A SECURITY FRAMEWORK IN CLOUD COMPUTING INFRASTRUCTURE

Arijit Ukil[1], Debasish Jana[2] and Ajanta De Sarkar[3]

[1] Innovation Lab, Tata Consultancy Services, Kolkata, India
[2,3] Birla Institute of Technology, Mesra Kolkata Campus, Kolkata, India

ABSTRACT

In a typical cloud computing diverse facilitating components like hardware, software, firmware, networking, and services integrate to offer different computational facilities, while Internet or a private network (or VPN) provides the required backbone to deliver the services. The security risks to the cloud system delimit the benefits of cloud computing like "on-demand, customized resource availability and performance management". It is understood that current IT and enterprise security solutions are not adequate to address the cloud security issues. This paper explores the challenges and issues of security concerns of cloud computing through different standard and novel solutions. We propose analysis and architecture for incorporating different security schemes, techniques and protocols for cloud computing, particularly in Infrastructure-as-a-Service (IaaS) and Platform-as-a-Service (PaaS) systems. The proposed architecture is generic in nature, not dependent on the type of cloud deployment, application agnostic and is not coupled with the underlying backbone. This would facilitate to manage the cloud system more effectively and provide the administrator to include the specific solution to counter the threat. We have also shown using experimental data how a cloud service provider can estimate the charging based on the security service it provides and security-related cost-benefit analysis can be estimated.

KEYWORDS

Cloud computing, security, PaaS, IaaS& authentication

1. INTRODUCTION

Cloud computing provides a distributed computing environment comprising of heterogeneous facilitating components like hardware, software, firmware, networking as well as services. Challenges arise when access through the cloud infrastructure is done from a public domain like internet. Even when privately held, security challenges prevail. Internet or even a private network provides the required backbone to deliver the cloud services. Common cloud services are: IaaS, PaaS, and Software-as-a-Service (SaaS). Cloud computing is a paradigm shift to traditional computing technology. It provides on demand storage, application execution, information processing, data availability, analytics and other services. This kind of model of metered usage of infrastructure, application, data and services bring about economy of scale, reduced computing and storage cost. However, without adequate assessment of the capability, benefit, vulnerability and optimality, cloud computing may pose severe challenges and threats, which can transform the immense advantages to massive risk and catastrophic loss. Cloud security is such an area which deals with the concerns and vulnerabilities of cloud computing for ensuring safer computing environment. The unorthodox architecture and operation of cloud operation bring in different security and privacy vulnerabilities. The very characteristics of cloud to offer shared infrastructure, virtualization, and redundancy are key enabler for different security attacks. Cloud

DOI : 10.5121/ijnsa.2013.5502

security helps in delivering the resilience for different attacks to disrupt the confidentiality, integrity and availability of cloud information and user data. In order to enable the security primitives in cloud computing systems, both clients and cloud server should act cooperatively and an efficient architecture should be capable of handling most of the security challenges. It is understood that traditional blacklist approach is no longer valid and whitelist (trusted) approach has to be incorporated to mitigate the problems of cloud security. It is worthy to find trustworthiness of cloud service providers based on some parameters like system update frequency, mean down time, previous attack history [1]. Cloud security suffers from 'principal agent' problem [2]. Other problem like multiplexing clients' VMs in a shared physical infrastructure causes security threats to clients while maximizing revenue to the provider [3]. In order to create substantially secure cloud computing environment from client's perspective, it is required that we should explore the challenges and issues of security in cloud computing like:

- To provide data confidentiality for clients / cloud users.
- To enable cloud information integrity.
- To ensure application independent single sign-on (SSO) kind of authentication.

In this paper, we explore the above-mentioned challenges and issues of security concerns of cloud computing through different standard and novel solutions. This paper identifies basic security challenges and enumerates a number of well-known techniques that can be used for improving security. We propose a security-enabled cloud environment which enforces significant protection of client's interest and security concerns over its data. Our proposed architecture is modular in nature, i.e. we consider the threats individually and seek solution for that. This helps to manage the cloud system more effectively and provide the administrator to include the specific solution to counter the threat. For example, in some cases or for some clients, confidentiality is the only requirement whereas for some clients other primitives are also required. Based on the requirement like security strength, latency, bandwidth, the administrator can choose the appropriate primitives. This also helps in mitigating the scalability issues.

This paper is organized as follows. In Section 2, related work done by several researchers is documented. In Section 3, we discuss about the security in cloud infrastructure, its key issues and open challenges. Section 4 depicts proposed architecture for implementing cloud system security, particularly XACML based authentication and the Security-as-a-Service in a cloud system. We describe experimental results of secure registration service and the related cost-benefit analysis in Section 6. Finally, In Section 6 we conclude the paper citing our future work.

