A Policy-based Security Framework for Storage and Computation on Enterprise Data in the Cloud

Sourya Joyee De
Management Information Systems Group
Indian Institute of Management Calcutta
sjoyeede@gmail.com

Asim K. Pal
Management Information Systems Group
Indian Institute of Management Calcutta
asim@iimcal.ac.in

Abstract

A whole range of security concerns that can act as barriers to the adoption of cloud computing have been identified by researchers over the last few years. While outsourcing its business-critical data and computations to the cloud, an enterprise loses control over them. How should the organization decide what security measures to apply to protect its data and computations that have different security requirements from a Cloud Service Provider (CSP) with an unknown level of corruption? The answer to this question relies on the organization's perception about the CSP's trustworthiness and the security requirements of its data. This paper proposes a decentralized, dynamic and evolving policy-based security framework that helps an organization to derive such perceptions from knowledgeable and trusted employee roles and based on that, choose the most relevant security policy specifying the security measures necessary for outsourcing data and computations to the cloud. The organizational perception is built through direct user participation and is allowed to evolve over time.

1. Introduction

"Despite of all the hype surrounding the cloud, enterprise customers are still reluctant to deploy their business in the cloud. Security is one of the major issues which reduces the growth of cloud computing and complications with data privacy and data protection continue to plague the market."[27] An organization possesses various types of data that have a wide range of sensitivity. Parts of these data (such as customer data, engineering designs etc.) are business-critical for which confidentiality, integrity and availability could be very important for the survival or growth of the organization. Employees need to access data of different sensitivity according to their roles in the organization. The users of enterprise data are not restricted to employees of the organization in question, it could be any other individuals like ordinary customers, or employees belonging to other organizations as clients, suppliers or partners. With the advent of cloud computing, an organization (or enterprise) often faces the question of whether to outsource all these data and computations to what is known as a public cloud. It has several technological, organizational and environmental factors to consider [18]. Cloud computing research shows that security is one of the most important technological factors that inhibit cloud adoption. It includes concerns about loss of control over data, dissolution of the concept of perimeter security, trustworthiness of CSPs (Cloud Service Providers), data confidentiality, integrity, data and service availability, software vulnerabilities, legal and trans-border issues about data location and data privacy etc. ([13], [14], [21], [22]).

Although researchers have pointed out several cloud security concerns and proposed solutions for many of them ([2], [3], [5], [15]-[17], [24], [26], [28]-[32]) especially for the case of cloud storage, there has hardly been any holistic approach that can help an organization to take decisions about which data or computation to outsource based on sensitivity or security requirements. And how to do so securely based on its perception (this can be the perception held by the organization as an entity or held by the users using the data in question) about data sensitivity and the trustworthiness of the CSP. In cloud storage and computation security literature, how much a CSP is trusted is reflected in the adversarial models assumed for the CSP when a secure data storage and computation method is proposed ([4], [7], [19], [28]-[31]). When data and computations are outsourced to the cloud, the organization confers a certain degree of trust on the CSP to take proper security measures to protect its data and applications from external as well as from insider attacks. Although the organization can sign security SLAs with the CSPs, monitoring whether

978-1-4799-2504-9/14 $31.00 © 2014 IEEE
DOI 10.1109/HICSS.2014.613

these are being properly implemented is yet another task the organization has to perform. Sometimes, it is not even clear who should perform this monitoring activity, the CSP, the organization or a trusted third party [6]. Therefore, organizations must build their own perception about how the CSP will behave i.e. to what extent it can be trusted with different items of data and computations. This will help building policies to retain control over data and computations outsourced to the cloud.

Under such circumstances, organizations, especially those who can invest little in IT security expertise, may be prone to take an overall optimistic or pessimistic view about the trustworthiness of a CSP. This, in turn, will cause it to implement security policies for data outsourcing and computation that are either too lax or too strict, making it either insecure or inefficient, respectively. It may be performing too many costly activities (such as encryption, decryption, huge data upload/download during computations etc) that may offset the benefit of using the cloud in the first place. Similarly, it may be performing too few security activities putting its data at risk. This is especially true for an organization that has just begun to use the cloud computing technology.

In the current situation the users of the enterprise data (who often cut across the globe much beyond even the traditional boundary of the organization, belong to different cultures and different legal systems, have different interests and capabilities), e.g. supply chain for a large manufacturing firm, require to develop their understanding of the sensitivity and criticality of the applications and data they handle also their perceptions about trustworthiness of the CSP based on their own interactions with the cloud while performing their job on the cloud. We are interested in a mechanism of developing a trust based system through direct user participation and learning which continuously evolves based on user experience and changing scenario in the business or cloud environments. And this in turn will be used to create a policy based security system which is both efficient and flexible while minimizing risk by choosing the most relevant security policy that specifies the security measures necessary for outsourcing data and computations to the cloud. This decentralized approach is supported by [10] which refers to people centric security (PCS) of Scholtz at Gartner, that suggests "empower users with responsibility for systems and data important to their work, sprinkle in consequences for breaching that responsibility and users will do the right things to secure their environment … The current approach in developing policies and controls doesn't scale to current realities … the convergence of social, mobile, cloud and big data and the changes it brings to enterprise computing. The forces are eroding corporate boundaries and controls in many areas long thought to be state-of-the-art defenses". On a similar note, [25] reported about two studies indicating that user participation contributes to improved security control performances through better awareness and alignment between IS security risk management and the business environment and improved control development. It further noted that while the IS security literature often considered users as the weak link in security, according to the studies, users may be an important resource to IS security by providing required business knowledge contributing to more effective security measures. Further, user participation is "also a means to engage users in protecting sensitive information in their business processes" [25].

