Bournemouth University

Faculty of Science and Technology

Department of Computing & Informatics

Submission Coursework Brief
Undergraduate

Unit: COMPUTER FUNDAMENTALS (COMP4029)		
Unit Contact: Tim Orman	**Credits:** 20	**Level:** 4

Assessment Title: CF Concept Mappping Assignment	
Assessment Number: 1 **of** 2	
Assessment Type: Individual	**Weighting:** 50%
Deadline: 22/10/2024 at 12:00 PM	**Submission Method:** Brightspace

Quality Assessor (QA): Jiankang Zhang	**Other Marker(s):** N/A

Can I use Generative AI tools?
Basic spelling and grammar correction tools are permitted.

The following originality requirements will apply to this assignment:

You are not allowed to use any Generative AI or other AI powered tools, such as ChatGPT, for this assessment. Any use of these tools for any part of this assessment would be considered an academic offence.

Task:

TASKS

Read and reflect on the paper:

1. Cloud computing security: A survey of service-based models Fatemeh Khoda Parast, Chandni Sindhav, Seema Nikam, Hadiseh Izadi Yekta, Kenneth B. Kent, Saqib Hakak (supplied).

2. Create a concept map that provides an overiew of the concepts, themes, issues etc. within the paper (100%).
    a. Concepts (40%)
    b. Relationships (40%)
    c. Overall readability and cohesion (20%)

DELIVERABLES

- A single PDF (.pdf) document containing the concept map.

This should be submitted by 12:30hrs on the due date.

RESOURCES

Lectures and labs will cover some of the knowledge and skills need to complete this assignment but additional resources which you might find useful are:

- Coggle - Simple Collaborative Mind Maps [WWW Document], n.d. URL https://coggle.it/ (accessed 9.13.22).
- Free Concept Map Maker with Online Templates | Adobe Express [WWW Document], n.d. URL https://www.adobe.com/express/create/chart/concept-map (accessed 9.13.22).
- Lucidchart [WWW Document], n.d. URL https://lucid.app/pricing/lucidchart?referer=https%3A%2F%2Fwww.lucidchart.com%2F#/pricing/chart(accessed 9.13.22).
- Mind mapping, concept mapping, outlining and Gantt Charts [WWW Document], n.d. . Mindomo. URL https://www.mindomo.com/ (accessed 9.13.22).

Intended Learning Outcomes (ILOs)

This unit assesses your ability to:

1. 1) Understand the basics of computer architectures.
2. Understand and describe the basic elements of operating systems, programming languages, database applications, peripheral devices, networking and the internet.
3. Understand the principles of securing computing systems.
4. Understand societal, legal and ethical issues of computing systems.

Submission Format:

A single page document in **PDF format** containing a detailed concept map (the page size does not need to be restricted to A4). **Other file formats will not be marked!**

How will this be assessed?

Indicative Marking Criteria					
Criteria/Award	1st	2:i	2:ii	3rd	Fail
Concepts (40%)	A highly detailed set of concepts.	A comprehensive set of concepts identified	A fair amount of concepts identified at higher and lower levels	Most high-level concepts identified, includes some lower-level ones	Minimal concepts identified
Relationships (40%)	Adds more subtle relationships	A comprehensive set of appropriate relationships included	A fair number of appropriate relationships included	A number of Relationship labels inappropriate or missing	Minimal relationships added
Readability & Cohesion (20%)	Adds exceptional clarity and aesthetic.	Clear to read and an overall cohesion to the map.	Reasonable easy to comprehend with a few issues	Mostly readable but lacking overall cohesion	Poorly laid out, Difficult to follow

Questions about the assessment:

Questions about this assignment should be directed primarily at the Unit Leader (Tim) or your lab/seminar leader. The best time to ask questions is during the lab/seminar sessions (as you are assured of our full attention). Other than this please contact us via MS Teams (preferably) or email. Please remember though that electronic communications can have a 3 working day turnaround.

In-Year Retrieval (IYR)

This assessment is eligible for an "in-year retrieval" (IYR) opportunity to enable those who fail this assessment in semester one to rework their initial submission within 15 term-time days, rather than having to wait for the reassessment period to "make good" on the failure. This additional attempt will be capped at the pass mark.

You can find further information about IYR in the FAQs here https://www.bournemouth.ac.uk/students/help-advice/looking-support/supporting-your-learning/support-assessments and in this guidance flowchart.

In-year retrieval is not compulsory; students "opt-in" by resubmission. Exceptional Circumstance and Extensions will not apply for in-year retrieval.

The following assessment scenarios (affecting the first submission) will not be eligible for in-year retrieval:

i. non-submission,
ii. assessments submitted more than 72 hours after the original deadline,
iii. extensions due to exceptional circumstances,
iv. submissions subject to an academic offence investigation.

Academic Integrity

The work you submit must be your own. Any attempt to gain an unfair advantage in your assessment by **cheating**, deception or fraud is considered an academic offence. The 'Assessment help and support' section of the unit (found under 'Assessment' in the content area) provides more guidance on avoiding academic offences, including **any guidance on what will or will not be considered an academic offence in this specific assessment**

Help and support

The 'Assessment help and support' section of the unit (found under 'Assessment' in the content area) provides information and guidance, including specific information on support for this assessment. It provides help with our policies on deadline extensions and information on support available in the university, including academic skills support and additional learning support for students with disabilities.

Disclaimer

The information provided in this assignment brief is correct at time of publication. In the unlikely event that any changes are deemed necessary, they will be communicated clearly via e-mail and via the VLE and a new version of this coursework brief will be circulated.

Date Issued: 30/09/2024

Computers & Security 114 (2022) 102580

Contents lists available at ScienceDirect

Computers & Security

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

TC 11 Briefing Papers

Cloud computing security: A survey of service-based models

Fatemeh Khoda Parast*, Chandni Sindhav, Seema Nikam, Hadiseh Izadi Yekta, Kenneth B. Kent, Saqib Hakak

Faculty of Computer Science, University of New Brunswick, Canada

ARTICLE INFO

Article history:
Received 21 July 2021
Revised 23 October 2021
Accepted 5 December 2021
Available online 17 December 2021

Keywords:
Security
Cloud computing
Service-based cloud computing
IaaS
PaaS
SaaS

ABSTRACT

Cloud computing has recently attracted significant attention due to its economical and high-quality services. In the last decade, cloud services have inevitably entangled with businesses' and individuals' daily lives through products and services. On-demand, pay-per-use characteristics encourage corporations to outsource part of their businesses to accelerate their services and multiply value. The latest market tendency toward migration to cloud environments, started in 2019, indicates a flourishing trend in the next few years. Despite the numerous benefits of the cloud computing model for businesses or individuals, security issues still have been stated as the top cloud challenge in 2020. Although various factors affect security, technologies enabling cloud computing such as virtualization and multitenancy, in addition to on-demand characteristics, initiate new security entrances for malevolent activities. In this study, we surveyed service-based cloud computing security issues to establish the current state of the field. The main contribution of this paper is to analyze the state of cloud security in the last decade and provide a unified taxonomy of security issues over the three-layer model, i.e., IaaS, PaaS, and SaaS.

1. Introduction

Cloud computing has received notable attention, providing *flexibility, scalability, reliability, sustainability*, and *affordability* (Varghese and Buyya, 2017; Vasiljeva et al., 2017). The pillar concept of the cloud, pay-per-use, has attracted not only individuals but businesses to benefit from the new approach to make profits (Becker et al., 2017; Bohn et al., 2011; Weinman, 2018). According to a survey conducted in 2020 among 750 global cloud professionals (Flexera, 2020), due to the COVID-19 impact, organizations will spend 47% more on cloud services in 2021 alone. Top growing cloud service consumers, i.e., IoT, machine learning/AI, data warehouse, and serverless markets will grow 47.2% on average (Bahrami and Singhal, 2015). Although tech giants such as *Google, Microsoft*, and *IBM* compete to provide the best solutions to users, the field still requires more research on security solutions (Kaur et al., 2018; Kumar and Goyal, 2019; Zissis and Lekkas, 2012).

Despite the obvious benefits of cloud computing, the complexity of the model and shared technologies have given rise to security concerns (Flexera, 2020; Rajaraman, 2014). The diversity of involved elements in the cloud paradigm, i.e., network, architecture, APIs, and hardware, increases the intricacy of security issues (Pancholi and Patel, 2016). As a result, a cloud provider or client would encounter security vulnerabilities caused by a different combination of a cloud configuration (Ghobaei-Arani et al., 2018). The *National Institute of Standards and Technology (NIST)* has introduced the service-based model as a standard for cloud computing. This model includes *Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS)*, and *Software-as-a-Service (SaaS)* defining all IT sharable resources such as software, hardware, or network (Huang et al., 2015; Modi et al., 2013).

The **Multitenancy, Elasticity**, and **Deployment** model raises important security implications Hwang et al. (2016). Multitenancy allows Cloud Service Providers (CSP)s to share resources among numerous customers. Through this feature, several users coexist in a single instance of a physical device at the same time, increasing the probability of Virtual Machine (VM) or Hypervisor (HV) attacks. Elasticity provides a scaling up/down capability for increasing/decreasing resources (Diaby and Rad, 2017; e Rubab et al., 2020). Once a user requires fewer resources, those could be allocated to another customer as needed. In such cases, the previous user's data might still exist in the allocated location, which opens up security issues (Almorsy et al., 2016; George Amalarethinam and Rajakumari, 2019). In addition to the cloud computing enabling technology, the availability of resources creates a perfect environment for intruders to apply attacks to other systems. At-

* Corresponding author.
E-mail addresses: fkhoda@unb.ca (F. Khoda Parast), csindhav@unb.ca (C. Sindhav), snikam@unb.ca (S. Nikam), hizadi@unb.ca (H. Izadi Yekta), ken@unb.ca (K.B. Kent), shakak@unb.ca (S. Hakak).

https://doi.org/10.1016/j.cose.2021.102580
0167-4048/

tackers have the opportunity to execute multiple penetration tests targeting known vulnerabilities to find VMs' security holes via low-priced services (Zhang et al., 2014). The administration of layers defines the other important factor in the security of service-based cloud computing. Non-uniform management in a layer creates multiple vulnerability entry points and exposes the system to more threats (Bohn et al., 2011; Kumar and Goyal, 2019).

In this study, we focus on the service-based cloud computing security concerns to analyze the current state of the field and classify them into a service-based taxonomy. The main contributions of this paper can be noted as follows.

1) Recent state-of-the-art service-based cloud vulnerabilities are presented.
2) A taxonomy of service-based cloud vulnerabilities and countermeasures is proposed.
3) Research challenges and future research directions are explored.
4) A classification of vulnerabilities and countermeasures is established.
5) Generic security issues in the service-based model are identified and enumerated.

The rest of the article is organised as follows: Section 2 describes background concepts of the field. The status of the current research is presented in Section 3. Current research challenges and future research directions are discussed in Sections 4 and 5. The article presents the future and conclusions in Sections 6 and 7.

2. Cloud computing overview

NIST addresses cloud security concerns according to three main categories, *service-based models, deployment models*, and *characteristics* (Bohn et al., 2011). Regarding the focus of this study, an overview of the service-based model technologies and concepts is discussed in this section.

