the most appropriate one. Our scheme is designed to be incorporated into a larger system which is designed for the end-user, so it must provide a single scheme, and the flexibility of our system is obtained by setting (according to the clients' specifications) few parameters "once and for all" in the larger, "wrapper" system. Second, formats in our scheme are defined directly, and naturally, from their user defined properties, and is therefore flexible since the user can define new formats himself. Defining new formats in libFTE requires a developer's involvement to construct a regex from the user-defined format properties. This representation using regexes has the additional disadvantage of nonuniformity, since the performance of the resultant scheme depends on the specific regex chosen to represent the format, as opposed to the "complexity" of the format (as in our scheme). Moreover, there is no method of predicting whether the resultant scheme would have poor performance, and if it does, the developer cannot know whether a different regex would give better performance. We note that both our scheme, and libFTE, have the same security guarantees (since the underlying non-FPE scheme is the same in both).

Security notions for FPEs

Intuitively, encryption schemes should be "as unpredictable as possible", i.e., given a ciphertext an adversary should be unable to deduce any properties of the encrypted message, and this should hold even given prior knowledge on the message, and (possibly also) other ciphertexts encrypted using the same key. However, FPEs cannot achieve these security notions since they inherently reveal the message format. Consequently, the following four FPE-specific game-based security notions have been suggested [3]. Pseudo-Random Permutation (*PRP*) security requires that an adversary cannot distinguish encryptions with a randomly chosen key from random permutations over the format domain; single-point indistinguishability (*SPI*) requires that the adversary cannot distinguish the encryption of any message of its choice from a random ciphertext; message privacy (*MP*) requires that ciphertexts reveal no information on the encrypted message, except its format (this is formalized by comparing the "performance" of the real-world adversary to that of a degenerate adversary that can only make equality queries of the form "is *m* the encrypted message?"); and similar to *MP*, but weaker than it, message recovery (*MR*) only requires that the ciphertext does not completely reveal the encrypted message. The two latter security notions should hold even if the adversary can choose the message distribution to its advantage. (These security notions are non trivial since the degenerate adversary *S* operates on the same message distribution. For example, if the distribution is concentrated on one message, *A* has no advantage over *S* since both can recover the original message.) In all cases, the adversary is also given an encryption oracle. Roughly speaking, the advantage $Adv^X(A)$ of an adversary A (where *X* ϵ *{PRP, SPI,MP,MR}*) is the difference between the probability that *A* correctly guesses which situation he is in, and the probability of guessing correctly when only the format is known (in the first two cases, this probability is $1/2$) [3]. Bellare et al. [3] show that PRP→SPI→MP→MR, meaning PRP is the strongest security notion and MR is the weakest. We note that though PRP is the best security notion one can hope to achieve for FPEs, the three weaker notions can, in many concrete cases, offer better security for the same efficiency, and may therefore suffice in practice.

4.2. Order Preserving Encryption

One of the earliest treatments of the concept of order preserving encryption (OPE) was proposed in the database community by Agrawal et al. [14] in 2004. The proposed method allows efficient range queries on encrypted data (i.e., an untrusted remote server can index the sensitive data while the data is encrypted). However, the scheme is not always practical, because the encryption algorithm must take as input all the plaintexts in the database, and, in some cases, the users do not know all the plaintext in advance. Another drawback is that the scheme does not have a rigorous security analysis.

The first formal cryptographic treatment of OPE was presented in [1] by Boldyreva et al. in 2009. The authors define and formalize the security definition for OPE. They show that the security notion IND-OCPA (indistinguishability under ordered chosen-plaintext attacks, which is the strongest security notion for OPE) is unachievable by any OPE scheme, unless the ciphertext-space is exponential to the plaintext-space. The authors propose a new security notion called POPF-CCA (pseudorandom order-preserving function against chosen-ciphertext attack) and a practical and efficient deterministic blockcipher based scheme. The authors prove that the scheme's security meets the new security definitions. However, the authors pointed out that even the “ideal” object (ROPF – random order-preserving function) in POPF-CCA inherently leaks some information about the underlying plaintexts. The study of the leakage was out of the scope of the paper. So, due to the lack of understanding the security properties of the “ideal” object in POPF-CCA, the authors did not recommend using the scheme. Furthermore, they showed that ROPF with practical range size does not hide the distances between plaintexts. Despite of that, the OPE scheme received considerable attention in the applied community [15] [16].

