International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 3 (2011) pp. 228-235

Context Protecting Privacy Preservation in Ubiquitous Computing

Arijit Ukil

Innovation Labs, Tata Consultancy Services
Kolkata, India
arijit.ukil@tcs.com

***Abstract*: Ubiquitous computing attempts to provide services in an invisible way with least user intervention. In ubiquitous computing domain context awareness is an important issue. So, mere protection of message confidentiality is not sufficient for most of the applications where context-awareness can lead to near deterministic ideas. An adversary might deduce sensitive information by observing the contextual data, which when correlated with prior information about the people and the physical locations that are being monitored by a set of sensors can reveal most of the sensitive information. So, it is obvious that for security and privacy preservation in ubiquitous computing context protection is of equal importance. In this paper, we propose a scheme which provides two layer privacy protection of user's or application's context data. Our proposed context protecting privacy preservation scheme focuses on protecting spatial and temporal contextual information. We consider the communication part of ubiquitous computing consists of tiny sensor nodes forming Wireless Sensor Networks (WSNs). Through simulation we show the efficacy of our scheme. We also demonstrate the capability of our scheme to overcome the constraints of WSNs.**

***Keywords*:** privacy preservation, ubiquity, security, context-awareness, wireless sensor networks.

I. Introduction

Ubiquitous computing enhances computer use by making many computers available throughout the physical environment, while making them effectively invisible to the user. This requires functioning of multitude of devices in the environment to be oblivious to the users. Mark Weiser in his paper "The Computer for the Twenty-First Century" defines ubiquitous computing as a technology that "weave itself into the fabric of everyday life until it is indistinguishable from the it" [1]. This point of view leads to the notion that almost everything in the fabric of ubiquity leads to context-awareness. Context-awareness evolved from the idea to express the ability of systems and components to respond to the situation of the user that interacted with this system. Research in context-aware computing has made numerous attempts to model not only human attributes and behavior but also how we relate to our environment [2]. Ubiquitous applications requires continuous monitoring, gathers vast amounts of sensitive electronic information about the users, which is the basis of finding opportunities for data interception, theft and surveillance. For example, the disclosure of both spatial and temporal data through traffic analysis, may allow tracking the relative or actual information through correlation with prior knowledge. With so much retrievable personal information available through Internet, it is now becoming difficult and challenging to protect privacy. Good amount of research effort has been gone into the research of privacy preservation, particularly in ubiquitous domain, where distributed computing poses a great threat [18]. In [3], it is observed that ubiquitous computing environments require security and privacy architecture based on trust rather than just user authentication and access control. Burnside et al. [4] described a resource discovery and communication system designed for security and privacy. Privacy preserving schemes and protocols are designed to preserve privacy even in the presence of adversarial participants that attempt to gather information about the inputs of their peers and mostly with malicious intention. There are two major classes of privacy preservation schemes are applied. In additive perturbation, randomized noise is added to the data values. The overall data distributions can be recovered from the randomized values. Another is multiplicative perturbation, where the random projection or random rotation techniques are used in order to perturb the values. In tune of their argument [20], we apply the second technique of masking the private data by some random numbers to form additive perturbation. Along with privacy preservation, another important thing needs to be considered is the data aggregation requirement. Data aggregation [21] is an efficient mechanism in query processing in which data are processed and aggregated within the network. In-network processing is forwarding the raw data from the sender/sink nodes in a processed form, by reducing redundancy or by extracting information out of the data. In [22], the authors reviewed privacy-preserving techniques for protecting two types of private information: data-oriented and context-oriented. The proposed scheme utilizes data-oriented privacy preservation concept for data hiding and context-oriented privacy preservation concept to nullify the effect of context-awareness in ubiquitous computing.

So, we can observe that along with user privacy protection by hiding users' identity or location, we require to protect data privacy. In a privacy-preserving data aggregation (PPDA)

protocol, sensor data are partially exposed to neighboring trusted sensor nodes so that data aggregation can be achieved on the way to the source node without revealing the actual data to the trusted sensor nodes or adversaries [5-7]. In his famous paper [8], Yao has introduces the millionaire problem, which can be summarized as: A and B are two millionaires who want to find out who is richer without revealing the precise amount of their wealth. The objective of privacy preserving data mining is to meet the required privacy requirements and to provide data mining outcome [9]. There are number of research proposals and algorithms exist for solving the stated problem. Privacy preserving schemes and protocols are designed to preserve privacy even in the presence of adversarial participants that attempt to gather information about the inputs of their peers and mostly with malicious intention [16]. In [10], the problem of privacy preservation in a peer-to-peer network application is addressed. In [11], Zhang et al. [14] proposed the Perturbed Histogram-based Aggregation (PHA) to preserve privacy for queries targeted at special sensor data or sensor data distribution. The perturbation technique is applied to hide the actual individual readings and the actual aggregate results sent by sensor nodes. For this, every sensor node is preloaded with a unique secret number which is known exclusively by the sink and the node itself. Sensor nodes and the sink form a tree. Wenbo He et.al. propose schemes to achieve data aggregation while preserving privacy. The scheme they proposed, CPDA (Cluster-based Private Data Aggregation) performs privacy-preserving data aggregation in low communication overhead with high computational overhead. In CPDA, each cluster leverages the additive property of polynomials to calculate the desired aggregate value.

