International Journal of Computational Intelligence in Control

# Challenges in Multi-Cloud and Benefits from Leveraging Cloud Native Strategy to Digital Transformation of Business

**Muhammad Iqbal, Muhammad Ijaz Khan, Asia Zaman, Muhammad Shahjahan, Muhammad Farhan, Muhammad Shahjahan, Rizwan Ullah, Muhammad Mustafa, Asim Khalil**

malikiqbalprince1@gmail.com
ijaz171@gmail.com
aasiazaman123@gmail.com
shahjhnmughal@gmail.com
muhammmadfarhan01@gmail.com
rizwanullah99100@gmail.com
attleramustafa@gmail.com
asimkhalil729@gmail.com

**Institute of Computing and Information TechnologyGomal University, Pakistan.**

## Abstract:

For many organizations, the use of cloud computing increases rapidly. Cloud computing offers many advantages in consideration of the relatively inexpensive and accessibility of data. Making sure cloud data security is a significant factor in the cloud computing world, as users frequently store confidential information in cloud storage providers, but they may not be trusted by such services. Cloud computing, a modern model for scalable and cost-effective computing, has been discussed as a potential alternative for sustainable business Technology. Cloud computing also provides innovative cutting-edge native cloud technologies to build micro services-composed applications. This paper looks at the transformation approaches and procedures in a cloud computing world. Currently, there has arisen a trend toward "inter-clouds" in other terms, "multi-clouds," or it is called "cloud-of-clouds." The ability to run cloud-native applications will bring tremendous growth and profitability to the enterprise. Cloud-native security has a lot of influence in having business results. Researchers studied the concepts of cloud-native application architecture, service providers of public clouds, and business technology standards. This paper addresses

the idea of multi-cloud computing and the numerous problems of multi-cloud computing. This paper also discusses cloud-native security and its architecture. the new approaches are highlighted in this paper using cloud-native architecture patterns to develop and deploy Security-as-a-Service (SecaaS) applications. The latest SecaaS solutions do not tackle the imminent danger to computer systems and applications effectively. Cloud-native design patterns overcome this problem by allowing for certain attributes such as significant optimization and durability through the mixture of microservice designs and cloud-focused interface design.

**Keywords:**Multi-cloud computing;Challenges,Cloud native, SaaS, Design patterns, Architecture, SecaaS, Digital Transformation, Cloud Native Applications

## 1. INTRODUCTION

In many organizations, the use of cloud computing has rapidly increased. Subashini and Kavitha[1] claim that small and medium-sized enterprises utilize cloud computing services for just a variety of reasons, particularly that they provide quick access to their programs and decrease the cost of services. Cloud companies should take large and immediate primary responsibility to resolve security and privacy issues. Trying to deal with "single cloud" suppliers has become less popular with consumers because of possible issues like failure of network connectivity and the prospect of insider threats within the cloud environment. There's been a move in recent decades forward into "multi-clouds," "intercloud" or "data center-of-clouds." [2]Frameworks currently seek to exploit cloud infrastructure through the multi-provider use of heterogeneous devices. This is in comparison to how assets have customarily been used from either a cloud-based service or database servers. This results in the emergence of new data centers.Digitization innovations and instruments have become widespread and compelling. Nations around the world compete with each other to observe and learn digitization systems, channels, trends, goods, activities, and processes to be intelligently reactive and attentive to their people. [3]All types of business institutions and corporations are gleefully planning in their operational activities, packages, and outcomes to be beautifully digitized. Modern beings are informed of the profound effects of digitization innovations. IT organizations, likewise, are keen to develop an Everton of digitization-enabled products and solutions. The strategies as well as the operational consequences of digitization are largely informed by research institutions, entrepreneurs, and persons. Digitization movement 's knowledge and expressiveness are actually on the rise with an increased knowledge of the organization, scientific, and policy side of digitization technologies such as cloud, business intelligence (big, real-time, broadcasting, and artificial intelligence), business intelligence, Web 2.0 (social internet) and internet of things (Semantic Website), artificial intelligence ( AI). Perhaps not our machines, but also certain daily gadgets, portable devices, smartwatches, medical devices, aerial planes, robotic arms, electronic goods, heavy weaponry, production machines, household items and cutlery, personal smart devices, and neural implants including such detectors, solenoid valves are all routinely linked to each other and also to remote software programs. Such sequence diagrams combine microservices with process models based on the cloud to create a Cloud-Native Application (CNA). The performance of companies like Netflix, Amazon, Uber and is typically due to their CNA deployment[4]. We think the application of these trends to the applications of SecaaS could enhance the transportation of security services. Nonetheless, implementing cloud-native design principles is a complicated procedure, requiring thorough planning and no excellently-defined steps[8] are needed.