We propose a solution to this problem which is dynamic, evolving following a decentralized approach for secure outsourcing to the cloud. Initially, the inexperienced organization may decide to start with an organization-wide, centrally-decided, uniform pessimistic or optimistic view about the trustworthiness of the CSP depending on various factors such as the CSP's reputation, cost etc. However, as individual users gain experience through security trainings and direct usage of cloud applications, the organization can begin to move away from this uniform view to have a varied, decentralized outlook about the trustworthiness of the CSP. Decentralization is achieved by taking into account the perception of employee roles (such as employees in data critical positions of the organization) who constantly deal with certain types of data or who understands the importance or values attached to data elements. For example, the CFO of an organization may be the best person to tell how sensitive financial data of the organization is and can help to decide the security requirements of such data. Initially, he, being inexperienced, may still take a pessimistic view while the CISO will have an optimistic view. As he gains more experience interacting with cloud-based applications and after being exposed to information security trainings, the CFO may revise his perception about the trustworthiness of the CSP. This gradual shift from an extreme, uniform view (pessimistic/ optimistic) to a mixed view (pessimistic for some data elements, optimistic for others) is a result of decentralization and helps in optimizing security and efficiency. The additional benefit of this approach is that users are likely to gain a better understanding of the technologies they use, become aware of the security

issues associated with such technologies and as a result are likely to Cloud Security Support System has been advocated in this respect. There are other security issues related to other activities of the cloud (such as data movement, maintenance etc) which are not in the scope of our paper.

In section 2 we summarize the contributions, in section 3 we discuss related works in secure storage and computation outsourcing in the cloud and review the different adversarial models that have been considered in the past. Next, in section 4 we present the proposed policy based security framework consisting of three layers - each subsection deals with a different layer of the framework. In section 5 we suggest an organizational implementation of the framework and finally, we conclude in section 6, followed by acknowledgement and references.

2. Our Contributions

In this paper we propose a decentralized, dynamic and evolving policy-based security framework for enterprise data and computation outsourcing to the cloud such that it allows an organization to retain control over its data and computations while being both efficient and flexible. Here, we attempt to address this issue comprehensively. We emphasize that storage and computation needs to be addressed separately, yet in an integrated manner. We elaborate on the set of policies which we call the secure data policies consisting of storage security policies, upload security policies and computation security policies to guide the organization in finding out the right security level for each combination of data security requirement and perceived adversarial behavior of storage and computation nodes (VMs) of the CSP. The framework develops a people centric highly evolving and dynamic organizational view of the outsourcing operation of the enterprise data vis-à-vis the cloud. Further, the framework is based on the principle of risk minimization while optimizing efficiency and flexibility. We discuss the building blocks such as how the user at the individual level as well as the enterprise at the organization level express their data security requirements and CSP trustworthiness based on which we arrived at the security policies. Lastly, we suggest a possible organizational implementation of the above security framework.

3. Related Work

3.1. Secure Storage and Computation in Cloud

Several recent works in cloud computing focus on storage and computation security. [24] proposes a system architecture allowing organization-wide integration of untrusted public storage cloud. The architecture guarantees confidentiality, availability and integrity while requiring only a minimum level of trust on the cloud. It uses Information Dispersal Algorithms (IDA) to ensure availability, and by combining symmetric encryption with IDA, achieves high confidentiality. Integrity is ensured by using AES-CMAC operation mode for encryption which produces a MAC for each data fragment and enables replacement in case of any integrity violation. [26] presents a similar, advanced architecture where the end-devices inside an organization are considered to be within a Personal Secure Cloud or π-Cloud controlled by the π-Box that acts as an intermediary between the π-Cloud and the external cloud. π-Box performs all security operations for data storage and distribution such as information dispersal, encryption, checksum etc. Data is first dispersed using an IDA, encrypted and signed and then the shares are distributed to multiple clouds. When the user inside the organization needs to access data, the shares are fetched from the multiple clouds and the data is reconstructed if enough shares could be withdrawn.

[16] proposes a cryptographic cloud storage service consisting of the following components: 1) a data processor that processes data before being sent to the cloud; 2) a data verifier that verifies whether data stored in the cloud has been tampered with; 3) a token generator that generates token to enable the CSP to retrieve customer data segments and 4) a credential generator that implements access control policy by issuing credentials to various parties in the system. It allows integrity, confidentiality as well as secure data erasure. The authors suggest the use of searchable encryption to enable confidentiality and retrieval of data based on keywords and attribute-based encryption to enable implementation of credentials and proof of storage to verify integrity. [17] suggests a general-purpose protocol for securely computing any function in the cloud without revealing any information about the input or output by using multiple VMs. The usage of principles of secure multi-party computation (SMC) ensures that if at least a single VM is honest, no information is revealed. In our work, we also use similar methods, derived from the literature of secure multi-party computation. [5] has proposed the Twin Cloud architecture for securely outsourcing data and arbitrary computations to the cloud. It consists of the usage of two types of clouds, the trusted cloud (such as a private cloud) which performs all security-critical operations such as encryption, decryption etc.

and the untrusted commodity cloud which performs all performance-critical operations on encrypted data. The trusted cloud has a limited storage and computation resources whereas the commodity cloud has large amount of resources. Authenticated encryption or symmetric encryption along with MACs is used for ensuring confidentiality and integrity while the concept of Garbled Circuits is used for secure computation. [29] classifies cloud data security into cloud storage security concerned with integrity of data stored in untrusted cloud and cloud computation security concerned with correctness of outputs of computations outsourced to untrusted cloud. They propose the SecCloud framework consisting of an auditing scheme based on probabilistic sampling technique and designated verifier signature. The work of [28] has only concentrated on particular issues like integrity and have suggested a flexible, distributed storage integrity auditing mechanism based on homomorphic token and distributed erasure-coding. [2], [3] and [15] have focused on how multiple clouds can be used for secure storage and computation of data in cloud. They have also suggested the use of (k, n) secret sharing method for generation of shares from the data. [20] presents a privacy model where independent and heterogeneous data centres outsource their query processing tasks to a service provider on cloud by expressing their mutual privacy requirements on components of query, data and results vis-à-vis themselves, the anonymous customer making the query and the service provider. This treats different components of storage (data in the databases, schemas, results) and computation (query, result processing) for the privacy implementation.