2. RELATED WORK

Conner et al [1] have presented an effective reputation management system with associated trust establishment through multiple scoring functions and implemented the security service on a realistic application scenario in distributed environments. Friedman and West [2] and Ristenpart et al [3] have depicted several privacy as well as security issues that arise in a cloud computing framework. Yan et al [4] had proposed a nice scheme for handling data protection in terms of confidentiality through amalgamation of identity management with hierarchical identity-based cryptography for distribution of the key as well as mutual authentication in the cloud infrastructure. In [21], trust and reputation based scheme in collaborative computing is presented. Hu et al [23] presented Law-as-a-Service (LaaS) model for automatic enforcing of legal policies in the super-peer to handle queries for consumers and clients. The law-aware super-peer acts as a guardian providing data integration as well as protection. Sun et al [22] presented the pay-as-you-go business model of cloud infrastructure and put forward the urge of providing high security for cloud computing as this is going over publicly accessible internet domain. Trust needs to be established means for better security of cloud platforms. They [22] presented a dynamic multi-dimensional trust model with time-variant comprehensive evaluation multi-dimensional method. With this backdrop, we present our proposed architecture and security model towards better protection of confidentiality, privacy in a public domain cloud infrastructural backbone.

3. SECURITY IN CLOUD COMPUTING

Security is a big challenge in cloud system due to its nature of outsourced computing. Mainly, confidentiality, integrity and authentication are the primary pain areas. Unless robust security scheme and user-centric security policy is implemented, cloud system would be vulnerable to different attacks and susceptible by the users. Below we mention the key issues of ensuring the cloud security and the open challenges to be addressed for making cloud security system atleast at the smae level of current IT systems.

3.1. Key Issues

Confidentiality prevents intentional (malicious) or unintentional disclosure of sensitive information. In cloud systems, confidentiality incorporates data encryption to minimize vulnerability due to covert channels, traffic analysis, and sensitive inference. Web Service Security is frequently used by the cloud service provider, where data confidentiality and integrity are done using XML encryption which is endorsed by X.509 certificate and Kerberos tickets into SOAP message header [4]. Malicious activities can be defended through a protected hypervisor through HyperWall architecture using the concept of hardware centric hypervisor-secure virtualization [20]. For guaranteeing data integrity at rest or storage, particularly in IaaS and PaaS systems, trusted infrastructure [23] needs to be incorporated. For data integrity in transit, traditional digital signature can be used. However, for guaranteeing data integrity at rest or storage, particularly in IaaS and PaaS systems, trusted infrastructure [13] needs to be incorporated. Traditional security techniques for enterprise and home computing systems cannot address the cloud server security problems [14]. Currently, authentication, authorization and access control services are provided using OpenID, OAuth, SAML, XACML types of primitives [8 -11]. However, XACML has the capability of attribute based access control, which is most suitable in cloud environment.

3.2. Open Challenges

One of the primary focuses to provide cloud security is to have one integrated solution enabling the required security primitives like confidentiality, authentication and integrity. Cloud security cannot be solved using conventional IT security tools as private data is migrated from local machines to global or distributed systems for storage, processing and computing. It is required to consider cloud security from a holistic point of view rather than solving the problem requirement basis. In [2, 17], it is described that cloud specific security solutions like confidentiality-enabled computing, user-defined authentication and access control, atomic data integrity are the main issues to be addressed for a sustainable and scalable. In this paper, our approach is to find an integrated solution for cloud security catering the requirements for confidentiality, integrity and authentication.

4. PROPOSED SECURITY MODEL AND IMPLEMENTATION ARCHITECTURE

We propose a framework for satisfying cloud security ensuring the main primitives: confidentiality, integrity and authentication (with access control). Then we integrate the individual proposals to provide an integrated cloud security which would be offered as a security service.

4.1. Confidentiality in Cloud Infrastructure

With the ownership of client's sensitive data at cloud service provider, it becomes highly unlikely to protect data from the respective service provider while there are well-established techniques available to resist the external threats [5]. One of the solutions is to introduce the concept of data analysis and processing at the provider without the content of client's data gets revealed. However, it is difficult to provide services on the encrypted data. Client's data needs to be processed and analysed in original or raw form at the cloud to enable meaningful applications. This defeats the confidentiality of user's data. In order to retain confidentiality as well as deriving services out of data by third party application, processing on encryption domain is required. This is termed as homomorphic encryption in [6 - 7]. Suppose, cloud service provider requires to compute some arbitrary function f on client's (one or many in number) data d1, … ,dN. This can be done two ways:

- $f(d_1, d_2, ... d_N)$
- $f(E(d_1), E(d_2), ... E(d_N))$

Where E is encryption on d.