2.1. Cloud computing enabling technologies

The existence of cloud computing has been possible only in the presence of essential concepts such as *virtualization, multitenancy*, and *Service Oriented Architectures (SOA)*. These techniques implement resource sharing among users from a physical instance (Sengupta et al., 2011; Verma and Kaushal, 2011).

2.1.1. Virtualization

Virtualization defines an abstract approach to create a computer, enabling resource partitioning in the cloud environment. Sharing resources becomes feasible with the help of a *VM*, via a file generally known as an image, which either can be produced by users or achieved from external sources (Barrowclough and Asif, 2018; Tabrizchi and Rafsanjani, 2020). In practice, any sharable IT resources could be virtualized to provide multi-user access to one resource instance. Desktop, network, storage, data, application, CPU, and cloud virtualization are the most adopted forms of virtualization. Cloud virtualization embodies IaaS, PaaS, and SaaS models, which implies resource virtualization (IBM Cloud Education, 2021; Malik et al., 2018; Rashid and Chaturvedi, 2019). Fig. 1 presents an abstract model of the service-based cloud computing environment. In this model, physical resources are allocated to numerous users of different layers with the help of a hypervisor through virtualization.

2.1.2. Hypervisor

A hypervisor (HV) or *Virtual Machine Monitor (VMM)* in the cloud environment behaves relatively similar to the OS in a traditional system. As a software layer between physical hardware and

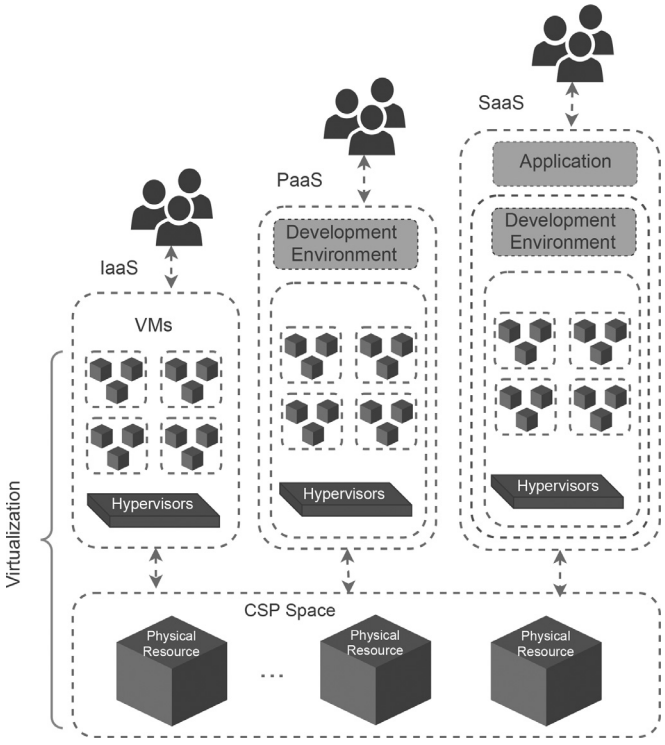


Fig. 1. Abstraction of multitenancy by virtualization in service-based cloud environments.

VMs, it coordinates the various VMs and assures them of receiving requested resources (Asvija et al., 2019; Barrowclough and Asif, 2018). Having numerous VMs altogether at the same time on one machine becomes possible with the HV technology. The most accepted HV classification has been defined as the two-type model, the *bare-metal* and *hosted* types. The former type operates directly on the bare hardware, e.g., Xen, and ESX, while the second type runs as an application on the host OS, e.g., KVM, QEMU, and VirtualBox. As a result of direct resource communication, the latency of the latter type decreases remarkably. While the high-performance capability makes bare-metal HVs a great option in a cloud environment, the root privileges turn them into an excellent target for security attacks (Bauman et al., 2015). CSPs such as Amazon AWS[1] offer various types of virtualizations such as paravirtual (PV) and hardware VM (HVM) that would be mapped into the hosted and bare-metal HV. Fig. 1 depicts a bare metal HV in which the HV directly translates user VM commands to the hardware and there is no need for a host OS.

2.1.3. Multitenancy

Multitenancy defines a software architecture helping several customers access one instance of software simultaneously. In this technique, multiple VMs located in a server utilize the same use the same physical entities to service end-users. A *Service-Oriented Architecture (SOA)* utilizes several mediatory technologies, e.g., *HTTP* and *Simple Object Access Protocol (SOAP)*, to provide the promised services to multiple customers. Fig. 1 depicts a possible resource virtualization leading to the pictured multitenancy scenario. In multitenancy, physical instances such as CPU and memory are divided into sharable elements, through virtualization, and allocated to different clients. The simultaneous access to one instance would degrade the shared resource performance on the one hand but maximize resource usage on the other hand. In theory,

[1] https://docs.aws.amazon.com

an isolated user-space would prevent security issues such as data leakage; however, that is not the case in real-world scenarios, and more vulnerabilities could be introduced to the cloud paradigm as a result of this technology (Kumar and Goyal, 2019; Tabrizchi and Rafsanjani, 2020).

2.1.4. Service oriented architecture

Service-Oriented Architecture (SOA) defines a reusable software development methodology in which components are loosely coupled to enhance interoperability and reusability. Undependability of services improves development agility and makes the SOA pattern a proper fit for new computation environments such as service-based cloud computing. In this model, target functionalities are provided through service interfaces. Services are typically defined through Web Service Definition Language (WSDL) standards and exhibited through SOAP or Representational State Transfer (REST) network protocols. This model of software development involves numerous benefits. A user needs the minimum amount of information to utilize the interface due to the loosely coupled components. The language of the provider could be different from the consumer, which increases undependability (IBM Cloud Education, 2021; Wang et al., 2014).

2.2. Service-based cloud computing

Providing economical high-quality services to users defines the main goal of the cloud computing paradigm. These services can be any sharable IT resources such as hardware, software, or network (Huang et al., 2015; Modi et al., 2013). IaaS, PaaS, and SaaS define three famous service-based cloud models that make the cost-effectiveness, availability, and scalability of these services popular for mid-size to large businesses (IBM Cloud Education, 2021), (see Fig. 2).

2.2.1. Infrastructure-as-a-Service

IaaS defines all computational resources in a virtual environment such as networking, data storage, servers, virtualization, and OS to facilitate remote services for clients. A user then can access the presented services through APIs via the internet. A company can rent all required IT resources to build a software ecosystem on a pay-per-use subscription basis. Amazon EC2 (Elastic Compute Cloud) represents an example of IaaS providers. In this environment, users have a higher level of flexibility in terms of having numerous VMs simultaneously. In IaaS, a user can deploy a private or public image, which is a template to configure a VM. Private images are configured by users while public images are published by an external source such as a company or an open-source organization. The architecture of IaaS might be different from that depicted in Fig. 2 based on a client desired model. In the *hosted HV* the VM OS operates on the host OS, which means the CSP manages the host OS, and the user governs the VM OS. The bare metal HV type, on the other hand, can be directly executed on the hardware, which eliminates the need for the host OS (IBM Cloud Education, 2021; Shaikh and Meshram, 2020; Vaquero et al., 2011).

2.2.2. Platform-as-a-Service

PaaS incorporates a cloud-based development environment with all required resources through the web medium. Normally, programming languages, IDEs, databases, web servers, and OS are accessible through shared resources so that a developer can produce a program free from the lower layer dependencies (Bach-Nutman, 2020; Pham et al., 2017). In this model, services are accessible through a Graphical User Interface (GUI) via the internet. As shown in Fig. 2, all IaaS layers plus *middleware* and *runtime* constitute the *PaaS* concept. Amazon web services and Windows Azure are two examples of the *PaaS* model (Singh et al., 2019; Toraskar and Borse, 2018). A software development team would find the required technologies for all software lifecycles, e.g., design, implementation, test, version control, and continuous integration and delivery in the PaaS model.

2.2.3. Software-as-a-Service

As depicted in Fig. 2, SaaS is a combination of all *IaaS* and *PaaS* layers with the addition of *data* and *application* panels that supply on-demand application services such as email, word processors, and design applications to the end-users. In this model, a client utilizes an application located in a remote cloud environment through a single instance of the application allowing several customers to execute the software simultaneously. In this model, all software stack and hardware components are provided and managed by CSP, and a user utilizes the ready-to-use application by an annual/monthly payment. The subscription model is beneficial for both providers and users. Clients pay less than a licensing model, and providers would have more clients as the software is more affordable. Google, Microsoft and Amazon are pioneers in providing such services, e.g., Google Drive, Microsoft 365, and Amazon AWS (Cook, 2018; Li et al., 2010; Loukis et al., 2019).

2.3. Cloud computing management

Management specifies the other important concept in service-based cloud computing. In a common classification schema, a cloud computing architecture is divided into nine layers, *networking, storage, servers, virtualization, OS, middleware, runtime, data*, and *application*. Management of layers might be granted to a user or CSP according to the service model (Varghese and Buyya, 2017). In IaaS, the management of networking, storage, server, virtualization, and OS are assigned to the CSP and a customer manages middleware, runtime, data, and application. In PaaS, a user only controls the data and application layer, and the CSP oversees the other layers. In SaaS, all layers of the cloud are administrated by the vendors and the consumer has limited administrative authority over an application (Manvi and Shyam, 2014). Fig. 2 illustrates a general resource management scenario without any assumptions about the service-based cloud model. It is worth noting that in real-world scenarios, the management of layers might differ according to the users' desired configurations. As discussed in virtualization (Section 2.1.1)and HV (2.1.2, regarding the type, the architecture and as a result, the management of layers might change, respectively (Malik et al., 2018; Rashid and Chaturvedi, 2019).

2.4. Deployment model

The deployment model defines the access exclusivity of the shared resources. A *public* cloud provides services to any user through the internet, whereas the *private* cloud computing model grants exclusive resource access to an organization. In this model, the administration could be either operated by the CSP or a customer. Likewise, the infrastructure could be located in the CSP location or out-sourced to a third-party private host (Kim and Vouk, 2014). A *community* model presents cloud services to a group of customers with common concerns, such as security. The administration and resource access are similar to the private model; however, a consumer could access other organizations' services through the organization. A *hybrid* model describes a combination of two or more deployment models (Bohn et al., 2011).

3. Recent advances in cloud computing security

Researchers have studied cloud computing security issues from various viewpoints; however, *virtualization, multitenancy, data security*, and *general* vulnerabilities are the most discussed topics in the

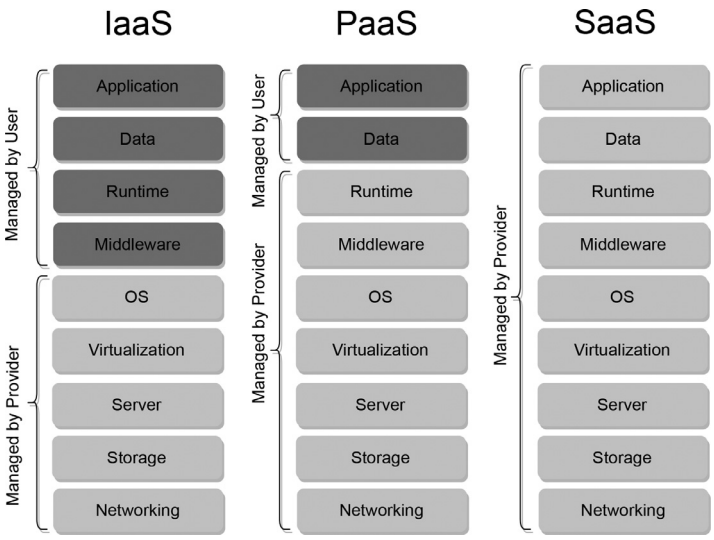


Fig. 2. Resource management in IPS model (IBM Cloud Education, 2021).

