International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-8 Issue-10, August 2019

Measuring Security for Applications Hosted in Cloud

Nitin Singh Chauhan, Ashutosh Saxena, J.V.R. Murthy

Abstract*: Despite the numerous benefits of cloud computing, concerns around security, trust and privacy are holding back the cloud adoption. Lack of visibility and tangible measurement of the security posture of any cloud hosted application is a disadvantage to cloud service customers. Decision to migrate workloads on the Cloud requires thoughtful analysis about security implications and ability to measure the security controls after hosting. In this paper, we propose a framework to quantitatively measure different aspects of information security for Cloud applications. This framework has a system through which we can define applications specific controls, gather information on control implementation, calculate the security levels for applications and present them to stakeholders through dashboards. Framework also includes detailed method to quantify the security of a Cloud application considering different aspects of security, control criticalities, stakeholder responsibilities and cloud service models. System and method provide visibility to Cloud customer on the security posture of their cloud hosted applications.

***Index Terms*: Cloud, Security, Authentication, Privacy, Security Metrics.**

I. INTRODUCTION

Cost advantages, faster infrastructure provisioning and scalability are key advantages of Cloud. Increasing uses of Cloud clearly indicates that Information Technology (IT) and IT enabled businesses are finding value in Cloud migration of their business applications and data. However, security, privacy, and trust remain major concerns for the customer in the cloud adoption journey [1]. Cloud customers are worried about relinquishing controls of application/data to a third party as they have limited or no visibility about security and privacy practices followed by the Cloud Service Provider (CSP) [2]. Technology building blocks of cloud service delivery model may also have security vulnerabilities and poor configurations that can lead to cyber-attacks on these services. Service Level Agreements (SLAs) establish some level of trust between CSP and cloud customer. However, SLAs mainly focus on quality of services parameters and do not consider adequate security metrics which can be used to measure risk. Further, SLAs does not help technically in gauging the security posture of an application in real-time. There are third-party tools and services which provide visibility on performance and uptime of CSP services, but they are more focused on the availability of the services instead of security. The decision to migrate any enterprise application to Cloud depends on the CSP's ability to protect the customer application and data. Further, when cloud customer is dealing with multiple CSP's and applications, it becomes very complex to track security control implementation status. An enterprise adopting Cloud also wants to ensure that security controls defined in the enterprise policy gets enforced while adopting the cloud services. Such capability provides them the comfort of achieving a similar level of security when the applications and data are hosted on on-premise data centers. Traditional security controls and monitoring practices are applicable if workloads are hosted in the enterprise data centers but elastic and multi-tenant environment of Cloud makes it difficult to have same level of visibility and control. Strong data privacy requirements and regulations also drives the need of having cloud customer control on the data movement. Cloud customer desires a holistic system which empowers them to define the security controls for their cloud hosted workloads, provide mechanisms to have real-time visibility of control implementation and measures the security status. Security control enforcement in Cloud follows shared responsibility model which is based on the type of cloud service. In a SaaS (Software as a Service) CSP has highest level of responsibilities as application software is also provided and managed by CSP. However, if customer is subscribed to Infrastructures as a Service (IaaS), CSP is only responsible for managing infrastructure level security. Therefore, it is also very important to determine security control implementation efficiency for CSP and CS separately. Cloud customers may have different implications of security compromise based on the business type and the data value. Ability to measure security control status on parameters of confidentiality, integrity and availability provides cloud customer much better insights on the impact of such compromise. The proposed framework provides a realistic view of security control requirement, implementation status and possible risks for a cloud hosted application. The Framework has a provision to extend the enterprise level policy to the Cloud while considering the cloud service delivery model (e.g. SaaS, PaaS, IaaS). The proposed system and method for security calculation measures security at different levels. Our framework calculates security at metrics level, control level and at overall application level. Using this system and measurement method, cloud customer can also identify the security level based on confidentiality, integrity, and availability parameters. In this work, we also considered different perspective

Manuscript published on 30 August 2019.
*****Correspondence Author(s)
Nitin Singh Chauhan*, Jawahar Lal Nehru Technological University, Kakinada, India. Email: raju.nitin@gmail.com
Dr. Ashutosh Saxena, CRRAO-AIMSCS, UoH Campus, Hyderabad, India. Email: saxenaaj@gmail.com
Dr. J.V.R Murthy, CSE Dept., Jawahar Lal Nehru Technological University, Kakinada, India. Email: mjonnalagedda@gmail.com

(i.e. Data, Network, Compute, Storage, Application, Physical), stakeholders (i.e. Cloud Provider, Cloud Customer) and security criteria (i.e. Confidentiality, Integrity, Availability) while measuring the security.

