Volume 5, Issue 4, 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Survey on Importance of Context aware Security in Distributed Computing Environment like Cloud

Sharmistha Dey
Assistant Professor, Department of Computer Application,
Gurunanak Institute of Technology,
India

Abstract: Most of today's security infrastructure is static – enforcing policies defined in advance in environments where IT infrastructure and business relationships are relatively static. This is no longer sufficient in an environment that is highly dynamic, multi-sourced and virtualized, and where consumer-oriented IT is increasingly used in lieu of enterprise-owned and provisioned. Cloud Computing, being a highly dynamic and altering technology, providing 24x7 support with high storage capacity, reduced cost, potentiality of multitenancy, is suffering from the security infrastructure, for which reason the cloud is being interrupted in getting the maturity it should get. Context-aware security has the potential to make network security a lot smarter. This paper focuses on the importance of context aware security in the cloud computing environment, enlightens the advantage of adapting context aware security in cloud scenario, where the technology and infrastructure itself is dynamic and challenging.

Keywords: Cloud Agent, Personalization, Secondary Context, Vulnerability,

I. INTRODUCTION

1.1 Motivation for the Work:

Cloud computing, being a buzzword in today's highly dynamic technological world, facing some problems with its security concern. There are no publically available standards specific to cloud computing security. As in most of the cases, the security infrastructure is static, unable to properly handle the dynamic technology like cloud. High-level context information is typically obtained from context services that aggregate raw context information sensed by various sensors and mobile devices. Given the massive amount of sensed data, traditional context services are lacking the necessary resources to store and process these data, as well as to disseminate high-level context information to a variety of potential context consumers. One of the benefits of the approach is that context providers can scale up and down, in terms of cloud resources they use, depending on current demand for context information. Besides, the selection algorithm allows ranking context services by matching their QoS and QoC offers against the QoS and QoC requirements of the context consumer. This work makes a study on the effectiveness of context aware security information on the cloud computing models.

1.2 Cloud Computing:

The term "cloud" was coined from the computer network diagrams which uses it to hide the complexity of infrastructure involved. Cloud computing provides software, platform and infrastructure as a service.
According to NIST," *Cloud computing is a model for enabling ubiquitous, convenient, on demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*"[1]
A cloud reduces capital investment, hardware cost and software license cost. Cloud computing also raises severe challenges especially regarding the security level required for the secure use of services provided by it.

1.3 Architecture of Cloud:

From an architectural perspective, there is much confusion surrounding how cloud is both similar to and different from existing models of computing and how these similarities and differences impact the organizational, operational, and technological information security practices [Fig I].
Cloud computing architecture actually refers to the components and subcomponents required for cloud computing. These components typically consist of a front end platform (fat client, thin client, mobile device), back end platforms (servers, storage), a cloud based delivery, and a network (Internet, Intranet, Intercloud). Combined, these components make up cloud computing architecture. The architecture of a cloud computing system is broadly dividing it into two sections: the **front end** and the **back end**. They connect to each other through a network, usually the Internet. The front end is the side the computer user, or client, sees. The back end is the "cloud" section of the system.
The front end includes the client's computer (or computer network) and the application required to access the cloud computing system. Not all cloud computing systems have the same user interface. Services like Web-based e-mail programs leverage existing Web browsers like Internet Explorer or Firefox. Other systems have unique applications that provide network access to clients.