3.2. Adversarial Models

Previous works on secure cloud storage and computation have considered different adversarial models. [29] considers a Byzantine adversary i.e. an adversary that can behave arbitrarily to corrupt a small number of servers. In their work, a corrupted cloud can launch three types of attacks: 1) storage-cheating where corrupted servers may delete rarely accessed files to reduce storage cost or arbitrarily modify stored data; 2) computation-cheating in which the servers either generate incorrect results of computations or uses different inputs for computations to reduce computational cost and 3) privacy-cheating in which corrupted cloud server can leak user's confidential information to other parties. [4] considers that the un-trusted cloud can fail in a Byzantine way i.e. stored user data can be deleted, modified or leaked to other parties and argue that this is the most general fault model that takes into account both malicious attacks on CSPs as well as events like accidental data corruption. A set of scenarios of different trust levels assigned to cloud has been identified by [7]. According to them, a trusted cloud is one which, in the absence of unpredictable failures, serves users correctly in accordance with SLA and there are no malicious insiders. They classify un-trusted clouds into 1) data curious cloud if insiders find it beneficial to breach user data confidentiality; 2) access curious cloud in which insiders try to link user access patterns to data and find out outsourced computational logic and 3) malicious cloud which modify data and generate incorrect results of computation. In [30]'s CloudSeal that ensures end-to-end security for distributing and sharing content through the public cloud, the content provider is trusted but the CSP is assumed to be semi-honest. [28] differentiates between internal and external attacks on CSPs and provide an adversarial model that takes into account both types of attacks. According to them, a CSP can be self-interested, untrusted and malicious and thus cause internal attacks comprising movement of rarely used data for cost minimization, hiding of incidents of security breach etc. On the other hand, external, economically motivated attackers may attack CSPs by corrupting data storage servers and subsequently modify or delete user data without being detected by the CSP. In Depot, [19] assume that storage nodes are either Byzantine or correct. While Byzantine nodes can fail, corrupt data, collude, process messages incorrectly, introduce inconsistencies in data, a correct node is one that never deviates from the prescribed protocol nor remains unavailable forever.

4. Decentralized, Evolving Policy-based Security Framework

The policy-based security framework (Figure 1) that we propose represents the conversion of user's perception of CSP and data security requirements into secure data policies that guide the organization in outsourcing its data and computations. It consists of three layers: 1) Super-user perception layer; 2) Technical specifications layer and 3) Secure data policy layer. The super-user (defined below) perception layer represents the super-user perception about CSP and data security. The Technical specifications layer represents different adversarial models for CSPs and data security levels to be used to devise security algorithms for storage and computations and is derived from the super-user perception layer.

The top-most layer, the secure data policy layer specifies the type of security measures (secret-sharing, multi-party computations, signature schemes etc) to be undertaken for data storage, upload and computations for different choices of cloud adversarial models and data security levels. In the next sections we describe each layer in details. It consists of three layers: 1) Super-user perception layer, 2) Technical specifications layer and 3) Secure data policy layer. The three layers respectively user perception about CSP and data security; different adversarial models for CSPs and data security levels used to devise security algorithms for storage and computations; and the types of security measures (secret-sharing, SMC, signature schemes etc) needed to be undertaken for data storage, upload and computations for different choices of cloud adversarial models and data security levels.

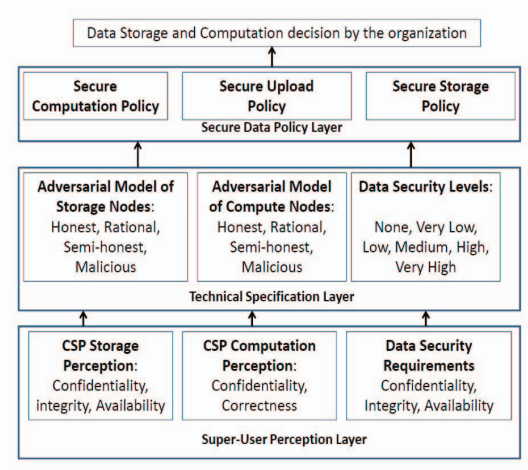

Figure 1. Policy-based Security Framework

4.1. Super-user Perception Layer

There are three basic trust relationships that form the basis of our security framework: organization vs. user; user vs. CSP and organization vs. CSP.

The organization does not trust all the users equally, e.g. people in the top positions are more trusted than others. Users are trusted with only those data and computations that are connected to the role the user is assigned. The organization vis-a-vis user trust relationship is guided by the Enterprise Data Access Policy (EDAP) matrix which tells for each user and data element pair what kind of accesses and rights are permitted. A user who is allowed to choose a security policy for data elements he has access to is called super-user for those data elements. For other data elements for which also he has access, the same user has just 'ordinary' rights over the latter, meaning thereby no power to choose a security policy for those. There may be users who do not have super user rights for any data elements. The organization bases its perception of trustworthiness of the CSP on the perceptions of super-users who in turn form their perceptions by applying some aggregation rule on the perceptions of individual users having access to corresponding data elements.

4.1.1. EDAP Matrix. Access control matrices specify access rights on objects. An object is the abstraction of resources controlled by a computer system [23]. Role based access control (RBAC) policies regulate a user's access to objects in a system based on the activities he performs on these. We present the EDAP matrix as an intermediate access control matrix derived from role-based access control policies used in the organization. It specifies user's data access rights from which one can validate the computation permitted for each role in the organization. We note that computations require different data elements as inputs and produce new data elements and / or modify existing data elements as output. Therefore, a user can perform a computation only when he has the necessary access rights to relevant input and output data elements. Roles that hold higher responsibility w.r.t. specific functions are generally assigned super-user roles for corresponding data elements. Using this capability, a user can fix the sensitivity and security requirements of data they have access to and thus later on it can help a user to form a perception about the CSP for outsourcing. Data security policies are finally selected according to these perceptions. Moreover, super-users can take opinions of other users having access to the same data elements and use overall aggregated perception for moderating his perception about CSP.

