From Confidential kNN Queries to Confidential Content-based Publish/Subscribe

Emanuel Onica[1], Hugues Mercier[2] and Etienne Rivière[3]
[1]*Faculty of Computer Science, Alexandru Ioan Cuza University of Iaşi, Romania*
[2]*Institute of Computer Science, University of Neuchâtel, Switzerland*
[3]*ICTEAM, UCLouvain, Belgium*

Keywords: Publish/Subscribe, Data Confidentiality, Security, kNN Queries, Information Dissemination.

Abstract: Content-based publish/subscribe (pub/sub) is an effective paradigm for information dissemination in distributed systems. In brief, publishers generate feeds of information, and subscriber clients register their interests with a pub/sub service tasked with delivering the published data to interested subscribers. Modern pub/sub services are often externalized to public clouds. This brings economic advantages that are unfortunately overshadowed by associated security risks, in particular related to the confidentiality of both the published data as well as of the subscriptions. Guaranteeing confidentiality for content-based pub/sub in an efficient fashion is an active research area. A promising direction is to leverage specific cryptographic solutions that permit the execution of the pub/sub service over encrypted data. In this article we describe a simple and general methodology to derive new mechanisms for pub/sub confidentiality out of another category of data protection schemes: confidential kNN query mechanisms designed for encrypted databases. We exemplify this framework with a concrete use case. We believe that this initial step will lead to more secure and efficient adaptations of kNN solutions to the pub/sub domain.

1 INTRODUCTION

Publish/subscribe (pub/sub) is a paradigm for information dissemination that is particularly fit for large-scale distributed systems and service-oriented architectures. A common pub/sub service deployment consists in an overlay of brokers tasked with matching subscriptions registered by clients of the service (*subscribers*) with publications emitted by providers of information (*publishers*). Two major models exist for the pub/sub paradigm in respect to the structuring of information included in publications and subscriptions and their matching: topic-based and content-based. In topic-based pub/sub each subscription and publication is associated with a topic of interest and the match is decided according to these topics. In content-based pub/sub the publication includes a header composed of key-value attributes representative for the publication content. Subscriptions are formed as conjunctions of constraints used to match some or all of these attributes. An often met example in literature (Eugster et al., 2003; Yang, 2010) for content-based pub/sub is of a stock market scenario, where a publication corresponding to a stock quote includes attributes such as `(symbol='AMD', value=27.85, variation=0.03)`, and a subscription could be formed as `(symbol='AMD' and value < 28)`. In our work we focus on content-based pub/sub, the more expressive model, where a subscriber has more flexibility in indicating interests.

The applications of pub/sub services have a wide range, from stock market transactions (Bernstein and Newcomer, 2009) to management of electronic medical records (Narus et al., 2018). The brokers tasked with matching publications and subscriptions in such services are often externalized to public cloud infrastructures, via virtual machine instances. This brings economic benefits but also introduces security challenges. The chance that a malicious virtual machine will be co-located with a victim instance on Amazon EC2, Google Compute Engine and Microsoft Azure cloud infrastructures were estimated as ranging from 30% to 100%, depending on the power of the attacker and the number of victim instances (Varadarajan et al., 2015). Leaks of information on co-located virtual machines (Ristenpart et al., 2009) have been documented for more than 10 years and are still under scrutiny for finding appropriate protection measures (Han et al., 2017).

Specific to pub/sub is the fact that subscriptions must be stored by untrusted brokers in order to be matched with publications. In many use cases as ref-

Onica, E., Mercier, H. and RiviÃlre, E.
From Confidential kNN Queries to Confidential Content-based Publish/Subscribe.
DOI: 10.5220/0007950506770682
In *Proceedings of the 14th International Conference on Software Technologies (ICSOFT 2019)*, pages 677-682
ISBN: 978-989-758-379-7