From this background, we propose our context protecting privacy preservation scheme. This scheme has two layers. In first layer, the contextual information derived from spatial and temporal domain identity of the user is protected. In second layer, the actual contextual data privacy protection is made using the concept of PPDA.

The paper is organized as follows. In Section 2, we illustrate the system architecture. In Section 3, we present first layer of our scheme, where location and timing data privacy is protected. In Section 4, we describe the privacy preservation method of the contextual data itself by using PPDA. In Section 5, we analyze and compare the computational time in our proposed scheme and in that is proposed in CPDA [11]. In section VI, we present the simulation result and analysis. Lastly, we conclude the paper in Section 7 with conclusion and future scope of work.

II. System Model

In this section, we illustrate the system model, based on which our context protecting privacy preservation scheme is based. We consider N number of sensor nodes is present in the distributed network, which is a part of overall ubiquitous computing system. We take Home Gateway (HG) as the central server which is connected to the Internet. The sensor nodes are bi-directional and they have single-hop (direct) or multi-hop link with the HG in order to maintain connectivity with the outside world. For illustration purpose we have taken N= 8. This is shown in Fig.1.

It is observed from the model that few nodes have direct or single hop communication with the HG, like node 1 and 2. Whereas, in other nodes, multi-hop communication is required to reach HG. So, node 1 and 2 are very critical from security perspective. If these nodes are compromised, the overall network may collapse. These nodes are to be made extremely secured. However, private information from node 1 and 2 can reach HG directly without considering attacks at the routing.

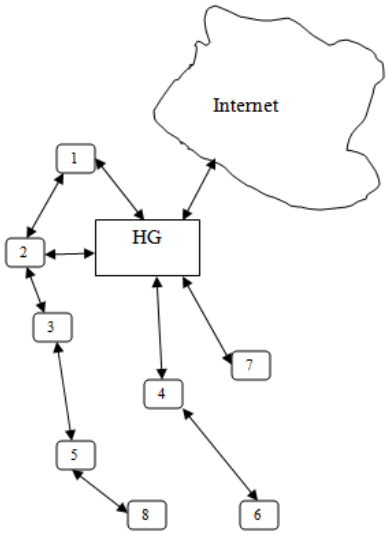

Figure 1. HG based ubiquitous computing architecture

The objective of our scheme is that the contextual information regarding any of the sensor nodes $(s_i, i \in N)$ need to be protected. These information are (s_{is}, s_{it}), which are for spatial and temporal identity of the nodes. Apart from that it is required that even in the case these data are revealed to an attacker, he/she cannot make out the content of the data. This is in fact, a doubly locked privacy preservation scheme, where both the locks (schemes) are independent. From functional point of view, contextual data has two parts. First part, it is being made anonymous so that even in the case actual data is revealed, source or destination turns out to be vague. The second part consists of preserving the privacy of the actual data itself by data perturbation technique so that attacker gets confused. This is shown in Fig. 2.

Our proposed scheme enables users to control their personal data. If the user does not require concealing its contextual information, then the system bypasses the scheme and directly delivers the information in traditional way. But in the case, the user wants its contextual data to be privacy protected; the user needs to inform the system about the amount of protection it requires. If only anonymity is sufficient, then second layer is ignored. If data privacy preservation or data perturbation based PPDA is required then first layer is bypassed. If the user wants absolute privacy protection, then both the layers are used for its contextual data protection. This algorithm is shown in Fig. 3.