In [17], Boldyreva et al., analysed ROPF (random order-preserving function, and the “ideal” object in the security notion POPF) and showed that for a database which contains randomly distributed plaintext and appropriate choice of parameters, ROPF leaks at least half of the bits (this was also shown in [18]), which allows an adversary to approximate the value of any plaintext as well as approximate distance between any two plaintexts, each to an accuracy of about square root of the domain size. Then, they propose a technique that improves the security of any OPE scheme. The improved scheme is not OPE, but it allows range queries. The technique is simple and generic: the encryption algorithm just adds a secret offset to the message before encryption (the secret offset is the same for all the messages). The new improved scheme is called modular OPE (MOPE). It hides the plaintext values, but still leaks the distances between any two plaintexts. Additionally, a significant drawback of the scheme is that if the adversary has at least one plaintext-ciphertext pair, then the security of the scheme falls back to that of a ROPF.

Pursuing more secure schemes, in [19], Popa et al. showed that it is infeasible to achieve ideal security IND-OCPA (indistinguishability under ordered chosen plaintext attacks) with non-mutable ciphertext OPE without certain implicit assumptions. They present a mutable order-preserving scheme (mOPE) which achieves ideal security. The scheme’s main technique is called mutable ciphertexts, meaning that over time, the ciphertexts for a small number of plaintext values change.

The scheme is interactive (unlike other schemes, which have restrictive and non-interactive interface). The average communication complexity (between the client and the server) of the scheme is $O(n \cdot \log(n))$ (each insert/delete/search operation takes $\mathcal{O}(\log(n))$, where n is the number of elements in the database). However, the authors claim that their testing results shows that the performance of mOPE achieves 1-2 orders of magnitude higher performance than the scheme in [1].

In [20], Kerschbaum and Schroepfer proposed an ideal OPE scheme, which is an improvement of the scheme in [19]. Although, the average communication complexity is $O(n)$, the proposed scheme is still not practical because it is interactive.

In [21], Malkin et al. present a new indistinguishability-based security notion for OPE called (X, θ, q)-Indistinguishability, which can ensure secrecy of the lower bits of a plaintext. Then they propose a new scheme satisfying this security notion. However, the scheme has limitations - to ensure the secrecy of the lower bits, the plaintexts should be distributed uniformly at random, which is almost never the case in practice.

The fundamental problem of deterministic OPE algorithms is that the ciphertexts can leak information about the distribution of the plaintexts. To overcome this problem, Redday et al. [22] propose a randomized OPE (ROPE). Although ROPE meets the security notion of IND-OCPA, it is less efficient than deterministic OPE (such as MOPE, basic OPE and even mOPE).

5. Use case needs

This section lays out the relevant requirements of the Smart Cities – European Disable Badge for Public Parking Areas use case [23] as this use case is the one interested in Format Preserving Encryption (FPE) and Order Preserving Encryption (OPE). We describe in details the requirements for the use case and show how they can be addressed by employing FPE and OPE discussed in this document.