4.1.2. User Perception. The vulnerability of a CSP to various security threats is dependent on what it does to protect itself. Even if security level agreements are in place, the proper implementation of this agreement depends on the CSP.

Table 1. EDAP Matrix

User Roles ↓	Data Element 1		Data Element 2		. . .
	Data Access	Super User	Data Access	Super User	
Role 1	Read, Write	Yes	Read	No	
Role 2	Read	No	Write	No	
…					

Architectures for monitoring whether the security metrics in a security SLA is being met have been proposed, but again such monitoring architectures need to be secured, otherwise they may prove to be a weak point in the monitoring process [6]. Therefore, when data security is to be ensured, users should exercise their own idea about how sensitive their data is, how much security it requires and how vulnerable the CSP can be. Users who handle certain types of data may be the best persons to express the level of sensitivity of those data and consequently the security requirements of the data. This method enables the organization to trade-off dynamically between security and efficiency instead of compromising on any one. However, the success will depend on how effectively users can choose their perceptions. This requires providing sufficient training to users both on the technologies and the applications they handle to emphasize the relevance and importance of security.

Users may have varied expertise in judging the trustworthiness of a CSP or the sensitivity of the data. Generally users can base their judgment on their previous experiences, reputation of the CSP, security breach incidents in the news, security certifications etc. In this work, we consider that users have some level of experience/ knowledge to be able to express their perception about a certain CSP and the sensitivity of the data they work on regularly. This perception need not be very realistic. It could be pessimistic or optimistic depending on personal choice or organizational strategy. But we can reasonably assume that a user will not perceive a CSP that has suffered many security breaches and outages in recent times as trusted while it considers one untrusted which has suffered few security breaches. An organization may choose to develop and maintain a feedback mechanism to keep track on users to reward or punish users based on their activities, e.g. a super-user who constantly provides misleading perceptions can be demoted to an ordinary user. In our framework, the user can state his perception about a CSP's trustworthiness by expressing whether, according to him, the CSP will be successful in providing confidentiality, integrity and availability with respect to storage and confidentiality and correctness with respect to computations. Similarly, the user can talk about its perception about the security requirements of data by specifying whether the data needs to be confidential and whether its integrity and availability are important. For both cases, the user is guided by the Cloud Security Support System (Section 3). A typical organization (e.g. one not so IT oriented) starts with a relatively small number of super-users. As their interaction with the CSP grows, with time, more number of ordinary users becomes super users. (e.g. Initially CFO has super user rights over all financial items. Later on he delegates this power to his junior officers.) The organization learning through evolution and dynamism occurs because of three factors: business practices change, CSP performance fluctuates and leaning interaction grows over time.

Data security requirements users are concerned with are: 1) Confidentiality i.e. no data should be leaked during storage; 2) Integrity i.e. data should not be modified, deleted or fabricated while in storage and 3) Availability i.e. data should be available, whenever required, to the legitimate user. Users specify their requirements for each of these parameters. The user specification relies on the cost to the user or organization if that security parameter is violated. For example, if confidentiality of customer credit card numbers is breached from an online shop, then it will suffer severe legal repercussions as well as loss of reputation. Since loss of confidentiality is costly for this data element, the user positively selects (i.e. marks 'Yes') this parameter as a requirement. On the other hand, general information about the organization (eg., when it was established, its products or services etc) do not require confidentiality. However since an outsider can form a general impression about the organization by reading this information on its website, it may require integrity protection.

Not all combinations of security parameters are valid. For example, we cannot consider that a data element requires availability but not integrity, since availability of fabricated data does not make sense.

For storage security the user specifies his perception about the CSP in terms of its ability to maintain confidentiality, integrity and availability. Similarly for computation security, the user specifies whether he thinks that the CSP is capable of maintaining confidentiality and correctness. For example, if the user has any prior experience of data leakage of data stored with a CSP, then he may state that the CSP is unable (i.e. marks 'No') to maintain confidentiality.

4.1.3. Cloud Security Support System. Users of an organization encompass employees, suppliers, customers and any other stakeholders. Therefore, it is very natural to assume that the average user has little knowledge about cloud and its security issues thus making it difficult for them to form an opinion about the trustworthiness of a CSP. Moreover, many users may not be fully aware about the security implications of the data they handle. Their decision-making may be affected by bounded rationality,

incomplete information or deviations from rationality [1]. An immediate gain may lead them to engage in activities without thinking of their security consequences [8]. Therefore, for our framework to be successfully implemented, firstly, we require a support mechanism that will guide users in taking the right decisions with respect to their perception about CSP trustworthiness as well as data security requirements. This will make user perceptions more reliable. Secondly, users must be trained to gain an overall understanding of the cloud computing technology as well as its security vulnerabilities in order to enable them recognize and appropriately react to security incidents. Thirdly, a monitoring system is required to constantly monitor users' activities in terms of stating their perceptions and take corrective actions (such as taking away super-user status from a user) when a user is found to deviate suspiciously. Accordingly one can develop the Cloud Security Support System or CS3 with three layers: 1) User Perception Guide; 2) User Training Module and 3) User Perception Monitoring and Feedback Module. CS3 is to be used by all users (whether super user or not).

4.2. Technical Specifications Layer

Before formulating security policies, we must first map user perception about data security requirements into different security levels into which data can be classified and user perception about CSP trustworthiness into suitable adversarial models representing the behavior of CSPs.