Table 1
Current status of surveys in cloud computing security (Sorted by Year).

Reference	Focused Topic	IaaS	PaaS	SaaS	Generic	Countermeasure	Year
Subashini et al. Subashini and Kavitha (2011)	Data Security	√	√	√	✗	√	2011
Vaquero et al. Vaquero et al. (2011)	Multitenancy	√	✗	✗	✗	✗	2011
Verma et al. Verma and Kaushal (2011)	General	√	√	√	√	✗	2011
Hashizume et al. Hashizume et al. (2013)	General	√	√	√	✗	√	2013
Modi et al. Modi et al. (2013)	Availability, confidentiality and integrity of cloud resources	√	√	√	✗	√	2013
Kim et al. Kim and Vouk (2014)	General	✗	✗	√	✗	√	2014
Fernandes et al. Fernandes et al. (2014)	General	✗	✗	✗	√	√	2014
Huang et al. Huang et al. (2015)	Compare industry best-practices with academia solutions	√	✗	✗	✗	√	2015
Chouhan et al. Chouhan et al. (2015)	Effect of data & application security in SaaS architecture	✗	√	✗	✗	✗	2015
Khan Khan (2016)	General	✗	✗	✗	√	√	2016
Singh et al. Singh et al. (2016)	General	✗	✗	✗	√	√	2016
Liu et al. Liu et al. (2015)	General	√	√	√	✗	✗	2016
Almorsy et al. Almorsy et al. (2016)	Cloud architecture, stakeholder, and characteristic	√	√	√	✗	✗	2016
Singh et al. Singh and Chatterjee (2017)	General	√	√	√	√	√	2017
Chawki et al. Chawki et al. (2018)	CSP & user behaviour	√	✗	✗	✗	✗	2018
Basu et al. Basu et al. (2018)	Virtualization & data	√	√	√	✗	√	2018
Kumar et al. Kumar and Goyal (2019)	Big Data, IoT, software defined network, & function virtualization	√	√	√	✗	✗	2019
Guerbouj et al. Guerbouj et al. (2019)	IoT and Cloud of Things (CoT)	✗	✗	✗	√	√	2019
Nadiah Almutairy (2019)	Virtualization	√	✗	✗	✗	√	2019
Agarwal et al. Agarwal et al. (2020)	Cryptography technique	√	√	√	✗	√	2020
Tabrizchi et al. Tabrizchi and Rafsanjani (2020)	General	✗	✗	✗	√	√	2020
Isharufe et al. Isharufe et al. (2020)	General	√	✗	✗	✗	✗	2020
Shyam et al. Shyam and Theja (2021)	Software defined networking	✗	✗	√	✗	✗	2021
Panda et al. Panda et al. (2021)	General	√	√	√	✗	✗	2021

literature. Table 1 summarizes the current status of surveys in the community. In this study, we propose a security taxonomy based on the aforementioned security concerns. In this taxonomy, vulnerabilities are generalized into two primary classes: *cloud-specific* and *cloud-generic*. The former discusses service-based specific security concerns, i.e., *IaaS, PaaS*, and *SaaS*. The latter addresses common security issues regardless of the layer. Fig. 3 illustrates the proposed taxonomy.

3.1. Cloud-Specific

The cloud computing paradigm becomes possible only in the presence of enabling technologies and concepts, such as virtualization and multitenancy. The service-based models, IaaS, PaaS, and SaaS, apply various techniques to provide services to the target customers. However, each technology might introduce a new security vulnerability to the cloud ecosystem. In this section, the most addressed security vulnerabilities and countermeasures in the literature are presented.

3.1.1. IaaS

The virtualized physical hardware is presented as a service to customers in the IaaS model. In this layer, the most common security issues are established around the virtualization concept. VM image, virtual network, HV, and hardware define top vulnerabilities in this layer (Bouayad et al., 2012). Table 2, presents a summary of vulnerabilities over the IaaS layer in addition to an associated countermeasure.

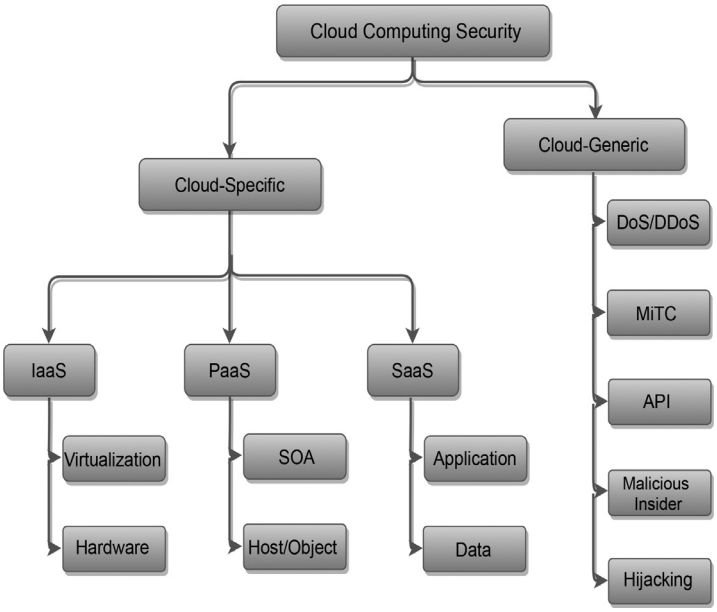


Fig. 3. Cloud computing security taxonomy.

Virtualization

VM images are files including important information such as VM configurations and logs, a well-known target of attacks. Image alteration by code injection and information theft encompass some examples of image template vulnerabilities (Almorsy et al., 2016). Gonzales et al. (2017) analyzed four IaaS architectures with different security configurations. The architecture with a series of encryption, access control, signature policy, and isolation is introduced as the most robust model. In this model, although VM images are protected through encryption mechanisms, still VM vulnerabilities such as VM CPU timing, side-channel attack, VM attack through the HV, disk injection to live VM, etc., threaten the system. A Bayesian network approach has been proposed to mitigate the mentioned vulnerabilities. In this approach, a network of elements located in a trusted zone is produced and the probability of an attack is calculated from the network paths (Gonzales et al., 2017).

Zhang et al. (2014) analyzed Amazon Machine Images (AMIs) security vulnerabilities on the AWS EC2. The established model calculates the risk-gain value of a vulnerability through tactical-game modelling in the system. In this model, an earlier reaction to an attack acquires more reward. The study revealed that more than 50% of VM vulnerabilities are related to the Ubuntu OS that makes the attack scenario easier for intruders. Having a misconfigured VM image increases the chance of a DoS attack. Maintaining all instances up-to-date is recommended as a feasible countermeasure, which defines a difficult obligation in practice. However, patching public VM images, maintaining running instances, giving patching priority to prevalent vulnerabilities, and shuffling cloud infrastructure smartly are introduced as more applicable solutions for mitigating VM image vulnerabilities.

Huang et al. (2015) elaborated on IaaS security analysis from a stakeholder perspective. Malicious activities are categorized into CSP and user attacks. The CSP can monitor and manipulate storage, VM images, HV, and Service Level Agreements (SLAs). In contrast, the user could cause cache-based or general leakage channels. Users of the public cloud services should trust the CSP to protect their data from other clients. This approach has led to new threats on confidentiality, integrity, and data availability that can be caused by malicious CSP or other clients. Contractual security is a new security property of customers specific to the cloud business model that is of interest to attackers. Dedicating a VM is proposed as a solution for cross-VM leakage and cloud-side encryption issues to protect the message confidentiality in this layer (Anwar et al., 2017).

VM escape denotes a vulnerability allowing an attacker in a VM to bypass the hypervisor to interact with the host OS to obtain root privilege. HyperSafe, Trusted Cloud Computing Platform (TCCP) and Trusted Virtual Datacenter (TVD) are introduced as three countermeasures of the VM escape problem. HyperSafe prevents hypervisor bypassing by preventing write-protected memory changes. TCCP provides an isolated execution environment. In this model, there exists a Trusted Third Party (TTP) that maintains a Trusted Coordinator (TC) and a Trusted Virtual Machine Monitor (TVMM). In the TVD approach, virtual machines are divided into groups with a common interest. Then, the intragroup communications are protected through secure channels (Hashizume et al., 2013).

The HV incorporates the multitenancy concept in the shared environment. The high privilege ability the HV it vulnerable as a target for intruders. If attackers penetrate an HV, they can execute any type of attack such as kernel structure manipulation and rootkit. Trusted Platform Module (TPM) is a hardware security solution that utilizes hardware capabilities to insure the security of the components. The technology applies the BIOS signature mechanism for secure boot time. Modern hardware processors are equipped with a crypto chip insuring secure boot time that prevents HV tampering. The processor can verify the software boot-time information through a series of assessments on the chip (Gonzales et al., 2017).

Mazhar et.al. (Ali et al., 2015) outlined the security concerns mainly related to the third-party service providers. The primary security issues related to third-party CSPs are the points that emerge due to virtualization, multitenancy, and shared resource pools. Although the research in this context essentially focuses on the communication and architectural perspectives, the virtual network demands more attention. Even though virtual devices were presented to secure the virtual network, a comprehensive, well-planned design is required to regulate or monitor the traffic to prevent information leakage. Shared technologies such as virtualization, HV, and VMs have generated new security gates for adversaries. Rewriting the packets could be a solution for VM security matters, maintaining a balance between privacy and monitoring. The tamper-

Table 2
IaaS security countermeasures.

Vulnerability	Threat	Countermeasures	Definition	References
	VM escape	HyperSafe	An approach to prevent hypervisor bypassing through preventing write-protected pages alteration	Hashizume et al. (2013)
	VM escape	TCCP	Trusted cloud computing platform provides an isolated execution environment through a trusted third party	Hashizume et al. (2013)
	VM escape	TVD	Trusted virtual data center divides VMs into groups with common interests and limits communication between groups through secure channels	Hashizume et al. (2013)
	Vulnerable VM image	Patching public images	Establish policies to keep public VM images up-to-date, either by CSP, provider, or user	Zhang et al. (2014)
Virtualization	Cross-VM side-channel attack	MetaMORP(h)OSY	A thermal behaviour analysis tool that evaluates run-time thermal status	Amato et al. (2018)
	Network virtualization	VM mapping	Hanging a VM to the related host by devoted physical channels	Chawki et al. (2018)
	Cross VM leakage	Dedicated instances	Normally used by large scale businesses as a resource is entirely allocated to a customer	Huang et al. (2015)
	HV DoS	Isolation	Isolating the security monitoring VM from guest VMs	Rakotondravony et al. (2017)
	HV tampering	TPM attestation, patch HV	Trusted platform module is a secure co-processor on the motherboard of a computer system for signing a measurement	Huang et al. (2015)
	Cache-based side-channel attack	S-Box access	Turn off cache S-Box access, avoid lookup table, and perform cache warming	Coppolino et al. (2017)
	Cache-based side-channel attack	SGX & ARM TrustZone	Intel software guard extension provides a hardware base solution to isolate an application's memory access	Coppolino et al. (2017)
Hardware	Information leakage	PC, PLC	Partitioned cache divides cache into protected sections and allocates each to a process. In partition-locked cache, a fine-grained locking mechanism is in place for isolating only a line of cache	Coppolino et al. (2017)
	XML attack	XML signature and encryption	An approach for creating an XML signature via an XML syntax	Arora et al. (2012)

proof key management makes trusted computing a good candidate for providing a comprehensive security solution in cloud computing (Cabuk et al., 2008). SLA specifies a countermeasures for virtualization and multitenancy; however, having the whole benefit of the solution depends on the CSP's policy. Google and Microsoft are some examples of CSPs who are reluctant to reveal all required information of SLA transparency (Halboob et al., 2014).

