Journal of Network and Computer Applications 34 (2011) 1–11

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Review

A survey on security issues in service delivery models of cloud computing

S. Subashini *, V. Kavitha

Anna University Tirunelveli, Tirunelveli, TN 627007, India

ARTICLE INFO

Article history:
Received 3 March 2010
Received in revised form
11 July 2010
Accepted 11 July 2010

Keywords:
Cloud computing
Data privacy
Data protection
Security
Virtualization

ABSTRACT

Cloud computing is a way to increase the capacity or add capabilities dynamically without investing in new infrastructure, training new personnel, or licensing new software. It extends Information Technology's (IT) existing capabilities. In the last few years, cloud computing has grown from being a promising business concept to one of the fast growing segments of the IT industry. But as more and more information on individuals and companies are placed in the cloud, concerns are beginning to grow about just how safe an environment it is. Despite of all the hype surrounding the cloud, enterprise customers are still reluctant to deploy their business in the cloud. Security is one of the major issues which reduces the growth of cloud computing and complications with data privacy and data protection continue to plague the market. The advent of an advanced model should not negotiate with the required functionalities and capabilities present in the current model. A new model targeting at improving features of an existing model must not risk or threaten other important features of the current model. The architecture of cloud poses such a threat to the security of the existing technologies when deployed in a cloud environment. Cloud service users need to be vigilant in understanding the risks of data breaches in this new environment. In this paper, a survey of the different security risks that pose a threat to the cloud is presented. This paper is a survey more specific to the different security issues that has emanated due to the nature of the service delivery models of a cloud computing system.

Contents

* Corresponding author. Tel.: +91 9840638819.
E-mail addresses: subasundararajan@gmail.com (S. Subashini), kavinayav@gmail.com (V. Kavitha).

1084-8045/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jnca.2010.07.006

1. Introduction

Today Small and Medium Business (SMB) companies are increasingly realizing that simply by tapping into the cloud they can gain fast access to best business applications or drastically boost their infrastructure resources, all at negligible cost. Gartner (Jay Heiser, 2009) defines cloud computing (Stanojevi et al., 2008; Vaquero et al., 2009; Weiss, 2007; Whyman, 2008; Boss et al., 2009) as "a style of computing where massively scalable IT-enabled capabilities are delivered 'as a service' to external customers using Internet technologies". Cloud providers currently enjoy a profound opportunity in the marketplace. The providers must ensure that they get the security aspects right, for they are the ones who will shoulder the responsibility if things go wrong. The cloud offers several benefits like fast deployment, pay-for-use, lower costs, scalability, rapid provisioning, rapid elasticity, ubiquitous network access, greater resiliency, hypervisor protection against network attacks, low-cost disaster recovery and data storage solutions, on-demand security controls, real time detection of system tampering and rapid re-constitution of services. While the cloud offers these advantages, until some of the risks are better understood, many of the major players will be tempted to hold back (Viega, 2009). According to a recent IDCI survey, 74% of IT executives and CIO's cited security as the top challenge preventing their adoption of the cloud services model (Clavister, 2009). Analysts' estimate that within the next five years, the global market for cloud computing will grow to $95 billion and that 12% of the worldwide software market will move to the cloud in that period. To realize this tremendous potential, business must address the privacy questions raised by this new computing model (BNA, 2009). Cloud computing moves the application software and databases to the large data centers, where the management of the data and services are not trustworthy. This unique attribute, however, poses many new security challenges (Cong Wang et al., 2009). These challenges include but not limited to accessibility vulnerabilities, virtualization vulnerabilities, web application vulnerabilities such as SQL (Structured Query Language) injection and cross-site scripting, physical access issues, privacy and control issues arising from third parties having physical control of data, issues related to identity and credential management, issues related to data verification, tampering, integrity, confidentiality, data loss and theft, issues related to authentication of the respondent device or devices and IP spoofing.

Though cloud computing is targeted to provide better utilization of resources using virtualization techniques and to take up much of the work load from the client, it is fraught with security risks (Seccombe et al., 2009). The complexity of security risks in a complete cloud environment is illustrated in Fig. 1.

In Fig. 1, the lower layer represents the different deployment models of the cloud namely private, community, public and hybrid cloud deployment models. The layer just above the deployment layer represents the different delivery models that are utilized within a particular deployment model. These delivery models are the SaaS (Software as a Service), PaaS (Platform as a Service) and IaaS (Infrastructure as a Service) delivery models. These delivery models form the core of the cloud and they exhibit certain characteristics like on-demand self-service, multi-tenancy, ubiquitous network, measured service and rapid elasticity which are shown in the top layer. These fundamental elements of the cloud require security which depends and varies with respect to the deployment model that is used, the way by which it is delivered and the character it exhibits. Some of the fundamental security challenges are data storage security, data transmission

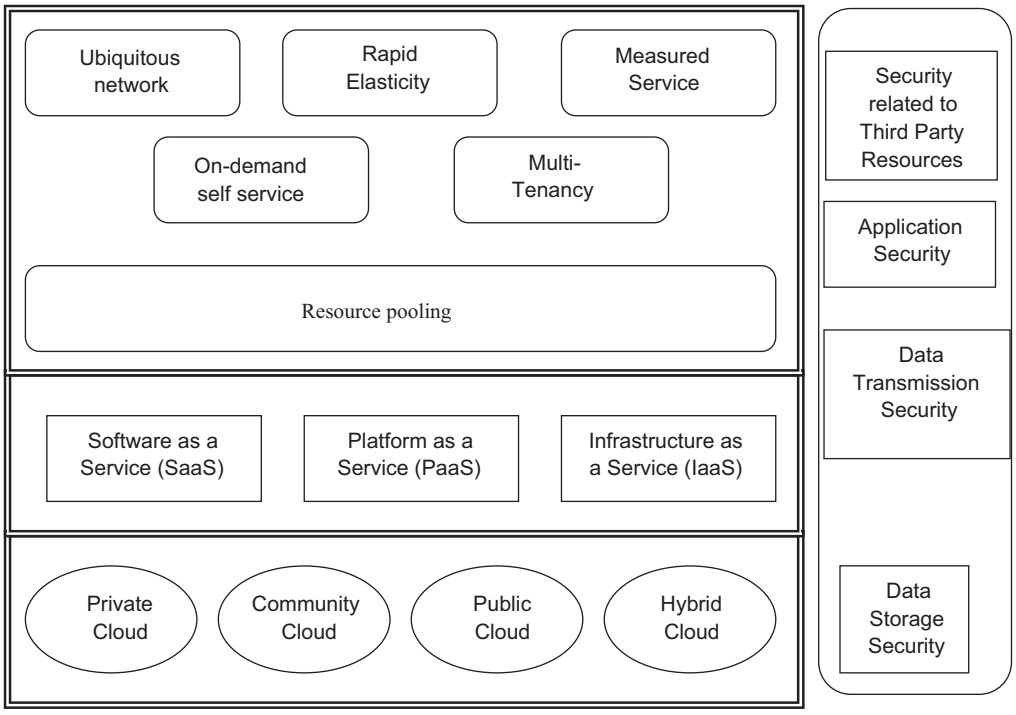

Fig. 1. Complexity of security in cloud environment.

security, application security and security related to third-party resources.

