Information Sciences 305 (2015) 357–383

Contents lists available at ScienceDirect

Information Sciences

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

Security in cloud computing: Opportunities and challenges

Mazhar Ali [a,c,*], Samee U. Khan [a], Athanasios V. Vasilakos [b]

[a] North Dakota State University, USA
[b] Kuwait University, Kuwait
[c] COMSATS Institute of Information Technology, Abbottabad, Pakistan

ARTICLE INFO

Article history:
Received 5 September 2014
Received in revised form 28 January 2015
Accepted 29 January 2015
Available online 7 February 2015

Keywords:
Cloud computing
Multi-tenancy
Security
Virtualization
Web services

ABSTRACT

The cloud computing exhibits, remarkable potential to provide cost effective, easy to manage, elastic, and powerful resources on the fly, over the Internet. The cloud computing, upsurges the capabilities of the hardware resources by optimal and shared utilization. The above mentioned features encourage the organizations and individual users to shift their applications and services to the cloud. Even the critical infrastructure, for example, power generation and distribution plants are being migrated to the cloud computing paradigm. However, the services provided by third-party cloud service providers entail additional security threats. The migration of user's assets (data, applications, etc.) outside the administrative control in a shared environment where numerous users are collocated escalates the security concerns. This survey details the security issues that arise due to the very nature of cloud computing. Moreover, the survey presents the recent solutions presented in the literature to counter the security issues. Furthermore, a brief view of security vulnerabilities in the mobile cloud computing are also highlighted. In the end, the discussion on the open issues and future research directions is also presented.

1. Introduction

Since its inception, the cloud computing paradigm has gained the widespread popularity in the industry and academia [88]. The economical, scalable, expedient, ubiquitous, and on-demand access to shared resources are some of the characteristics of the cloud that have resulted in shifting the business processes to the cloud [25,2]. The cloud computing attracts the attention of research community due to its potential to provide tremendous benefits to the industry and the community [9,88]. The resources are provided to the users and released based on demands from the pool of shared resources [4]. The on-demand resource provisioning ensures the optimal resource allocation and is also cost effective [78]. The consumers (individuals and business organizations) no longer need to invest heavily in the information technology (IT) infrastructure [4]. Customers use resources provided by the cloud and pay according to the use. On the other hand, cloud providers can re-use resources as soon as they are released by a particular user resulting in improved resource utilization [78]. Ease of use is yet another advantage being offered by the cloud computing because it does not require the customers to possess extraordinary expertise pertaining to the cloud specific technologies [5]. The management of the technology and services has moved from user to the service provider's end [5].

* Corresponding author.
E-mail addresses: mazhar.ali@ndsu.edu (M. Ali), samee.khan@ndsu.edu (S.U. Khan), vasilako@cs.ku.edu.kw (A.V. Vasilakos).

http://dx.doi.org/10.1016/j.ins.2015.01.025
0020-0255/

The cloud computing provides virtualized resources to the customers using various technologies, for example, Web services, virtualization, and multi-tenancy. The cloud services are delivered to the customer through the Internet [25]. The Web applications are used to access and manage cloud resources that makes Web applications an important component of the cloud computing [70]. The customers' processes are executed in virtualized environment that in turn utilize the physical resources [35]. Multiple virtual processes of various users are allocated to same physical machines that are segregated logically. This gives rise to a multi-tenant environment in the cloud. Despite the provided advantages, the cloud computing is not exclusive of risks with security being the key risk [57].

Security is one of the biggest obstacles that hamper the widespread adoption of cloud computing [28]. Several business and research organization are reluctant in completely trusting the cloud computing to shift digital assets to the third-party service providers [57]. The conventional IT infrastructure keeps the digital assets in the administrative domain of the organizations. All of the processing, movement, and management of data/application are performed within the organizational administrative domain. On the other hand, organizations do not enjoy administrative control of cloud services and infrastructure [52]. The security measures taken by the cloud service providers (CSP) are generally transparent to the organizations. The presence of large numbers of users that are not related to the organizations, aggravate the concerns further [57]. The users might be trusted by the CSP but they may not be of trust to each other. The aforementioned reasons keep the customers under uncertainties about their digital assets located at the cloud resulting in reluctance to adopt cloud computing [57].

There are various studies in the literature discussing the security issues of the cloud computing. The authors in [85,101] presented reviews on the security issues of the cloud computing. However, the aforesaid studies are limited to the discussion of security issues only and the security solutions are not discussed. Ref. [71] reviewed the security issues at different levels of cloud computing. The security solutions have also been presented in [71]. However, the future discussion has not been discussed comprehensively and overview of the cloud technology is missing. The authors in [1] presented a comprehensive study of privacy preservation in the cloud with focus only on e-health clouds. Moreover, the study in [1] is limited in scope to the privacy only. Ref. [121] reviewed the security and privacy challenges in the cloud computing and discussed the defense strategies for the existing vulnerabilities. However, the discussion of the security issues in [121] was centered on confidentiality, integrity, availability, accountability, and privacy-preservability with little discussion on the technologies causing the vulnerability origination. The authors in [74] elaborated the security issues in the cloud along with the approaches that can be employed to tackle the vulnerabilities. Nevertheless, the discussion on future research directions is lacking in the survey. Likewise, the work in [39] detailed the security issues in the cloud computing in depth with brief discussion on current and latest security solutions. The work in [18] surveyed the popular security models of cloud computing, such as cube model, multi-tenancy model, and risk assessment model. Moreover, the authors of [18] have discussed the security risks of cloud computing. However, the risks are discussed from the perspective of different stack holders, like customers, government, and service providers. Security issues from the technological and operational point of view were not in the scope of the aforesaid study. Similarly, the strategies to relieve the security issues are discussed in terms of "what" components and processes should be secured and evaluated. "How" the security objectives are achieved in current research is not elaborated. Similarly, the article [104] describes the security issues in cloud computing and associated security solutions. However, the discussion is more focused on the privacy part of cloud security. Moreover, there is no discussion on future research directions. Our survey differs significantly from the aforesaid surveys in terms of its extensiveness, comprehensive discussion on security issues in cloud computing, and emphasizes on latest security solutions presented in the literature. We also provide the tabulated comparisons of the presented techniques. Moreover, we briefly discuss the security issues pertaining to mobile cloud computing and generic strategies that can lead to solutions. The contributions of this survey with respect to the aforesaid surveys are presented in Table 1. The "✓" and "✗" denote whether the domain specified in the column has been discussed in the survey or not.

The remainder of the paper is organized as follows. Section 2 provides the architectural framework of the cloud computing. The security issues in the cloud computing paradigm are detailed in Section 3 whereas the existing solutions in the contemporary literature are presented in Section 4. Section 5 highlights the security concerns in the mobile cloud computing (MCC). Section 6 discusses the techniques and open issues and Section 7 concludes the survey.

Table 1
Contributions of this study with respect to the discussed surveys.

Work	Cloud overview	Security issues	Counter measures	Open issues
[85]	✓	✓	✗	✗
[101]	✓	✓	✗	✗
[71]	✗	✓	✓	✗
[1]	✗	Privacy only	✓	✓
[121]	✓	✓	✓	Privacy only
[74]	✗	✓	✓	✗
[39]	✗	✓	✓	✗
[18]	✗	✓	✓	✗
This survey	✓	✓	✓	✓

2. Cloud computing architectural framework

Cloud computing integrates various computing technologies to provide services to the end users. To understand the security issues pertaining to the cloud computing, it is important to briefly introduce the concepts that contribute to the cloud computing. The National Institute of Standards and Technology's (NIST) definition [69] of cloud computing is widely accepted [28]. The NIST definition considers the cloud computing as a threefold model of service provisioning (Fig. 1), comprising of: (a) essential characteristics, (b) service models, and (c) deployment models. The cloud computing concepts in the light of NIST definition are presented below.

2.1. Essential characteristics

2.1.1. On-demand self-service

Customers can request and manage the services from the cloud without any human interaction with the CSP. The provision of the services and the associated resources is accomplished as and when required. This is usually done through Web services and management interfaces [28].

2.1.2. Broad network access

The services and the customer's applications and data present on the cloud must be accessible to the customers using the standard mechanisms and protocols. The characteristic further demands that the availability of services should support heterogeneous thin or thick environment (for example, mobile phones, laptops, workstations, tablets). Broad network access is sometimes referred to as ubiquitous network access in the literature [28].

2.1.3. Resource pooling

The cloud's resources are shared among multiple customers by pooling in a multi-tenant environment. The customers are transparent about the location of the resources. There is a mapping between physical and virtual resources provided to the customers.

2.1.4. Rapid elasticity

The resources can be rapidly and elastically scaled as per customer's demands. The customer has a view of unlimited resources that can be purchased as needed in a pay-as-you-go manner.

2.1.5. Measured service

The scaling of resources up and down is performed dynamically and the usage of services is metered and reported to the customer and CSP. The metering also helps the optimization of resource usage automatically while the users are charged in a pay-as-you-use manner.

2.1.6. Multi-tenancy

The NIST defines the above mentioned five characteristics of the cloud computing. However, the Cloud Security Alliance (CSA) adds multi-tenancy as an important characteristic of the cloud computing (although not an essential characteristic)

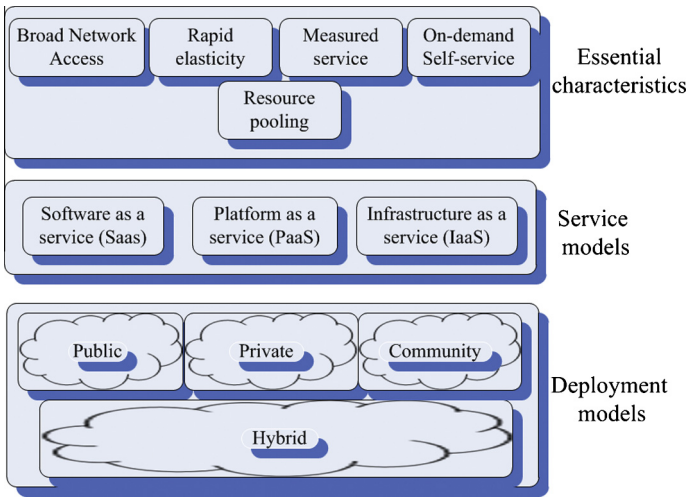

Fig. 1. NIST definition of cloud computing.

[21]. Multi-tenancy is the property that enables the use of a single resource by multiple customers that may or may not belong to the same organization. Multi-tenancy results in optimal use of resources and different customers are segregated logically.

2.2. Service models

The NIST divides the services provided by the cloud computing into three categories, namely: (a) software as a service (SaaS), (b) platform as a service (PaaS), and (c) infrastructure as a service (IaaS). The cloud service model is referred to as SPI (software, platform, and infrastructure).

2.2.1. SaaS

The SaaS enables the customers to use CSP's applications, running on the cloud infrastructure, through the Internet. The thin client interface can be used to access the applications such as web browser. The SaaS does not provide the facility to create an application or software. The SaaS only provides software through Internet making it a model to distribute the software through Web. The customers pay for the usage and do not own the software [96].

2.2.2. PaaS

The applications that are owned by the customer need a frame work where they can be executed and managed. This includes integrated development environments (IDE), operating systems, and platform layer resources (run time engine that executes the applications). The aforementioned services are provided as PaaS. The PaaS does not provide customers with the control over the underlying cloud infrastructure but only on the applications that are moved to the cloud.

2.2.3. IaaS

The IaaS refers to the hardware infrastructure provided by the CSP including the network, storage, memory, processor, and various other computing resources. The resources are provided in the form of virtualized systems accessible through Internet. The CSP has a control over the underlying resources [39].

2.3. Deployment models

There are four models that can be used to deploy a cloud computing infrastructure, namely: (a) private cloud, (b) public cloud, (c) community cloud, and (d) hybrid cloud.

2.3.1. Private cloud

The cloud that is run and managed only for a single organization is the private cloud. The organization may or may not own the physical infrastructure and can be managed by the organization itself or by a third party. Similarly, private cloud may or may not be located at organization's geographical site. However, whatever the case may be, private cloud is for the use of only single organization and the resources are not utilized by any other customer.

2.3.2. Public cloud

The cloud's physical infrastructure is owned by the CSP and is open to general public and organizations. The resources are shared among all the customers. The customers pay the cloud owner according to the services and resources they use. The physical infrastructure is located off-site to the customers and is managed by the CSP.

2.3.3. Community cloud

The community cloud is shared by a number of organizations and/or customers forming a community. Generally, the community shares common interests, such as the mission, security requirements, policy, and compliance considerations. The community cloud may be managed by any of the organizations in the community or a third party. Similarly, it may be located on premise or off-premise.

2.3.4. Hybrid cloud

The hybrid cloud is the mix of two or more clouds (public, private, or community). All of the participating clouds retain their status of a unique entity, but share standardized or proprietary technology.

3. Cloud security challenges

The characteristics and models of the cloud computing presented in previous section offer improved, optimized, and low cost services to the customers. The above given models providing the mentioned characteristics are implemented using various technologies, for example virtualization and multi-tenancy. The technologies along with the cloud service and deployment models introduce cloud specific security risks and vulnerabilities in addition to shared risks with the conventional IT infrastructure [87]. The security risks in cloud may differ from the risks of conventional IT infrastructure either in nature or

intensity or both. Resource pooling allows the use of same pool by multiple users through multi-tenancy and virtualization technologies. Although, the technologies introduce rapid elasticity and optimal management of resources, they also introduce certain risks in the system. Multi-tenancy leads to the risks of data visibility to other users and trace of operations. On-demand self-service characteristic is provided to the customers by means of Web based management interfaces that causes the probability of unauthorized access to the management interface higher than the traditional systems [28]. Similarly, virtualized environment introduces its own set of risks and vulnerabilities that includes malicious cooperation between virtual machines (VM) and VM escape. Likewise, from the cloud service model view point, the service models are dependent on each other. The SaaS applications are built and deployed over the PaaS and the PaaS is dependent on the underlying IaaS. This operational dependency of the service models on each other brings in the security dependency also. For example, if an attacker succeeds to take control of IaaS, the result will be a compromised PaaS that is utilizing IaaS. A compromised PaaS can lead to compromised SaaS. In short, any compromised service model gives access to other layer of the service model. The private cloud deployment model inherits the same set of vulnerabilities as possessed by the conventional IT infrastructure. The reason being the private cloud is meant for the use of a single organization. The public, community, and hybrid clouds possess more cloud specific vulnerabilities and risks due to presence of users from different origins and administrative control of a third party [15]. The presence of multi-tenants using virtualized resources that may correspond to same physical resources introduces many security concerns. The perfect segregation of numerous tenants and allocated resources is a complex task and needs much higher level of security. In the following discussion we present the security challenges being faced by the cloud computing. There are numerous works that look upon the cloud security challenges from service model perspective. In this paper we detail the challenges based on three abstract domains, namely, (a) architectural issues, (b) communication issues, and (c) contractual and legal issues. Some of the technologies in cloud computing do not affect any particular service model. Instead more than one models become affected, such as, virtualization that can affect both the IaaS and PaaS. Therefore, we look at the challenges at abstract level irrespective of the service model. The taxonomy of the security challenges in the cloud computing is depicted in Fig. 2.

3.1. Challenges at communication level

The cloud services are normally available to the customers through the Internet [125]. Standard Internet protocols and mechanisms are used for communication between the customers and the cloud [101]. The communication process results in transmission of either data/information or applications between the customer and the cloud. Moreover, there exists communication within cloud between VMs. We divide the cloud communication into two categories, namely: (a) communication external to the cloud (between customers and cloud) and (b) communication internal to the cloud (communication occurring within cloud infrastructure).