4.2.1. Data Security Levels. Computations use one or more data elements as input and may modify/ create one or more data elements as output. So, data security can be breached during storage and/ or during computation. Therefore it is necessary to implement security for both storage and computation. However, the level of security required in each case will be determined by the security requirements of the data element itself.

Our model internally maps (Table 2) the user-specified security requirement to storage and computation security requirements. For storage, the parameters remain same and indicate the level of security to be provided while the data is stored. For computation, the security parameters used are: 1) Confidentiality i.e. no information about the data should be leaked during computation, the input output that the computation yields and in any intermediate steps and 2) Correctness of output i.e. the result of the computation should be correct (for e.g., if an application to add two numbers is run, then it should perform only that and nothing else). Since computations create new data elements or modify existing ones, a wrong output affects integrity of data elements. These parameters define the level of security required during computation so that the desired data security is maintained. Since we are going to use the concept of SMC for securing computations in the cloud, we derive the security parameters for computation from the SMC literature [11]. Sometimes guaranteed output delivery i.e. if a computation is performed then the user should get a result corresponding to that computation may be a requirement. Guaranteed output delivery may be looked upon as availability of outputs of a computation. However, we do not consider this in our work as we consider that majority of computing VMs can be dishonest and then guaranteed output delivery cannot be achieved [11].

We have not included availability of computation as it does not have direct security implication in terms of security policy formulation. It does not affect the security protocols. However, computation availability is an important issue, which affects the up time of an application. The application which runs continuously, say a stock tracker, is affected a lot due to unavailability of computation, which is primarily due to non-allocation of resources by the CSP at run time or due to failure of CSP operations. The organization may choose to maintain perception on this parameter as well, that will not really affect our framework.

We have considered only valid combinations for storage and computation security and have been guided by the following issues. First, when data confidentiality is desired, both storage and computation should be confidential. Second, when outputs of computations are viewed as stored new data or modified data, if computation results are incorrect integrity of stored data is also affected.

Table 2. Data Security Levels

Type of Security → Type of Data ↓	Storage Security			Computation Security		Security Levels
	Confidentiality	Integrity	Availability	Confidentiality	Correctness	
Public	No	No	No	No	No	None
	No	Yes	No	No	Yes	Very Low
	No	Yes	Yes	No	Yes	Low
Private	Yes	No	No	Yes	No	Medium
	Yes	Yes	No	Yes	Yes	High
Sensitive	Yes	Yes	Yes	Yes	Yes	Very High

Data which does not require confidentially we call 'public'. The non-public data which requires strictest security imposition is called 'sensitive', the other 'private'. This has been done to simplify the security requirements. Any organization may choose to apply finer classification, for example, public may be of three types, unrestricted public, public with integrity, and public with integrity and availability. But latter processes would have also to be refined accordingly.

In this context we find it relevant to mention Itani et al's [12] Privacy-as-a-Service (PasS) that, depending on sensitivity of data, enables secure storage and processing of user data in the cloud with the help of tamper-proof cryptographic co-processors. PasS allows users themselves to determine the type of privacy mechanisms they desire from the CSP based on how sensitive their data is and how much they trust the CSP with respect to the sensitivity of the data in question. Unlike [12], we do not link this classification directly to the level of trust assigned to cloud storage. Here, the adversarial nature of storage nodes are determined and linked to user's perception about the CSP in general. Therefore, if a storage node is deemed honest, any data irrespective of their sensitivity can be stored unencrypted. The strength of security mechanisms applied to each kind of data depends on the sensitivity of data as well as the adversarial nature of the storage node.

4.2.2. Adversarial Models for CSPs. We define adversarial models for storage nodes which are virtual data storages as well as for computing nodes which are VMs performing computations. An honest storage node stores all data exactly as provided by the user and data stored by such a node remains fully secure i.e. its confidentiality, integrity or availability is not affected. A semi-honest storage node does not deviate from the protocol but may passively gather information for e.g., by looking at unencrypted data it stores or by observing access patterns of encrypted data etc. These activities of a semi-honest storage node lead to break in confidentiality of data. However, such a node does not affect the integrity (say by modifying data or data shares) or availability of data (by not sending shares when protocols instruct them to do so). Semi-honest storage nodes can however collude, i.e., they can get involved in exchange of data shares or other information among themselves. A malicious storage node indulges in activities that can affect confidentiality, availability or integrity of data. Malicious storage nodes may also collude. Whereas honest and semi-honest nodes do not deviate from assured deletion, malicious nodes may not delete data even when they are instructed to do so. CSPs may often partially or fully delete data that are not used frequently to release storage space. Rational storage nodes, like honest ones, guarantee confidentiality but, unlike honest nodes, do not guarantee integrity or availability of data.

An honest computing node performs all computations without deviating from the protocol. A semi-honest computing node does not deviate from the computation but keeps copies of intermediate steps which it can later use to extract information about the data on which the computation takes place. It can also perform unauthorized computations on data leaked during computation. However, it always performs the computation correctly and delivers outputs, except for unavoidable failures. Semi-honest computing nodes do not collude. A malicious computing node deviates from the protocol, performs unauthorized computations on data leaked during computations, may not perform the computation correctly and may not even deliver output. Malicious computing nodes can collude and exchange information, data shares etc. among them. Rational compute nodes do not affect confidentiality but may not perform computations at all and hence return random results thereby affecting correctness of output. In each of the above cases we consider that the number of corrupted parties is at most c out of a total of n where $c < n/2$.

4.3. Secure Data Policy Layer

Secure data policy consists of a set of rules that can guide secure data outsourcing taking into account the adversarial models of storage nodes and data security requirements. Data can be either uploaded beforehand or uploaded as and when computations demand them, depending on the security policy. While secure data upload policy ensures that dishonest computation nodes cannot infer anything from input data that the users upload during computations, secure storage policy prevents dishonest storage nodes from extracting any information from the data stored in them as well as dishonest computation nodes from extracting any information from input data they fetch from these nodes during computations (see Table 6). Similarly, secure computation policies (Table 5) guide secure computation offloading taking into account the adversarial models of computation nodes and data security requirements.