The use case is built upon the results of SIMON project [24]. As described in details in [23], citizens holding a personal European Disable Badge and make use of a reserved parking lot will register their use of the lot by using a combination of the three technologies involved in the system, namely RFID badge, smartphone and park meter. The SIMON system will check if the user has the permission to park there, granting or denying the operation. Parking controllers make use of SIMON system to enforce a fair and efficient use of the reserved parking lots. The following figure, taken from [23] summarizes the scenario:

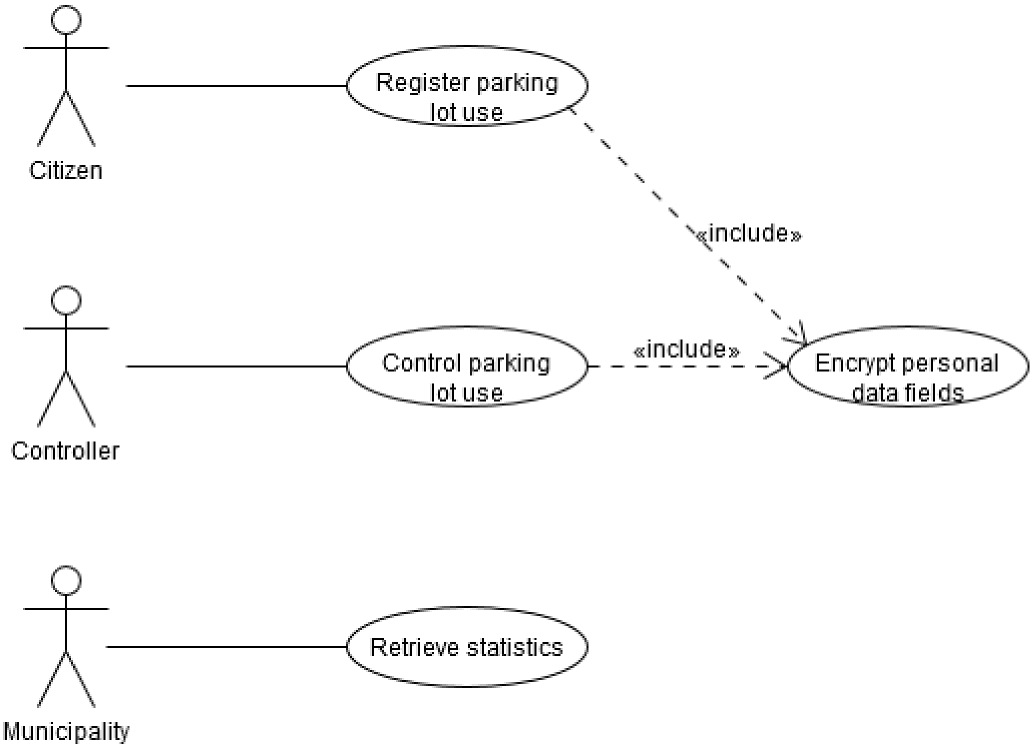

Figure 1 - Smart City Use Case

SIMON deploys a supporting service platform called SIMON SAYS. SIMON SAYS provides the core identity management functions to enable the validation and verification of users and parking spaces. This service interacts with three applications: a mobile application for citizens called SIMON LEADS, a mobile application for parking controllers called SIMON CONTROLS, and backoffice application for the public authorities, parking and transport operators called SIMON Trusted Service Manager (SIMON TSM). Currently, backoffice systems are deployed in a private datacenter, which is located at ETRA facilities. This may be enough for pilot purposes, but involves personnel of the company in tasks regarding 24/7 support and maintenance of the physical infrastructure and is not actually scalable to meet the requirements of a regular deployment. Migration of these backoffice systems to the cloud is therefore desirable. Currently, since all backoffice systems are deployed in private dependencies, only a minor focus has been put on the privacy issues involved. Users of the system are only identified by their User Id (UID) - no user personal identification (name, physical address, etc.) is stored in SIMON dependencies. Nevertheless, attached to these UIDs, the following personal information is stored:

- User phone number
- User email
- User vehicle's license plate
- User location when making use of the system
    - GPS location provided by the smartphone or
    - Parking meter ID being used
- Disabled Citizen Database identification number (this identification number is linked to extensive personal information of the user in another database - out of the scope of SIMON - which is maintained by the municipality