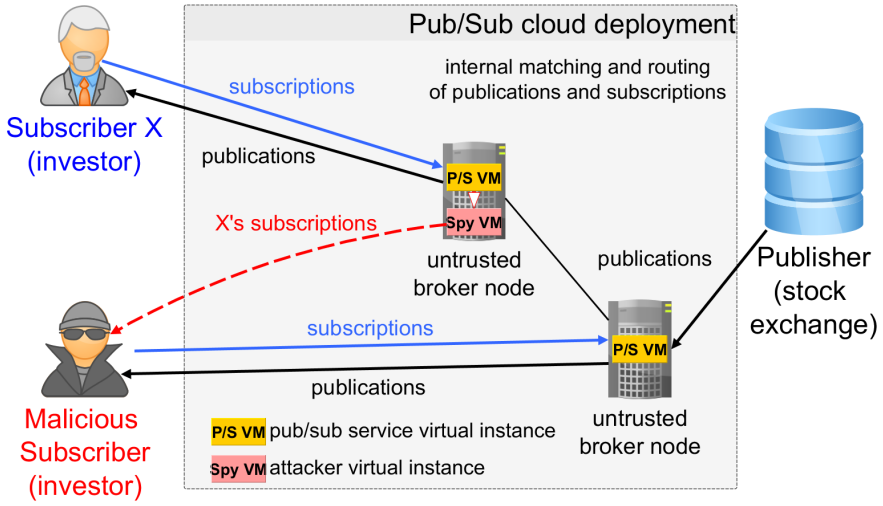

Figure 1: High-level overview of subscription leaks in a cloud deployed pub/sub service.

erenced before, both subscriptions and publications may convey sensitive information about the users' interests. Figure 1 displays a high-level overview of an attack scenario where a malicious investor spies on the subscriptions of another investor, which could gain him unfair advantage on the stock market. A similar example could be drawn for the case of publications that include private medical records of patients. It is, therefore, necessary to protect the confidentiality of both publications and subscriptions.

Unfortunately, "classical" end-to-end encryption (e.g., AES) does not allow brokers to perform the matching operation. Latest technology advances in mainstream processors, such as Intel SGX or AMD SEV (Mofrad et al., 2018) offer a possible context for performing the matching in trusted hardware (Pires et al., 2016). However, the availability of these mechanisms depends on the infrastructure offered by the cloud providers. Also, such hardware protection comes with limitations on memory size and vulnerability to side-channel attacks as the recent Foreshadow exploit (Van Bulck et al., 2018). Designing a solution purely based on particular cryptographic primitives that permit the matching remains, therefore, a valid option for offering data protection. Several attempts in this area exist (Onica et al., 2016), but current lack of adoption in practice is a proof that this is still an open research direction.

We observe that matching between encrypted subscriptions and publications bears similarities with querying mechanisms used in encrypted databases, in particular for implementing k-Nearest-Neighbor queries (kNN for short). This paper presents our initial work towards a general methodology for transforming kNN encrypted query schemes to solutions for confidentiality-preserving pub/sub.

We start in Section 2 by presenting a generic methodology for adapting any confidentiality-preserving kNN scheme to the pub/sub context. We follow in Section 3 with a concrete example of an adaptation that complies with the defined generic methodology. We discuss aspects related to security and performance in Section 4, and conclude in Section 5.

2 FROM KNN TO CONTENT-BASED PUB/SUB

We first present the principles of kNN queries and their similarities to pub/sub. We then identify the necessary steps for adapting a plaintext kNN query over a database to a plaintext pub/sub context. We finally discuss the integration of encryption in the defined methodology.

2.1 Similarities of kNN and Pub/Sub

In a database, a kNN query returns the k nearest neighbors to a query point, according to some distance function (e.g., the Euclidean distance). Therefore, the query is represented as a set of values over a set of attributes (dimensions) and the reply is a set of k entries from the database using the same schema. An encryption scheme for kNN queries must both preserve the confidentiality of the database records and of the query point while still allowing to execute the query.

In a content-based pub/sub system, subscription constraints are expressed as ranges over some or all

of the attributes (dimensions), also respecting a predefined publication schema. Subscriptions are stored permanently at the brokers. Publications are points in the attribute space, i.e., they define values for all attributes. The result of the matching operation is a decision on whether or not all the range constraints of a subscription are matched by the values of the publication. An encryption scheme for content-based pub/sub has similar objectives as kNN queries: both subscriptions and publications must be encrypted to preserve their confidentiality, while untrusted brokers must remain able to perform the matching.

2.2 From kNN to Pub/Sub in Plaintext

We consider a generic kNN query on a database, which searches for the nearest k records $A_1,...,A_k$ to a query Q, using the Euclidean distance. Both database records and the query can be modeled as points in a n-dimensional space, where each dimension corresponds to a field in the database schema. Subscriptions and publications in a pub/sub system can also be modeled as points in a n-dimensional space. The main difference comes from the fact that a subscription is typically formed as a conjunction of constraints (e.g., $<,>,=$) over the values of individual dimensions.

