CHAPTER 1

INTRODUCTION TO CLOUD COMPUTING

1.1 What is Cloud Computing?

NIST defines Cloud Computing[1] as "a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction."

According to Buyya et al. [3], Cloud computing promises reliable services delivered through next-generation data centers that are built on compute and storage virtualization technologies. Consumers will have access to data from any point, on demand, since clouds have a single point of access for all computing requests. From the business point of view, Cloud infrastructures aim to provide robustness and availability at any time, in order to appear as reliable "partners".

ACM gives yet another definition of Cloud Computing. It says, "[12]Cloud computing is about moving services, computation and/or data—for cost and business advantage—off-site to an internal or external, location-transparent, centralized facility or contractor. By making data available in the cloud, it can be more easily and ubiquitously accessed, often at much lower cost, increasing its value by enabling opportunities for enhanced collaboration, integration, and analysis on a shared common platform."

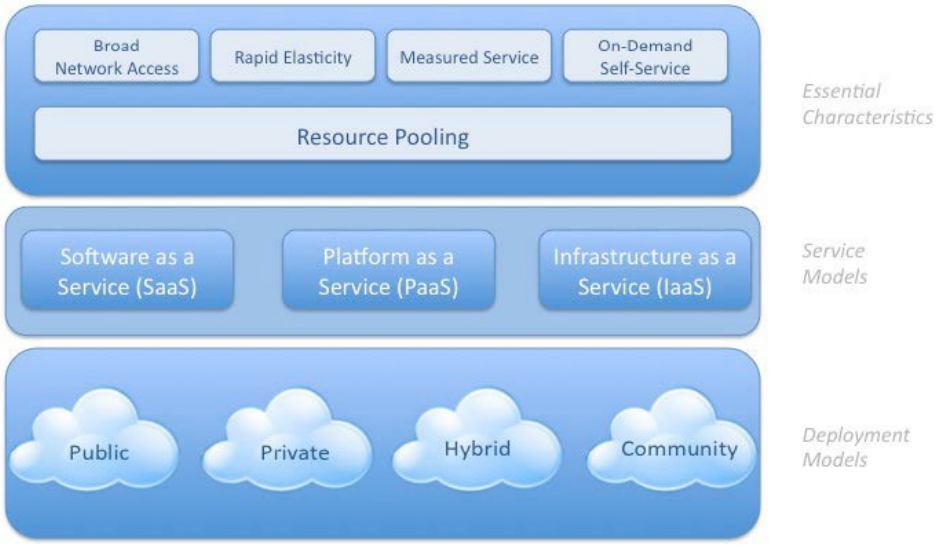

Figure 1.1: NIST visual model for Cloud Computing [14]

Cloud Computing is the combination of three known computing techniques:

- *Virtualization*: The main enabling technology for cloud computing is virtualization. Virtualization, in computing, is the creation of a virtual (rather than actual) version of something, such as a hardware platform, operating system, a storage device or network resources. It abstracts the physical infrastructure,

which is the most rigid component, and makes it available as a soft component that is easy to use and manage. By doing so, virtualization provides the agility required to speed up IT operations, and reduces cost by increasing infrastructure utilization.

- *Grid Computing*: It's a form of distributed and parallel computing, whereby a 'super and virtual computer' is composed of a cluster of networked, loosely coupled computers acting in concert to perform very large tasks. What distinguishes grid computing from high performance computing systems such as cluster computing is that grids tend to be more loosely coupled, heterogeneous, and geographically dispersed.
- *Utility Computing*: Utility computing is the packaging of computing resources, such as computation, storage and services, as a metered service. This model has the advantage of a low or no initial cost to acquire computer resources; instead, computational resources are essentially rented. This repackaging of computing services became the foundation of the shift to "on demand" computing, software as a service and cloud computing models.

1.2 Essential Characteristics of Cloud Computing

- *On-demand self-service*. A consumer can unilaterally provision computing capabilities, such as server time and storage, as needed automatically without requiring human interaction with each service provider.
- *Broad network access*. Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g., mobile phones, tablets, laptops, and workstations).
- *Resource pooling*. The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g., country, state, or datacenter). Examples of resources include storage, processing, memory, and network bandwidth.
- *Rapid elasticity*. Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be appropriated in any quantity at any time.
- *Measured service*. Cloud systems automatically control and optimize resource use by leveraging a metering capability1 at some level of abstraction appropriate to the type of service (e.g., storage, processing, bandwidth, and active user accounts). Resource usage can be monitored, controlled, and reported, providing transparency for both the provider and consumer of the utilized service.

1.3 Advantages of Cloud Computing

- *Easy Management:* The maintenance of the infrastructure, be it hardware or software is simplified, thus, less headaches for the IT team. Also applications that are quite storage extensive are easier to use in the cloud environment compared to the same when used by the organization by its own. Also, at the user level, what you mostly need is a simple web browser with internet connectivity.
- *Cost Reduction:* The main advantage for SMEs lies here. Cloud computing drastically reduces the IT spending for SMEs. Costly systems need not be required for occasional use of intensive computing resources. Also, the man power required for such systems is not required. Even simple applications like email can be set up and mostly free through applications like Google Apps. Also as most of the time such providers are quite reliable in terms of availability, it is clear winner.

- *Uninterrupted Services:* Lower outages are provided by cloud computing services, thus providing uninterrupted services to the user. However, some occurrences of outages have occurred in the past, like the Gmail outage in 2009. Also other cloud vendors like EC2 have failed at some point of time, but however, they are much more dependable compared to the infrastructure installed on the organization.
- *Disaster Management:* In case of disasters, an offsite backup is always helpful. Keeping crucial data backed up using cloud storage services is the need of the hour for most of the organizations. Also cloud storage services not only keep your data off site, but they also ensure that they have systems in place for disaster recovery.
- *Green Computing:* Harmful emissions due to extensive use of systems in organizations, electronic waste generated as the time passes and energy consumption is the main disadvantage of the present day computing systems. This can be reduced to some extent by using cloud computing services. This leads to environment preserving. Also the e-waste is generated to minimum extent.

1.4 Service Models of Cloud

- Software as a Service (SaaS). The capability provided to the consumer is to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through either a thin client interface, such as a web browser (e.g., web-based email), or a program interface. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings. Example: Google docs.
- Platform as a Service (PaaS). The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider.3 The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment. Example: Google Drive.
- Infrastructure as a Service (IaaS). The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications; and possibly limited control of select networking components (e.g., host firewalls). Example: Amazon EC2.

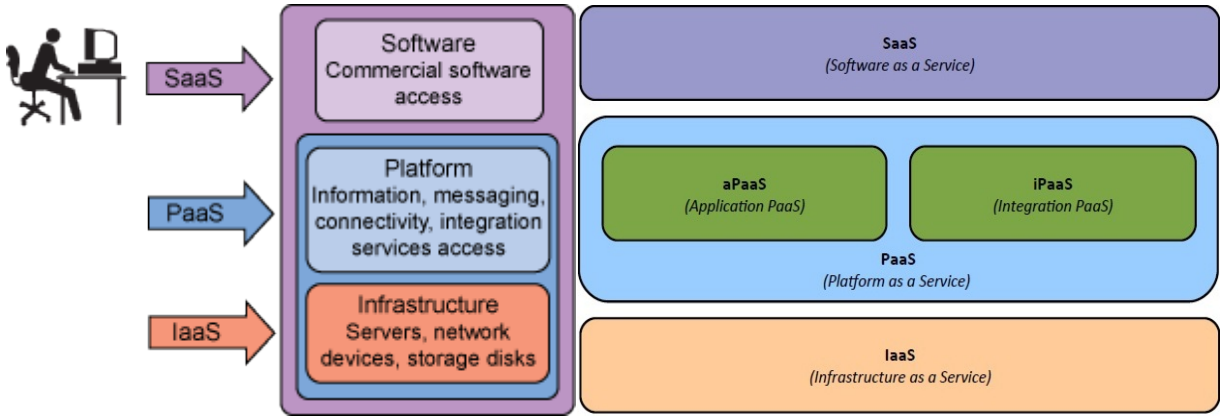

Figure 1.2: Service Models of Cloud

1.5 Deployment Models of Cloud:

- Private cloud. The cloud infrastructure is provisioned for exclusive use by a single organization comprising multiple consumers. It may be owned, managed, and operated by the organization, a third party, or some combination of them, and it may exist on or off premises. Example: NASA's Nebula.
- Community cloud. The cloud infrastructure is provisioned for exclusive use by a specific community of consumers from organizations that have shared concerns. It may be owned, managed, and operated by one or more of the organizations in the community, a third party, and it may exist on or off premises. Example: All government organisations in a state sharing computing infrastructure to manage citizens.
- Public cloud. The cloud infrastructure is provisioned for open use by the general public. It may be owned, managed, and operated by a business, academic, or government organization, or some combination of them. It exists on the premises of the cloud provider. Example: Amazon, Microsoft.
- Hybrid cloud. The cloud infrastructure is a composition of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities, but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load balancing between clouds). Example: vCloud services.

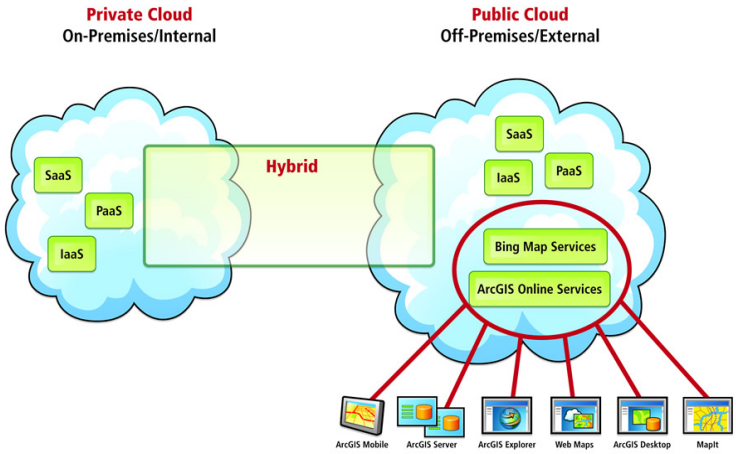

Figure 1.3: Deployment Models of Cloud

1.6 Grid vs. Cloud Computing

Ian Foster [2] defined it as

"A system that coordinates resources which are not subject to centralized control, using standard, open, general- purpose protocols and interfaces to deliver nontrivial qualities of service. "

Grid computing leverages a combination of hardware/software virtualization and the distributed sharing of those virtualized resources. It is also called CPU scavenging or cycle stealing.

Grid Computing creates a "grid" from the unused resources in a network of participants. Grids are used in: generic domain names, globally connected resource pools, enterprise departmental computing grids

Advantages of Grid Computing:

- boost productivity
- maximize resource utilization

- simply access to need of computer resource access

Disadvantage of Grid Computing:

- Availability and performance of grid resources are unpredictable : Request from within a domain may gain more priority over requests from outside

Grid Computing Vs Cloud Computing

- In Grid, resources are allocated on the basis of **turn on or off** technique. The end user may use that resource partially or fully however cloud computing is one step ahead the resources are **provisionally provided on-demand.**
- Grid computing may lead to over provisioning or even wastage of resources but in cloud computing over provisioning/wastage is not an issue.
- Both rely on virtualization and coordination of heterogeneous resources that are geographical distributed. Hence, grids enhance a fair share of resources across organization while the cloud belongs to one company and resources are ordered on demand.
- Applications for the two are developed differently: clouds are able to accept a wider range of applications, in some cases a simple deployment of the desktop version of it is possible, while for grids there is the need to "gridifie" them.
- From the point of view of usability, clouds are much easier to use since they hide most of the deployment details. According to Travor, cloud computing is the user-friendly version of grid computing. On the other hand, when it comes to standardization, there is a lot of efforts regarding grids in this manner, while for clouds there are some problems on this aspect.
- Regarding security, each site in a grid can have its own policy regarding how to access the resources, due to multiple administrators. A cloud does not have this complexity since it has a single interface that allows a user to access the entire infrastructure.

Feature	Grid	Cloud
Resource Sharing	Collaboration (Virtual organization,fair share)	Assigned resources are not shared
Resource Heterogeneity	Aggregation of heterogeneous resources	Aggregation of heterogeneous resources
Virtualization	Virtualization of data and computing resources	Virtualization of hardware and software platforms
Security	Security through credential delegations	Security through isolation
Platform Awareness	The client software must be Grid-enabled	The software works on a customized environment
Centralization Degree	Decentralized control	Centralized control
Standardization	Standardization and interoperability	Lack of standards for Clouds interoperability
Usability	Hard to manage	User friendliness

Table 1.1: Grid vs. Cloud Characteristics

1.7 Challenges of Cloud Computing

Being a relatively new technology, cloud computing still has some issues that must be overcome. The nature of these could be expressed as:

- *Data management*: still needs a lot of work. Currently there is no concurrency at IaaS level or there is a very simple mechanism at PaaS level [9]. Complex applications with high concurrency can suffer or even cannot benefit from the cloud technology until better scheme for concurrency are delivered. There are limitations on the size of the objects that can be stored, which can create some complications in the development process. The fine-grain access is another issue since for example IaaS provides just simple mechanisms like get and put for managing the data, and these operations cannot access just small parts.
- *Computational*: The cloud ecosystem is very heterogeneous and this is rejected at several levels. The diverse experience of various cloud costumers starts with the network connection that the cloud has, which can be either regular or high-performance. Along with the problem of mitigating the applications between clouds, at IaaS level, the problem of compatibility of the hypervisors also rises. There is a real need of moving from private clouds to hybrid ones or to public clouds, but there are not currently general standards regarding the deployment of VMs. But there are efforts regarding this issue [8], For instance the Open Geospatial Consortium has created an annual process between the major stakeholders for developing such standards.
- *Security issues*: In general, there are simple password mechanisms for identification, but more secured methods for authentication have been developed [11]. Recent studies have shown that a limit of the potential damage in case of an attack, like _ne-grained delegation or limits on the traffic, would be needed. Another issue concerns the total trust that the clients must have in the cloud owner regarding their data. A security measure can be the encryption of data stored inside the clouds. The encryption can be the solution also for legal constraints, like data confidentiality.
- *The programming models*: that are imposed when using cloud technology could also create drawbacks. Referring to the imposed architectures like Web Role and Worker Role in Azure, not all applications can comply to them. Moreover, the stateless feature imposed by a load-balancer, in charge of distributing the requests, creates difficulties for existing REST (representational state transfer) applications, which are mostly state full, to mitigate into clouds. Issues regarding MapReduce programs refer to data location awareness, since the efficiency depends on the placement of the mappers close to the data.

1.8 Architecture of Cloud Computing

The system level or the IaaS layer of the architecture corresponds to the various computers, servers and data storage systems that create the "cloud" of computing services.

The middle layer, as is depicted, is further divided into two parts – the core middleware and the user level middleware. Core Middleware is the PaaS layer equivalent. This is equivalent to middleware in the traditional (non-cloud computing) delivery of application platforms and databases[36]. Middleware allows networked computers to communicate with each other apart from taking care of pricing control when the resources are being used by the customers. At its very base is the virtual machine deployment. Here is how it works - It's possible to fool a physical server into thinking it's actually multiple servers, each running with its own independent operating system. The technique is called server virtualization. This reduces the need for many physical machines and maximizes usage of the available resources.

User level Middleware or SaaS layer consists of libraries, scripting and distributed programming interfaces used to write and run the applications hosted on the Cloud platform. These applications are offered as a service to the end users.

The top layer/ User level has services such as ISV(Independent Service Vendors) and CDN(Content Delivery Networks).These might be the users themselves or might serve as a step between the users.

A more detailed description of each of the three service models – IaaS,PaaS and SaaS follows below.

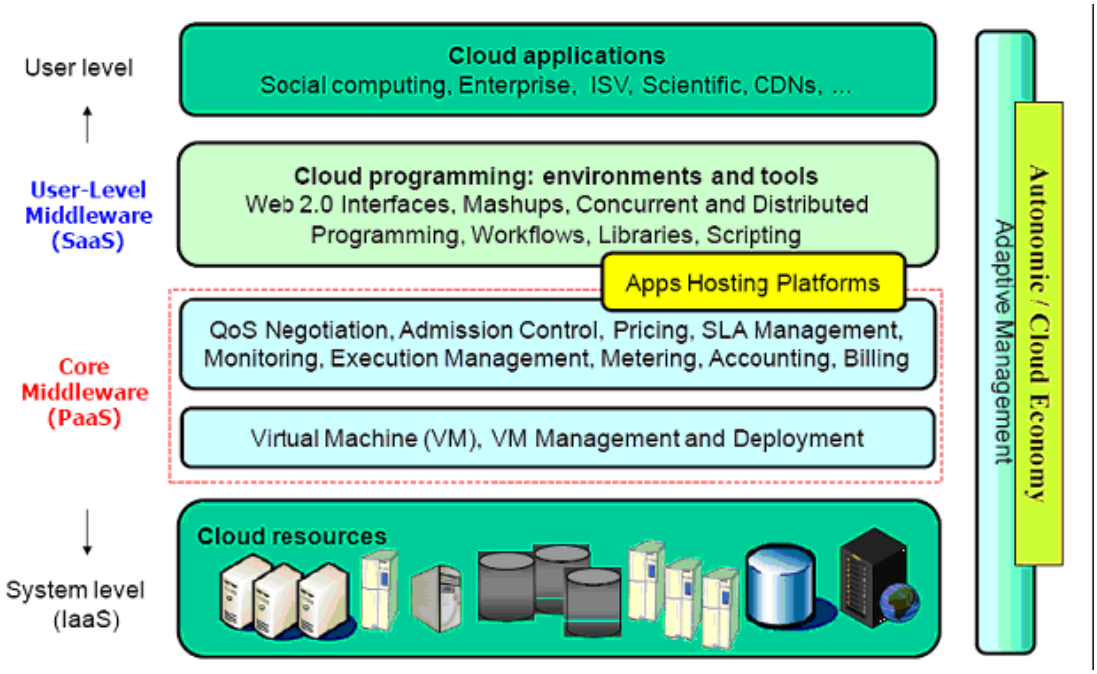

Figure 1.4: Layered Architecture of Cloud Computing

With respect to the ways clouds can be used, the "de facto" consensus achieved led to the definition of 3 major exploitation levels: Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS) [4, 5, 6]. They can be conceptually viewed in Figure 5. The particularities of these will be highlighted and exemplified.

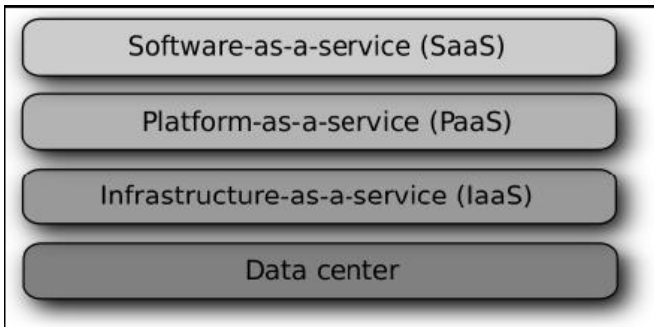

Figure 1.5: Cloud layers

Infrastructure as a Service offers a large set of computing resources (computation power or storage capacity) [4, 5]. This is exploited by means of virtualization, being highly dynamic and allowing the creation of ad-hoc systems on demand. Instead of buying servers, disks or networking equipment, cloud consumers rent and customize virtual machine images. Fees are charged in general, on a utility basis that reflects the amount of

raw resources used: storage space-hour, bandwidth, aggregated CPU cycles consumed, etc. [4]. The most successful cloud systems at IaaS level are: Amazon EC2, Nimbus, OpenNebula, Eucalyptus. All these systems offer a simple on-line interface through which the infrastructure can be used, users have an account for logging to the front end and launching multiple VM 0instances on the cloud. The generic architecture is shown on Figure 6. One important component is the hypervisor, which is a low-level software that presents the guest operating systems with a virtual operating platform which monitors the execution of the guest operating systems [28, 35]. It is present in each compute node, with the role of supervising the multiple virtual machines (VMs) that run on the cloud nodes.