Hardware

Cryptography mechanisms are applied to increase the security level of data in the transit and storage processes. Despite the mechanism in place, data should be decrypted for process purposes at some point. The multitenancy feature of the cloud facilitates access to storage mediums such as disk, memory, and cache. Intruders located in a shared host with a victim can access the plain value of the key, or any form of confidential data in the storage mediums. Cache-based side-channel attacks, a family of cross-VM side-channel vulnerabilities, represent a form of these concerns. Another complaint with cloud-based services is the access limitation of upper layer users to the lower layers. Intel is working on the Software Guard Extension (SGX) technology that provides a protected memory area to run an application called an enclave. In the secure enclaves, even privileged software such as the OS has no right to access the protected area (Coppolino et al., 2017).

3.1.2. PaaS

In this layer, all required services are provided to customers for deployment purposes via an SOA model. Resource sharing via multitenancy and an SOA increase the risk of numerous security issues Bouayad et al. (2012). Table 3, presents a summary of vulnerabilities over the PaaS layer in addition to an associated countermeasure.

SOA

Resource sharing raises serious issues when there exists a conflict of interest between customers. A possible scenario is the colocation of two competitors in a single host. The *Chinese Wall Model* mitigates the accidental or intentional access to shared resources by dividing users into conflict of interest groups and allocating physical resources accordingly (Hay et al., 2011). Arora et al. (2012) propose a combination of policies, monitoring, and restrictions as the solution for multitenancy and virtualization. *SLA* represents one of the policies determining the advantages and liabilities of each participant. *Secure Configuration Policy (SCP)* describes another policy that guarantees a secure configuration in the hardware/software layer or SLA configuration.

Freet et al. (2015) investigate the digital forensic security challenges in the service-based cloud environment. In this paradigm, processing power, data storage, and other shared resources rely on the IaaS layer. In a common shared resource environment, the data packets of the VM traverse in all possible ways via a host machine. Although each VM is separated from other VMs on an actual device, any undermined VM can assault another VM in the organization. Additionally, any misconfiguration of an HV results in a DoS attack as it permits one VM to utilize all framework assets against other VMs in a shared environment. Any changes in the configuration and settings by a malicious user in the PaaS layer can influence the whole cloud architecture. As PaaS has a service-oriented architecture, the primary security challenges are XML-related, DoS, injection, and MiTC attacks in this layer.

Table 3
PaaS security countermeasures.

Vulnerability	Threat	Countermeasures	Definition	References
	Shared resource	Chinese wall model	An approach to allocate physical resources according to class of customers	Hay et al. (2011)
SOA	Resource starvation	Resource accounting	Applying tools such as Java VM tooling interface (JVMTI) to limit resource access	Rodero-Merino et al. (2012)
	Information leakage	Safe thread termination	A thread should be properly terminated to prevent leakage of information in a thread related to a user	Rodero-Merino et al. (2012)
Vulnerable Host/Object	Vulnerable object	TCB	Trusted computing base is a secure layer over the OS to cope with the lack of interoperability	Sandikkaya and Harmanci (2012)

Rodero-Merino et al. (2012) discuss two popular programming language technologies in the PaaS paradigm, i.e., Java, and.Net, in terms of multitenancy subjects. According to experimental results, Java and.NET do not offer a fully secured hosting setting. More specifically, a detailed analysis of the security features over the *Enterprise Java Bean (EJB)* and *Open Service Gateway Initiative (OSGi)* were conducted to evaluate the security of the most prominent Java containers in the cloud domain. In this study, *isolation*, *resource accounting*, and *safe thread termination* are proposed as remedies for the multitenancy technology vulnerabilities. Java supports isolation through JVM technologies, i.e., EJB containers and servlets, whereas the.Net platform facilitates the isolation through Common Language Runtime (CLR) profiling. Google App Engine (GAE), a Java-based cloud engine, applies another approach for isolation. In a GAE, each entity resides in an isolated VM and has restricted access to resources.

Vulnerable Host/Object

In a shared environment, customers' objects are threatened by multiple elements, i.e., other tenants, hosts, and external attackers. The combined *Trusted Computing Base (TCB)* is normally applied as a solution for vulnerable objects and lack of interoperability for intra-API communications (Verma and Adhikari, 2020). A model including four practices is proposed to address vulnerable objects: *Transport Layer Security (TLS), Sticky Access Control Policy (SACP), Policy Enforcement Points (PEPs)*, and *Undeniable Logging Protocol (ULP).* constitute the model. The well-known network protocol, TLS, delivers secure communication through a cryptography mechanism. The SACP and PEP deploy a fine-grained object-based access control and ULP assures the authenticity of the logging system (Sandikkaya and Harmanci, 2012). Interoperability defines the fundamental concept enabling cloud paradigm, APIs, and platforms to communicate together but is recognized as a PaaS vulnerability. A TCB provides a solution for vulnerable hosts and lack of interoperability. An encryption layer can be added to protect against exposed objects in the proposed method (McKay and Cooper, 2018).

3.1.3. SaaS

Built on top of two layers, SaaS inherits security issues of lower layers. In addition, dependence on web APIs makes the model vulnerable to web technology security issues (Bouayad et al., 2012). Table 4 presents a summary of vulnerabilities over the SaaS layer in addition to the associated countermeasures.

Application

In the OWASP project, web technology vulnerabilities have been studied and the top ten are introduced. Broken authentication, injection, XML External Entities (XXE), broken access control, sensitive data exposure, security misconfiguration, insecure deserialization, cross-site scripting (XSS), and insufficient logging and monitoring define part of web API concerns (Li, 2020).

Chouhan et al. (2015) classify SaaS security issues into three main categories, data, application, and deployment. Data security includes security of data in storage, transit, backup, recovery, integrity, and access control. The delivery of SaaS services strongly depends on web technologies and concepts. Software design flow, user interface technologies, web services, and malware define application security points. Design and implementation of a web application include front-end and back-end languages, libraries, and dependencies such as HTML, JavaScript, PHP, Java, Python, SOAP, etc. Normally, a design might not cover all security aspects and introduces subsequent vulnerabilities into the system.

Grobauer et al. (2011) reviewed the security subjects of the core cloud computing technologies and their characteristics. Vulnerabilities are divided into the following categories: core technologies, essential cloud characteristics, prevalent security concerns, defects in known security controls, and architectural components. The authentication topic has been introduced as the primary vulnerability of the cloud system that compromises user data. As the SaaS layer communicates directly with the end-users through web APIs, the layer is vulnerable to web technology security issues such as authorization, access control, and session hijacking. Each cloud environment must have strong mechanisms and protocols to control identity, authentication, authorization, and auditing. The cryptographic algorithms are counted as a remedy for a majority of security issues. A secure channel through cryptography highly alleviates the hijacking threat.

Data security is an important issue in all layers of the cloud; however, SaaS users totally rely on CSP to protect any breaches of credential information either in transit or in storage. Hashizume et al. (2013) discussed three countermeasures to address the data breach problem, Fragmentation-Redundancy-Scattering (FRS), digital signatures, and homomorphic encryption. FRS splits primary data into parts with little meaningful information and propagates the parts across the whole system in a redundant way. In the digital signature approach, the RSA algorithm is applied to verify data authenticity after transit through the network. Homomorphic algorithms are applied to messages that are manipulated in an encrypted format. Depending on the goal of a system, one or a combination of the aforementioned approaches would be a solution for protection against a data breach.

Data vulnerabilities in this layer can be mitigated by protocols such as Secure Socket Layer (SSL) and TLS. The protocols create a secure channel between a client and server to establish end-toend secure communication. HTTP examination is another solution to enhance data issues. To this end, a web application scanner examines the HTTP requests and responses regularly to provide log files via read- only APIs through a central log server (Freet et al., 2015).

3.2. Cloud-generic vulnerabilities

Cloud computing embodies network technologies that comprise inherited security issues such as TCP/IP communication vulnerabilities (Khalaf et al., 2019). In this model, a layer would have specific and generic security concerns. The former arises due to applied technology in a layer such as virtualization in IaaS; the latter can be either for the network or common cloud-based issues in all

Table 4
SaaS security countermeasures.

Vulnerability	Threat	Countermeasures	Definition	References
Application	Unauthorized access	IAAA	A cloud service must have a strong identity, authentication, authorization, and auditing control mechanism	Grobauer et al. (2011)
	Web API security	Isolation	Isolating transactions in memory to limit data access of tenants located in the same instance	Bouayad et al. (2012)
Data	Data breach	FRS	FRS technique splits data into low informative parts and propagates them in the whole system in a redundant way	Hashizume et al. (2013)
	Data breach	Digital signature	A cryptographic approach, normally using an RSA algorithm, to confirm authenticity of data after transit	Hashizume et al. (2013)
	Data breach	Homomorphic encryption	A cryptographic approach that provides process over encrypted context	Hashizume et al. (2013)
	Data recovery	Cryptography	Cryptography and strong key management mechanisms help to mitigate user-related data recovery	Grobauer et al. (2011)
	Data alteration	SSL/TLS	SSL and TLS create a secure tunnel between a client and server making communication end-to-end secure	Freet et al. (2015)
	Data vulnerability	HTTP examination	A web application scanner examines HTTP requests and responses, and provides log files via read only APIs through a central log server	Freet et al. (2015)
	Data breach	Cryptography	Applying cryptographic algorithms and facilitates data storage backup	Chawki et al. (2018)

Table 5
Cloud-generic security countermeasures.

Vulnerability	Countermeasures	Definition	References
DoS	Filtering	Ingress filtering insurers that an IP address matches with a domain prefix, otherwise it will be dropped	Coppolino et al. (2017)
	SYN analysis	Prevent incompleted TCP handshake synchronization (SYN) processes	Coppolino et al. (2017)
	Connection limit	Limiting connection numbers from one IP address	Coppolino et al. (2017)
	MetaMORP(h)OSY	Thermal behaviour evaluation as a formal massive object-based profiler for the real-time modelling behaviour of a software system	Amato et al. (2018)
MiTC	A series of predictive steps	Activate firewall and IDS, disable ping, conceal sensitive information, close unused ports	Jabir et al. (2016)
	Central log server	Deploy a central log server, equipped with encryption and signature to prevent eavesdropping of files or any alterations	Freet et al. (2015)
API	SAML & MFA	Multi-factor authentication, and security assertion markup language for transfering user credentials purposes	Isharufe et al. (2020)
	Shibboleth	Shibboleth is an open-source middleware software that applies the SAML standard to guarantee proper authentication and authorization.	Zissis and Lekkas (2012)
	Bayesian network	Statistical approach for finding the attack path	Gonzales et al. (2017)
Malicious insider	DAC, MAC	Discretionary and mandatory access control, applied to the OS for access control restrictions	Coppolino et al. (2017)
	Agreement and breach notifications	Using agreement reporting and breach notifications in addition to transparent security and management policies	Chawki et al. (2018)
	Two-factor authentication	Using a secondary device or approach for authentication purposes	Chawki et al. (2018)
	Dynamic credential	A parameter based credential change approach, it can be sensitive to user location, or TCP/IP changes	Hashizume et al. (2013)
Hijacking	PDP	Provable data possession is a cryptography approach that regularly checks server activity on data	Selvamani and Jayanthi (2015)
	NIDS	Applying an erasure code technique in network-based intrusion detection system to recognize vulnerabilities and fix them simultaneously	Chawki et al. (2018)

layers (Coppolino et al., 2017). Table 5 presents a summary of cloud generic vulnerabilities in addition to associated countermeasures.

