ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 2, pp: (477-484), Month: April - June 2015, Available at: www.researchpublish.com

Integrity Attestation Security Frameworks for Software-As-A Service Cloud

SUMA.S.HOSAMANI

M.Tech 4th Semester, Department Of Computer Science and Engineering, Shri Taralabalu Jagadguru Institute of Technology, Ranebennur, India

Abstract: **Software-as-a-service (SaaS) cloud systems allow application service providers to deliver their applications via massive cloud computing infrastructures. SaaS is becoming an Increasingly prevalent delivery model as underlying technologies that support Web services and service-oriented architecture (SOA) mature and new developmental approaches, such as Ajax, become popular. SaaS clouds are vulnerable to malicious attacks because of their sharing nature. Many security frameworks have been developed to address cloud security issues like IntTest, Privacy Proxy, Trusted virtual data center, Placement and Extraction method for Exploring Information Leakage, Stateful Dataflow Processing, Building Privacy-Conscious Composite Web Services, Anomaly Extraction and Mitigation using Efficient- Web Miner Algorithm. Breif Study on the above frame works are explained below. In this paper, we present IntTest, an effective service integrity attestation framework for SaaS clouds. IntTest provides an integrated graph attestation analysis method that can pinpoint malicious service providers than existing methods. Also IntTest will automatically correct the corrupted result that are produced by the malicious service providers and replace it with good results produced by benign service providers.**

Keywords**: Distributed Service, Integrity attestation, Cloud computing, Multitenanat.**

1. INTRODUCTION

Cloud computing is a technology helps us to keep up data and its application by using internet and central remote servers . Cloud computing has greater flexibility and availability at lower cost. The four deployment models operated by cloud computing are the: Public Cloud, Private Cloud, Community Cloud, and Hybrid Cloud. Private cloud The cloud infrastructure is operated solely foan organization. It may be managed by the organization or a third party and may exist on premise or off premise.Community cloud -- The cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns. It may be managed by the organizations or a third party and may exist on premise or off premise. Public cloud -- The cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling thecloudservices and the comparison of private and public cloud.Hybrid cloud -- The cloud infrastructure is a composition of two or more clouds (private, community, or public). There are different types of cloud service providers like Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS).Here we are discussing about SaaS Cloud system. Software as a Service (SaaS) is a software distribution model in which applications are hosted by a vendor or service provider and made this is available to customers over a network.SaaS is becoming an increasingly prevalent delivery model as underlying technologies that support Web services and service-oriented architecture (SOA) and many other new developmental approaches. SaaS service are suffered from many malicious attacks hence they need security. Below are the various frameworks proposed to provide security.

Now a days the cloud computing technology is popular because it is an attracting technology in computer science field. This paper concentrate on the integrity attacks on software as a service clouds and because of that the user will receive bad results after processing the data. Fig.1 shows the integrity attacks in software as a service clouds. Majority of software as a service cloud solutions are based on a multi-tenant architecture. In the previous research papers confidentiality and

privacy protection problems are studied extensively but the service integrity attestation problem was not properly addressed. In software as a service cloud one of the most important problems that need to be addressed is this service integrity, no matter whether the data processing in cloud is public or private data. In the previous papers they are provided some software integrity attestation techniques but most of them requires special trusted hardware or secure kernel supports and because of these reasons that cannot be deployed in large scale cloud computing. This paper presents IntTest, a new framework for multi tenant cloud systems. This technique provides the novel integrated attestation graph analysis technique that will provide a stronger attacker pinpointing power than the existing schemes. It will automatically enhance the result quality by replacing the bad results that are produced by the attackers by good results that are produced by the benign service providers. This can achieve higher attacker pinpointing accuracy than existing techniques Run Test and Adap Test.