Homomorphic encryption scheme allows to efficiently compute arbitrary functions over encrypted data i.e., given encryptions E(d1), …, E(dN) of d1, … , dN for any computable function *f*. In order to ensure client data security from cloud service provider where cloud service provider needs to compute on client data, homomorphic encryption is the only available option. Though it is in developing stage and incurs high computational cost for sophisticated functions, we propose the following principle to balance between the security and usability as depicted in Fig. 1.

```
1. Begin
2. Find trust score of the CSP as per client's confidentiality
   requirement
3. If the CSP $\in$ "Blacklisted CSP" $\vee$ "Sensitive application"
     Use Homomorphic encryption
4. Elseif client data $\in$ "Sensitive" $\wedge$"Statistically important"
     Negotiate with privacy primitives
5. Else
     Use standard TLS/SSL
6. End
```

Fig. 1. Confidentiality-usability algorithm

4.2. Authentication architecture and Policy in Cloud Infrastructure

For providing identity management to warrant authentication and authorization, OpenID and OAuth standards are defined using cloud specific security and privacy policy. OpenID is an open standard, which enables users to authenticate in a decentralized manner, eliminating the need for services to provide their own ad hoc systems and allowing users to consolidate their digital identities [8]. The most important feature is that it provides unique web identity, very much suitable for different cloud applications. OAuth (Open Authorization), on the other hand, is an open standard for authorization. Through OAuth protocol, cloud users can share their data among different cloud service provider without disclosing the authentication credential [9].

On the other hand, XACML (eXtensible Access Control Markup Language), an OASIS-ratified, is a declarative access control policy language [1o, 11]. It is general-purpose and XML-based for security-privacy policy management, implementation and provisioning for access decisions. Advantage of XACML is its ability to provide support for role based access control service [11]. XCAML is suited for providing policy-based access control and authorization services in cloud environment. A trusted third party or the cloud service provider hosts the XACML decision engine consisting of decision implementation by Policy Decision Point (PDP) and policy based enforcement by Policy Enforcement Point (PEP).

The proposed protocol for XACML-based cloud authentication is described below:

- When an user request for a resource at the cloud, a Request.xml file is generated by PEP at the user side and is sent to cloud.
- Cloud service provider's PEP intercepts user's access request to a resource.
- Cloud maintains Policy.xml for that resource which can be defined by the resource owner.
- Cloud maintains PDP module to evaluate and issue authorization decisions and generates Response.xml from
- Cloud user with PEP may additionally request the admin to update Policy.xml to customize the access policy.

Following we depict an activity diagram for better understanding of the protocol for XACML based cloud authentication. In this case, we consider cloud service provider as a reliable (trusted) third party.

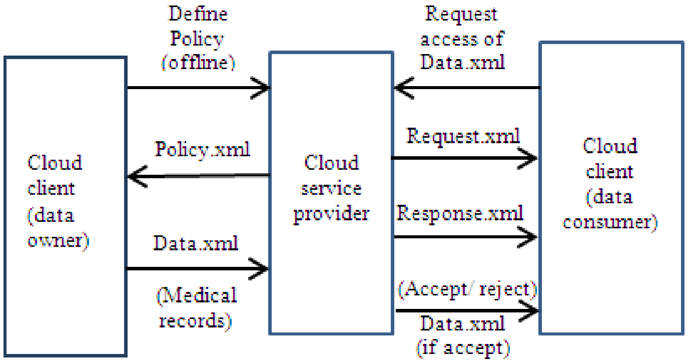

Fig. 2. XACML based cloud authentication protocol

It is to be noted that Policy.xml is a very sensitive and an important file maintaining the access and authorization policy for different applications and cloud users. The cost of security breach on Policy.xml is very high and should be stored in encrypted or hardware-secured method.

Another important requirement is to ensure higher usability such that client users with multiple application subscription. Cloud accounts should easily access data while security is safeguarded. One of the striking usability features is to provide Single-Sign-On (SSO) based authentication so that the user can maintain only single authentication credential for accessing different applications, even different cloud service providers. The usability of SSO also has a reverse side, through which security breach is possible as mentioned in [12]. SSO in cloud is to be introduced after closing the vulnerabilities as stated in [12], where the authors pointed out eight logical flaws with popular SSOs. However, due to the ease and usability point of view, of late SSO becomes highly popular [18]. Another implementation for cloud SSO is found in [19], where SaaS application's audit and access control for public or private cloud is SSO-based. Following architecture can be conceptualized as depicted in fig. 3, where a cloud user authenticates through a cloud SSO hosted by a particular cloud service provider to access other cloud apps, other (owned) cloud service provider accounts, even authorized data of other cloud user.

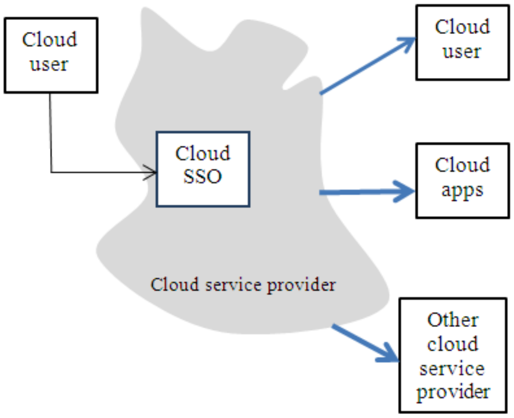

Fig. 3. SSO- based cloud authentication

4.3. Integrity architecture and Policy in Cloud Infrastructure

Cloud service is very much prone to non-invasive side channel attacks like software attacks (malware, virus), statistical attacks (password guess) due to its distributed nature. The most effective solution to counter this problem is to provide secure execution environment to ensure protected execution of software and other processing entities including external and internal memory, which can be offered through secure hardware platform.