Migration of the backoffice to the cloud as is will raise privacy related issues. Indeed, the personal data mentioned above will no longer be handled in private dependencies. It must be therefore ensured that it cannot be accessed and used by third parties (not even the cloud provider) in any case. To address this requirement, mechanisms will be put in place to ensure that, on the one hand, the information stored in the cloud is properly encrypted, and on the other hand, backoffice's functionality is not compromised. In order to achieve this objective, the following architecture of the cloudified system will be set [23]:

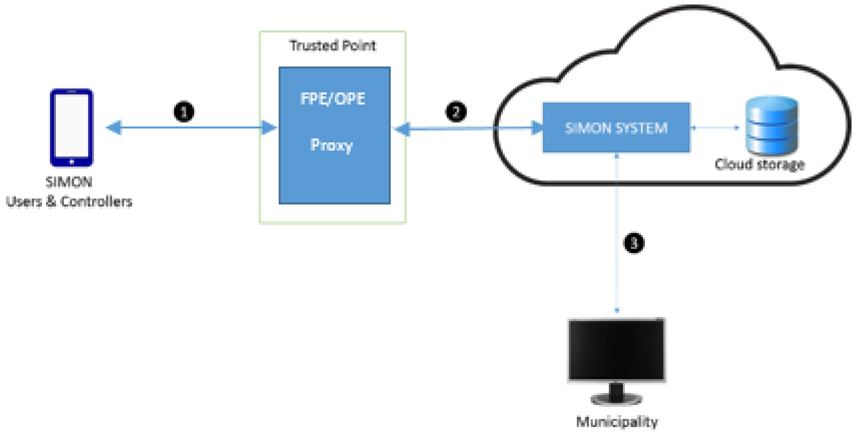

Figure 2 - Cloudified System Architecture

Messages going to and coming from the smartphones (citizens and controllers) will pass through the Format/Order Preserving Encryption Proxy, that implements the necessary mechanisms to encrypt sensitive fields of the messages on the fly. Personal information fields in messages in point 1 of Figure 2 do have clear text, while the same fields in messages in point 2 of Figure 2 are encrypted. As a consequence, the SIMON system can be cloudified with minor or no changes at all, while personal data is managed in a secure way since it is encrypted when it reaches the cloud.

A direct communication channel to the cloud (point 3 in Figure 2) will still exist for exploitation operations where no personal information of the users is required (e.g., retrieving statistics of the system). The proposed solution ensures that applications employing this second channel will at most have access to encrypted versions of the personal data items.

6. Requirement specification

In order to support the use case needs discussed above, we will develop an encryption library. Following is a summary of the requirements for the library:

1. Provide the user the ability to define general formats.
2. Support Format Preserving Encryption for the following formats:
    - User phone number
    - User email
    - User vehicle's license plate
    - User location:
        - GPS location provided by the smartphone or
        - Parking meter ID being used
    - Disabled Citizen Database identification number
3. Support Order Preserving Encryption for the following formats:
    - User location:
        - GPS location provided by the smartphone
4. Support API to perform FPE and OPE
5. FPE and OPE of single message should complete within reasonable time (e.g., tens of miliseconds)

The library will be integrated, as part of the use case, in the trusted point shown in Figure 2. For demonstration purposes only, the trusted point may be realized using a proxy provided by IBM. While the complete description of this proxy is out of the scope of this document, it is important to note that the proxy will be deployed at IBM Haifa and will support HTTP messages only. To support other formats, another proxy could be deployed. The encryption library could be integrated in this proxy as well.

Originally, we planned to develop Format Preserving Tokenization (FPT) techniques (in addition to FPE and OPE). However, according to our analysis of use case requirements [23], there is no need for FPT functionality in this project. With this being said, we will concentrate our efforts on developing FPE techniques, OPE techniques, and a proxy that applies aforementioned encryption techniques.

7. Conclusions

In this document we provided a state of the art in the area of Format Preserving Encryption (FPE) and Order Preserving Encryption (OPE). We also analysed the use case needs and refined the specific requirements of the FPE and OPE components.