This paper is concentrated towards the issues related to the service delivery models. SaaS is a model of software deployment whereby a provider licenses an application to customers for use as a service on demand. One example of SaaS is the Salesforce.com CRM application. IaaS is the delivery of computer infrastructure (typically a platform virtualization environment) as a service. Rather than purchasing servers, software, data center space or network equipment, clients instead buy those resources as a fully outsourced service. One such example of this is the Amazon web services. PaaS is the delivery of a computing platform and solution stack as a service. It facilitates the deployment of applications without the cost and complexity of buying and managing the underlying hardware and software layers. PaaS provides the facilities required to support the complete lifecycle of building and delivering web applications and services. An example of this would be GoogleApps.

This paper describes the various security issues of cloud computing due to its service delivery models. In the first place, the underlying technology of cloud by itself provides a major security risk. This paper is organized as follows: Section 2 describes the common security issues that are posed by the cloud service delivery models. Section 3 describes the security threats posed by the "Software as a Service" (SaaS) delivery model. Section 4 describes the security threats posed by the "Platform as a Service" (PaaS) delivery model. Section 5 describes the security threats posed by the "Infrastructure as a Service" (IaaS) delivery model. Section 6 lists some of the current solutions which partly target the security challenges posed by the cloud. Section 7 provides conclusions derived out of this survey.

2. Security issues in service models

Cloud computing utilizes three delivery models by which different types of services are delivered to the end user. The three delivery models are the SaaS, PaaS and IaaS which provide infrastructure resources, application platform and software as services to the consumer. These service models also place a different level of security requirement in the cloud environment. IaaS is the foundation of all cloud services, with PaaS built upon it and SaaS in turn built upon it. Just as capabilities are inherited, so are the information security issues and risks. There are significant trade-offs to each model in the terms of integrated features, complexity vs. extensibility and security. If the cloud service provider takes care of only the security at the lower part of the security architecture, the consumers become more responsible for implementing and managing the security capabilities.

A recent survey by Cloud Security Alliance (CSA) & IEEE indicates that enterprises across sectors are eager to adopt cloud computing but that security are needed both to accelerate cloud adoption on a wide scale and to respond to regulatory drivers. It also details that cloud computing is shaping the future of IT but the absence of a compliance environment is having dramatic impact on cloud computing's growth. Organizations using cloud computing as a service infrastructure, critically like to examine the security and confidentiality issues for their business critical insensitive applications. Yet, guaranteeing the security of corporate data in the "cloud" is difficult, if not impossible, as they provide different services like SaaS, PaaS, and IaaS. Each service has its own security issues (Kandukuri et al., 2009).

SaaS is a software deployment model where applications are remotely hosted by the application or service provider and made available to customers on demand, over the Internet. The SaaS model offers the customers with significant benefits, such as improved operational efficiency and reduced costs. SaaS is rapidly emerging as the dominant delivery model for meeting the needs of enterprise IT services. However, most enterprises are still uncomfortable with the SaaS model due to lack of visibility about the way their data is stored and secured. According to the Forrester study, "The State of Enterprise Software: 2009," security concerns are the most commonly cited reason why enterprises are not interested in SaaS. Consequently, addressing enterprise security concerns has emerged as the biggest challenge for the adoption of SaaS applications in the cloud (Heidi Lo et al., 2009). However, to overcome the customer concerns about application and data security, vendors must address these issues head-on. There is a strong apprehension about insider breaches, along with vulnerabilities in the applications and systems' availability that could lead to loss of sensitive data and money. Such challenges can dissuade enterprises from adopting SaaS applications within the cloud.

IaaS completely changes the way developers deploy their applications. Instead of spending big with their own data centers or managed hosting companies or colocation services and then hiring operations staff to get it going, they can just go to Amazon Web Services or one of the other IaaS providers, get a virtual server running in minutes and pay only for the resources they use. With cloud brokers like Rightscale, enStratus, etc., they could easily grow big without worrying about things like scaling and additional security. In short, IaaS and other associated services have enabled startups and other businesses focus on their core competencies without worrying much about the provisioning and management of infrastructure. IaaS completely abstracted the hardware beneath it and allowed users to consume infrastructure as a service without bothering anything about the underlying complexities. The cloud has a compelling value proposition in terms of cost, but "out of the box" IaaS only provides basic security (perimeter firewall, load balancing, etc.) and applications moving into the cloud will need higher levels of security provided at the host.

PaaS is one layer above IaaS on the stack and abstracts away everything up to OS, middleware, etc. This offers an integrated set of developer environment that a developer can tap to build their applications without having any clue about what is going on underneath the service. It offers developers a service that provides a complete software development lifecycle management, from planning to design to building applications to deployment to testing to maintenance. Everything else is abstracted away from the "view" of the developers. The dark side of PaaS is that, these advantages itself can be helpful for a hacker to leverage the PaaS cloud infrastructure for malware command and control and go behind IaaS applications.

3. Security issues in SaaS

In SaaS, the client has to depend on the provider for proper security measures. The provider must do the work to keep multiple users' from seeing each other's data. So it becomes difficult to the user to ensure that right security measures are in place and also difficult to get assurance that the application will be available when needed (Choudhary, 2007). With SaaS, the cloud customer will by definition be substituting new software applications for old ones. Therefore, the focus is not upon portability of applications, but on preserving or enhancing the security functionality provided by the legacy application and achieving a successful data migration (Seccombe et al., 2009).

The SaaS software vendor may host the application on its own private server farm or deploy it on a cloud computing infrastructure service provided by a third-party provider (e.g. Amazon,

Google, etc.). The use of cloud computing coupled with the pay-as-you-go (grow) approach helps the application service provider reduce the investment in infrastructure services and enables it to concentrate on providing better services to customers.