In subsequent sections of this paper, we discuss related work in section II. Section III defines Security Measurement System components, their description and activity flow. Section IV presents the methodology for calculating the security levels for an application. We consider one sample application to demonstrate the functioning of our measurement model and present related experiments and results in Section V. Section VI is the conclusion.

II. Related Work

Most of the prior art in security measurement area is around identifying and defining security controls and metrics. One of the initial works by Jelen [3] presented two possible classifications of security metrics. Hale and Gamble [4] created a framework, called SecAgreement. This framework proposes that service description and service objectives mentioned in SLA should include the security metrices. Dipankar et al. [5] designed a framework which concentrates on the tools and technologies available for cloud services to measure the security exposure and coverage. There is considerable research work available on identification and standardization of the cloud security metrics [6]. Interesting work of defining the economic security model for the cloud system using the security metrics is carried out by M Jouini et al. [7]. ENISA's report [8], presents a risk-driven approach that focuses on risk-based considerations, associates qualitative scores, a set of assets and vulnerabilities that are used to derive the Cloud specific security metrics. Grobauer et al., [9] proposes an approach to measure security level of Cloud provider through identifying the vulnerabilities. Savola et al., [10] presented threat-based taxonomy and metrics for measuring the security, privacy and trustworthiness of the cloud service. In direction to achieve the transparency, Trapero at al. [11] presented a solution that monitor and enforce the fulfillment of cloud security SLAs. Leading work in the direction of Cloud security metrics framework establishment is going on by the Cloud Security Alliance's (CSA) [12]. The CSA proposes the security control requirements for the Cloud in form of the Cloud Controls Matrix (CSA CCM). CSA also issued security questionnaire that helps Cloud customers and vendors in evaluating security risk of CSP services. NIST has also published the guidance on cloud computing service metrics around the quality of service, availability and reliability [13]. Syed rizavi et al. [14] proposed a framework which assesses the security of CSPs based on the requirements and preferences of cloud customer.

Mentioned prior work is useful in identifying and defining security controls and metrics. However, we did not find any framework or system which can utilize these controls and metrics definitions, collect information on control implementation, and use them to measure the security index of a cloud hosted application. There are tools used by CSP's to monitor the performance and security of cloud infrastructure. However, they provide visibility to only Cloud providers. Customers may not have dedicated view of his own workload tenant in such shared infrastructure. Some research on public auditability is centered around data storage and integrity issue only [15]. The framework presented in this paper considers all the key aspects of security and does not confine to any specific technology.

III. Security Measurement System

In this section, we present a system having multiple components which provides options to define security controls and collect necessary information about control implementation status in the cloud environment. This system processes the information and measures overall security of Cloud hosted application. Section A describes component details and Section B covers workflow among these components.

A) Security Measurement System Components

Figure-1 displays the key components of SMS and their interaction with other systems. SMS has two key components- 1) Security Measurement Machine (SMM) which is an off-cloud component and 2) Cloud Agent Server (CAS) which is an on-cloud component.

SMM enables cloud customer to define applicable security controls and metrics for each control for the specific application. This system communicates with CSP and external systems to get relevant information on control implementation status.

Security Measurement Machine (SMM) has the following subcomponents:

Application and Cloud Provider Management (ACM): This module provides an interface for managing the list of cloud hosted applications. Cloud customer can add, delete and modify the application details. This module also manages the list of cloud providers used by cloud customer in their environment.

Control and Metrics Management (CMM): This module manages security control information and related metrics for an application.