Figure 1: Software-as-a Service

In large-scale multitenant cloud systems, large number of malicious attackers may launch colluding attacks on the targeted service functions to make them malicious. To address this challenge, IntTest takes a holistic approach by systematically examining both consistency and inconsistency relationships among different service providers within the entire cloud system. IntTest checks both per-function consistency and the global inconsistency graphs. An advantage of using this IntTest is it cannot only pinpointing the malicious attackers more efficiently but also it can suppress aggressive attackers and also limit the scope of damage that are caused by the attacks. The experimental result shows that IntTest can achieve more accuracy in pinpointing malicious attackers than any other existing schemes. Also this IntTest is more scalable and it will reduce overhead produced by the attestation more than the other voting schemes.

2. REVIEW OF EXISTING FRAMEWORKS

A. Privacy Proxy:

Zhendong Ma, Jurgen Mangler, Wagner proposes a privacy enhanced design in the paper [2] that mainly aims to minimize personal data disclosure in nested web service by proposing a scalable and light-weight design that uses a privacy proxy to achieve data privacy. This paper also proposes the utilization of service level agreements (SLA's)for user benefits. Two design principles: (1) minimal disclosure- tight control over how many times personal data is accessed, (2) direct disclosure- allow for a means to determine by whom personal data is accessed. Privacy Proxy Service (PPS) is established as a trusted third party in the interaction between a customer and composite services. Its main function of PPS is to temporarily store a customer's personal data items, as well as to control and trace the access to such data. This design offers the following properties:

- Each PDI (personal data item) is stored separately
- Each PDI is stored only for a limited amount of time.
- Each PDI is identified by a unique key, furtherreferred to as ticket.
- Each PDI is only accessible once.
- Tickets are not linkable.

There are three interaction phases: Negotiation, Storage and Retrieval. During storage phase customer stores each PDI in the PPS and for each PDI, a ticket is returned. During the retrieval phase the Business services are communicating solely

with the PPS only (no interaction with intermediate service).However this design does not prevent services from colluding.

***Advantages*:** Scalable, light weight, uses SLA's, transparent, no need to modify existing service and underlying infrastructure, less impact on overall performance.

Drawback: Doesn't prevent service from colluding hence cannot keep the overhead at minimal.SLA negotiations are not dynamic.

***B.* Placement and Extraction method for Exploring Information Leakage:**

This paper [3] aims at the practicality of mounting cross-VM attacks in existing third-party compute clouds. There are two main steps while considering attacks we consider require two main steps: placement and extraction.

Placement refers to the adversary making arrangement for place their malicious VM on the same physical machine similar to target customer. Extraction refers to extract confidential information via a cross-VM attack. This mainly occurs due to the sharing of sharing of physical resources. Here there are two kinds of attackers being considered – first is those who cast a wide net and are interested in being able to attack some known hosted service and second is those focused on attacking a particular victim service. Amazon's Elastic Compute Cloud (EC2) service is taken as example here. Network Probing is used for understanding VM placement in the EC2 system and achieving co-resident we use hard-disk-based covert channel between EC2 instances or determining coresidence In the case of network based co-resident check we say two instances are likely co-resident if they have

(1) Matching Domo IP address,

(2) Small packet round-trip times, or

(3) Numerically close internal IP addresses

Brute force placement is the technique that is being used earlier .Later this was replaced by the new one that assumes an attacker can launch instances relatively soon after the launch of a target victim. The attacker then engages in instance flooding that is running as many instances in parallel as possible, in the appropriate availability zone and appropriate type. Since The EC2 placement algorithms seems inefficient to stop a dedicated attacker there is another method to "patch" all placement vulnerabilities: offload choice to users that is users re-quest placement of their VMs on machines that can only be populated by VMs from their account. Another kind of attack is cryptographic cross-VM attacks. But these kinds of attacks are very difficult to $realize. Co-residence detection can also detected by analysising load variation due to a publicly accessible service running on the target.

Advantages: Help to determine where in the cloud infrastructure an instance is located, whether two instance co resident on the same physical machine, whether an adversary launch instance can be co-resident with other user's instance, whether an adversary can extract cross VM-information leakage, make use of cache based load balancing for keystroke timing attack. Binding techniques are used to minimize the information leakage.