Trusted Computing Group (TCG) is engaged in formulating a separate layer of hardware security within cloud infrastructure [15]. This is a hardware-enforced security environment for isolating code execution process and execution area and incorporating safe storage and integrity checking during different periods of software execution. Ensuring trusted hardware (like Trusted Platform Module (TPM) [16]), secure booting, public key based integrity checking along with frequent system validation and consequent application access control helps to build a trusted cloud platform as shown in fig. 4. The main objective is to provide secure execution of the application by employing application access control through software and hardware level security. This kind of hardware-based security is very much needed in securing storage service in PaaS and IaaS. But such TPM-enabled environment is computational inefficient, it is design challenge to optimally partition the secure and normal execution sections such that only sensitive data parts are executed at the protected section, where as non-sensitive data can be stored and executed from the normal (not TPM-enabled) section.

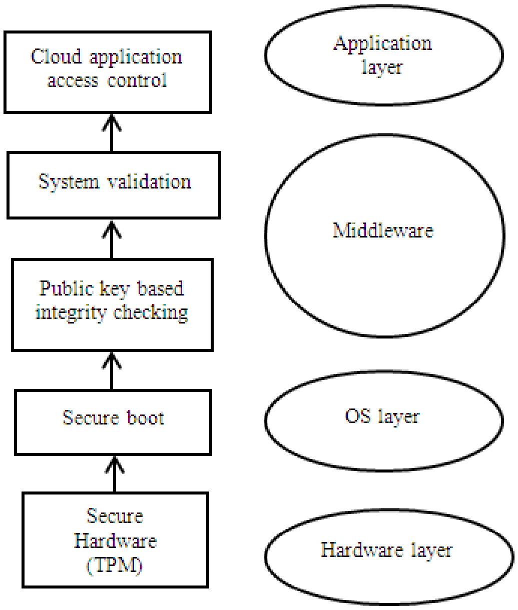

Fig 4. Integrity-ensured trusted cloud computing

It can be observed that through this integrity-ensured trusted cloud, an end-to-end trust (from hardware layer to application layer) chain of trust can be established. When the client is capable of moderate processing power and transacting sensitive records like financial data, medical reports, the proposed trust establishment is necessary to ensure reliability for data at rest.

4.4. Integrated Cloud Security Architecture

In order to make the cloud ecosystem capable of handling the security aspects, discrete components for countering specific attack may not be manageable. Another interesting and unique feature of cloud security is that security can be provided as a service like software, platform or infrastructure. Security-as-a-service has potential because of two reasons. Firstly, due to continuous and fast shift of IT and enterprise security load through outsourcing and customization. Secondly for scalability purposes, security solutions require to handle growing complexity of the underlying processes and to adapt to the paradigm shift to the cloud computing [17]. The concept of security as a top-up on different applications may not suffice the requirement of cloud system. An important feature of cloud security is to provide on-demand. This means that the cloud user or cloud application based on requirement can subscribe the particular security components and thus introducing security as a service. On the other hand this feature can be handy to create an adaptive-secure cloud system, where based on applications or users context certain security primitives or APIs will be called to defend from possible threats. We envision that architecture as:

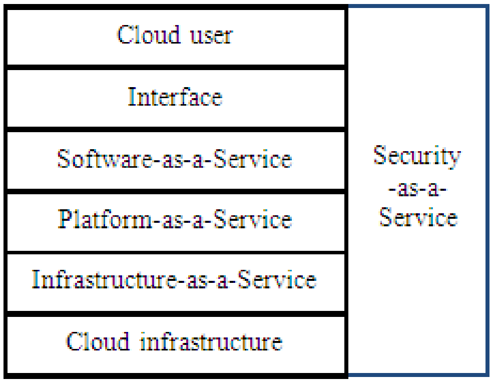

Fig 5. Conceptual cloud service model with security-as-a-service

“Security-as-a-Service” consists of different components like PaaS consists of service, compute components; IaaS consists of storage, network components. The main components of “Security-as-a-Service” are:

- Data confidentiality, e.g. homomorphic encryption
- Data integrity, e.g. TPM (at storage), digital signature (at transit, for data exchange), SHA-2
- Data authentication, e.g. OpenID, OAuth, XACML
- Network confidentiality, e.g. HTTPS (SSL/ TLS)
- Network integrity, e.g. SHA-2, digital signature
- Network authentication, e.g. Cloud SSO

Based on the requirements, these components can be incorporated on demand basis. For example, for storage security or data at rest integrity data integrity components with TPM can be used; cloud users can negotiate with cloud service provider for homomorphic encryption such that user’s data is processed in encrypted domain. Similarly, this security primitive can be integrated in a proactive or adaptively to different rendered services for seamless protection against possible attacks. For example, when SaaS is handling request for financial transactions more security primitives (like HTTPS, XACML, digital signature) are used while handling request for chat applications, HTTPS, digital signature are not needed. Table 1.shows a typical requirement of the primitives of security-as-a-service for other services and stake holders.