3.2.1. DoS/DDoS

In Denial of Service (DoS) or Distributed Denial of Service (DDoS) attacks, the attacker exploits a TCP vulnerability to cause resource starvation for a legitimate user. In the cloud paradigm, VMs are exposed to DoS attacks due to the insufficient bandwidth under-provisioning issue. A series of papers propose solutions to mitigate this attack (Arora et al., 2012; Jabir et al., 2016).

SYN cookie analysis and connection limitation outline a series of solutions to prevent DoS attacks. In SYN flow analysis, a se-

quence of synchronization messages are communicated between client and server. In the SYN flood attack, the attacker initiates the handshaking for synchronization, but never completes it entirely. To prevent this type of DoS attack, the server never waits for an acknowledgement. The handshaking process will be completed once the SYN-ACK is correctly received. By this approach, the server is never occupied with the incomplete synchronization mechanism required for a TCP connection. The other DoS attack is applied by holding resources busy through open connections. This type of attack is normally prevented by limiting connections from a client with the same IP address (Coppolino et al., 2017).

DoS/DDoS attacks can be detected by thermal behaviour analysis. MetaMORP(h)OSy is a formal massive object-based profiler for the real-time behaviour modelling of a software system. The profiler can be extended to examine thermal requirements and behaviours. Formal models are adopted to determine strange actions, verify functional and non-functional properties at the early stage, and implement monitoring at run-time. MetaMORP(h)OSY produces an *observer* at runtime for evaluating thermal properties. The observer monitors central interfaces and thermal regions on a remote system to find any differences between expected and normal thermal behaviour; the monitor evaluates new process parameters in small time fractions (Amato et al., 2018).

3.2.2. MiTC

Man-in-The-Cloud (MiTC) attack determines the second important vulnerability in the generic category. In this threat, the intruder commences by collecting information from the web to target a victim. Once detecting vulnerabilities, such as open ports or unprotected servers, the attacker performs a malicious activity. Therefore, a series of operations is recommended to mitigate the probability of MiTC (Arora et al., 2012). Probes should be prevented through strongly configured firewalls and IDSs and critical data should be hidden. Closing non-essential ports and preventing routing bypass through any mechanism are strongly suggested. In addition, the current recommendations are beneficial in DoS attack prevention (Jabir et al., 2016).

3.2.3. API

API denotes the key concept of the cloud ecosystem for communication. Although this feature facilitates a convenient mechanism for information transmission it raises security concerns. A lack of interoperability could lead to a serious issue if a well-defined policy is not already in place (Sandikkaya and Harmanci, 2012). To prevent malicious activity, such as eavesdropping or alteration, the customer can benefit from a central server log system that captures all activities. Then, the log file is stored signed and encrypted to prevent alteration (Freet et al., 2015).

Isharufe et al. (2020) address PaaS security issues due to cloud features, i.e., *on-demand self-service, broad network access, resource pooling, rapid elasticity*, and *measured service*. On-demand self-service capability is provided through an API to customers. As recommended, the CSP should utilize a Multi-factor Authentication (MFA) mechanism to ensure the identity of the user and the confidential data should be transferred through a secure system such as the latest version of Security Assertion Markup Language (SAML).

Zissis and Lekkas (2012) analyzed authentication and authorization as two important processes in an information system. Authentication verifies user identity versus authorization defines the level of access to either hardware or software resources. Shibboleth is open-source middleware software that applies SAML standards to guarantee proper authentication and authorization. Shibboleth implies a Single Sign-On (SSO) standard that relies on third-party mediation (Zissis and Lekkas, 2012).

3.2.4. Malicious insider

A malicious insider in a cloud computing environment could be a CSP employee who misuses his/her access privileges in a nefarious way. Discretionary Access Control (DAC) and Mandatory Access Control (MAC) introduce two approaches for preventing such vulnerabilities (Sengupta et al., 2011). Both methods are applied to the OS to restrict access through strict permission policies. MAC is stated as the proper access restriction mechanism for the cloud environment due to the higher level of security. AppArmor in Linux and TrustedBSD in the Mac OS are examples of the MAC approach (Coppolino et al., 2017).

Kamongi et al. (2013) proposed a framework for evaluating the security vulnerabilities of the Cloud Computing System (CCS) named *VULCAN*, a comprehensive security assessment via a *Natural Language Processing (NLP)* method and *ontology reasoning*. The framework benefits from the *Ontology Vulnerability Database (OVDB)* and *National Vulnerabilities Database (NVD)* repositories to identify known security issues or new patterns. The framework indexes all possible vulnerabilities based on the script of NVD and OVDB. Then, a cloud-based system should be tested for any security concerns. The VULCAN framework functions similar to a classifier in which all vulnerabilities are categorized and labelled into groups. When a vulnerability is discovered, the framework will mark that new instance as one of the known categories based on the vulnerability features. Gonzales et al. (2017) applied a Bayesian network to find the malicious insider paths. The primary purpose of finding the attack path is to understand the vulnerability level of the information system to derive probabilistic standards of enterprise network security. The proposed approach has been extended to the CCSs by constructing an acyclic directed graph through the attack paths. This approach attempts to consider the contributions of specific CCS security features in reducing the vulnerabilities of elements in a CCS to reduce the overall security profile of an IaaS cloud.

3.2.5. Hijacking

Account or service hijacking is the theft of user credential data allowing further malicious activities. Proper identity and access management policies help in mitigating the issue. Dynamic credentials are a recommended countermeasure for hijacking. This method changes the secret values based on predefined parameters such as user location or received packages (Hashizume et al., 2013).

Khan and Al-Yasiri (2016) present current and future privacy and security arguments by interviewing cloud developers, providers, and IT managers. A vulnerability might originate from a misconfiguration or improper action in various layers or stages. Finding the source of the issue helps in detecting and preventing hijacking. The result of the study shows that weak credentials and improper authorization validations are typical causes that lead to an account or service hijacking. Inappropriate data handling in transit, processing and storage by an untrusted third party could cause data leakage. Insecure third-party APIs increase the DoS attack probability. By cross-site scripting or SQL injection, attackers can manipulate user data. An unprotected virtual machine increases the virtual network sniffing/spoofing attack probability.

In account or service hijacking, the attacker steals credential information so that they can exploit the system. As the cloud paradigm opens new entry points for the intruders, hijacking can be operated from either layer. Two-factor authentication, Provable Data Possession (PDP), Network-based Intrusion Detection Systems (NIDS), and cryptography algorithms are a group of solutions to mitigate the account or session hijacking threats (Selvamani and Jayanthi, 2015). In two-factor authentication, normally a secondary device is involved to validate the user authenticity. Whereas the PDP, defines a public key-based method to verify the manipulated

data by a server, NIDS apply the erasure codes method for intrusion detection (Chawki et al., 2018).

4. Challenges

The cloud computing paradigm provides numerous benefits for businesses and individuals. However, applied technologies and complex architectures raise challenges that need to be addressed. Regarding the scope of the current study, we will discuss IaaS, PaaS, SaaS, and generic security challenges.

4.1. IaaSs security challenges

According to the literature, lower layers' vulnerabilities have the most destructive effect on the whole system. That means, if a security issue initiates in the IaaS layer, that would propagate to the upper layers and endanger the whole system. Due to the access restrictions in lower layers, the customer has few chances to apply an appropriate security countermeasure (Hashizume et al., 2013). Virtualization and multitenancy are the top security concerns in this layer. The virtualization in the IaaS environment would cause associated security issues such as DoS and cross-VM side-channel attacks due to the bandwidth under-provisioning issues or co-location VM escape (Chawki et al., 2018).

VM images in an IaaS model include potential vulnerabilities. In this environment, the user can use a public or private image to configure VMs. Public images are published with outer providers such as individuals, open-source communities, or IT companies. In Amazon EC2 alone, a user can apply more than 6000 public images. Regarding the result of a current study, on public images have the potential of backdoors as providers might forget to remove keys or other critical information properly (Zhang et al., 2014).

Stability of network configurations create a conventional attack surface for intruders. In the IaaS layer, normally the range of IP addresses is more predictable and stable compared to the traditional computing model. In addition, non-cloud machines might utilize firewalls or another protective mechanism. The heterogeneous environment of the cloud makes it easier for an intruder to exploit security holes (Zhang et al., 2014).

4.2. PaaS security challenges

Lock-in defines a PaaS concern. The PaaS paradigm provides a development environment for a software developer to enjoy extensible software and hardware resources for developing a product. As there is no unified standard for all vendors, the *lock-in* issue threatens PaaS customers. Lock-in happens once the customer requires services that are not available in the primary vendor environment Kritikos et al. (2017). To this end, the customer should migrate to another provider or host facility. However, the migration to other vendor servers is highly inefficient in terms of budget or time for the customer. Therefore, there would be a lock-in condition for PaaS customers who should decide between staying in a vendor with limitations or accept the cost of migration Arora et al. (2012).

The **SOA** model enables PaaS customers to deploy their application or software in a shared environment. Although the model provides numerous benefits, it limits access to the lower layer, making it difficult for a PaaS customer to apply security tools. Therefore, the primary configuration makes the system vulnerable to MiTC, DoS, injection, and XML-related attacks. However, the PaaS API should necessarily include high security standards for service delivery to the upper layer customers Freet et al. (2015).

4.3. SaaS security challenges

Multitenancy facilitates a cheaper service either for a provider or end-user. In some cases, users should only pay-as-you-go, while in others they might receive free services such as Google Docs. This concept supports the coexistence of numerous users in a single instance of software/hardware at the same time. In this environment, data management could become a challenging affair. Preserving data locality, integrity, confidentiality, segregation, and backup could become difficult to manage as several users will be using the same system.

Web API defines the SaaS delivery model to final customers that exposes the method to various web technology security flaws. Therefore, lack of a proper policy could lead to severe security or privacy problems such as access to user data in a common area in the presence of multitenancy and data leakage in a shared database system (Vaquero et al., 2011).

Control limitation explains another main challenge since a user has no control over the application, OS, and middleware in the SaaS cloud. All services are controlled by the CSP and users can access only the rented application from the CSP. This limitation intuitively means that a customer has no access to the log files or any tools to monitor, alter, apply or improve security policies.