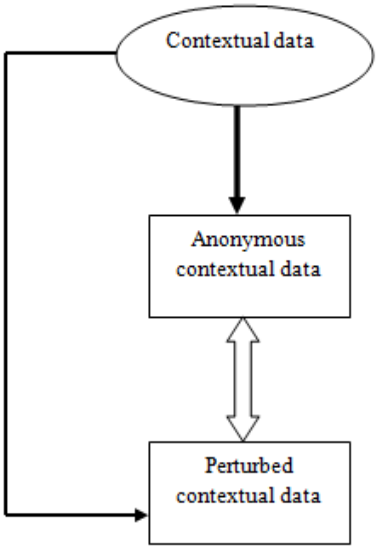

Figure 2. Functional model

III. Anonymity of Contextual Information

In this section, we propose our scheme of contextual data privacy protection by anonymity. In this case, we consider only spatial and temporal privacy. We would like to remark that the classification of context-oriented privacy into the two categories of spatial and temporal privacy only reflects the current state-of-the art, and should not be treated as a comprehensive classification. In order to illustrate this, we present our scheme for location privacy, which can be extended to temporal privacy by slight modification.

In order to conceal the location or spatial information of a node, we need to protect the poor privacy protection performance of conventional routing protocols. To achieve our objective of location privacy protection, we consider the phantom routing strategy [12]. In this scheme, the source location privacy is protected through directing the periodic messages from the source node towards different paths in the network. This prohibits the malicious nodes or the attacker from receiving a stable stream of messages that would enable back-tracing the source. Instead of that, by the received messages the attacker is led towards phantom sources. This routing strategy is depicted in Fig. 4. In this scheme, there are two phases exist for packet delivery. First one, which originates from the source (S) is a pure or directed random walk for a given number of hops that directs the message to a phantom source or flooding source F away from the original source S. The second phase, which is message flooding phase delivers the message to the destination D. In this case random walk at the first phase leads to different flooding node, which makes tracing back more difficult. If the malicious node M detects a message forwarded by node F and moves to that node to get closer to the source, the next message is unlikely to follow the same random path. This makes M's previous move worthless. For more protection, we can apply greedy random walk approach, which is a two way random walk, performed both from the source and the source. It is inspired by the observation that if M gains a good coverage of the network by distributing a number of observation points around the source, the source location could be approximated because the flooding phase would reveal too much information. In order to avoid this, instead of using flooding to deliver the message to the destination D, the destination node sets up a random walk which serves as the receptor of the messages. Each message is randomly forwarded from a source until it reaches the receptor, and is then forwarded to the destination through the pre-established path. A further advantage that the random walk offers is that the safety period improves as the network size increases, as the paths followed by subsequent messages, and consequently the malicious nodes, become more diverse. The diversity of the paths is not, however, the only issue that the random walk implementation needs to ensure. The main purpose of this phase is to send each message to a phantom source that is far from the original source. The second problem is finding the flooding zone such that it is as minimum (with respect to number nodes message needs to be flooded to) as possible which will make trace back low probabilistic. Finding this optimality condition can be explained by an example.

Let, probability of original source detection be P_r, which is a very small number, $P_r \rightarrow 0$. Typically, $P_r \leq 10^{-2}$. In order to find the destination (assuming that our first phase has been broken and source S, flooding node F are identified) D, the attacker has to try out each of the flooded message from F. Now, if N number of nodes are available in the flooding zone and average H number of hops possible in that zone, then message is broadcast to K number of nodes, where

$$K = \frac{N!}{H!(N-H)!}$$

So, finding S from K possible nodes is of probability: 1/K. So, the optimal condition is:

$$P_r < \frac{1}{K}$$

$$K > \frac{1}{P_r}$$

Now, let us consider it numerically for practical case. Consider,

$$P_r = 10^{-2}$$

$$H = 3$$

N turns out to be = 10 to satisfy the condition: $K > \frac{1}{P_r}$.

This is very nominal. In the case the average hop number is 4, N is approximately 8. So, we observe that, very low probability of trace-back condition can be achieved with few numbers of nodes selected in the flooding zone. The proposed scheme when compared to basic flooding, the energy consumption, which mainly depends on the number of the transmitted messages, is not increased. The message latency,

however, could be significantly increased depending on the length of the random walk. This may not be a problem as most of the current day's privacy preservation applications are non real-time in nature.