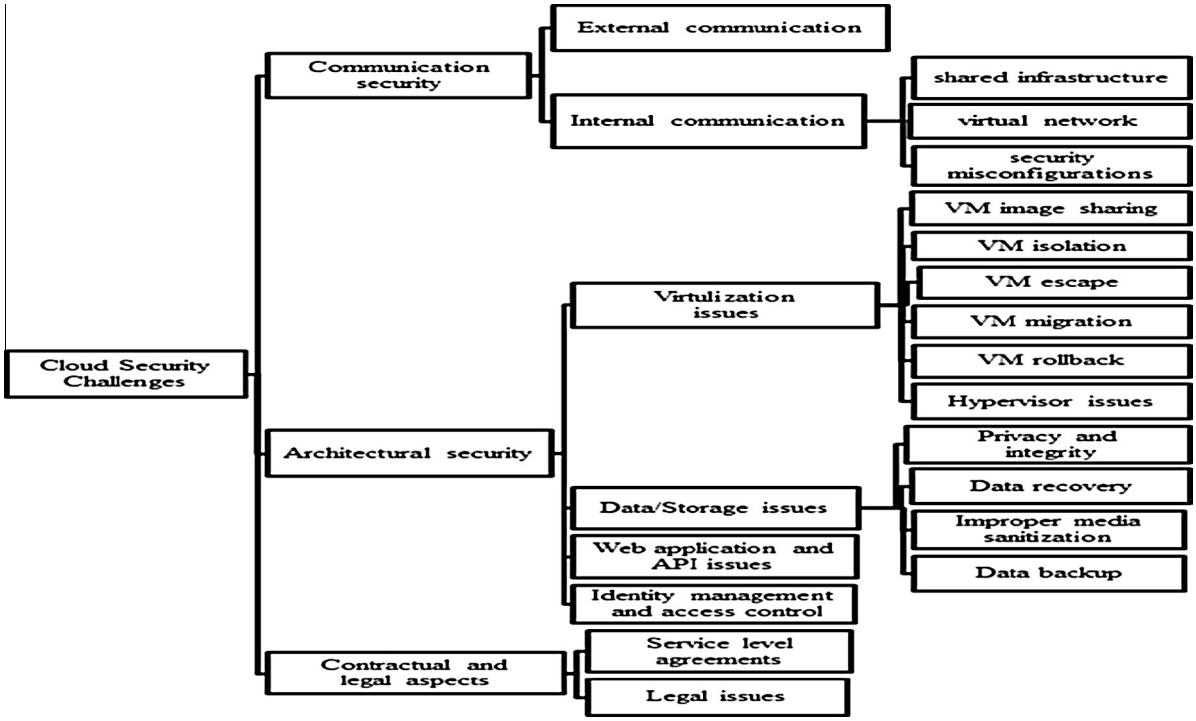

Fig. 2. Cloud security challenges.

The external communication of cloud is similar as any other communication over the Internet. Therefore the challenges faced by the cloud due to Internet characteristic are same as the challenges of conventional IT communication [30]. These challenges include denial-of-service, man-in-the-middle, eavesdropping, IP-spoofing based flooding, and masquerading [94,62]. The solutions to these challenges are also the same as employed conventionally, such as, Secure Socket Layer (SSL), Internet Security Protocol (IPSec), cryptographic algorithms, intrusion detection and prevention systems, traffic cleaning, and digital certificates [94,43]. As these challenges and solutions are inherent from conventional IT infrastructure, we do not detail them in this study. We focus on internal cloud communication that generates cloud specific challenges because of cloud specific characteristics and technologies.

3.1.1. Shared communication infrastructure

Resource pooling not only results in sharing of computational and storage resources but also sanctions the sharing of network infrastructure components [28]. The sharing of network components provides attacker the window of cross-tenant attack [39]. The vulnerability stems from the resource pooling characteristic of the cloud computing and affects the IaaS service model of the cloud. Due to the fact that it is hard to distinguish between a legal vulnerability scan of network and attacker activity, usually such scans are not allowed by the service providers. Similarly, the IP-based segregation of network portions are not applied as network resources are dynamically provisioned and released and cannot be associated to particular set of users.

The users on the cloud are usually granted with the super-user access for the purpose of managing their VMs [12]. The access capability empowers the malicious user to acquire system IP or MAC addresses and make malicious use of IaaS network interfaces. The malicious user with super-user access to the real network components may launch attacks, such as, sniffing and spoofing over the real network.

3.1.2. Virtual network

In cloud computing systems, the communication takes place not only on real networks but virtualized networks also play an important role in communication. Virtual network is a logical network built over a physical network [116]. The virtual networks are responsible for communication between VMs. The software-based network components, such as bridges, routers, and software-based network configurations, support the networking of VMs over the same host. The virtualized networks are able to generate the following security challenges in the cloud environment.

Security and protection mechanisms over the physical network are not able to monitor the traffic over virtualized network. This becomes a serious challenge as malicious activities of the VMs go beyond the monitoring of security tools. Intrusion detection and prevention mechanisms usually depend on the traffic patterns and activities to judge the anomalies and detect the possibility of the attack. Virtualized network poses a hindrance to the goal of such preventive measures [47].

The virtualized network is shared among multiple VMs that causes the possibility of certain attacks, such as, Denial of Service (DoS), spoofing and sniffing of virtual network. The traffic rates can be monitored for malicious purposes. The cryptographic keys become vulnerable to leakage, in case of malicious sniffing and spoofing of virtual network [33]. The data in transit belonging to users can suffer from costly breaches due to risks presented in Section 3.1.1 and Section 3.1.2.

3.1.3. Security misconfigurations

Security configurations of the cloud network infrastructure are of significant importance in providing secure cloud services to the user [24]. Misconfigurations can radically compromise the security of customers, applications, and the whole system [26]. Customers outsource their applications and data to the cloud with the trust that their assets are secure within the cloud environment. A small misconfiguration can breach the security of the system. The configurations need to be well in place not only at the time of cloud infrastructure development, deployment, and operations but subsequent changes in the cloud network should also keep the configuration consistent with the security policies [24]. One of the most common misconfiguration occur when administrators select such a configuration tool that they are familiar with but not necessarily covers all the security requirements [24]. The migration of VMs, data, and applications across multiple physical nodes, changes in traffic patterns, and topology can generate the requirement of varied security policies [66]. In such a scenario, the configuration of the cloud should dynamically be managed to ensure the security of the cloud. Likewise, any weakness in session configurations and protocol configurations can be exploited for session hijacking and to gain user sensitive data [101].

3.2. Challenges at architectural level

3.2.1. Virtualization issues

Virtualization is one of the strategic components of the cloud. Virtualization allows the use of same physical resources by multiple customers. A separate VM is instantiated for each user that virtually provides a complete operating machine to the user [74]. Several VMs can be mapped to the same physical resources allowing the resource pooling in multi-tenant environment. A VM monitor (VMM) or hypervisor is the module that manages the VMs and permits various operating systems to run simultaneously on the same physical system [74]. Nevertheless, virtualization also introduces security challenges to the cloud users and infrastructure [83]. We discuss the security issues related to virtualization below.

3.2.1.1. VM image sharing. A VM image is used to instantiate VMs. A user can create his/her own VM image or can use an image from the shared image repository [39]. The users are allowed to upload and download images from the repository (for example Amazons image repository) [39]. Sharing of VM images in the image repositories is a common practice and can evolve as a serious threat if it is used in malicious manner [47]. A malicious user can investigate the code of the image to look for probable attack point. On the other hand, a malicious user can upload an image that contains a malware [47]. The VM instantiated through the infected VM image will become source of introducing malware in the cloud computing system. Moreover, an infected VM can be used to monitor the activities and data of other users resulting in privacy breach. Likewise, if the image is not properly cleaned, it can expose some confidential information of the user [39].

3.2.1.2. VM isolation. VMs running on the same physical hardware need to be isolated from each other. Although logical isolation is present between different VMS, the access to same physical resources can lead to data breach and cross-VM attacks. Isolation is not only needed on storage devices but memory and computational hardware also needs fine grained isolation of VMs [33,101].

3.2.1.3. VM escape. VM escape is a situation in which a malicious user or VM escapes from the control of VMM or hypervisor [97]. A VMM is a software component that manages all the VMs and their access to the hardware. The VM escape situation can provide attacker access to other VMs or can bring the VMM down [47]. A successful VM escape attack can provide access to the computing and storage hardware. The IaaS service model is affected that can in turn effect other service models [73].

3.2.1.4. VM migration. The VM migration is the process of relocating a VM to another physical machine without shutting down the VM [128]. The VM migration is carried out for a number of reasons, such as load balancing, fault tolerance, and maintenance [39,20]. During the migration phase, the contents of the VM are exposed to the network that might lead to data privacy and integrity concerns. Besides data, the code of VM also becomes vulnerable to attackers during migration [128,47]. The migration module can be compromised by an attacker to relocate the VM to a compromised server or under the control of compromised VMM. The VM migration is a crucial phase and needs to be carried out in a secured manner [39].

3.2.1.5. VM rollback. Virtualization allows the rollback of a VM to some previous state whenever it is needed. The rollback feature provides flexibility to the user. However, rollback also raises security concerns [90]. For example, the rollback can enable the security credentials that were previously disabled [39]. Moreover, the rollback can also render the VM to a vulnerability that was previously patched [116]. Furthermore, the rollback can revert the VM to previous security policies and configuration errors [39].

3.2.1.6. Hypervisor issues. The key module of virtualization is hypervisor or VMM. The VMs management and isolation is the responsibility of the VMM. Generating and managing virtual resources, is yet another function performed by the VMM. A VMM may affect the execution of VMs running on the host system [106]. A compromised VMM can put all the VMs that are managed by the victim VMM under attacker's control [102]. The metadata of the VMs, kept by the VMM, may also be exposed to an attacker if the attacker takes control of a VMM [128,102]. A VMM can provide larger attack vector due to more entry points and interconnection complexities [102]. There are many reported bugs in the VMM that let the attacker to take control of the VMM or bypass security restrictions. For example, vulnerabilities in the Xen, Microsoft Virtual PC, and Microsoft Virtual Server can be abused by attackers to gain privileged rights [102].

3.2.1.7. VM sprawl. VM sprawl is a situation where a number of VMs on the host system is continuously increasing and most of the already instantiated VMs are in idle state [80]. The VM sprawl causes the resources of the host machine to be wasted on large scale [97].

3.2.2. Data/storage issues

The cloud computing model does not deliver users with full control over data. Distinct to conventional computing model, the cloud computing permits the service providers to exercise control to manage servers and data. The user enjoys certain level of control only on the VMs [114]. The lack of control over the data results in greater data security risks than the conventional computing model. Moreover, the characteristics of cloud computing like multi-tenancy and virtualization also come up with the possibilities of attacks different than the conventional computing model. Below we provide an overview of the security challenges faced by the data in cloud computing environment.

3.2.2.1. Data privacy and integrity. Although the cloud computing ensures the cost economy and also relieves the users from infrastructure management activities, it also entails security issues. The data in the cloud is much more vulnerable to risks in terms of confidentiality, integrity, and availability in comparison to the conventional computing model [110]. The ever increasing number of users and applications leads to enhanced security risks. In a shared environment, the security strength of the cloud equals the security strength of its weakest entity [93]. Not only the malicious entity collocated with the victim data, but also any non-malicious but unsecure entity can result in breach of data. A successful attack on a single entity will result in unauthorized access to the data of all the users. Violation of integrity may also result from multi-tenant nature of the cloud. Employee of SaaS providers, having access to information may also act as a potential risk [39].

Besides the data at rest, the data being processed also comes across security risks [99]. Due to virtualization physical resources are shared among multiple tenants. This eventually may allow malicious users (sharing computing resources) to launch attacks on the data of other users while in processing phase [39,65]. Moreover, if the data backup process is outsourced to a third party by the CSP, risks boundary is also broadened [39].

The cryptographic key generation and management for cloud computing paradigm is also not standardized [47]. Absence of secure and standard key management techniques for the cloud does not allow the standard cryptographic mechanisms to scale well to the cloud computing model [47]. Therefore, domain of cryptography also enhances the potential risks to the data.

3.2.2.2. Data recovery vulnerability. Due to resource pooling and elasticity characteristics, the cloud ensures dynamic and on-demand resource provisioning to the users. The resource allocated to a particular user may be assigned to the other user at some later point of time. In case of memory and storage resources, a malicious user can employ data recovery techniques to obtain the data of previous users [10,28]. The authors in [10] were able to recover Amazon machine images files 98 % of the times. The data recovery vulnerability can pose major threats to the sensitive user data [17].

3.2.2.3. Improper media sanitization. The issue is related to the destruction of physical storage media due to a number of reasons, for example, (a) the disk needs to be changed, (b) the data no longer needs to be there, and (c) termination of service [47]. If the CSP does not sanitize the devices properly, the data can be exposed to risks [107]. Sometimes, the multi-tenancy also contributes to the risk of device sanitization. At the end of the device life cycle, it may not be possible to destroy it as it is in use of some tenants [28].

3.2.2.4. Data backup. The data backup is also an important issue that needs to be dealt carefully. A regular data backup is needed at the CSP side to ensure the availability and recovery of data in case of intentional and accidental disasters. Moreover, the backup storage also needs to be protected against unauthorized access and tampering [101].

3.2.3. Web application and application programming interface (API) security

As discussed in Section 1, services and applications to the cloud users are provided through the Internet [16]. In fact, it is one of the essential requirements for a cloud application to be utilized and managed over the Web [127]. The application provided by the CSP is always located at the cloud with users accessing it ubiquitously. One of the important characteristics of cloud applications is that they are not bonded with specific users [101]. Different users may access the same application possibly at the same time. The cloud applications inherit the same vulnerabilities as traditional Web applications and technology. However, the traditional security solutions are not adequate for the cloud computing environment because the vulnerabilities in web application in cloud can prove to be far more devastating than the traditional Web applications. Co-location of multiple users, their data, and other resources makes it much greater issue. The top ten risks in the web applications have been identified by Open Web Application Security Project in 2013 to be the following [75].

- Injection (SQL, OS, and LDAP)
- Broken Authentication and Session Management
- Cross-Site Scripting (XSS)
- Insecure Direct Object References
- Security Misconfiguration
- Sensitive Data Exposure
- Missing Function Level Access Control
- Cross-Site Request Forgery (CSRF)
- Using Known Vulnerable Components
- Invalidated Redirects and Forwards

The development, management, and use of Web applications must take into consideration the above given risks to safeguard the web applications and users resources. The user and the services in the cloud are bridged by the APIs. The security of APIs highly influences the security and availability of the cloud services [126]. The secure APIs ensure the protected and non-malicious use of the cloud services [11]. An API can be thought of a user guide that describes the details about the CSPs cloud architecture and features. The users build or extend the services using the APIs [126]. The CSPs usually publish their APIs to market the features of their cloud. At one hand, the publishing of APIs helps the users to know the details about the components and functions of the cloud. On the other hand, the cloud architecture to some extent is exposed to the attackers [126]. Therefore, insecure APIs can be troublesome for both the cloud and the users. The vulnerabilities of APIs include weak credentials, insufficient authorization and input-data validation. Moreover, the frequent updates of APIs may introduce security holes in the applications [14].

