International Journal of Advanced Science and Technology
Vol.124 (2019), pp.21-32
http://dx.doi.org/10.33832/ijast.2019.124.02

A STUDY OF SECURE DISTRIBUTED MANAGEMENT OF SENSING DATA IN IOT ENVIRONMENT

Ho-Kyung Yang[1], Hyun-Jong Cha[2] and You-Jin Song[3*]

[1]*Division of Information Technology Education, Sunmoon University, 70, Sunmoon-ro 221 beon-gil, Tangjeong-myeon, Asan-si, Chungcheongnam-do, 31460, Korea*
[2]*Division of Information Technology Education, Sunmoon University, 70, Sunmoon-ro 221 beon-gil, Tangjeong-myeon, Asan-si, Chungcheongnam-do, 31460, Korea*
[3]*Dept. Of Management, Dongguk University, 707, Seokjang-dong, Gyeongju, Gyeongsangbuk-do, 38066, Korea*
[1]porori0421@naver.com, [2]chj826@kw.ac.kr, [3]*song@dongguk.ac.kr

***Abstract*— For commercialization of communication with IoT (Internet of Things) and M2M (Machine-to-Machine) in connecting tens of trillions of sensors and other devices, communication infrastructure that can significantly reduce the number of data flows compared to existing infrastructure is required to effectively deal with data originating from several transmitters. This paper discusses the secret sharing- based safe sensing data management method in terms of distributed data management in the Fog Computing environment. The study method is based on a privacy function allowing context aware secret sharing in disclosing information by adjusting the amount and degree of disclosed personal information to an individual circumstance. Dynamic information partition and recovery is possible by controlling personal information, locational information and medical information by an individual; in other words, introducing the concept of context space.**

***Keywords*— Internet of Things, Secret Share, Distributed Management, Context Space**

1. INTRODUCTION

If the Internet of Things (IoT) service is activated, serious security issues including infringement of privacy and cyber-attacks may occur, as well as network performance issues due to the increase of network traffic. In addition, increased use of cloud services to analyze big data about increased traffic makes movement paths of data longer, which may cause network overload and exposure of sensitive data. Furthermore, real-time connectivity, low response latency and improved security are required to realize ideal IoT in which data is sent anytime anywhere. To meet these requirements, Fog Computing processes need to handle a greater percentage of data near network edges [2][3].

Fog Computing can be divided into three major areas; things (*e.g.*, smart devices) area, fog area and cloud area. First, the things area consists of items that detect the surrounding environment. The fog area includes fog nodes that have higher computing power compared to ordinary nodes. Examples include PCs, routers and home appliances. Items

Received: January 6, 2019
Reviewed: March 4, 2019
Accepted: March 8, 2019
* Corresponding Author

ISSN: 2207-6360 IJAST

included in the things area and nodes in the fog area are connected by near field communication. Fog nodes are also connected with the cloud area by wire/wireless communication. There exists a systemic connection between nodes in the fog area.

For commercialization of communication with the IoT and M2M (Machine-to-Machine) to tens of trillions of sensors and other devices, communication infrastructure that can significantly reduce the number of data flows compared to existing infrastructure is required to effectively deal with data originating from several transmitters. Sensor data keeps generating although the size is normally small. To classify and process such data (streaming data), it is important to process a certain amount of data within the network by collecting data at a place near a sensor (device) and installing an edge (Fog) to conduct pre-treatment, real-time treatment or other Quality of Service(QoS) control of sensing data and deal with change of a protocol[5-7].

With the development of Fog Computing, sensing information could be easily exposed due to the application of sensor and contextual information. In addition, infringement of privacy could be more serious due to increased exposure of sensing information deriving from the increased automatic support systems. Furthermore, it is expected that privacy issues will influence the daily life environment because of sensing and contextualization of private information, behaviour patterns, habits, medical history and symbols. Particularly, personal identity and locational information are main elements of privacy that could be infringed in the Fog Computing environment. Privacy is the right that allows an individual, a group or an organization to decide when, how and what information is delivered or exposed. The ultimate method to protect privacy is complete prevention of exposing personal information. However, information is valued when it is delivered or distributed. There is existing privacy protection technology regarding privacy protection and management and use of personal information known as PET (Privacy Enhancing Technology). However, most Internet privacy studies have mainly focused on personal information access control, DB encryption and data mining concerning personal information (ID, resident registration number) leakage prevention technology. Existing privacy protection technologies assume a model to duly and legitimately use personal information. Although privacy can be protected if personal information is completely hidden, a user may not enjoy customer service provided in the Fog Computing environment, such as customized service. Therefore, there should be a plan for how sensitive information is disclosed. In other words, a mechanism reflecting dynamic information (locational and biometric information) is required.