Arbitrary computations are not possible on encrypted data. During computation, if data is decrypted then there may be data leakage during computation because of a corrupted computation node. Therefore, we use the technique of threshold

secret sharing for ensuring data confidentiality and availability while signature schemes are used for detecting data modifications. The computation nodes, depending on the level of corruption, compute on these shares. Whenever a node is found to be corrupted (eg., a computation node not providing correct results of computations or storage/ computation nodes violating integrity), the CSP is made aware of such corruptions. In addition, the super-user is also made aware of this issue so that he can take suitable measures such as updating his perception about the CSP, re-uploading data that has been modified, deleting data from corrupted nodes (for this to be possible, suitable methods for assured deletion must be implemented during data storage/ upload) etc.

Table 3. Adversarial Models for Storage Nodes

Type of Security → Adversarial behaviour ↓	Storage Security		
	Confidentiality	Integrity	Availability
Honest	Yes	Yes	Yes
Rational	Yes	Yes	No
	Yes	No	Yes
	Yes	No	No
Semi-honest	No (with collusion)	Yes	Yes
Malicious	No	Yes	No
	No	No	Yes
	No	No	No

Table 4. Adversarial Models for Computation Nodes

Type of Security → Adversarial Behaviour ↓	Computation Security	
	Confidentiality	Correctness
Honest	Yes	Yes
Rational	Yes	No
Semi-honest	No (without collusion)	Yes
Malicious	No	No

When storage nodes are considered to be of a lesser level of corruption than computing nodes, suitable modifications are required for securely storing data as per the level of corruption of the computing node. For example, the security measures to be adopted for storing data in a semi-honest storage node when the computing node is malicious are stricter than when both storage and computation nodes are semi-honest. For lack of space, we present the policies for such combinations only. Also, the security measures for computation nodes are irrespective of the corruption level of storage nodes.

For all the policies we assume that the communication channel between cloud and the organization is secure.

Table 5. Secure Computation Policies

Adversarial Model of Computation Node	Secure Computation Policy
Honest	A single VM performs computations.
Semi-honest	For other than 'None' and 'Medium' security data, a single VM is given a random number of challenge computations before the actual computation. The o/p of actual computation is assumed to be correct only when challenge computations are correctly answered. Otherwise the VM is reported to be corrupted. Computations on other data are done by a single VM.
Rational	A single VM can perform computations on public data. For private and sensitive data, VMs engage in SMC.
Malicious	A single VM can perform computations for security level 'None'. For Low and Very Low security data, a single VM is given the Challenge. For other data, VMs engage in SMC.

5. Organizational Implementation

In this section we propose an implementation model (Figure 2) of our policy-based security framework. Within its security perimeter, the enterprise must run an application which we call the Enterprise Data Controller (EDC), a software implementation of our security framework. The organization may choose to run this application on a secure, private cloud. EDC takes the responsibility of controlling all kinds of data access, storage, movement and computations in the public cloud. Specifically, its tasks can be classified into user-facing tasks and policy-facing tasks. Under user facing tasks, it interacts with the user directly for 1) controlling users' access on data and computations using EDAP; 2) collecting user perception about data security requirements and CSP trustworthiness, again guided by EDAP and 3) accepting data for upload, storage and computation in the cloud. Under policy facing tasks, the control cloud does the following: 1) interpretation of user perception about data security requirements and CSP trustworthiness according to our security framework; 2) actual data upload operations before or during computations as per secure storage and upload policies and 3) computation logic activation in the cloud according to secure computation policies.

Table 6. Secure Storage and Upload Policies

Storage Adversarial Model, Computation Adversarial Model	Secure Storage Policy* (for data uploaded beforehand) (*For brevity, reserved words 'None', 'Low', etc. are used to indicate data belonging to corresponding security levels.)	Secure Upload Policy (for data uploaded during computation)
Honest, Honest	All data stored in plain text.	None
Honest, Rational	All data stored in plain text. Other than 'None' are signed.	None
Honest, Semi-honest	Public data stored in plain text; other data in (k, n) shares (k> n/2).	None
Honest, Malicious	'None' stored in plain text. 'Very Low' signed and stored in plain text while those with security 'Low' signed and stored in multiple nodes. (n, n) signed shares of all other data.	None
Rational, Honest/Rational	Data with security level 'None' and 'Medium' stored in plaintext; 'Very low' and 'High' signed and stored in a single node; all other data signed and stored in multiple nodes such that they can be retrieved even if one node is honest.	None
Rational, Semi-honest	'None' stored in plaintext; 'Very Low' signed and stored in a single node; 'Low' signed and stored in multiple nodes; 'Medium' stored in (k, n) shares while 'high' stored in signed (k, n) shares; other data remain with the organization.	Upload (k, n) signed shares of data.
Rational, Malicious	'None' stored in plaintext; 'Very Low' signed and stored in a single node; 'Low' signed and stored in multiple nodes; 'Medium' and 'high' signed and stored using (n, n) shares; other data remain with the organization.	Upload (n, n) signed shares of data.
Semi-honest, Honest/Rational/Semi-honest	Public data stored in plain text; other data stored in (n, n) shares.	None
Semi-honest, Malicious	Public data stored in plain text; other data stored in (n, n) signed shares.	None
Malicious, Honest/Rational	'None' stored in plain text; 'Low' signed and stored in multiple nodes; signed 'Very Low' stored in a single storage node. (n, n) hares of 'medium' and (n, n) signed shares of 'high' stored; 'very high' remain with organization.	Upload 'Very High"
Malicious, Semi-honest	Same as in previous case.	Upload (k, n) shares of 'Very High'.
Malicious, Malicious	Same as in previous case.	Upload (n, n) signed shares of 'Very High'