4.4. Cloud-Generic security challenges

Patching software regularly reduces security concern at all layer. Despite the provided taxonomy, we realize that the late patching process of software was mentioned as a potential vulnerability. By releasing on-time patching, software providers reduce the risk of various security issues. On the other side, users that ignore new updates would face the same consequences (Zhang et al., 2014).

Cost-effectiveness of a cloud environment is not only a beneficial concept for users but also intruders. Having inexpensive machines makes the penetration test easier for attackers. Intruders can benefit from this concept to exploit cloud customers (Zhang et al., 2014).

Lack of interoperability explains a potential security issue of APIs. Interoperability means the ability to communicate between different components or platforms of a system in a compatible way that normally is operated through an API. There are various scenarios underlying this ability such as migration from one CSP to another, or upgrading services in a CSP or customer side (Machado et al., 2009).

Internet protocol is associated with known security flaws such as DoS, DDoS, MiTC, and account or service hijacking. All aforementioned vulnerabilities can be operated from all layers. However, according to the PaaS architecture model, intruders tend to operate the attack scenario in this layer more than others (Bouayad et al., 2012).

Malicious insiders is a known security issue that needs more attention. According to Bouayad et al. (2012), more than 70% of attacks are related to the company's human resources. Although products such as SGX can help us to make some progress in this direction, it still demands greater efforts (Coppolino et al., 2017).

5. Discussion

In the literature review DoS/DDoS, session hijacking and shared technologies are the top three discussed security concerns in the cloud computing environment (Agarwal et al., 2020; Yan et al., 2014). According to the research results, scalability is one of the core cloud characteristics, such that a user can request more resources based on a given workload. That means, on one hand, a cloud system provides scalability to users, on the other hand, it

Table 6
Top security threats in service-based cloud computing environments.

Vulnerability	Description	Research
DoS/DDoS	DoS attack, prevents a legitimate user from achieving desired resources. Normally, the attacker occupies all resources such as network bandwidth up to the maximum capacity.	Coppolino et al. (2017) Jabir et al. (2016) Khan and Al-Yasiri (2016) Rakotondravony et al. (2017) Freet et al. (2015) Amato et al. (2018) Chawki et al. (2018) Arora et al. (2012)
Shared technology	Shared technologies such as virtualization facilitate the cloud computing model. The attacker tries to obtain control of VMs through the HV, which has root privilege.	Krishna et al. (2016) Rodero-Merino et al. (2012) Coppolino et al. (2017) Khan and Al-Yasiri (2016) Rakotondravony et al. (2017) Gonzales et al. (2017)
Session hijacking	In this attack, the attacker obtains a user's credentials by staying in a TCP/IP communication. As a result, the attacker can access user's resources, and steal their identity and sensitive data.	Grobauer et al. (2011) Krishna et al. (2016) Khan and Al-Yasiri (2016) Freet et al. (2015) Chawki et al. (2018) Isharufe et al. (2020)
Multitenancy	Multitenancy is the result of virtualization technology in a cloud platform. It permits coexistence of multiple users in one physical resource instance through VMs.	Almorsy et al. (2016) Rakotondravony et al. (2017) Bouayad et al. (2012) Rodero-Merino et al. (2012)
VM side channel	In this attack, the attacker locates a malicious VM in the target host to collect cryptography algorithm information that allows the attack to occur in cipher texts.	Gonzales et al. (2017) Rakotondravony et al. (2017) Isharufe et al. (2020) Chawki et al. (2018)
MiTC	In a cloud environment, a synchronization token is used for access to user data. Utilizing a malware, the attacker alters the token to access the required info. Having a successful implementation, the attacker has data access from any machine.	Bouayad et al. (2012) Freet et al. (2015) Isharufe et al. (2020) Chawki et al. (2018)
Malicious insiders	In this attack, one of the CSP employees accesses confidential data and utilizes collected information in a malicious way. This attack defines one of the most dangerous vulnerabilities.	Coppolino et al. (2017) Krishna et al. (2016) Chawki et al. (2018)
Data breach	The possibility of data leakage increases in the cloud environment. Shared technologies and multitenancy are some examples of cloud technologies that raise the data breach probability.	Coppolino et al. (2017) Khan and Al-Yasiri (2016)

supports inexpensive resources for intruders and attackers to target other systems (Coppolino et al., 2017; Deshmukh and Devadkar, 2015). The second most discussed issue is the shared technology that makes the cloud so fascinating and is also a point of criticality in terms of security (Coppolino et al., 2017). In a shared environment, if attackers succeed in compromising the HV, they can take control over the host system due to the HV root privilege. Apart from this, a session hijacking vulnerability is also in the top three issues in the cloud where a legitimate host can lose control over its own system, allowing intruders to compromise security requirements such as confidentiality, availability, and integrity of the deployed services in the host system. Table 6 depicts top security concerns in the literature. The results indicate that the lower layer vulnerability is more important and malicious insiders are one of the unexplored issues in the cloud system. The main contribution of this study is to provide a detailed understanding of the security vulnerabilities in the service-based cloud model. Moreover, previous studies have discussed countermeasures or vulnerabilities separately, which is fairly difficult to identify the solution for a particular vulnerability. That is why our result describes a classified table that has been produced to provide a pair of security countermeasure information.

6. Future research directions

Based on our literature search, most studies analyzed IaaS and SaaS security issues and there is limited research on PaaS vulnerabilities. Unfortunately, those limited number of studies poorly address countermeasures or a framework to solve security flaws. It intuitively implies either the model is less popular among others, or it has fewer known vulnerabilities in the community. However, according to the current software development trend, study of the security concerns of this layer requires more effort.

6.1. Transparent policy compliance

The cloud computing model suffers from a lack of coherent policies in terms of security and service compatibility. Vendors have no integrated instructions either for security policies or service delivery constraints. As discussed, the *lock-in* issue for PaaS customers arises due to the mentioned limitations of the model. Working on a universal security schema in the cloud platform is an essential future direction that necessarily improves the service quality and security status (Kritikos et al., 2017).

6.2. Cloud-based hardware security concerns

Well-known companies, such as Intel, are working on hardware-level protection concepts such as the aforementioned technology, SGX; however, these technologies are still experimental and need more investigation. In addition, most of them are deployable in the traditional computing model. Cloud-based hardware is still developing and requires more research and enhancement efforts (Coppolino et al., 2017).

6.3. Exploiting vulnerabilities in virtualization technologies

The other important finding of this study is that the literature is focused on the vulnerabilities of virtualization technologies. That is obviously due to the importance of the concept in the service-based cloud computing model. Although virtualization stands among well-studied vulnerabilities, in current research exploration, it was difficult to find a study investigating all possible scenarios. Virtualization is mainly classified into four categories as discussed. However, comprehensive surveys of virtualization vulnerabilities over class types were scarce in the literature. In the future, investigating the different security aspects of one vulnerability regarding various scenarios would add value in this context. The results of such investigations would help both CSPs and users to employ best practices.

7. Conclusion

Our research studied the last decade of service-based cloud computing security issues through a comprehensive analysis of high-quality published papers. This study aimed to provide a summary of the current research status and establish a taxonomy that maps vulnerabilities to proper countermeasures. To this end, security vulnerabilities were categorized into four classes, IaaS, PaaS, SaaS, and generic. The first three classes discuss common issues in

each layer, while the generic category address vulnerabilities possible in all layers. Although the security concerns have more varieties, we tried to summarize the most discussed topics in the literature.

According to the research results, DoS/DDoS, shared technology, and session hijacking were among the most addressed issues. As has been presented, DoS/DDoS attacks were the most frequent concern in the literature. A series of vulnerabilities are common network security issues that arise due to the medium of the cloud, such as DoS and MiTC. While others, e.g., multitenancy, are cloud-specific groups. As studies show, common issues would be a concern for more study regardless of purposes, service model, or architecture. The complexity of cloud architectures, in addition to service diversity and user configurations, could initiate new security threats. Currently, the variety of cloud services has increased to address any form of user requirements. Although it provides more flexibility, new security holes might be introduced for malicious activities. In the literature, most vulnerabilities are discussed in the general configuration or the traditional structure; however, both providers and customers need to be more conscious of risks and challenges associated with individual decided compositions.

Declaration of Competing Interest

We wish to draw the attention of the Editor to the following facts which may be considered as potential conflicts of interest and to significant financial contributions to this work.

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

We confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed. We further confirm that the order of authors listed in the manuscript has been approved by all of us.

We confirm that we have given due consideration to the protection of intellectual property associated with this work and that there are no impediments to publication, including the timing of publication, with respect to intellectual property. In so doing we confirm that we have followed the regulations of our institutions concerning intellectual property.

We understand that the Corresponding Author is the sole contact for the Editorial process (including Editorial Manager and direct communications with the office). He/she is responsible for communicating with the other authors about progress, submissions of revisions and final approval of proofs. We confirm that we have provided a current, correct email address which is accessible by the Corresponding Author.

CRediT authorship contribution statement

Fatemeh Khoda Parast: Conceptualization, Methodology, Validation, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization, Project administration. **Chandni Sindhav:** Investigation, Writing – original draft. **Seema Nikam:** Investigation, Writing – original draft. **Hadiseh Izadi Yekta:** Investigation, Writing – original draft. **Kenneth B. Kent:** Writing – review & editing, Supervision, Funding acquisition. **Saqib Hakak:** Methodology, Validation, Writing – review & editing, Supervision, Project administration.

Acknowledgments

Authors Khoda Parast and Kent would like to acknowledge the financial support of the Lockheed Martin Cybersecurity Research Fund (LMCRF), mitacs and the Natural Sciences and Engineering Research Council (NSERC) for their research contributions to this paper. In addition, we would like to express special appreciation to Stephen MacKay for his professional guidance in improving the quality of the current research document.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.cose.2021.102580

References

Agarwal, V., Kaushal, A.K., Chouhan, L., 2020. A survey on cloud computing security issues and cryptographic techniques. In: Social Networking and Computational Intelligence. Springer, pp. 119–134.

Ali, M., Khan, S.U., Vasilakos, A.V., 2015. Security in cloud computing: opportunities and challenges. Inf. Sci. 305, 357–383. doi:10.1016/j.ins.2015.01.025.

Almorsy, M., Grundy, J.C., Müller, I., 2016. An Analysis of the Cloud Computing Security Problem. abs/1609.01107, http://arxiv.org/abs/1609.01107

Almutairy, N.M., 2019. A taxonomy of virtualization security issues in cloud computing environments. Indian Journal of Science and Technology.

Amato, F., Moscato, F., Moscato, V., Colace, F., 2018. Improving security in cloud by formal modeling of iaas resources. Future Generation Computer Systems 87, 754–764.

Anwar, S., Inayat, Z., Zolkipli, M.F.B., Zain, J.M., Gani, A., Anuar, N.B., Khan, M.K., Chang, V., 2017. Cross-vm cache-based side channel attacks and proposed prevention mechanisms: a survey. J. Netw. Comput. Appl. 93, 259–279. doi:10.1016/j.jnca.2017.06.001.

Arora, P., Wadhawan, R.C., Ahuja, E.S.P., 2012. Cloud computing security issues in infrastructure as a service. International journal of advanced research in computer science and software engineering 2 (1).

Asvija, B., Eswari, R., Bijoy, M.B., 2019. Security in hardware assisted virtualization for cloud computing - state of the art issues and challenges. Comput. Networks 151, 68–92. doi:10.1016/j.comnet.2019.01.013.