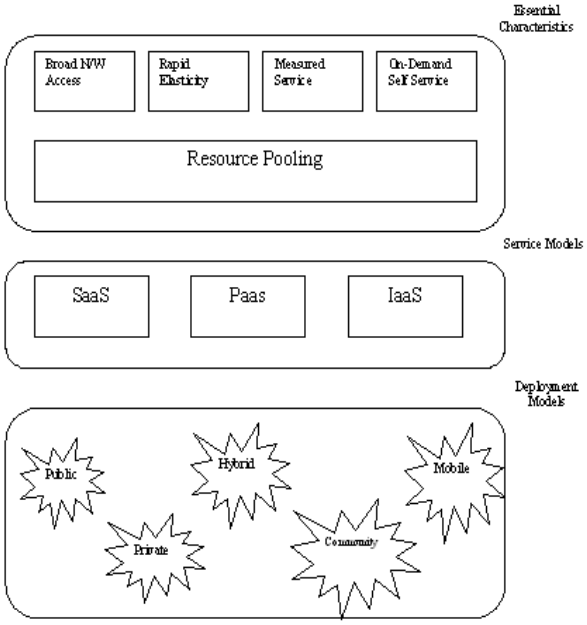

Fig I: Architecture of Cloud Computing

On the back end of the system are the various computers, servers and data storage systems that create the "cloud" of computing services. In theory, a cloud computing system could include practically any computer program you can imagine, from data processing to video games. Usually, each application will have its own dedicated server.

1.4. Types of Clouds:

There are basically three main service model types of clouds:

Public Cloud, Private Cloud and Hybrid Cloud. However, there are some special purpose clouds commercially used in this field, viz., Community cloud and mobile cloud.

1.4.1. Public Cloud : Public cloud applications, storage, and other resources are made available to the general public by a service provider, using a free to all services or a pay per use model [Fig II]. The public cloud allows end users to create services on systems that are hosted and managed outside their firewalls.

Typically public clouds are operated and managed at datacenters belonging to service providers and shared by multiple customers (multi-tenancy). Such a shared model helps reduce vendor costs, which manifests itself in better cloud economics.

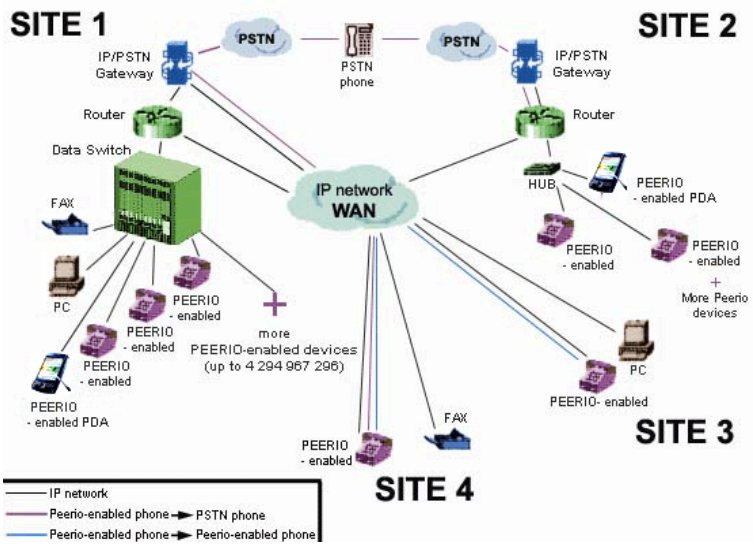

Fig II: Public Cloud

1.4.2. Private Cloud: Private cloud is cloud infrastructure operated solely for a single organization, whether managed internally or by a third-party and hosted internally or externally .It is also known as Internal Cloud or Corporate Cloud. Marketing media that uses the words "private cloud" is designed to appeal to an organization that needs or wants more control over their data than they can get by using a third-party hosted service.

In Fig III, a diagram for private cloud has been shown. Using Virtual Middleware(VM) or Virtual data center, the company can run their private cloud services. Microsoft Azure, IBM, Salesforce etc they run private cloud services.

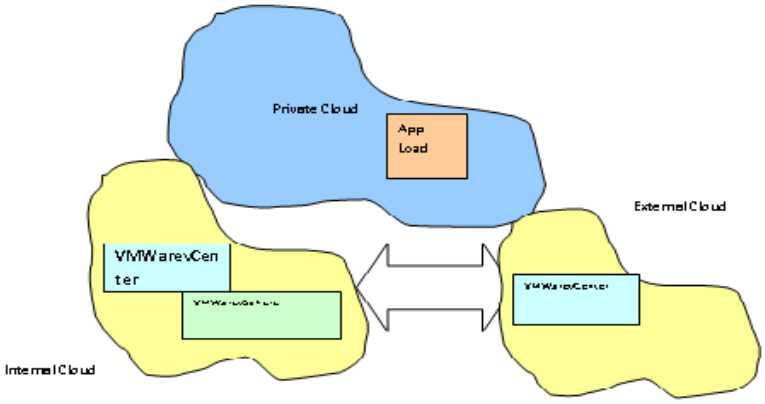

Fig III: Private Cloud

In private cloud, the new concept is Virtual private cloud. A **Virtual Private Cloud (VPC)** is a private cloud existing within a shared or public cloud (i.e. the Intercloud). The VPC is an on demand configurable pool of shared computing resources in a public cloud, isolated between the tenants of the public cloud. The isolation between tenants of a public cloud is performed via access control mechanism. With the introduction of isolation levels the providers multi-tenant architecture is transformed to a single-tenant architecture.