Table 1.Security-as-a-service for other cloud services and stakeholders

Services/ Stakeholders	**Security primitives of Security-as-a-service**
Cloud user	Homomorphic encryption, TPM
Cloud infrastructure	TPM, SSO
SaaS	OpenID, OAuth, XACML, HTTPS
PaaS	Homomorphic encryption, OpenID, OAuth, XACML
IaaS	TPM

4.5. Cloud Computing with Security-As-A-Service

Security-as-a-service, as defined earlier is to be availed as a horizontal service in a cloud service model. In this section, we describe a use case of an e-health system using security-as-a-service. There are different parties in the e-health system like medical practitioner, patient, hospital, medicine retailer, nursing staff, insurance agency, medical researcher, so on and so forth designated as $\tau = [\tau_i]$, i= medical practitioner, patient, medicine retailer, nursing staff, hospital, insurance agency, medical researcher… The e-health system is hosted in a cloud service provider C with PaaS model. We denote the sensitive medical record of the patient as D. The patient intends to share $D = [D_p, D_s]$, where p stands for public, s stands for sensitive. The cloud client with medical record (patient) D is hosted in C with following security constraints S :

1. For τ_i, where i = medical researcher, only aggregated result on D_s would be shared.
2. For τ_i, where i= medical retailer only medicine part of D_s is to be shared.
3. For τ_i, where i= nursing staff, only medicine and some related part of D_s is to be shared.
4. For τ_i, where *i*= insurance agency, cost, primary investigation and medicine part of D_s is to be shared.
5. All τ_i is to be authenticated.
6. D_s is to be stored securely in C, D_s is to be shared to τ_i through C in a secure channel, i.e. $C \xrightarrow{D_s \text{secured}} \tau_i$.

When the cloud client, the owner registers to the cloud C for allowing C to host D, client $\tau_{patient}$ gets registered and authenticates to C using OpenID. When $\tau_{patient}$ avails the service of e-health application, it posts its medical record D to C undersigning with the constraints S. There can be a negotiation process between $\tau_{patient}$ and C such that C accepts a subset of S. For sake of simplicity, we do not consider the negotiation phase. In fig. 6, we show the initial data hosting and constraint sharing between $\tau_{patient}$ and C.

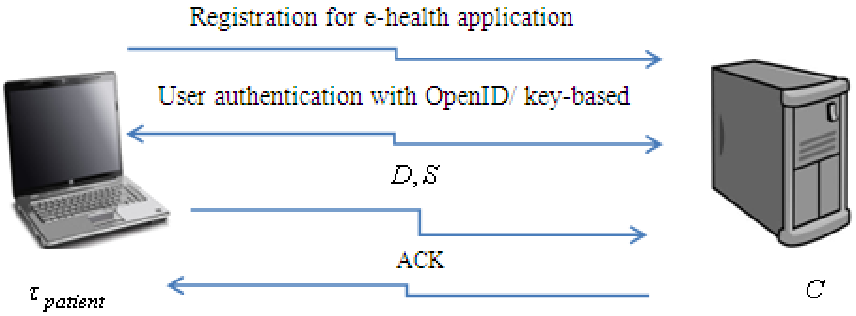
Registration for e-health application
User authentication with OpenID/ key-based
D, S
ACK
τ patient
C

Fig 6. Registration and security-constraint sharing

After $\mathcal{T}_{patient}$ registers in $\mathcal{C}$, shares $\mathcal{D}$ and $\mathcal{S}$, it is the responsibility of $\mathcal{C}$ to ensure the security requirements $\mathcal{S}$ as per when acknowledgement is made. It is to be mentioned that service and business model does also participate as the amount of $\mathcal{D}$ and $\mathcal{S}$ directly impact the pricing of rendering the service.

After $\tau_{patient}$ registers in C , shares D and S , it is the responsibility of C to ensure the security requirements S as per when acknowledgement is made. It is to be mentioned that service and business model does also participate as the amount of D and S directly impact the pricing of rendering the service.

Let us consider that medical researcher intends to avail some information from D through query function Q , which can be searching for a piece of data, aggregated result etc. So, $\tau_{medical\_researcher}$ queries C on D for Q . In order to retain secrecy, C negotiates with $\tau_{medical\_researcher}$ for homomorphic key exchange, public and private key (K_{pu}, K_{pr}) and installing homomorphic encryption agent (if already not present) on $\tau_{medical\_researcher}$. C performshomomorphic encryption on D with K_{pu} and $\tau_{medical\_researcher}$ decrypts with K_{pr} .The decrypted content is Q on D . For example, D may consist of medical investigation data of $\tau_{patient}$ and Q requires information on the investigation data that is higher than reference range. We depict the protocol in fig. 7. Our proposal is to address this issue through functional encryption. However, other cryptographic primitives can be used.