Drawbacks: Methods used for inhibiting side channel attack has two drawback-high overhead, nonstandard hardware, application specific and are not sufficient for mitigating risk, keystroke attacking can be applied only when attackers and victim shares the same core.

***C.* Stateful Dataflow Processing Services:**

This paper [4] propose Robust Service Integrity Attestation (ROSIA) framework. This can efficiently verify the integrity of stateful dataflow processing services and pinpoint malicious service within a large-scale cloud ystem. ROSIA support stateful dataflow Services and hence achieves robustness. ROSIA performs integrity attestation by examining both consistency and inconsistency relationships. This frame work attains higher attack detection accuracy and also limits the scope of the damage caused by colluding attackers. This proposes two methods to attest stateful functions. One method is called indirect state recovery, which relies on replaying a sequence of historic input data to indirectly bring back the state. Another method is difference check, which derives consistency relationship between two stateful service components by comparing result difference produced bytwo consecutive input data. The basic idea behind this is Replay-based Consistency Check.

Advantage: This supports both stateless and stateful service functions, effective and imposes low overhead .This, based on the assumption that the total number of malicious service components is less than that of benign ones in the entire cloud system, higher detection rate and lower false positive rate in certain attack scenarios.

Drawbacks: Malicious service providers can escape from being detected by trying to form a majority clique in the per-function consistency Graph.

***D*. Trusted Virtual Data Center:**

This paper [5] talks about a new technology called The trusted virtual data center (TVDc) which can address the need for strong isolation and integrity guarantees in virtualized environment. We can have controlled access to networked storage based on security labels and prototypes for the enforcement of isolation constraints and integrity checking.

Virtualization is a technology used in data enters for both commodity and high-end servers. This has the ability to aggregate multiple workloads to run on the same set of physical resources, thus resulting in increased server utilization and reduced space and power consumption. Virtualization utilizes a software layer, called virtual machine monitor (VMM) for creating virtual machines (VMs). TVDc provides isolation through employing an isolation policy and different types of workload isolation mechanisms. This policy will abstracts the physical infrastructure and allows for automated policy-driven configuration management of data center resources. The boundaries of a TVD can be defined by labeling all VMs and associated resources within the TVD with a unique TVD identifier known as a security label. Isolation policy has two parts: (1) the label Definition (2) anti-collocation definitions. The access control management is based on the security labels. The different kinds of isolation supported on the workload and administration planes are Data sharing, VMM system authorization, Collocation constraints and Management constraints. For integrity management we can use TVDc to establish trust in a remote computer by verifying the integrity of the software loaded that computer, whether it is a physical or virtual system. For Integrity attestation we require a database of reference measurements that can be compared with run-time measurements from VMs. Therefore isolation management, workload management, and access control are important aspects of cloud computing since there are increased possibilities of misconfiguration.This can cause an additional vulnerability.

Advantages: Provides strong Isolation, Guarantees integrity, ability to aggregate multiple workloads, increased server utilization and reduced space and power consumption, flexibility in server deployment, workload mobility, global service availability at large scale at low cost, ensure that viruses and other malicious code cannot spread from one customer workload to another, prevent data from leaking from one customer workload to another provides policy-driven security management.

Drawbacks: Placing different customers' workloads on the same physical machines may lead to security vulnerabilities, such as denial of service attacks, and possible loss of sensitive data, misconfiguration which caused increased vulnerability.

***E*. Building Privacy-Conscious Composite Web Services:**

This paper [6] proposes a framework that can addresses consumer privacy concerns in the context of highly customizable composite web services. This approach involves service producers that exchange their terms-ofuse with consumers in the form of models. This framework has automated techniques for checking these models at the consumer site for compliance of consumer privacy policies.