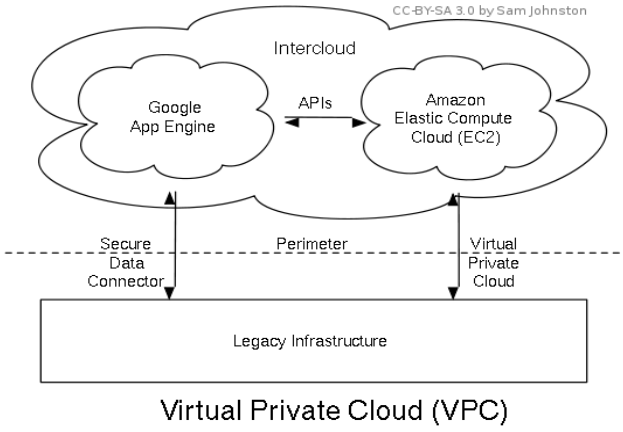

Figure IV: Virtual private cloud

1.4.3. Hybrid cloud: Organizations may host critical applications on private clouds and applications with comparatively less security concerns on the public cloud. In case of hybrid cloud computing, both private and public type of cloud computing is combined together. This is a composition of two or more clouds (private, community or public) that remain unique entities but are bound together, offering the benefits of multiple deployment models.
By utilizing "hybrid cloud"[Fig V] architecture, companies and individuals are able to obtain degrees of fault tolerance combined with locally immediate usability without dependency on internet connectivity. Hybrid cloud architecture requires both on-premises resources and off-site (remote) server-based cloud infrastructure.

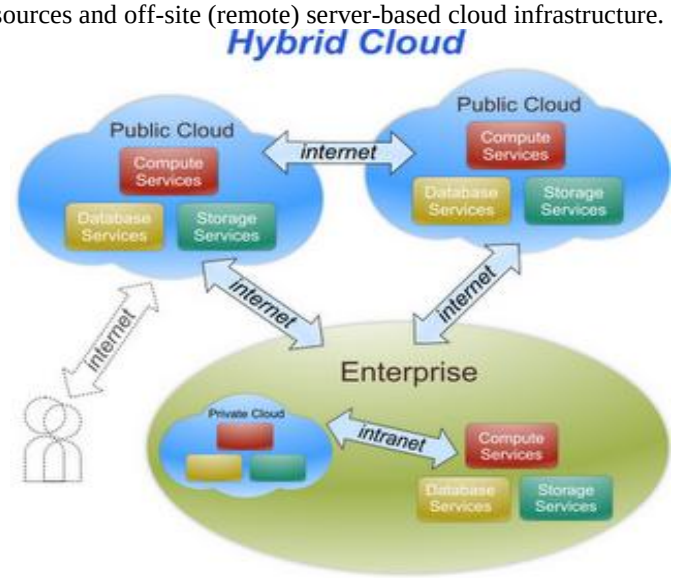

Fig V: Hybrid Cloud

1.4.4. Community Cloud: A community cloud in computing is a collaborative effort in which infrastructure is shared between several organizations from a specific community with common concerns (security, compliance, jurisdiction, etc.), whether managed internally or by a third-party and hosted internally or externally. The costs are spread over fewer users than a public cloud (but more than a private cloud), so only some of the cost savings potential of cloud computing are realized. Community cloud is a special type of service where a group of service providers united together and form a community depending on service type.