Over the past decade, computers have become widespread within enterprises, while IT services and computing has become a commodity. Enterprises today view data and business processes (transactions, records, pricing information, etc.) themselves as strategic and guard them with access control and compliance policies. However, in the SaaS model, enterprise data is stored at the SaaS provider's data center, along with the data of other enterprises. Moreover, if the SaaS provider is leveraging a public cloud computing service, the enterprise data might be stored along with the data of other unrelated SaaS applications. The cloud provider might, additionally, replicate the data at multiple locations across countries for the purposes of maintaining high availability. Most enterprises are familiar with the traditional on-premise model, where the data continues to reside within the enterprise boundary, subject to their policies. Consequently, there is a great deal of discomfort with the lack of control and knowledge of how their data is stored and secured in the SaaS model. There are strong concerns about data breaches, application vulnerabilities and availability that can lead to financial and legal liabilities.

The layered stack for a typical SaaS vendor and critical aspects that must be covered across layers in order to ensure security of the enterprise data is illustrated in Fig. 2.

The following key security elements should be carefully considered as an integral part of the SaaS application development and deployment process:

- Data security
- Network security
- Data locality
- Data integrity
- Data segregation
- Data access
- Authentication and authorization

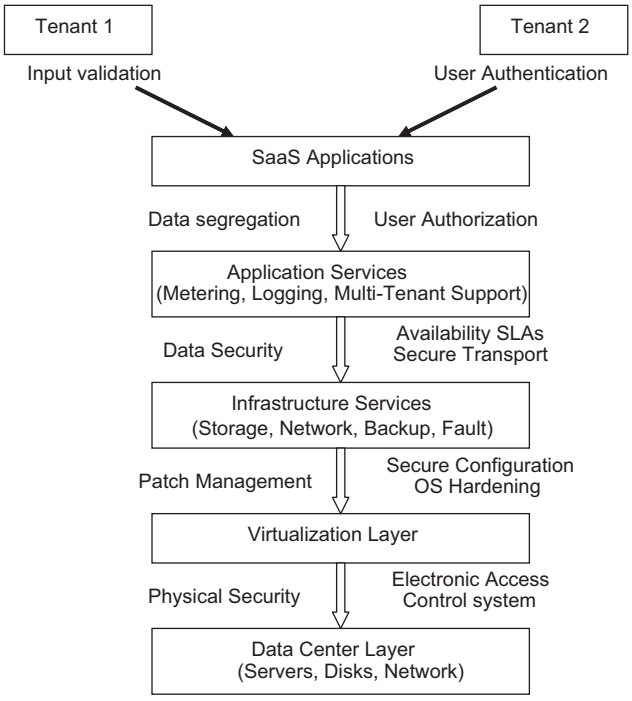

Fig. 2. Security for the SaaS stack.

- Data confidentiality
- Web application security
- Data breaches
- Virtualization vulnerability
- Availability
- Backup
- Identity management and sign-on process.

The different security issues of SaaS are discussed as follows.

3.1. Data security

In a traditional on-premise application deployment model, the sensitive data of each enterprise continues to reside within the enterprise boundary and is subject to its physical, logical and personnel security and access control policies. However, in the SaaS model, the enterprise data is stored outside the enterprise boundary, at the SaaS vendor end. Consequently, the SaaS vendor must adopt additional security checks to ensure data security and prevent breaches due to security vulnerabilities in the application or through malicious employees. This involves the use of strong encryption techniques for data security and fine-grained authorization to control access to data.

In cloud vendors such as Amazon, the Elastic Compute Cloud (EC2) administrators do not have access to customer instances and cannot log into the Guest OS. EC2 Administrators with a business need are required to use their individual cryptographically strong Secure Shell (SSH) keys to gain access to a host. All such accesses are logged and routinely audited. While the data at rest in Simple Storage Service (S3) is not encrypted by default, users can encrypt their data before it is uploaded to Amazon S3, so that it is not accessed or tampered with by any unauthorized party.

Malicious users can exploit weaknesses in the data security model to gain unauthorized access to data. The following assessments test and validate the security of the enterprise data stored at the SaaS vendor:

- Cross-site scripting [XSS]
- Access control weaknesses
- OS and SQL injection flaws
- Cross-site request forgery [CSRF]
- Cookie manipulation
- Hidden field manipulation
- Insecure storage
- Insecure configuration.

Any vulnerability detected during these tests can be exploited to gain access to sensitive enterprise data and lead to a financial loss.

3.2. Network security

In a SaaS deployment model, sensitive data is obtained from the enterprises, processed by the SaaS application and stored at the SaaS vendor end. All data flow over the network needs to be secured in order to prevent leakage of sensitive information. This involves the use of strong network traffic encryption techniques such as Secure Socket Layer (SSL) and the Transport Layer Security (TLS) for security.

In case of Amazon WebServices (AWS), the network layer provides significant protection against traditional network security issues, such as MITM (Man-In-The-Middle) attacks, IP spoofing, port scanning, packet sniffing, etc. For maximum security, Amazon S3 is accessible via SSL encrypted endpoints. The

encrypted endpoints are accessible from both the Internet and from within Amazon EC2, ensuring that data is transferred securely both within AWS and to and from sources outside of AWS.

However, malicious users can exploit weaknesses in network security configuration to sniff network packets. The following assessments test and validate the network security of the SaaS vendor:

- Network penetration and packet analysis
- Session management weaknesses
- Insecure SSL trust configuration.

Any vulnerability detected during these tests can be exploited to hijack active sessions, gain access to user credentials and sensitive data.

3.3. Data locality

In a SaaS model of a cloud environment, the consumers use the applications provided by the SaaS and process their business data. But in this scenario, the customer does not know where the data is getting stored. In many a cases, this can be an issue. Due to compliance and data privacy laws in various countries, locality of data is of utmost importance in many enterprise architecture (Softlayer, 2009). For example, in many EU and South America countries, certain types of data cannot leave the country because of potentially sensitive information. In addition to the issue of local laws, there's also the question of whose jurisdiction the data falls under, when an investigation occurs. A secure SaaS model must be capable of providing reliability to the customer on the location of the data of the consumer.

3.4. Data integrity

Data integrity is one of the most critical elements in any system. Data integrity is easily achieved in a standalone system with a single database. Data integrity in such a system is maintained via database constraints and transactions. Transactions should follow ACID (atomicity, consistency, isolation and durability) properties to ensure data integrity. Most databases support ACID transactions and can preserve data integrity.

Next in the complexity chain are distributed systems. In a distributed system, there are multiple databases and multiple applications. In order to maintain data integrity in a distributed system, transactions across multiple data sources need to be handled correctly in a fail safe manner. This can be done using a central global transaction manger. Each application in the distributed system should be able to participate in the global transaction via a resource manager. This can be achieved using a 2-phase commit protocol as per XA standard.