In the case of a policy violation, this framework can supports automatic generation of obligations. These obligations are automatically enforced through a dynamic program analysis approach on the web service composition code. This framework consists of five major components: a) service composition code, b) service models, c) privacy policies, d) policy compliance checker and obligation generation, and e) obligation enforcer. Two important problems that need to be addressed in this are1) Policy compliance checking 2) Obligation enforcement. Another important technique which can be built by composing a number of smaller services is Service composition. This is introduced with the goal of providing consumer data privacy in service composition. The natural formalism in service composition includes construing every Component service as a function that maps a set of inputs to a set of outputs. The Privacy policy used here will describe the privacy requirements of a user by defining constraints on how her data could flow between different entities. We use the concept labels to specify the privacy policies. Labels are classified as data labels and principal labels. Labels have two types of attributes 1) Data label attributes and 2) Principal label attributes. A privacy policy can be formalized as a set of

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 2, pp: (477-484), Month: April - June 2015, Available at: www.researchpublish.com

policy rules. To enforce obligations, the composite service needs to track whether the flow of consumer's data inputs respect these obligations. Hence through our framework, consumers can have facilities to specify their privacy concerns through use of privacy policies, while service providers express their terms of use through models.

Advantage: It provides consumer privacy concerns in the context of highly customizable composite web services, supports automatic generation of obligations.

Drawbacks*:* Does not address malicious service providers that intentionally lie about their usage of consumer data, this frame work doesn't give any feedback to the service.

***F*. Anomaly Extraction and Mitigation using Efficient- Web Miner Algorithm:**

This paper [7] deals with Anomaly deviation that affects network security. Anomaly extraction aims to automatically find the inconsistencies in large set of data observed during an anomalous time interval. Those extracted anomalies can be used for root cause analysis,network forensics, attack mitigation and anomaly modeling.

Efficient-Web Miner Algorithm will be used to generatethe set of association rules applied on metadata. Thus these algorithms effectively find the flow associated with the anomalous events.

Advantage: Root cause analysis, network forensics, attack mitigation and anomaly modeling.

Drawbacks: Cannot reduced the problem of candidate set generation by providing an improved candidate set pruning

3. PROPOSED SYSTEM

Software as a service and service oriented architecture are the basic concepts of SaaSclouds and this will allow the application service provider to deliver their application via cloud computing infrastructure. In our proposed method we are introducing a new concept called IntTest. The main goal of IntTest is, it can pinpoint all the malicious service providers. IntTest will treat all the service providers as black boxes and this does not need any special hardware or secure kernel support. When we are considering the large scale cloud system multiple service providers may simultaniously compromised by a single malicious attacker. In this we assume that the malicious nodes are not having any knowledge about the other nodes except those which they are directly interacting. In this proposed system we are making some assumptions. First of all we are assuming that the total number malicious service components are less than that of the total number of benign service providers in the entire cloud. This assumptions is very important because without this assumption, it would be difficult for any attack detecting scheme to work succesfully. The second assumption is the data processing services are important deterministic. That is, the same input that are giving by a benign service component will always produce the same output. And finally we assume that the inconsistency caused by hardware or software faults can be excluded from malicious attacks. Fig. 2 shows the over all aarchitecture of the proposed system. In this the user give request to cloud the serive will be deployed in the cloud the cloud will forward the user request to the SaaS and the response will be send to the cloud by the SaaS. And then the IntTest process will be done. After that the result auto correction will be done. After that the result will be send to the user by the cloud.The architecture shows this IntTest module detail.

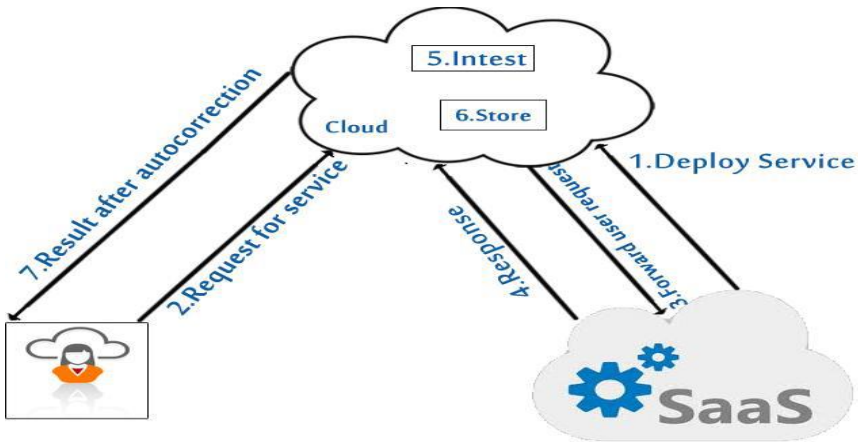

Figure2: Over all architecture of the proposed method

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 2, pp: (477-484), Month: April - June 2015, Available at: www.researchpublish.com