The National Institute of Standards and Technology (NIST) describes the community cloud as a cloud infrastructure that is shared by several organizations and supports a specific community, such as healthcare, that has shared concerns around mission, policy and compliance considerations. In case of community cloud, building the high-quality cloud computing infrastructure needed to make this happen requires massive investment in terms of expertise, equipment and support. Hence, what we currently see is a small group of councils building and sharing their own "internal" clouds, painting themselves into a corner with both a limited set of services and the resources needed to support and deliver them.

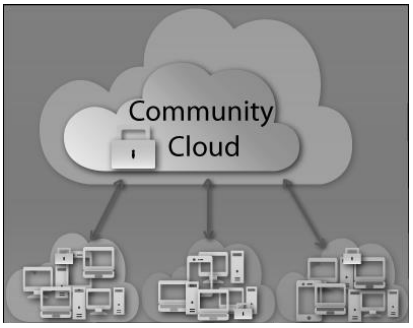

Fig VI : Community cloud

Community clouds are clouds that are tailored to the shared needs of a business community, which in general is a specific industry such as healthcare. Community clouds provide the capability to realize business processes "in the cloud" and at the same time preserve a high security level by means of hybrid deployment models. Non business-critical information and processing can be sourced to the public cloud, while business critical services are kept in-house or in a private cloud environment at a trusted outsourcing partner.

1.4.5. Mobile Cloud:

Mobile cloud computing is the usage of cloud computing in combination with mobile devices. Cloud computing exists when tasks and data are kept on the Internet rather than on individual devices, providing on-demand access. In case of mobile cloud, Applications can run on a remote server and then sent to the user. Because of the advanced improvement in mobile browsers thanks to Apple and Google over the past couple of years, nearly every mobile should have a suitable browser. This means developers will have a much wider market and they can bypass the restrictions created by mobile operating systems. Mobile applications are a rapidly developing segment of the global mobile market. In case of this system; Applications run on a remote server and then sent to the user. Because of the advanced improvement in mobile browsers thanks to Apple, Google, Microsoft and Research in Motion, nearly every mobile should have a suitable browser. This means developers will have a much wider market and they can bypass the restrictions created by mobile operating systems.

II. DEFINITION OF CONTEXT

In order to understand the concept of context-awareness, we need to define what the context is, and how it can be used to protect user's privacy. In this section, first we review the various definitions of context. The term context has been defined by many researchers. Dey et al. [2-5] evaluated and highlighted the weaknesses of these definitions. Dey claimed that the definition provided by Schilit and Theimer was based on examples and cannot be used to identify new context. Further, Dey claimed that definitions provided by Brown ,Franklin and Flachsbart, Rodden et al. [1-4], Hull et al. [2-5], and Ward et al. used synonyms to refer to context, such as environment and situation. Therefore, these definitions also cannot be used to identify new context. Context in a broader sense and provided a definition for context as follows:

"Context is any information that can be used to characterise the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves"

2.1. Context Aware Security:

Short of unplugging our computers from the network and locking them in a room, there is no absolute security. At its most fundamental level, the security is a risk analysis activity in which practitioners decide what they wish to protect from whom and at what cost. The key to understanding these tradeoffs are three properties, which we define to make up a network's security context:

Vulnerability profile: The vulnerability profile represents the space of all possible targets and ideally all methods of unauthorized access to those services. In the traditional sense, this is a mapping between the device (i.e., machine), operating system, applications, and the list of known vulnerabilities for each. More broadly, this encompasses unknown vulnerabilities in server software and the social engineering path for access acquisition in client software.

Attack surface: The attack surface represents the unique threats posed by attackers to the defenders of a particular network. In a traditional sense, it is a measure of the remote network exploits (either attempted or successful) directed at a particular network. In a broader sense, it encompasses the notion of who the attackers are, what resources they are interested in, and the current techniques for acquiring those resources. For example, while a network might run a large number of (potentially vulnerable) printer services, attackers may avoid these services due to their uniqueness (and hence difficulty in exploiting), as well as the limited value in compromising them. Of course, other attackers may feel just the opposite about having access to printed documents – the context matters.

Usage model: While the attack surface helps prioritize the potential targets specified in the vulnerability surface by defining what the attackers are interested in and the current tools used to achieve them, the usage model helps defenders prioritize the importance of the services on the network. This prioritization may be as simple as defining what services are most used on a network, but may layer in notions of data importance, disclosure liability, opportunity costs on failure in availability, etc.