Enter the world of SOA and Cloud computing, and the problem of the data integrity gets magnified even more, as there is a mix of on-premise and SaaS applications exposed as service. SaaS applications are multi-tenant applications hosted by a third party. SaaS applications usually expose their functionality via XML based APIs (Application Program Interfaces). Also, in SOA based environments, many on-premise applications expose their functionality via SOAP and REST web services as well. One of the biggest challenges with web services is transaction management. At the protocol level, HTTP (Hyper Text Transfer Protocol) does not support transactions or guaranteed delivery, so the only option is to implement these at the API level. Although there are standards available for managing data integrity with web services such as WS-Transaction and WS-Reliability, these standards are not yet mature and not many vendors have implemented these. Most SaaS vendors expose their web services APIs without any support for transactions. Also, each SaaS application may have different levels of availability and SLA (service-level agreement), which further complicates management of transactions and data integrity across multiple SaaS applications.

The lack of integrity controls at the data level (or, in the case of existing integrity controls, bypassing the application logic to access the database directly) could result in profound problems. Architects and developers need to approach this danger cautiously, making sure they do not compromise databases' integrity in their zeal to move to cloud computing.

3.5. Data segregation

Multi-tenancy is one of the major characteristics of cloud computing. As a result of multi-tenancy multiple users can store their data using the applications provided by SaaS. In such a situation, data of various users will reside at the same location. Intrusion of data of one user by another becomes possible in this environment. This intrusion can be done either by hacking through the loop holes in the application or by injecting client code into the SaaS system. A client can write a masked code and inject into the application. If the application executes this code without verification, then there is a high potential of intrusion into other's data. A SaaS model should therefore ensure a clear boundary for each user's data. The boundary must be ensured not only at the physical level but also at the application level. The service should be intelligent enough to segregate the data from different users.

A malicious user can use application vulnerabilities to handcraft parameters that bypass security checks and access sensitive data of other tenants. The following assessments test and validate the data segregation of the SaaS vendor in a multi-tenant deployment:

- SQL injection flaws
- Data validation
- Insecure storage.

Any vulnerability detected during these tests can be exploited to gain access to sensitive enterprise data of other tenants.

3.6. Data access

Data access issue is mainly related to security policies provided to the users while accessing the data. In a typical scenario, a small business organization can use a cloud provided by some other provider for carrying out its business processes. This organization will have its own security policies based on which each employee can have access to a particular set of data. The security policies may entitle some considerations wherein some of the employees are not given access to certain amount of data. These security policies must be adhered by the cloud to avoid intrusion of data by unauthorized users (Blaze et al., 1999; Kormann and Rubin, 2000; Bowers et al., 2008). The SaaS model must be flexible enough to incorporate the specific policies put forward by the organization. The model must also be able to provide organizational boundary within the cloud because multiple organization will be deploying their business processes within a single cloud environment.

3.7. Authentication and authorization

Most companies, if not all, are storing their employee information in some type of Lightweight Directory Access Protocol (LDAP) servers. In the case of SMB companies, a segment that has the highest SaaS adoption rate, Active Directory (AD) seems to be the most popular tool for managing users (Microsoft White Paper, 2010). With SaaS, the software is hosted outside of the corporate firewall. Many a times user credentials are stored in the SaaS providers' databases and not as part of the corporate IT infrastructure. This means SaaS customers must remember to remove/disable accounts as employees leave the company and create/enable accounts as come onboard. In essence, having multiple SaaS products will increase IT management overhead. For example, SaaS providers can provide delegate the authentication process to the customer's internal LDAP/AD server, so that companies can retain control over the management of users.

3.8. Data confidentiality issue

The definitional borders of cloud computing are much debated today. Cloud computing involves the sharing or storage by users of their own information on remote servers owned or operated by others and accesses through the Internet or other connections. Cloud computing services exist in many variations, including data storage sites, video sites, tax preparation sites, personal health record websites and many more. The entire contents of a user's storage device may be stored with a single cloud provider or with many cloud providers. Whenever an individual, a business, a government agency, or any other entity shares information in the cloud, privacy or confidentiality questions arise. Some of the findings related to the confidentiality issues are:

1. Cloud computing has significant implications for the privacy of personal information as well as for the confidentiality of business and governmental information.
2. A user's privacy and confidentiality risks vary significantly with the terms of service and privacy policy established by the cloud provider.
3. For some types of information and some categories of cloud computing users, privacy and confidentiality rights, obligations, and status may change when a user discloses information to a cloud provider.
4. Disclosure and remote storage may have adverse consequences for the legal status of protections for personal or business information.
5. The location of information in the cloud may have significant effects on the privacy and confidentiality protections of information and on the privacy obligations of those who process or store the information.
6. Information in the cloud may have more than one legal location at the same time with differing legal consequences.
7. Laws could oblige a cloud provider to examine user records for evidence of criminal activity and other matters.
8. Legal uncertainties make it difficult to assess the status of information in the cloud as well as the privacy and confidentiality protections available to users.

In an electronic environment, the Electronic Communications Privacy Act of 1986 (ECPA) provides some protections against government access to electronic mail and other computer records held by third parties. The privacy protections available under ECPA for the wide range of cloud computing activities are difficult to predict. Indeed, simply identifying all cloud computing applications would be a significant challenge by itself. Factors that may affect the proper applications of ECPA to cloud computing activities include

1. The precise characterization of the activity as a communication or as a storage, complicated by the recognition that an activity can move from being a communication to being store communication depending on time and possibly other factors.
2. Whether the information in question is content or non-content (e.g., header or transaction information).
3. The terms of service established by the cloud provider.
4. Any consent that the user has granted to the provider or others.
5. The identity of the service provider, for example, if the cloud provider is itself a government agency, the provider's obligation would be different from those of a non-governmental cloud provider, and the rights of users would be different.

3.9. Web application security

SaaS is software deployed over the internet and/or is deployed to run behind a firewall in local area network or personal computer. The key characteristics include Network-based access to, and management of, commercially available software and managing activities from central locations rather than at each customer's site, enabling customers to access application remotely via the Web. SaaS application development may use various types of software components and frameworks. These tools can reduce time-to-market and the cost of converting a traditional on-premise software product or building and deploying a new SaaS solution. Examples include components for subscription management, grid computing software, web application frameworks and complete SaaS platform products. One of the "must-have" requirements for a SaaS application is that it has to be used and managed over the web (in a browser) (Michal Zalewski, 2009). The software which is provided as a service resides in the cloud without tying up with the actual users. This allows improvising the software without inconveniencing the user. Security holes in the web applications thus create a vulnerability to the SaaS application. In this scenario, the vulnerability can potentially have detrimental impact on all of the customers using the cloud. The challenge with SaaS security is not any different than with any other web application technology, however one of the problems is that traditional network security solutions such as network firewalls, network intrusion detection and prevention systems (IDS & IPS), do not adequately address the problem. Web applications introduce new security risks that cannot effectively be defended against at the network level, and do require application level defenses.