Our plan for year 2 of PRISMACLOUD is to perform a detailed design of the encryption library, including description of algorithms that will be developed and definition of interfaces for the proxy and end user.

8. Bibliography

[1] A. Boldyreva, N. Chenette, Y. Lee and A. O'Neill, "Order-Preserving Symmetric Encryption," in *EUROCRYPT*, 2009.

[2] J. Black and P. Rogaway, "Ciphers with arbitrary finite domains," *Topics in Cryptology,* pp. 114-130, 2002.

[3] M. Bellare, T. Ristenpart, P. Rogaway and T. Stegers, "Format-preserving encryption," *Selected Areas in Cryptology,* pp. 295-312, 2009.

[4] A. V. Goldberg and M. Sipser, "Compression and ranking," *SIAM J. Comput.,* vol. 20, no. 3, pp. 524-536, 1991.

[5] M. Bellare, P. Rogaway and S. Terence, "The FFX mode of operation for format-preserving encryption," Submission to NIST, 2010.

[6] E. Brier, T. Peyrin and J. Stern, "BPS: a format-preserving encryption proposal," Submission to NIST, 2010.

[7] D. N. Hoover, "Format-preserving encryption via rotating block encryption," US patent application 0280394, 2011.

[8] Z. Liu, C. Jia, J. Li and X. Cheng, "Format preserving encryption for datetime," *Intelligent Computing and Intelligent Systems (ICIS),* vol. 2, pp. 201-205, 2010.

[9] L. W. Martin, T. Spies and M. J. Pauker, "Format preserving encryption systems for data strings with constraints," patent application 0103579, 2011.

[10] M. Bellare, P. E. Catinella, P. K. Hazel, C. Von Mueller and S. R. Yale, "System and method for variable length encryption," US patent application 0211689, 2011.

[11] D. Luchaup, K. P. Dyer, S. Jha, T. Ristenpart and T. Shrimpton, "Libfte: A toolkit for constructing practical, formatabiding encryption schemes," in *23rd USENIX Security Symposium*, San Diego, 2014.

[12] M. Weiss, B. Rozenberg and M. Barham, "Practical Solutions For Format-Preserving Encryption," in *W2SP*, 2015.

[13] B. Rozenberg and M. Weiss, "Complex format-preserving encryption scheme," US Patent Application Number: 14/296484, 2014.

[14] R. Agrawal, J. Kiernan, R. Srikant and Y. Xu, "Order Preserving Encryption for Numeric Data," in

ACM SIGMOD, 2004.

[15] W. Lu, A. Varna and M. Wu, "Security analysis for privacy preserving search of multimedia," in *Image Processing (ICIP)*, 2010.

[16] C. Wang, N. Cao, J. Li, K. Ren and W. Lou, "Secure Ranked Keyword Search over Encrypted Cloud Data," in *ICDCS*, 2010.

[17] A. Boldyreva, N. Chenette and A. O'Neill, "Order-Preserving Encryption Revisited: Improved Security Analysis and Alternative Solutions," in *CRYPTO*, 2011.

[18] L. Xiao, O. Bastani and L. Yen, "Security analysis for order preserving encryption schemes," in *CISS*, 2012.

[19] R. Popa, F. Li and N. Zeldovich, "An Ideal-Security Protocol for Order-Preserving Encoding," in *IACR*, 2012.

[20] F. Kerschbaum and A. Schropfer, "Optimal Average-Complexity Ideal-Security Order-Preserving Encryption," in *ACM Conference on Computer and Communications Security*, 2014.

[21] T. Malkin, I. Teranishi and M. Yung, "Order-Preserving Encryption Secure Beyond One-Wayness," in *ASIACRYPT*, 2014.

[22] K. s. Reddy and S. Ramachandram, "A New Randomized Order Preserving Encryption," *International Journal of Computer Applications,* 2014.

[23] PrismaCloud, "Deliverable 2.3, Use case specification," 2016.

[24] "SIMON Project," [Online]. Available: http://simon-project.eu/.