IntTest a powerful integrated service integrity attestation framework [11][12][13][14][10]called IntTest [1] for multitenant cloud systems which can pinpoint malicious attackers [3][17] even if they become majority for some service functions. This scheme does not need any application modifications or it does not assume trusted entities on third-party service provisioning sites. In large-scale cloud systems, multiple malicious attackers may launch colluding attacks on certain kinds targeted service functions and hence invalidate the service. In order to address this challenge, IntTest takes a holistic approach by systematically examining both consistency and inconsistency relationships among different service providers within the whole cloud system. IntTest examines both per-function consistency graphs and also the global inconsistency graphs. Per-function graph analysis can limit the scope of damage caused by colluding attackers. The global inconsistency graph analysis can effectively expose those attackers that may try to compromise many service functions. IntTest can help to suppress aggressive attackers and limit the scope of the damage caused by colluding attacks. This is based onreplay-based consistency check and the integrity attestation graph model. Now consider the consistency check scheme for attesting three service providers' let them be p1, p2, and p3 that offer the same service function f. Here the portal [9][10] sends the original input data d1 to p1and gets back the result f (d1). Next, the portal sends d'1, a duplicate of d1 to p3 and gets back the result f (d'1). The portal then compares both outputs to see whether p1 and p3 are consistent. The main idea behind this approach is that if two service providers disagree with each other on the processing result of the same input, then at least one of them should be malicious. We do not send an input data item and its duplicates concurrently. Instead, we replay the attestation data on different service providers. After receiving the processing result of the original data In order to reduce the delay caused by replay we can overlap the attestation and normal processing of consecutive tuples in the data stream and hence can hide the attestation delay from the user.

4. MODULES

In this section we present the main modules in the proposed system. Mainly it consists of four modules that are described below.

4.1 Baseline Attestation Scheme IntTest is used to detect the service integrity attack and to pinpoint malicious service providers. For that first we are deriving the consistency and inconsistency relationship between service providers. Consider the fig 3 it shows the consistency check method. In that p1,p2 and p3 are the service providers. All of them offers the same function f. The portal sends the original data d1 to the service providers p1 and gets the processing result f(d1). Then the portal sends the duplicate of d1 to p3 and gets the result f(d1'). And if both of them are same means it is consistent and if not means they are inconsistent.that is if two service providers disagree with each other, when processing the same input then any one of them will be malicious. Thus the malicious attackers cannot escape from detecting when they are providing bad results with good results.