A kNN query is typically evaluated by comparing the Euclidean distances between the query point and each of the database record points (e.g., $dist(Q,A_1)$ compared with $dist(Q,A_2)$), with the purpose of determining which points are *the closest*. Matching in pub/sub is fundamentally different. Pub/sub matching requires to determine *the exact relation* $(>,<,=)$ between each individual dimension in the subscription point and the corresponding dimension in the publication point. This relation between individual dimensions is not important in a kNN query, while the distance matters. It is the opposite for pub/sub. For instance, in Figure 2, points $A_1 = (1,2)$ and $A_2 = (3,0)$ are equally close to query point $Q = (2,1)$. A kNN query for point Q will not distinguish between points A_1 and A_2. Therefore, in general, the result of a kNN query does not help us determine the exact relation between individual dimensions (x and y coordinates) in Q and the corresponding values in either A_1 or A_2.

However, there is a particular situation where the kNN query result can be used to determine the relation between individual dimensions, which we leverage for content-based pub/sub. Let us consider a dimension of interest d_i for which we want to find the relation $(>,<,=)$ between the corresponding values in the query point and another point. We assume a query point Q and two record points A_1 and A_2 where all corresponding dimensions in A_1 and A_2 are equal, except for the dimension of interest d_i. Furthermore, let us consider the point A_m at the middle of the segment $[A_1A_2]$, and assume that we know the relation $(<,>,=)$ between the value of d_i in points A_1 and A_2. Figure 3 gives an example of this context for bi-dimensional points $A_1 = (1,1)$, $A_2 = (5,1)$, $A_m = (3,1)$ and $Q = (2,2)$, where we consider d_1 as the dimension of interest (d_1 corresponds to the x axis). We want to determine the exact relation between the value of d_1 in point Q and the value of d_1 in the point A_m using Q as kNN query over points A_1 and A_2. In this example the kNN query result for query point Q would clearly show that $dist(Q,A_1) < dist(Q,A_2)$.

Let Q_P be the projection of query Q on the axis determined by A_1 and A_2. It is trivial to prove, under the assumptions of the above context, that if $dist(Q,A_1) < dist(Q,A_2) \Rightarrow dist(Q_P,A_1) < dist(Q_P,A_2)$. More generally, if $dist(Q,A_1) \sim dist(Q,A_2) \Rightarrow dist(Q_P,A_1) \sim dist(Q_P,A_2)$ where $\sim \in \{<,>,=\}$. Since A_1, A_2, A_m and Q_P are on the same axis and have all dimensions equal except of d_1, from $dist(Q_P,A_1) < dist(Q_P,A_2)$ follows that d_1 in Q_P is smaller than d_1 in A_m. In the assumed context where Q_P is the projection of Q on an axis where all dimensions are equal except d_1, we also clearly can infer that d_1 in Q is equal to the value of d_1 in Q_P. Therefore, the relation $(<,>,=)$ we determined between d_1 in Q_P and d_1 in A_m is always preserved when comparing d_1 in Q with d_1 in A_m. This leads to our searched result: d_1 in Q is smaller than d_1 in A_m.

2.3 From Encrypted kNN to Encrypted Pub/Sub

We can now define a generic approach for adapting any confidential kNN query to a confidential pub/sub solution. In the pub/sub scenario a subscription S will be represented using the format of records A_1 and A_2 considered in the example above, and a publication P will take the representation of the kNN query Q. More precisely, for each dimension of interest d_i in a subscription, a subscriber will consider two points S_{i1} and S_{i2}, such that d_i is the *middle* of the segment $[S_{i1}S_{i2}]$. All other dimensions in S_{i1} and S_{i2} can be chosen randomly, but must be the same for the two points. Note that this does not restrict the subscriber: any value can be used for any dimension of interest d_i. A publication P will simply resemble the query Q.

Cryptographic schemes for confidentiality-preserving kNN queries typically define a type of encryption for both query and records, such that the query result can be obtained over the encrypted form without leaking information about the values in the dimensions. We distinguish two variants in the

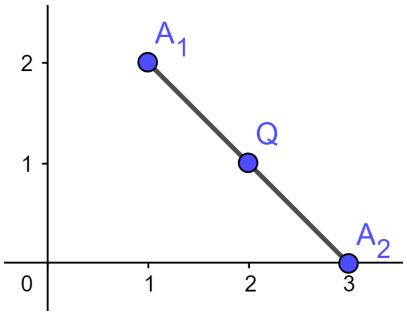

Figure 2: Distance comparison between points in a kNN query.