CMM can connect to enterprise system or GRC (Governance, Risk and Compliance) tool which is hosting the security control details. However, this system has provision to manage (add/delete/modify) the list of controls through the manual interface as well. Each control may carry a different weight factor.

Retrieval Number: J11250881019/19©BEIESP
DOI: 10.35940/ijitee.J1125.0881019
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

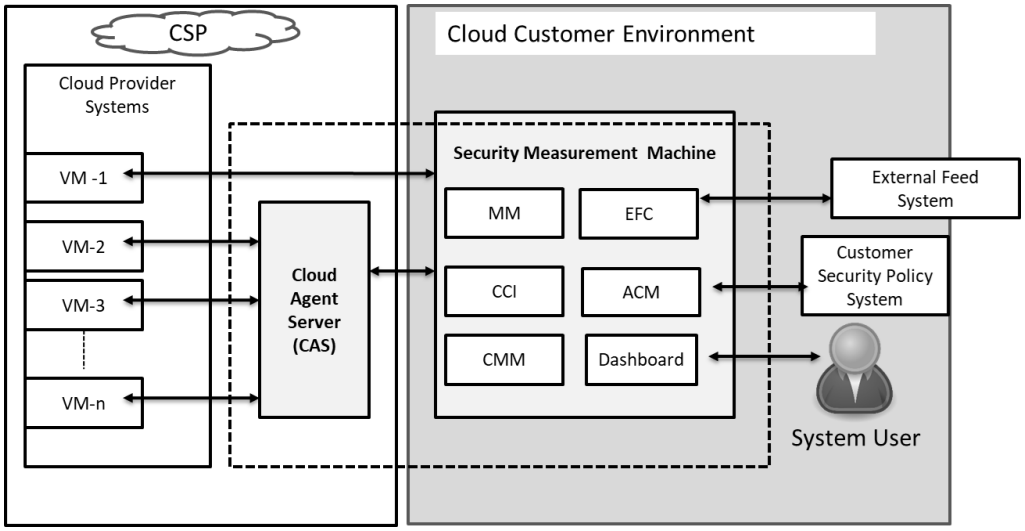

Figure 1 Diagram showing the components and interfaces of Security Measurement System

Therefore, this module has provision to map weight factors against the security controls depending on the control criticality. Information of security control, related metrics and corresponding weightages are encapsulated for an application as a security profile object in the XML or JSON format and stored in the system. Manual feed option is also available on the system to include the security control implementation status generated through independent audits and assessments.

Cloud Connect Interface (CCI): To gather information from the cloud environment, we developed many scripts which connect to the hosting environment and gather information of controls implementation status and related metric values. These scripts are managed through CCI.

External Feed Connect (EFC): There are many third-party threat management systems and reputation services which provide feeds about the security and performance information about cloud services. This module connects such services and collects information which can impact security controls and metrics related to CSP environment.

Measurement Machine (MM): Controls and metrics information gathered through various input sources require processing to calculate the final security control implementation status for an application in the cloud. Measurement machine is the computational engine that computes the gathered information to generate the final security level of an application. Various logics are built to calculate the information from different perspectives and user requirements. MM generates the security level of an application for different security criteria (e.g. Confidentiality, Integrity, Availability), security perspectives (e.g. Data, Compute, Storage, Application, Network and Physical) Cloud provider type (IaaS, SaaS, PaaS), Stakeholder type (CSP, Cloud Customer) and control criticalities (e.g. High, Medium, Low).

Dashboard Module (DM): Dashboard interface retrieves calculated security levels from Measurement Machine and displays in multiple formats based on viewing requirements and defined roles.

Cloud Agent Server (CAS): Cloud Agent Server is an on-cloud component of SMS. CAS is hosted in Cloud service tenant environment and acts like a proxy between the customer hosted system component of SMS and cloud workload. CAS has two implementation approaches. One is the agent-based approach in which small agent is deployed on cloud machines which capture the security metrics and control information from cloud machines associated with the application. CAS manages agent deployment and collects information from agents. The second approach is gathering control information through remote scripts hosted on CAS. The implementation depends on the cloud services models (e.g. IaaS, PaaS, SaaS). In IaaS customer has the opportunity to install and manage the applications. Therefore, they can install the agents on the machines hosted in the cloud. For SaaS application, CSP will usually not allow such agents to be installed on the servers.