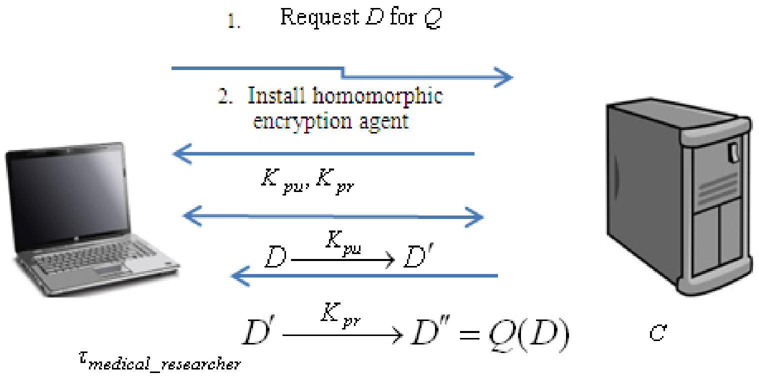
1. Request D for Q
2. Install homomorphic encryption agent
K_{pu}, K_{pr}
$D \xrightarrow{K_{pu}} D'$
$D' \xrightarrow{K_{pr}} D'' = Q(D)$
$\tau_{medical\_researcher}$
C

Fig 7. Functional confidentiality in cloud computing

In order to satisfy other constraints primitives from security-as-a-service needs to be incorporated. For example, satisfying 6 requires HTTPS channel set up among $\tau_{patient}$, D and $\tau_{medical\_researcher}$ for data sharing. For 5, OpenID and OAuth primitives need to be set up.

5. RESULTS AND ANALYSIS

We have implemented the registration and security-enabled data sharing (as described in Fig. 6) in a PC with following specification: Intel Core2Duo CPU with speed 3.00 GHz, 2 GB RAM. The development platform is Python 2.6.1 in Python integrated Development Environment

(IDLE). Instead of, we implemented key based secure-registration service. We experimented with three types of keys:

Table 2. Key-based secure-registration

Authentication primitive	**Key-length**
AES	128
MD5	256
RSA	1024

In cloud computing environment, client device is mostly thin and the cloud system needs to handle millions of services per second. Service registration and secure data transmission being integral part of every transaction, the cost of such computation is of utmost importance while providing Security-As-A-Service, particularly when metering the authentication service. Below is the latency measurement of different key-based secure-registration mechanisms. In Fig. 8, we observe that the secure-registration latency using AES- 128 and MD5 is similar while that of using RSA 1024 costs substantial latency (around 3 times). It is understood that public key based (RSA 1024) authentication is better [25], but it costs more. So, we can propose that the clients want RSA-based authentication has to be charged more. This ensures better secure-registration at the cost of higher charge.

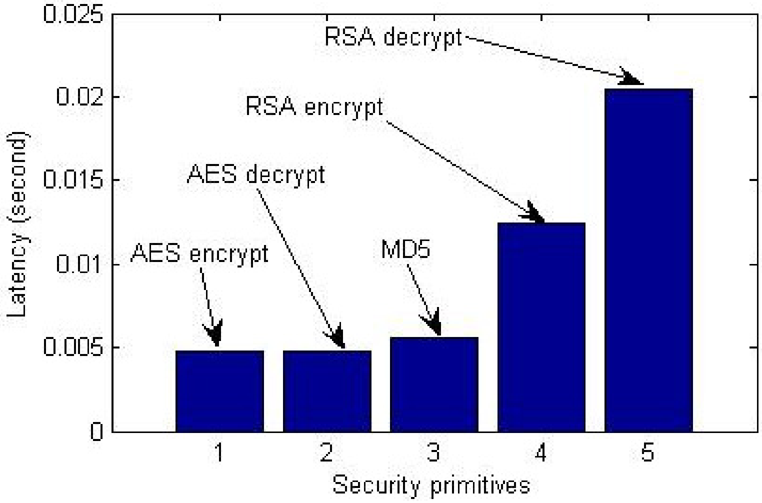

Fig 8. Key-based secure-registration latency

Another important feature needs to be considered is the amount of cloud resources like bandwidth consumed while performing the security services. Based on that parameter, client is charged. Under the similar computing environment, we have experimented with fixed data size of 37 Bytes and found that RSA-based mechanism consumes highest bandwidth as shown in fig. 9.

Based on our experiments conducted, we can conclude that providing security-as-a-service components like secure-registration service, cost-benefit trade-off and requirement analysis are required. We propose that when the requirement is for high-level security, then RSA-based scheme, which is expensive in terms of both latency and bandwidth consumption is to deployed, otherwise AES or MD5 based scheme is to be used. Cloud service provider would also charge incremental for RSA-based scheme over other mechanisms, as we have shown that it is computationally more expensive. Our analysis eases the burden of cloud service provider to meter the security service it renders.