Verizon Business in their 'Verizon Business 2008 Data Breach Investigation Report' (Wade et al., 2008) reported 59% of the breaches involve hacking with the following breakdown:

- Application/service layer—39%
- OS/platform layer—23%
- Exploit known vulnerability—18%
- Exploit unknown vulnerability—5%
- Use of back door—5%.

Attacks targeting applications, software, and services were by far the most common techniques, representing 39% of all hacking activity leading to data compromise. This follows a trend in recent years of attacks moving up the stack. Far from past, operating system, platform, and server-level attacks accounted for a sizable portion of breaches. Eighteen percent of hacks exploited a specific

known vulnerability while 5% exploited unknown vulnerabilities for which a patch was not available at the time of the attack. Evidence of re-entry via backdoors, which enable prolonged access and control of compromised systems, was found in 15% of hacking-related breaches. The attractiveness of this to criminals desiring large quantities of information is obvious.

SQL injection (Robert Auger, 2009) is one type of attack which makes the web application more vulnerable. If the application is vulnerable to such type of attacks, the entire data behind the application is at risk. The data can be either belonging to the organization from where the attack is launched or it can also be private data of some other organization hosted in the same cloud.

Since the web applications and SaaS are tightly coupled in providing services to the cloud users, most of the security threats of web application are also posed by the SaaS model of the cloud. The Open Web Application Security Project has identified Top 10 security risks faced by web applications. Those threats are:

1. Injection flaws like SQL, OS and LDAP injection
2. Cross-site scripting
3. Broken authentication and session management
4. Insecure direct object references
5. Cross-site request forgery
6. Security misconfiguration
7. Insecure cryptographic storage
8. Failure to restrict URL access
9. Insufficient transport layer protection
10. Unvalidated redirects and forwards.

3.10. Data breaches

Since data from various users and business organizations lie together in a cloud environment, breaching into the cloud environment will potentially attack the data of all the users. Thus the cloud becomes a high value target (Bernard Golden, 2009; Kaufman, 2009). In the Verizon Business breach report blog (Russ Cooper, 2008) it has been stated that external criminals pose the greatest threat (73%), but achieve the least impact (30,000 compromised records), resulting in a Psuedo Risk Score of 67,500. Insiders pose the least threat (18%), and achieve the greatest impact (375,000 compromised records), resulting in a Pseudo Risk Score of 67,500. Partners are middle in both (73.39% and 187,500) resulting in a Pseudo Risk Score of 73,125. Though SaaS advocates claim that SaaS providers can provide better security to customers' data than by conventional means, Insiders still have access to the data but it is just that they are accessing it in a different way. Insiders do not have direct access to databases, but it does not reduce the risk of insider breaches which can be a massive impact on the security. The SaaS providers' employees have access to a lot more information and a single incident could expose information from many customers. SaaS providers must be compliant with PCI DSS (Payment Card Industry—Data Security Standards) (PCI DSS, 2009) in order to host merchants that must comply with PCI DSS.

3.11. Vulnerability in virtualization

Virtualization is one of the main components of a cloud. But this poses major security risks. Ensuring that different instances running on the same physical machine are isolated from each other is a major task of virtualization which is not met completely in today's scenario. The other issue is the control of administrator on host and guest operating systems. Current VMMs (Virtual Machine Monitor) do not offer perfect isolation. Many bugs have been found in all popular VMMs that allow escaping from VM. Virtual machine monitor should be 'root secure', meaning that no privilege within the virtualized guest environment permits interference with the host system.

Some vulnerability has been found in all virtualization software which can be exploited by malicious, local users to bypass certain security restrictions or gain privileges. For example, the vulnerability of Microsoft Virtual PC and Microsoft Virtual Server could allow a guest operating system user to run code on the host or another guest operating system. Vulnerability in Virtual PC and Virtual Server could allow elevation of privilege. Another example would be the vulnerability in Xen caused due to an input validation error in tools/pygrub/src/GrubConf.py. This can be exploited by 'root' users of a guest domain to execute arbitrary commands in domain 0 via specially crafted entries in grub.conf when the guest system is booted. A perfection of properties like isolation, inspection and interposition is yet to be completely achieved in VMMs.

3.12. Availability

The SaaS application needs to ensure that enterprises are provided with service around the clock. This involves making architectural changes at the application and infrastructural levels to add scalability and high availability. A multi-tier architecture needs to be adopted, supported by a load-balanced farm of application instances, running on a variable number of servers. Resiliency to hardware/software failures, as well as to denial of service attacks, needs to be built from the ground up within the application.

At the same time, an appropriate action plan for business continuity (BC) and disaster recovery (DR) needs to be considered for any unplanned emergencies. This is essential to ensure the safety of the enterprise data and minimal downtime for enterprises.

With Amazon for instance, the AWS API endpoints are hosted on the same Internet-scale, world-class infrastructure that supports the Amazon.com retail site. Standard Distributed Denial of Service (DDoS) mitigation techniques such as synchronous cookies and connection limiting are used. To further mitigate the effect of potential DDoS attacks, Amazon maintains internal bandwidth that exceeds its provider-supplied Internet bandwidth.

These assessments test and validate the availability of the SaaS vendor.

- Authentication weaknesses
- Session management weaknesses.

Many applications provide safeguards to automatically lock user accounts after successive incorrect credentials. However, incorrect configuration and implementation of such features can be used by malicious users to mount denial of service attacks.

3.13. Backup

The SaaS vendor needs to ensure that all sensitive enterprise data is regularly backed up to facilitate quick recovery in case of disasters. Also the use of strong encryption schemes to protect the backup data is recommended to prevent accidental leakage of sensitive information.

In the case of cloud vendors such as Amazon, the data at rest in S3 is not encrypted by default. The users need to separately encrypt their data and backups so that it cannot be accessed or tampered with by unauthorized parties.

The following assessments test and validate the security of the data backup and recovery services provided by the SaaS vendor:

- Insecure storage
- Insecure configuration.

Any vulnerability detected during these tests can be exploited to gain access to sensitive enterprise data stored in backups.

3.14. Identity management and sign-on process

Identity management (IdM) or ID management is a broad administrative area that deals with identifying individuals in a system (such as a country, a network or an organization) and controlling the access to the resources in that system by placing restrictions on the established identities. Identity management can involve three perspectives

1. *The pure identity paradigm*: Creation, management and deletion of identities without regard to access or entitlements.
2. *The user access (log-on) paradigm*: For example: a smart card and its associated data used by a customer to log on to a service or services (a traditional view).
3. *The service paradigm*: A system that delivers personalized role-based, online, on-demand, multimedia (content), presence-based services to users and their devices.