Other important security issues include safe management of various content and large data files to prevent security vulnerability or privacy infringement caused by outside attackers or insider users as a variety of data service is available.

In this paper, distributed data management in the Fog Computing environment concerning safe management of sensing data based on secret sharing is discussed. Chapter 2 introduces relevant studies. Chapter 3 describes the structure of context space. The design of context-based weighted tables (context table) is explained in Chapter 4. The conclusion follows in Chapter 5.

2. RELATED RESEARCHES

2.1. SECURE SHARING SCHEME

The secure sharing scheme, firstly proposed by Blakely and Shamir [8] in 1979, divides secrete information in several pieces, evenly distributes the pieces to people and reconstructs information with these pieces as necessary. Shamir's (k, n) threshold-based secret sharing scheme adopting polynomial interpolation divides secrete information into (n) pieces and reconstructs the information if (k) pieces are recovered. Many secrete

sharing schemes [8-10] have been proposed since the first secrete sharing concept was introduced.

However, the threshold-based secret sharing scheme proposed by Shamir is not suitable for dynamic data processing environments due to its static nature in which a new polynomial interpolation is created, partial information is redistributed, and relevant data should be provided on the request of information by a distributor. To address these concerns, Cachin's online secrete sharing scheme was proposed in 1995. Although the scheme is suitable for a dynamic structure, it requires additional computation processing to protect information in each area. Therefore, the scheme could not reflect dynamic data under Fog Computing as the scheme has difficulties in processing dynamic data. In other words, a data scheme to allow distributed dynamic data processing is required.

2.2. FOG COMPUTING

Fog computing, proposed by CISCO, has gained popularity as the concept of IoT is widely introduced. IoT exchanges a variety of information between devices including home appliances, cars and equipment with built-in chips through wired/wireless networks [4]. Fog computing allows a company the opportunity to process data at various locations. Particularly, it is very useful in areas requiring quick data processing.

Fog Computing creates a network connection with low latency between a device and an end point of analysis. This method reduces required bandwidth compared to systems that send data to a data center for processing. Fog Computing can be used when data needs to be processed at a place near where it was generated because it can be difficult to secure bandwidth for transmitting data. Another advantage of Fog Computing is the increased security from split network traffic to a virtual fire wall to protect traffic.

Examples of Fog Computing are found in several areas. First, data generating from vehicles are rapidly increasing due to the application of automatic and semi-automatic driving systems. In addition, it is also used in a manufacturing system that requires immediate action when an event happens and in real-time analysis systems of financial institutions that use real-time data to provide information for decision making and fraud transaction monitoring. In such systems, Fog Computing makes data transmission smoother between the location in which data is generated and the location where data is received.

3. STRUCTURE OF CONTEXT SPACE

3.1. DEFINITION OF CONTEXT

Context is defined as information that can be used to define the characteristics of a substantial circumstance [11]. Existing computing systems are unable to consider the circumstances of service users. For example, the service provider is unable to cognize the current activities of service users and provide them with services in time. Accordingly, context information is essential in a ubiquitous environment.

For example, context can be used to define characteristics of information requesters. One entity can have several contexts and, by contrast, one context can define the characteristics of only one entity and cannot explain the characteristics of multiple entities.

Context-aware deals with diverse technologies that enable analysis, correction, and reuse of the meanings of data of existing contexts. This paper focuses on both the real-time nature and meaning analysis of context.

3.2. DEFINITION OF CONTEXT SPACE

Context space is multi-dimensional space that consists of dimensions for each contextual element. A type defined in a context attribute decides a single reference line.

For simplicity, the context space in this paper indicates a generated area of context that can be expressed in a four-dimensional form (location, activity, role, and time). All contexts can be shown as a point in a context space. A conceptual diagram of the four-dimensional Context Space is shown in Fig. 1.