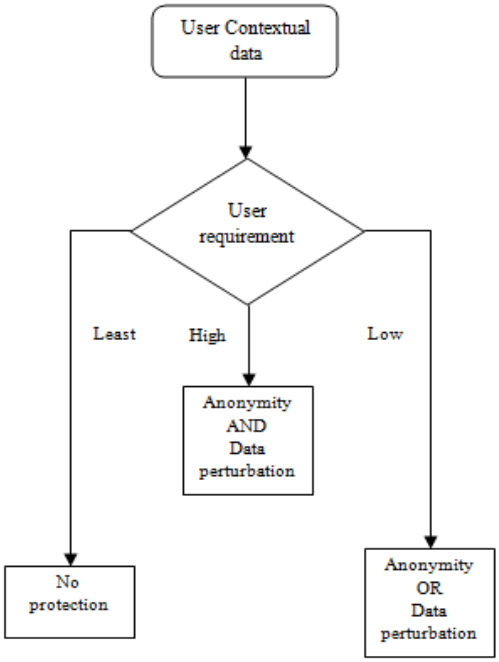

Figure 3. Flowchart

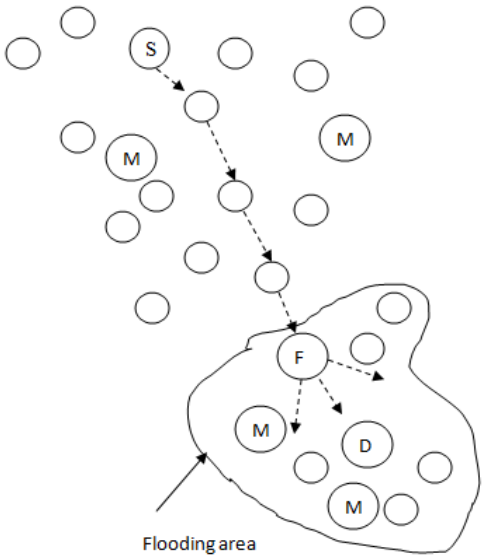

Figure 4. Context anonymity routing

IV. Perturbation of Contextual Information

Good amount of research works have been done to find privacy preserving data aggregation; like using modular arithmetic through secure multi party computation [19]. In this section, we present the scheme for privacy preservation of user data. The objective is similar that of [8], i.e., we consider a scenario where data aggregation needs to be done in privacy-preserved way for distributed computing platform. There are number of data sources which collect or produce data. The data collected or produced by the sources is private and the owner or the source does not like to reveal the content of the data. But the collected data from the source is to be aggregated by an aggregator, which may be a third party or part of the network, where the data sources belong. The data sources do not trust the aggregator. So the data needs to be secure and privacy protected. In tune of that, we propose a scheme which is secure and privacy preserved. The computation for the aggregation is based on the concept of Secure Multiparty Computation (SMC) [13]. In this case, we need to slightly modify the routing process initialization from the source. Instead of one, we consider two sources (*S1* and *S2*) and one Aggregator-Forwarder (AF) node, where the data of the source nodes will aggregate. The AF node aggregates the data of *S1* and *S2* and forwards the aggregated value towards the Flooding node. Aggregation process is governed by data perturbation technique, where the AF cannot find out the exact content of the data of *S1* and *S2*. This is depicted in Fig. 5. Here, we follow the scheme proposed in [7] by Wenbo He et al. The scheme they proposed, CPDA (Cluster-based Private Data Aggregation) performs privacy-preserving data aggregation in low communication overhead with high computational overhead. They have assumed a self-organized multi-hop wireless sensor networks. The scheme CPDA though very much effective, but suffers from two critical limitations:

1. Computation of the privacy preservation algorithm increases exponentially with the number of source nodes. In fact, its computational complexity is $O(N^2)$, where N= number of nodes.
2. In most of the practical scenarios, the source nodes cannot communicate directly with each other or peer-to-peer. In these cases, CPDA is useless.

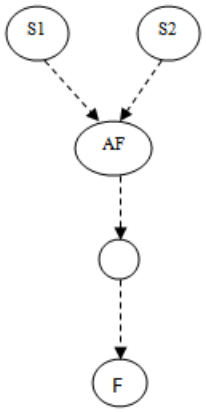

Figure 5. Privacy routing

Our proposed scheme consists of three parts:

1. Key management
2. Data value distortion
3. Data aggregation

A. Key management

In [15], Eschenauer and Gligor proposed one random key pre-distribution for secure key distribution in sensor networks. To provide support for ensuring privacy and integrity of messages sent from sink nodes to their corresponding aggregator or server, robust key exchange and management scheme is required [17]. In this work, the philosophy is to present a key management scheme designed to satisfy operational and security requirements of a non hierarchical, single-hop sensor network by selectively distributing and removing keys from sensor nodes (including sink nodes and server) as well as re-keying nodes without substantial computations or bandwidth usage. The objective of our key management scheme is as follows.

1. The scheme must establish a key between all the sensor nodes that must exchange data securely.
2. Node addition or deletion should be supported.
3. It should work in undefined deployment environment and unauthorized nodes should not be allowed to establish.