The SaaS vendor can support identity management and sign on services using any of the following models.

3.14.1. Independent IdM stack

The SaaS vendor provides the complete stack of identity management and sign on services. All information related to user accounts, passwords, etc. is completely maintained at the SaaS vendor end.

3.14.2. Credential synchronization

The SaaS vendor supports replication of user account information and credentials between enterprise and SaaS application. The user account information creation is done separately by each tenant within the enterprise boundary to comply with its regulatory needs. Relevant portions of user account information are replicated to the SaaS vendor to provide sign on and access control capabilities. The authentication happens at the SaaS vendor end using the replicated credentials.

3.14.3. Federated IdM

The entire user account information including credentials is managed and stored independently by each tenant. The user authentication occurs within the enterprise boundary. The identity of the user as well as certain user attributes are propagated on-demand to the SaaS vendor using federation to allow sign on and access control.

The security challenges for adopting these models and the relative advantages and disadvantages are listed in Table 1.

The following assessments test and validate the security of the identity management and sign-on process of the SaaS vendor:

- Authentication weakness analysis
- Insecure trust configuration.

Any vulnerability detected during these tests can be exploited to take over user accounts and compromise sensitive data.

4. Security issues in PaaS

In PaaS, the provider might give some control to the people to build applications on top of the platform. But any security below the application level such as host and network intrusion prevention will still be in the scope of the provider and the provider has to offer strong assurances that the data remains inaccessible between applications. PaaS is intended to enable developers to build their own applications on top of the platform. As a result it tends to be more extensible than SaaS, at the expense of customer-ready features. This tradeoff extends to security features and capabilities, where the built-in capabilities are less complete, but there is more flexibility to layer on additional security.

Applications sufficiently complex to leverage an Enterprise Service Bus (ESB) need to secure the ESB directly, leveraging a

Table 1
Security challenges in identity management [IdM] and sign-on process.

IdM and SSO model	Advantages	Disadvantages	Security challenges
Independent IdM stack	• Easy to implement • No separate integration with enterprise directory	• The users need to remember separate credentials for each SaaS application	• The IdM stack should be highly configurable to facilitate compliance with enterprise policies; e.g., password strength, etc.
Credential synchronization	• Users do not need to remember multiple passwords	• Requires integration with enterprise directory • Has higher security risk value due to transmissions of user credentials outside enterprise perimeter	• The SaaS vendor needs to ensure security of the credentials during transit and storage and prevent their leakage
Federated IdM	• Users do not need to remember multiple passwords • No separate integration with enterprise directory • Low security risk value as compared to credential synch	• Relatively more complex to implement	• The SaaS vendor and tenants need to ensure that proper trust relationships and validations are established to ensure secure federation of user identities

protocol such as Web Service (WS) Security (Oracle, 2009). The ability to segment ESBs is not available in PaaS environments. Metrics should be in place to assess the effectiveness of the application security programs. Among the direct application, security specific metrics available are vulnerability scores and patch coverage. These metrics can indicate the quality of application coding. Attention should be paid to how malicious actors react to new cloud application architectures that obscure application components from their scrutiny. Hackers are likely to attack visible code, including but not limited to code running in user context. They are likely to attack the infrastructure and perform extensive black box testing. The vulnerabilities of cloud are not only associated with the web applications but also vulnerabilities associated with the machine-to-machine Service-Oriented Architecture (SOA) applications, which are increasingly being deployed in the cloud.

5. Security issues in IaaS

With IaaS the developer has better control over the security as long as there is no security hole in the virtualization manager. Also, though in theory virtual machines might be able to address these issues but in practice there are plenty of security problems (Attanasio, 1973; Gajek et al., 2007). The other factor is the reliability of the data that is stored within the provider's hardware. Due to the growing virtualization of 'everything' in information society, retaining the ultimate control over data to the owner of data regardless of its physical location will become a topic of utmost interest. To achieve maximum trust and security on a cloud resource, several techniques would have to be applied (Descher et al., 2009).

The security responsibilities of both the provider and the consumer greatly differ between cloud service models. Amazon's Elastic Compute Cloud (EC2) (Amazon, 2010) 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 the security controls that relate to the IT system including the OS, applications and data (Seccombe et al., 2009).

5.1. Impact of deployment model

IaaS is prone to various degrees of security issues based on the cloud deployment model through which it is being delivered. Public cloud poses the major risk whereas private cloud seems to have lesser impact. Physical security of infrastructure and disaster management if any damage is incurred to the infrastructure (either naturally or intentionally), is of utmost importance. Infrastructure not only pertains to the hardware where data is processed and stored but also the path where it is getting transmitted. In a typical cloud environment, data will be transmitted from source to destination through umpteen number of third-party infrastructure devices (Ristenpart et al., 2009). There is a high possibility that data can be routed through an intruder's infrastructure. The complexity involved in IaaS due to each of the service deployment models is illustrated in Table 2.

Although cloud architecture is an improvised technology, the underlying technologies remain the same. The cloud is just built over the internet and all the concerns related to security in internet are also posed by the cloud. The basis of the cloud technology makes the consumer and provider reside at different location and virtually access the resources over the Internet. Even if enormous amount of security is put in place in the cloud, still the data is transmitted through the normal underlying Internet technology. So, the security concerns which are threatening the Internet also threaten the cloud. But, in a cloud, the risks are overwhelmingly high. This is because of its vulnerability and the asset value of the resources and their nature of them residing together. Cloud systems still uses normal protocols and security measures that are used in the Internet but the requirements are at a higher extent. Encryption and secure protocols cater to the needs to a certain extent but they are not context oriented. A robust set of policies and protocols are required to help secure transmission of data within the cloud. Concerns regarding intrusion of data by external non users of the cloud through the internet should also be considered. Measures should be set in place to make the cloud environment secure, private and isolated in the Internet to avoid cyber criminals attacking the cloud.

6. Current security solutions

There are several research works happening in the area of cloud security. Several groups and organization are interested in developing security solutions and standards for the cloud. The Cloud Security Alliance (CSA) is gathering solution providers, non-profits and individuals to enter into discussion about the current and future best practices for information assurance in the cloud ("Cloud Security Alliance (CSA)—security best practices for cloud computing," 2009 (Cloud Security Alliance, 2010a, 2010b)). The Cloud Standards web site is collecting and coordinating information about cloud-related standards under development by the groups. The Open Web Application Security Project (OWASP) maintains list of top vulnerabilities to cloud-based or SaaS models which is updated as the threat landscape changes ("OWASP", 2010). The Open Grid Forum publishes documents to containing security and infrastructural specifications and information for grid computing developers and researchers ("Open Grid Forum", 2010).