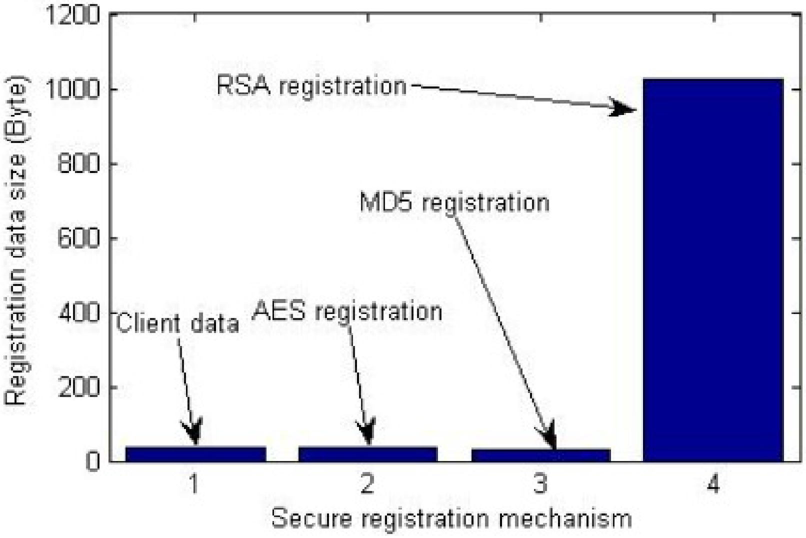

Fig 9. Key-based secure-registration bandwidth consumption

6. CONCLUSION

In this paper, we have analyzed the problem of security in cloud computing. This paper provides security architecture and necessary support techniques for securing cloud computing infrastructure. It assumes to address following challenges to provide data confidentiality for clients / cloud users, to enable cloud information integrity and to ensure application independent single sign-on (SSO) kind of authentication. We have emphasized the data security with the assumption that the problem of network security or security of data at transit can be handled by the present state-of-the-art solution. Our main focus is to describe the problems on data confidentiality, data integrity and data authentication and our security concern aims at the cloud user perspective. We envision that in cloud computing, cloud users or clients are most vulnerable to different security threats. We have provided solutions to counter these threats for securing cloud user's data when exchanged with the cloud service provider (and processed at the cloud service provider), among different cloud service providers and between other cloud users. We have also used "Security-as-a-Service" as a horizontal service model to support the security requirements of other service models like IaaS and PaaS. However, it is to be noticeable that, cloud security research has just started its journey and it is long way to go before ensuring full-fledged cloud security. For example, computation on encrypted data is very much essential to provide data confidentiality from cloud security provider while allowing computation. To enable such feature, homomrphic encryption [6 -7] is a good candidate. However, fully homomrphic encryption incurs high computational cost and is not feasible with existing state-of-the-art cloud hardware. There exists immense scope of research to introduce light-weight homomorphic encryption scheme.

REFERENCES

[1] Conner, W., Iyengar, A., Mikalsen, T. Rouvellou, I., &Nahrstedt K, (2009) “A Trust Management Framework for Service-Oriented Environments”, *WWW Conference*, pp891- 900.

[2] Friedman, A. A., & West D. M, (Oct. 2010) “Privacy and Security in Cloud Computing,” *Issues in Tech. Innovation*.

[3] Ristenpart, T. Tromer, E. Shacham, H., & Savage S, (2009) “Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds,” *16th ACM Conference on Computer and Communications Security*, pp199 – 212.

[4] Yan, L., Rong, C., & Zhao G, (2009) “Strengthen Cloud Computing Security with Federal Identity Management Using Hierarchical Identity-Based Cryptography,”*CloudCom*, pp167–177.

[5] Yau, S., S., & Ho G, (2010) “Protection of users' data confidentiality in cloud computing,”*2nd Asia-Pacific Symposium on Internetware*.

[6] Rivest, R. L., Adleman, L., &Dertouzos, M L, (1978) “On data banks and privacy homomorphisms,” *Foundations of Secure Computation*.

[7] Gentry C (2009), “Fully Homomorphic Encryption Using Ideal Lattices,” *41st ACM Symposium on Theory of Computing*, pp169 – 178.

[8] Leiba B, (2012) “OAuth Web Authorization Protocol,” *IEEE Internet Computing*, pp74-77.

[9] Ahmed, A.S, (2011) “OpenID authentication as a service in OpenStack,” *7th International Conference on Information Assurance and Security*, pp372-377.

[10] Keleta, Y., Eloff, J. H. P., & Venter, H S, (2005) “Proposing a Secure XACML Architecture Ensuring Privacy and Trust,” *Research in Progress Paper, University of Pretoria*, http://icsa.cs.up.ac.za/issa/2005/Proceedings/Research/093_Article.pdf (accessed on 24 Aug, 2012)

[11] Xu, M., Wijesekera, D., & Zhang X, (2011) “Runtime Administration of an RBAC Profile for XACML,” *IEEE Transactions on Services Computing*, 4, 4, pp286-299.

[12] Wang, R., Chen, S., & Wang, X F, (2012) “Signing Me onto Your Accounts through Facebook and Google: A Traffic-Guided Security Study of Commercially Deployed Single-Sign-On Web Services,” *IEEE Symposium on Security and privacy*, pp365-379.