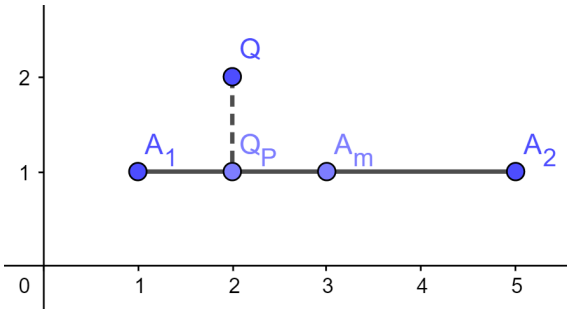

Figure 3: Distance comparison between points in the pub/sub case.

operation of the confidential kNN query:

1. The scheme provides or exposes the ordering of the k records based on their distance from the query;
2. The scheme only provides the final result of the k closest records to the query point, without leaking anything else.

In the first variant it is enough to represent subscriptions as records, as defined above, encrypt them, proceed similarly with encrypting publications in the same way the queries are encrypted, and execute the confidential kNN query with P over all points in pairs (S_{i1}, S_{i2}) up to when we can extract the comparison result between the distances. Then, from this result: $dist(S_{i1}, P) \sim dist(S_{i2}, P)$ (where $\sim \in \{<, >, =\}$), following the reasoning in Section 2.2, we obtain the relation between the value of d_i in P and of d_i in S, which allows us to decide on the pub/sub match for dimension d_i. This is similar to our example, where we obtained the relation between dimension d_1 in Q and d_1 in A_m.

In the second variant, we can execute a 1NN confidential query with the encrypted P over each pair of encrypted points (S_{i1}, S_{i2}) sequentially. This will obviously show which of S_{i1} or S_{i2} is closer to P and implicitly provide the needed distance comparison result $dist(S_{i1}, P) \sim dist(S_{i2}, P)$ for proceeding as above to determine the pub/sub match.

We note that the second variant can be applied in any situation. Defining the first variant is merely for efficiency purposes: if the ordering is exposed, i.e., the scheme mechanism permits comparing distances between encrypted points, we can leverage it. Furthermore, the structure of the encrypted subscriptions and how they are stored might vary (e.g., if the scheme permits evaluating subscription coverage, encrypted subscriptions could be organized in containment trees (Barazzutti et al., 2017)). Executing 1NN queries sequentially over each pair of encrypted points might, therefore, require reorganizing their storage. In such a case, if a scheme exposes the ordering in its default query run, that scheme might be more efficient to use than a scheme that does not.

3 APPLICATION EXAMPLE

The first and only application of a confidentiality preserving kNN query to pub/sub was presented by (Choi et al., 2010), using asymmetric scalar-product-preserving encryption (ASPE), a scheme introduced earlier by (Wong et al., 2009). The adaptation follows a path similar to the methodology defined in Section 2, but considers mostly the case of one single dimension/attribute for publications and subscriptions schemas. Also, it does not include any generalization or analysis of the adaptation procedure. We addressed the multidimensional case and hardened the security of the original scheme, offering proofs of security in (Onica et al., 2015). However, the main focus of our previous work was on performing key updates in a secure pub/sub context, and a generalization that could be applied for adapting other cryptographic schemes was not defined. Nevertheless, ASPE can serve as a valid proof for the generic adaptation methodology we are proposing, and we summarize it in the following.

For each dimension d_i in a subscription, the subscriber prepares two points S_{i1} and S_{i2} as described in Section 2. These points are encrypted as: $S'_{i1} = M^T(S_{i1}, -0.5\|S_{i1}\|)^T$ and $S'_{i2} = M^T(S_{i2}, -0.5\|S_{i2}\|)^T$ where $\|S_{i1}\|$ and $\|S_{i2}\|$ represent the Euclidean norm of the two points, and M is an invertible non-orthogonal matrix representing the encryption key. This is the same method of encryption used for database records in the original confidential kNN scheme (Wong et al., 2009). The publisher encrypts the publication as: $P' = M^{-1}q(P, 1)^T$, where q is a random positive obfuscation factor. Again, this is the same method of encryption used for the kNN query point (Wong et al., 2009). Finally, an un-

trusted broker evaluates the result of $(S'_{i2} - S'_{i1})P' = q0.5(dist(S_{i1},P)^2 - dist(S_{i2},P)^2)$. Simply comparing this result with 0 exposes the relation $dist(S_{i1},P) \sim dist(S_{i2},P)$ (where $\sim \in \{<,>,=\}$), which places the scheme in the first variant identified in Section 2, and enables the decision on pub/sub matching.