The best security solution for web applications is to develop a development framework that has tough security architecture. Tsai W, Jin Z, and Bai X, put forth a four-tier framework for web-based development that though seems interesting, only implies a security facet in the process (Tsai et al., 2009). "Towards best practices in designing for the cloud" by Berre, Roman, Landre, Heuval, Skar, Udnaes, Lennon, and Zeid is a road map toward cloud-centric development (Berre et al., 2009), and the X10 language is one way to achieve better use of cloud capabilities of

Table 2
Cloud service deployment model.

	Infrastructure management	Infratructure ownership	Infrastructure location	Access and consumption
Public cloud	Third-party provider	Third-party provider	Off-premise	Untrusted
Private/community cloud	Organization or third-party provider	Organization or third-party provider	On-premise or off-premise	Trusted
Hybrid cloud	Both organization and third-party provider	Both organization and third-party provider	Both on-premise and off-premise	Trusted and untrusted

massive parallel processing and concurrency (Saraswat Vijay, 2010).

Krugel et al. (2002) point out the value of filtering a packet-sniffer output to specific services as an effective way to address security issues shown by anomalous packets directed to specific ports or services (Krugel et al., 2002). An often-ignored solution to accessibility vulnerabilities is to shut down unused services, keep patches updated, and reduce permissions and access rights of applications and users (Krugel et al., 2002).

Raj et al. (2009) suggest resource isolation to ensure security of data during processing, by isolating the processor caches in virtual machines, and isolating those virtual caches from the hypervisor cache (Raj et al., 2009). Hayes points out that there is no way to know if the cloud providers properly deleted a client's purged data, or whether they saved it for some unknown reason (Hayes, 2008).

Basta and Halton (2007) suggest one way to avoid IP spoofing by using encrypted protocols wherever possible. They also suggest avoiding ARP poisoning by requiring root access to change ARP tables; using static, rather than dynamic ARP tables; or at least make sure changes to the ARP tables are logged.

Hayes (2008) points out an interesting wrinkle here, "Allowing a third-party service to take custody of personal documents raises awkward questions about control and ownership: If you move to a competing service provider, can you take a data with you? Could you lose access to documents if you fail to pay a bill?". The issues of privacy and control cannot be solved, but merely assured with tight service-level agreements (SLAs) or by keeping the cloud itself private.

One simple solution, which Milne (2010) states to be a widely used solution for UK businesses is to simply use in-house "private clouds" (Milne, 2010). Nurmi, Wolski, Grzegorczyk, Obertelli, Soman, Youseff, & Zagorodnov show a preview of one of the available home-grown clouds in their (2009) presentation "The Eucalyptus Open-Source Cloud-Computing System" (Nurmi et al., 2009).

7. Conclusion

As described in the paper, though there are extreme advantages in using a cloud-based system, there are yet many practical problems which have to be solved. Cloud computing is a disruptive technology with profound implications not only for Internet services but also for the IT sector as a whole. Still, several outstanding issues exist, particularly related to service-level agreements (SLA), security and privacy, and power efficiency. As described in the paper, currently security has lot of loose ends which scares away a lot of potential users. Until a proper security module is not in place, potential users will not be able to leverage the advantages of this technology. This security module should cater to all the issues arising from all directions of the cloud. Every element in the cloud should be analyzed at the macro and micro level and an integrated solution must be designed and deployed in the cloud to attract and enthrall the potential consumers. Until then, cloud environment will remain cloudy.

An integrated security model targeting different levels of security of data for a typical cloud infrastructure is under research. This model is meant to be more dynamic and localized in nature. My research questions will center on application and data security over the cloud, and I intend to develop a framework by which the security methodology varies dynamically from one transaction/communication to another. One of the pieces of the framework might be focused on providing data security by storing and accessing data based on meta-data information. This would be more like storing related data in different locations based on the meta-data information which would make information invaluable if a malicious intent user recovers it. Keeping this as a core concept I am doing research on a framework which would be practical. Another piece of the framework would be providing 'Security as a Service' to the applications by providing security as a single-tier or a multi-tier based on the application's requirement and addition to it, the tiers are enabled to change dynamically making the security system less predictable. This research is based on the conceptualization of the cloud security based on real world security system where in security depends on the requirement and asset value of an individual or organization. For example, a normal human does not require personal security but a well known personality needs a body guard, an organization needs a set of security persons and a state or country have their mass military to safe guard their assets. The intense of security is directly proportional to the value of the asset it guards. In a cloud where there are heterogeneous systems having a variation in their asset value, a single security system would be too costly for certain applications and if there is less security then the vulnerability factor of some applications like financial and military applications will shoot up. On the other side, if the cloud has a common security methodology in place, it will be a high value asset target for hackers because of the fact that hacking the security system will make the entire cloud vulnerable to attack. In such a scenario, if customized security is provided as a service to applications, it would make sense. Though there are many practical concerns regarding to dynamic security and data storage based on meta-data information my research is much concentrated to derive a framework which targets these concepts and provide a practical solution.

References

Amazon. Amazon Elastic Compute Cloud (EC2), 2010 〈http://www.amazon.com/ec2/〉 [accessed: 10 December 2009].

Attanasio CR. Virtual machines and data security. In: Proceedings of the workshop on virtual computer systems. New York, NY, USA: ACM; 1973. p. 206–9.

Auger R. SQL Injection, 2009 〈http://projects.webappsec.org/SQL-Injection〉 [accessed on: 15 February 2010].

Basta A, Halton W. Computer security and penetration testing. Delmar Cengage Learning 2007.

Bernard Golden. Defining private clouds, 2009 〈http://www.cio.com/article/492695/Defining_Private_Clouds_Part_One〉 [accessed on: 11 January 2010].

Berre AJ, Roman D, Landre E, Heuvel WVD, Skar LA, Udnaes M, et al. Towards best practices in designing for the cloud. In: Proceedings of the 24th ACM SIGPLAN conference companion on object oriented programming systems languages and applications, Orlando, Florida, USA, 2009, p. 697–8.

Blaze M, Feigenbaum J, loannidis J, Keromytis AD. The role of trust management in distributed systems security, secure Internet programming, issues for mobile and distributed objects. Berlin: Springer-Verlag; 1999. p. 185–210.

BNA. Privacy & security law report, 8 PVLR 10, 03/09/2009. Copyright 2009 by The Bureau of National Affairs, Inc. (800-372-1033), 2009 〈http://www.bna.com〉 [accessed on: 2 November 2009].

Boss G, Malladi P, Quan D, Legregni L, Hall H. Cloud computing, 2009, p. 4 〈http://www.ibm.com/developerswork/websphere/zones/ hipods/library.html〉 [accessed on: 18 October 2009].