3.2.4. Identity management and access control

In a cloud environment, the confidentiality and integrity of data and services is also linked with the identity management and access control. It is exceptionally important to keep track of the user's identity and controlling unauthorized access to

the information [61]. The issue of identity management and access control becomes more complex in a cloud environment due to the fact that the owner and resources are in different administrative domains and organization's authentication and authorization may not be exported to the cloud in the existing form [47]. Moreover, unlike the traditional IT setup, the cloud may deal with users of different organization with different authentication and authorization frameworks, at the same time and with the same physical resources [61]. The use of separate authentication and authorization systems for internal organization and cloud may give rise to complex situations over time [47]. The cloud services are elastic and dynamic, the IP addresses are frequently reassigned, the services are started or re-started over shorter periods of time, pay-as-you-use feature allows the users to join and leave cloud frequently. All these characteristics demand that conventional identity management and access control systems are not enough for the cloud environment [14]. A cloud needs a dynamic, fine-grained, and strict access control mechanisms to control unauthorized operations within the cloud [50]. Moreover, there is need of some control of organizations over identity management system to quickly update the access control policies in case of newly joining and leaving employees [101]. The are many issues that can arise in cloud due to weak identity management and access control, for example, denial of service by account lock-out, weak credential reset mechanisms, insufficient authorization checks, cross domain authentication, insufficient logging and monitoring possibilities, weakness of eXtensible Access Control Markupup Language (XACML) messages, and XML wrapping attacks.

3.3. Challenges at contractual and legal levels

Adopting the cloud computing, results in moving the organizations data and applications to the administrative control of CSP. This brings many issues to the front, for instance, performance assurance, regulatory laws compliance, geographic jurisdictions, monitoring of contract enforcement, etc. The aforesaid problems are related to the service level agreement (SLA), legalities, and physical locations of the data.

3.3.1. Service level agreements

The SLA is a document that specifies the terms and conditions between the user and CSP. The SLA also indicates (a) minimum performance level that CSP has to provide, (b) counteractive actions, and (c) consequences in case of breach of the agreement between user and CSP [5]. The users must be very clear about security requirements for their assets and all the requirement should be thoroughly agreed upon in the SLA. In case of ambiguities, it is harder to claim the loss at a CSP. For example, if a CSP sub-contracts any service to a third party then in case of a problem it becomes hard to claim at CSP. Accountability of a sub-contractor is often inadequate [3]. Likewise, monitoring of contract enforcement becomes an issue because the users cannot totally rely on statistics provided by the CSP. In such a case of conflict between the CSP and user statistics, evaluation of statistics and determination of responsibility also becomes an issue [33]. Moreover, the contracts are pre-defined and non-negotiable that results in CSP friendly agreements [3]. Furthermore, audit about the security provided by the CSP is harder to carryout and even agree upon in SLAs. For instance, it is difficult to measure that logical segregation of different organizational data is provided to the level as promised in the SLA. A third party audit may put the data of other organizations (that do not agree upon the audit conducting third party) to risk [3]. Enforcement of regulatory laws, such as Health and Human Services Health Insurance Portability and Accountability Act (HIPAA) and Sarbanes-Oxely also becomes an issue [68].

3.3.2. Legal issues

Besides the technical issues presented in the preceding discussion, legal issues pertaining to the cloud computing also arise due to presence of CSP resources in geographically different and sometimes conflicting legal jurisdictions [40]. If the data of the user is migrated to a location having different laws, it becomes difficult for the user to configure the security policies to comply with the new legal jurisdictions. Sometime, the data may be present in more than one location having different laws about digital security. Moreover, in case of a dispute the issue of jurisdiction arises as to which laws would be applicable [123,33,3]. The E-discovery poses another security issue. The E-discovery refers to an issue that arises when the hardware of the CSP gets seized for investigations related to particular customer according to the laws of geographic location. Such a case, results in risk of privacy breach of other users [89,33].

4. Security solutions in literature

In this section, we discuss various approaches proposed in the literature to counter the security issues discussed in Section 3. The discussed approaches are proposed to counter either one or multiple security issues. In each category, before going into the details of the counter measures, we describe the recommendations specified by the CSA in that particular category.

4.1. Counter measures for communication issues

To secure the communication and network, the CSA guidelines [21] recommend the use of a combination of virtual LANs, IDS, IPS, and firewalls to protect the data in transit. The guidelines also focus on leakage of customers data due to a virtual

network and the use of same underlying infrastructure. The CSA recommends the use of aforementioned tools with strict access management policies. Use of virtual devices and conventional physical devices with close-fitting assimilation with the hypervisor is endorsed by the CSA to ensure visibility and monitoring of traffic over the virtual network.

Advanced cloud protection system (ACPS) is proposed in [67] that aim at providing greater security to the cloud resources. The ACPS provides various security services to the CSP resources including network against attacks on user and CSP data. The cross tenant attacks are also neutralized by constant monitoring of the VMs running at host platform. Additionally, the ACPS also provides auditability for the actions of VMs. The ACPS is divided into multiple modules located at the host platform. The interceptor module is responsible for detecting any suspicious activities at the host. The detected suspicious activities are recorded by the warning recorder module and are stored in the warning pool. The assessment of recorded activities is performed by the evaluator. An increase in rate of warning generations is treated as a security threat that activates the actuator module for reaction according to the security policies. The ACPS computes the checksums for critical infrastructure including the network at the setup time. The state of the infrastructure is asynchronously determined by re-computing checksum for the scrutinized objects. In case of anomalies the warnings are sent to the evaluator. The periodic checksum verification also keeps the cloud entry points under constant monitoring. To prevent the attacks on network infrastructure, the ACPS utilizes the method presented in [84] where network probing is detected by using IP tables and warnings are recorded in the warning pool. In addition to securing network and other critical infrastructure, the ACPS provides security against malicious VMs and data attacks. One of the important features of the ACPS is that it remains transparent to the VMs and remains undetectable. The interceptor module does not block any system call to prevent itself being detected. However, if the attack activity is confirmed then the action is taken. Allowing the initial system call to be executed neutralizes the timing attacks for detection of any monitoring system. The ACPS prototype was implemented on Eucalyptus and OpenECP that are open source cloud systems.

A security tool for the cloud computing, called CyberGuarder proposed in [59] provides virtual network security through the deployment of virtual network devices. Moreover, virtual network isolation is introduced by utilizing layer-two tunnel Virtual Private Network (VPN) between virtual bridges. The data is transmitted between VMs in peer-to-peer (P2P) manner without transiting through the central server. However, the metadata is stored on the central node for optimized traffic between the VMMs. The software ports are designed to monitor the network traffic. Conventional network security systems like Intrusion Detection System (IDS) are adaptively deployed into the virtual network for security of applications running on the virtual network. Additionally, the CyberGuarder also provide VM security through the integrity verification of applications and by monitoring of system calls invoked by the applications. The experimental results showed a 10% overhead in performance due to Cyberguarder and 5% increase in the energy consumption.

Wu et al. [116] proposed a virtual network model that safeguards the virtual networks against sniffing and spoofing attacks. The Xen hypervisor is used to demonstrate the proposed model. The proposed model utilizes both the bridge and route modes of Xen hypervisor for virtual network configuration. In bridge mode the Xen attaches the VM directly to the virtual Ethernet bridge. The bridge in turn connects to the physical network. The route mode creates a P2P link between the VM and the domain 0 (the VM management domain). The proposed model is divided into three layers, namely: (a) routing, (b) firewall, and (c) shared network layer. The routing layer establishes a dedicated logical channel between virtual and physical network. Each channel is assigned a unique logical ID that is used to monitor the source of packets originating from the shared network. The firewall layer is responsible for safeguarding against the spoofing attacks from the shared network. This layer guarantees that any virtual interface connected to a shared virtual network does not communicate with any other virtual shared network. The monitoring is performed based on the logical IDs assigned by the routing layer. Secondly, firewall layer does not allow the packets to update the routing table. All such packets are discarded. The shared network layer prohibits the communication between VMs belonging to different virtual network channels.

He et al. presented a cloud network security solution in [42] by implementing a novel tree-rule firewall. The authors demonstrated that the conventionally used listed-rule firewalls are prone to security issues of shadowed rule, swapping positions, and redundant rules. Moreover, the listed-rule firewalls decrease performance due to sequential rule searching and arrangement of bigger rules after the smaller rules. To remove the aforementioned problems, the authors proposed the tree-rule firewall that arranges rules in the form of a tree instead of list. The scheme compares the first attribute of the packet header at the root nodes of the tree and on matching node search proceeds to next level of tree. The next level will check for the next attribute and the process continues till the firewall reaches the specified security policy for the given attributes. The tree structure for the proposed technique is shown in Fig. 3. The sequence of attributes can be decided according to the requirements of the system. For example in Fig. 3, the source IP can be at root with the destination IP at leaf nodes.

Authors in [72] presented a technique named DCPortalsNg for isolation of virtual networks for various VMs. The presented technique follows Software-Defined Network (SDN) methodology for isolating virtual network. The DCPortalsNg interacts with the open stack through a neutron plugin and obtains all of the required virtual network information. The DCPortalsNg then builds its own data of mapping networks to tenant and tenants to network. Subsequently, a unique identifier is assigned to each of the VMs. For network isolation, the concept of packet rewriting is used that opens the original packet and extracts source and destination addresses from the packet. The packets destined for the same network are further processed while other are discarded. In case of a valid transmission, the OpenFlow message is sent to the appropriate virtual switch to rewrite the packet with destination/source IP addresses replaced with identifiers. Moreover, the MAC addresses are replaced by the MAC addresses of the physical host. This avoids the cross tenant attack on the virtual network. The traffic is controlled by

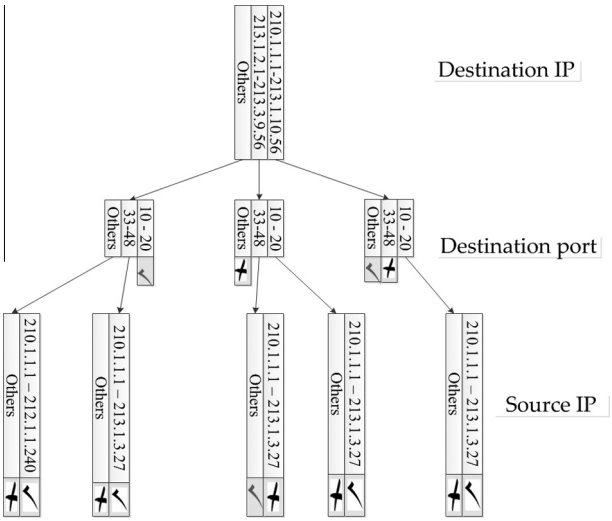

Fig. 3. Design of tree-rule firewall using IP address and port ranges [69].

MAC addresses only in the presented technique. The presented technique also prevents the cross VM denial of service (DoS) attack.

Xing et al. [120] proposed a system called SnortFlow for intrusion prevention within cloud environment. The SnortFlow utilizes the features of Snort and OpenFlow systems. The prototype of SnortFlow is built and tested over Xen-based cloud. The suspicious traffic is collected by the component called snortFlow demon. The alert is pushed into alert interpreter that analyzes the generated alert and invokes the rules generator. The rules generator develops the rules for the suspect traffic and forwards them to the openflow device. The openflow device reconfigures the network according to the developed rules. The evaluation of SnortFlow exhibited good performance in terms of traffic analysis and prevention against intrusion.

Table 2 presents a summary of the schemes that are presented as countermeasure for communication issues. The Table provides salient security features provided by the scheme and the technique used to provide the security measures. Moreover, it also highlights the scalability of the presented work.

Table 2
Comparison of techniques countering communication issues in cloud.

Work	Proposed scheme	Security features	Basic theory	Scalability
[67]	Architecture for monitoring integrity of VM and infrastructure components	• Secures network and other infrastructure • Avoids cross VM attacks • Auditability of VM actions	Computation of integrity checksum	Moderate
[59]	Application for virtual network security	• Secures virtual network • Secures VM • VM and network isolation	• Layer-2 tunnel VPN • Virtual IDS • VM Integrity verification • System call monitoring	Moderate
[116]	Model for virtual network security	Secures virtual network against sniffing and spoofing	• Combine bridge and route modes of Xen hypervisor • Firewall component to safeguard routing table • Logical IDs assigned to channels	Low
[42]	Cloud network security	• Eliminates shadowed and redundant firewall rules • Non sequential firewall rule searching	• Firewall • Tree based rules data structure • Non sequential search to improve performance	High
[72]	Application for virtual network security	• Isolates virtual network for every VM • Safeguards against cross tenant DoS attack	• SDN • OpenFlow • Packet rewriting	Moderate
[120]	Application for intrusion prevention	Safeguards against intrusions	• A mix of snort and OpenFlow • Firewall	Moderate

4.2. Counter measures for architectural issues

4.2.1. Virtualization

The VM images are entities that require high security and integrity as they specify the initial state of the VM. Moreover, the VM images are mostly used by various and unrelated users. Therefore, the security of VM images makes the basis for the security of the whole cloud computing system. The CSA recommends the following major measures [21] to safeguard the cloud computing system against the vulnerabilities introduced by virtualization.

- The implementers should secure each virtualized OS in each of the guest VMs.
- Built in security measures should be adopted for virtualized OS.
- Third party security technology should be used to cut down dependency on the CSP.
- The VMs at rest should be encrypted.
- Security vulnerability assessment tools should cover the virtualized environment.
- VM images at rest should be patched with the latest fixes as soon as required. Moreover, the protection mechanism should be in place until VMs are patched.
- Virtualization aware security tools should be implemented and used in the cloud computing environment.

Wie et al. [113] proposed Mirage, an image management system for the cloud environment. The Mirage provides a four-fold security to the VM images. The publishing and retrieval of the VM images is regulated by an access control framework. The filters are applied to the images both at publishing at retrieval time to detect and remove the unwanted information. A tracking mechanism is utilized to keep track of an image both in terms of auditability of actions and derivation. Moreover, maintenance of repository is also provided by the Mirage. The access control is provided at check-in and checkout times. Publishing, retrieving, and modification of VM images require proper permissions. The filters remove any leftover private information, malware, and pirated software from the image. Maintenance services execute periodic running of malware detection tools for the images in the repository and discovering vulnerabilities and patches. The prototype of the system revealed better security results.

In [51], the authors proposed encrypted virtual disk images in cloud (EVDIC) that exploits encryption to secure the VM images on the disk. The image encryption module encrypts an image whenever a VM is terminated. The EVDIC uses advanced encryption standard (AES) with a key size of 256 bits. The key is generated by key management server (a third party that is not a part of the cloud) through the password of the user. The encrypted image is then stored on the disk. During retrieval, the image decrypt module interacts with the key management server to retrieve the decryption key and decrypts the image for loading into a VM. The EVDIC also stores integrity information for the VM images. Therefore, it provides confidentiality and integrity services to the VM images. Moreover, any sensitive data loaded into the image is also protected by the proposed scheme. The authors integrated the EVDIC with OpenStack to reveal the successful interoperation.

To mitigate the vulnerabilities in VMs by patching fixes, Schwarzkopf et al. [90] proposed an approach that checks for the outdated software and vulnerabilities in the VM images. The proposed scheme handles both live and dormant virtual machines. Two modules work in the proposed scheme, namely: (a) update checker and (b) Online penetration suite (OPS). The update checker keeps the record of all of the software being used by VMs in the cloud setup, the version numbers, the update releases, etc. The VM at the time of registration is checked for software and record is kept that is matched against installed and available packages. The matching results in detection of any obsolete software. All of the information about software packages and the VMs is stored at the central database. Besides checking at the registration time, the update checker is invoked periodically to scan the VMs. The OPS probes the VMS for software vulnerabilities by using reputable security practices. The results of the update checker and OPS are generated in the form of report to inform both the user of VM and the system administrator. The advantage of the scheme is that the user can keep the VMS up-to-date and administrators can have a check that outdated software does not run on their system. However, the process of updating VMs is a manual process. The proposed technique is implemented on Xen grid engine and image creation station.