As described earlier, the construction of the encryption mechanism is orthogonal to the steps in our methodology for matching encrypted subscriptions against encrypted publications. The adaptation just requires the particular representation of the subscription points before the encryption, and determining the distance comparison result following the execution of the query scheme.

4 DISCUSSION

Although our adaptation methodology could theoretically apply to any confidentiality preserving kNN query scheme, several aspects related to security and performance deserve a more thorough discussion. We focus on these details in the following.

4.1 Security Aspects

First of all, the security of the obtained confidential pub/sub solution simply relies on the fact that the original encryption scheme is secure. Our adaptation does not influence the encryption scheme in any manner. Essentially, the only particularity is that it considers two specific subscription points for each dimension of interest, but these subscription points are first generated in plaintext and encrypted as any other record point in the original scheme. As long as the original encryption scheme is proven secure this guarantee holds for any encrypted point. The property that allows adapting a confidential kNN query scheme for pub/sub matching is *preserving the distance comparison results* in the encrypted form with respect to the plaintext context. As determined from variants 1 and 2 in Section 2, this preservation is normally valid. We note that preserving the distance comparison results is a weaker property than *preserving the actual distance values* after encryption. The latter is not desirable, resulting in confidentiality leaks, as discussed by (Wong et al., 2009).

4.2 Performance Aspects

Evaluating the pub/sub matching can increase up to a quadratic complexity compared to the plaintext matching, due to the subscription points representation. For each dimension of interest d_i in an original subscription, the two corresponding points that are created S_{i1} and S_{i2} as required by our adaptation, must include as the other random chosen dimensions the complete set of attributes in a publication. If we consider n such fields, this might lead to n operations per attribute of interest in the evaluation of the matching, and consequently to n^2 operations if we have a complete subscription. This is also multiplied with factor 2 per subscription, since for each attribute we have two corresponding points that are created. However, in most practical settings, especially for a high number n of attributes, it is unlikely that most subscriptions will set constraints on each field of a publication. Also, this potential increase in complexity is highly dependent on the actual encryption scheme and the means it uses to provide the distance comparison result, which can be subject of various optimizations. For instance, in the example case of ASPE, the two corresponding points created for each field of a subscription can be aggregated into one point in the encryption phase, therefore removing the factor 2 above. We observed this in previous work (Onica et al., 2015), and noted that it is actually desirable for stronger security reasons. Also, a scheme might also permit evaluating if an encrypted subscription covers another (i.e., the case where a publication matching the covering subscription will match all covered subscriptions), which can lead to consistent improvement, by minimizing the number of evaluated subscriptions (Barazzutti et al., 2017). However, coverage support can also lead to leaks of information on the subscription domain (Raiciu and Rosenblum, 2006).

5 CONCLUSION

We presented our current work in devising a generic methodology for deriving confidentiality preserving pub/sub schemes based on solutions for confidentiality preservation in kNN queries. We explained the reasoning behind the methodology and the series of simple steps needed for such an adaptation. We also exemplified how these were already applied for ASPE, a scheme initially designed for kNN queries and later transposed to pub/sub. The main advantage of our proposed methodology is that it is orthogonal to the specificities of the encryption scheme. Therefore, we believe that it could be applied to other schemes dedicated for preserving kNN confidentiality. In particular there are many solutions derived from ASPE that could potentially be adapted to the pub/sub realm (Tzouramanis, 2017; Tzouramanis and Manolopoulos, 2018; Zhu et al., 2016; Zhu et al.,

2013), as well as completely different solutions such as RASP (Xu et al., 2014). Our future work is to adapt some of these schemes and observe their performance versus the security they provide, using the proposed methodology. We believe that our work can be an important first step for positive outcomes in the research and exploration of such new solutions for pub/sub confidentiality.

REFERENCES

Barazzutti, R., Felber, P., Mercier, H., Onica, E., and Riviere, E. (2017). Efficient and confidentiality-preserving content-based publish/subscribe with prefiltering. *IEEE Transactions on Dependable and Secure Computing*, 14(3).