In order to accomplish these objectives, we first form cluster of the source nodes. Let, there be *N* number of source nodes and each cluster consists of n number of source nodes. So, there will be N/n number of clusters. The key management process starts by key pre-distribution stage. In the pre-distribution phase, a large key-pool of *K* keys and corresponding identities are generated. This *K* number of keys is divided into two banks. One bank consists of *k* number of keys, which is used for source node's communication with AF. The rest *K-k* number of keys form another bank which are required for communication between *S1* and *S2* via aggregator node. So, the key management scheme consists of two parts:

Source to AF: Each source node has *K-k* number of keys shared with AF. As, all the source nodes possess the same keys, it is totally unsecure when a source node communicates with AF node with the shared key. Any malicious source node can decipher the source nodes' communication with AF and can launch attack very easily. In order to avoid this, in the pre-distribution phase, the source-AF key bank is randomly permuted and reordered for each source-AF pair. This ordering of the key bank is stored in the AF for each source. Now, the source node communicates with AF through one of its shared keys. To accomplish this action, the source node first generates a random number between 1 and *K-k*. This random number (*Rc*) is sent to AF in plain text. AF understands that the source node will encrypt the next message by the *Rc*th number key of the key bank.

Source to Source (S1 to S2): It is assumed that source to source direct communication does not exist and this has to happen securely through AF. In order to achieve that, the k number of keys is stored in the source nodes, which AF is unaware of. It is also a requirement that other source nodes should not decipher the message source 1 sends to source 2. As the k keys are same for all the source nodes, it becomes easy for another source node to decrypt the plain text, i.e. source 3 can decrypt what source node 1 and source node 2 are communicating. To avoid this situation, source node 1 and source node 2 separately permute the key bank order of the k number of keys dedicated for source-source communication and reorder that randomly. After that, they pass the permute function to each through AF using their pair-wise key with the AF.

B. Data value distortion

After establishing the secure communication, we describe the privacy preservation algorithm. This privacy-preservation data aggregation policy is based on the additive property of the polynomial [11]. The objective of this algorithm is that the server or the aggregator cannot make out the individual content of the data sent by the source node. Let the data values at *S1* and *S2* be *x* and *y;* while *z* be the dummy variable at the aggregator (A). In the first step, the aggregator sends three seeds *a, b* and *c* to *S1* and *S2*. Based on that A computes:

$$\alpha_{S1}^{A} = z + R_1^A a + R_2^A a^2$$
$$\alpha_{S2}^{A} = z + R_1^A b + R_2^A b^2$$
$$\alpha_{A}^{A} = z + R_1^A c + R_2^A c^2$$

Where R_1^A and R_2^B are two random numbers generated by A. Similarly, S1 computes

$$\alpha_{S1}^{S1} = x + R_1^{S1} b + R_2^{S1} b^2$$
$$\alpha_{A}^{S1} = x + R_1^{S1} a + R_2^{S1} a^2$$
$$\alpha_{S2}^{S1} = x + R_1^{S1} c + R_2^{S1} c^2$$

Similarly S2 computes:

$$\alpha_{A}^{S2} = y + R_1^{S2} a + R_2^{S2} a^2$$
$$\alpha_{S1}^{S2} = y + R_1^{S2} b + R_2^{S2} b^2$$
$$\alpha_{S2}^{S2} = y + R_1^{S2} c + R_2^{S2} c^2$$

Where R_1^{S1} and R_2^{S1} are two random numbers generated by *S1*, R_1^{S2} and R_2^{S2} are two random numbers generated by *S2*. After that, the calculated, α_{S1}^{A} and α_{S2}^{A} are sent to *S1* and *S2* by *A*, securely as described earlier. Similarly, α_{A}^{S1} and α_{S2}^{S1} are sent to source node 2 and A by *S1* and α_{A}^{S2} and α_{S1}^{S2} are sent to *A* and *S1* by *S2*. We can note that with the addition of random numbers at the nodes, the transmitted values α s random in nature. It is also to be observed that the random numbers generated at each of the nodes (*S1, S2* and *A*) belong to the parent nodes only and they are not shared. So, it can be easily argued that the transmitted values are totally random as well as totally distorted in nature. Finding out the private values (*x, y* or *z*) from the transmitted values is provably impossible.