2.2. Context-aware Features

After analysing and comparing the two previous efforts conducted by Schilit et al. [4] and Pascoe [4], Abowd et al. [3] identified three features that a context-aware application can
support: presentation, execution, and tagging. Even though, the IoT vision was not known at the time these features are identified, they are highly applicable to the IoT paradigm as well. We elaborate these features from an IoT perspective.

2.3. Context Types and Categorisation Schemes

Different researchers have identified context types differently based of different perspectives. Abowd et al. [3] introduced one of the leading mechanisms of defining context types. They identified location, identity, time, and activity as the primary context types. Further, they defined secondary context as the context that can be found using primary context. For example, given primary context such as a person's identity, we can acquire many pieces of related information such as phone numbers, addresses, email addresses, etc. However, using this definition we are unable to identify the type of a given context. Let us consider two GPS sensors located in two different locations. We can retrieve GPS values to identify the position of each sensor. However, we can only find the distance between the two sensors by performing calculations based on the raw values generated by the two sensor. The question is, 'what is the category that distance belongs to?' 'is it primary or secondary?' The distance is not just a value that we sensed. We computed the distance by fusing two pieces of context. The above definition does not represent this accurately. Thus, we define a context categorisation scheme (i.e. primary and secondary) that can be used to classify a given data value (e.g. single data item such as current time) of context in terms of an operational perspective (i.e. how the data was acquired). However, the same data value can be considered as primary context in one scenario and secondary context in another. For example, if we collect the blood pressure level of a patient directly from a sensor attached to the patient, it could be identified as primary context. However, if we derive the same information from a patient's health record
by connecting to the hospital database, we call it secondary context. Therefore, the same information can be acquired using different techniques. It is important to understand that the quality, validity, accuracy, cost and effort of acquisition, etc. may varied significantly based on the techniques used. This would be more challenging in the IoT paradigm, because there would be a large amount of data sources that can be used to retrieve the same data value. To decide which source and technique to use would be a difficult task. We will revisit this challenge in Section VI. In addition, a similar type of context information can be classified as both primary and secondary. For example, location can be raw GPS data values or the name of the location (e.g. city, road, restaurant). Therefore, identifying a location as primary context without examining how the data has been collected is fairly inaccurate. Figure 5 depicts how the context can be identified using our context type definitions.

2.3.1 Primary context: Any information retrieved without using existing context and without performing any kind of sensor data fusion operations (e.g. GPS sensor readings as location information).

2.3.2. Secondary context: Any information that can be computed using primary context. The secondary context can be computed by using sensor data fusion operations or data retrieval operations such as web service calls (e.g. identify the distance between two sensors by applying sensor data fusion operations on two raw GPS sensor values). Further, retrieved context such as phone numbers, addresses, email addresses,
birthdays, list of friends from a contact information provider based on a personal identity as the primary context can alsobe identified as secondary context.

2.4. Levels of Context Awareness

Context awareness can be identified in three levels basedon the user interaction.

2.4.1. Personalisation: It allows the users to set their preferences, likes, and expectation to the system manually. For example, users may set the preferred temperature in a smart home environment where the heating system of the home can maintain the specified temperature across all rooms.

2.4.2. Passive context-awareness: The system constantly monitors the environment and offers the appropriate options to the users so they can take actions. For example, when a user enters a super market, the mobile phone alerts the user with a list of discounted products to be considered.

2.4.3. Active context-awareness: The system continuously and autonomously monitors the situation and acts autonomously. For example, if the smoke detectors and temperature sensors detect a fire in a room in a smart home

environment, the system will automatically notify the fire brigade as well as the owner of the house via appropriate methods such as phone calls.

2.5. Context Life Cycle

A data life cycle shows how data moves from phase to phase in software systems (e.g. application, middleware). Specifically, it explains where the data is generated and where the data is consumed. In this section we consider movement of context in context-aware systems. Context-awareness is no longer limited to desktop, web, or mobile applications. It has already become a service: Context-as-a-Service (CXaaS). The data life cycles are into two categories: Enterprise Lifecycle Approaches (ELA) and Context Lifecycle Approaches (CLA).