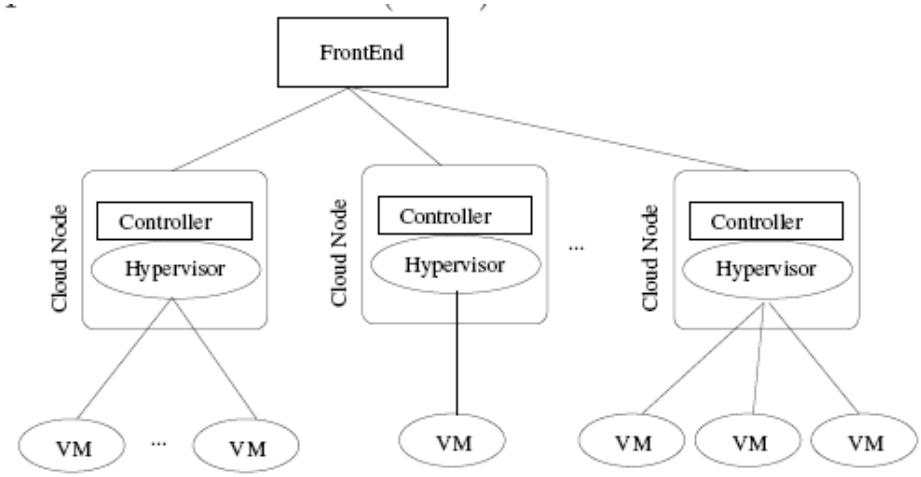

Figure 1.6: Generic IaaS architecture

Platform as a Service offers the possibility to exploit clouds in a different manner than using the virtualized infrastructure. Users can directly use a software platform, the requested hardware being provided transparently [4]. As it can be intuited, it is constructed on top of IaaS, providing a higher level of programming and freeing the customer from configuring VMs. A common way to develop applications at this level is to comply with specific roles, like in the case of MapReduce or Azure. Microsoft Azure is one of the most representative PaaS systems offered for commercial use, whose generic architecture can be seen on Figure 7. The developers program at a higher level, concentrating only on the code for the applications that will run inside the cloud, possibly conformed to a specific architecture (ex. Web Role and Worker Role for Azure) or/and on the data, which is stored through simple methods (eg: HTTP requests). Google offers as well commercial access at its PaaS infrastructure through Google Apps Engine, providing the customers fast development and deployment, simple administration, with no need to worry about hardware, patches or backups and effortless scalability. MapReduce [7] has emerged recently as a new programming paradigm used at PaaS level. Its open implementation, called Hadoop and supported by Yahoo!, has gained a lot in popularity recently. The MapReduce model consists in providing only 2 functions: Map and Reduce. The platform will be responsible for creating the specific number of workers that will run the code for each function and for the data flow to and from workers. In addition to Google and Yahoo!, Microsoft offers Dryad, which has the same programming principles as MapReduce, but is more general. The additional features provided, allow more complex data flows and compositions between the workers.

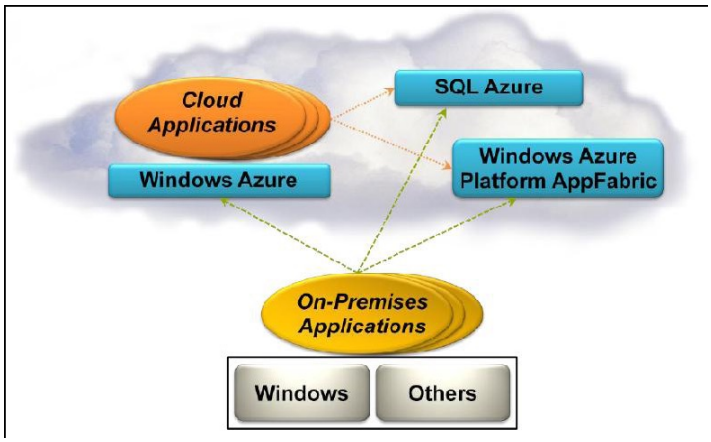

Figure 1.7: Azure architecture

Software as a Service is the highest level at which clouds can be used by the customers. Notorious examples of such services are given next. Google offers Google Docs, where users can store their documents out there and are able to access them from any place. Microsoft Live Services with Live.com is a set of personal Internet services and software designed to bring together in one place all of the relationships, information and interests people care about most, like mail, account, messenger, office etc.. Other players in the market like Amazon Web Services or saleforce.com concentrate mostly on E-commerce. These become more and more popular, being addressed to all types of users, relieving them from installing software, updates or patches [4].

In general a simple web browser is enough for accessing these softwares, as they can be reached from any location based on an ID. Others. As the popularity of cloud grows, new types of exploitation appear besides the 3 mentioned above. Microsoft Azure has successfully deployed its SQL Database into the Azure cloud. The major advantage is that it can be used identically as a normal database, having all the benefits of the cloud. They refer to it, as DataBase as a Service DaaS, but more common Data as a Service is used. The SaaS concept can even be extended to the notion of Models as a Service (MaaS) where semantic annotations and ontologies are used to compose computational models and execute them as a conceptual whole [8]. If the current trends hold, new such concepts will continue to appear, but as it can be expected, they can be integrated in one of the three main categories, as DaaS could be considered as part of the PaaS, or MaaS from SaaS.

CHAPTER 2

LITERATURE SURVEY

2.1 Threats in Cloud Computing

The biggest concerns about cloud computing are security and privacy[13].
Handing over of crucial confidential data to another company gives jitters to some people. Corporate users will definitely hesitate to some extent in adopting cloud services as they can't keep their company's information under lock and key.

Privacy is another factor. As these data are accessed from any location, it's possible the client's privacy could be compromised. One way to solve this issue is the use of proper authentication techniques. Another solution is to provide with an authorization - so that each user can access only the data and applications relevant to his or her job.

Replication time and costs also play an important role. How fast can the data be replicated is important for data resiliency. Reliability is an issue. Servers in the cloud can have the same problems as the organization's resident servers. Downtimes can occur with cloud servers too.

According to the CSA guide [15], the following are the top threats in cloud computing:

- Abuse and Nefarious Use of Cloud Computing:

  IaaS providers offer their customers the illusion of unlimited compute, network, and storage capacity often coupled with a 'frictionless' registration process where anyone with a valid credit card can register and immediately begin using cloud services. Some providers even offer free limited trial periods. By abusing the relative anonymity behind these registration and usage models, spammers, malicious code authors, and other criminals have been able to conduct their activities with relative impunity. PaaS providers have traditionally suffered most from this kind of attacks; however, recent evidence shows that hackers have begun to target IaaS vendors as well. Future areas of concern include password and key cracking, DDOS, launching dynamic attack points, hosting malicious data, botnet command and control, building rainbow tables, and CAPTCHA solving farms.

  Remediation: Stricter initial registration and validation processes, Enhanced credit card fraud, Monitoring and coordination, Comprehensive introspection of customer network traffic, Monitoring public blacklists for one's own network blocks [16, 17, 18]

- Insecure Application Programming Interfaces

  Cloud Computing providers expose a set of software interfaces or APIs that customers use to manage and interact with cloud services. Provisioning, management, orchestration, and monitoring are all performed using these interfaces. The security and availability of general cloud services is dependent upon the security of these basic APIs. From authentication and access control to encryption and activity monitoring, these interfaces must be designed to protect against both accidental and malicious attempts to circumvent policy. Furthermore, organizations and third parties often build upon these interfaces to offer value-added services to their customers. This introduces the complexity of the new layered API; it

also increases risk, as organizations may be required to relinquish their credentials to third parties in order to enable their agency.

Remediation: Analyze the security model of cloud provider interfaces, Ensure strong authentication and access controls are implemented in concert with encrypted transmission, Understand the dependency chain associated with the API [19, 20]

- Malicious Insiders:

  The threat of a malicious insider is well-known to most organizations. This threat is amplified for consumers of cloud services by the convergence of IT services and customers under a single management domain, combined with a general lack of transparency into provider process and procedure. For example, a provider may not reveal how it grants employees access to physical and virtual assets, how it monitors these employees, or how it analyzes and reports on policy compliance. To complicate matters, there is often little or no visibility into the hiring standards and practices for cloud employees. This kind of situation clearly creates an attractive opportunity for an adversary — ranging from the hobbyist hacker, to organized crime, to corporate espionage, or even nation-state sponsored intrusion. The level of access granted could enable such an adversary to harvest confidential data or gain complete control over the cloud services with little or no risk of detection.

  Remediation: Enforce strict supply chain management and conduct a comprehensive supplier assessment, Specify human resource requirements as part of legal contracts, Require transparency into overall information security and management practices, as well as compliance reporting, Determine security breach notification processes [21, 22]

- Shared Technology Vulnerabilities

  IaaS vendors deliver their services in a scalable way by sharing infrastructure. Often, the underlying components that make up this infrastructure (e.g., CPU caches, GPUs, etc.) were not designed to offer strong isolation properties for a multi-tenant architecture. To address this gap, a virtualization hypervisor mediates access between guest operating systems and the physical compute resources. Still, even hypervisors have exhibited flaws that have enabled guest operating systems to gain inappropriate levels of control or influence on the underlying platform. A defense in depth strategy is recommended, and should include compute, storage, and network security enforcement and monitoring. Strong compartmentalization should be employed to ensure that individual customers do not impact the operations of other tenants running on the same cloud provider. Customers should not have access to any other tenant's actual or residual data, network traffic, etc.

  Remediation: Implement security best practices for installation/configuration, Monitor environment for unauthorized changes/activity, Promote strong authentication and access control for administrative access and operations, Enforce service level agreements for patching and vulnerability remediation, Conduct vulnerability scanning and configuration audits [23, 24, 25, 26]

- Data Loss/Leakage

  There are many ways to compromise data. Deletion or alteration of records without a backup of the original content is an obvious example. Unlinking a record from a larger context may render it unrecoverable, as can storage on unreliable media. Loss of an encoding key may result in effective destruction. Finally, unauthorized parties must be prevented from gaining access to sensitive data.

  The threat of data compromise increases in the cloud, due to the number of and interactions between risks and challenges which are either unique to cloud, or more dangerous because of the architectural or operational characteristics of the cloud environment.

Remediation: Implement strong API access control, Encrypt and protect integrity of data in transit, Analyse data protection at both design and run time, Implement strong key generation, storage and management, and destruction practices, Contractually demand providers wipe persistent media before it is released into the pool, Contractually specify provider backup and retention strategies [27, 28, 29]

- Account, Service & Traffic Hijacking

  Account or service hijacking is not new. Attack methods such as phishing, fraud, and exploitation of software vulnerabilities still achieve results. Credentials and passwords are often reused, which amplifies the impact of such attacks. Cloud solutions add a new threat to the landscape. If an attacker gains access to your credentials, they can eavesdrop on your activities and transactions, manipulate data, return falsified information, and redirect your clients to illegitimate sites. Your account or service instances may become a new base for the attacker. From here, they may leverage the power of your reputation to launch subsequent attacks.

  Remediation: Prohibit the sharing of account credentials between users and services, Leverage strong two-factor authentication techniques where possible, Employ proactive monitoring to detect unauthorized activity, Understand cloud provider security policies and SLAs [30, 31]

- Unknown Risk Profile

  One of the tenets of Cloud Computing is the reduction of hardware and software ownership and maintenance to allow companies to focus on their core business strengths. This has clear financial and operational benefits, which must be weighed carefully against the contradictory security concerns — complicated by the fact that cloud deployments are driven by anticipated benefits, by groups who may lose track of the security ramifications.

  Versions of software, code updates, security practices, vulnerability profiles, intrusion attempts, and security design, are all important factors for estimating your company's security posture. Information about who is sharing your infrastructure may be pertinent, in addition to network intrusion logs, redirection attempts and/or successes, and other logs. Security by obscurity may be low effort, but it can result in unknown exposures. It may also impair the in-depth analysis required highly controlled or regulated operational areas.

  Remediation: Disclosure of applicable logs and data, Partial/full disclosure of infrastructure details (e.g., patch, levels, firewalls, etc.), Monitoring and alerting on necessary information [32, 33, 34, 35]

2.2 Security for Cloud Computing

Security controls [14] in cloud computing are, for the most part, no different than security controls in any IT environment. However, because of the cloud service models employed, the operational models, and the technologies used to enable cloud services, cloud computing may present different risks to an organization than traditional IT solutions.

An organization's security posture is characterized by the maturity, effectiveness, and completeness of the risk-adjusted security controls implemented. These controls are implemented in one or more layers ranging from the facilities (physical security), to the network infrastructure (network security), to the IT systems (system security), all the way to the information and applications (application security). Additionally, controls are implemented at the people and process levels, such as separation of duties and change management, respectively.

The security responsibilities of both the provider and the consumer greatly differ between cloud service models. Amazon's AWS EC2 infrastructure as a service offering, as an example, includes vendor responsibility for security up to the hypervisor, meaning they can only address security controls such as physical security, environmental security, and virtualization security. The consumer, in turn, is responsible for security controls that relate to the IT system (instance) including the operating system, applications, and data.

The inverse is true for Salesforce.com's customer resource management (CRM) SaaS offering. Because Salesforce.com provides the entire "stack," the provider is not only responsible for the physical and environmental security controls, but it must also address the security controls on the infrastructure, the applications, and the data. This alleviates much of the consumer's direct operational responsibility.

There is currently no way for a naive consumer of cloud services to simply understand what exactly he/she is responsible for [though reading this guidance document should help], but there are efforts underway by the CSA and other bodies to define standards around cloud audit. [14]

One of the attractions of cloud computing is the cost efficiencies afforded by economies of scale, reuse, and standardization. To bring these efficiencies to bear, cloud providers have to provide services that are flexible enough to serve the largest customer base possible, maximizing their addressable market. Unfortunately, integrating security into these solutions is often perceived as making them more rigid.

This rigidity often manifests in the inability to gain parity in security control deployment in cloud environments compared to traditional IT. This stems mostly from the abstraction of infrastructure, and the lack of visibility and capability to integrate many familiar security controls, especially at the network layer.

The figure [14] below illustrates these issues: in SaaS environments, the security controls and their scope are negotiated into the contracts for service; service levels, privacy, and compliance are all issues to be dealt with legally in contracts. In an IaaS offering, while the responsibility for securing the underlying infrastructure and abstraction layers belongs to the provider, the remainder of the stack is the consumer's responsibility. PaaS offers a balance somewhere in between; where securing the platform falls onto the provider, but both securing the applications developed against the platform and developing them securely, belong to the consumer.

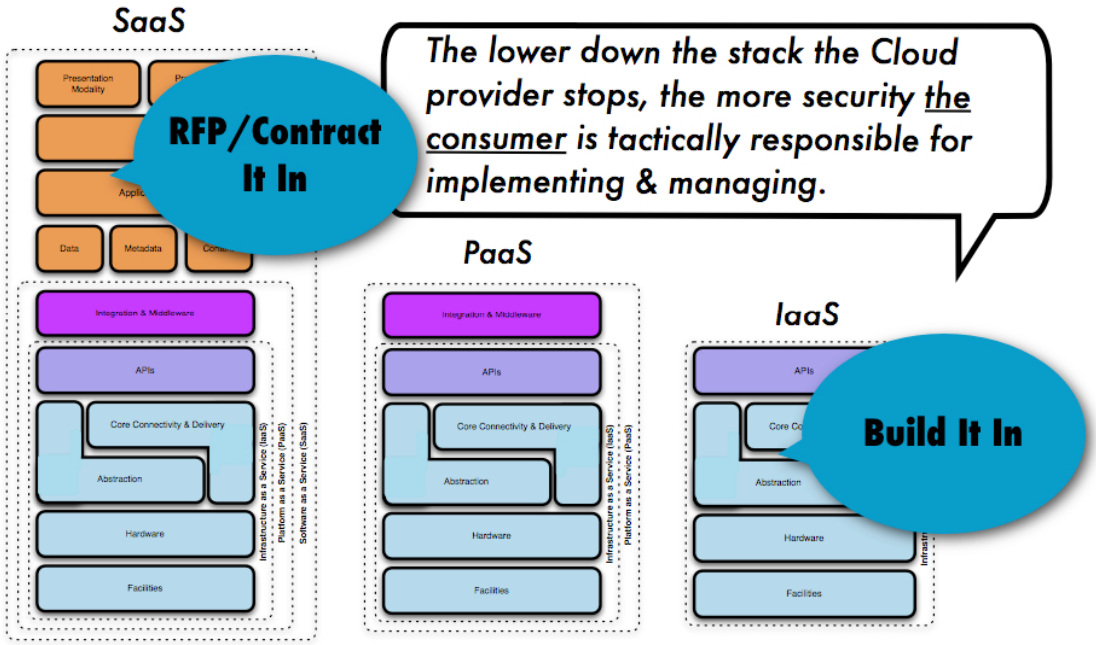

Figure 2.1: How Security Gets Integrated

Understanding the impact of these differences between service models and how they are deployed is critical to managing the risk posture of an organization.

2.3 Security Threats in Cloud Computing

The cloud system is running on the internet and the security problems in the internet also can be found in the cloud system. The traditional security problems such as security vulnerabilities, virus and hack attack can also make threats to the cloud system and can lead more serious results because of property of cloud computing. Hackers and malicious intruder may hack into cloud accounts and steal sensitive data stored in cloud systems.

The data and business application are stored in the cloud center and the cloud system must protect the resource carefully. The cloud must provide data control system for the user. The data security audit also can be deployed in the cloud system. Data integrity requires that only authorized users can change the data and Confidentiality means that only authorized users can read data. Cloud computing should provide strong user access control to strengthen the licensing, certification, quarantine and other aspects of data management.

In the cloud, the cloud provider system has many users in a dynamic response to changing service needs. The users do not know the position of the data and do not know which servers are processing the data. There in no way for the user to ensure that data privacy is operated by the cloud in a confidential way. The cloud system can deploy the cloud center in different areas and the data can be stored in different cloud node. Different areas have different laws so the security management can also face the law risk. Cloud computing service must be improved to include legal protection.

Security threats in cloud computing affect the users, the data as well as the cloud service provider.

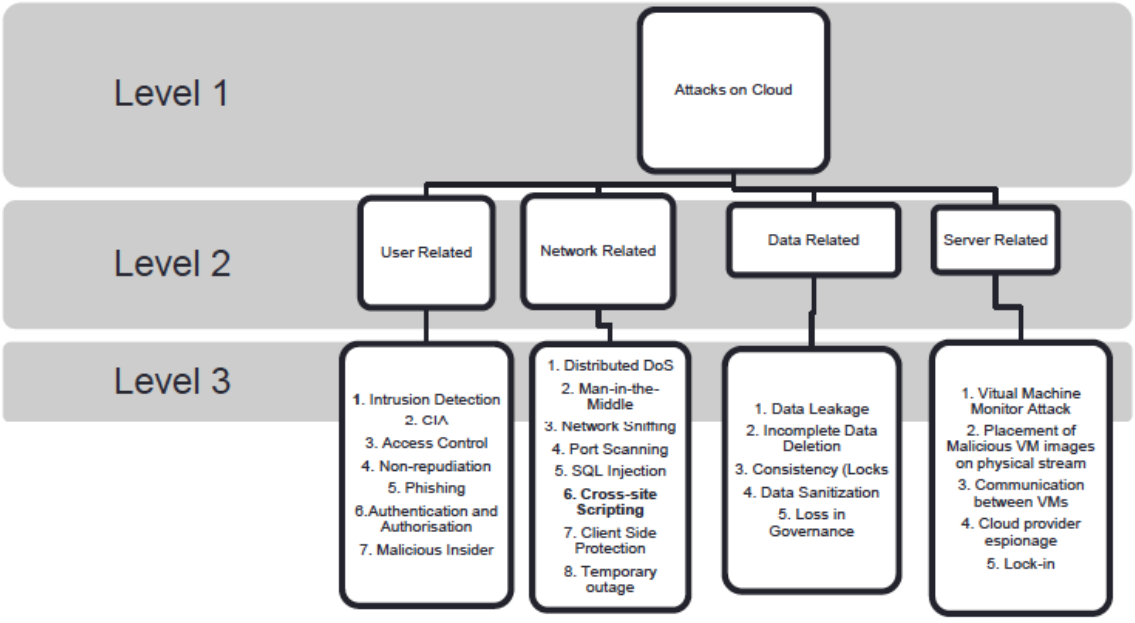

Figure 2.2: Security Threats in Cloud Computing

2.4 Solution Offered by Intel

A possible solution to a few of these problems is Intel's **Check Point Open Performance Architecture**. It works on multi-core Intel processors to provide security against application layer threats (eg: VoIP, mail etc) without compromising performance.

Securing a network is a constant tradeoff between enabling users to readily access data while protecting them from cybercriminals. Users want and expect instant access to data, to systems and to other people, as in the case of Voice over Internet Protocol (VoIP). On the other hand, security is about limiting unfettered access, thereby securing data and systems and keeping them virus-free.

Complicating matters is the fact that as security controls are increased, the security tools themselves come under a higher workload, which reduces performance and indirectly affects access levels. To protect against today's risks of highly advanced attacks and information leakage, it's necessary to perform a higher level of inspection on traffic passing through the perimeter gateway. When more security checks are placed on information, the security tools themselves face a greater processing load to implement the security policy—effectively slowing down security inspection.

This problem of balancing information access and security is evident in two key areas: increased bandwidth requirements and rising levels of application-layer threats.

Increased Bandwidth Requirements

Networks are transitioning from 1 Gb to 10 Gb Ethernet. Although this will not immediately translate into increased throughput requirements at the perimeter, security performance requirements will increase on the whole.