C. Data value aggregation

After the private data values (*x* and *y*) are distorted, all the nodes aggregate the values available to them and generate aggregated result. *S1* calculates ψ_{S1}, *S2* calculates ψ_{S2} and *A* calculates ψ_A.

$$\psi_A = \alpha_A^A + \alpha_A^{S1} + \alpha_A^{S2}$$
$$= (x+y+z) + R_1 b + R_2 b^2$$
$$\psi_{S1} = \alpha_{S1}^A + \alpha_{S1}^{S1} + \alpha_{S1}^{S2}$$
$$= (x+y+z) + R_1 c + R_2 c^2$$

$$\psi_{S2} = \alpha_{S2}^A + \alpha_{S2}^{S1} + \alpha_{S2}^{S2}$$
$$= (x+y+z) + R_1 a + R_2 a^2$$

Where, $R_1 = R_1^A + R_1^{S1} + R_1^{S2}$

$$R_2 = R_2^A + R_2^{S1} + R_2^{S2}$$

These aggregated results from S1 and S2 are securely sent to the aggregator A. Now, the aggregator has the simple task to solve the above equation for (x+y+z) with the knowledge of the values of a,b,c and ψ_A, ψ_{S1} and ψ_{S2}. After solving for $D = x+y+z$, node A knows its data z, so it can find out the result *(x+y).*

V. Complexity analysis

In this section, we analyze the complexity of our scheme.

A. Cluster Formation

The cluster formation stage has a complexity of *O(NcP)* i.e. pseudo polynomial in order *P*, where *Nc* stands for the probability of a node independently becoming a cluster leader. This is done each time cluster formation is required in order to counter the dynamics of the sensor networks.

B. Broadcasting seeds

Broadcasting of seeds within a cluster takes *O(k)* messages, where k is the cluster size, this is done in each cluster. So overall complexity for this is (number of clusters) x *O(k).*

C. Encrypt and send computed values

Every node within a cluster sends a message to every other node within the cluster and this happens in all the clusters. So, message complexity is *O* (cluster size x pseudo polynomial order), i.e *O(kP).*

D. Aggregate Information

Tthe message complexity is simply the size of cluster in each cluster. So, overall complexity is the order equal to number of nodes in a cluster (*P*) x number of clusters (*K*), which is *O(PK).*

VI. Simulation Results

In this section, we show a comparative study between our proposed scheme and the CPDA scheme in [11]. The objective of our work is to find a simpler, efficient privacy preserved data aggregation scheme, which has scalability and can be highly effective in some practical scenario like discussed in the motivation section. From that perspective, we see that computation time requirement to run SPPDA comes out to be around 1 msec in an Intel Core 2 duo PC with CPU speed 3 GHz and RAM of 2 GB. Where as if we increase the number of source nodes to 3, the overall computation speed becomes 3 msec as shown in fig. 6. As in our scheme, in most cases, there will be fixed two number of source nodes, the computational time becomes fixed. This is indeed a necessary requirement if the overall system is real-time in nature and for resource limited sensor nodes in WSN.

In CPDA scheme, there exists certain probability where private data may be disclosed. This can only happen when the source nodes exchange messages within the cluster. This can be estimated as:

$$P(b) = \sum_{m=pc}^{D_{\max}} P(k=m)(1-(1-b^{m-1})^m)$$

Where *Dmax* = maximum cluster size, *pc* = minimum cluster size (= 3, two source nodes and one aggregator), k = cluster size, *b* = probability that link level privacy is broken, *P(k=m)* = probability that a cluster size is m. In our case, pc = *Dmax* = *k* = 3, *P(k=m) =1*. So, we have plotted *P(b)* for CPDA and our scheme in Fig. 7. It is observed that the probability of privacy compromised in CPDA has much steeper slope.

In CPDA, a requirement is that a pair of source nodes possessing same pair of keys, where the keys are taken randomly from a large pool of key, should be high. Otherwise, the scheme cannot work. But, this requirement helps malicious nodes to capture at least some of the communication, if it has common pair of keys. This probability also increases with number of source nodes increase when total number of keys in the key pool is constant. In our scheme, there is no requirement like that.

In Fig. 8, we have compared the computational time requirement of our scheme to that of the CPDA scheme proposed in [11]. It is to be noted that we have compared only the algorithm performance. As in CPDA, with number of client nodes increases, the computational time increases, we constraint number of source nodes to five. It is also impractical in CPDA to have large number of nodes in a single cluster. The comparison figure reveals the computational efficiency of our algorithm. Our scheme has the additional advantage of the eliminating the complex cluster formation algorithm as in CPDA.