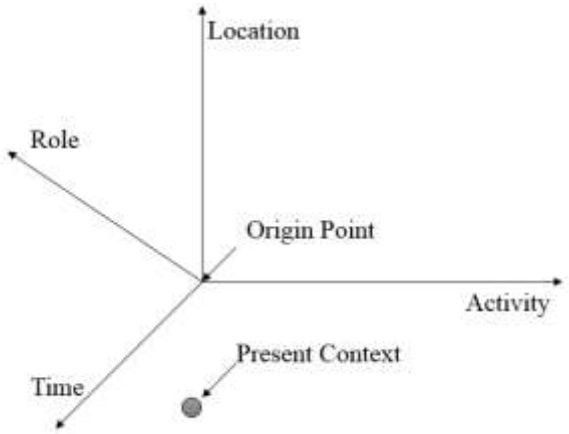

Fig. 1 Context Space

The distance between the current context and the origin shown in the context space has important meaning. If each context is ordered sequentially by a specific criterion, the gap between the current context point and the origin can provide a direct basis for how much information to disclose.

It is assumed that the student has specified the following preferences as listed in Table I.

Table I: User Preferences

Place	Time	Open or Not
School	10:00 am	Don't Open
Library	4:00 pm	Don't Open
Home	11:00 am	Don't Open

In this case, context space is formed as shown in Fig. 2.

When time and location are considered, the preference area (sensitive area) appears in the form of a circle as a two-dimensional space. On the other hand, when time, location and activity are considered, the preference area (sensitive area) appears in the form of a sphere, as a three-dimensional space as shown in Fig. 3.

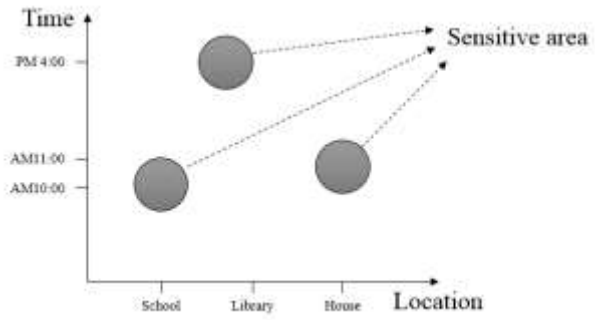

Fig. 2 Two-dimensional Context Space

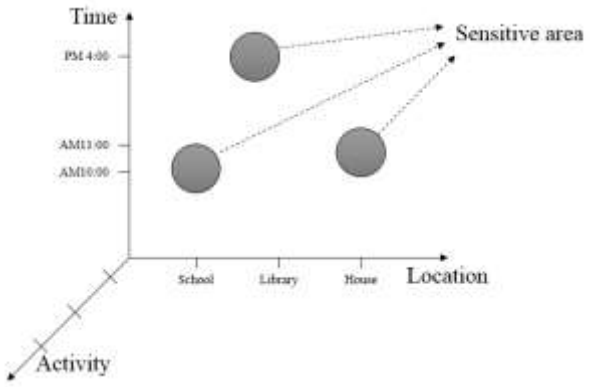

Fig. 3 Three-dimensional Context Space

User's information is not disclosed when a context value is located within the sensitive area.

3.3. HOW TO CONFIGURE CONTEXT SPACE

Context can be configured descending from high premise to low premise. For example, a meeting (context element) is configured under work (premise). Context code for each element can be defined using such method. Figure 4 indicates how to configure context code by element.