Increased Application-Layer Threats

Today, many attacks are masquerading as legitimate application-layer traffic, enabling them to threaten a whole host of applications: instant messaging, chat, peer-to-peer and Web applications, just to name a few. The reasoning behind these attacks is that traditional firewall based security focuses on network-layer access, preventing people from accessing specific IP addresses or networks unless authorized. Modern attacks mean that a supposedly trusted user is disguising the traffic so that it passes the firewall. From a security inspection viewpoint, the answer is to perform a deeper level of inspection, similar to intrusion prevention, on the firewall to detect application-layer threats. However, every additional security screening that is done decreases the ability for the firewall to efficiently process the traffic, slowing down its predictable performance.

Security solutions designed with open architectures, based on multi-core high performance processors, deliver the flexibility and performance needed to protect against existing and potential threats. By combining the security and platform expertise of Check Point and Intel, institutions can deploy world-class security devices designed to meet next-generation security challenges.

High Security for High-Performance Environments

Security appliances based on Check Point Open Performance Architecture and Quad-Core Intel® Xeon® processors 5400 series are changing the security performance equation. Check Point security software utilizes as many as eight CPU cores, supplied by two quad-core Intel® processors, which provide the performance headroom needed to protect networks into the future. Key performance statistics include:

• 12 gigabits per second (Gbps) firewall inspection delivers data center level speed for the most demanding enterprise environments.

• 5.3 Gbps intrusion prevention inspection with default settings provides a balance between security and performance.

• 1.8 Gbps intrusion prevention inspection with strict protection file offers maximum security without compromising performance.

To reach these speeds, the security appliance employs the Check Point Open Performance Architecture, which consists of three patented technologies:

• CoreXL™ Multi-core Acceleration—distributes security inspection duties throughout all the cores in a multi-core processor-based system, thereby fully utilizing the computing power of the security appliance.

• SecureXL™ Security Acceleration—accelerates security inspection by removing the latency introduced as network traffic passes through a security device.

• ClusterXL™ Smart Load Balancing— provides high availability and load sharing and enables near-linear performance as the cluster size increases. It distributes traffic between clusters of redundant gateways so that the computing capacity of multiple machines may be combined to increase total throughput.

These three technologies work together to fully accelerate security inspection along a unified path that ensures both high performance and high security.

2.5 Security Threats in Cloud computing: Detailed Explanation

2.5.1 Threats to User in Cloud

1) Ensuring user's CIA (confidentiality, integrity, availability)

The only way to ensure the CIA [38] of the user is to use highly secured system. That is to allow only authenticated users to access the data. Users can choose complex 10-digit passwords.

2) Malicious Insider

It is difficult to identify an unauthorized user if that malicious insider is passively observing the data. The solution to detect malicious insider is to have a hierarchy [39] of admin ex :- Application admin, system admin, Virtual admin, hosting company admin . Each of these admins in hierarchy will have different level of access rights.

3) Non Repudiation

To prevent any of the end user from denying receiving a message after the packets/ message/ data [42] has been successfully transmitted is to have a third party in between the user(client) and CSP(cloud service provider/server) which will keep track of the flow of message/data.

4) Access Control

Access control mechanisms [37] are tools to ensure authorized user can access and to prevent unauthorized access to information systems. Therewith, formal procedures should be in place to control the allocation of access rights to information systems and services. Such mechanisms should cover all stages in the lifecycle of user access, from the initial registration of new users to the final de-registration of users who no longer require access to information systems and services. Special attention should be given, where appropriate, to the need to control the allocation of privileged access rights, which allow users to override system controls. The following six control statements ensure proper access control management: Control access to

information, Manage user access, Encourage good access practices, Control access to network services, Control access to operating systems, Control access to applications and systems

5) Authentication & Authorization

When organizations start to utilize cloud services, authenticating [40] users in a trustworthy and manageable manner is a vital requirement. Organizations must address authentication-related challenges such as credential management, strong authentication (typically defined as multi-factor authentication), delegated authentication, and managing trust across all types of cloud services.

The requirements for user profiles and access control policy vary depending on whether the user is acting on their own behalf (such as a consumer) or as a member of an organization (such as an employer, university, hospital, or other enterprise). The access control requirements in SPI environments include establishing trusted user profile and policy information, using it to control access within the cloud service, and doing this in an auditable way.

6) Phishing

In computing, Phishing [41] is a criminal activity using social engineering techniques. Phishers attempt to fraudulently acquire sensitive information, such as passwords and credit card details, by masquerading as a trustworthy person or business in an electronic communication.

7) Intrusion Detection

In computing, Intrusion detection [43] is required because once the data goes out of the company's firewall; it becomes necessary to have some kind of intrusion detection system. Also, with expanded usage of Cloud services it is no more possible for a system admin to monitor transaction.

Protection Level Provided In Intrusion Detection Schemes (Ids)

Host Based IDs: - Run on individual host/devices -Utilize firewall and analyze reports, Network Based IDs: - Protect a network segment -More general than host based

Detection Techniques Used in IDS

a) Anomaly Based -The system is told the correct pattern. If any pattern is found that is different from the known pattern is termed as anomaly. System admin is is notified of anomaly being detected. This scheme can detect unknown attacks

b) Signature Based -The system is told the model of the abnormal behavior/signature .If the signature in the packet matches the previously known signature the system admin is notified of the detection. The disadvantage is that only previously known attacks can be detected.

2.5.2 Threats to Data in Cloud

1) Data Protection:

Cloud computing poses several data protection risks [38] for cloud customers and providers. In some cases, it may be difficult for the cloud customer (in its role as data controller) to effectively check the data handling practices of the cloud provider and thus to be sure that the data is handled in a lawful way. This problem is exacerbated in cases of multiple transfers of data, e.g., between federated clouds. On the other hand, some cloud providers do provide information on their data handling practices. Some also offer certification summaries on their data processing and data security activities and the data controls they have in place, e.g., SAS70 certification.

Also, since data is stored with a third-party, it needs to be protected from malicious insiders.

Solutions:

a) Firewall

b) Encrypting data to enhance security (but this impedes usage and query processing)

c) Error detecting and error correcting codes

2) Data Leakage and Loss:

There are many ways to compromise data. [44, 45, 47] Deletion or alteration of records without a backup of the original content is an obvious example. Unlinking a record from a larger context may render it unrecoverable, as can storage on unreliable media. Loss of an encoding key may result in effective destruction. Finally, unauthorized parties must be prevented from gaining access to sensitive data.

The threat of data compromise increases in the cloud, due to the number of and interactions between risks and challenges which are either unique to cloud, or more dangerous because of the architectural or operational characteristics of the cloud environment.

In cloud, data leakage exploits placement vulnerability i.e. launching instances till we find one that is co-resident with the victim instance (by brute force or by using a strategy) . After this, the attacker uses cross-VM leakage by extracting keys using side channels or uses DoS to force victim to use the common channel.

Solutions:

a)Data Loss Prevention: a mechanism to identify sensitive information by content, whether it's data being transmitted (through email etc), data on server or data at an endpoint, and prevent it from leaking outside.

b) Enterprise Rights Management: applies Digital Rights Management to corporate documents to permanently control access to documents whether they are inside or outside the company

c) Making work place secure:

- USB: erasing data if USB is lost, file redirect technology (data can be copied only to company prescribed USB)

- Safe data movement in PC: 'your PC anywhere' model for governance, requires fingerprint authentication and device integrity check

d) Letting users choose which virtual machine their instance should run on and only they can access that machine

3) Incomplete Data Deletion:

When a request is made to delete a cloud resource [46], as with most operating systems, it may not result in true wiping of the data . Adequate or timely data deletion may also be impossible (or undesirable from a customer perspective), either because extra copies of data are stored but are not available, or because the disk to be destroyed also stores data from other clients. In the case of multiple tenancies and the reuse of hardware resources, this represents a higher risk to the customer than with dedicated hardware.

Solutions: A robust, centralized management system to overlook the functions of distributed cloud applications (but this will lead to performance bottlenecks).

4) Data Consistency and Integrity:

This problem deals with the cloud service provider's ability to store and retrieve data in its correct and complete form. When a user sends a query, CSP should be able to retrieve the correct data and supply it to the user. Clouds usually only provide eventual integrity and not immediate, which means that user cannot always be sure of the integrity of the data that has been updated recently.

Cloud suffers from the traditional problems of data inconsistency in distributed applications. When two users simultaneously query the same data, the CSP is responsible for ensuring consistency.

Solutions [42, 44, 45, 46, 48]:

a) Traditional database solutions (eg: Locks) b) Semaphores

c) A robust, centralized management system to overlook the functions of distributed cloud applications

d) An extensible data-driven framework, called DS2 , that provides secure data processing and sharing between cloud users. DS2 is designed for collaborative deployments in which (potentially untrustworthy) cloud users share and exchange data. The data-driven approach enables us to build upon well-studied database techniques, including database access control, query results verification of outsourced databases, distributed query engines for enforcing extensible trust management policies in the cloud ecosystem, and declarative techniques for cloud analytics. The DS2 platform provides secure query processing in a multi-user cloud environment; seamless integration of declarative access control policies with data processing to enable secure sharing among users; system analysis and forensics by capturing accurate historical records of data exchanges in the form of distributed provenance; and client end-to-end verification of data that are partitioned across both cloud nodes and cloud users. In DS2, network protocol and security policies are specified using Secure Network Datalog (SeNDlog), a declarative language primarily rooted in Datalog that unites declarative networking and logic-based access control specifications.

5) Data Sanitization:

It states that data should be removed from a device before it is retired or reused. Since, in a cloud, a user instance is placed on a virtual machine which is shared by other users as well, it is extremely important to ensure that user data is completely removed from the machine once, the user leaves the cloud or shifts to another machine. [49, 50]

Solutions:

a) There are many softwares available in the market which can be used to remove data effectively from devices. Boot & Nuke, DBAN are two such softwares which can be used to sanitize devices.

b) Companies like Apple, Android provide technology to erase data remotely in case of theft etc.

6) Loss in Governance and Compliance Risk:

This threat arises due to lack of governance over audits and industry standard assessments [51]. Due to this, customers of cloud services do not have a view into the processes, procedures and practices of the provider in the areas of access, identity management and segregation of duties. Organizations that seek

to obtain certification, may be put at risk because cloud computing service providers may not be able to provide evidence of their own compliance with the necessary requirements or may not permit an audit by cloud customer.

Solutions:

a) Trusted cloud computing: installing monitors on CSP which will provide proofs of compliance to data owner [53]

b) Service level agreements [52]: To ensure guarantees from cloud service providers for service delivery, businesses using cloud computing services typically enter into service level agreements (SLAs) with the cloud service providers. Although SLAs vary between businesses and cloud service providers, they typically include the required/agreed service level through quality of service parameters, the level of service availability, the indication of the security measures adopted by the cloud service provider and the rates of the services.

2.5.3 Threats to Network in Cloud

1) Cross site scripting (XSS) attacks

It is an attack against web applications in which scripting code is injected into the output of an application that is then sent to a user's web browser. In the browser, this scripting code is executed and used to transfer sensitive data to a third party.

Types of XSS attacks:

• DOM based or local XSS- Precondition: the vulnerable page uses data from the document.location, document.URL or document.referer properties. The payload is never located in the html but in the URL

• Non-persistent or reflected XSS- Such holes show up when data provided by a web client is immediately used by the server to generate a page of result.

• Stored, persistent or second-order XSS- The payload is stored on the server

Solutions: [55, 56]

a. Client Side implementation using Noxes

b. Dynamic Data Tainting and Static Analysis (Client side implementation)

c. Static Detection of Cross-Site Scripting (Untrusted and insufficiently checked untrusted data)

d. Noncespaces

2) Distributed Denial of Service Attack

A very old kind of attack, when used in the area of cloud computing, means that the Cloud becomes inaccessible /difficult to access due to over-usage of its resources by malicious entities (users). Any virtual machine can attack its neighbour in same physical infrastructures and thus prevent it from providing its services or, which has been known as denial of service attack DoS attack as has been existed in AWS Amazon, that kind of attack can effect on cloud performance in general and can cause financial Losses and can cause harmful effect in other servers in same cloud infrastructure as in.

The various kinds of attacks include:

a. Flooding the pipes vs. exhausting the servers

b. Adversary Analysis

– Protection racketeers (extortion of money)

– Hacktivists

– Cyberwar

– Exfiltrators (diversion from a real attack)

– Competitors

– Success (eg. Over-excitement of users with launch of new i-phone)

c. Economic DoS – The elasticity of the Cloud computing can be misused to strain the actual user economically

Solutions:

Several kinds of flooding DoS attacks detecting approach has been suggested in Mohd Nazri Ismail, Abdulaziz Aborujilah, Shahrulniza Musa & AAmir Shahzad [54].

One of these methods has proposed a kind of pattern generation for spatial-temporal traffic pattern in the application layer according to document popularity and also Access Matrix and behavior of web access. This method depends on semi-Markov modeling, and potential DoS attack is identified by entropy of document popularity, which matching the model.

3) Network Sniffing

Sniffers are programs that allow a host to capture any network packet illicitly. Detection of sniffer attacks is very difficult task to handle. Specially, if the sniffers are active because active sniffer can alter or block network traffic while passive sniffer can only monitor network traffic.

Ways of sniffing: [57]

a. A host running a sniffer sets its NIC in promiscuous mode so it receives all packets whether targeted to it or not esp. in broadcast environment

b. ARP Cache poisoning – ARP cache poisoning depends on local ARP cache maintained by each host of network. This cache contains IP with corresponding Media Access Control (MAC) addresses of recently accessed hosts in non-broadcast environments.

Solutions:

a. ARP, RTT and DNS Detection Techniques
b. ARP Cache Poisoning Detection Technique

4) Man in the Middle Attack

It is a type of Network Sniffing Attacks that is especially used in cryptographic instances. Abbreviated as MITM, a man-in-the-middle attack is an active Internet attack where the person attacking attempts to intercept, read or alter information moving between two computers. MITM attacks are associated with 802.11security, as well as with wired communication systems.

Solutions: [58]

a. Reverse ARP poisoning with active IP probing

b. IP probing with CAM table poisoning

5) Port Sniffing

There may be some issues regarding port scanning that could be used by an attacker as Port 80(HTTP) is always open that is used for providing the web services to the user. Other ports such as 21(FTP) etc are not opened all the time it will open when needed therefore ports should be secured by encrypted until and unless the server software is configured properly.

Counter measure for this attack is that firewall is used to secure the data from port attacks.

Types of Port Sniffing [59]

Address Resolution Protocol (ARP), The Vanilla TCP connect scan, The TCP SYN (Half Open) , The TCP FIN scan, The TCP Reverse Ident scan, The TCP XMAS scan, The TCP NULL scan, The TCP ACK scan, The FTP Bounce Attack, The UDP ICMP port scan, The ICMP ping-sweeping scan

Solutions: [59]

a. TCP Wrappers - they will reject the incoming connection if it is not originated from an approved host or domain.
b. PortSentry offered by Psionic - PortSentry detects connection requests on a number of selected ports. PortSentry is customizable and can be configured to ignore a certain number of attempts.

2.6 Background and Related Work

In the course of our research we found that a major Security Challenge faced while operating a Cloud is Assured Deletion and Version Control

2.6.1 Replication of Data

Data is replicated almost a thousand times by cloud providers for many reasons, including [60]

- Designing Efficient Multi-Copy Provable Data Possession, (EMC-PDP) which efficiently and securely provide the owner with strong evidence that the CSP is in reality possessing all data copies that are agreed upon and these copies are intact.
- Seemless access of the file
- Minimal Computational Complexity from the sides of both Server and Client
- Limiting the bandwidth required as it enables access over more than one connection
- Enabling Public Verifiability
- Supporting *blockless* verification
- Allowing unbounded number of auditing rather than imposing a fixed limit
- Facilitating stateless verification where the verifier is not needed to hold and upgrade state between audits. Maintaining such state is unmanageable in case of physical damage of the verifier's machine or if the auditing task is delegated to a TPA.
- Enabling both probabilistic and deterministic guarantees. In probabilistic guarantee the verifier checks a random subset of stored file blocks with each challenge (spot checking), while the verifier checks all the stored file blocks in the deterministic guarantee.

Having and maintaining these many number of copies of the data raises the concern that

- When the Actual User asks for the data to be deleted, the data should become inaccessible to all
- The command has to be given by the actual user

2.6.2 Assured Deletion and Version Control

Assured Deletion – Data files that have been requested by the user to be deleted should become permanently inaccessible. Keeping data backups permanently is undesirable as sensitive information may be exposed in future due to data breach or cloud mismanagement.

This Security Concern often gets interlinked with another concern, that is, *Version Control*. Typically each back-up version is made from a previous version. Deleting the old version may make the present version unrecoverable.

2.6.3. Approaches for Algorithms Regarding Data Storage

Secure Overwriting

The old data is overwritten by the new data to make the old data unrecoverable. [61] The general concept behind an overwriting scheme is to flip each magnetic domain on the disk back and forth as much as possible (this is the basic idea behind degaussing) without writing the same pattern twice in a row. If the data was encoded directly, the desired overwrite pattern of ones and zeroes can be written on it repeatedly. However, disks generally use some form of run-length limited (RLL) encoding, so that the adjacent ones won't be written. This encoding is used to ensure that transitions aren't placed too closely together, or too far apart, which would mean the drive would lose track of where it was in the data.

The main disadvantages of this approach are [62] the fact that it requires internal modifications of a file system and is not feasible for outsourced storage since the storage backend is maintained by third parties. Thus, it has no guarantee that replicated data will be over-written. Moreover, they are susceptible to attacks such as

- the Distributed Denial of Service attack[63] where an attack on the network disables the cloud to service its clients, in this case to carry out the complete removal of all copies of the replicated data;
- and Incomplete Data Deletion[64] when a request to delete a cloud resource is made, as with most operating systems and this may not result in true wiping of the data. Adequate or timely data deletion may also be impossible (or undesirable from a customer perspective), either because extra copies of data are stored but are not available, or because the disk to be destroyed also stores data from other clients. In the case of multiple tenancies and the reuse of hardware resources, this represents a higher risk to the customer than with dedicated hardware.

Cryptographic Protection

This approach removes the cryptographic keys that are used to decrypt data blocks to make the encrypted blocks unrecoverable. The encrypted data blocks are stored in outsourced storage (e.g., clouds), while the cryptographic keys are kept independently by a key escrow system. For instance, FADE[65] supports policy-based assured deletion, in which data can be assuredly deleted according to revoked policies.

However, the disadvantage is that existing studies do not consider version control for this approach. Existing version control systems and assured deletion systems are incompatible with each other.

In the security context, recent studies propose ***convergent encryption***[66] such that the key for encrypting/decrypting a data chunk is a function of the content of the data chunk, so that the encryptions of two redundant data chunks will still return the same content. However, in convergent encryption, if we want to assuredly delete a data chunk of a particular version, we cannot simply remove its associated key, since it may make the identical chunks in other versions unrecoverable.

Deduplication, also known as single-instance storage, has been utilized as a method for maximizing the utility of a given amount of storage. Deduplication identifies common sequences of bytes both within and between files (chunks), and only stores a single instance of each chunk regardless of the number of times it occurs. By doing so, deduplication can dramatically reduce the space needed to store a large data set. Data security is another area of increasing importance in modern storage systems and, unfortunately, deduplication and encryption are, to a great extent, diametrically opposed to one another. Deduplication takes advantage of data similarity in order to achieve a reduction in storage space. In contrast, the goal of cryptography is to make ciphertext indistinguishable from theoretically random data. Thus, the goal of a secure deduplication system is to provide data security, against both inside and outside adversaries, without compromising the space efficiency achievable through single-instance storage techniques.

- First, convergent encryption is utilized to enable encryption while still allowing deduplication on common chunks. Convergent encryption uses a function of the hash of the *plaintext* of a chunk as the encryption key: any client encrypting a given chunk will use the same key to do so, so identical plaintext values will encrypt to identical ciphertext values, regardless of who encrypts them. While this technique does leak knowledge that a particular ciphertext, and thus plaintext, already exists, an adversary with no knowledge of the plaintext cannot deduce the key from the encrypted chunk.
- Second, all data chunking and encryption occurs on the client; plaintext data is never transmitted, strengthening the system against both internal and external adversaries.

- Finally, the map that associates chunks to a given file is encrypted using a unique key, limiting the effect of a key compromise to a single file. Further, the keys are stored within the system in such a way that users only need to maintain a single private key regardless of the number of files to which they have access.

2.7 FadeVersion – The Deciding Algorithm [67]

FadeVersion is a secure Cloud backup system that supports both, version control and assured data deletion.
The main idea is to use the approach of LAYERED ENCRYPTION.
Suppose that File F appears in Multiple Versions, we encrypt it using Key *k*. Then we encrypt *k* using many different keys 'c_i' (i=1,2,…,n) where *n* is the total number of versions and *i* corresponds to the version specified. Thus, even if the key of one version is removed, the data still exists and hence can be accessed by the versions that need to access it.