EDC has three main components: 1) the user interface; 2) the EDAP filter and 3) the control box. The user interface is responsible for receiving users' data storage, data access and computation requests and user inputs, perceptions and displays outputs of computations. The EDAP filter consists of the EDAP matrix. So whenever the user interface receives any user request it passes it on to the EDAP filter which subsequently decides with the help of the EDAP matrix whether it is a valid user request. If it is valid then the request or inputs related to the request are forwarded to the control box. Otherwise an invalid message is passed on to the user interface. The control box has three sub-components 1) policy controller; 2) data and storage controller and 3) computation controller. The policy controller receives super-user perceptions on CSP trustworthiness and converts them into adversarial models for storage and compute nodes as per Tables 3 and 4.

Similarly user perception about data security requirements are converted into security levels for different data from both storage and computation aspects as per Table 2. This is then used to select the appropriate secure storage and upload policy (using Table 6) and secure computation policy (using Table 5) which then guide data and storage controller to upload data to the cloud (with or without sharing, with or without signing). The data and storage controller is responsible for generating shares of data, digitally signing data or shares and uploading them to appropriate storage nodes in the cloud. It also retreives data from storage nodes upon user request, checks for integrity and reconstructs the original data if shares were retreived. Similarly, the selected secure computation policy directs the computation logic

activator sub-component of the computation controller to activate the chosen computation (multi-party or single-party computation) logic in the public cloud. The I/O Manager receives from the user data element identifiers to be used as input for computations and provides the user with the outputs corresponding to a computation.

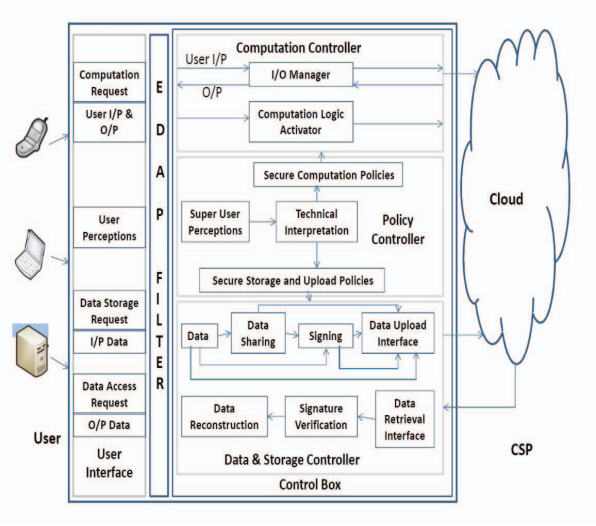

Figure 2. Implementation of Security Framework

6. Conclusion and Future Scopes

In this paper we have proposed a policy-based security framework which is highly evolving and dynamic for securely outsourcing enterprise data and computations. This framework, unlike other related works in the literature, elaborately takes into account varying user perceptions, gathered in a decentralized way directly from the users, about trustworthiness of CSPs and data security requirements to formulate secure data policies which ultimately help the organization to decide what data to outsource, how to secure data storage and computations in various scenarios. This work also deals with the aspects of secure storage and computation in the cloud in a very distinct yet integrated manner which is a novel concept by itself. In the process, the articulation of individual user's perception about the security requirements and the trustworthiness of the CSP are captured both in terms of computation and storage and is matched with various adversarial models for final decision making at the time of outsourcing the storage and computation to the cloud.

What we lack in this presentation a formal proof of concept, which we are unable to provide because of lack of space. Further, some empirical justification is called for. One could probably try to represent the model though object concepts such as use cases. The immediate next step is to analyze in details its efficiency vs. security tradeoffs with respect to a pessimistic or optimistic view of about CSP's trustworthiness. Developing a feedback and recommender system for monitoring user activities vis-à-vis their opinionating or user perceptions is another task of importance.

Instead of using several storage nodes or several computation nodes in the same CSP, multiple CSPs can be used for this purpose ([2], [3], [15]). This will increase reliability of the whole system. However, policy formulation will be more intricate as users will need to state their perception about multiple CSPs for a single data storage/ computation activity.

Aggregation of general user perception (i.e. apart from super-users) can be of interest. In fact, the framework can be modified and made useful for private users of cloud for storage and computation to classify their data, take informed decisions about which CSP to trust and also to know what security measures to take before uploading their data and offloading their computation.

7. Acknowledgements

We are indebted to the anonymous reviewers for their numerous useful comments and suggestions. We would like to thank them for their kind efforts to help us improve our work.

8. References

[1] A. Acquisti, and J. Grossklags, "Privacy and Rationality in Individual Decision Making", IEEE Security and Privacy Vol. 3 No. 1, IEEE, 2005, pp. 26-33.

[2] M. A. AlZain, and E. Pardede, "Using Multi Shares for Ensuring Privacy in Database-as-a-Service", 44th Hawaii International Conference on System Sciences, IEEE, 2011.

[3] M. A. AlZain, E. Pardede, B. Soh, and J. A. Thom, "Cloud Computing Security: From Single to Multi-Clouds", 45th Hawaii International Conference on System Sciences, IEEE, 2012.

[4] A. Bessani, M. Correia, B. Quaresma, F. Andre, and P. Sousa, "DEPSKY: Dependable and Secure Storage in a Cloud-of-Clouds", Proceedings of the 6th conference on computer systems EuroSys'11, ACM, New York USA, 2011, pp. 31-46.

[5] S. Bugiel, S. Nurnberger, A. Sadeghi, and T. Schneider, „Twin Clouds: An Architecture for Secure Cloud Computing", Workshop on Cryptography and Security in Clouds, 2011.