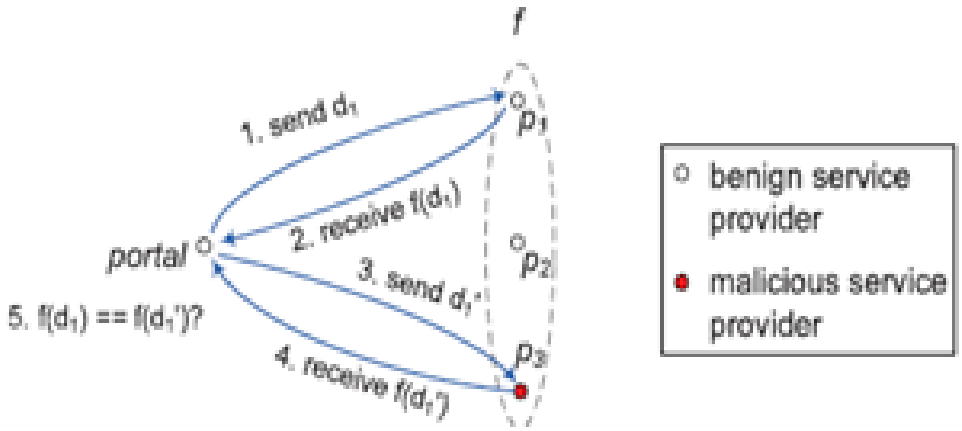

Figure 3: Consistency check

4.2 Integrated Attestation Scheme Here we present an integrated attestation graph analysis algorithm. Step 1: Consistency analysis: In the first step it will examine the per-function consistency graph and will pinpoint suspicious service providers. The consistency links in the consistency graph will provide a set of service providers. It will keep consistent with each other on a specific service function. The benign service providers will always keep consistent with each other and will form a clique in terms of consistency links. The colluding attackers can try to escape from being

detected. Then next we must examine the per-function in consistency graph too. Step 2: Inconsistency analysis: This inconsistency graph will contain only the inconsistency links, this may exist in different possible combinations of the benign node and the malicious node set. First we assume that the total number of malicious service providers in the cloud system is not more than the benign service providers, and then we can pinpoint a set of malicious service providers. If two service providers are connected by an inconsistency link, we can say that any one of them is malicious.

4.3 Result Auto Correction for Attacks:

IntTest can not only pinpoint malicious service providers but also it will autocorrect the corrupted data processing results with good results to improve the result quality of the cloud data processing service. Without our attestation scheme, once if an original data input is changed by any malicious attacker, then the processing result of that input will be corrupted and which will result in degraded result quality. IntTest provides the attestation data and the malicious node pinpointing results to detect and correct compromised data processing results[1]. IntTest will examine both the inconsistency and consistency graphs to make a final decision to pinpoint the malicious service provider. This technique can achieve higher detection rate than any other existing technique and will have low false alarm rate than others. Also IntTest can achieve higher detection accuracy than any other techniques when malicious service providers attack more nodes. This method will identify the attackers even though they attack a very low percentage of services.

5. RESULT ANALYSIS

Considering the parameters like integrity, server utilization, extendibility ,overhead and vulnerabilities we could find that Trust virtual data center and Placement and Extraction method has high server utilization ,low overhead but they have denial of service, malicious service provides can still escape.Stateful data processing method seems to have low overhead and scalability but malicious providers can still escape while Privacy proxy can provide security to user data it cannot avoid colluding attacks. Privacy conscious composite web services has automated techniques for checking models at the consumer site for compliance of consumer privacy policies but still they cannot address malicious service providers that intentionally lie about their usage of consumer data, also have low server utilization. It has low overhead and is scalable. Anomaly extraction using minor algorithm has high server utilization but it has high overhead. IntTest has low overhead, scalable ,high server utilization, doesnot require any special hard ware or secure kernel support and it can provide security from malicious service providers more effectively than any other frame works.

6. CONCLUSIONS

In this paper a wide survey of the different frameworks for providing security to SaaS has been carried out and pointed out their advantages and drawbacks. We need to further improve those frameworks or develop some efficient novel integrated service integrity attestation graph analysis scheme for multitenant software-as-a-service cloud system. IntTest uses a reply based consistency check to verify the service providers. IntTest will analyses both the consistency and inconsistency graphs to find the malicious attackers efficiently than any other existing techniques. And also it will provide a result auto correction to improve the result quality methods.

ACKNOWLEDGEMENT

First and foremost I offer my sincerest gratitude my guide, Asst Prof **Mrs. Pushpa .S. Tembad** who has supported me though out my thesis with her patience and knowledge and assistance of all those people who have made my work on this project pleasant endeavor and I thankful anonymous references for helpful suggestions.