ELA are focused on context. However, these life cycles are robust and well-established, based on industry standard strategies for data management in general. In contrast, CLA are specialised in context management. However, they are not tested or standardised strategies as much as ELA.

We assume that a context information provider delivers context information to a context-aware system following the life-cycle illustrated in Figure 1. The main steps in a life-cycle of context information are:

□ □ □□ □ □ □ **Discovery of context information**: In this step, a context-aware system discovers available context information providers. The discovery can be performed either in a push or a pull mode.

2.5.2. Acquisition of context information: In this step, a context-aware system collects context information from the discovered context information providers and stores it in a context information repository for further reasoning. Similar to the process of discovery, the acquisition is performed either in a pull or a push mode. In a pull mode, the context-aware system explicitly requests for context information whereas in a push mode, context information providers push context information to the context-aware system.

2.5.3.□ □□Reasoning about context information: reasoning mechanisms enable applications to take advantage of the available context information. The reasoning can be performed based on a single piece of context information or on a collection of such information. For example, in the case of a context-aware e-Health application, user's health status can be evaluated based on both his heart rate and blood pressure provided by medical sensors.

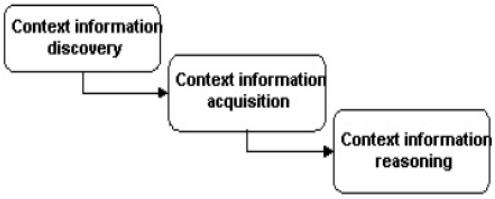

Fig. VII. Context life cycle.

2.6. Context Aware Security in Scenario of Cloud computing

In every business with a delivery/consumption model, brokers emerge to mediate between consumers and providers. This could be the case for context delivery. Context brokers may, then, be used to decouple context consumers from context services. Our interest in using brokers is motivated by the fact that they have been used for a while in Service Oriented Architecture (SOA) to mediate between services providers, service consumers, and partners. They have also been extensively used in multimedia systems and in mobile computing systems to deal mainly with the issue of QoS management.

Fig.VIII depicts our framework for context information provisioning. The main components of the framework are: *Context-aware Web services (context consumers)*, *Context Brokers*, and *Cloud-based Context Services*. Multiple context brokers may be deployed, one for each local domain for instance. A discovery service will allow context-aware consumers to bind to the right context broker.

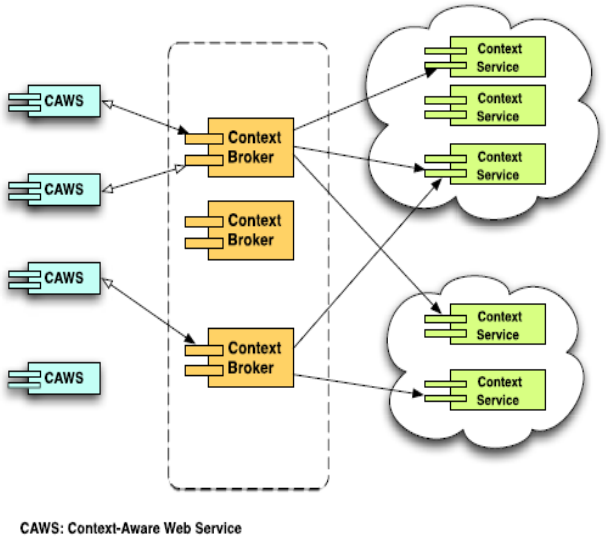

Fig VIII. Framework for Cloud-based Context Provisioning

2.6.1. Context Brokers

A *context broker* is a mediator service that decouples context consumers from context services. It is in charge of handling subscriptions of context consumers in which they express their interest to receive context information, and registration of context services. Context services may then publish their newly acquired context information to the context broker, which notifies context consumers about that newly acquired context information. Context brokers can also be deployed on the cloud. Fig. 2 illustrates our topic-based publish-subscribe system in which context services are the publishers and the CAWSs are the subscribers.