2.7.1 Working Model

The goal is to make both version control and assured deletion compatible with each other in a single design. The main idea of FadeVersion is as follows. The design of a version-controlled cloud backup system is that in which data objects are created that are to be archived on the cloud. On top of the version control design, a layered approach of cryptographic protection is added, in which data is encrypted with the first layer of keys called the data keys, and the data keys are further encrypted with another layer of keys called the control keys. The control keys are defined by fine-grained policies that specify how each file is accessed. If a policy is revoked, then its associated control key is deleted. If the data object is associated solely with the revoked policy, then it will be assured deleted; if the data object is associated with both the revoked policy and another active policy, then we still allow the data object to be accessed through the active policy.

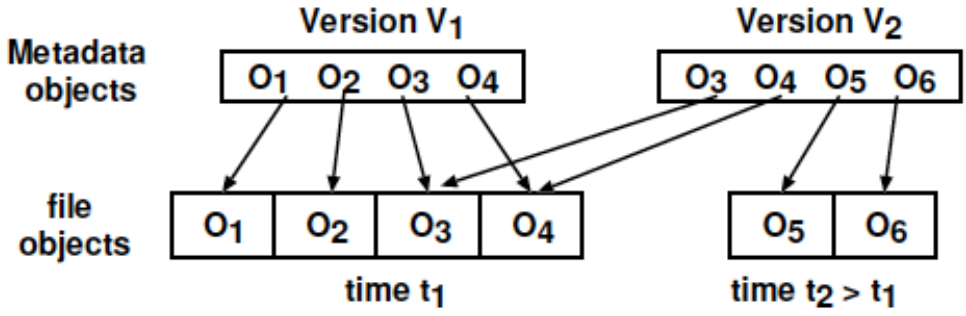

Figure 2.3a: Assured Deletion

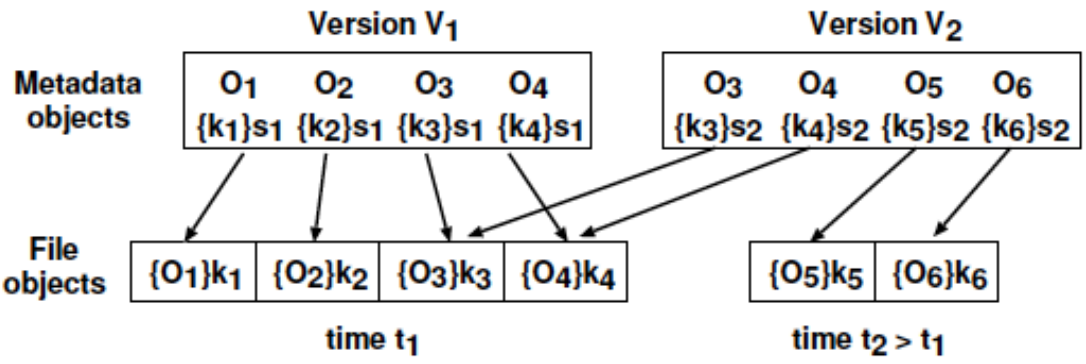

Figure 2.3b: Implementation of the Layered Approach

2.7.2 Implementation Details

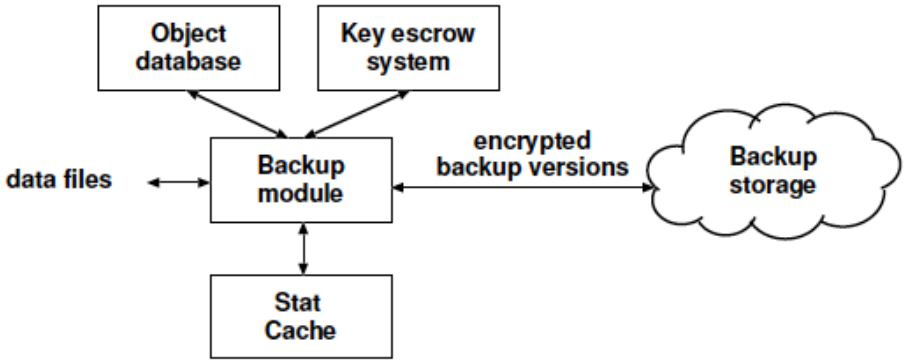

Figure 2.4: Architecture of the FadeVersion System

```
name: fileA
checksum: sha1=...
ctime: 1300000000
data: A/1
    A/2
    A/3
group: 1 (root)
inode: ...
key: ENCRPYTED KEY FOR A/1
    ENCRYPTED KEY FOR A/2
    ENCRYPTED KEY FOR A/3
mode: 0755
mtime: 1300000000
size: 3000000
type: f
user: 1 (root)
```

Figure 2.5: The Metadata Format for a Single File

Figure 2.7 shows the metadata formats for a single file in Cumulus and FadeVersion, assuming that the file contains three file objects (i.e., A/1, A/2, A/3). In FadeVersion, they add an additional field named key, which stores the data key of each associated data object. The data key is encrypted with the control keys of the corresponding policies, and the control keys are kept by the key escrow system. In the prototype, each file object is associated with three policies:

(i) user-based policy, which is described by the user field,

(ii) file-based policy, which is described by the name field, and

(iii) version-based policy, which is described by the version in which the file resides.

Based on the information, FadeVersion can know how to restore a file, i.e., by using the correct control keys from the key escrow system to decrypt the data keys, and how to revoke a policy and its associated files.

2.7.4 Key Management

The control keys are maintained by a key escrow system, which they assume can securely remove the control keys associated with revoked policies to achieve assured deletion. On the other hand, it is still important to maintain the robustness of the existing control keys that are associated with active policies.

- One approach is by encrypting all control keys with a single master key, while this master key is stored in secure hardware[69].
- Another approach is by using a quorum scheme based on threshold secret sharing. Each control key is split into N key shares and are distributed to N independent key servers, such that we need at least K<N of the key shares to recover the original control key[70].

2.7.4 Drawbacks of the Algorithm

The drawbacks of this algorithm as follows:

- There can be a Triple Key that can be used to break the Safety of the Algorithm
- There is no mention of memory management and as to what happens with the physical memory

ADLE, our Algorithm has dealt with these drawbacks in the following ways:

- The Data is hashed and then, that is used to make the Key which makes the key become subjective to the data content which renders the key hard to guess.
- We have dealt with memory management with the help og garbage collection.

CHAPTER 3

IMPLEMENTATION

3.1 Problem Statement

When a request is made to delete a cloud resource [46], as with most operating systems, it may not result in true wiping of the data. Adequate or timely data deletion may also be impossible (or undesirable from a customer perspective), either because extra copies of data are stored but are not available, or because the disk to be destroyed also stores data from other clients. In the case of multiple tenancies and the reuse of hardware resources, this represents a higher risk to the customer than with dedicated hardware. Replication is necessary for two reasons: data backup (security) and performance enhancement.

Incomplete Data Deletion poses a huge problem that could undermine the integrity of a cloud system. Out of Confidentiality, Integrity and Authentication, IDD is a threat to all. It can cause a breach of confidentiality of data, if leaked. Also, it undermines the integrity of the cloud if all the said functions in a cloud cannot be carried out. It may also lead to issues in Data Recovery and thus has to be handled.

However, the only solutions so far to counter Incomplete Data Deletion consists of a robust, centralized management system to overlook the functions of distributed cloud applications (but this will lead to performance bottlenecks).

Considering it to be an issue of major importance, we decided that the following issues be dealt with in the corresponding ways:

- The confidentiality of a user's data be maintained by encrypting the data before storng it in the cloud. If the data is wished by the user to be deleted then its key to be removed permanently to avoid the data to be decrypted in future.
- The integrity of the stored data be maintained by keeping the keys in a separate third party trustable environment. This would also help in removal of bottlenecks.
- Authentication of Data Deletion by a correct user be done by the fact each user operating on the cloud has to sign in with a unique username and password. The unique username is used as part of an encryption key that helps maintain the authentication of the data pertaining to that user.

Hence the problem of Incomplete Data deletion is solved through the removal of the key used to encrypt the data that is dependent on the data and the user.

3.2 Structure of the Algorithm – ADLE (ASSURED DELETION USING LAYERED ENCRYPTION)

2.3.1 Loading of Data to the Cloud: Encryption

- The User uploads/interacts with the cloud and inputs data. . A transaction ID is generated which uniquely identifies a file on the server.

- This data is read and it undergoes two levels of encryption, first with Key 1: the Data Key and then further encrypted with Key 2: the User Key.
- This data is stored in the cloud in its encrypted form.
- The Data key is saved safely in a third part key escrow system. It can be indexed using a combination of Transaction ID(unique to every file) and Username + password (or User Key, which is unique to every user)

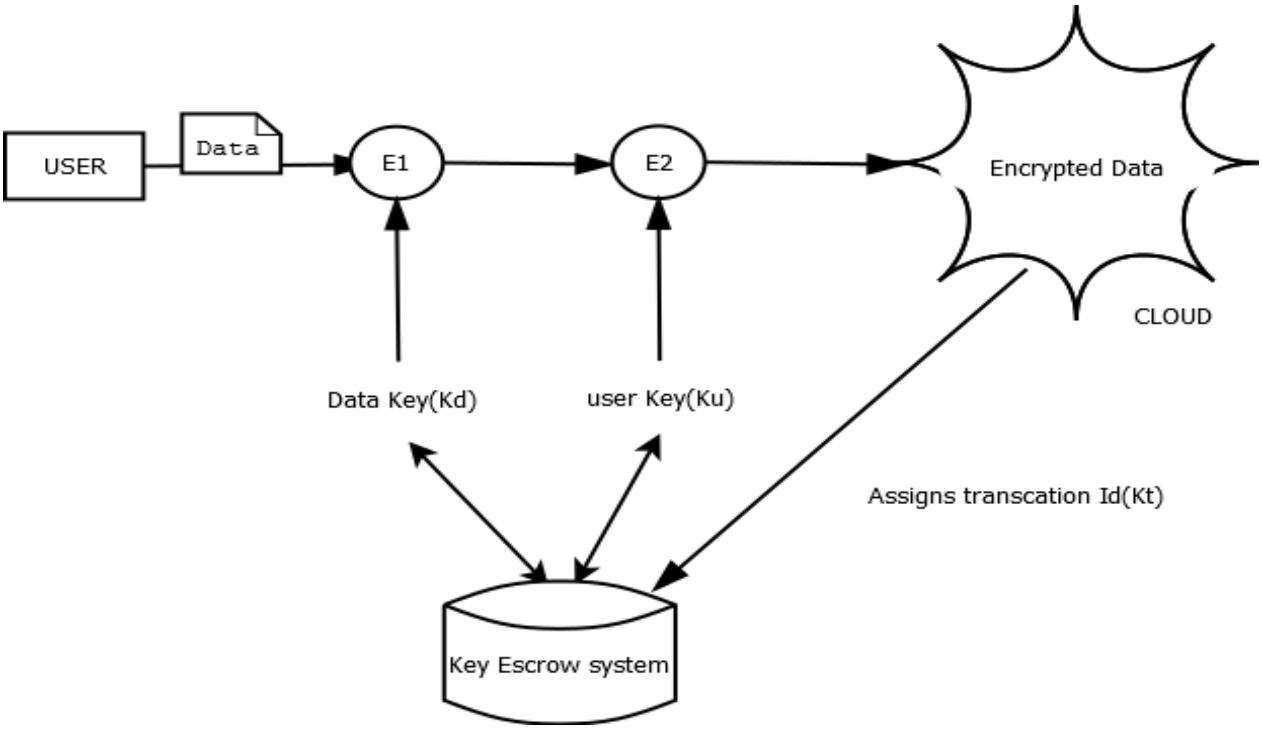

Figure 3.1: Insertion

3.2.2 Retrieval of Data by User upon Sign in Only

- The user signs. The unique username and password gives us the User Key (Key 2).
- She then specifies the exact data she wishes to retrieve from the cloud from the list she is presented.
- This gives the system Transaction ID which is generated by the cloud every time a new data is uploaded into it and is stored along with Ku and Kd in the key escrow system.
- Using the username and transaction ID the system accesses the Key Escrow system and the corresponding Kd is obtained.
- If the Kd(for this combination of Ku and Kt does not exist), the user gets an error message.
  If Kd is successfully obtained, it is used to decrypt the data and the data is supplied to the user.

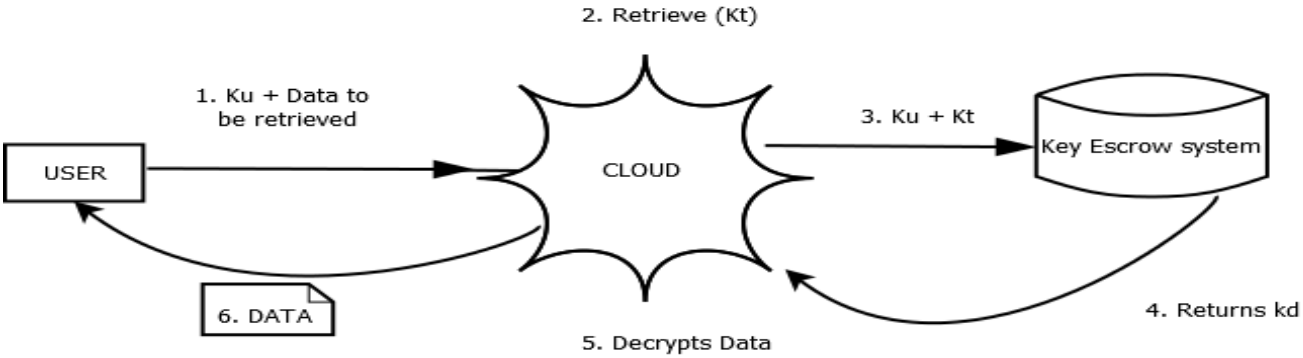

Figure 3.2: Retrieval

3.2.3 Deletion of Data by User upon Sign in Only

- The user signs in. The system can obtain the User Key (i.e the user name+password combination)
- She then specifies the exact data she wishes to delete from the cloud.
- This gives the system the Kt, (which is generated by the cloud every time new data is uploaded) into it and is stored along with Ku and Kd in the key escrow system.
- Both the Kt and User Key are then fed into the Key Escrow system and the corresponding Kd is obtained.
- If the Kd for this combination of Ku and Kt does not exist, the user gets an error message.
- If Kd is successfully obtained, it is promptly deleted from the key escrow system. Each entry of Ku+Kt without a Kd has a lifetime, which is at the discretion of the third party in charge of the key escrow system.
- A message is sent to the cloud, which stores the fact that a Kd has been deleted.
- The Memory Management System moves to accommodate the changes in the cloud as specified later in the text.
- The user gets a confirmation message as to the successful removal data and all its copies from the database of the cloud.

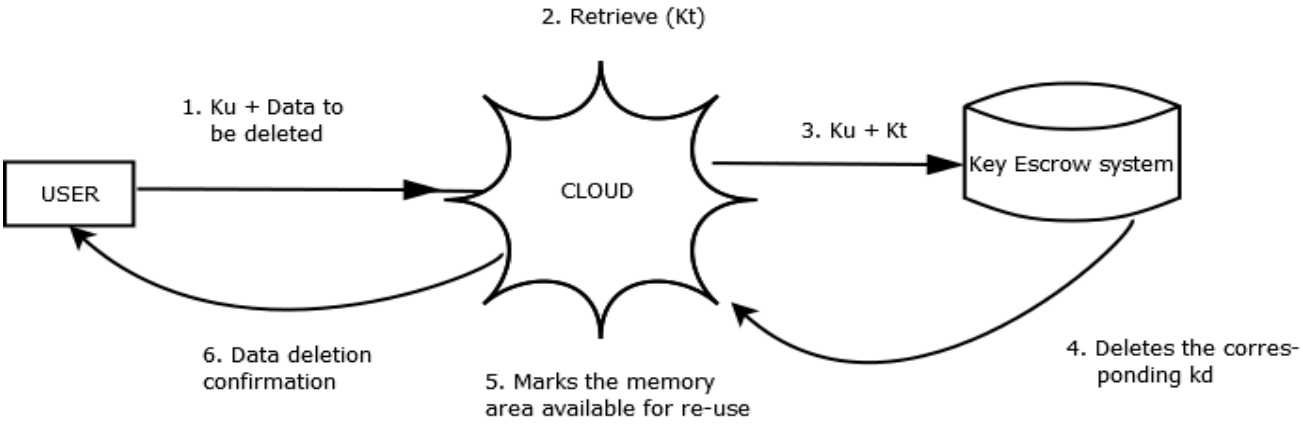

Figure 3.3: Deletion

3.2.4 Key Exchange between the Cloud and the Key Escrow System

- The Username and Password combination of a User proved her with a unique identity while dealing with the cloud. Each User has a unique password. The User signs in with her Username and Password. Each time, this generates her User Key Key1.
- When the user uploads data into the cloud, with the condition of no modification, a unique Transaction ID is generated by the Cloud-Server, called Key2. This means, that once uploaded, a user cannot modify the data (No Versions Available). Every time, she has to upload a change in the form of new data that gets its own Transaction ID.
- The System Input Key, FKey is generated by combining the User ID and Transaction ID, Key1 and Key2.
- The data undergoes encryption utilizing the Data Key, UKey.
- Both FKey and Ukey are then stored in the Key Escrow System in an array kind of data structure with the two keys being interdependent.
- For retrieval and deletion, the System Key is again generated through the same means as before. Since Hash Functions are always one way, hence they always give the same Key.
- This FKey is entered and the corresponding UKey (if it exists) is obtained and returned or deleted as the case may be.

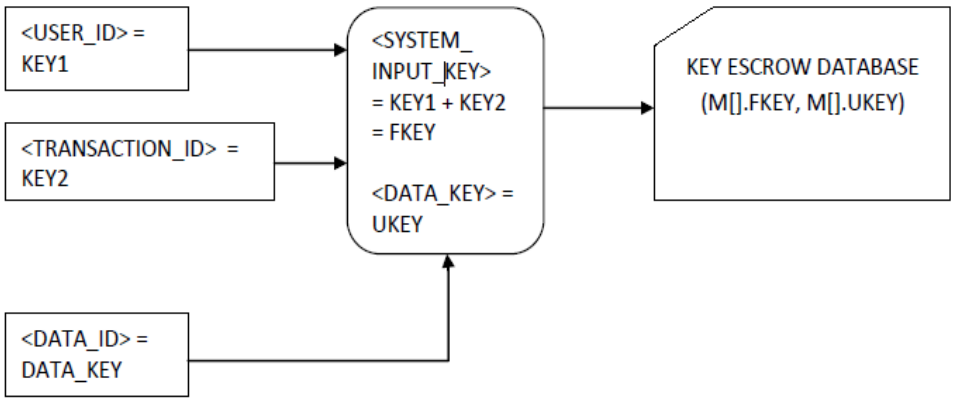

FIGURE 1: Insertion of the keys into the Key Escrow Database System

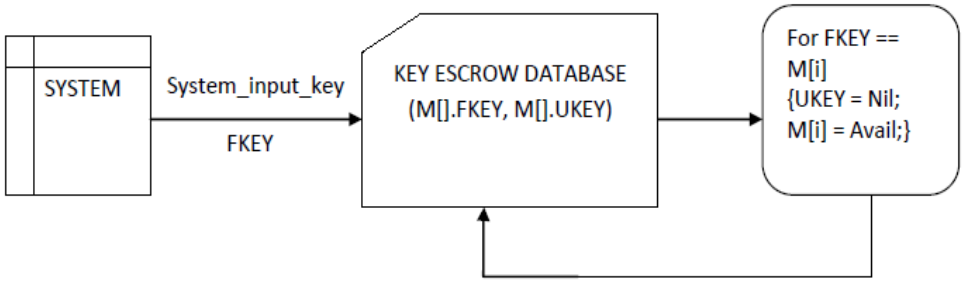

FIGURE 2: Deletion of the data keys from the Key Escrow Database System with respect to the input key by the system

Figure 3.4 (1 and 2): The Key Escrow System

3.3 THE ALGORITHM: ADLE

Variables : K_u = user Key , K_d = Data Key , K_t = Transaction key

1. Display User Login Page
2. Print: “Enter username and password”
3. Scan: User details
4. IF (user details == incorrect)
   4a. then (ask user to login again)
   4b. ELSE Display Home which gives users options such as – Upload, retrieve and delete
      4b(i)IF user clicks upload call routine UPLOAD()
      4b(ii)IF user clicks retrieve call routine RETRIEVE()
      4b(iii)IF user clicks upload call routine DELETE()

UPLOAD Routine

1. User clicks ‘Browse’ and uploads file from his local computer
2. File content is extracted to generate K_d using MD5 hash function
3. The transaction ID K_t is generated by the system
4. Store K_d, K_t, K_u in Key Escrow System
5. Data is encrypted first using K_d and stored as encrypted_data_1
6. encrypted_data_1 is encrypted using K_u ,replicated and stored in cloud
7. IF storage is successful Redirect user to Home

RETRIEVE Routine

1. Display all of the files that user has uploaded. If user has not uploaded any file, display error message
2. User selects one file for download
3. Retrieve Kd from Key Escrow System using the file’s K_t
4. Using K_t get encrypted Data from cloud
5. Encrypted Data is decrypted first using K_u and stored as decrypted_data_1
6. decrypted_data_1 is again decrypted using K_d
7. The final decrypted data is offered to the user for download using appropriate size and file type arguments
8. If successful Redirect user to Home

DELETE Routine

1. Display all of user's files.
2. Using the file's unique K_t, send command to Key Escorw System
3. Key Escrow System deletes K_d from its database
4. Get acknowledgement
5. if successful, Redirect user to Home

3.4 Encryption/Decryption Used In ADLE

Cryptographic algorithms are utilized for security services in various environments in which low cost and low power consumption are key requirements[71].