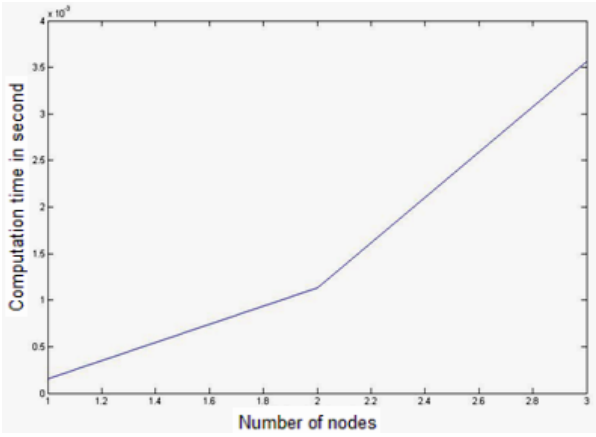

Figure 6. Computation time requirement

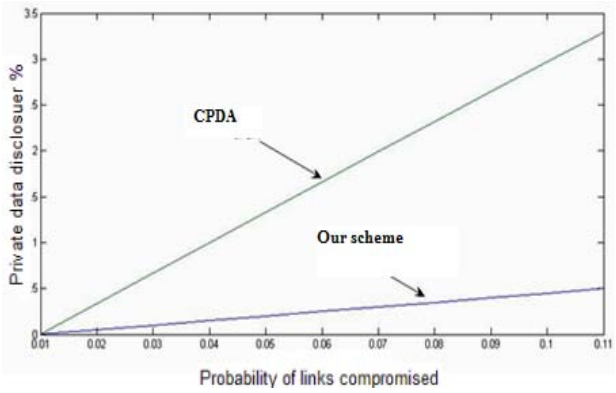

Figure. 7. Probability of private data disclosure

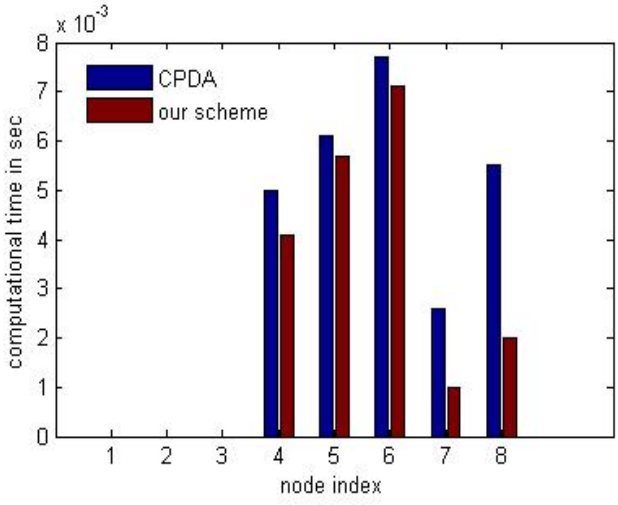

Figure 8. Computational time comparison with [11]

VII. Conclusion

In this paper, we proposed a scheme which aims to protect the privacy of the contextual data mainly in ubiquitous computing environment. It is a two-tier scheme and user has the choice of opting for none, both or any of the tiers as per its security and privacy requirements. With growing number of ubiquitous applications like Home Gateways developed, it becomes very important to conceal one's contextual information, which indirectly can destabilize the security policies. Our scheme, in that sense is a very important development.

Our future work will mainly focus on the complexity analysis and actual implementation in real test-bed. It is to be noted that the proposed scheme consists of three distinct parts and we feel that unit testing of individual parts is important. We also need to evaluate the performance metric of each of the units as well as the overall system. We require to analyze the real-time performance of the system as most of the next generation applications are multimedia in nature with online streaming feature.

References

[1] Mark Weiser, "The Computer for the Twenty First Century," *Scientific American*, pp. 94-104, September, 1991.

[2] R. da Rocha and M. Endler, "Evolutionary and efficient context management in heterogeneous environments," In *Proceedings of the 3rd International Workshop on Middleware for Pervasive and Ad-hoc Computing*, pp. 1-7, 2005.

[3] L. Kagal, T. Finin, and A. Joshi, "Trust-Based Security in Pervasive Computing Environments," *IEEE Computer*, vol. 34, no. 12, pp. 154 – 157, 2001.

[4] M. Burnside, D. Clarke, Mills, A. Maywah, S. Devadas, R. Rivest, "Proxy-Based Security Protocols in Networked Mobile Devices", In *Proceedings of 17th ACM Symp. on Applied Computing* , pp. 265–272, 2002.

[5] J. Yao, G. Wen, "Protecting classification privacy data aggregation in wireless sensor networks," In *Proceedings of the 4th International Conference on Wireless Communication, Networking and Mobile Computing*, WiCOM, Dalian, pp. 1–5, 2008.

[6] W.S. Zhang, C. Wang, and T.M. Feng, "GP2S: Generic privacy-preservation solutions for approximate aggregation of sensor data, concise contribution," In *Proceedings of the 6th Annual IEEE International Conference on Pervasive Computing and Communications*, pp.179–184, 2008.

[7] G. Taban and V.D. Gligor, "Privacy-preserving integrity-assured data aggregation in sensor networks," In *Proceeding of International Symposium on Secure Computing*, pp. 168–175, 2009.