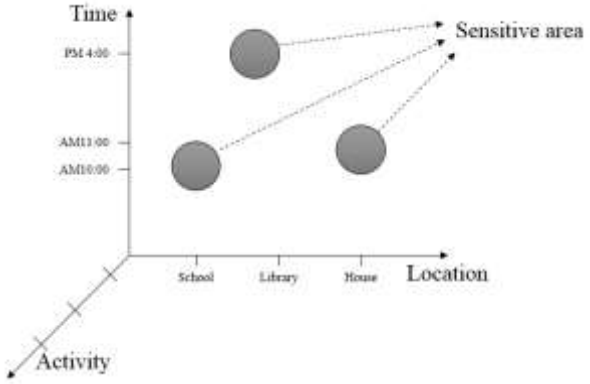

Fig. 4 Configuring Context Codes

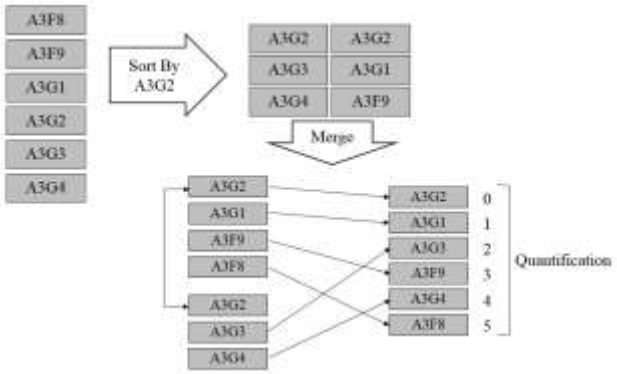

Fig. 5 Procedure for Quantification Through Context Code Sorting

Context code can act as a measure for context as shown in Figure 4. For example, a context similar to A3-G2 (for example, A3-G3) would be the most similar context. Therefore, the context code can be used to list specific contexts. Context code can be used as a basis for sorting context. For example, in the case of Figure 4, when sorting is performed in priority order, the result is shown in Figure 5.

The digitalization procedure involving the context codes that appear in Figure 5 is explained as follows:

1) The contexts within an appropriate scope are collected (the scope in which the user may appear).

2) The contexts are arranged via the reference point.

(1) The contexts are arranged below the reference point.

(2) The contexts are arranged above the reference point.

3) Two data sets are merged.

(1) A 2-1 data set is brought in and placed.

(2) A 2-2 data set is brought in and placed.

(3) This is repeated until 50 % of each set of data is reached (50 + 50 = 100 %).

4) The merged data are linearly sequenced.

5) The data is digitally sequenced.

3.4. MEANING OF SENSITIVE AREA IN CONTEXT SPACE

If privacy infringement occurs in the library at 12:00 a.m., it can be expressed as shown in Fig. 6.

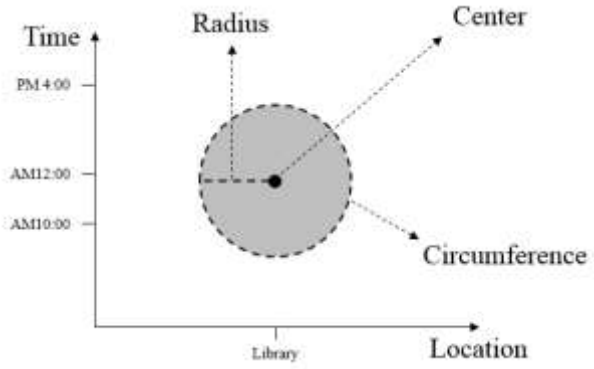

Fig. 6 Sensitive area of context space

This can be thought of in the following manner:

- Central point: The closer it is to the origin point (the corresponding section), more information infringement occurs.

- Radius: Privacy invasion value (degree of privacy infringement)

- Circumference: Privacy Invasion Point, or the threshold of privacy infringement. Privacy infringement occurs in reference to this point in time.

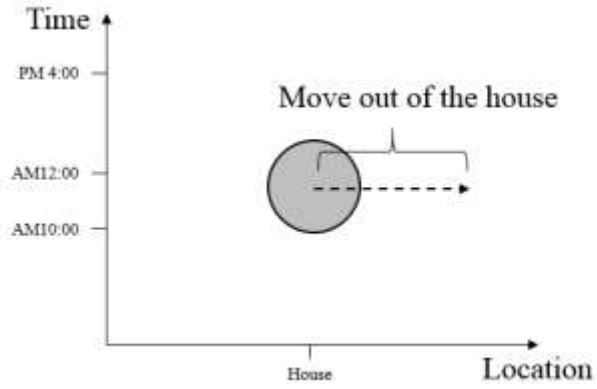

Fig. 7 Scenario of Location information

Examples of context space are as follows. Sally does not want to provide her locational information when she is at home. The ability to obtain locational information depends on people's location. Movement in the context space when Sally leaves home is shown in Fig. 7.

When Sally leaves home, risk of information exposure is low. Therefore, the possibility of Sally's locational information to be divulged is high when she is far from home.

4. DESIGN OF CONTEXT-BASED WEIGHT TABLE(CONTEXT TABLE)

4.1. CONFIGURATION OF WEIGHT TABLE AFFECTED BY CONTEXT

According to Beimel's paper, weight can be express in nlogn and τ=nlogn[12].