Context information -- such as *location*, *temperature*, and *user activity* -- represents the topics of the system. The Publish/subscribe messaging model is a one-to-many pattern of asynchronous message distribution based on registration of interest. In this model, publishers associate the name of a topic to each message publised rather than addressing it directly to subscribers. Then, the message system sends the message to all eligible recipients that expressed their interest in receiving messages on that topic (―subscribe‖). As opposed to point-to-point messaging systems, such as message queuing, the publish/subscribe model of asynchronous communication is a far more scalable architecture. This is because the source of the information has only to concern itself with creating the information, and can leave the task of servicing potential recipients to the messaging system. It is a loosely coupled architecture in which senders often do not need to know who their potential subscribers are, and the subscribers do not need to know who generates the information.

In addition to this publish/subscribe model for provisioning context information, a context broker implements a regular on-demand request/response model, in which it requests up-to-date context information from context services once a context consumer requires information for a given topic. Therefore, a context broker may either pull context information from context services or let context services push updated context information.

Context services, typically residing in different clouds, deliver context information to context consumers with various quality-of-context and quality-of-service (QoS). Therefore, the Context Broker is in charge of selecting appropriate context services to deliver context information to which a context consumer has subscribed. Context information may be delivered to the same consumer by several context services. Each one may deliver a piece of context information (a topic) that the consumer requires to adapt its behavior to the current context of a user. In Sub-section 4.5, we describe a selection algorithm that allows ranking context services with regard to the QoC and the topics required by a context consumer.

2.6.2. Context Consumers

In our framework, context-aware Web services (CAWS) are the consumers of context information obtained from the cloud-based context services. A CAWS is a Web service that can understand situational context and can adapt its behavior according the changing circumstances as context data may change rapidly

2.6.3. Cloud-based Context Services

As we have mentioned earlier in the related work section, high-level context information is typically obtained from context services that aggregate raw context information sensed by sensors and mobile devices. Given the massive amount of context data processed and stored by context services and the wide acceptance of the cloud computing technology, context providers now can leverage their services by deploying them on the cloud.

Fig. IX depicts the process of context acquisition and the deployment of context services on the cloud to provide high-level context information to context consumers. Raw context data sensed by various devices and sensors is processed, aggregated by *Context Aggregator* components in a structured format, and then uploaded to the cloud-based context services

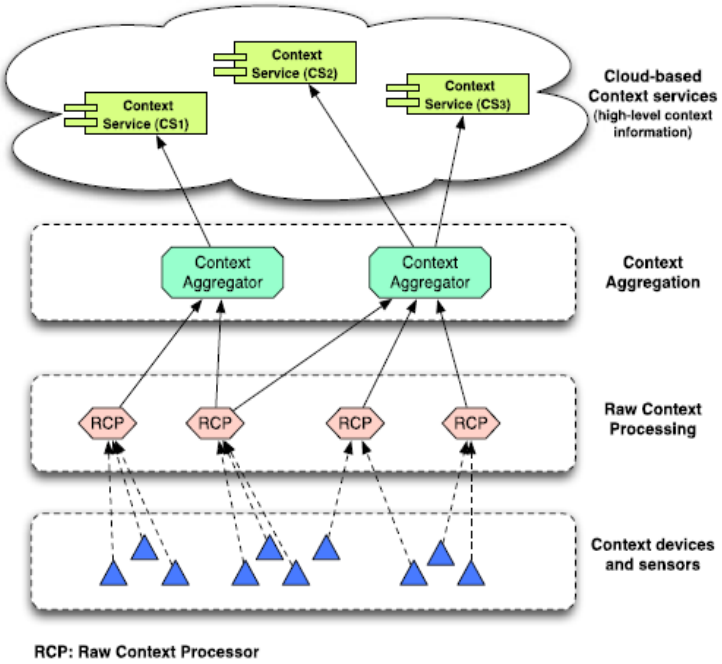

Fig IX. Deployment of high-level context information on the cloud

One of the underlying advantages of the deployment of context services in the cloud is the economy of scale. By making the most of the cloud infrastructure provided by a cloud vendor, a context provider can offer better, cheaper, and more reliable services than is possible within its premises. The context service can utilize the full processing and storage resources of the cloud infrastructure if needed. Another advantage is scalability in terms of computing resources. Context providers can scale up when additional resources are required as a result of a rise in the demand for context information. Conversely, they can scale down when the demand for context information is decreasing. Another benefit of the approach is to enable context-aware application services to acquire their required context information on a pay-as-you-go basis and to select cloud-based context services on the basis of the price they have to pay and other criteria, such as the QoC they can get. Furthermore, context-aware applications can obtain context information from cloud-based context services without having to be involved in context management. The net benefit for consumers and mobile users, in particular, is the ability to receive better services tailored to their current context.