[13] Ukil, A.,Sen, J., &Koilakonda S, (2011) “Embedded Security for Internet of Things,” *2nd IEEE National Conference on Emerging Trends and Applications in Computer Science*, pp1-6.

[14] Koopman, P, (2004) “Embedded system security,” *IEEE Computer*, 37, pp795-97.

[15] http://www.trustedcomputinggroup.org (accessed on 27 Aug, 2012)

[16] http://www.atmel.com (accessed on 27 Aug, 2012)

[17] Mather, T., Kumaraswamy, S., &Latif S, (2009) “Cloud Security and Privacy: An Enterprise perspective of Risks and Compliance,” *O'Reilly Media, Inc*.

[18] https://developers.google.com/google-apps/sso/saml_reference_implementation (accessed on 27 Aug, 2012)

[19] http://www.cloudaccess.com/saas-sso (accessed on 27 Aug, 2012)

[20] Szefer, J. Lee, R.B. Ruby & B. Lee (2012) “Architectural Support for Hypervisor-Secure Virtualization,” I *7th International Conference on Architectural Support for Programming Languages and Operating System, pp*437 – 450.

[21] Ukil, A (2010) "Trust and Reputation Based Collaborating Computing in Wireless Sensor Networks," *IEEE International Conference on Computational Intelligence, Modelling and Simulation*, pp464 – 469.

[22] Hu Y., Wu W., & Cheng D (2012) “Towards law-aware semantic cloud policies with exceptions for data integration and protection,” *2nd International Conference on Web Intelligence, Mining and Semantics*.

[23] Ukil, A (2011) "Secure Trust Management in Distributed Computing Systems," *IEEE DELTA*, pp116 – 121.

[24] Sun D., Chang G., Sun L., Li F., &Wang X, “A dynamic multi-dimensional trust evaluation model to enhance security of cloud computing environments,” *International Journal of Innovative Computing and Applications, vol. 3, Issue. 4,* pp200 – 212.

[25] http://support.microsoft.com/kb/257591

AUTHORS

Arijit Ukil is currently working in Innovation Labs, Tata Consultancy Services (TCS) Ltd., Kolkata as a Scientist. He is primarily engaged with the research activity on Internet of things, security and privacy and wireless networking. Before joining TCS in 2007, he has worked as Scientist in Deference Research and Development Organization (DRDO) for four years. He has done his B.Tech in Electronics and Telecommunication Engineering in 2002 and currently pursuing PhD from Birla Institute of Technology, Mesra. He has published more than 30 conference and journal papers of national and international repute. He has already published three book chapters. He has been reviewer of a number of IEEE journals like IEEE Transactions on Wireless Communications, IEEE Transactions on Vehicular Technology and conferences like, IEEE VTC, IEEE WCNC. He has been invited and delivered keynote and tutorials in many international and national conferences and symposia. He is enlisted in 2010 Marquis' "Who's Who" as a renowned contributor in the field of computer science and information technology.

Debasish Jana, Ph.D. (Computer Science, Jadavpur University), is currently affiliated with TEOCO Software Pvt Ltd, Kolkata. He obtained his M.Math (CS) degree from the University of Waterloo, Canada, B.E. (CS) from Jadavpur University, and MBA (Finance) from IGNOU, New Delhi. He has extensive professional experience of about twenty-six years in IT industry including PricewaterhouseCoopers, Anshin Software, Techna, Millenium, BFL Software. He has been serving as Visiting Faculty for more than fifteen years at premiere institutions such as Jadavpur University, Army Institute of Management, BIT Mesra Kolkata Campus. A Fellow Member of IETE and IE(I), Senior Member of IEEE, ACM, CSI, Dr. Jana has authored three popular books on C++, Java and Computer Graphics published by PHI Learning. He has authored many papers in national and international conferences and journals. He has been actively involved in Computer Society of India Kolkata Chapter activities in several national and international conferences including EAIT 2006, CSI-2006, CSI-RDHS 2008, EAIT 2011, EAIT 2012 as spearheading role in Program Committee and Editorial role in the Proceedings.

Ajanta De Sarkar is working as Associate Professor in the department of Computer Science and Engineering in Birla Institute of Technology, Mesra. She is having altogether 17 years of experience including 6 years of Industry experience. She had worked with renowned, pioneer companies like TATA STEEL, Jamshedpur, India and Lexis Nexis Inc., Boston, USA as senior programmer and software developer. Having graduated from Bethune College, University of Calcutta in B.Sc. (Mathematics) in 1993, and obtained MCA degree in 1996 from Jadavpur University. She has been awarded PhD (Computer Science and Engineering) from Jadavpur University in 2009. Her field of Specialization is Distributed Computing, specifically Grid Computing. Her focused research area includes Grid Computing, Cloud Computing and Wireless Sensor Network. She has published many reputed journal and conference papers and also selected reviewer of a few reputed conference and journals.