We have used the following encryption standards in our algorithm:

- Data is stored as encrypted text using **AES**, with 128 bit key. The Data key is Key1 for the 1st level of Encryption and the User key is Key2 for the second level of encryption.
- Data key is generated using **MD5 Hash function** by hashing the Data content. Since hashing is a Many to One mapping, the hash produced is same every time ensuring same key.

The Key Characteristics of our System are:

- Encryption of data – Ensures privacy and confidentiality. The data is accessible only to the user. It is not even visible to the cloud because it has been encrypted before storing and can be accessed only using a combination of Userkey and Datakey. The Datakey, Kd is stored with Key Escrow System and is out of reach of the cloud.
- Two levels of encryption – Ensures that if one of the keys is compromised, another is safe with the key escrow system (which stores the data key only). This adds multiple level of security against attacks and makes the system robust. Even if the attacker knows the Username, he must have access to the password to generate the Userkey, Ku. If somehow the attacker gets hold of this combination, he must be able to crack the security of the Key Escrow System and get the Datakey, Kd which is highly unlikely.
- Using User ID – Ensures Access Control i.e. another user cannot fake identity of legitimate user because the combination of the Username and Password generates UserKey Ku, which decrypts the data.

3.4.1 AES Encryption/Decryption

AES[72] is a symmetric cipher that processes data in 128-bit blocks. It supports key sizes of 128, 192, and 256 bits and consists of 10, 12, or 14 iteration rounds, respectively. Each round mixes the data with a roundkey, which is generated from the encryption key. Decryption inverts the iterations resulting in a partially different data path.

Double Key Double AES[73]

The Advanced Encryption Standard (AES) involves a new strong encryption algorithm. It works with two *blocks* of 128 bits. Given a message block *p* (plaintext) and a key block *k*, the AES encryption function *E* returns an encrypted block *c* (ciphertext):

$c = E(p, k)$.

The inverse of the AES encryption function *E* is the decryption function *D* such that
$D (E(p, k), k) = p$, $E (D(c, k), k) = c$.

In *Double AES*, two independent key blocks k_1 and k_2 are used in succession, first k_1, then k_2:
$c_2 = E (E(p, k_1), k_2)$.

Double AES is the simplest variant. In Double AES, we use two keys K1 and K2 and the ciphertext C and plaintext P are computed as follows.[74]
C = EK2 (EK1 (P))
P = DK1 (DK2(C))
This Encryption Method is considered to be one of the strongest and has not been broken till now.

3.4.2 MD5 Hash Function
The MD5 algorithm (Message Digest 5) is a cryptographic message digest algorithm.
MD5 was designed by Ron Rivest, who is also the `R' in `RSA' in 1991. MD5 is described in rfc1321. C source code is included with the RFC. It is basically MD4 with "safety-belts" and while it is slightly slower than MD4, it is more secure.

The algorithm consists of four distinct rounds, which have a slightly different design from that of MD4. Message-digest size, as well as padding requirements, remains the same.

The MD5 algorithm takes as input a message of arbitrary length and produces as output a 128-bit "fingerprint" or "message digest" of the input. It is conjectured that it is computationally infeasible to produce two messages having the same message digest, or to produce any message having a given pre-specified target message digest. The MD5 algorithm is intended for digital signature applications, where a large file must be "compressed" in a secure manner before being encrypted with a private (secret) key under a public-key cryptosystem such as RSA or PGP.

3.5 Memory Management

Garbage Collection is a term associated with the use of solid-state drive (SSD) technology in data storage applications. One of the biggest issues is the way these drives collect garbage and delete it from data blocks prior to a write operation. Garbage collection, as it is known in the industry, improves write performance by eliminating the need to perform whole block erasure prior to every write. Working in the background, garbage collection accumulates data blocks previously marked for deletion, performs a whole block erasure on each "garbage" block, and returns the reclaimed space for reuse by subsequent write operations.
Data in cloud computing is very similar to memory collection in object oriented languages and when it's not done automatically by the framework, temporary data tends to leak. In particular, in cloud computing, it has been found that it's pretty easy to end up with *storage leaks* due to:

- Collection omission
- Application crash
- Service interruption

The Algorithm used is as follows:

In order to setup a Windows 7 machine to run cron.php (or any web page) at a specific time, creating a Scheduled Task

- Open Scheduler (type Sched into the search box and it'll appear) OR go to Start > Programs > Accessories > System Tools > Scheduled Tasks
- Use 'Create Task'.
- The Scheduled Task Wizard will appear. Give the task you wish to generate a name.
- Click the 'Actions' tab then click 'New' to 'Start Program'.
- In the 'Program/script' box click browse to your favourite browser and select its executable file (eg: firefox.exe)
- In the 'Add arguments' box put the address of your cron.php file, eg:http://www.mysite.com/cron.php or http://localhost/mysite/cron.php
- Go to the 'Triggers' tab and select any time once per day (or whatever frequency is preferred) then fine tune the conditions from within the 'Settings' tab.

3.6 Identity and Access Management

Managing identities and access control for enterprise applications [10] remains one of the greatest challenges facing IT today. While an enterprise may be able to leverage several Cloud Computing services without a good identity and access management strategy, in the long run extending an organization's identity services into the cloud is a necessary precursor towards strategic use of on-demand computing services. Supporting today's aggressive adoption of an admittedly immature cloud ecosystem requires an honest assessment of an organization's readiness to conduct cloud-based Identity and Access Management (IAM), as well as understanding the capabilities of that organization's Cloud Computing providers.

Hence, we have implemented the following Concepts:

- Each user is choses a unique username and a password that it uses to sign in. The Cloud server authenticates it.
- The Username acts as the precursor for the User ID Key.
- Each time the user uploads data on the cloud, that instance of transaction is given its automatically server generated Transaction Key.
- The data is Hashed to produce its Data Key which is a one way key to enable authentication of Data.
- The combination of the User Key, Transaction Key and Data Key provides the required Access Control and Identity Management for both Cloud and the Key Storage System.

3.7 Key Escrow System

The control keys are maintained by a key escrow system, which they assume can securely remove the control keys associated with revoked policies to achieve assured deletion. On the other hand, it is still important to maintain the robustness of the existing control keys that are associated with active policies.

The system indexes each of the incoming System Generated Keys and sorts them into one of the two linked and corresponding columns: The User Column and the Data Column.

CHAPTER 4

PERFORMANCE ANALYSIS

4.1 Software Requirement

ADLE requires the following:

- Server Operating System: Windows XP/7
- Technology: PHP
- Interpretation Language: HTML 4.0
- Diagramming Tool: Adobe Photoshop/Microsoft Paint
- Browser: IE 5.0 or above/ Chrome/ Mozilla
- 3rd Party Component: None
- Hardware Interfaces: None
- Localhost Server: Wamp Server
- Text Editor: Notepad++
- RDBMS: SQL Server 2000

The project is coded in PHP and results are obtained at the Localhost accessing MySQL as storage space.

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1995, the reference implementation of PHP is now produced by The PHP Group.

MySQL is (as of 2008) the world's most widely used open source relational database management system (RDBMS) that runs as a server providing multi-user access to a number of databases. MYSQL allows users to access the database using command line , graphical interface or programming.

While testing, the file size is kept varied from 1MB to approximately 70 MB. Process of garbage collection is done by windows task scheduler/file system.

In this section of testing, we shall compare our proposed approach with traditional approach used at cloud's storage end where data is stored without any encryption and hence the problem of incomplete data deletion may follow.

The date is stored in a special field type provided by MySql "BLOB" - a field type for storing Binary Data. BLOB values are treated as binary strings (byte strings). They have no character set, and sorting and comparison are based on the numeric values of the bytes in column values.

MySQL has four BLOB types:

- BLOB
- TINYBLOB
- MEDIUMBLOB
- LONGBLOB

BLOB columns, there is no padding on insert and no bytes are stripped on select. BLOB columns cannot have DEFAULT values. Each BLOB value is represented internally by a separately allocated object. This is in contrast to all other data types, for which storage is allocated once per column when the table is opened.

Different Types of file formats used in the testing process:

- *Text Files*: .txt is a file format for files consisting of text usually containing very little formatting (e.g., no bolding or italics). A text is a kind of computer file that is structured as a sequence of lines of electronic text. A text file exists within a computer file system. The end of a text file is often denoted by placing one or more special characters known as an end-of-file marker.
- *Portable Network Graphics (PNG) Files*: It is a raster graphics file format that supports lossless data compression. PNG supports palette-based images (with palettes of 24-bit RGB or 32 bit RGB colors), grayscale images and full-color non-palette-based RGB images.
- *The Windows Installer* is a software component used for the installation, maintenance, and removal of software on modern Microsoft Windows systems. The installation information, and often the files themselves, are packaged in installation packages, loosely relational databases structured as COM Structured Storages and commonly known as "**MSI files**", from their default file extension.
- *ZIP Files*: **.zip** is an archive file format that supports lossless data compression. A .ZIP file may contain one or more files or folders that may have been compressed. The .ZIP file format permits a number of compression algorithms
- *MPEG-4 Part 14 or MP4 Files*: MP4 is a digital multimedia format most commonly used to store video and audio, but can also be used to store other data such as subtitles and still images. Like most modern container formats, it allows streaming over the Internet. The only official filename extension for MPEG-4 Part 14 files is .mp4
- *Exe Files*: exe is a computer file that ends with the extension ".exe" otherwise known as an *executable* file. When one clicks on an exe file, a built-in routine automatically executes code that can set several functions into motion. Exe files are used to install and run programs and routines.
- *JPG Files*: **JPEG** is a commonly used method of lossy compression for digital photography. The degree of compression can be adjusted, allowing a selectable tradeoff between storage size and image quality. JPEG typically achieves 10:1 compression with little perceptible loss in image quality.
- *PDF Files*: **Portable Document Format** is a file format used to represent documents in a manner independent of application software, hardware, and operating systems. Each PDF file encapsulates a complete description of a fixed-layout flat document, including the text, fonts, graphics, and other information needed to display it.

4.2 Result

In order to ensure the effectiveness of the algorithm, we have compared it to a scenario where the Data insertion, Retrieval and Deletion is a normal scenario with No Encryption or Decryption. Therefore,

- Scenario1 – A normal running Cloud with Direct Data Insertion, No Encryption/ Decryption/ Key Escrow System.
- Scenario2 – Implemented our algorithm – **the ADLE** in the Cloud.

We have used two experiment setups that compare certain qualities of the two scenarios. The third Experimental setup compares ADLE's performance for different storage types used.

4.2.1 Experimental Setup 1

We compare the time taken by the two scenarios in the following two cases:

- Data of following sizes is uploaded - <1MB, 3MB, 10MB, 20MB, 50MB and 66MB. The number of replications is kept constant as 20.
- No of replications is changed as – 10,20,30,40 and 60. The data size is kept constant as 16 MB.

The tests were run on the system described earlier and graphs were plotted for results. A brief analysis of the results follows ahead.

1A Time taken VS Size of Data. As can be seen from the Graph 1 (Figure 4.1), the time taken for uploading data to the cloud increases exponentially as the size of the data to be uploaded is increased. This is expected. However the rise is greater in Scenario 2 as compared to Scenario 1.

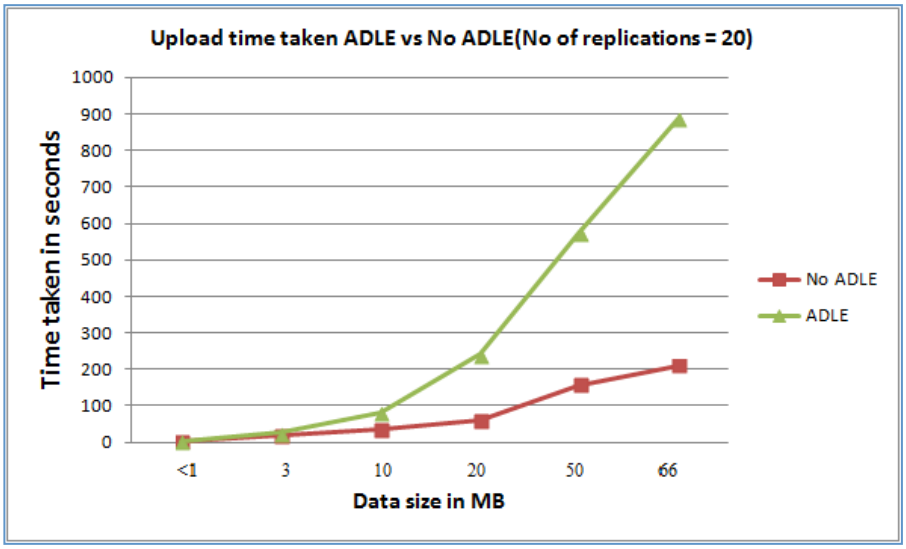

Figure 4.1

This change can be attributed to two factors –

- The size of the data to be inserted is increased when the encryption is done (as in scenario 2). This is because padding is done the plain text before encryption. Furthermore, we apply two levels of encryption. This means that padding is done twice.
- The increased size of data is inserted for N no of clouds hence the time gets multiplied by N.

Size of normal data $= S$,

Size of normal data $= S + r$

Time taken by one bit for one cloud $= t$

Time taken for N clouds $= t$ X N

Time taken by normal data $= t$ X N X s

Time taken by encrypted data on N clouds $= t$ X (S+r) X N

Therefore, time gets increased by a factor of (r X N X t).

In Graph 2 (Figure 4.2), we see that the time taken for retrieval is also greater in case of Scenario 2. This again is attributed to the fact that double decryption is carried out over a larger piece of data in the second scenario and that takes time.

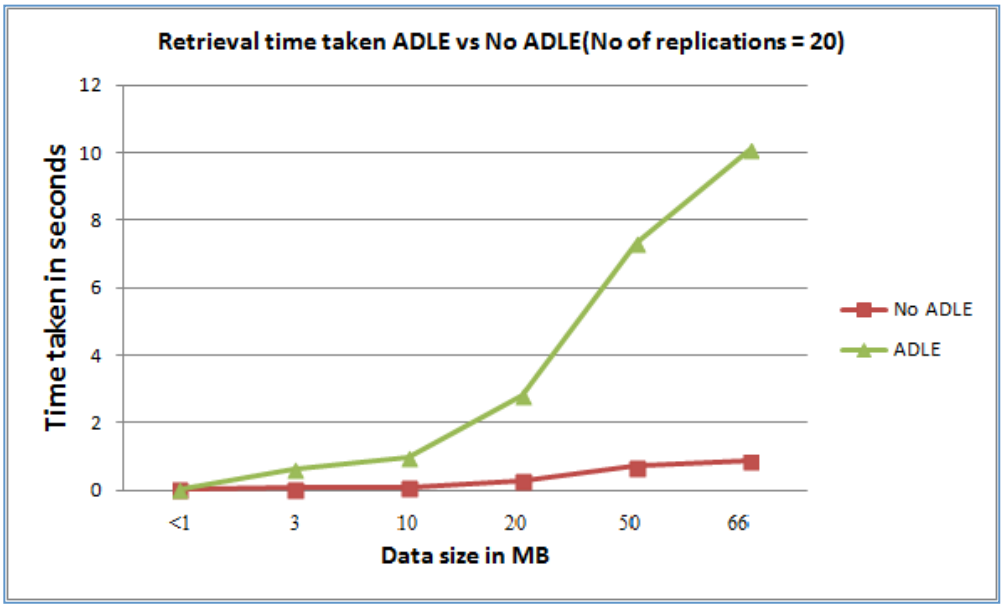

Figure 4.2

As we shall see ahead, this overhead is acceptable because Scenario 2 presents a huge advantage in terms of our actual goal, which is preventing incomplete data deletion.

In Graph 3 (Figure 4.3), we can see that the time taken for deletion in Scenario 1 rises exponentially whereas the time taken in Scenario 2 is almost constant. This is a huge advantage. Only the key needs to be deleted in second scenario to prevent access to data.

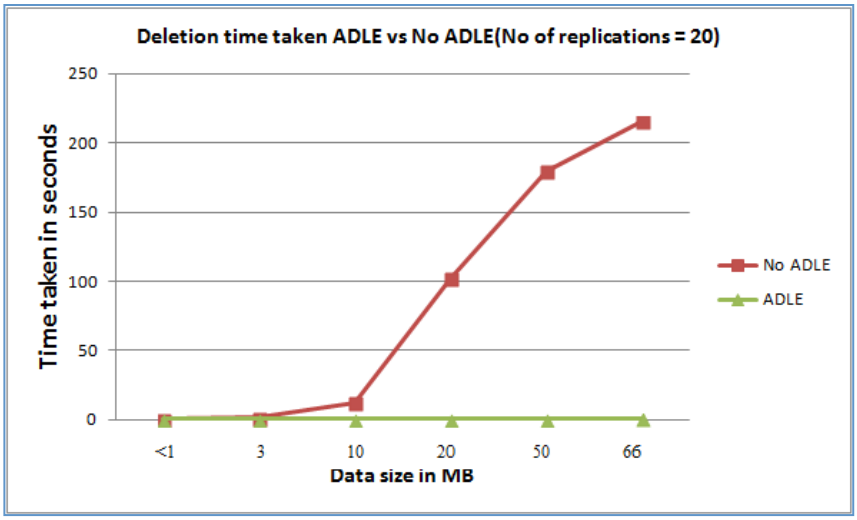

Figure 4.3

1B Time taken VS no of Replications. In Graph 4 (Figure 4.4), we have compared the upload time taken by both scenarios vs. the no of times data is replicated in the cloud. As we can see that the time is increased, both the times rise. However the difference between the time taken in both the cases is lesser this time that before and doesn't rise as drastically as before. Hence for a suitable no of replications and data size, the tradeoff in case of Scenario 2 would not be so bad.

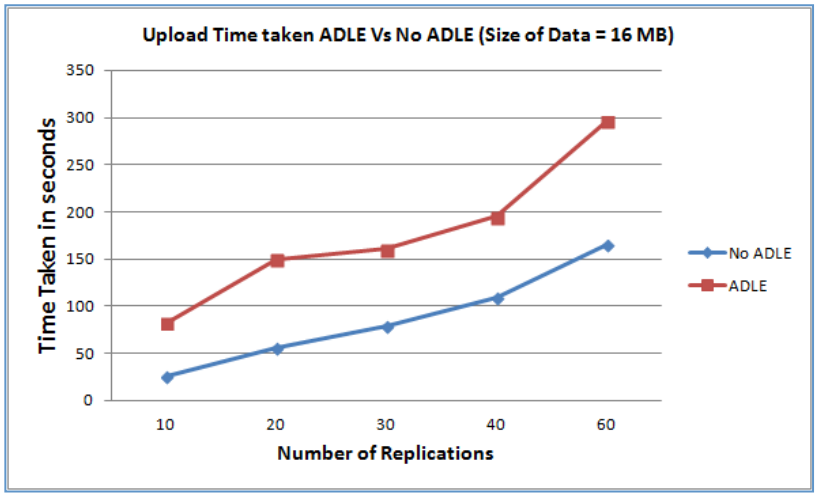

Figure 4.4

In Graph 5 (Figure 4.5), we can see that retrieval time, even though greater in case of scenario 2 is constant no matter what the number of replications be. The increase as has been mentioned earlier is because of the time taken for decryption.

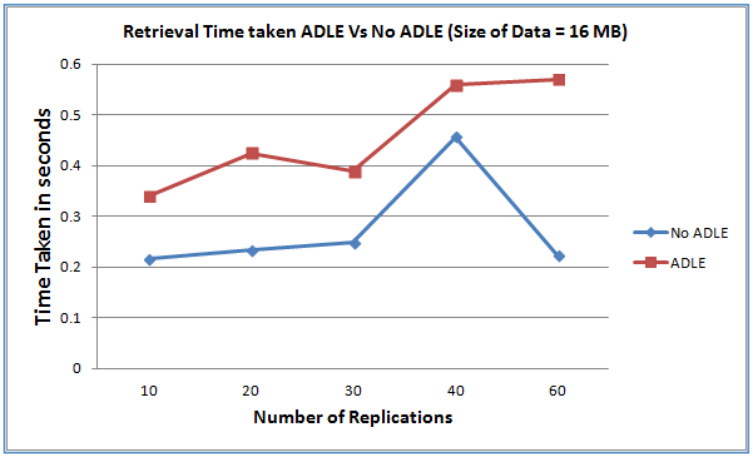

Figure 4.5

In the final Graph 6 (Figure 4.6), we compare the time taken against in case of deletion. We see that Scenario 2 yet again gives phenomenally good results when compared to scenario 1, a major advantage.