Bach-Nutman, M., 2020. Understanding the Top 10 deldDel deliIns deltThinspace OWASP Vulnerabilities. abs/2012.09960 https://arxiv.org/abs/2012

Bahrami, M., Singhal, M., 2015. DCCSOA: A dynamic cloud computing service-oriented architecture. In: 2015 IEEE International Conference on Information Reuse and Integration, IRI 2015, San Francisco, CA, USA, August 13–15, 2015. IEEE Computer Society, pp. 158–165. doi:10.1109/IRI.2015.33.

Barrowclough, J.P., Asif, R., 2018. Securing cloud hypervisors: a survey of the threats, vulnerabilities, and countermeasures. Secur. Commun. Networks 2018, 1681908:1–1681908:20. doi:10.1155/2018/1681908.

Basu, S., Bardhan, A., Gupta, K., Saha, P., Pal, M., Bose, M., Basu, K., Chaudhury, S., Sarkar, P., 2018. Cloud computing security challenges & solutions-a survey. In: IEEE 8th Annual Computing and Communication Workshop and Conference, CCWC 2018, Las Vegas, NV, USA, January 8–10, 2018. IEEE, pp. 347–356. doi:10.1109/CCWC.2018.8301700.

Bauman, E., Ayoade, G., Lin, Z., 2015. A survey on hypervisor-based monitoring: approaches, applications, and evolutions. ACM Comput. Surv. 48 (1), 10:1–10:33. doi:10.1145/2775111.

Becker, S., Brataas, G., Cecowski, M., Huljenic, D., Lehrig, S., Stupar, I., 2017. Introduction. In: Becker, S., Brataas, G., Lehrig, S. (Eds.), Engineering Scalable, Elastic, and Cost-Efficient Cloud Computing Applications - The CloudScale Method. Springer, pp. 3–21. doi:10.1007/978-3-319-54286-7_1.

Bohn, R.B., Messina, J., Liu, F., Tong, J., Mao, J., 2011. NIST cloud computing reference architecture. In: World Congress on Services, SERVICES 2011, Washington, DC, USA, July 4–9, 2011. IEEE Computer Society, pp. 594–596. doi:10.1109/SERVICES.2011.105.

Bouayad, A., Blilat, A., Mejhed, N.E.H., Ghazi, M.E., 2012. Cloud computing: Security challenges. In: 2012 Colloquium in Information Science and Technology, CIST 2012, Fez, Morocco, October 22–24, 2012. IEEE, pp. 26–31. doi:10.1109/CIST.2012.6388058.

Cabuk, S., Dalton, C.I., Edwards, A., Fischer, A., 2008. A comparative study on secure network virtualization. HP Laboratories.

Chawki, E.B., Ahmed, A., Zakariae, T., 2018. Iaas cloud model security issues on behalf cloud provider and user security behaviors. In: Yasar, A., Shakshuki, E.M. (Eds.), The 15th International Conference on Mobile Systems and Pervasive Computing (MobiSPC 2018) / The 13th International Conference on Future Networks and Communications (FNC-2018) / Affiliated Workshops, Gran Canaria, Spain, August 13–15, 2018. Elsevier, pp. 328–333. doi:10.1016/j.procs.2018.07.180.

Chouhan, P.K., Yao, F., Sezer, S., 2015. Software as a service: Understanding security issues. In: 2015 Science and Information Conference (SAI). IEEE, pp. 162–170.

Cook, B., 2018. Formal reasoning about the security of amazon web services. In: Chockler, H., Weissenbacher, G. (Eds.), Computer Aided Verification - 30th International Conference, CAV 2018, Held as Part of the Federated Logic Conference, FloC 2018, Oxford, UK, July 14–17, 2018, Proceedings, Part I. Springer, pp. 38–47. doi:10.1007/978-3-319-96145-3_3.

Coppolino, L., D'Antonio, S., Mazzeo, G., Romano, L., 2017. Cloud security: emerging threats and current solutions. Comput. Electr. Eng. 59, 126–140. doi:10.1016/j.compeleceng.2016.03.004.

Deshmukh, R.V., Devadkar, K.K., 2015. Understanding ddos attack & its effect in cloud environment. Procedia Comput Sci 49, 202–210.
Diaby, T., Rad, B.B., 2017. Cloud computing: a review of the concepts and deployment models. International Journal of Information Technology and Computer Science 9 (6), 50–58.
Fernandes, D.A.B., Soares, L.F.B., Gomes, J.V.P., Freire, M.M., Inácio, P.R.M., 2014. Security issues in cloud environments: a survey. Int. J. Inf. Sec. 13 (2), 113–170. doi:10.1007/s10207-013-0208-7.
Flexera, 2020. Flexera 2020 state of the cloud report. Applied Computing and Informatics.
Freet, D., Agrawal, R., John, S., Walker, J.J., 2015. Cloud forensics challenges from a service model standpoint: IaaS, PaaS and SaaS. In: Chbeir, R., Manolopoulos, Y., Mammana, V.P., Modena, E.A., Traina, A.J.M., Filho, O.S.S., Badr, Y., Andrès, F. (Eds.), Proceedings of the 7th International Conference on Management of Computational and Collective Intelligence in Digital EcoSystems, Caraguatatuba, Brazil, October 25, - 29, 2015. ACM, pp. 148–155. doi:10.1145/2857218.2857253.
George Amalarethinam, D., Rajakumari, S., 2019. A Survey on Security Challenges in Cloud Computing.
Ghobaei-Arani, M., Jabbehdari, S., Pourmina, M.A., 2018. An autonomic resource provisioning approach for service-based cloud applications: a hybrid approach. Future Gener. Comput. Syst. 78, 191–210. doi:10.1016/j.future.2017.02.022.
Gonzales, D., Kaplan, J.M., Saltzman, E., Winkelman, Z., Woods, D., 2017. Cloud-trust - a security assessment model for infrastructure as a service (iaas) clouds. IEEE Trans. Cloud Comput. 5 (3), 523–536. doi:10.1109/TCC.2015.2415794.
Grobauer, B., Walloschek, T., Stöcker, E., 2011. Understanding cloud computing vulnerabilities. IEEE Secur. Priv. 9 (2), 50–57. doi:10.1109/MSP.2010.115.
Guerbouj, S.S.E., Gharsellaoui, H., Bouamama, S., 2019. A comprehensive survey on privacy and security issues in cloud computing, internet of things and cloud of things. Int. J. Serv. Sci. Manag. Eng. Technol. 10 (3), 32–44. doi:10.4018/IJSSMET.2019070103.
Halboob, W., Abbas, H., Haouam, K., Yaseen, A., 2014. Dynamically changing service level agreements (slas) management in cloud computing. In: Huang, D., Jo, K., Wang, L. (Eds.), Intelligent Computing Methodologies - 10th International Conference, ICIC 2014, Taiyuan, China, August 3–6, 2014. Proceedings. Springer, pp. 434–443. doi:10.1007/978-3-319-09339-0_44.
Hashizume, K., Rosado, D.G., Fernández-Medina, E., Fernández, E.B., 2013. An analysis of security issues for cloud computing. J. Internet Serv. Appl. 4 (1), 5:1–5:13. doi:10.1186/1869-0238-4-5.
Hay, B., Nance, K.L., Bishop, M., 2011. Storm clouds rising: Security challenges for iaas cloud computing. In: 44th Hawaii International International Conference on Systems Science (HICSS-44 2011), Proceedings, 4–7 January 2011, Koloa, Kauai, HI, USA. IEEE Computer Society, pp. 1–7. doi:10.1109/HICSS.2011.386.
Huang, W., Ganjali, A., Kim, B.H., Oh, S., Lie, D., 2015. The state of public infrastructure-as-a-service cloud security. ACM Comput. Surv. 47 (4), 68:1–68:31. doi:10.1145/2767181.
Hwang, K., Bai, X., Shi, Y., Li, M., Chen, W., Wu, Y., 2016. Cloud performance modeling with benchmark evaluation of elastic scaling strategies. IEEE Trans. Parallel Distributed Syst. 27 (1), 130–143. doi:10.1109/TPDS.2015.2398438.
IBM Cloud Education, 2021. IaaS vs PaaS vs SaaS, understand and compare the three most popular cloud computing service models.
Isharufe, W., Jaafar, F., Butakov, S., 2020. Study of security issues in platform-as-a-service (PAAS) cloud model. In: 2020 International Conference on Electrical, Communication, and Computer Engineering (ICECCE). IEEE, pp. 1–6.
Jabir, R.M., Khanji, S.I.R., Ahmad, L.A., Alfandi, O., Said, H., 2016. Analysis of cloud computing attacks and countermeasures. In: 2016 18th International Conference on Advanced Communication Technology (ICACT). IEEE, pp. 117–123.
Kamongi, P., Kotikela, S., Kavi, K., Gomathisankaran, M., Singhal, A., 2013. VULCAN: vulnerability assessment framework for cloud computing. In: IEEE 7th International Conference on Software Security and Reliability, SERE 2013, Gaithersburg, MD, USA, June 18–20, 2013. IEEE, pp. 218–226. doi:10.1109/SERE.2013.31.
Kaur, A., Raj, G., Yadav, S., Choudhury, T., 2018. Performance evaluation of aws and ibm cloud platforms for security mechanism. In: 2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS). IEEE, pp. 516–520.
Khalaf, B.A., Mostafa, S., Mustapha, A., Ismaila, A., Mahmoud, M., Jubaira, M.A., Hassan, M., 2019. A simulation study of syn flood attack in cloud computing environment. AUS journal 26 (1), 188–197.
Khan, M.A., 2016. A survey of security issues for cloud computing. J. Netw. Comput. Appl. 71, 11–29. doi:10.1016/j.jnca.2016.05.010.
Khan, N., Al-Yasiri, A., 2016. Identifying cloud security threats to strengthen cloud computing adoption framework. In: Shakshuki, E.M. (Ed.), The 11th International Conference on Future Networks and Communications (FNC 2016) / The 13th International Conference on Mobile Systems and Pervasive Computing (MobiSPC 2016) / Affiliated Workshops, August 15–18, 2016, Montreal, Quebec, Canada. Elsevier, pp. 485–490. doi:10.1016/j.procs.2016.08.075.
Kim, D., Vouk, M.A., 2014. A survey of common security vulnerabilities and corresponding countermeasures for saas. In: 2014 IEEE GLOBECOM Workshops, Austin, TX, USA, December 8–12, 2014. IEEE, pp. 59–63. doi:10.1109/GLOCOMW.2014.7063386.
Krishna, B.H., Kiran, S., Murali, G., Reddy, R.P.K., 2016. Security issues in service model of cloud computing environment. Procedia Comput Sci 87, 246–251.
Kritikos, K., Kirkham, T., Kryza, B., Massonet, P., 2017. Towards a security-enhanced paas platform for multi-cloud applications. Future Gener. Comput. Syst. 67, 206–226. doi:10.1016/j.future.2016.10.008.
Kumar, R., Goyal, R., 2019. On cloud security requirements, threats, vulnerabilities and countermeasures: a survey. Comput. Sci. Rev. 33, 1–48. doi:10.1016/j.cosrev.2019.05.002.
Li, J., 2020. Vulnerabilities Mapping Based on OWASP-SANS: a Survey for Static Application Security Testing (SAST). abs/2004.03216 https://arxiv.org/abs/2004.03216
Li, X., Zhou, L., Shi, Y., Guo, Y., 2010. A trusted computing environment model in cloud architecture. In: International Conference on Machine Learning and Cybernetics, ICMLC 2010, Qingdao, China, July 11–14, 2010, Proceedings. IEEE, pp. 2843–2848. doi:10.1109/ICMLC.2010.5580769.
Liu, Y., Sun, Y.L., Ryoo, J., Rizvi, S., Vasilakos, A.V., 2015. A survey of security and privacy challenges in cloud computing: solutions and future directions. J. Comput. Sci. Eng. 9 (3). doi:10.5626/JCSE.2015.9.3.119.
Loukis, E.N., Janssen, M., Mintchev, I., 2019. Determinants of software-as-a-service benefits and impact on firm performance. Decis. Support Syst. 117, 38–47. doi:10.1016/j.dss.2018.12.005.
Machado, G.S., Hausheer, D., Stiller, B., 2009. Considerations on the interoperability of and between cloud computing standards. 27th open Grid Forum (OGF27), G2C-Net Workshop: From Grid to Cloud Networks.
Malik, M.I., Wani, S.H., Rashid, A., 2018. Cloud computing-technologies. International Journal of Advanced Research in Computer Science 9 (2).
Manvi, S.S., Shyam, G.K., 2014. Resource management for infrastructure as a service (iaas) in cloud computing: a survey. J. Netw. Comput. Appl. 41, 424–440. doi:10.1016/j.jnca.2013.10.004.
McKay, K., Cooper, D., 2018. Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations (2nd Draft). Technical Report. National Institute of Standards and Technology.
Modi, C., Patel, D.R., Borisaniya, B., Patel, A., Rajarajan, M., 2013. A survey on security issues and solutions at different layers of cloud computing. J. Supercomput. 63 (2), 561–592. doi:10.1007/s11227-012-0831-5.
Pancholi, V.R., Patel, B.P., 2016. Enhancement of cloud computing security with secure data storage using aes. International Journal for Innovative Research in Science and Technology 2 (9), 18–21.
Panda, D.R., Behera, S.K., Jena, D., 2021. A survey on cloud computing security issues, attacks and countermeasures. In: Advances in Machine Learning and Computational Intelligence. Springer, pp. 513–524.
Pham, V.V.H., Liu, X., Zheng, X., Fu, M., Deshpande, S.V., Xia, W., Zhou, R., Abdelrazek, M., 2017. Paas - black or white: an investigation into software development model for building retail industry saas. In: Uchitel, S., Orso, A., Robillard, M.P. (Eds.), Proceedings of the 39th International Conference on Software Engineering, ICSE 2017, Buenos Aires, Argentina, May 20–28, 2017 - Companion Volume. IEEE Computer Society, pp. 285–287. doi:10.1109/ICSE-C.2017.57.
Rajaraman, V., 2014. Cloud computing. Resonance 19 (3), 242–258.
Rakotondravony, N., Taubmann, B., Mandarawi, W., Weishäupl, E., Xu, P., Kolosnjaji, B., Protsenko, M., de Meer, H., Reiser, H.P., 2017. Classifying malware attacks in iaas cloud environments. J. Cloud Comput. 6, 26. doi:10.1186/s13677-017-0098-8.
Rashid, A., Chaturvedi, A., 2019. Virtualization and its role in cloud computing environment. International Journal of Computer Sciences and Engineering 7 (4), 1131–1136.
Rodero-Merino, L., Vaquero, L.M., Caron, E., Muresan, A., Desprez, F., 2012. Building safe paas clouds: a survey on security in multitenant software platforms. Comput. Secur. 31 (1), 96–108. doi:10.1016/j.cose.2011.10.006.
e Rubab, K., Azhar, T., Anwar, M., Majeed, S., 2020. Security threats in cloud computing: trend and challenges. International Journal of Computing and Communication Networks 2 (1), 29–35.
Sandikkaya, M.T., Harmanci, A.E., 2012. Security problems of platform-as-a-service (PaaS) clouds and practical solutions to the problems. In: IEEE 31st Symposium on Reliable Distributed Systems, SRDS 2012, Irvine, CA, USA, October 8–11, 2012. IEEE Computer Society, pp. 463–468. doi:10.1109/SRDS.2012.84.
Selvamani, K., Jayanthi, S., 2015. A review on cloud data security and its mitigation techniques. Procedia Comput Sci 48, 347–352.
Sengupta, S., Kaulgud, V.S., Sharma, V.S., 2011. Cloud computing security-trends and research directions. In: World Congress on Services, SERVICES 2011, Washington, DC, USA, July 4–9, 2011. IEEE Computer Society, pp. 524–531. doi:10.1109/SERVICES.2011.20.
Shaikh, A.H., Meshram, B., 2020. Security issues in cloud computing. In: Intelligent Computing and Networking. Springer, pp. 63–77.
Shyam, G.K., Theja, R.S.S., 2021. A survey on resolving security issues in saas through software defined networks. Int. J. Grid Util. Comput. 12 (1), 1–14. doi:10.1504/IJGUC.2021.112475.
Singh, A., Chatterjee, K., 2017. Cloud security issues and challenges: a survey. J. Netw. Comput. Appl. 79, 88–115. doi:10.1016/j.jnca.2016.11.027.
Singh, A., et al., 2019. Security concerns and countermeasures in cloud computing: a qualitative analysis. International Journal of Information Technology 11 (4), 683–690.
Singh, S., Jeong, Y., Park, J.H., 2016. A survey on cloud computing security: issues, threats, and solutions. J. Netw. Comput. Appl. 75, 200–222. doi:10.1016/j.jnca.2016.09.002.
Subashini, S., Kavitha, V., 2011. A survey on security issues in service delivery models of cloud computing. J. Netw. Comput. Appl. 34 (1), 1–11. doi:10.1016/j.jnca.2010.07.006.
Tabrizchi, H., Rafsanjani, M.K., 2020. A survey on security challenges in cloud computing: issues, threats, and solutions. J. Supercomput. 76 (12), 9493–9532. doi:10.1007/s11227-020-03213-1.