B) System Activity Flow

Step 1. In this step we create security profile for an application by mapping security controls. Each control may have multiple associated metrics. Metrics are also mapped to the respective controls. Controls and associated metrics applicable for an application defines the security profile. This profile is generated using the CCM module and stored in XML format. Step 2. In this step, we identify the systems which will provide the information related to security control and metric values. We develop queries and scripts which will connect to theses identified system and fetch the data. CCI module provides the interface to feed probe queries and scripts. Step 3. System gathers the security metrics values and control implementation status information. Measurement engine uses the measurement model as described in section IV and processes the gathered information for arriving at the value of security index for a cloud hosted application. System collects information through cloud systems directly or using the CAS depending on the probe condition and access permissions on the cloud environment.

If security index calculation is dependent on other external parameters or manual feed, such data values are captured through EFC.

Step 4. Measurement Machine (MM) is the core engine that parses the collected information and perform necessary calculations as defined in measurement model. This process generates the security levels for individual controls and subsequently overall security level for an application having those controls.

Step 5. Once MM generates the security level for implemented controls and overall application, this information can be accessed from dashboard for viewing. Based on the roles and interest, technical teams and management can have different views and level of details on the dashboard. This dashboard provides users customized perspectives with graphical information based on their viewing needs.

The Security Measurement System presented in this section has detailed functionalities and features to measure security index for a cloud hosted application. However, core engine of this system is Measurement Machine (MM) which perform calculation for security metrics. This Machine requires a comprehensive measurement model which we discuss in the next section.

IV. MEASUREMENT MODEL

To measure the security posture of an application, we developed the Cloud Security Measurement Model (CSMM). This measurement model provides implementation guidelines for the Security Measurement System explained in the previous section. The model is flexible to adopt the control categories, security perspectives, cloud service models criticality criteria, and weight factors. Model presents calculation methods for security levels to be consumed in the Measurement Machine (MM). It also suggests control categorization and metrics to be considered for ACM. Security measurement model considers the following key aspects:

Security Perspective: We associate each control for a cloud application with a specific security area. The security areas proposed are Data, Compute, Storage, Application, Network and Physical. The advantage of this approach is that it permits to evaluate the security of the information system with these different perspectives and identify security gaps between these areas of control.

Ownership of security controls depend on the Cloud Service Model (e.g. SaaS, PaaS, IaaS). We define security control ownership for CSPs and Cloud customers. We also calculate the security level for each stakeholder separately to highlight the opportunity of improvement with respect to security control implementation. However, these are guiding principles and can be customized as per organization requirements. Cloud Security Alliance (CSA) has also listed applicable metrics for these controls.

Criticality: Each enterprise entity has different objectives and it operates in the environment having different threats and vulnerabilities. Therefore, it is important to understand the environment and identify the criticality aspect of each control based on the identified risk level. We consider three levels of criticality-High, Medium and Low and assign weight factors as High =5, Medium =3 and Low=1. Using the criticality weight factors, we generate the security level as follows:

$$SL_C = (wf_{hc} * \%\, IC_{hc}) + (wf_{mc} * \%\, IC_{mc}) + (wf_{lc} * \%\, IC_{lc}) \quad (1)$$

where, IC_{hc} = High Criticality Implemented Controls, IC_{mc} = Medium Criticality Implemented Controls, IC_{lc} = Low Criticality Implemented Controls, wf_{hc}= High Criticality Weight Factor, wf_{mc}= Medium Criticality Weight Factor, wf_{lc} = Low Criticality Weight Factor.

Stakeholder: The stakeholder field establishes the primary responsibility to ensure the appropriate action to address the control requirement. In the cloud context, Cloud Service Provider (CSP) and Cloud Customer (CS) have responsibilities of implementing and monitoring the security control. Responsibilities vary based on the cloud service and deployment model.