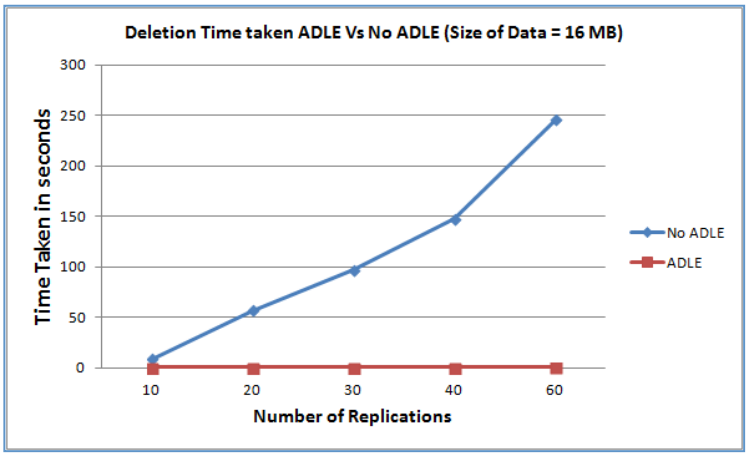

Figure 4.6

4.2.2 Experimental Setup 2

For complete Deletion of the data from the cloud, we analyze whether data has actually been deleted from the cloud or not, in case of an attack or failure on part of the cloud infrastructure.

The tests were run on the system described earlier and graphs were plotted for results. A brief analysis of the results follows ahead.

We compare the performance of the two scenarios when a fault is likely to prevent the cloud from deleting all the replicated data when user instructs the cloud to delete hi/her file.

In Scenario 2, in order to ensure that the data gets deleted, we simple issue one command to delete the data key from the key escrow system. It is fair to assume that a single command in cloud would be atomically executed. If this is the case then, a single command renders the data inaccessible. Hence, 'deleting' data. The problem of reclamation of space is solved using garbage collection, which runs checks in the background to ensure that the space rendered useless is reclaimed back. To implement this we have in our algorithm, executed a Windows Scheduled task that runs at every 't' interval and reclaims space from the database, which has been rendered useless by a deleted key.

However, compared to this we see in the screenshots below that in Scenario 1, the Cloud deletes data one by one from all the replicated locations in some random order. If a fault/attack was to occur and this message doesn't reach a few servers, the data on those servers remains intact. This is the basic problem of Incomplete Data deletion which has been shown on our implemented project. As we can see in phpmyadmin, one cloud server shows the file <name> whereas the other doesn't. This problem has been solved in Scenario 2 as show earlier.

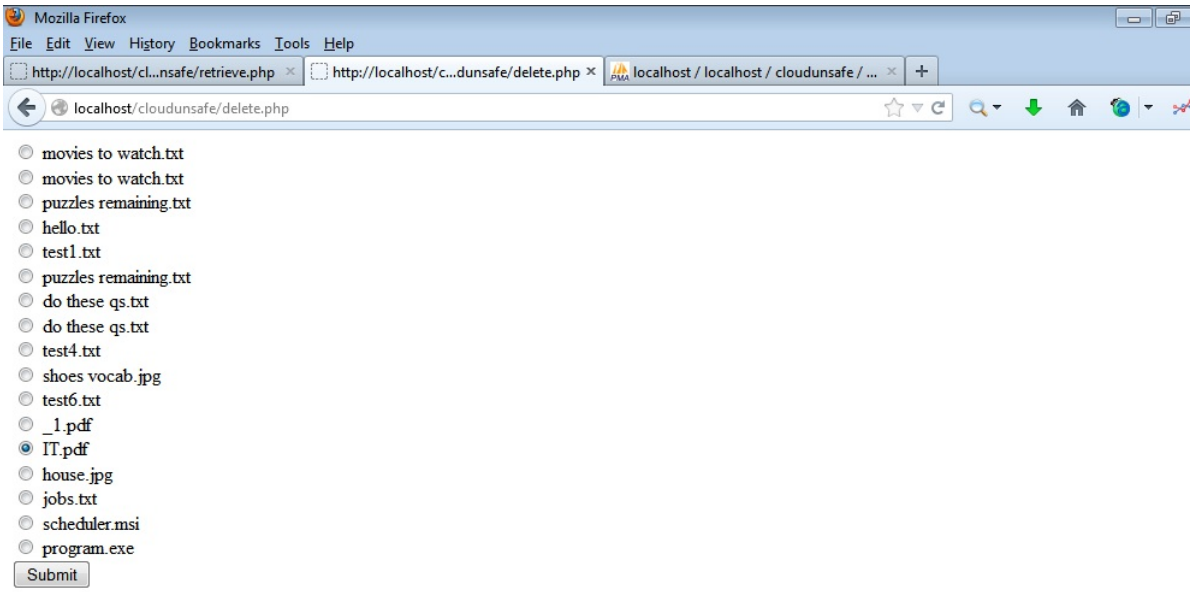

Figure 4.7: Selection of File 'IT' for Deletion

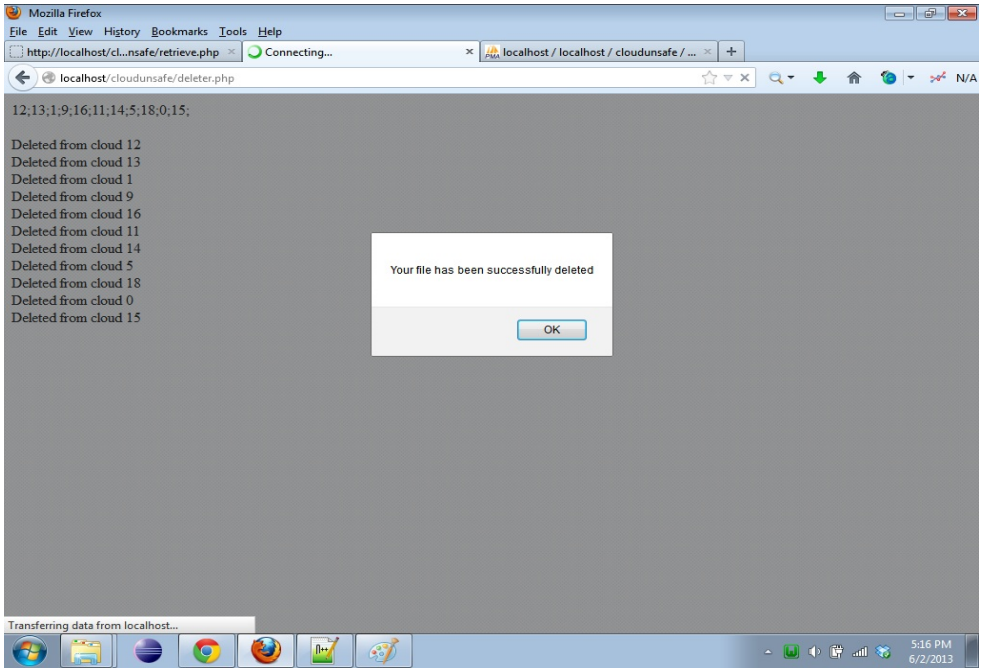

Figure 4.8: Unsuccessful Deletion of File 'IT'

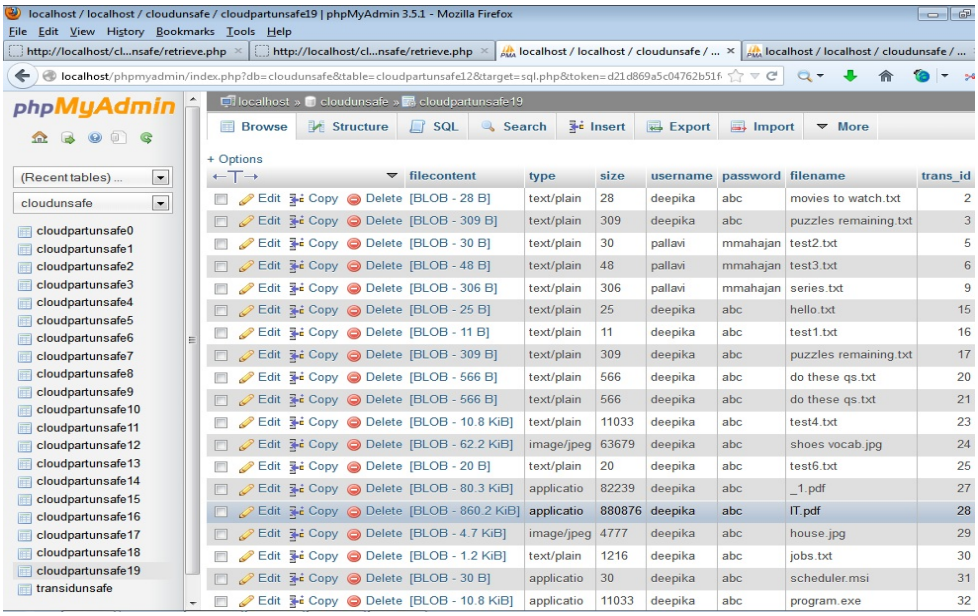

Figure 4.9: Backend Showing the File IT.pdf still present in the database even though it's been deleted.

4.2.3 Experimental Setup 3

In the following setup we use different storage systems – MySQL and Windows Filesystem. The time taken for upload, retrieval and deletion are compared in two scenarios – when no of replications is fixed and when the data size is fixed.

Time taken Vs Size of Data. As we can see in graph 7, the time taken for upload when Filesystem is used as storage is very less than the time taken for upload when MySQL is used. The performance is relatively alright till about 10 MB but gets worse after that. The same occurs in the case of retrieval time and deletion time as can be seen in graph 8 and graph 9 respectively.

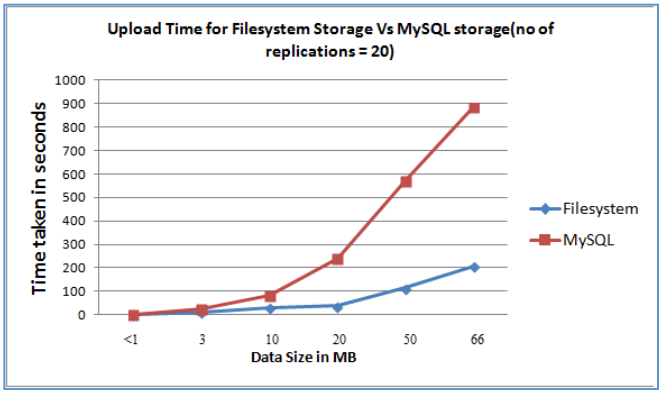

Figure 4.10

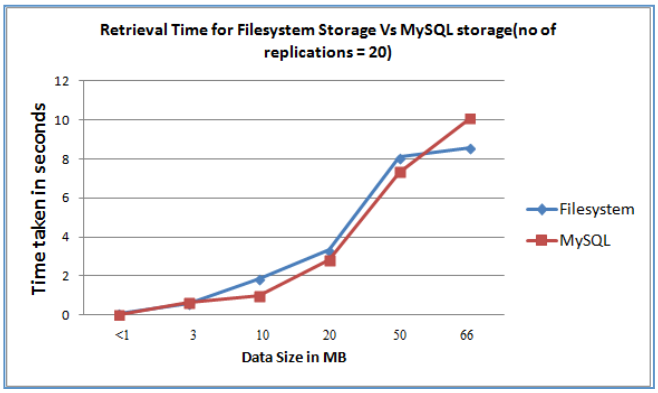

Figure 4.11

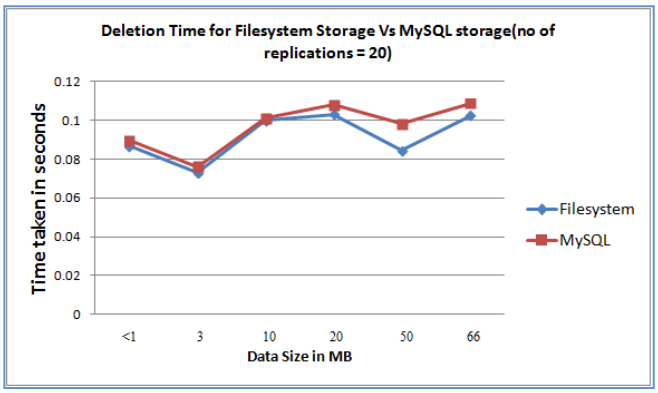

Figure 4.12

Time Taken Vs Number of Replications. In this setup we compare the time taken for the two storage system data access when the size of data is fixed and the number of replications is changed. In graph 10 we notice as before that the time taken is less in case of Filesystem Storage. The case remains the same for Retrieval and Deletion time.

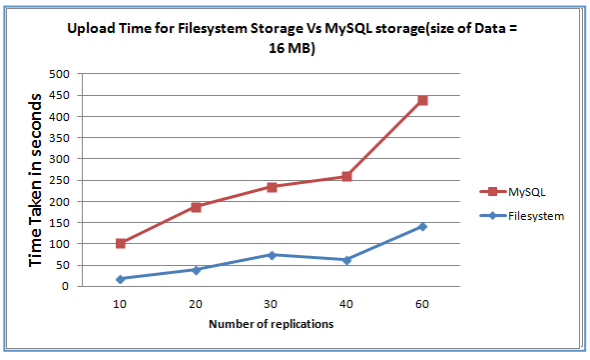

Figure 4.13

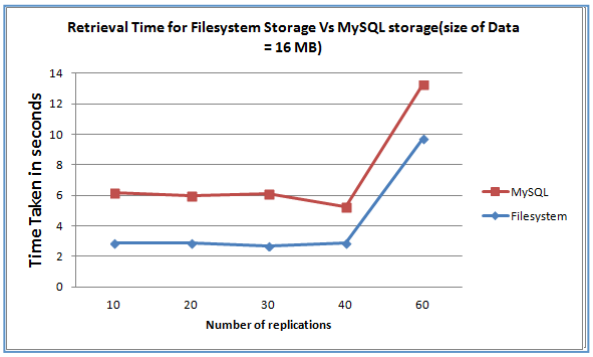

Figure 4.14

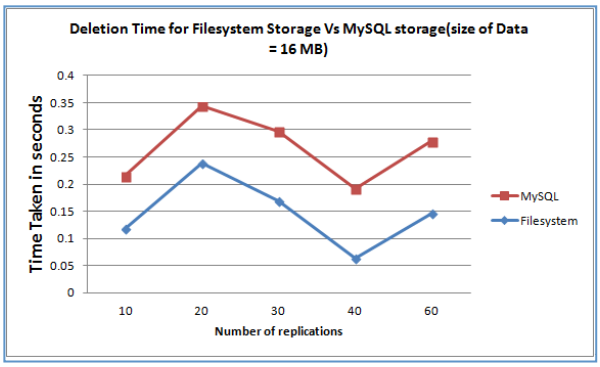

Figure 4.15

The above 6 graphs point to a very important observation – MySQL or SQL databases in general are not a good choice when it comes to storing larger data(>10MB). Filesystems are far more reliable, fast and effective. They do not cause delay in operations as was noticeable during the usage of MySQL as storage.

CHAPTER 5

CONCLUSION AND FUTURE WORK

Cloud computing relies on sharing of resources to achieve coherence. In spite of the numerous features and functionalities provided by the cloud providers, companies/users have trust issues because the common concern is that once the data goes out of the firewall of a trusted organization the data might be subjected to a variety of attacks.

The data may be put to malicious use. This concern is one of the bottlenecks in the popularity/acceptance of cloud as a reliable technology.

In particular, Cloud (backend layer of Cloud Architecture) is subject to numerous attacks like data leakage, cross side scripting etc. One of the major concerns is the problem of incomplete deletion.

In the algorithm proposed in this thesis we have tried to completely handle the issue of incomplete data deletion using layered encryption and key escrow system. Though, as clear from the results, the process of layered encryption increases the size of data to be stored in the database because of padding if compared with traditional storage method used by the cloud besides increasing the time of retrieval because of the fact that double layered decryption is involved, our proposed approach has turned out to be a major improvement over traditional cloud when compared with data deletion process. The deletion time taken by our algorithm remains constant because the only thing which is to nullified is the data key, generated at the time of data encryption.

An additional advantage of our approach is in this case once a file/data has been deleted; the corresponding data can now never be used or put to malicious use. However, in traditional approach the data is always at risk because the data may or may not get deleted from geographically distributed clouds.

Apart from this we have compared the performance of MySQL vs Filesystems as storage backends for cloud. MySQL fails drastically and the time taken for all types of file access is lesser in case of Filesystems.

We have added a garbage collection routine that utilizes space rendered useless by the deleted key using Windows equivalents of Unix Cron jobs in php. Hence, using our proposed approach as presented in the thesis we can successfully handle one the issues of security of cloud.

As part of future work we would like to focus on decreasing the time taken for uploading files via compression as well as encryption.

REFERENCES

[1] The NIST Definition of Cloud Computing | Peter Mell, Timothy Grance

[2] Ian Foster|What is the Grid? A Three Point Checklist, Technical report| Argonne National Laboratory and University of Chicago, July 2002.

[3] Rajkumar Buyya, Chee Shin Yeo, and Srikumar Venugopal| Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities| HPCC, 2008.

[4] Luis M. Vaquero, Luis Rodero-Merino, Juan Caceres, and Maik Lindner| A Break in the Clouds: Towards a Cloud Definition. Technical report| Telefonica Investigacion y Desarrollo and SAP Research Madrid, Spain and Belfast, UK, 2008.

[5] Bogdan Nicolae, PhD Thesis: BlobSeer:Towards efficient data storage management for large-scale, distributed system, Rennes 1, 2010.

[6] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia. A view of cloud computing. Communication of the ACM, April 2010.

[7] Jeffrey Dean and Sanjay Ghemawat. MapReduce: Simplified Data Processing on Large Clusters Communication of the ACM, January 2008.

[8] Craig A. Lee. A Perspective on Scientific Cloud Computing. High Performance Distributed Computing, 2010.

[9] Zach Hill, Jie Li, Ming Mao, Arkaitz Ruiz-Alvarez, and Marty Humphrey | Early Observations on the Performance of Windows Azure. High Performance Distributed Computing, 2010.

[10] Security Guidance for Critical Areas of Focus in Cloud Computing V2.1. Prepared by the Cloud Security Alliance December 2009

[11] B. Nicolae, G. Antoniu, L. Boug, D. Moise, and A. Carpen-Amarie. BlobSeer: Next Generation Data Management for Large Scale Infrastructures| Journal of Parallel and Distributed Computing, 2010.