Context is affected by weight. We assume context as τ for convenience of calculation. In other words, the context space shown in Figure 8 can be created.

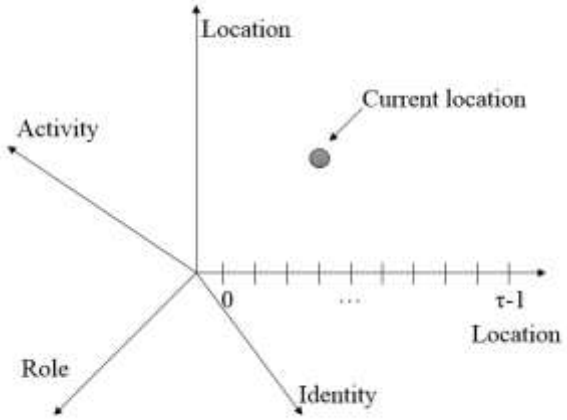

Fig. 8 τ-1 Context space

The τ-1 scale can be marked on each vertical line. Therefore, the following W-table is created. For instance, an example is the context figure lower than 32. Assuming user's context, it is the same as Table II.

Table II. User's Context

Location	Time	Activity	Role	Identity
12	5	27	18	7

Therefore, the following W-table is created.

- L(Location): Convert the number 2 to 1100 in binary
- T(Time): Convert the number 5 to 101 in binary
- A(Activity): Convert the number 27 to 11011 in binary
- R(Role): Convert the number 18 to 10010 in binary
- I(Identity): Convert the number 7 to 111 in binary

These values can be expressed in a W-table format as shown in Fig. 9.

L	T	A	R	I
0	0	1	1	0
1	0	1	0	0
1	1	0	0	1
0	0	1	1	1
0	1	1	0	1

Fig. 9 Weight Table

The sum of horizontal sides (i) starting from the bottom, ai, can be expressed as a0=3, a1=3, a2=3, a3=2 and a4=2. The w-table is created on the basis of formulas. Therefore, the w-table shown in Table III is created.

Table III. The weight generated according to the context

Snid	Weight Value
S1	3
S2	3
S3	3
S4	2
S5	2

The buffer size is (a0=3) + (a1=3) + (a2=3) + (a3=2) + (a4=2), which is 13. The threshold is +1, which is 7.

There are advantages when a weight table is configured on the basis of context. First, a structure that only allows restoration when a restorer is located at a certain context is possible. For example, for information that is only allowed accessible noon, a weight table is created through a relevant context, and secrete sharing work starts based on the table. When secrete pieces are transmitted without a w-table in delivering information, a restorer obtains a w-table from his/her own context. Information can be restored only when the w-table is located in a corresponding context.

4.2. SAFETY

Assume that there is a third party who has bad intentions to obtain information. The unauthorized person is located at a position within the context space as shown in Fig. 10.

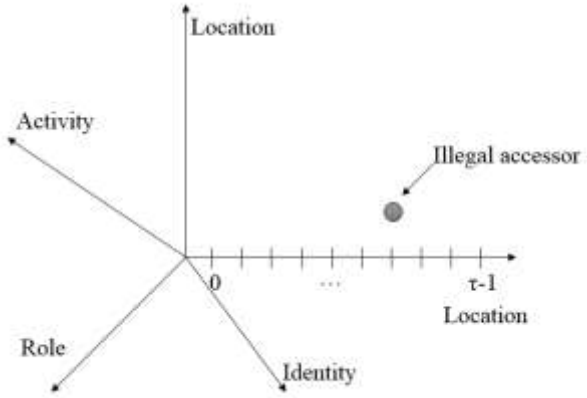

Fig. 10 Context Space for Illegal Accessors

The corresponding context values of the unauthorized accessor is indicated in Table IV.

Table IV. Context of illegal accessors

Location	Time	Activity	Role	Identity
15	7	18	31	4

With the context, the following w- table is created.

- L(Location): Change the number 15 to 1111 in binary.
- T(Time) : Change the number 7 to 111 in binary.
- A(Activity) : Change the number 18 to 10010 in binary
- R(Role) : Change the number 32 to 11111 in binary
- I(Identity) : Change the number 7 to 100 in binary

These values can be expressed in a W-table format as shown in Fig. 11.