The ImageElves is another technique proposed in [48] targeted at providing updated software installs, and patches for the VMs in the cloud. The basic working of ImageElves resembles the technique presented in [90] in that it keeps record of all the software running on the VMs. It checks for the updates of the installed software and identifies the VMs (both dormant and running) that need to be updated. The difference in both techniques, however, is that ImageElves automatically updates the VMs. The ImageElves groups the similar VMs into classes and applies updates to those classes. The update is first installed on a single VM of the class and the image of update is created alongside. In case of successful update, the image is applied to all other VMs of that particular class. The process reduces the time consumption of each VM for proper functioning after updates and roll backing in case of errors. The ImageElves works both on the running and dormant VM images.

An Offline Patching Scheme (OPS-offline) is introduced in [27] to identify and rectify images with outdated software and malware vulnerabilities. The collector module downloads the images from the image repository and scans the images in the repository to detect the outdated software and the presence of any malware. The patcher module runs after the collector to patch the rectified vulnerabilities. The scheme works only on dormant images in the image repository. The OPS-offline is implemented on OpenStack Glance image repository. Table 3 presents the comparative summary of the presented schemes for VM image security.

Table 3
Comparison of presented techniques for securing VM images.

Work	Proposed scheme	Handled images type	Privacy	Integrity	Access control	Outdated software detection	Leftover owner's data removal	Malware protection	Scalability	Other features
[113]	Mirage, a VM image management system	Dormant	✗	✗	✓	✓	✓	✓	High	Auditability
[51]	EVDIC, for VM image's privacy and integrity	Dormant	✓	✓	✓	✗	✗	Dormant images only	Medium	-
[90]	A scheme for patch management for VM images	Running and dormant	✗	✗	✗	✓	✗	✗	High	Reports CSP about vulnerable VMs
[48]	ImageElves, for patch management for VM images	Running and dormant	✗	✗	✗	✓	✗	✗	Low	Automatic updating of outdated software
[27]	OPS-offline, for patch management for VM images	Dormant	✗	✗	✗	✓	✗	✓	Low	Automatic updating of outdated software

A VM needs to be protected against attacks not only in repository but also needs to be secured during execution time. The authors in [58] proposed an architecture that provides a secure runtime virtualization environment to a VM. The proposed architecture assumes management domain (Dom0) of Xen virtualization structure to be untrusted alongside conventional attackers. The mechanism to handle untrusted Dom0 amplifies the security level of the runtime environment of a VM. The security mechanism proposed in [58] prohibits any memory access from Dom0 to DomU (user domain) using foreign mapping. Any memory access to the DomU is allowed after the grant of permissions by the DomU. Moreover, all the memory accesses from Dom0 to DomU are continuously monitored by the hypervisor. If the memory pages and vCPU contain private information of DomU, the hypervisor make sure that they are encrypted. Therefore, the Dom0 has only encrypted view of confidential memory regions. The hypervisor checks the integrity of the DomU state after every management function executed by the Dom0. The restart of VM is only allowed if the integrity checks are valid. The proposed architecture provides confidentiality, integrity, and availability services for VMs during execution phase.

The idea of decoupling the security and VM management to protect the runtime environment of the VM is utilized in CloudVisor [129]. A CloudVisor is a light weight security module that works beneath VMM using nested virtualization. A CloudVisor provides privacy and integrity to the VM resources (CPU, memory, and I/O devices) during execution. All the control transitions between VMM and VMs are intercepted by the CloudVisor to perform security operations. For example, the CloudVisor may hide the general purpose registers (by encrypting) from the VMM, while exposing only the necessary ones. A CloudVisor also monitors the address translation to enforce memory isolation. A VMM is prohibited to overwrite the extended page table for any VM in case of a mismatch between the page ownership and the page table. Moreover, in case of aforesaid mismatch it encrypts the contents of the page table. The encryption is performed over the virtual disk owned by a VM to protect its contents. The encryption and decryption is performed for every disk I/O by a VM. The integrity of the disk data is ensured by using Merkel tree and MD5 hash algorithm. The integrity of CloudVisor is also ensured using Trusted Platform Module (TPM).

The encryption and integrity checks have also been used in [118] to provide a secure runtime environment to the VMs in a cloud environment. The proposed scheme in [118], named HyperCoffer, also separates the security from the tasks of VM management as was proposed in [129]. The HyperCoffer trusts only the processor chip and considers everything else as the untrusted components. The HyperCo_er involves both the hardware and software to protect VMs in execution. The secure processor technology is used to encrypt memory data. The Address Independent Seed Encryption (AISE) and Merkle tree are used for encryption and integrity checking, respectively. To avoid cross VM attacks on data in cache, each cache line is tagged with a unique VM identifier Like CloudVisor, the HyperCoffer also secures the control transitions between VMM and VMs through encryption and integrity functions and exposes only the necessary information to VMM or other VM. The control transitions are monitored and secured by the software portion called VM-shim that works between the hypervisor and VMs. The encryption and decryption on disk and network I/O is also performed by the VM-shim. Besides privacy and integrity, the HyperCoffer also provides security against VM rollback by using logging and auditing. The proposed model is implemented and tested on QEMU full-system emulator.

The authors in [46] proposed CloudSec that monitors the VMs physical memory externally by using VM Introspection (VMI) technique. The CloudSec reenacts and screens the vigorously changing kernel data structures to detect and prevent against kernel data rootkits. Upon launching of a VM, the Cloud Sec identifies the memory layout of the VMs hardware by inspecting the control registers of the VMs CPU. Afterwards, the CloudSec requests for Kernel Structure Definition (KSD) through the hypervisor (a hypervisor is assumed to be a trusted entity in the CloudSec). After the specification of

KSD, the CloudSec maps the physical memory bytes (obtained through hypervisor) to the KSD that generates the operating system (OS) view of the live VM. Afterwards, the memory access and time based triggers are installed by the CloudSec on the memory pages that need to be monitored. The generated OS view is used by the defense modules of the CloudSec. In case of access request for the monitored memory page, the defense module halts the execution of the VM, loads the required pages to KSD and evaluates the KSD for security in the new state.

Another VMI based technique to monitor the execution of VM externally, called exterior, is proposed in [31]. Exterior is a dual VM architecture that launches a Secure Virtual Machine (SVM) for executing a guest virtual machine (GVM). The SVM executes the kernel that is similar to the kernel of GVM. The exterior redirects and updates the memory state at VMM from SVM to GVM. This result in the modification of the GVM and gives the effect that program is being run in the GVM. All the programs can be run entirely exterior to the OS. The kernel data rootkit attacks and intrusions are detected by introspection of code in the SVM. In case any hidden malicious process or device driver is detected, it is removed from the GVM. The inspection utility is used in the SVM to introspect the code of GVM.

In a cloud environment, VMs are migrated between different physical locations and cloud facilities due to various factors, such as, load balancing, physical machine failures, energy savings, and hardware/software up gradations. The VM migration poses different security issues as discussed in Section 3.2 and, therefore, requires high security considerations. In the following we present techniques presented in the literature that handle VM migration. A comparative analysis of the schemes to provide secure execution of VMs is presented in Table 4.

Aslam et al. [9] presented a VM migration technique that allows VM migration only if the destination platform is secure up to the user defined level. The proposed technique rests on the foundations of trusted computing. A Trust Assurance Level (TAL) is introduced that specifies the trust level of the cloud platform. The TAL is computed using the credentials of the TPM and additionally the Trust Token credential that is proposed by the authors. The TPM credentials measure the trust level of hardware while Trust Token specifies the trust level of the software stack. The user specifies the required TAL (least, low, average, normal, high) during the VM launch process. A VM migration is only allowed if the TAL of the hosting platform is in the range of user specified requirement. The process can also be applied at the time of VM launch to guarantee the TAL of hosted platform. The authors assume Platform Trust Assurance Authority (PTAA) as a third party for trust certification. The proposed technique also lets the user to audit the TAL of the platform after VM migration to assure that his requirements are met.

The trusted computing technology has also been used by authors in [109] for secure intra-cloud migration of VMs. The authors not only ensure the integrity of the destination platform but also secure the migrating contents on the communication channel. The authors used a Virtual TPM (vTPM) bound with a VM that certifies the integrity of the VM. The vTPM is also migrated along with the VM to ensure the integrity of the VM during the migration process. Property based remote attestation is used to verify the integrity and security conditions of the remote host before migration. Afterwards, the source and destination hosts established a tunnel trusted channel by mutual authentication and establishment of a session key. The transfer of VM and vTPM is carried on the established trusted channel. The same channel can be used for transfer of multiple VMs. The cryptographic mechanisms are used to ensure confidentiality, integrity, and freshness of the transmitted data. Authors in [22] also utilized trusted computing for secure VM-vTPM migration. The security for migration is provided by using the key hierarchy of vTPM in [22].

Table 4
Comparison of techniques dealing with VM security during execution.

Work	Proposed scheme	Basic theory	Privacy	Integrity	Kernel rootkit	Scalability	Other features
[33]	Secure runtime environment for VM	• Cryptography • Access control	✓	✓	✗	Low	Availability
[129]	CloudVisor, Secure runtime environment for VM	• Decoupling of security and VM management tasks • Nested virtualization • Trusted computing • cryptography	✓	✓	✗	High	Security of Cloudvisor itself
[118]	HyperCoffer, Secure runtime environment for VM	• Decoupling of security and VM management tasks • Trusted computing • cryptography	✓	✓	✗	High	Security against VM rollback
[46]	CloudSec, an approach to detect and prevent memory based kernel rootkits	• Bridging of semantic gap between external and internal VMI • Construction of KSD externally	-	-	✓	Low	Live migration of VM in certain situations
[31]	Exterior, A dual VM architecture to secure VM execution	• VMI • Use of dual-VM for program execution	✓	-	✓	Low	• Intrusion detection • Removal of malicious code

The authors in [7] used multiple basic theories to propose a framework for secure live migration of VMs and to provide other security services, such as, privacy and integrity. The trusted computing is used for attestation and integrity verification of source and destination platforms. The attestation and integrity verification ensure that the VM is not migrated to a compromised host. The framework proposed in [7] also used role based access control policies to ensure security against VM hopping and useless migrations. The encryption and digital signature are used to protect data confidentiality and integrity during the VM transmission. A per-VM firewall (IP-table rules) is also implemented to control the communication of VM with other components. The proposed framework also introduced the host based firewall and intrusion detection system to provide network security for the overall host platform. The proposed framework was implemented on Xen hypervisor.

The authors in [105] proposed a framework that migrate not only the VM but the security context is also migrated to the destination host. The proposed framework migrates both the static and dynamic security contexts to ensure the same security status for a VM on the destination host as was present at the source. The migration of a VM is coordinated by the migration manager module that ensures the migration of all of the relevant information (VM state and security context state) to the destination. At first, the security context manager module migrates the static security context state, followed by the migration of VM state information by the VM state migrator module. The final phase is the migration of the dynamic security context to the destination host by the security context migrator. The proposed framework provides the same level of privacy and integrity at the destination as that of source host. However, it is not clear that how the information is secured during communication from source to destination. Table 5 presents the summarized properties of the discussed schemes dealing with the VM migration.

Hypervisor or VMM is software that essentially manages and controls the virtualization in a cloud computing system. The prime status of the VMM also makes it a key target for attacks. A compromised hypervisor can subvert the whole system. Therefore, the security of the hypervisor needs distinctive focus. In the following, we detail some of the solutions in the literature that aim at securing the hypervisor.

Zhang et al. [130] presented a framework named HyperCheck to ensure a secure execution of the hypervisor. The HyperCheck is a hardware assisted framework that uses the CPU system management mode (SMM) of x86 architecture for viewing the CPU and memory state of the machine. The SMM resides in the BIOS and is made inaccessible from other CPU modes. The SMM module reads and verifies the contents of the CPU registers of the protected hypervisor. The CR3 and IDTR registers are focused primarily as they play central role in rootkit detection. The PCI network card is utilized to read the physical memory. The driver for the PIC network card is handled in the SMM module to avoid any attack. The memory contents and the results of CPU registers verification are sent to a separate machine called monitor machine (that acts as a trusted third party). The analysis module on the monitor machine analyses the memory contents based on the linearity, stability, and perpetuity properties. Moreover, the complete snapshot is compared with the initial snapshot of the hypervisor. In case of any malicious activities, human operators are notified. The transmission of the contents to the monitor machine is performed through secure connection. The key for transmission is managed by the monitor machine. The HyperCheck was implemented both for open and closed source BIOS. The proposed framework showed detection and defense capabilities against rootkit, code and data integrity, DoS, and evasion attacks.

The large code base of the hypervisors broadens the attack surface of the hypervisors. The larger the code, the greater the number of points, that can be used to attack the hypervisor. The situation aggravates with the fact that the hypervisor runs in privileged mode. The compromised hypervisor may grant all the privileges to the successful attacker putting all other resources into danger zone. The authors in [115] utilized the principle of least privilege to reduce the attack surface of hypervisor and to ensure the security of other system components and resources. The proposed scheme divides the hypervisor (KVM was used to implement a proof-of-concept prototype) into two major components. The de-privileged DeHype component, that is decoupled from the OS and is executed in the user mode. The dependency decoupling is performed by dividing the code of the hypervisor into smaller modules. The modules that require little or no interaction with the OS are moved to de-privileged DeHype. The portions that require host OS for functionalities, were replaced by the user-mode equivalents.

Table 5
Comparison of techniques for secure VM Migration.

Work	Proposed scheme	Basic theory	Privacy	Integrity	Scalability	Other features
[9]	Secure and trust preserving VM migration mechanism	• Trusted computing • Remote auditing	✓	✓	Medium	Novel credentials for trust level quantification
[109]	Protocol for vTPM based VM migration	• Trusted computing • Remote auditing • Tunneled communication channel	✓	✓	Medium	Data freshness
[22]	Protocol for VM-vTPM migration	Trusted Computing	✓	✓	Medium	Migration initiation authenticity
[7]	Framework for secure live VM migration	• Trusted computing • Role based access control • Cryptography	✓	✓	Medium	Security against • VM hopping • Useless migrations
[105]	Framework for security context and VM migration	Migration of security context to ensure security	✓	✓	High	-

Moreover, the portions of the code that were not possible to move to user-mode, were kept privileged in a separate module called HypeLet. The aforementioned strategy resulted in movement of approximately 93% of the hypervisor code into user mode causing the reduction of hypervisor attack surface. The DeHype greatly reduces the risk of system subversion as most of the hypervisor code does not have privileges. Even the compromised hypervisor will not let the attacker take full control of the system. Moreover, the data leakage to the user space from the kernel space is avoided as the calls in DeHype are executed as a user space function call.

The authors in [111] adopted a similar approach to reduce the attack surface by providing an isolated runtime environment for hypervisor execution. The proposed scheme called HyperLock provides an isolated address space than the host OS and with the limited instruction set. The proposed strategy restricts the hypervisor to obtain direct access of the host system. Any access to the host system is regulated and mediated by the HyperLock. Moreover, the authors in [111] proposed hypervisor shadowing technique to further safeguard the VMs running on the host system. Shadow hypervisors are created and each of the VMs is paired with a separate shadow hypervisor. Therefore, a compromised hypervisor will only affect the paired VM keeping the other VMs on the host secure. The HyperLock also removes the Quick EMUlator (QEMU, a user program of KVM hypervisor) from the trusted computing base greatly reducing the attack surface. The prototype was implemented with KVM hypervisor.

Pan et al. [76] also reduce the trusted computing base and restrict the functionality of hypervisor in root mode for securing the hypervisor and running VMs. The proposed methodology makes use of the hardware extensions for virtualization in modern processors. The authors also divide the functionality of hypervisor into sub modules, namely: (a) Guestvisor and (b) Splitvisor. The Guestvisor runs in non-root mode excluding it from trusted computing base. The Splitvisor executes in root mode and is responsible for isolating multiple Guestvisors. The Guestvisor emulates the hardware for the VMs. The users can add and exclude functions (a hypervisor code is split into small functions) from the Guestvisor so as to avoid vulnerabilities due to unwanted functions.