[12] Mache Creeger, acmqueue, Cloud Computing: An overview

[13] Cloud Computing - Concepts, Architecture and Challenges | Yashpalsinh Jadeja, Kirit Modi | 2012 International Conference on Computing, Electronics and Electrical Technologies [ICCEET]

[14] Security Guidance For Critical Areas Of Focus In Cloud Computing V3.0, Cloud Security Alliance

[15] Top Threats to Cloud Computing V1.0 | Cloud Security Alliance

[16] http://www.malwaredomainlist.com/

[17] http://blogs.zdnet.com/security/?p=5110

[18]http://voices.washingtonpost.com/securityfix/2008/07/amazon_hey_spammers_get_off_my.html

[19] http://www.programmableweb.com

[20] http://securitylabs.websense.com/content/Blogs/3402.aspx

[21] http://blogs.bankinfosecurity.com/posts.php?postID=140

[22] http://technicalinfodotnet.blogspot.com/2010/01/tetheredespionage.html

[23] http://theinvisiblethings.blogspot.com/2008/07/0wning-xen-invegas.html

[24] http://www.blackhat.com/presentations/bh-usa-09/KORTCHINSKY/BHUSA09-Kortchinsky-Cloudburst-PAPER.pdf

[25] http://www.microsoft.com/technet/security/Bulletin/MS10-010.mspx

[26] http://blogs.vmware.com/security/2010/01/announcingvsphere-40-hardening-guide-public-draft-release.html

[27] http://en.wikipedia.org/wiki/Microsoft_data_loss_2009
[28] http://news.cnet.com/8301-13846_3-10029707-62.html
[29] http://nylawblog.typepad.com/suigeneris/2009/11/does-cloudcomputing-compromise-clients.html
[30] http://www.infoworld.com/d/cloud-computing/hackers-findhome-in-amazons-ec2-cloud-742
[31] http://vmetc.com/2009/03/12/virtual-machine-sniffer-on-esxhosts/
[32] http://searchsecurity.techtarget.com/magazineFeature/0,296894,sid14_gci1349670,00.html
[33] http://chenxiwang.wordpress.com/2009/11/24/follow-up-cloudsecurity/
[34] http://www.forrester.com/cloudsecuritywebinar
[35]http://www.cerias.purdue.edu/site/blog/post/symposium_summary_security_in_the_cloud_panel/
[36] http://www.enterpriseirregulars.com/15231/tuesdays-tip-understanding-the-many-flavors-ofcloud-computing-and-saas/
[37] Cloud Computing Security Threats and Responses | Farzad Sabahi| IEEE 2011
[38] Cloud Computing Security | Danish Jamil, Hassan Zaki | International Journal of Engineering Science and Technology (IJEST)
[39] Insider Threats to Cloud Computing: Directions for New Research Challenges| William R Claycomb, Alex Nicoll | CERT Program
[40] Security Guidance for Critical Areas of Forces in Cloud Computing V2.1, CSA 2009
[41] Controlling Data in the Cloud: Outsourcing Computation without Outsourcing Control, Wirayudha Rohandi (241520)| HFU
[42] Analysis of Integrity Vulnerabilities and a Non-repudiation Protocol for Cloud Data Storage Platforms | †Jun Feng*, †Yu Chen, §Wei-Shinn Ku, ‡Pu Liu | SSC 2010
[43] Distributed Cloud Intrusion Detection Model Irfan Gul, M. Hussain | International Journal of Advanced Science and Technology Vol. 34, September, 2011
[44] Survey on Data Security Issues and Data Security Models in Cloud Computing | Ramasami S., Umamaheswari P. | International Journal of Engineering and Innovative Technology (IJEIT), Volume 1, Issue 3, March 2012
[45] Data Loss Prevention Technologies | Tomoyoshi, Hiroshi, Takayuki, Ryusuke | FUJITSU Sci Tech J, Vol 46, January 2010
[46] Cloud Computing, Benefits, risks and recommendations for information security, Enisa, November 2009
[47] Hey, You, Get Off of My Cloud: Exploring Information Leakage in Third-Party Compute Clouds | Thomas Ristenpart¤ Eran Tromer† Hovav Shacham¤ Stefan Savage| ACM 2009
[48] Towards a Data-centric View of Cloud Security | Wenchao Zhou[Micah Sherr\ William R. Marczak] Zhuoyao Zhang, Tao Tao[Boon Thau Loo[Insup Lee, ACM 2010
[49] http://searchsecurity.techtarget.com/tip/Data-sanitization-policy-How-to-ensure-thorough-data-scrubbing
[50] http://pcsupport.about.com/od/toolsofthetrade/tp/free-data-destruction-software.htm
[51] Cloud Computing Security Considerations | Alok Tripathi, Abhinav Mishra| IEEE Explore
[52] Cloud computing: Challenges and future directions | Kim-Kwang Raymond Choo| Australian Institute of Criminology, October 2010
[53] Trusted Cloud Computing with Secure Resources and Data Colouring, Kai Hwang
[54] Mohd Nazri Ismail, Abdulaziz Aborujilah, Shahrulniza Musa & AAmir Shahzad| International Journal of Computer Science and Security (IJCSS), Volume (6) : Issue (4) 229

[55] Noxes: A Client-Side Solution for Mitigating Cross-Site Scripting Attacks by Engin Kirda§, Christopher Kruegel§, Giovanni Vigna‡, and Nenad Jovanovic§

[56] Cross-Site Scripting Prevention with Dynamic Data Tainting and Static Analysis by Philipp Vogt§, Florian Nentwich§, Nenad Jovanovic§,Engin Kirda§, Christopher Kruegel§, and Giovanni Vigna

[57] An Intelligent Approach of Sniffer Detection | Abdul Nasir Khan, Kalim Qureshi, and Sumair Khan| T*he International Arab Journal of Information Technology, Vol. 9, No. 1, January 2012*

[58] Active Detection and Prevention of Sophisticated ARP-Poisoning

Man-in-the-Middle Attacks on Switched Ethernet LANs | K. Kalajdzic and A. Patel | *Proceedings of the Sixth International Workshop on Digital Forensics & Incident Analysis (WDFIA 2011)*

[59] Port Scanning Techniques and the Defense Against Them | SANS Institute

[60] Provable Possession and Replication of Data over Cloud Servers by Ayad F.Barsoum and M.Anwar Hasan

[61]P. Gutmann. Secure deletion of data from magnetic and solid-state memory. In Proc. of USENIX Security Symposium, 1996

[62]Z. N. J. Peterson, R. Burns, J. Herring, A. Stubblefield, and A. D. Rubin. Secure Deletion for a Versioning File System. In Proc. of USENIX FAST, 2005

[63] K. M. Elleithy, D. Blagovic, W. Cheng and P. Sideleau. Denial of Service Attack Techniques: Analysis, Implementation and Comparison.

[64]D. Jamil and H. Zaki. Cloud Computing Security.

[65] Y. Tang, P. Lee, J. Lui, and R. Perlman. FADE: Secure Overlay Cloud Storage with File Assured Deletion. In Proc. of SecureComm, 2010.

[66] M. W. Storer, K. Greenan, D. D. E. Long, and E. L. Miller. Secure Data Deduplication. In Proc. of StorageSS, 2008.

[67] Arthur Rahumed, Henry C. H. Chen, Yang Tang, Patrick P. C. Lee, and John C. S. Lui. A Secure Cloud Backup System with Assured Deletion and Version Control, 2011 International Conference on Parallel Processing Workshops

[68] M. Vrable, S. Savage, and G. Voelker. Cumulus: Filesystem backup to the cloud. In Proc. of USENIX FAST, 2009

[69] Trusted Computing Group| http://www.trustedcomputinggroup.org/.

[70] A. Shamir. How to Share a Secret. CACM, 22(11):612–613, Nov 1979.

[71] P. Chodowiec and K. Gaj. Very compact FPGA implementation of the AES algorithm. In Proc. 5th Int. Workshop on Cryptographic Hardware and Embedded Systems

[72] National Institute of Standards and Technology (NIST). Advanced Encryption Standard (AES), 2001. FIPS-197.

[73]www.ioinformatics.org/locations/ioi01/contest/day2/**double**/**double**.rtf

[74] Sliman Arrag, Abdellatif Hamdoun, Abderrahim Tragha and Salah eddine Khamlich. Several AES Variants under VHDL language In FPGA