Bernstein, P. A. and Newcomer, E. (2009). *Principles of transaction processing*. Elsevier/Morgan Kaufmann Publishers, Amsterdam, Netherlands; Boston, USA.

Choi, S., Ghinita, G., and Bertino, E. (2010). A privacy-enhancing content-based publish/subscribe system using scalar product preserving transformations. In *Proceedings of the 21st Database and Expert Systems Applications*, DEXA 2010.

Eugster, P. T., Felber, P., Guerraoui, R., and Kermarrec, A.-M. (2003). The many faces of publish/subscribe. *ACM Computing Surveys*, 35(2).

Han, Y., Chan, J., Alpcan, T., and Leckie, C. (2017). Using virtual machine allocation policies to defend against co-resident attacks in cloud computing. *IEEE Transactions on Dependable and Secure Computing*, 14(1).

Mofrad, S., Zhang, F., Lu, S., and Shi, W. (2018). A comparison study of Intel SGX and AMD memory encryption technology. In *Proceedings of the 7th International Workshop on Hardware and Architectural Support for Security and Privacy*, HASP 2018.

Narus, S. P., Rahman, N., Mann, D. K., He, S., and Haug, P. J. (2018). Enhancing a commercial EMR with an open, standards-based publish-subscribe infrastructure. *AMIA Annual Symposium Proceedings*, 2018.

Onica, E., Felber, P., Mercier, H., and Rivière, E. (2015). Efficient key updates through subscription re-encryption for privacy-preserving publish/subscribe. In *Proceedings of the 16th ACM/IFIP Annual Middleware Conference*, Middleware 2015.

Onica, E., Felber, P., Mercier, H., and Rivière, E. (2016). Confidentiality-preserving publish/subscribe: A survey. *ACM Computing Surveys*, 49(2).

Pires, R., Pasin, M., Felber, P., and Fetzer, C. (2016). Secure content-based routing using Intel software guard extensions. In *Proceedings of the 17th ACM/IFIP International Middleware Conference*, Middleware 2016.

Raiciu, C. and Rosenblum, D. S. (2006). Enabling confidentiality in content-based publish/subscribe infrastructures. In *Proceedings of the Second IEEE/CreatNet International Conference on Security and Privacy in Communication Networks*, Securecomm 2006.

Ristenpart, T., Tromer, E., Shacham, H., and Savage, S. (2009). Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds. In *Proceedings of the 16th ACM Conference on Computer and Communications Security*, CCS 2009.

Tzouramanis, T. (2017). Secure range query processing over untrustworthy cloud services. In *Proceedings of the 21st International Database Engineering & Applications Symposium*, IDEAS 2017.

Tzouramanis, T. and Manolopoulos, Y. (2018). Secure reverse k-nearest neighbours search over encrypted multi-dimensional databases. In *Proceedings of the 22nd International Database Engineering & Applications Symposium*, IDEAS 2018.

Van Bulck, J., Minkin, M., Weisse, O., Genkin, D., Kasikci, B., Piessens, F., Silberstein, M., Wenisch, T. F., Yarom, Y., and Strackx, R. (2018). Foreshadow: Extracting the keys to the Intel SGX kingdom with transient out-of-order execution. In *Proceedings of the 27th USENIX Security Symposium*.

Varadarajan, V., Zhang, Y., Ristenpart, T., and Swift, M. (2015). A placement vulnerability study in multi-tenant public clouds. In *Proceedings of the 24th USENIX Security Symposium*.

Wong, W. K., Cheung, D. W.-l., Kao, B., and Mamoulis, N. (2009). Secure kNN computation on encrypted databases. In *Proceedings of the 35th ACM SIGMOD International Conference on Management of Data*, SIGMOD 2009.

Xu, H., Guo, S., and Chen, K. (2014). Building confidential and efficient query services in the cloud with RASP data perturbation. *IEEE Transactions on Knowledge and Data Engineering*, 26(2).

Yang, L. T. (2010). *Research in Mobile Intelligence: Mobile Computing and Computational Intelligence*. John Wiley & Sons, Hoboken, USA.

Zhu, Y., Huang, Z., and Takagi, T. (2016). Secure and controllable kNN query over encrypted cloud data with key confidentiality. *Journal of Parallel and Distributed Computing*, 89.

Zhu, Y., Xu, R., and Takagi, T. (2013). Secure kNN query on encrypted cloud database without key-sharing. *International Journal of Electronic Security and Digital Forensics*, 5(3/4).