Security Criteria: Security Criteria refers to the fundamental aspects of Information Security, namely Confidentiality, Integrity and Availability. Each security control should address one or more of these aspects. We consider assigning different weight to each security aspect while calculating the overall security index. In some scenario's confidentiality is more important compare to integrity and availability while in some other scenario's integrity becomes the vital requirement. Therefore, each control has different security implication with respect to confidentiality, integrity and availability. Here as a sample we have assigned each criteria a different weight factor on scale of 10. Below is the formula which calculates the security level considering the security criteria weight factors.

$$CAS_{SL} = (wf_c * SL_c) + (wf_i * SL_i) + (wf_a * SL_a) \quad (2)$$

Where, CAS_{SL} = Cloud Application Security Level, SL_c = Security Level for Confidentiality, SL_i = Security Level for Integrity, SL_a= Security Level for Availability, wf_c=Confidentiality Weight Factor, wf_i= Integrity Weight Factor, wf_a= Availability Weight Factor

Cloud Security Efficiency Metrics: In the framework, there are some controls for which implementation status will be a Boolean value (Yes or No). While others will have related security efficiency metrics which define the degree of control implementation. To calculate the efficiency, we consider the maximum possible value of the identified metrics. We also gather the actual value of the identified metrics from Cloud environment. With these two values we obtain the efficiency of security metrics. To understand better, we consider an example. Let us assume if metric name is -" % Server Patched with latest OS updates". To arrive at the metrics value let us assume 80 servers are patched while 20 servers still does not have latest patches updated. In this case we can say the efficiency of cloud security metrics is 80/100. This is defined as ratio of measured and maximum values for the metrics.

Retrieval Number: J11250881019/19©BEIESP
DOI: 10.35940/ijitee.J1125.0881019
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

$$CSEMV = \frac{Measured\ metrics\ value}{Maximum\ metrics\ value}$$

Where,

$$0 \leq CSEMV \leq 1$$

We can add up the values of all the CSEM for identified metrics and arrive on total CSEM:

$$TCSEM = \sum CSEMV \quad (3)$$

Similarly, for calculating the security level, we consider number of implemented control and total controls. Ratio of these number along with weight factors for criticality (e.g. high, medium, low) provide the security index of a cloud application for different perspectives of confidentiality, integrity and availability. We use these formulas in Measurement Machine (MM) to calculate the security level.

The corresponding security levels are calculated as follows:

$$SL_C = wf_{hc}[\{(\%\ IC_{hcp}) + (\%\ IC_{hcc})\}/2] + wf_{mc}[\{(\%\ IC_{mcp}) + (\%\ IC_{mcc})\}/2] + wf_{lc}[\{(\%\ IC_{lcp}) + (\%\ IC_{lcc})\}/2] \quad (4)$$

$$SL_I = wf_{hc}[\{(\%\ IC_{hip}) + (\%\ IC_{hic})\}/2] + wf_{mc}[\{(\%\ IC_{mip}) + (\%\ IC_{mic})\}/2] + wf_{lc}[\{(\%\ IC_{lip}) + (\%\ IC_{lic})\}/2] \quad (5)$$

$$SL_A = wf_{hc}[\{(\%\ IC_{hap}) + (\%\ IC_{hac})\}/2] + wf_{mc}[\{(\%\ IC_{map}) + (\%\ IC_{mac})\}/2] + wf_{lc}[\{(\%\ IC_{lap}) + (\%\ IC_{lac})\}/2] \quad (6)$$

IC_{hcp}-High criticality Provider Implemented Confidentiality Controls, IC_{hcc} --High criticality Customer Implemented Confidentiality Controls, IC_{mcp} - Medium criticality Provider Implemented Confidentiality Controls, IC_{mcc} -Medium criticality Customer Implemented Confidentiality Controls, IC_{lcp} - Low criticality Provider Implemented Confidentiality Controls , IC_{lcp} - Low criticality Customer Implemented Confidentiality Controls , IC_{hip} - High criticality Provider Implemented Integrity Controls, IC_{hic} -High criticality Customer Implemented Integrity Controls, IC_{mip} - Medium criticality Provider Implemented Integrity Controls , IC_{mic}- Medium criticality Customer Implemented Integrity Controls, IC_{lip} - Low criticality Provider Implemented Integrity Controls, IC_{lic} -Low criticality Customer Implemented Integrity Controls, IC_{hap}- High criticality Provider Implemented Availability Controls, IC_{hac} - High criticality Customer Implemented Availability Controls, IC_{map}- Medium criticality Provider Implemented Availability Controls, IC_{mac} - Medium criticality Customer Implemented Availability Controls, IC_{lap} Low criticality Provider Implemented Availability Controls, IC_{lac} - Low criticality Customer Implemented Availability Controls