APPENDIX A

SCREENSHOTS

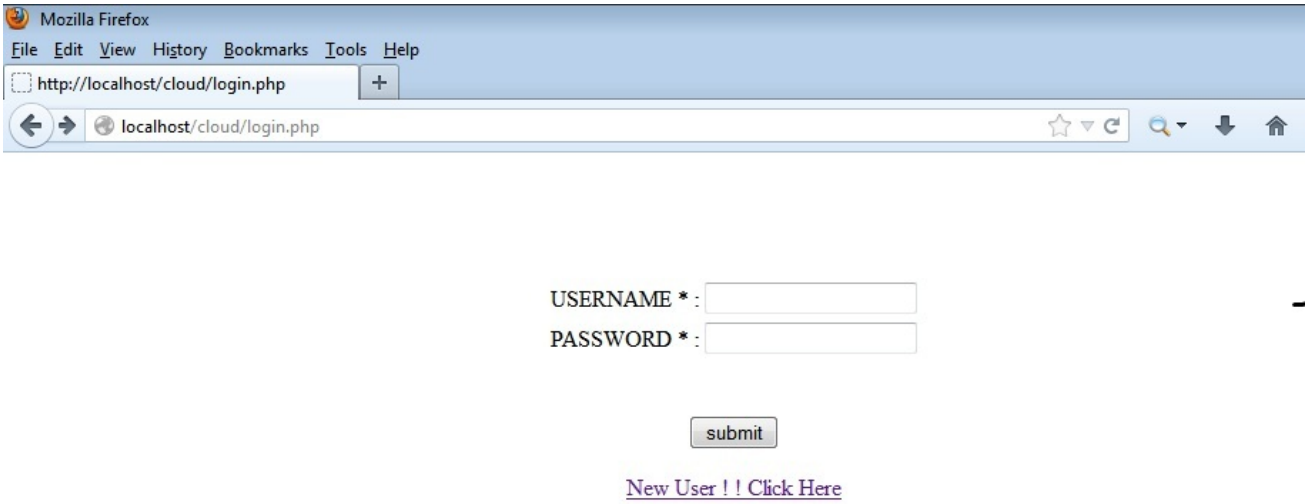

Figure A1.1: User Log in Page

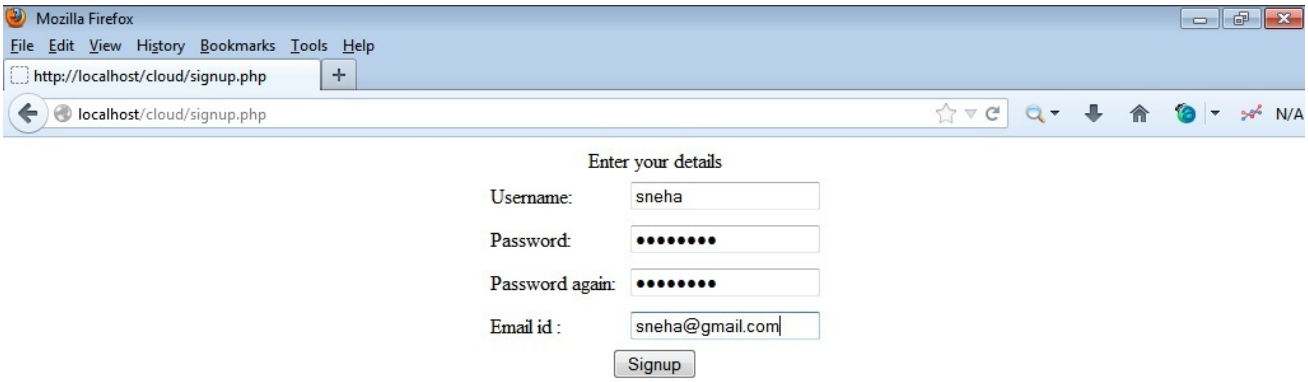

Figure A1.2: User Sign in Page

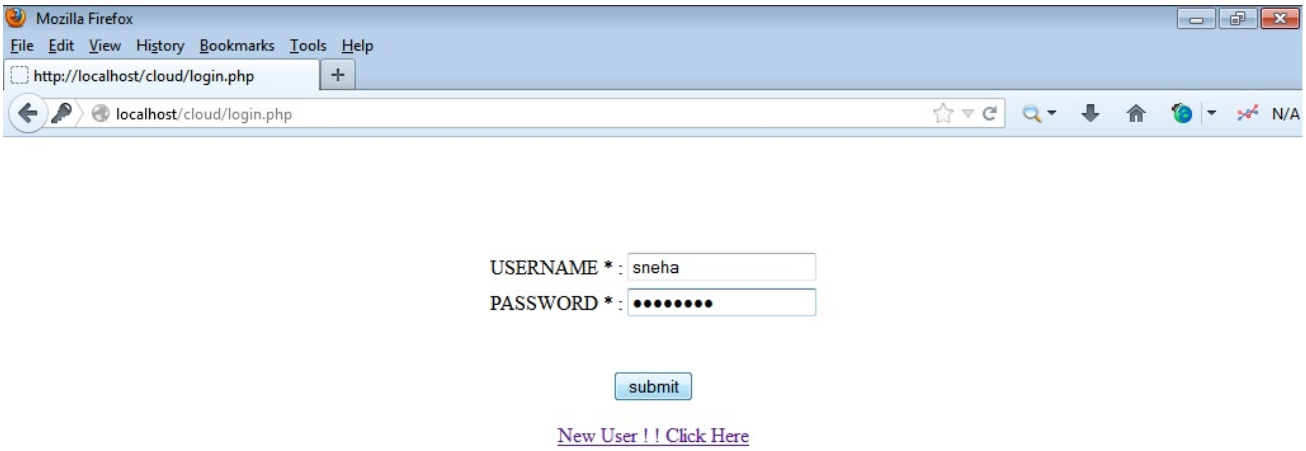

Figure A1.3: User Log In

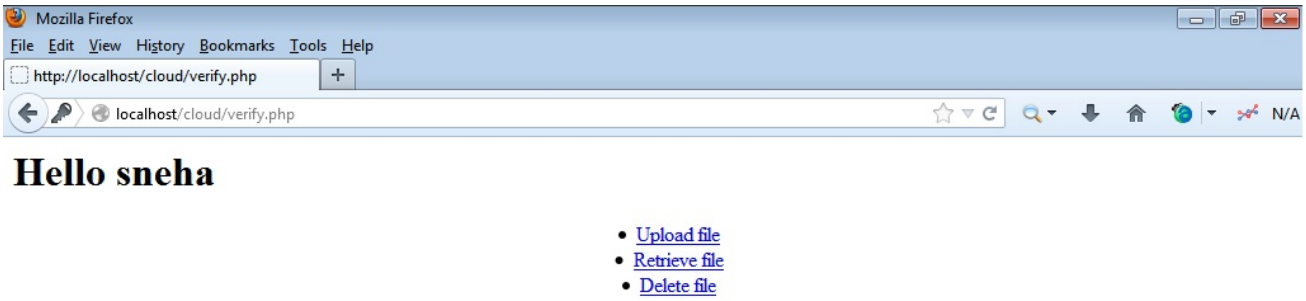

Figure A1.4: Home Page

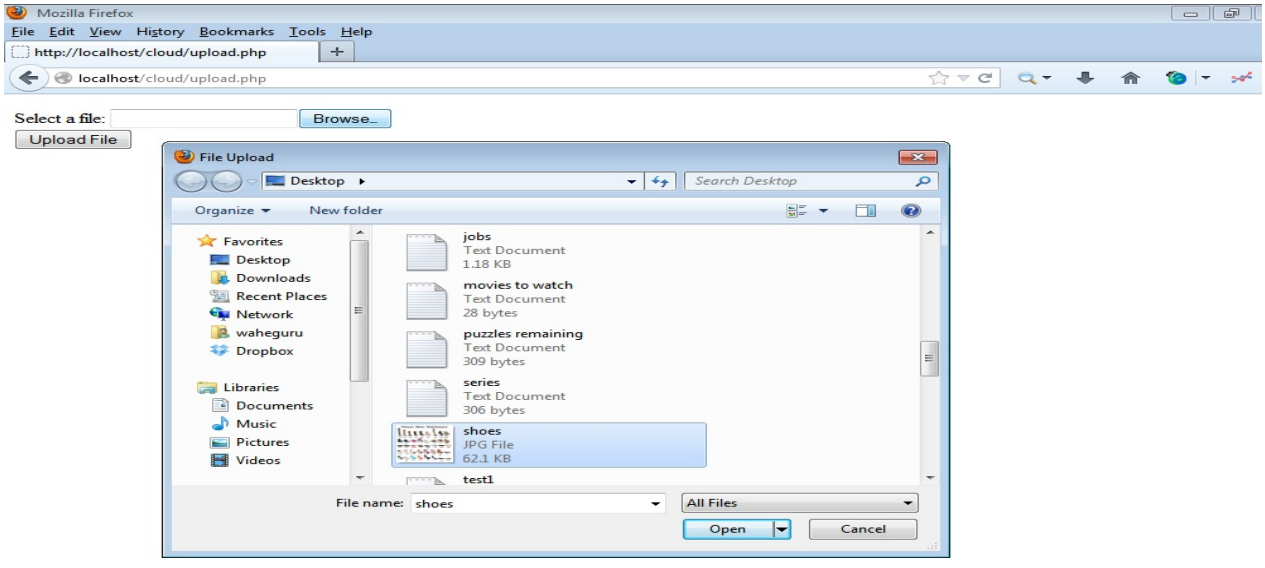

Figure A1.5: Uploading a File - jpeg

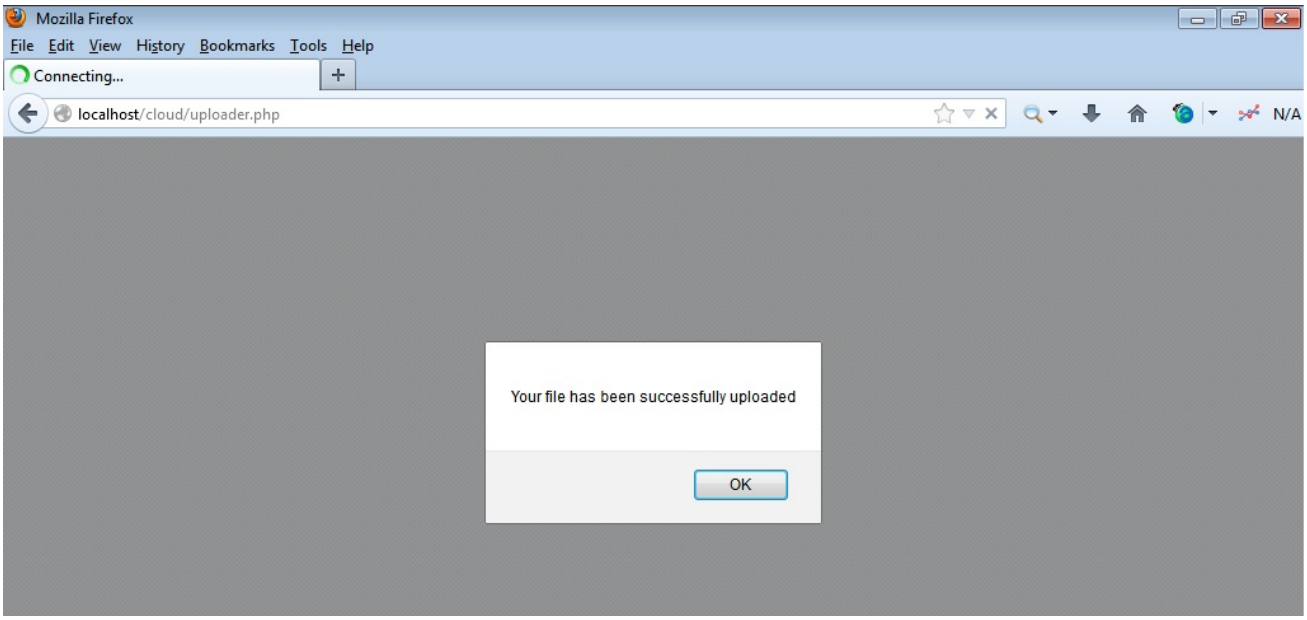

Figure A1.6: Uploading a File - Confirmation

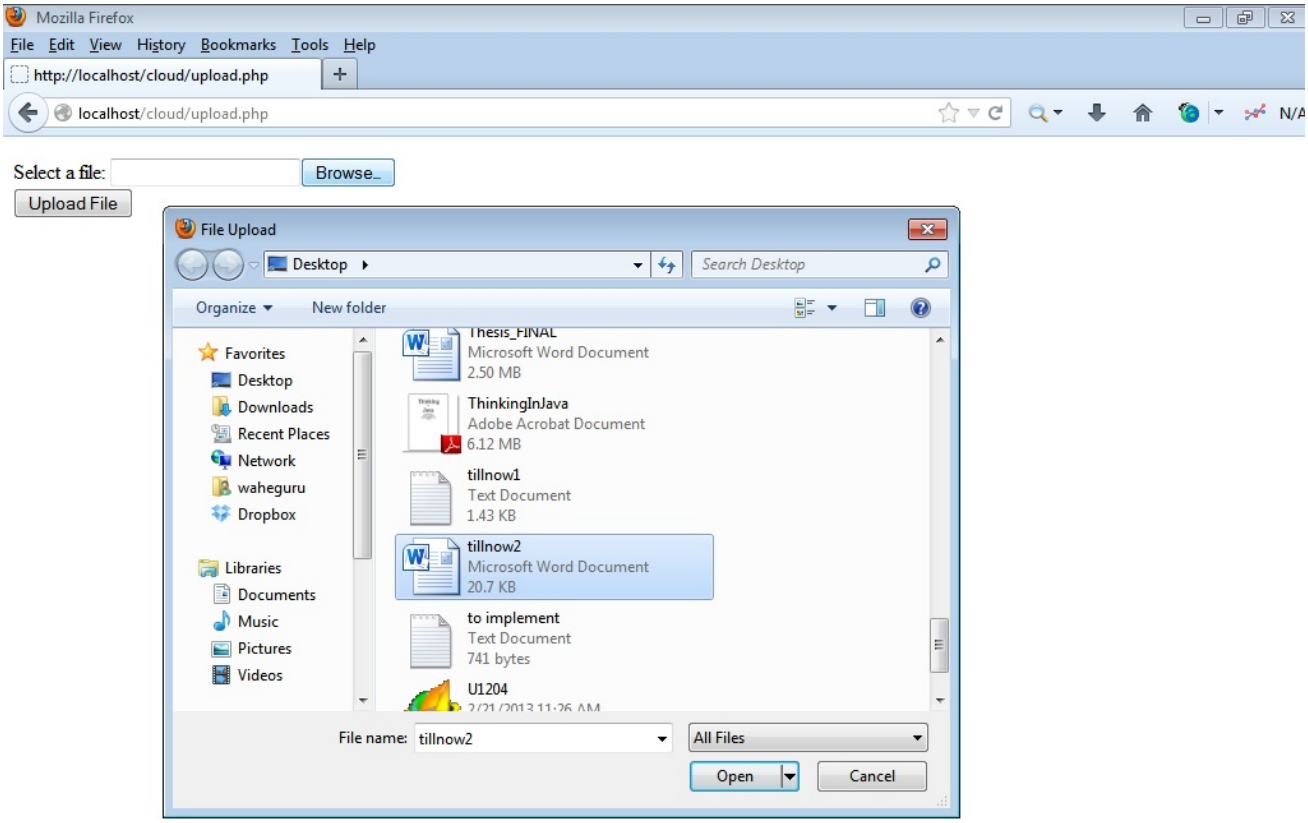

Figure A1.7: Uploading a File - .doc file

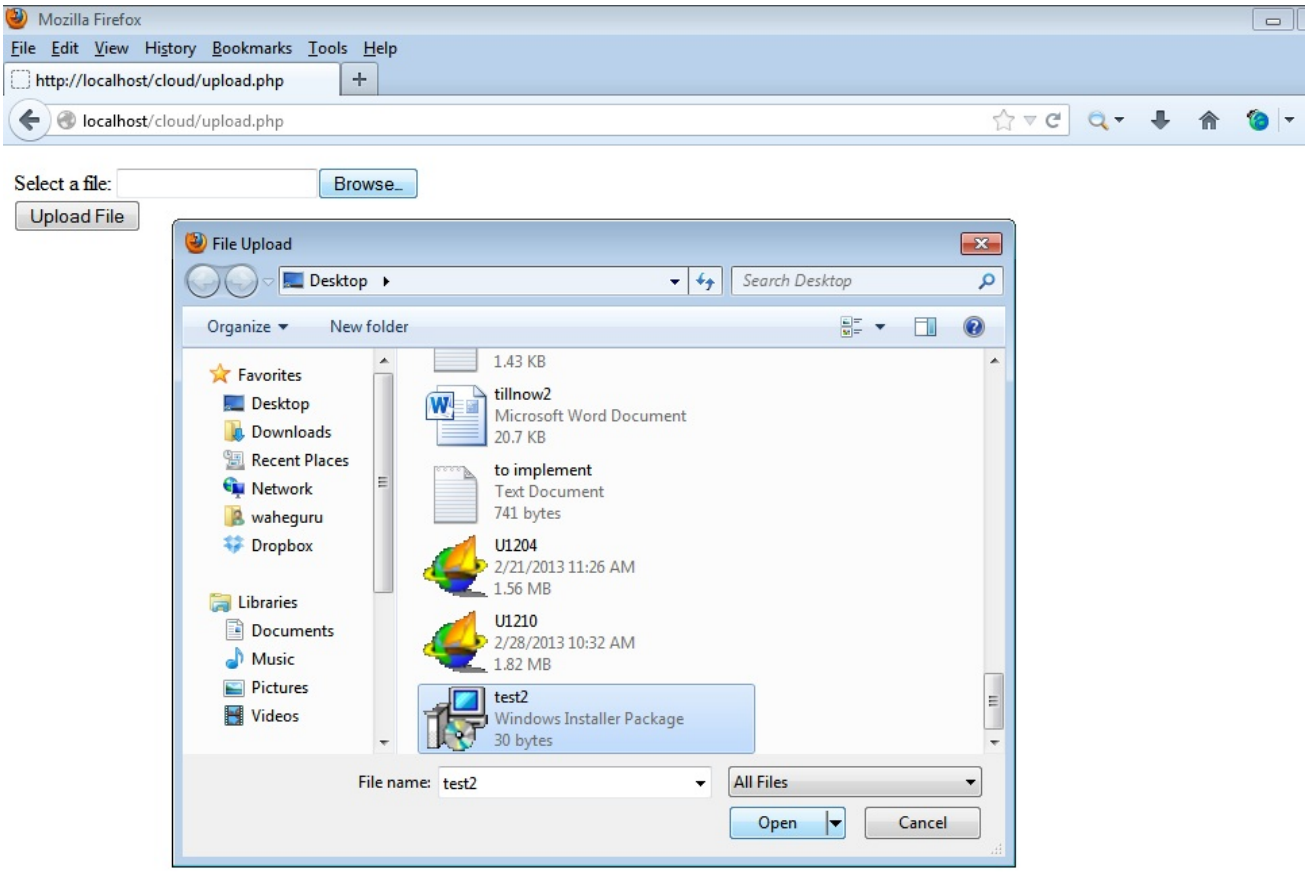

Figure A1.8: Uploading a File - .exe file

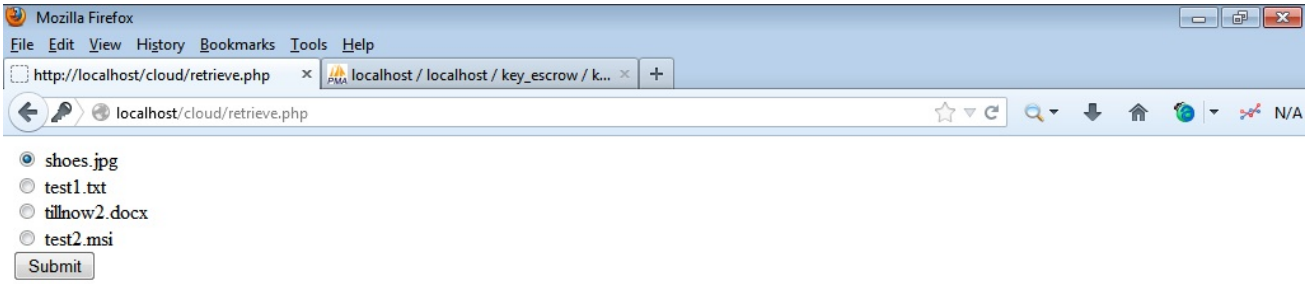

Figure A1.9: Selecting a File for Retrieval

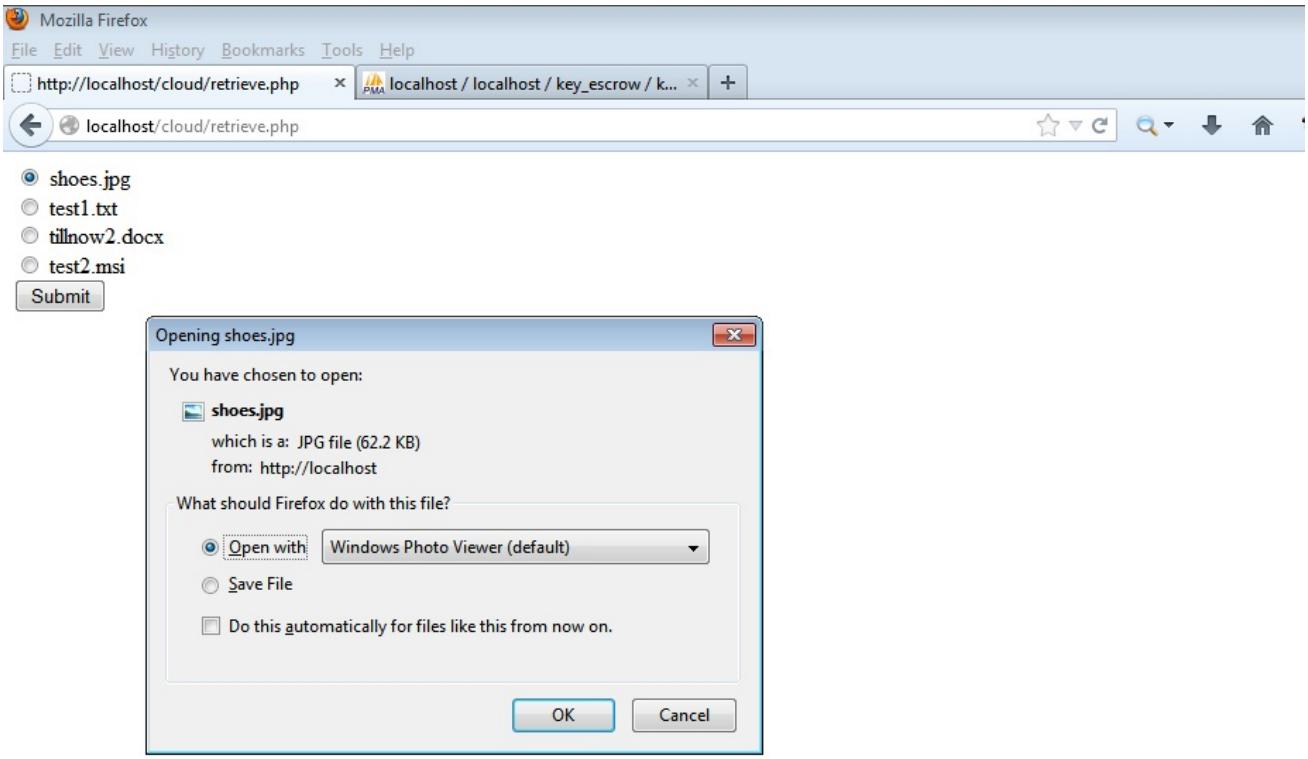

Figure A1.10: Opening the File

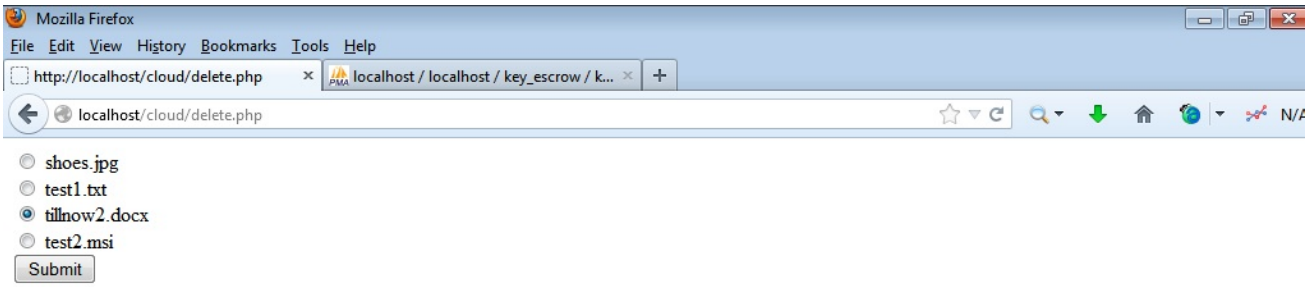

Figure A1.11: Selecting File for Deletion

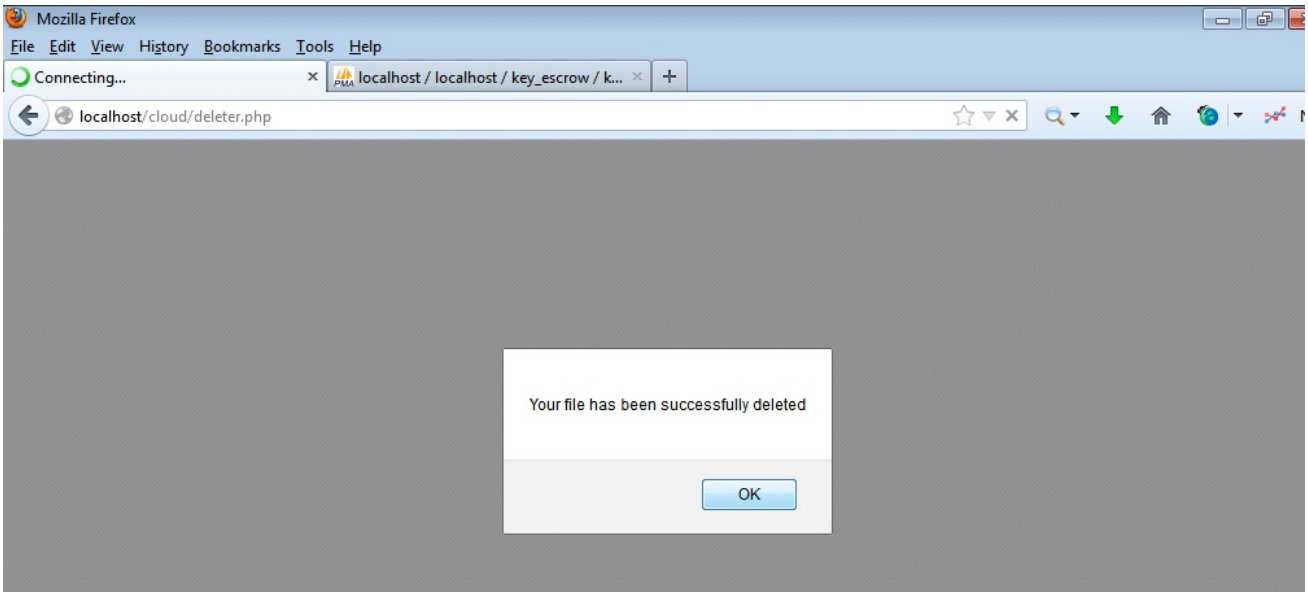

Figure A1.12: Successful Deletion of File

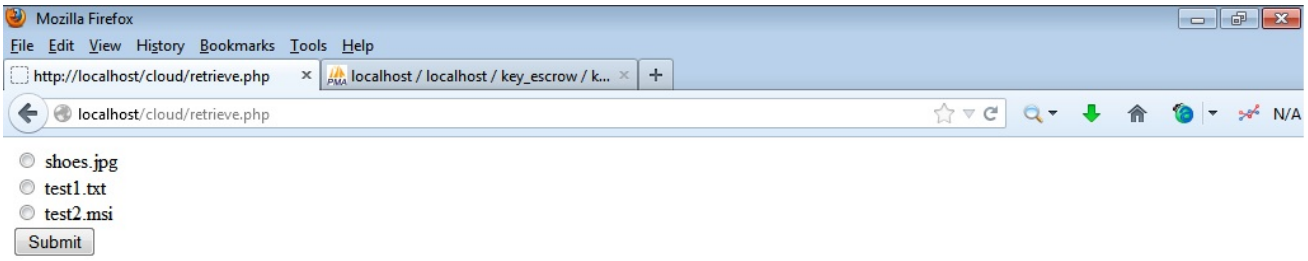

Figure A1.13: Showing the File Page with the Deleted File Missing

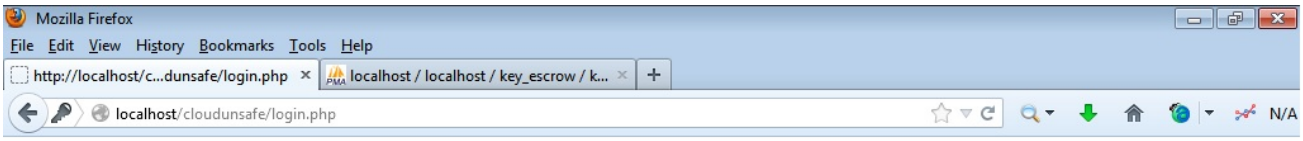

Figure A1.14: Re Log in by User

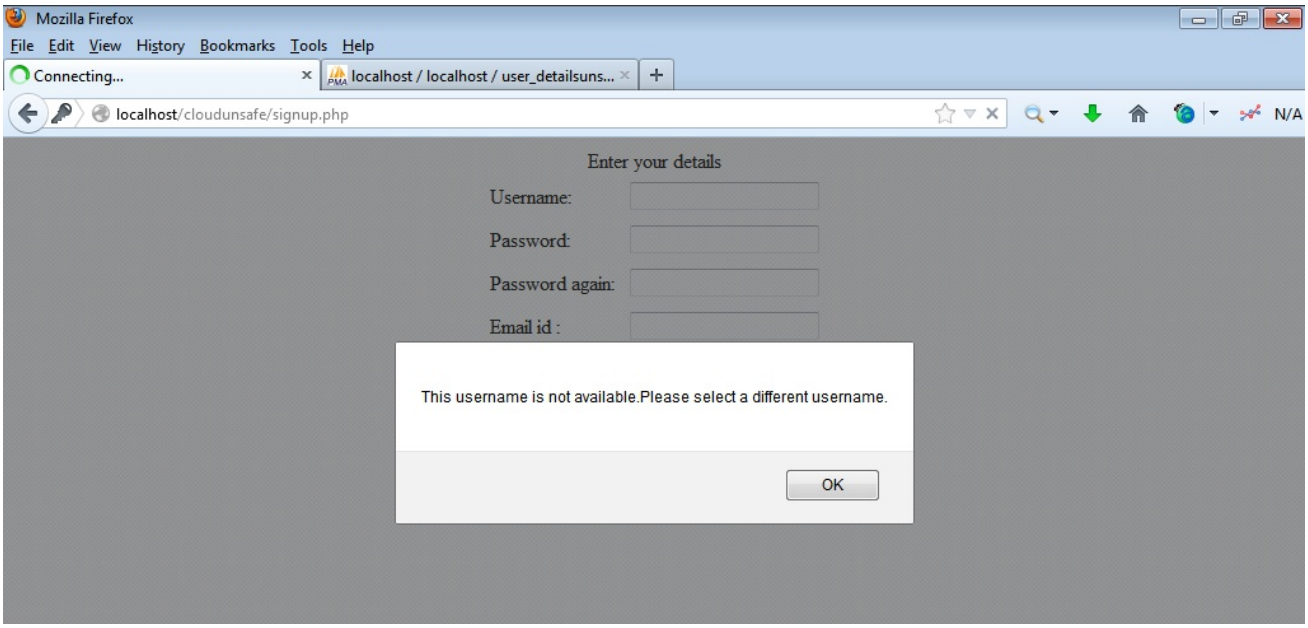

Figure A1.15: Incorrect Log in Details

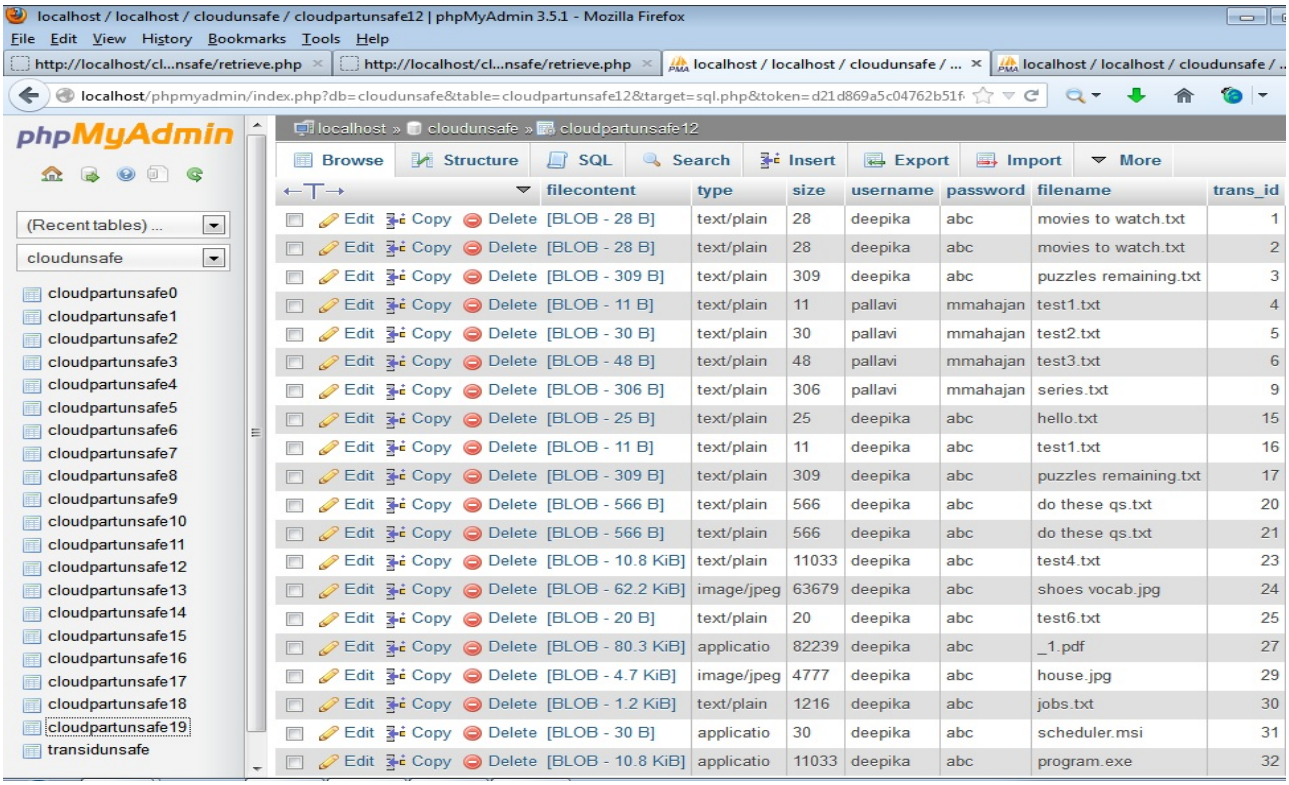

Figure A1.16: Backend Showing User+Data Correspondence