The authors in [102] presented a design that does not reduce the hypervisor attack surface. Instead, it eliminates the hypervisor attack surface completely. The authors claim to implement a prototype in which multiple VMs can be run with the hypervisor. The management of the resources is accomplished either by the extended hardware capabilities or by the tiny system management software. The system named NoHype [102] is based on the following key ideas, (a) pre-allocation of the memory and cores, (b) use of virtualized I/O devices only, (c) system discovery process at the boot time of VM OS, and (d) avoiding indirections. The pre-allocation of resources eliminates the need of hypervisor to dynamically manage them. The use of virtualized I/O devices eliminates the need of emulating them. The modern hardware capabilities are used in NoHype to isolate the resources of multiple VMs from each other, for example, the hardware paging mechanism in modern processors. The guest OS is marginally modified to check for available system configuration and resources. This scan is only allowed at the boot up time with a temporary hypervisor so as to avoid any attack from user by executing malicious code. After the scan the temporary hypervisor is disabled. The indirections are avoided because of the dedicated cores and the hardware for the guest VM. The NoHype removes the hypervisor layer altogether, still providing the multitenant architecture by running VMs. The proposed strategy relies on the hardware capabilities to ensure isolation between VMs. Table 6 presents a comparative analysis of the detailed techniques for hypervisor security.

Besides the techniques presented in Table 1–6, we briefly go through some techniques handling different aspects of virtualization security, in the following text. The authors in [119] proposed a methodology to defend against the VM rollback

Table 6
Comparison of presented strategies for secure hypervisor.

Work	Proposed scheme	Basic theory	VM protection	Scalability	Other feature(s)
[130]	HyperCheck, a hardware assisted integrity monitor	• Hypervisor state monitoring through third party • Secure transmission of VMM state	✗	Low	• Data and code integrity • Security against Rootkit DoS, and evasion attacks
[115]	DeHype, a technique to reduce hypervisor attack surface	• Least privilege principle • Dependency decoupling between VMM and host OS • Reduction of TCB	✓	Medium	Prevents data leakage from kernel to user space
[111]	HyperLock, for isolating hypervisor from host OS	• Shadow hypervisor for every VM • Controlled access to host system • Reduction of TCB	✓	Medium	Exclusion of QEMU from
[76]	SplitVisor, for reducing root mode code	• Reduced functionality in root mode • Modern hardware virtualization • Reduction of TCB	✓	Medium	-
[102]	NoHype, for virtualization without hypervisor	• Elimination of hypervisor • Pre-allocation of Resources • Use of virtualized I/O only • No indirections	✗	Medium	-

attack by secure logging and auditing of VM operations (suspend, resume, migration). At the start of each operation the hash of the VM snapshot is calculated over its registers, memory contents, and image disk. The hash value at each state is subsequently used for later activation of the snapshot. A similar mechanism of logging and auditing to protect against the VM rollback is also used in [118]. Ref. [100] provides a mechanism (HyperShot) for secure snapshots of a VM by ensuring the integrity of the snapshots. The HyperShot uses TPM-based attestation, digital signature, trusted initialization of hypervisor to ensure the integrity of the snapshot and the hypervisor. To protect the private information from leakage due to rollback phenomenon, the authors in [32] proposed SPARC. The SPARC is secure check pointing mechanism that allows the users to exclude those processes and terminal applications from being check pointed that process the private information. By avoiding the check pointing of confidential information, the SPARC essentially reduces the life time of private information. The authors in [44] also proposed a strategy named Privacy-Preserving Checkpointing (PPC) for exclusion of confidential information from check points.

The PPC tracks the confidential information by information flow analysis and at time of snapshot, removes the confidential information. The authors of [77] leveraged the execution path analysis for introducing resilience in the cloud computing system. Execution path analysis allows the hypervisor to trace the VM state and avoid the VM to reach a faulty state. The proposed technique (CloRExPa) in [77] keeps track of execution and analyzes system behavior through meditation.

4.2.2. Data/storage security solutions

The following major recommendations of the CSA [21] are presented regarding the data security and key management.

- The key management should be performed by either the organizations/users themselves or by a trusted cryptographic service from a credible source.
- The best practices regarding the key management and encryption products from reliable sources should be used.
- It is recommended to use off-the-shelf-technology where possible.
- The key scope should be maintained at the individual or group level.
- The use of standard algorithms is recommended and proprietary encryption algorithms are discouraged.

Wei et al. [114] presented SecCloud, a storage security protocol that not only secures the user data uploaded into the cloud but also secures the computations performed on the user data. The SecCloud uses encryption for achieving the storage security. The bilinear pairing (with cyclic additive and multiplicative groups) is used to generate keys for the user, cloud, and a trusted third party. The user gets the storage space from the CSP to store data. The data (divided into m number of messages) is signed by the trusted third party (called the verification agency). The data along with the verifiable signatures is sent to the cloud by encrypting with the session key. The session key is calculated through Bilinear Deffie-Hellman both by the user and the cloud. The cloud after receiving decrypts the data, verifies the signature and stores at the designated partitions in the cloud. The computational security is ensured against partial computation and use of invalid data to save computational cost. It also verifies that data is stored at the correct partitions in the cloud. For the computation security the SecCloud utilizes Merkle hash tree. The computational results are verified by the verifying agency by rebuilding the Merkle tree. To reduce the computational redundancy, the verifier does not build the whole tree but uses probabilistic sampling.

The author in [98] used a combination of established and specialized procedures besides additional proposed steps to secure the data in the cloud. The proposed scheme allows the user to rate the requirement of confidentiality, availability, and integrity between values of one to ten (1–10). The values are used to determine Sensitivity Rating (SR) of the user data. Based on the SR value, the data is allotted space in one of the three proposed partitions in the cloud. The proposed partitions are public, private, and limited access partitions. The SR value above eight assigns data to limited access partition and below value three to public partition. The data is encrypted with 128-bit SSL encryption and MAC is appended afterwards. An index is also prepared and encrypted to employ searching capabilities over encrypted data. The data and index are sent to the cloud where they are stored depending on the SR value. The download is allowed based on user authentication that is carried out cooperatively by data owner and the cloud. The data in the public partition needs no authentication. The data is transmitted over SSL in both the directions.

To ensure the quality of the cloud storage, integrity and availability of data in the cloud, authors in [110] proposed effectual methodology that supports on-demand data correctness verification. The proposed methodology conducts the verification of the cloud data correctness without explicit knowledge of the whole data. The erasure correcting code and homomorphic tokens are used for the aforesaid purpose. The homomorphic token are pre-computed by the user and data is fragmented and stored redundantly across the cloud servers. To verify data correctness, a challenge containing random data blocks indices is transmitted to the cloud. The cloud computes the response and sends back to the user where decision is made based on the comparison of received result with the pre-computed tokens. Additionally, the proposed scheme performs error localization by detecting the misbehaving server. Moreover, insertion, deletion, modification, and appending of data blocks is supported in the proposed scheme. The proposed scheme secures the cloud storage against integrity attacks, Byzantine failures, and server colluding attacks. The authors in [60] utilized the concept of proxy re-encryption in addition to erasure correcting codes to provide resident and forwarding data security.

The File Assured Deletion (FADE) protocol that furnishes key management along with the data privacy and integrity is proposed in [103]. The FADE is a light weight protocol that uses both the symmetric and asymmetric encryption. The

symmetric keys are protected by using Shamirs (k, n) scheme to ample the trust level in the key. The FADE works with a group of key managers (KM) that act as a trusted third party. The data key (K) is used to encrypt a file F of the client. Another symmetric key S is used to encrypt K. The public/private key pair generated by KMs is represented by (e_i, d_i) and is used to encrypt S. A policy file P postulates the policies under which access to the file is valid. In order to upload the data, a user requests the KM to generate a key pair by sending P. The KM generates public/private key pair associated with the P and transmits public part to the user. The user encrypts the file with randomly generated K and encrypts K with S that is further encrypted with the public key generated by the KM. The MAC is also calculated with the integrity key. The whole encrypted package is stored at the cloud along with the P. For decryption all the data is downloaded from the cloud and S is sent to the KM for decryption through blinded RSA. Upon decryption of S other keys and subsequently F is decrypted. The FADE also supports policy renewal and revocation. Upon the expiration of the policy the KM deletes the corresponding keys and P through secure overwriting that makes the data inaccessible and therefore assuredly deleted. The working of FADE is depicted in Fig. 4.

The encryption of data before outsourcing to the cloud ensures the privacy of the data but poses certain restriction. The restrictions are specific to the situations where data is to be shared among the group and/or requires forwarding. Such an environment is accompanied with frequent user revocations that require the re-encryption of data with changed keys for avoiding data leakage to the revoked user. Liu et al. [64] proposed a time based proxy re-encryption combined with Attribute Based Encryption (ABE) to support secure data sharing in group along with the fine grained access control. The proposed scheme (TimePRE) ensures that data is securely forwarded to the group users and deals with the user revocation. Unlike other proxy re-encryption schemes, the TimePRE does not require the data owner to be online for user revocation and generation of new re-encryption keys. The TimePRE associates the time period with every user and upon expiration of the time period the user is automatically revoked by the CSP. A pre-shared master key between the data owner and the CSP allows the CSP to generate the re-encryption keys. The access control is ensured by use of ABE that identifies user by set of attributes rather than identity. The ABE in TimePRE uses eligible time periods for a user along with other attributes to identify a user. The proposed scheme ensures privacy and availability of the data within the group. However, it does not focus on the data integrity. Table 7 presents the comparison of the methodologies presented in Section 4.2.2.

4.2.3. Security solutions for cloud applications and APIs

The cloud applications and APIs on the SaaS and PaaS layers require special security attention to have secure development and execution life cycle. The cloud security alliance [21] recommends that the security to the cloud applications and APIs must be provided without any assumption about the external environment. The following are the focal recommendations by the CSA with respect to cloud applications and APIs.

- Security and privacy requirements (both functional and regulatory) should be defined in accordance to the needs of the cloud development and deployment. The defined requirements should also be in the order based on the impact and possibility.
- The risks and attack vectors specific to the cloud computing must be explored and assimilated into the security requirements. The risk models and attack models should be continuously built and maintained.
- The secure software development life cycle and software architecture should be developed and maintained.
- The re-useable software components that are known to alleviate the known security and breach scenarios should be used.

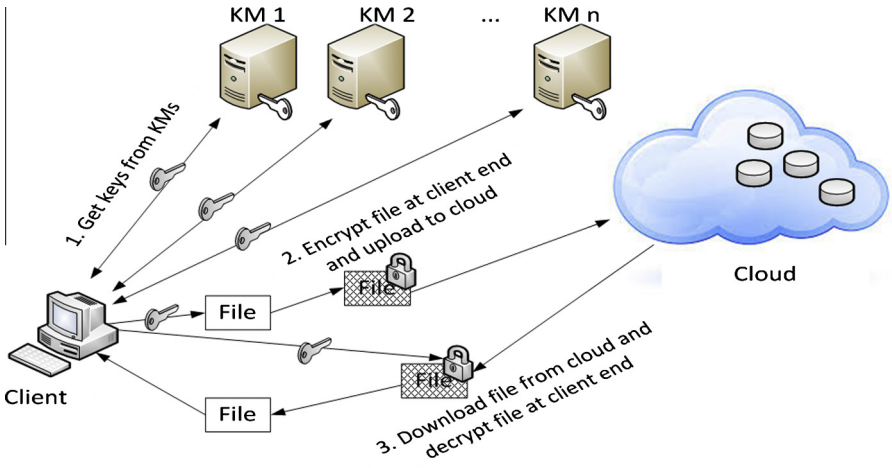

Fig. 4. System architecture of FADE protocol [103].

Table 7
Comparison of techniques presented for secure cloud storage.

Work	Proposed scheme	Basic theory	Privacy	Integrity	Availability	Scalability	Other feature(s)
[114]	SecCloud, a protocol for storage security and privacy	• Bilinear pairing • Trusted third party • Signature verification • Encryption	✓	✓	✗	Medium	Computational audit
[98]	A scheme for security of resident data	SSL symmetric encryption	✓	✓	✓	Low	• Access control • Searchable encryption
[110]	A methodology for security of resident data	• Erasure correcting Code • Data redundancy	✗	✓	✓	Medium	Secure against • Byzantine Failures • Server colluding
[103]	FADE, a protocol for data privacy and integrity	• Encryption • Trusted third party • Assured deletion • Threshold secret sharing	✓	✓	✗	High	• Access Control • Assured deletion
[64]	TimePRE, a scheme for secure data sharing in cloud	• Proxy re-encryption • Attribute based encryption	✓	✗	✗	Medium	Access control

- Regular penetration testing for web applications should be carried out.
- Manual tests must be carried out periodically to ensure secure session management of web applications.

To protect the cloud applications from unauthorized access, the authors in [91] proposed the use of Diameter-AAA protocol. The diameter-AAA employs network based access control to filter the illegitimate access request to the cloud applications. All the requests, initially received by the network access server, are forwarded to the diameter server. The server checks for the authentication and authorization parameters and based on the results the access is granted or denied for the requested application. Besides authentication and authorization the diameter protocol also provides the accounting services within the cloud.

Alowolodu et al. [6] proposed the use of TPM and Elliptic Curve Cryptography (ECC) to provide a secure platform for application execution in the cloud. The keys are generated by using ECC and are stored in the TPM configuration registers. The integrity of the platform is ensured before moving any application to it. The proposed methodology also recommends the use of encryption while moving applications between platforms. The integrity of the application is checked at the destination platform before launching.

The authors in [45] proposed the provision of Security as a Service (SECaaS) in the cloud environment. The SECaaS recommends the security services provided by different clouds and an in dependent cloud (manager cloud) that keeps track of these services. The user specifies the security requirements to the manager cloud that identifies the cloud(s) providing those services. The user application is then registered with the security providing clouds that provide security services. The SECaaS works at all levels (SaaS, PaaS, IaaS) and secures the services.

Ref. [117] proposed an API management platform for the cloud that provides access control architecture for the cloud APIs. The access control in the proposed platform is based on the OAuth (Open Authorization) that is token based access control mechanism. The token based access control uses tokens instead of user credentials in order to access the resources. The applications can use token on behalf of the user. In the proposed scheme, the API provider registers and publishes the API with the API management platform and obtains a key for validating the tokens. An API consumer requests for access token from the API management platform that is granted along with a key after validating the request. Both the keys, issued to API provider and consumer, are the private keys. The consumer calls the API by using the token signed with its private key. The provider sends the token to the API management platform for validation. If valid, the access is granted to the consumer. Table 8 presents the summary of the presented techniques.

Table 8
Comparison of strategies proposed for security of cloud applications and APIs.

Work	Proposed scheme	Basic theory	Security features	Scalability
[91]	Access control for cloud applications	Diameter protocol	• Authentication • Authorization • Accounting	High
[6]	Scheme for ensuring application integrity in cloud	• Trusted platform module • Elliptic curve cryptography	• Application integrity • Platform integrity	Low
[45]	Security as a service for cloud applications	Security as a service in clouds	As offered by security service by clouds	Low
[117]	API management platform for secure cloud APIs	Token based open authentication	Access control	Medium

4.2.4. Identity management and access control

Access control and identity management in cloud environment is highly needed to make the cloud computing adopted by the community, according to CSA. The CSA recommends the following key points for access control and identity management.

- Open standard federations, for example, SAML and OAuth, should be preferred if possible.
- The source of the attributes should be as close to master source as possible.
- The attributes should be validated at master source or as close as possible.
- All characteristics of the entities should have an identified trust level.
- Bi-directional trust should be ensured for secure relationship and transactions.
- The services should have import/export function into standards such as XACML and OASIS.