[6] S. Chaves, C. B. Westphall, and F. R. Lamin, "SLA Perspective in Security Management for Cloud

Computing", 6th International Conference on Networking and Services, IEEE, 2010.
[7] Y. Chen, and R. Sion, "On Securing Untrusted Clouds with Cryptography", Proceedings of the 9th annual ACM Workshop on Privacy in Electronic Society WPES'10, ACM, New York USA, 2010, pp. 109-114.
[8] N. Christin, S. Egelman, T. Vidas, and J. Grossklags, „ It's All About the Benjamins: An empirical study on incentivizing users to ignore security advice", Financial Cryptography and Data Security, Springer Berlin Heidelberg, 2012, pp. 16-30.
[9] S. De, S. Saha, and A. K. Pal, "Achieving Energy Efficiency and Security in Mobile Cloud Computing", Proceedings of the 3rd International Conference on Cloud Computing and Services Sciences CLOSER 2013, SciTePress, 8-10 May 2013, Aachen, Germany.
[10] J. Fontana**, "**Are human firewalls the enterprise info. sec of the future? http://www.zdnet.com/are-human-firewalls-the-enterprise-info-sec-of-the-future-7000008497/" (a discussion on Tom Scoltz et al, Gartner's Report on People Centric Information Security Strategy, 2012.)
[11] O. Goldreich, "Foundations of Cryptography Volume II Basic Applications". Cambridge, UK: Cambridge University Press, 2004.
[12] W. Itani, A. Kayssi, and A. Chehab, "Privacy as a Service: Privacy-Aware Data Storage and Processing in Cloud Computing Architectures", 8th IEEE Conference on Dependable, Autonomic and Secure Computing, IEEE, 2009, pp. 711-716.
[13] A. W. Jansen, "Cloud Hooks: Security and Privacy Issues in Cloud Computing", 44th Hawaii International Conference on System Sciences, 2011, pp. 1-10.
[14] M. Jensen, J. Schwenk, J. Bohli, N. Gruschka, and L. Iacono, "On Technical Security Issues in Cloud Computing", IEEE International Conference on Cloud Computing, IEEE, 2009.
[15] M. Jensen, J. Schwenk, J. Bohli, N. Gruschka, and L. Iacono, "Security Prospects through Cloud Computing by Adopting Multiple Clouds", IEEE 4th International Conference on Cloud Computing, IEEE, 2011.
[16] S. Kamara, and K. Lauter, "Cryptographic Cloud Storage", Financial Cryptography and Data Security, Springer Berlin Heidelberg, 2010, pp. 136-149.
[17] S. Kamara and M. Raykova, "Secure Outsourced Computation in Multi-tenant Cloud", IBM Workshop on Cryptography and Security in Clouds, 2011.
[18] C. Low, Y. Chen, and M. Wu, "Understanding the determinants of cloud computing adoption", Industrial Management and Data Systems Vol. 111 Issue 7, Emerald Group Publishing Ltd., 2011, pp. 1006-1023.
[19] P. Mahajan, S. Setty, S. Lee, A. Clement, L. Alvisi, M. Dahlin, and M. Walfish, "Depot: Cloud Storage with Minimal Trust", ACM Transactions on Computer Systems Vol. 29, No. 4, Article 12, ACM, 2011.
[20] A. K. Pal and S. Bose, "Information Retrieval as a Service for Multiple Heterogeneous Data–Privacy Model", The Third International Conference on Parallel, Distributed, Grid and Cloud Computing for Engineering (PARENG 2013), Pecs, Hungary, 25-27 March 2013.
[21] S. Pearson and A. Benameur, "Privacy, Security and Trust Issues Arising from Cloud Computing", 2nd IEEE Cloud Computing Technology and Science CloudCom, IEEE, 2010, pp. 693-702.
[22] F. Rocha, and M. Correia, "Lucy in the Sky with Diamonds: Stealing Confidential Data in the Cloud", 2011 IEEE/ IFIP 41st International Conference on Dependable Systems and Networks Workshops, 2011, pp. 129-134.
[23]R. Sandhu, and P. Samarati, "Access Control: Principle and Practice", IEEE Communications Magazine Vol. 32 Issue 9, IEEE, 1994, pp. 40-48.
[24] R. Seiger, S. Groβ, and A. Schill, "SecCSIE: A Secure Cloud Storage Integrator for Enterprises", IEEE Conference on Commerce and Enterprise Computing, IEEE, 2011.
[25] J. L. Spears and H. Barki, "User participation in information systems security risk management", MIS Quarterly, Vol 34, Issue 3, 2010, pp. 503-522.
[26] A. Strunk, "Secure Cloud computing with FlexCloud", DAAD Summer School CTDS, Sousse, Tunisia, 2012.
[27] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing", J. Network and Computer Applications, 34 (2011) 1–11.
[28] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, "Towards Secure and Dependable Storage Services in Cloud Computing", IEEE Transactions on Services Computing Vol. 5. No. 2., IEEE, 2012, pp. 220-232.
[29] L. Wei, H. Zhu, Z. Cao, W. Jia, and A. Vasilakos, "SecCloud: Bridging Secure Storage and Computation in Cloud", 30th International Conference on Distributed Computing Systems Workshops, IEEE, 2011.
[30] H. Xiong, X. Zhang, D. Yao, X. Wu, and Y. Wen, "Towards End-to-end Secure Content Storage and Delivery with Public Cloud", Proceedings of the 2nd ACM Conference on Data Security and Privacy CODASPY'12, ACM, New York USA, 2012, pp. 257-266.
[31] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving Secure, Scalable, and Fine-grained Data Access Control in Cloud Computing", Proceedings of IEEE INFOCOM, 2010, pp. 1-9.
[32] D. Zissis, and D. Lekkas, "Addressing Cloud Computing Security Issues", Future Generation Computer Systems Vol. 28 Issue 3, Elsevier, 2012, pp. 583-592.