V. Experiment And Results

To demonstrate the functioning of measurement model, we have considered one web-based application hosted in the Infrastructure as a Service (IaaS) cloud environment. We evaluated the security controls and current security level of application. For security calculation purpose we considered three levels of control criticality (i.e. High, Medium and Low), three priorities for information criteria (Confidentiality, Integrity and Availability) and six different Cloud security perspective (Data, Compute, Storage, Network, Application, Physical) and two stakeholders (Cloud Provider and Cloud Customer).

Input: Suppose scalar values for criticality, information criteria and impact levels are as follows: We have assumed Criticality Weight Factor is High-5, Medium -3, Low-1 and Security Criteria Weight Factor as Confidentiality-4, Integrity-5 and Availability-6.

Table I displays the number of applicable and identified controls are captured in sample application. Control types (i.e. Data, Compute, Storage etc.) are presented as rows and each column represents the information criteria (i.e. CIA). Columns are further categorized based on criticality (H-High, M-Medium, L-Low) and responsible stakeholder (CSP – cloud Service Provider and CS- cloud Customer).

These controls are listed from the application security profile which is a database of applicable security control and metrics for the cloud application under consideration.

After evaluating the implementation status, we come up with a table II, which reflects the status of implemented controls.

Table- I: Number of applicable controls

Perspective	Applicable Controls																	
	Confidentiality						Integrity						Availability					
	CSP			CC			CSP			CC			CSP			CC		
	H	M	L	H	M	L	H	M	L	H	M	L	H	M	L	H	M	L
Data	2	2	1	3	3	0	2	3	1	3	3	0	1	2	1	1	1	0
Compute	2	2	1	2	3	1	3	2	1	1	3	1	2	2	1	1	0	1
Storage	2	1	1	1	1	1	2	1	1	0	2	1	3	1	1	1	0	1
Network	3	1	0	1	2	0	3	1	0	1	2	0	3	2	0	1	1	0
Application	0	1	0	3	2	1	0	1	0	3	2	1	1	0	0	4	3	0
Physical	3	0	2	1	0	1	3	0	2	1	0	1	3	0	2	1	0	1

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Total	12	7	5	11	11	4	13	8	5	9	12	4	13	7	5	9	5	3

Table -II: Number of implemented controls

Perspective	Implemented Controls																	
	Confidentiality						Integrity						Availability					
	CSP			CC			CSP			CC			CSP			CC		
	H	M	L	H	M	L	H	M	L	H	M	L	H	M	L	H	M	L
Data	2	2	1	1.8	2	0	2	1	1	1.7	3	0	1	2	1	1	1	0
Compute	1.4	2	1	2	1	1	2.5	2	1	1	1	1	1	1	1	1	0	1
Storage	1	1	1	0	1	1	2	1	1	0	2	1	1	0	0	0	0	1
Network	2.5	1	0	0	1	0	1	1	0	1	2	0	1	1	0	1	0	0
Application	0	1	0	0	2	1	0	1	0	3	2	1	1	0	0	1	1	0
Physical	2	0	1	1	0	0	2	0	2	1	0	1	2	0	2	1	0	1
Total	8.9	7	4	4.8	7	3	9.5	6	5	7.7	10	4	7	4	4	5	2	3

As mentioned earlier, there are two categories of controls i.e. Control A and Control B. Cloud Security Efficiency Metrics Value (CSEMV) is calculated for category "B" controls in form of percentage implementation and added to the count of category "A" implemented control. For example, this sample application has two high priority Category-B controls for CSP, one in the compute area and another in the network area. CSEMV value for compute area control is 0.4, which indicates that the effectiveness of one control implementation is 40%. This value is added to count of category- A controls to arrive on final implemented controls count. Table-III presents the security level numbers on scale of 0 to 1 where "1" indicates the 100% implementation of desired control. Here, we present the security control implementation status for Cloud Service Provider (CSP). Similarly, we can calculate the implementation status of security controls for the responsibilities owned by Cloud Customer (CS). Overall security level for an application can be calculated by averaging the CSP and CS security levels.