The Attribute Based Encryption (ABE) has been employed to provide access control in the cloud environment that specifies and enforces the access control policies cryptographically. The ABE was introduced in [92] and associates the encrypted messages using the attributes and decryption can be performed by users possessing those attributes. Attribute Set Based Encryption (ASBE) [13], an extension of the ABE, categorizes user attributes into a recursive set based arrangement and allows users to enforce dynamic constraints on how those attributes mutually fulfills access control policy. The authors in [108] extended the ASBE to present Hierarchical Attribute-Set-Based Encryption (HASBE) that utilizes hierarchical user structure. The HASBE assumes a hierarchy of users with trusted authority as root level authority. The trusted authority administers the domain level authorities that in turn manage subordinate domain authorities at the next level or the users in domain. The domain is trusted by the sub domain authorities or users in a similar fashion as in the certification authority's hierarchy. The trusted authority generates and distributes the system parameters and root master key to the domain authorities. The keys are generated using bilinear multiplicative groups. The system parameter includes the parameters to generate the groups. The public/private keys issued to the users by the domain authorities are hierarchical tree structures with each element being an attribute or set of attributes. The access control is also defined as a hierarchical tree structure. The data is encrypted with the data encryption key. The data encryption key is protected with the HASBE using the access key structure that specifies the access control policies and attributes. An expiration time is added to the access key structure for user revocation purposes. The access for decryption is granted to the users satisfying the attributes and policies in the access key structure. The HASBE ensures access control only for the cloud data. Fig. 5 represents the system model for HASBE.

Ruj et al. [86] proposed a decentralized approach for authentication and controlling access to the cloud storage. The proposed scheme in [86] makes use of ABE and the Attribute Based Signature (ABS) for access control and anonymous authentication, respectively. The anonymous authentication allows the user authentication without revealing the user identity. The signature is computed and verified based on the attributes that eliminates the requirement of identity for authentication. The scheme also makes use of a trusted third party that issues tokens to the users. The users present the tokens to the Key Distribution Center (KDC). The KDC issues the encryption/decryption and signing keys based on bilinear pairing. The user encrypts the data, signs, and transmits it to the cloud. The cloud verifies the signature that is attribute based and stores the data in case of valid user. The user revocation is dealt by changing the encryption parameters of all such data that has attributes similar to that of the revoked user.

The authors in [122] proposed Role Based Multi-tenancy Access Control (RB-MTAC) scheme that combines identity management and role based access control. The scheme requires the users to register with the cloud and obtain unique ID. The user sets the password during registration process. To enter the cloud, a user has to pass through identity management module that identifies the user on the basis of registered identity credentials. After identity verification the user is directed to the role assignment module that connects to the RB-MTAC database and assigns roles to the user based on registered role information. All the resources are accessed by the user through the RB-MTAC module that maintains the access control lists for resources.

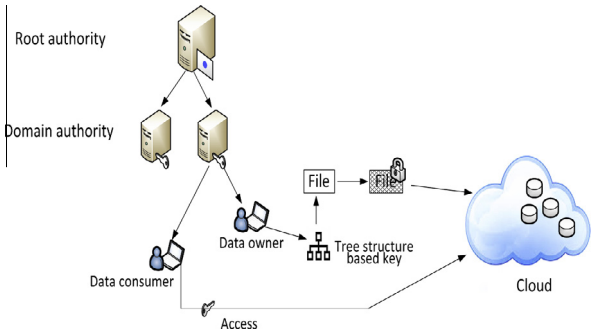

Fig. 5. System model for HASBE [108].

The authors in [19] presented a digital identity management system named Simple Privacy-preserving Identity-Management for Cloud Environment (SPICE). The SPICE exploits the concept of group signature and randomization for providing the anonymous authentication (to prove user authenticity without revealing identity), delegatable authentication, unlinkability (CSPs are unable to link the transactions of the same user), accountability, and user centric access control. Moreover, the SPICE provides the aforementioned properties with only a single registration. The user registers with a trusted party called the registrar and obtains a single credential for all the services provided by the CSP. The user generates an authentication certificate from the obtained credentials. As different CSPs require different attributes to authenticate, the user can generate different version of authentication certificate from the same credential. The group signatures are used over the certificates for authentication. The SPICE extends the Waters signature [112] and Groth-Sahai proof system [34] for group signature. The group signature authenticates the user by ensuring that the signature is from a valid user of the group with the need of the identity. Randomization is applied to the signatures for providing unlinkability. Randomization is also used to hide the attributes that are not required by any particular CSP.

Dhungana et al. [23] presented an identity management framework for the cloud networking infrastructure that is centered on User Managed Access (UMA) protocol. The infrastructure in the proposed scheme is seen as the Authorization Manager (AM). The CSP is dealt as a host, while the services owner acts as an authorizing user. The services are controlled by the AM. The requesting users' identities are also managed by the AM. Any request to the services is mediated by the AM that can grant or deny resource according to the access control policies. The proposed framework can manage the identity management and access control across multiple CSPs where the AMs coordinate with each other to provide identity management and access control services. The highlights of presented techniques are tabulated in Table 9.

4.2.5. Contractual and legal level solutions

There is a calamitous need of standardized procedures for SLA management with security perspective to assuage the security issues related to SLA and geographic legalities. The (web services agreement) ws-agreement [8] defines the syntax and semantics of publicizing the competences of the service providers and to create the template based agreements, and to monitor the agreement acquiescence. The ws-agreement mainly captures the agreement based on quality of service. The security constraints are not semantically netted for risk quantification in ws-agreement. Hale and Gamble [36] extended the ws-agreement to propose a framework, SecAgreement that articulates the security parameters and services for provision in the SLA. The SecAgreement extends the template of the ws-agreement to incorporate security constraints and metrics into the terms of SLA. The extended template also integrates the elements that quantify the risks of using specific cloud services. The authors in [36] also proposed an algorithm that sorts the services based on the risk they incur for customer. Based on the extensions in ws-agreement the users can quantify the risk of using the services of any CSP and opt for the cloud services that fulfill the security requirements.

In a cloud environment, the user assets are exposed to extreme risk in case of violation of security SLA or cancelation of any of the security services. The authors in [37] proposed a method react to the SLA violations (pertaining to the security) or service cancelation to reduce the security risks in post violation/cancelation environment. The methodology is centered on an algorithm that performs risk-aware renegotiation. The algorithm utilizes the risk weighted services of [36] to determine a service with the minimum risk that fulfills the organizational need. The algorithm performs the renegotiation and scrutinizes the obtainable services at runtime as a replacement to the canceled or problematic service. The algorithm updates the risk evaluation according to the changes in the SLA. The algorithm is capable of negotiating cloud federations to lower the risk and ensure optimal fulfillment of customer's security needs.

Rak et al. [79] delineated the SPECS, an architecture to provide the SLA-based security as a service. The proposed architecture divides and focuses on three stages of the SLA life cycle, namely: (a) negotiation, (b) enforcement, and (c) monitoring. The employed approach includes security parameters in the SLA to let the end user judge the security offerings and requirements. The attachment of security parameters with the SLA also makes it mandatory for the CSP to provide explicit security. The SPECS articulates the architecture only and makes use of established work to carry out the phases of the SLA life cycle. For example, the SPEC recommends the use of ws-agreement or SecAgreement during negotiation phase. During the enforcement phase, the SPEC recommends the enforcement either by activating parameters at system startup time or by monitoring and reporting. Likewise, for monitoring the m-OSAIC-based framework can be used in monitoring phase.

Ref. [38] built a compliance vocabulary and used ontologies to automate the process of negotiation and selection of better security parameters for the SLA. The vocabulary is populated with the set of SLA security terms and the associated security controls that fulfill the corresponding security requirements. The SLA security terms are selected from various standard documents, for instance, NIST, the common criteria, the CSA, etc. The vocabulary is represented as an XML schema. The vocabulary allows the organizations to compare the security services of different CSPs at a glance. The scheme in [38] also built an ontology that is used to provide automated selection and negotiation. The ontologies used the concept of service matchmaking to differentiate between different offerings. The scheme is compatible with current SLA frame works, for instance, ws-agreement. The presented research in context of legal and contractual issues is summarized in Table 10.

The vendors having data centers at various geographic locations usually migrate the user data and application between locations. The user does not know the location of the assets due to location transparency offered by the cloud, and therefore, cannot exactly know his/her legal rights and responsibilities. Moreover, there might be some locations that are unwanted by the user. Although nothing can be done by the user or the CSP about the laws of the land but user can be given the option during SLA negotiation to mark places that he does not want to migrate his/her assets. This way compliance with the laws

Table 9
Identity management and access control strategies comparison.

Work	Proposed scheme	Basic theory	Security features	Scalability
[108]	HASBE, access control scheme for cloud	• Attribute set based encryption • Trust hierarchy	• Access control for cloud storage • User revocation • Re-encryption • Privacy	High
[86]	Decentralized access control for cloud storage	• Bilinear pairing • Attribute based encryption • Attribute based signature	• User authentication • Access control for cloud storage	Medium
[122]	Role based access control scheme	Role based access control	Access control for cloud resources	Low
[19]	SPICE, identity management framework	• Anonymous and delegatable Authentication • Unlinkability • Accountability • Access control	• Group signatures • Randomization	High
[23]	Identity management framework	User managed access protocol	• Identity management • Authentication • Access control	Low

Table 10
Comparison of techniques countering contractual and legal issues in the cloud.

Work	Proposed scheme	Basic theory	Negotiation	Enforcement	Monitoring
[36]	SecAgreement, security risk calculation at cloud	• Embedding security parameters into SLA • ws-agreement • Risk quantification	✓	✗	✗
[37]	A framework for reacting to change in security environment at runtime	• Re-negotiation • Risk quantification • Matchmaking	✓	✓	✓
[79]	SPECS, SLA-based approach to security as a service	• Embedding security • parameters into SL Matchmaking	✓	✓	✓
[38]	A solution for embedding security controls in cloud SLA	• Compliance • Vocabulary • Ontologies • Matchmaking	✓	✗	✗

can be managed in more effective manner. Moreover, the users and the CSP must have mutual understanding about the roles and responsibilities of each other. Similar recommendations are made by the CSA to deal with the legal issues regarding cloud computing.

5. Security issues in MCC

A ubiquitous access to the cloud computing allows the mobile devices to connect and use the traditional cloud computing services. However, unlike the normal computing machines, the mobile devices are resource constrained [55]. The precincts of low processing power, less storage capacity, limited energy, and capricious internet connectivity does not allow compute and storage mandating applications to run on mobile devices [29]. The aforementioned limitations served as motivation for a new computing paradigm called MCC that enhances the abilities of mobile devices by moving the storage and compute intensive tasks of mobile to the cloud [56,82]. The mobile devices can now execute heavy compute and storage intensive processes by using the computation and storage services of the cloud. The MCC paradigm enables the users to access and manage their applications and data through the mobile device without the need to move to traditional computing machines [81]. However, like the traditional cloud computing, the MCC also faces the security issues that in fact prove to be a hurdle in its widespread adoption. This section provides a brief discussion on the security issues having roots in the MCC paradigm and general approach to deal them.

The MCC has its foundations in the traditional cloud computing, therefore, all the security issues discussed in Section 3 become inherited to the MCC. However, resource scarcity in the mobile devices does not permit the adoption of security solutions proposed for the regular cloud. It is noteworthy that the security solutions that are to be executed at the cloud end will remain the same. The security solutions at the client end (mobile device) need lighter versions that mobile devices are capable to run [54]. The major security issues in the MCC are: (a) mobile application security, (b) user privacy, (c) authentication, and (d) data security.

5.1. Mobile application security

Like traditional computing devices, the mobile devices are also prone to vulnerabilities of malicious code. The malicious code can be in the form of Trojan horse, virus, and worm and can cause the compromise of mobile application running at the mobile device. The traditional security software like antivirus and IDS are not possible to run continuously on the mobile device [41]. The compromised security application or the device may result in compromised identity as well [54]. The basic concept of offloading computation can also be used to run heavy security programs on the cloud that provide malicious code and intrusion detection on the mobile device [124].

5.2. User privacy

The mobile device can be the source of user location leakage especially due to location based services [63,81]. The aforesaid is the serious privacy issue and leads to even worse situation if a foe knows the user whereabouts [41]. The concept of location cloaking can be used to preserve user location privacy by concealing the user exact geographic position [49].

5.3. Authentication

Authentication is another issue on resource constrained mobile devices. With limited abilities of text input, passwords are usually used for authentication purposes in the MCC that can be vulnerable to theft over time [41]. Dynamic credential generation can be used for secure authentication. The credential generation can be offloaded to a trusted third party due to low processing power of the mobile device [54].

5.4. Data security

Due to low processing power of mobile devices, computation intensive encryption algorithms with large keys are not feasible to be run at the mobile device [53]. The MCC requires the form of encryption that requires least storage, processing and communication overhead [95]. Alternatively, the compute intensive tasks of encryption/decryption can be moved to trusted third party for securing the user data [53].

6. Discussion and open issues

The discussion on the security issues presented in the preceding sections elaborates that the cloud not only retains the orthodox security concerns but also entails the novel issues arising due to the use of new technologies and practices. The issues of web services and applications, communication and network, data privacy, etc. are the conventional issues that were present in the respective technologies even before the appearance of the cloud computing paradigm. The issues that arise due to virtualization, multi tenancy, and shared resource pool are novel issues. Moreover, certain orthodox issues become even more sensitive and critical when dealt in the cloud environment. For example, data security becomes more critical and difficult to deal with because of the absence of administrative control of the data owner. The colocation of various organizations data and applications adds more to the severity.

Section 4 has also witnessed that academia and research community is actively pursuing the security issues and several security solutions have been developed. At the communication level, the physical network infrastructure retains more of the conventional issues and solutions. However, the virtual network needs more attention. Although virtual devices have been proposed to secure the virtual network, a comprehensive strategy to monitor the traffic on the virtual network is needed to avoid malicious flow of information. The packet rewriting seems to be an effective approach in this regard. Nevertheless, a stringent methodology is required for traffic monitoring that creates a balance between privacy and monitoring. With regards to traffic on virtual network, the privacy and monitoring become contradicting requirements. The VM image protection techniques also need to be comprehensive in nature for dealing with all required aspects of VM image life cycle. Table 3 indicates that none of the presented technique fulfills all the tabulated security requirements. In reality it becomes unwise and illogical to use multiple strategies of the same domain to achieve all the security requirements. The same can be observed from other presented domains. It can also be observed that trusted computing can form a good basis of providing secure and trusted platforms because of the fact that it secures the platform right from the boot time and keep on monitoring the states periodically. Moreover, the temper proof key management makes trusted computing a good candidate for providing a comprehensive security solution in cloud computing.

Despite of intensive research efforts by the research community, there still are open issues that need to be addressed for providing a secure cloud environment. The first and the foremost need is to develop a comprehensive and integrated security solution that encompasses most of the major security requirements in the cloud environment. The research activities mostly focus on the specific issue and try to resolve that issue or in most encouraging scenario few related issues may be the target of the researchers. The specific issue scenario results in development of multiple solutions catering various security needs. In reality it is not wise and feasible to deploy the number of security tools equivalent or near to the number of security requirements. Deployment and configuration of large number of security solutions itself may be risky. A more integrated

solution will result in easy management of the security tool. At the least level, there is a need to harmonize different security solutions to produce the desired security level.

Shared pool of resources creates a need of a trustworthy access control system that can avoid the unauthorized access to the resources. The dynamism of the resources and heterogeneity of the services makes access control system to be more complex. Likewise, identity management is also a key issue in the cloud computing paradigm. The process of mapping the organizational identities to the cloud and the time it takes to translate the changes of the identities into the cloud is a crucial factor affecting the security in general and access control in particular. Moreover, future efforts to integrate assurance and auditing tool to ensure policy acquiescence among different involved entities are direly needed.