2.7. Importance of Context Aware Security in Scenario of Cloud Computing

One of the significant benefits of cloud computing is that it provides scalable access to computing resources and information technology (IT) services. This innovative and accessible resource is an integration of hardware and software which organizations or individuals can utilise anywhere in the world *via* the internet. Additionally, it will ground the opportunity to penetrate and disseminate the computing resources and information technology (IT) services within the world industries, such as the construction industries. If appropriately implemented and adapted, it will ultimately enhance the productivity, efficiency and effectiveness of the construction industry.

Information communication technologies (ICT) are reaching their main target of developing an environment in which anyone can easily access any information they may need at any time and from anywhere (cloud computing). On the other hand, the cost reduction of information delivery has drawn the construction industry into an ocean of unmanageable amounts of information (both useful and useless). Therefore, it is important to provide useful information, merely for a specified user, at a given time and according to the user's context (context-awareness). Awareness of a user's context (such as their role, task, preferences, location and site conditions, etc.) in mobile construction applications will enhance the effectiveness of project delivery by providing information and services relevant to a particular context (Fath i.*et al*., 2009). Therefore, awareness of a user's context could provide efficient and effective information, communication and services throughout the entire construction supply chain to enhance the success of construction projects. The following part of this research will briefly discuss the proposed system for integrated, context aware, cloud computing within the construction industry.

III. CONCLUSION AND FUTURE SCOPE:

This paper focuses on cloud environment as a distributed computing scenario, being a popular model for today's highly growing technological industry, enlighten with the perspective of context awareness, adapting the dynamic nature of information security. Briefly describing the cloud environment as well as context scenario, this paper has not provided any particular solution, rather focuses on impact of context aware scenario on distributed environment like cloud.

REFERENCES

[1] Elarbi Badidi, Larbi Esmahi,"A Cloud-based Approach for Context Information Provisioning", published on World of Computer Science and Information Technology Journal (WCSIT) Vol. 1, No. 3, 63-70, 2011

[2] "The future of information security is context aware and adaptive", Gartner White Paper, May 2010

[3] A.K. Dey, "Understanding and Using Context,‖ Journal of Pervasive and Ubiquitous Computing, vol. 5(1), pp. 4–7, 2001.

[4] K. Henricksen, J. Indulska, T. McFadden, and S. Balasubramaniam,"Middleware for Distributed Context-Aware Systems,‖OTM Confederated International Conferences, pp. 846-863, Springer-Verlag, 2005.

[5] Maneesha Sharma, Himani Bansal, Amit Kumar Sharma ,"Cloud Computing: Different Approach & Security Challenge", published in International Journal of Soft Computing and Engineering (IJSCE), Volume-2, Issue-1, March 2012

[6] Tripathi,A.; Mishra, A.; "Cloud Computing Security Considerations", Signal Processing, Communications and Computing (ICSPCC), 2011 IEEE International Conference.

[7] T. Strang, C.Linnhoff-Popien, "A Context Modeling Survey, In Workshop on Advanced Context Modelling, Reasoning and Management", UbiComp2004, Nottingham/England, 2004.

AUTHOR'S BIOGRAPHY:

SHARMISTHA DEY received the B.Sc Degree from University of Calcutta in 2004, the MCA degree from West Bengal University of Technology in 2007 and M.Tech degree in Computer Science and Applications in 2013 from University of Calcutta. Diploma in Mass Communication, Public Relations and Journalism. She is an Associate Member of CSI(I). She is currently working as Assistant Professor under the Department of Computer Application in Gurunanak Institute of Technology. Her teaching and research areas include Cloud Computing and Computer Networks and Mobile Communication. Ms. Dey may be reached at papri.dey5@gmail.com