Table- III: Security level values for CSP on Scale of 0 to 1

Perspective	Security Level for CSP		
	Confidentiality	Integrity	Availability
Data	1.000	0.778	1.000
Compute	0.833	0.444	0.278
Storage	0.722	1.000	0.185
Network	0.907	0.630	0.463
Application	0.667	1.000	1.000
Physical	0.759	0.815	0.630
Total	0.787	0.767	0.536

The overall security of application considering the weight factors for security criteria is calculated as $CAS_{SL} = 0.70858$ (on the scale of 0 to 1).

Figure 2 presents a graphical representation of security level corresponding to each security perspective and security criteria.

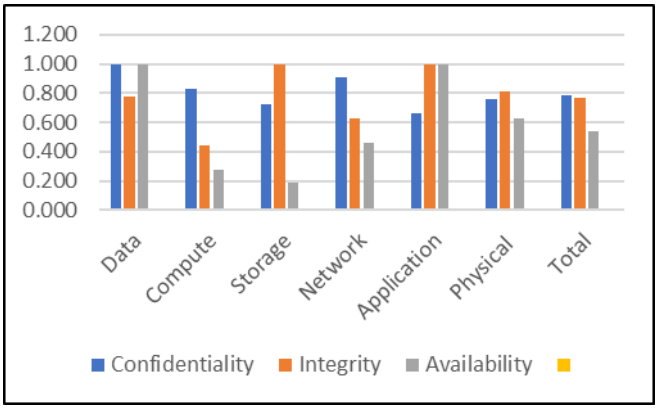

Figure 2 Security level for application from different perspectives

VI. CONCLUSION

Enterprises are finding value in Cloud adoption by increased efficiency through faster IT resource provisioning and cost effectiveness. However, limited visibility and control on their own application remain the key concerns for the Cloud adoption. Ensuring security of an application in the cloud environment is shared responsibility between Cloud Service Provider (CSP) and their customers. Cloud customer desire to have system which can provide them 360-degree view on the security status of their cloud hosted workloads. In this paper, we presented measurement system and method which provide a holistic approach of measuring the security of cloud hosted applications. System provides a consolidated view on security posture and identifies the specific vulnerable areas and responsible stakeholders based on quantifiable security scores. Security dashboard generated through this framework can help decision makers to take informed decisions about their cloud and security implementation strategy. As the technology evolves, new hosting and computational model will emerge. However, the basic principle of security will remain same i.e. CIA, and hence our framework will be applicable in the new era as well.

Retrieval Number: J11250881019/19©BEIESP
DOI: 10.35940/ijitee.J1125.0881019
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