Multi tenancy being an essential characteristics of cloud computing is used to optimize resource utilization. However, multi tenancy also poses threats to the cloud computing system. Security and privacy for the multi tenancy is one of the grave challenge for the cloud computing. Research endeavors in this respect to find the solutions for multi tenancy security issues is highly desirable. Currently, there exists little work in solving multi tenancy issues.

The issues of SLA and incorporation of legal aspects into the SLA are still unsettled. The issue of auditing, whether service level is met as was promised in SLA or not, needs to be explored. Run time assurance mechanism to ensure that services are delivered as per requirement is also an important open research area. The current audit, based on the statistics provided by the CSP itself might not be a satisfactory option for many. Moreover, the pricing of the service usage is also totally dependent on the CSP calculations.

More focus is required to ensure the privacy during computations. Not all the operations can be performed over the data in encrypted form. Most of the operations require the plain form of data during computations. The memory locations within the processors and outside used for storing data temporarily may be the target of attack. Therefore, a broad framework that ensures privacy while performing computations is the need for security. Moreover, the data recovery vulnerability must be focused in the same perspective.

The customers due to many reasons may want to migrate the digital assets to some other cloud. However, migration to a different cloud is not an easy task. There is a need of standardized formats and protocols that can help the customers to migrate their data/applications to a different cloud.

The cloud computing also needs security against insider threats. In this context identification of indicators for insider attacks in the cloud environment is an open area of research. Although there are many such indicators for conventional system and they are still applicable to the cloud environment, the identification of cloud based indicators of insider threats will increase the potential of securing the cloud systems. Similarly, the differentiation between a normal and malicious user within the cloud is another area of possible research. The insider attacks can be avoided to an extent by having definite criteria of judging a normal and malicious behavior.

Lastly, it is worthy to mention that although the security solutions provide advantage to both the users and CSPs, they also introduce computational and cost overhead. From the user perspective, the overhead can potentially over throw the economic and computational benefits provided by the cloud. Quantifying the trade-offs between the security overheads and cloud advantages is another important area. Moreover, there is a need to find security solutions that create a balance between the security requirements and performance. The work in these areas will greatly help the users to perform quantitative and technical analysis and make better decisions before shifting to the cloud.

7. Conclusions

Despite several advantages offered by the cloud computing, it also fosters security concerns that hamper the fast rate adoption of the cloud computing. All of the users whether individual or organization should be well aware of the security threats existing in the cloud. Comprehending the security threats and counter measures will help organizations to carry out the cost benefit analysis and will urge them to shift to the cloud. As the cloud computing utilizes many traditional along with novel technologies, it possesses conventional as well as unique security issues. Virtualization and multi-tenancy permits various users (possibly from different origins) to utilize same physical resource. The aforesaid technologies generate cloud specific security threats that need to be understood and dealt keeping in view novel characteristics of cloud. Likewise, virtual networks raise some unique security concerns in addition to the concerns faced by conventional physical networks. The absence of impeccable isolation between virtual machines necessitates the development of specific methodologies capable of delineations that can prove to be equivalent of physical isolation. The geographical spread of cloud computing introduces various legal issues pertaining to users' assets and the laws under which they are governed. The identity management and access control over the organization's digital resources also takes distinctive forms in cloud computing due to lack of administrative control of owner organization.

This survey presented the security issues that arise due to the shared, virtualized, and public nature of the cloud computing paradigm. Subsequently, the counter measures presented in the literature are presented. The tabulated analysis of the presented techniques highlighted the scope of security services provided by the reviewed techniques. Tabulated analysis will greatly help the readers to compare and analyze the pros and cons of the research endeavors. Due to increased use of smartphones and mobile devices, the MCC has also taken off. We briefly discuss the security concerns of the MCC. The discussion of the presented technique has led ways to highlight some open issues to motivate the research community and academia to focus on the subject.

Acknowledgements

This research was in part supported by a grant from the National Science Foundation, CNS 1229316.

References

[1] A. Abbas, S.U. Khan, A review on the state-of-the-art privacy preserving approaches in e-health clouds, IEEE J. Biomed. Health Inform. (2014), http://dx.doi.org/10.1109/JBHI.2014.2300846.
[2] A. Abbas, K. Bilal, L. Zhang, S.U. Khan, A cloud based health insurance plan recommendation system: a user centered approach, Future Gener. Comput. Syst. (2014), http://dx.doi.org/10.1016/j.future.2014.08.010.
[3] R. Agrawal, Legal issues in cloud computing, in: IndicThreads.com, Conference on Cloud Computing, 2011.
[4] K. Alhamazani, R. Ranjan, K. Mitra, F. Rabhi, S.U. Khan, A. Guabtni, V. Bhatnagar, An Overview of the Commercial Cloud Monitoring Tools: Research Dimensions, Design Issues, and State-of-the-Art, arXiv preprint arXiv:1312.6170, 2013.
[5] M. Ali, R. Dhamotharan, E. Khan, S.U. Khan, A.V. Vasilakos, K. Li, A.Y. Zomaya, SeDaSC: secure data sharing in clouds, IEEE Syst. J. (2015), http://dx.doi.org/10.1109/JSYST.2014.2379646.
[6] O.D. Alowolodu, B.K. Alese, A.O. Adetunmbi, O.S. Adewale, O.S. Ogundele, Elliptic curve cryptography for securing cloud computing applications, Int. J. Comput. Appl. 66 (2013).
[7] M.R. Anala, J. Shetty, G. Shobha, A framework for secure live migration of virtual machines, in: IEEE International Conference on Advances in Computing, Communications and Informatics, 2013, pp. 243–248.
[8] A. Andrieux, K. Czajkowski, A. Dan, K. Keahey, H. Ludwig, T. Nakata, J. Pruyne, J. Rofrano, S. Tuecke, M. Xu, Web services agreement specification (WS-agreement), <http://www.ogf.org/documents/GFD.107.pdf> (accessed 26.05.14).
[9] M. Aslam, C. Gehrmann, M. Bjorkman, Security and trust preserving VM migrations in public clouds, in: IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), 2012, pp. 869–876.
[10] M. Balduzzi, J. Zaddach, D. Balzarotti, E. Kirda, S. Loureiro, A security analysis of amazon's elastic compute cloud service, in: Proceedings of the 27th Annual ACM Symposium on Applied Computing, 2012, pp. 1427–1434.
[11] R. Bhadauria, R. Borgohain, A. Biswas, S. Sanyal, Secure Authentication of Cloud Data Mining API, arXiv preprint arXiv:1308.0824, 2013.
[12] K. Bilal, S.U.R. Malik, S.U. Khan, A.Y. Zomaya, Trends and challenges in cloud data centers, IEEE Cloud Comput. Mag. 1 (1) (2014) 10–20.
[13] R. Bobba, H. Khurana, M. Prabhakaran, Attribute-sets: a practically motivated enhancement to attribute-based encryption, in: Computer Security ESORICS, Springer, Berlin, Heidelberg, 2009, pp. 587–604.
[14] S. Carlin, K. Curran, Cloud computing security, Int. J. Ambient Comput. Intell. 3 (1) (2011) 14–19.
[15] R. Chandramouli, M. Iorga, S. Chokhani, Cryptographic key management issues and challenges in cloud services, in: Secure Cloud Computing, Springer, New York, 2014, pp. 1–30. doi: 10.1007/978-1-4614-9278-8_1.
[16] S. Chaisiri, B. Lee, D. Niyato, Optimization of resource provisioning cost in cloud computing, IEEE Trans. Services Comput. 5 (2) (2012) 164–177.
[17] D. Chen, H. Zhao, Data security and privacy protection issues in cloud computing, in: International Conference on Computer Science and Electronics Engineering (ICCSEE, IEEE), vol. 1, 2012, pp. 647–651.
[18] J. Che, Y. Duan, T. Zhang, J. Fan, Study on the security models and strategies of cloud computing, Proc. Eng. 23 (2011) 586–593.
[19] S.M.S. Chow, Y. He, L.C.K. Hui, S.M. Yiu, Spicesimple privacy-preserving identity-management for cloud environment, in: Applied Cryptography and Network Security, Springer, Berlin, Heidelberg, 2012, pp. 526–543.
[20] A. Corradi, M. Fanelli, L. Foschini, VM consolidation: a real case based on openstack cloud, Future Gener. Comput. Syst. 32 (2014) 118–127.
[21] Cloud security alliance, security guidelines for critical areas of focus in cloud computing v3.0, 2011.
[22] B. Danev, R.J. Masti, G.O. Karame, S. Capkun, Enabling secure VM-vTPM migration in private clouds, in: Proceedings of the ACM 27th Annual Computer Security Applications Conference, 2011, pp. 187–196.
[23] R.D. Dhungana, A. Mohammad, A. Sharma, I. Schoen, Identity management framework for cloud networking infrastructure, in: IEEE International Conference on Innovations in Information Technology (IIT), 2013, pp. 13–17.
[24] H.T. Dinh, C. Lee, D. Niyato, P. Wang, A survey of mobile cloud computing: architecture, applications, and approaches, Wireless Commun. Mobile Comput. 13 (18) (2013) 1587–1611.
[25] Q. Duan, Y. Yan, A.V. Vasilakos, A survey on service-oriented network virtualization toward convergence of networking and cloud computing, IEEE Trans. Netw. Service Manage. 9 (4) (2012) 373–392.
[26] A. Eghtesadi, Y. Jarraya, M. Debbabi, M. Pourzandi, Preservation of security configurations in the cloud, in: IEEE International Conference on Cloud Engineering (IC2E), 2014, pp. 17–26.
[27] K. Fan, D. Mao, Z. Lu, J. Wu, OPS: offine patching scheme for the images management in a secure cloud environment, in: IEEE International Conference on Services Computing (SCC), 2013, pp. 587–594.
[28] D. AB. Fernandes, L. FB. Soares, J.V. Gomes, M.M. Freire, P. RM Inácio, Security issues in cloud environments: a survey, Int. J. Inform. Sec. 13 (2) (2014) 113–170.
[29] N. Fernando, S.W. Loke, W. Rahayu, Mobile cloud computing: a survey, Future Gener. Comput. Syst. 29 (1) (2013) 84–106.
[30] M. Ficco, M. Rak, Stealthy denial of service strategy in cloud computing, IEEE Trans. Cloud Comput. (2014), http://dx.doi.org/10.1109/TCC.2014.2325045.
[31] Y. Fu, Z. Lin, Exterior: using a dual-vm based external shell for guest-os introspection, configuration, and recovery, in: Proceedings of the 9th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments, 2013, pp. 97–110.
[32] M.I. Gofman, R. Luo, P. Yang, K. Gopalan, Sparc: a security and privacy aware virtual machine check pointing mechanism, in: Proceedings of the 10th Annual ACM Workshop on Privacy in the Electronic Society, 2011, pp. 115–124.
[33] N. Gonzalez, C. Miers, F. Redgolo, M. Simplcio, T. Carvalho, M. Nslund, M. Pourzandi, A quantitative analysis of current security concerns and solutions for cloud computing, J. Cloud Comput. 1 (1) (2012) 1–18.
[34] J. Groth, Amit Sahai, Efficient non-interactive proof systems for bilinear groups, in: Advances in Cryptology EUROCRYPT, Springer, Berlin, Heidelberg, 2008, pp. 415–432.
[35] B. Guan, J. Wu, Y. Wang, S.U. Khan, CIVSched: a communication-aware inter-VM scheduling technique for decreased network latency between co-located VMs, IEEE Trans. Cloud Comput. 2 (3) (2014) 320–332.
[36] M.L. Hale, R. Gamble, Secagreement: advancing security risk calculations in cloud services, in: IEEE Eighth World Congress on Services (SERVICES), 2012, pp. 133–140.
[37] M.L. Hale, R. Gamble, Risk propagation of security SLAs in the cloud, in: IEEE Globecom Workshops (GC Wkshps), 2012, pp. 730–735.
[38] M.L. Hale, R. Gamble, Building a compliance vocabulary to embed security controls in cloud SLAs, in: IEEE Ninth World Congress on Services (SERVICES), 2013, pp. 118–125.
[39] K. Hashizume, D.G. Rosado, E. Fernndez-Medina, E.B. Fernandez, An analysis of security issues for cloud computing, J. Internet Services Appl. 4 (1) (2013) 1–13.
[40] B. Hay, K. Nance, M. Bishop, Storm clouds rising: security challenges for IaaS cloud computing, in: 44th Hawaii International Conference on System Sciences (HICSS), IEEE, 2011, pp. 1–7.
[41] T.D. Hoang, C. Lee, D. Niyato, P. Wang, A survey of mobile cloud computing: architecture, applications, and approaches, Wireless Commun. Mobile Comput. 13 (18) (2013) 1587–1611.