Toraskar, T.R., Borse, Y., 2018. Implementation of cloud computing service delivery models (IaaS, PaaS) by aws and microsoft azure: a survey. Int J Comput Appl 975, 8887.
Vaquero, L.M., Rodero-Merino, L., Morán, D., 2011. Locking the sky: a survey on iaas cloud security. Computing 91 (1), 93–118. doi:10.1007/s00607-010-0140-x.
Varghese, B., Buyya, R., 2017. Next Generation Cloud Computing: New Trends and Research Directions. abs/1707.07452, http://arxiv.org/abs/1707.07452
Vasiljeva, T., Shaikhulina, S., Kreslins, K., 2017. Cloud computing: business perspectives, benefits and challenges for small and medium enterprises (case of latvia). Procedia Eng 178, 443–451.
Verma, A., Kaushal, S., 2011. Cloud computing security issues and challenges: A survey. In: Abraham, A., Mauri, J.L., Buford, J.F., Suzuki, J., Thampi, S.M. (Eds.), Advances in Computing and Communications - First International Conference, ACC 2011, Kochi, India, July 22–24, 2011, Proceedings, Part IV. Springer, pp. 445–454. doi:10.1007/978-3-642-22726-4_46.
Verma, G., Adhikari, S., 2020. Cloud computing security issues: a stakeholder's perspective. SN Computer Science 1 (6), 1–8.
Wang, S., Liu, Z., Sun, Q., Zou, H., Yang, F., 2014. Towards an accurate evaluation of quality of cloud service in service-oriented cloud computing. J. Intell. Manuf. 25 (2), 283–291. doi:10.1007/s10845-012-0661-6.
Weinman, J., 2018. The economics of pay-per-use pricing. IEEE Cloud Comput. 5 (5), 101. doi:10.1109/MCC.2018.053711671.
Yan, Q., Huang, W., Luo, X., Gong, Q., Yu, F.R., 2018. A multi-level ddos mitigation framework for the industrial internet of things. IEEE Commun. Mag. 56 (2), 30–36. doi:10.1109/MCOM.2018.1700621.
Zhang, S., Zhang, X., Ou, X., 2014. After we knew it: empirical study and modeling of cost-effectiveness of exploiting prevalent known vulnerabilities across iaas cloud. In: Moriai, S., Jaeger, T., Sakurai, K. (Eds.), 9th ACM Symposium on Information, Computer and Communications Security, ASIA CCS '14, Kyoto, Japan - June 03 - 06, 2014. ACM, pp. 317–328. doi:10.1145/2590296.2590300.
Zissis, D., Lekkas, D., 2012. Addressing cloud computing security issues. Future Gener. Comput. Syst. 28 (3), 583–592. doi:10.1016/j.future.2010.12.006.

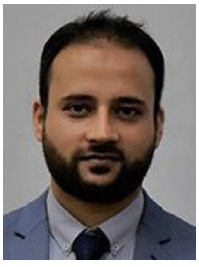
Fatemeh Khoda Parast is currently a Ph.D. student at Faculty of Computer Science, University of New Brunswick (UNB). Her Ph.D. research ispartially funded the Lockheed Martin Cybersecurity Research Fund (LM-CRF) and MITACS. She researches the security of large storage management systems in cloud environments. She has accomplished numerous academic and industrial software projects.

Chandni Sindhav holds masters degree in a Computer Science from Gujrat Technological University, Gujrat, India. Currently pursuing Master of Applied Cybersecurity at University of New Brunswick, Fredericton, Canada, She has been certified by the AWS cloud as Certified Cloud practitioner. She participated has been participated in Agorize "AI for future Business Challenge 2020" and was selected in top 25 teams out of 123 teams across Canada. Currently she is working on Android malware classification and developing framework for assessing CVE usage by malware. She has been awarded tuition fee scholarship from Bell Canada.

Seema Balakrishna Nikam holds a B.E. in Information Science & Engineering from Atria Institute of Technology, Bengaluru, India a Master of Computer Science from the University of New Brunswick, Fredericton, Canada. She has two years of experience as a Software Developer and her work and interests are based on the optimization of algorithms, automation scripts, AI & ML, Cloud Computing, and cloud & software security.

Hadiseh Izadi Yekta, received her B.Sc. in Computer Engineering - Software Engineering from the University of Science and Technology, Iran. She is currently pursuing a Master of Computer Science at the University of New Brunswick, Fredericton, Canada. Her primary research interests include privacy-preserving, mobile edge computing, and federated learning.

Dr. Kenneth Kent is a Professor in the Faculty of Computer Science since 2002. He is also the Director for the Centre for Advanced Studies - Atlantic. Dr. Kent has supervised over 70 graduate students, postdocs, and researchers and has published more than 150 refereed journal articles, conference papers and patents. Dr. Kent is an Honorary Professor at Hochschule Bonn-Rhein-Sieg where he is also involved in research through the Institute for Visual Computing and Department of Computer Science.



Dr. Saqib Hakak is an assistant professor at the Canadian Institute for Cybersecurity (CIC), Faculty of Computer Science, University of New Brunswick (UNB). Having more than 5+ years of industrial and academic experience, he has received the number of Gold/Silver awards in international innovation competitions and is serving as the technical committee member/reviewer of several reputed conference/journal venues.