REFERENCES

[1] Juan Du, Member, IEEE, Daniel J. Dean, Student Member, IEEE, Yongmin Tan, Member, IEEE, Xiaohui Gu, Senior Member, IEEE, and Ting Yu, Member, IEEE" Scalable Distributed Service IntegrityAttestation for Software-as-a-Service Clouds"

[2] Zhendong Ma_, J¨urgen Manglery, Christian Wagner_, Thomas Bleier__Austrian Institute of Technology, "Enhance Data Privacy In Service Compositions Through A Privacy Proxy" .

[3] Thomas Ristenpart_ Eran Tromer† Hovav Shacham_ Stefan Savage_Dept. of Computer Science and Engineering †Computer Science and Artificial Intelligence Laboratory University of California, San Diego, USA Massachusetts

Institute of Technology, Cambridge, USA" Hey, You, Get Off of My Cloud: Exploring Information Leakage in Third-Party Compute CloudsDept. Of Computer Science and Engineering".

[4] Juan Du, Xiaohui Gu, Ting Yu Department of Computer Science, North Carolina State University" On Verifying Stateful Dataflow Processing Services in Large-Scale Cloud Systems"

[5] S.Berger,Caceres,K.Goldm,D. Pendarakis,".Security for the cloudinfrastructure :Trustedvirtual data center implementation".

[6] Wei Xu ¤ V.N. Venkatakrishnan y R. Sekar I.V. Ramakrishnan ,Department of Computer Science tony BrookUniversityStony Brook, NY 11790-4400 Email:venkat@cs.uic.edu5" A Framework for Building Privacy-Conscious Composite Web Services"

[7] Gargi Joshi,Department of computer Engineering Dr. D. Y. Patil College of Engineering Of Pune, Ambi, Pune – 410506," Anomaly Extraction and Mitigation using Efficient-Web Miner Algorithm"

[8] QoS-Assured Service Composition in Managed Service Overlay Networks," Proc. 23rd Int'l Conf. DistributedComputing Systems (ICDCS '03), pp. 194-202, 2003.

[9] Towards Standardized Web Services Privacy Technologies," IEEE Int'l Conf. Web Services, pp. 174-183, June 2004.

[10] Managing and Securing Web Services with VPNs," Proc. IEEE Int'l Conf. Web Services, pp. June 2004.

[11] Service-Oriented Virtual Private Networks for Grid Applications," Proc. IEEE Int'l Conf. Web Services, pp. 944-951, July 2007.

[12] .F3ildCrypt: End-to-End Protection of Sensitive Information in Web," Proc. 12thInt'l Conf. Information Security (ISC), pp. 491-506, 2009.

[13] Managing Security in the Trusted Virtual Datacenter," ACM SIGOPS Operating Systems Rev, vol. 42, no. 1,pp. 40-47, 2008.

[14] Security Issues and Security Algorithms in Cloud Computing K.S. Suresh , Prof K.V. Prasad

[15] G. Alonso, F. Casati, H. Kuno, and V. Machiraju, Web Services Concepts, Architectures and Applications (Data-Centric Systems andApplications). Addison-Wesley Professional, 2002.

[16] T. Erl, Service-Oriented Architecture (SOA): Concepts, Technology, and Design. Prentice Hall, 2005.

[17] T.S. Group, "STREAM: The Stanford Stream Data Manager," IEEE Data Eng. Bull., vol. 26, no. 1, pp. 19-26, Mar. 2003.

[18] D.J. Abadi et al., "The Design of the Borealis Stream Processing Engine," Proc. Second Biennial Conf. Innovative Data Systems Research (CIDR '05), 2005.

[19] B. Gedik et al., "SPADE: The System S Declarative Stream Processing Engine," Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD '08), Apr. 2008.

[20] S. Berger et al., "TVDc: Managing Security in the Trusted Virtual Datacenter," ACM SIGOPS Operating Systems Rev., vol. 42, no. 1, pp. 40-47, 2008.

[21] T. Ristenpart, E. Tromer, H. Shacham, and S. Savage, "Hey, YouGet Off My Cloud! Exploring Information Leakage in Third-Party Compute Clouds," Proc. 16th ACM Conf. Computer and Communications Security (CCS), 2009.