[8] A. Yao, "Protocols for secure computations," In *Proceedings of the 23rd Annual Symposium on Foundations of Computer Science*, pp.160-164, 1982.

[9] S.R.M. Oliveira and O.R. Zaïane, "Achieving Privacy Preservation when Sharing Data for Clustering," *Springer LNCS, 3178*, pp. 67-82, 2004.

[10] Q. Huang, H.J. Wang, and N. Borisov, "Privacy-preserving friends troubleshooting network," In Proceedings of *Symposium on Network and Distributed Systems Security (NDSS)*, pp. 184-194, 2005.

[11] W. He,et al., "PDA: Privacy-preserving Data Aggregation in Wireless Sensor Networks," In *Proceedings of IEEE INFOCOM*, pp. 2045 – 2053, 2007.

[12] C. Ozturk, Y. Zhang, and W. Trappe, "Source-location privacy in energy-constrained sensor network routing," In *Proceedings of 2nd ACM workshop on Security of ad hoc and sensor networks*, pp. 88–93, 2004.

[13] S. Goldwasser, "Multi-party computations: Past and present," In *Proceedings of 16th Annual ACM symposium on Principles of distributed computing*, 1997.

[14] W.S. Zhang, C. Wang, C, and T.M. Feng, "GP2S: Generic privacy-preservation solutions for approximate aggregation of sensor data, concise contribution," In *Proceedings of the 6th Annual IEEE International Conference on Pervasive Computing and Communications, PerCom*, Hong Kong, pp.179–184, March 2008.

[15] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," In *Proceedings*

of 9th ACM Conference on Computer and Communication Security, pp. 41–47, 2002.

[16] A. Ukil, "Security and Privacy in Wireless Sensor Networks," In *Smart Wireless Sensor Networks*, Intechweb, Croatia, pp. 395 - 418, 2010.

[17] J. Sen, "A Distributed Trust and Reputation Framework for Mobile Ad Hoc Networks," First International Workshop on Trust Management in Peer-to-Peer Systems (IWTMP2PS), pp. 538-547, 2010.

[18] B.B. Sarkar and N. Chaki, "Transaction Management for Distributed Database using Petri Nets," In *International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM)*, vol.2, pp. 69- 76, 2010.

[19] A. Ukil and J. Sen, "Secure multiparty privacy preserving data aggregation by modular arithmetic," In Proceedings of *International Conference on Parallel Distributed and Grid Computing (PDGC)*, pp. 344 - 349, 2010.

[20] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar, "Random-data perturbation techniques and privacy-preserving data mining," *Knowledge and Information Systems*, vol. 7, pp. 387—414, 2005.

[21] H. Chan, A. Perrig, B. Przydatek, and D. Song, "SIA: Secure Information Aggregation in Sensor Networks," *J. Com. Sec.*, vol. 5, no. 1, pp. 69-102, 2007.

[22] N. Li, N. Zhang, S. Das, and B. Thuraisingham, "Privacy-preserving in wireless sensor networks: A state-of-the-art survey," *Elsevier Ad Hoc Networks*, vol. 7, pp. 1501 – 1514, 2009.

Author Biography

Arijit Ukil is currently working in Innovation Labs, Tata Consultancy Services Ltd., Kolkata as Research Scientist. He is primarily engaged with the research activity on ubiquitous computing, security and privacy and wireless networking. Before joining Tata Consultancy Services Ltd in 2007, he has worked as Scientist C in Deference Research and Development Organization (DRDO) for more than four years, where his primary research area was digital signal processing, embedded systems, wireless communication for radar applications. He was mainly involved in naval based Radar systems. He has completed his B.Tech in Electronics and Telecommunication Engineering in 2002. He has published a number of conference and journal papers of national and international repute. He has published two book chapters titled "Advanced Scheduling Schemes in 4G Systems" in the book "Fourth-Generation Wireless Networks: Applications and Innovations" published by IGI-Global in Dec, 2009 and "Security and Privacy in Wireless Sensor Networks," in the book "Smart Wireless Sensor Networks," published by Intechweb, Croatia in Dec 2010. He has been reviewer of a number of IEEE conferences like, IEEE VTC, IEEE WCNC and also in Eurasip Journal of Wireless Communications. He has been invited to deliver keynote lectures in many international and national conferences like ETCC'08 in NIT Hamirpur, ICCET'09 in Singapur, NCERDM-IT'09 in Jaipur, NCETAC2010-IT in Trichy. He has delivered one tutorial in ICWMC'10 on "Ubiquitous Computing". He is enlisted in 2010 Marquis' "Who's Who" as a renowned contributor in the field of computer science, communication, and information technology.