REFERENCES

1. Ristenpart, T., Tromer, E., Shacham, H. and Savage, S., 2009, November. Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds. In *Proceedings of the 16th ACM conference on Computer and communications security* (pp. 199-212). ACM
2. Hallappanavar, V.L. and Birje, M.N., 2019. Trust Management in Cloud Computing. In Cloud Security: Concepts, Methodologies, Tools, and Applications (pp. 1686-1711). IGI Global.
3. Jelen, G., 2000, June. SSE-CMM security metrics. In NIST and CSSPAB Workshop.
4. Hale, M.L. and Gamble, R., 2012, June. Secagreement: Advancing security risk calculations in cloud services. In Services (SERVICES), 2012 IEEE Eighth World Congress on (pp. 133-140). IEEE
5. Dasgupta, D. and Rahman, M.M., 2011, October. Estimating security coverage for cloud services. In Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on (pp. 1064-1071). IEEE.
6. Saripalli, P. and Walters, B., 2010, July. Quirc: A quantitative impact and risk assessment framework for cloud security. In Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on (pp. 280-288). IEEE.
7. Jouini, M., Aissa, A.B., Rabai, L.B.A. and Mili, A., 2012. Towards quantitative measures of Information Security: A Cloud Computing case study. International Journal of Cyber-Security and Digital Forensics (IJCSDF), 1(3), pp.248-262.
8. Trimintzios, P., 2011. Survey on Resilience Metrics. European Network and Information Security Agency (ENISA).
9. Grobauer, B., Walloschek, T. and Stocker, E., 2011. Understanding cloud computing vulnerabilities. IEEE Security & Privacy, 9(2), pp.50-57.
10. Savola, R.M., 2007, October. Towards a taxonomy for information security metrics. In Proceedings of the 2007 ACM workshop on Quality of protection (pp. 28-30). ACM.
11. Trapero, R., Modic, J., Stopar, M., Taha, A. and Suri, N., 2016. A novel approach to manage cloud security SLA incidents. Future Generation Computer Systems.
12. Cloud controls matrix v3.0.1 (10-6-16 update): Cloud security alliance (2009) Available at: https://cloudsecurityalliance.org/download/cloud-controls-matrix-v3-0-1/ (Accessed: 3 March 2017).
13. de Vaulx, F.J., Simmon, E.D. and Bohn, R.B., 2018. Cloud computing service metrics description (No. Special Publication (NIST SP)-500-307).
14. Rizvi, S., Ryoo, J., Kissell, J., Aiken, W. and Liu, Y., 2018. A security evaluation framework for cloud security auditing. The Journal of Supercomputing, 74(11), pp.5774-5796.
15. Hsien, W.F., Yang, C.C. and Hwang, M.S., 2016. A Survey of Public Auditing for Secure Data Storage in Cloud Computing. IJ Network Security, 18(1), pp.133-142.

AUTHORS PROFILE

Nitin Singh Chauhan obtained Master of Computer Applications (MCA) degree from Jai Narain Vyas University, Jodhpur, India in 2000. He started his professional career as Project Executive with Institute for Development and Research in Banking Technology (Established by RBI) Hyderabad India (2001-2005). He worked for AppLabs (2005-2006) as a Security Lead and at Genpact (2006-2008) as an Assistant Manager- Information Security. He was Senior Technology Architect at Infosys Limited (2008-2016). He is a Research Scholar at Jawahar Lal Nehru University, Kakinada. He has 4 granted patents and around 10 published research papers to his credit. Nitin holds multiple certifications including CISSP, CISA.

Ashutosh Saxena is M.Sc. M.Tech. and Ph.D. in Computer Science (1999). He has over two decades of industry and academic experience. He has authored/co-authored 90+ publications, 26 US patents, and a book on PKI (published by Tata McGraw Hill). He worked as Associate Professor at IDRBT (established by RBI), from 1998-2006. Worked at Infosys Ltd. (2006-16) as Principal Research Scientist & AVP. He is member of the review board for many international journals, conferences and committees. He served as Adjunct Faculty at NIT Warangal & Professor and Dean R&D at CMR Technical Campus. Presently, he is Professor (CS) at CRRAO-AIMSCS, UoH Campus, Hyderabad.

Dr. J.V.R.Murthy is B.Tech, M.Tech, and Ph.D. He is Professor (Computer Science and Engineering) at Jawahar Lal Nehru Technology University Kakinada (JNTUK). He also holds position of Director, In-charge of Incubation Center JNTU Kakinada. He has 33 years of academic, Industrial and research experience in the Computer Science field.
He has published more than Sixty-five research papers in international Journals/Conferences including IEEE and Elsevier sciences. Thirteen scholars were awarded Ph.D. under his guidance. He was appointed as Independent Director, Kakinada Smart City Corporation Ltd., and Chairmen Nominations and Remunerations Committee by Ministry of Urban Development Government of India in January 2018. He is also recipient of Obama-Singh Initiative grant in collaboration with Chicago State University USA.