L	T	A	R	I
0	0	1	1	0
1	0	0	1	0
1	1	0	1	1
1	1	1	1	0
1	1	0	1	0

Fig. 11 The context-generated weight table

Therefore, a0=3, a1=3, a2=3, a3=2 and a4=2. The w-table is created on the basis of formulas. Therefore, the w-table shown in Table V is created.

Table V. The weight generated according to the context

Snid	Weight Value
S1	3
S2	4
S3	4
S4	2
S5	2

As a result, the value differs from that on the weight table previously created. In this case, the data could not be properly restored.

5. CONCLUSION

Various sensors constituting the Internet environment of things generate a large amount of data in real time. However, in order to generate too much data, the user may not immediately respond to the request, so the reliability and immediacy of the user's data request must be guaranteed. Although the size of each data is small, the number of generated data is large, so a system capable of efficient data processing is required.

In existing environments, data processing technology is difficult to apply things to the Internet environment as a problem that you cannot use devices with heterogeneous devices and mobility. Therefore, methods considering heterogeneous characteristics of heterogeneous devices and techniques for processing devices having dynamic characteristics have been studied. The data generated in the Internet environment of things is in the form of big data. Therefore, data processing technologies that solve the problems of existing methods are studied, but research on distributed data processing technology considering the dynamic network environment in the Internet environment of things is in short supply.

Although the integration of IoT infrastructure, platform and the cloud environment for service is in progress, the lack of consistent security policy and delayed decision making remain challenges. Therefore, a new type of customized security service that is effective for different network environments (*e.g.*, IoT, CPS, Cloud and Bigdata) is required. This paper discusses secret sharing- based safe sensing data management method in terms of a distributed data management under the Fog Computing environment. The method is based on a privacy function allowing context aware secret sharing in disclosing sensing information by adjusting the amount and degree of disclosed personal information to individual circumstances.

ACKNOWLEDGEMENT

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2016R1D1A1B03931689). This work was also supported by the Dongguk University Research Fund of 2016.

REFERENCES

[1] D. Bandyopadhyay and J. Sen, “Internet of Things: Applications and Challenges in Technology and Standardization”, Wireless Personal Communications, Vol.58, No.1, (2011), pp.49-69.
[2] D. Miorandi, S. Sicari, F. De Pellegrini and I. Chlamtac, “Internet of things: Vision, applications and research challenges”, Ad Hoc Network, Vol.10, No.7, (2012), pp.1495-1516.
[3] L. Coetzee and J. Eksteen, “The internet of things-promise for the future? An introduction”, Proceedings of the IST-Africa Conference, Gaborone, Botswana, (2011), pp.1-9.
[4] L. Atzori, A. Iera, and G. Morabito, “The internet of things: A survey”, Computer Networks, Vol.54, No.15, (2010), pp.2787-2805.

[5] J. Gubbia, R. Buyyab, S. Marusica and M. Palaniswamia, “Internet of Things (IoT): A vision, architectural elements, and future irections”, Future Generation Computer Systems, Vol.29, No.7, (2013), pp.1645-1660.
[6] C.Sarma, Amardeo and J. Girao, “Identities in the future internet of things”, Wireless personal communications, Vol.49, No.3, (2009), pp.353-363.
[7] Jeff Dean, “Handling Large Datasets at Google: Current Systems and Future Directions”, Data-Intensive Computing Symposium, (2008).
[8] A. Shamir, “How to share secret”, Communications of the ACM, Vol.22, No.11, (1979), pp.612-613.
[9] J. Kurihara, S. Kiyomoto, K. Fukushima, and T. Tanaka, “On a Fast (k,n)-Threshold Secret Sharing Scheme”, IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Vol.91, No.9, (2008), pp.2365-2377.
[10] C. Cachin, “On-line Secret Sharing”, Proceedings of the IMA International Conference on Cryptography and Coding, Vol.1025, (1995), pp.190-198.
[11] A. K. Dey, D. Salber, G. D. Abowd, M. Futakawa, “Providing Architectural Support for Context-Aware applications”, PhD thesis Georgia Institute of Technology, (2000).
[12] Beimel, Amos and Yuval Ishai, “On the power of nonlinear secret-sharing”, SIAM Journal on Discrete Mathematics, Vol.19, No.1, (2005), pp.258-280.