[42] X. He, T. Chomsiri, P. Nanda, Z. Tan, Improving cloud network security using the tree-rule firewall, Future Gener. Comput. Syst. 30 (2014) 116–126.
[43] Hewlett Packard, 5 cloud security concerns you must address, Business white paper by Hewlett Packard, No. 4AA3-8247ENW, rev. 1, 2012.
[44] Y. Hu, T. Li, P. Yang, K. Gopalan, An application-level approach for privacy-preserving virtual machine checkpointing, in: IEEE Sixth International Conference on Cloud Computing, 2013, pp. 59–66.
[45] M. Hussain, H. Abdulsalam, SECaaS: security as a service for cloud-based applications, in: ACM Proceedings of the Second Kuwait Conference on e-Services and e-Systems, 2011, p. 8.
[46] A.S. Ibrahim, J. Hamlyn-Harris, J. Grundy, M. Almorsy, Cloudsec: a security monitoring appliance for virtual machines in the iaas cloud model, in: IEEE 5th International Conference on Network and System Security (NSS), 2011, pp. 113–120.
[47] W.A. Jansen, Cloud hooks: Security and privacy issues in cloud computing, in: 44th Hawaii International Conference on System Sciences (HICSS), 2011, pp. 1–10.
[48] D. Jeswani, A. Verma, P. Jayachandran, K. Bhattacharya, ImageElves: rapid and reliable system updates in the cloud, in: IEEE 33rd International Conference on Distributed Computing Systems (ICDCS), 2013, pp. 390–399.
[49] P. Juncheng, D. Huimin, S. Yinghui, L. Dong, Potential attacks against k-anonymity on LBS and solutions for defending the attacks, in: Advanced in Computer Science and its Applications, Springer, Berlin, Heidelberg, 2014, pp. 877–883.
[50] T. Jung, X. Li, Z. Wan, M. Wan, Control cloud data access privilege and anonymity with fully anonymous attribute based encryption, IEEE Trans. Inform. Forensics Sec. 10 (1) (2014) 190–199.
[51] M. Kazim, R. Masood, M.A. Shibli, Securing the virtual machine images in cloud computing, in: Proceedings of the ACM 6th International Conference on Security of Info and Networks, 2013, pp. 425–428.
[52] A.N. Khan, M.L.M. Kiah, M. Ali, S.A. Madani, S. Shamshirband, BSS: block-based sharing scheme for secure data storage services in mobile cloud environment, J. Supercomput. 70 (2) (2014) 946–976.
[53] A.N. Khan, M.L.M. Kiah, S.A. Madani, M. Ali, S. Shamshirband, Incremental proxy re-encryption scheme for mobile cloud computing environment, J. Supercomput. 68 (2) (2014) 624–651.
[54] A.N. Khan, M.L.M. Kiah, S.A. Madani, M. Ali, Enhanced dynamic credential generation scheme for protection of user identity in mobile-cloud computing, J. Supercomput. 66 (3) (2013) 1687–1706.
[55] A.N. Khan, M.L.M. Kiah, S.U. Khan, S.A. Madani, Towards secure mobile cloud computing: a survey, Future Gener. Comput. Syst. 29 (5) (2013) 1278–1299.
[56] A.R. Khan, M. Othman, S.A. Madani, S.U. Khan, A survey of mobile cloud computing application models, IEEE Commun. Surv. Tutorials 16 (1) (2014) 393–413.
[57] R. Latif, H. Abbas, S. Assar, Q. Ali, Cloud computing risk assessment: a systematic literature review, in: Future Information Technology, Springer, Berlin, Heidelberg, 2014, pp. 285–295.
[58] C. Li, A. Raghunathan, N.K. Jha, A trusted virtual machine in an untrusted management environment, IEEE Trans. Serv. Comput. 5 (4) (2012) 472–483.
[59] J. Li, B. Li, T. Wo, C. Hu, J. Huai, L. Liu, K.P. Lam, Cyber-guarder: a virtualization security assurance architecture for green cloud computing, Future Gener. Comput. Syst. 28 (2) (2012) 379–390.
[60] H.Y. Lin, W.G. Tzeng, A secure erasure code-based cloud storage system with secure data forwarding, IEEE Trans. Parallel Distrib. Syst. 23 (6) (2012) 995–1003.
[61] B. Liu, E. Blasch, Y. Chen, A.J. Aved, A. Hadiks, D. Shen, G. Chen, Information fusion in a cloud computing era: a systems-level perspective, IEEE Aerospace Electron. Syst. Mag. 29 (10) (2014) 16–24.
[62] B. Liu, J. Bi, A. Vasilakos, Towards incentivizing anti-spoofing deployment, IEEE Trans. Inform. Forensics Sec. 9 (3) (2014) 436–450.
[63] F. Liu, P. Shu, H. Jin, L. Ding, J. Yu, D. Niu, B. Li, Gearing resource-poor mobile devices with powerful clouds: architectures, challenges, and applications, IEEE Wireless Commun. 20 (3) (2013).
[64] Q. Liu, G. Wang, J. Wu, Time-based proxy re-encryption scheme for secure data sharing in a cloud environment, Inform. Sci. 258 (2014) 355–370.
[65] W. Liu, S. Peng, W. Du, W. Wang, G.S. Zeng, Security-aware intermediate data placement strategy in scientific cloud workflows, Knowl. Inform. Syst. 41 (2) (2014) 423–447.
[66] W. Lloyd, S. Pallickara, O. David, J. Lyon, M. Arabi, K. Rojas, Performance implications of multi-tier application deployments on Infrastructure-as-a-Service clouds: towards performance modeling, Future Gener. Comput. Syst. 29 (5) (2013) 1254–1264.
[67] F. Lombardi, R.D. Pietro, Secure virtualization for cloud computing, J. Netw. Comput. Appl. 34 (4) (2011) 1113–1122.
[68] S. Marston, Z. Li, S. Bandyopadhyay, J. Zhang, A. Ghalsasi, Cloud computing the business perspective, Decis. Support Syst. 51 (1) (2011) 176–189.
[69] P. Mell, T. Grance, The NIST definition of cloud computing (draft), NIST Special Publ. 800 (145) (2011) 7.
[70] M. Menzel, R. Ranjan, L. Wang, S.U. Khan, J. Chen, CloudGenius: a hybrid decision support method for automating the migration of web application clusters to public clouds, IEEE Trans. Comput. (2014), http://dx.doi.org/10.1109/TC.2014.2317188.
[71] C. Modi, D. Patel, B. Borisaniya, A. Patel, M. Rajarajan, A survey on security issues and solutions at different layers of Cloud computing, J. Supercomput. 63 (2) (2013) 561–592.
[72] H. Moraes, R. Nunes, D. Guedes, DCPortalsNg: efficient isolation of tenant networks in virtualized datacenters, in: Thirteenth International Conference on Networks, 2014, pp. 230–235.
[73] S.H. Na, E.N. Huh, A broker-based cooperative security-SLA evaluation methodology for personal cloud computing, Sec. Commun. Netw. (2014), http://dx.doi.org/10.1002/sec.1086.
[74] Y. Neng-Hai, Z. Hao, J. Xu, W. Zhang, C. Zhang, Review of cloud computing security, Acta Electron. Sinica 41 (2) (2013) 371–381.
[75] Open Web Application Security Project Top 10-2013, The ten most critical Web application security risks, <https://www.owasp.org/index.php/Top10]OWASP> Top 10 for 2013 (accessed 08.04.14).
[76] W. Pan, Y. Zhang, M. Yu, J. Jing, Improving virtualization security by splitting hypervisor into smaller components, in: Data and Applications Security and Privacy XXVI, Springer, Berlin, Heidelberg, 2012, pp. 298–313.
[77] R.D. Pietro, F. Lombardi, M. Signorini, CloRExPa: cloud resilience via execution path analysis, Future Gener. Comput. Syst. 32 (2014) 168–179.
[78] A. Prasad, S. Rao, A mechanism design approach to resource procurement in cloud computing, IEEE Trans. Comput. 63 (1) (2014) 17–30.
[79] M. Rak, N. Suri, J. Luna, D. Petcu, V. Casola, U. Villano, Security as a service using an SLA-based approach via SPECS, in: IEEE 5th International Conference on Cloud Computing Technology and Science (CloudCom), vol. 2, 2013, pp. 1–6.
[80] K.S. Rao, P.S. Thilagam, Heuristics based server consolidation with residual resource defragmentation in cloud data centers, Future Gener. Comput. Syst. (2014), http://dx.doi.org/10.1016/j.future.2014.09.009.
[81] M.R. Rahimi, J. Ren, C.H. Liu, A.V. Vasilakos, N. Venkatasubramanian, Mobile cloud computing: a survey, state of art and future directions, Mobile Netw. Appl. 19 (2) (2014) 133–143.
[82] M.R. Rahimi, N. Venkatasubramanian, S. Mehrotra, A.V. Vasilakos, MAPCloud: mobile applications on an elastic and scalable 2-tier cloud architecture, in: Proceedings of the 2012 IEEE/ACM Fifth International Conference on Utility and Cloud Computing, 2012, pp. 83–90.
[83] K. Ren, C. Wang, Q. Wang, Security challenges for the public cloud, IEEE Internet Comput. 16 (1) (2012) 69–73.
[84] T. Ristenpart, E. Tromer, H. Shacham, S. Savage, Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds, in: Proceedings of the 16th ACM Conference on Computer and Communications Security, 2009, pp. 199–212.
[85] C. Rong, S.T. Nguyen, M.G. Jaatun, Beyond lightning: a survey on security challenges in cloud computing, Comput. Electr. Eng. 39 (1) (2013) 47–54.
[86] S. Ruj, M. Stojmenovic, A. Nayak, Decentralized access control with anonymous authentication of data stored in clouds, IEEE Trans. Parallel Distrib. Syst. 25 (2) (2014) 384–394.
[87] M.D. Ryan, Cloud computing security: the scientific challenge, and a survey of solutions, J. Syst. Softw. 86 (09) (2013) 2263–2268.
[88] M. Sadiku, S. Musa, O. Momoh, Cloud computing: opportunities and challenges, IEEE Potentials 33 (1) (2014) 34–36.

[89] E. Schweitzer, Reconciliation of the cloud computing model with US federal electronic health record regulations, J. Am. Med. Inform. Assoc. 19 (2) (2012) 161–165.
[90] R. Schwarzkopf, M. Schmidt, C. Strack, S. Martin, B. Freisleben, Increasing virtual machine security in cloud environments, J. Cloud Comput. 1 (1) (2012) 1–12.
[91] S.K. Sah, S. Shakya, H. Dhungana, A security management for cloud based applications and services with diameter-AAA, in: IEEE International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), 2014, pp. 6–11.
[92] A. Sahai, B. Waters, Fuzzy identity-based encryption, in: Advances in Cryptology EUROCRYPT, 2005, pp. 457–473.
[93] K. Salah, J.M.A. Calero, S. Zeadally, S. Al-Mulla, M. Alzaabi, Using cloud computing to implement a security overlay network, IEEE Sec. Privacy 11 (1) (2013) 44–53.
[94] K. Sankar, S. Kannan, P. Jennifer, On-demand security architecture for cloud computing, Middle-East J. Sci. Res. 20 (2) (2014) 241–246.
[95] Z. Sanaei, S. Abolfazli, A. Gani, R. Buyya, Heterogeneity in mobile cloud computing: taxonomy and open challenges, IEEE Commun. Surveys Tutorials 16 (1) (2014) 369–392.
[96] L. FB Soares, D. AB Fernandes, J.V. Gomes, M.M. Freire, P. RM Inácio, Cloud security: state of the art, in: Security, Privacy and Trust in Cloud Systems, Springer, Berlin, Heidelberg, 2014, pp. 3–44.
[97] M.H. Song, Analysis of risks for virtualization technology, in: Applied Mechanics and Materials, vol. 539, 2014, pp. 374–377.
[98] S.K. Sood, A combined approach to ensure data security in cloud computing, J. Netw. Comput. Appl. 35 (6) (2012) 1831–1838.
[99] M. Sookhak, H. Talebian, E. Ahmed, A. Gani, M.K. Khan, A review on remote data auditing in single cloud server: taxonomy and open issues, J. Netw. Comput. Appl. 43 (2014) 121–141.
[100] A. Srivastava, H. Raj, J. Gi_n, P. England, Trusted VM snapshots in untrusted cloud infrastructures, in: Research in Attacks, Intrusions, and Defenses, Springer, Berlin, Heidelberg, 2012, pp. 1–21.
[101] S. Subashini, V. Kavitha, A survey on security issues in service delivery models of cloud computing, J. Netw. Comput. Appl. 34 (1) (2011) 1–11.
[102] J. Szefer, E. Keller, R.B. Lee, J. Rexford, Eliminating the hypervisor attack surface for a more secure cloud, in: Proceedings of the 18th ACM Conference on Computer and Communications Security, 2011, pp. 401–412.
[103] Y. Tang, P.P. Lee, J.C.S. Lui, R. Perlman, Secure overlay cloud storage with access control and assured deletion, IEEE Trans. Dependable Secure Comput. 9 (6) (2012) 903–916.
[104] Z. Tari, Security and privacy in cloud computing, IEEE Cloud Comput. 1 (1) (2014) 54–57.
[105] Z. Tavakoli, S. Meier, A. Vensmer, A framework for security context migration in a firewall secured virtual machine environment, in: Information and Communication Technologies, Springer, Berlin, Heidelberg, 2012, pp. 41–51.
[106] V. Varadharajan, U. Tupakula, Counteracting security attacks in virtual machines in the cloud using property based attestation, J. Network Comput. Appl. 40 (2014) 31–45.
[107] V. Vladimir, Cloud adoption issues: interoperability and security, in: Cloud Computing and Big Data, 2013, pp. 53–65.
[108] Z. Wan, J. Liu, R.H. Deng, HASBE: a hierarchical attribute-based solution for flexible and scalable access control in cloud computing, IEEE Trans. Inform. Forensics Sec. 7 (2) (2012) 743–754.
[109] X. Wan, X.F. Zhang, L. Chen, J.X. Zhu, An improved vTPM migration protocol based trusted channel, in: IEEE International Conference on Systems and Informatics, 2012, pp. 870–875.
[110] C. Wang, Q. Wang, K. Ren, N. Cao, W. Lou, Toward secure and dependable storage services in cloud computing, IEEE Trans. Services Comput. 5 (2) (2012) 220–232.
[111] Z. Wang, C. Wu, M. Grace, X. Jiang, Isolating commodity hosted hypervisors with hyperlock, in: Proceedings of the 7th ACM European Conference on Computer Systems, 2012, pp. 127–140.
[112] B. Waters, Efficient identity-based encryption without random oracles, in: Advances in Cryptology EUROCRYPT, Springer, Berlin, Heidelberg, 2005, pp. 114–127.
[113] J. Wei, X. Zhang, G. Ammons, V. Bala, P. Ning, Managing security of virtual machine images in a cloud environment, in: Proceedings of the 2009 ACM Workshop on Cloud Computing Security, 2009, pp. 91–96
[114] L. Wei, H. Zhu, Z. Cao, X. Dong, W. Jia, Y. Chen, A.V. Vasilakos, Security and privacy for storage and computation in cloud computing, Inform. Sci. 258 (2014) 371–386.
[115] C. Wu, Z. Wang, X. Jiang, Taming hosted hypervisors with (mostly) deprivileged execution, in: Proceedings of the Network and Distributed System Security Symposium (NDSS), San Diego, CA, 2013.
[116] H. Wu, Y. Ding, C. Winer, L. Yao, Network security for virtual machine in cloud computing, in: 5th International Conference on Computer Sciences and Convergence Information Technology, 2010, pp. 18–21.
[117] M.Y. Wu, T.H. Lee, Design and implementation of cloud API access control based on OAuth, in: IEEE TENCON Spring Conference, 2013, pp. 485–489.
[118] Y. Xia, Y. Liu, H. Chen, Architecture support for guest-transparent VM protection from untrusted hypervisor and physical attacks, in: IEEE 19th International Symposium on High Performance Computer Architecture, 2013, pp. 246–257.
[119] Y. Xia, Y. Liu, H. Chen, B. Zang, Defending against VM rollback attack, in: IEEE/IFIP 42nd International Conference on Dependable Systems and Networks Workshops, 2012, pp. 1–5.
[120] T. Xing, D. Huang, L. Xu, C. Chung, P. Khatkar, Snortflow: a openflow-based intrusion prevention system in cloud environment, in: IEEE Research and Educational Experiment Workshop, 2013, pp. 89–92.
[121] Z. Xiao, Y. Xiao, Security and privacy in cloud computing, IEEE Commun. Surveys Tutorials 15 (2) (2013) 843–859.
[122] S. Yang, P. Lai, J. Lin, Design role-based multi-tenancy access control scheme for cloud services, in: IEEE International Symposium on Biometrics and Security Technologies (ISBAST), 2013, pp. 273–279.
[123] Z. Yan, P. Zhang, A.V. Vasilakos, A survey on trust management for Internet of things, J. Netw. Comput. Appl. 42 (2014) 120–134.
[124] S. Yazji, P. Scheuermann, R.P. Dick, G. Trajcevski, R. Jin, Efficient location aware intrusion detection to protect mobile devices, Personal Ubiquitous Comput. 18 (1) (2014) 143–162.
[125] Y.A. Younis, M. Merabti, K. Kifayat, Secure cloud computing for critical infrastructure: a survey, Technical Report, Liverpool John Moores University, United Kingdom, Tech. Rep. ISBN: 978-1-902560-27-4, 2013.
[126] H. Yu, N. Powell, D. Stembridge, X. Yuan, Cloud computing and security challenges, in: Proceedings of the 50th Annual Southeast Regional Conference, ACM, 2012, pp. 298–302.
[127] M. Zalewski, Browser security handbook, Google Code, 2010. <https://code.google.com/p/browsersec/> (accessed 08.04.14).
[128] F. Zhang, H. Chen, Security-preserving live migration of virtual machines in the cloud, J. Netw. Syst. Manage. 21 (4) (2013) 562–587.
[129] F. Zhang, J. Chen, H. Chen, B. Zang, Cloudvisor: retrofitting protection of virtual machines in multi-tenant cloud with nested virtualization, in: Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles, 2011, pp. 203–216.
[130] F. Zhang, J. Wang, K. Sun, A. Stavrou, HyperCheck: a hardware-assisted integrity monitor, IEEE Trans. Dependable Sec. Comput. (2013), http://dx.doi.org/10.1109/TDSC.2013.53.