Bowers KD, Juels A, Oprea A. HAIL: a high-availability and integrity layer for cloud storage, Cryptology ePrint Archive, Report 2008/489, 2008 〈http://eprint.iacr.org/〉 [accessed on: 18 October 2009].

Choudhary V. Software as a service: implications for investment in software development. In: International conference on system sciences, 2007, p. 209.

Clavister. Security in the cloud, Clavister White Paper 〈http://www.it-wire.nu/members/cla69/attachments/CLA_WP_SECURITY_IN_THE_CLOUD.pdf〉 [accessed on: 21 October 2009].

Cloud Security Alliance. Guidance for identity & access management V2.1, 2010a 〈http://www.cloudsecurityalliance.org/guidance/csaguide-dom12-v2.10.pdf〉 [accessed on: 9 May 2010].

Cloud Security Alliance. Security best practices for cloud computing, 2010b 〈http://www.cloudsecurityalliance.org〉 [accessed on: 10 April 2010].

Cooper R. Verizon Business Data Breach security blog, 2008 〈http://securityblog.verizonbusiness.com/2008/06/10/2008-data-breach-investigations-report/〉 [accessed on: 11 February 2010].

Descher M, Masser P, Feilhauer T, Tjoa AM, Huemer D. Retaining data control to the client in infrastructure clouds. In: International conference on availability, reliability and security, ARES '09, 2009, p. 9–16.

Gajek S, Liao L, Schwenk J. Breaking and fixing the inline approach. In: SWS '07, Proceedings of the ACM workshop on secure web services. New York, NY, USA: ACM; 2007. p. 37–43.
Hayes B. Cloud computing. Commun ACM 2008:9–11.
Heiser J. What you need to know about cloud computing security and compliance, Gartner, Research, ID Number: G00168345, 2009.
Kandukuri BR, Paturi VR, Rakshit A. Cloud security issues. In: IEEE international conference on services computing, 2009, p. 517–20.
Kaufman LM. Data security in the world of cloud computing, security and privacy. IEEE 2009;7(4):61–4.
Kormann D, Rubin A. Risks of the passport single signon protocol. Comput Networks 2000;33(1–6):51–8.
Krugel C, Toth T, Kirda E. Service specific anomaly detection for network intrusion detection. In: Proceedings of the 2002 ACM symposium on applied computing, 2002, p. 201–8.
Lo H, Wang R, Garbani J-P, Daley E, Iqbal R, Green C, Forrester report. The State of Enterprise Software: 2009.
Microsoft White Paper. MS Strategy for Lightweight Directory Access Protocol, 2010 〈http://technet.microsoft.com/en-us/library/cc750824.aspx〉 [accessed on: 2 February 2010].
Milne J. Private cloud projects dwarf public initiatives, 2010 〈http://www.cbronline.com/news/private_cloud_projects_dwarf_public_initiatives_281009〉 [accessed: 19 June 2010].
Nurmi D, Wolski R, Grzegorczyk C, Obertelli G, Soman S, Youseff L et al. The Eucalyptus Open-Source Cloud-Computing System. In: Proceedings of the 2009 ninth IEEE/ACM international symposium on cluster computing and the grid, 2009, p. 124–31.
Open Grid Forum, 2010 〈http://www.ogf.org/〉 [accessed on: 20 May 2010].
Oracle. Wiring through an Enterprise Service Bus, 2009 〈http://www.oracle.com/technology/tech/soa/mastering-soa-series/part2.html〉 [accessed on: 19 February 2010].
OWASP, 2010 〈http://owasptop10.googlecode.com/files/OWASP%20Top%20 10%20-%202010.pdf〉 [accessed: 19 June 2010].
PCI DSS. Requirements and Security Assessment Procedures, 2009 〈https://www.pcisecuritystandards.org/security_standards/download.html?id=pci_dss_v1-2.pdf〉 [accessed on: 24 January 2010].
Raj H, Nathuji R, Singh A, England P. Resource management for isolation enhanced cloud services. In: Proceedings of the 2009 ACM workshop on cloud computing security, Chicago, Illinois, USA, 2009, p. 77–84.
Ristenpart T, Tromer E, Shacham H, Savage S. Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, US (2009). Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds. In: Proceedings of the CCS 2009, ACM Press, 2009, p. 270–4.
Saraswat Vijay. Report on the Programming Language X10, x10-lang.org, 2010 〈http://dist.codehaus.org/x10/documentation/languagespec/x10-latest.pdf〉 [accessed on: 17 June 2010].
Seccombe A, Hutton A, Meisel A, Windel A, Mohammed A, Licciardi A, et al. Security guidance for critical areas of focus in cloud computing, v2.1. CloudSecurityAlliance, 2009, 25 p.
Softlayer. Service Level Agreement and Master Service Agreement, 2009 〈http://www.softlayer.com/sla.html〉 [accessed on: 11 December 2009].
Stanojevi R, Shorten R. Fully decentralized emulation of best-effort and processor sharing queues. ACM SIGMETRICS international conference on the measurement and modeling of computer systems. New York: ACM Press; 2008ACM SIGMETRICS international conference on the measurement and modeling of computer systems. New York: ACM Press; 2008. p. 383–94.
Tsai W, Jin Z, Bai X. Internetware computing: issues and perspective. In: Proceedings of the first Asia-Pacific symposium on Internetware. Beijing, China: ACM; 2009. p. 1–10.
Vaquero LM, Rodero-Merino L, Caceres J, Lindner M. A break in the clouds: towards a cloud definition. In: ACM SIGCOMM, editor. Computer communication review 2009. New York: ACM Press; 2009. p. 50–5In: ACM SIGCOMM, editor. Computer communication review 2009. New York: ACM Press; 2009. p. 50–5.
Viega J. Cloud computing and the common man. Computer 2009;42(8):106–8.
Wade HB, David Hylender C, Andrew Valentine J. Verizon Business 2008 data breach investigation report, 2008 〈http://www.verizonbusiness.com/resources/security/databreachreport.pdf〉 [accessed on: 19 February 2010].
Wang C, Wang Q, Ren K. Ensuring data storage security in cloud computing, Cryptology ePrint Archive, Report, 2009 〈http://eprint.iacr.org/〉 [accessed: 18 October 2009].
Weiss A. Computing in the clouds. In: ACM networker, December 2007, 2007, p. 16–25.
Whyman B. Cloud computing. Information Security and Privacy Advisory Board; 2008. 11–3.
Zalewski M. Browser security handbook, 2009 〈http://code.google.com/p/browsersec/〉 [accessed on: 19 February 2010].