In this paper, there are a total of six sections. In section2 the concept of multi-cloud computing and the challenges in multi-cloud computing are described in full detail. In section 3 native cloud and its architecture is briefly discussed. The different applications of cloud-native are discussed in section 4. Section 5 has the cloud-native design patterns and SaaS discussion. Chapter 6 is about the digital transformation of business. At the last in section 7, there is the conclusion.

## 2. THE CONCEPT OF MULTI-CLOUD COMPUTING

The multi-cloud is the combination of two or more distinct cloud computing services from various cloud vendors. An area with multi-clouds may become all-private, all-public, or a mixture of both. Firms use multi-cloud systems to spread assets in computation and reduce the risk of interruption and destruction of information. They may also improve an enterprise's computing storage and processing. Cloud developments in recent years as a result in a transition from private singular-user clouds to multi-tenant public clouds and community cloud which is known as hybrid cloud to a centralized network that leverages multiple computing environments such as the public cloud and private cloud. [2] The term 'multi-cloud' is similar to those used in the term "Intercloud" or "fog-of-clouds" released by Vukolic[5]. Such terms suggest cloud computing is not supposed to end in a cloud environment. Utilizing their example, a cloudy sky combines various cloud colors and shapes that lead to specific deployment and operational dependencies. The latest research has centered on the multi-cloud environment[6],[7] that governs multiple clouds and prevents reliance on individuals' clouds. Cachin et al. [5] recognize two levels in the multi-cloud atmosphere: the internal cloud is the lower part, while the cross-cloud is the second level.

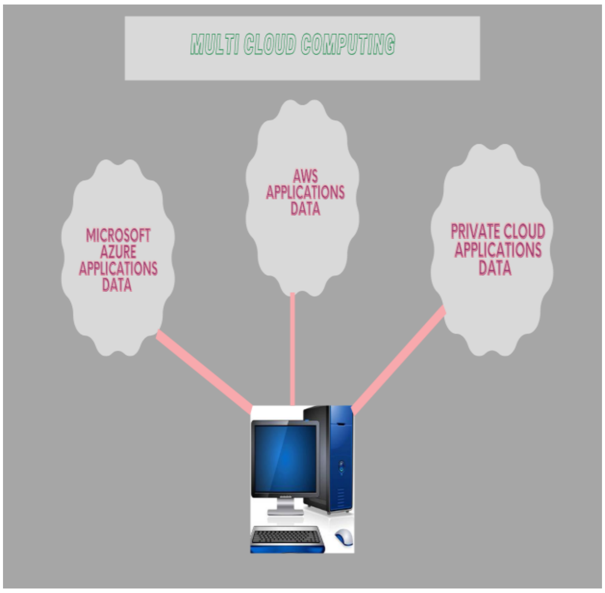
MULTI CLOUD COMPUTING
MICROSOFT AZURE APPLICATIONS DATA
AWS APPLICATIONS DATA
PRIVATE CLOUD APPLICATIONS DATA

**Figure 1.** Multi-cloud computing

With the introduction and expressiveness of a variety of powerful cloud implementation technology and products, the Cloud idea, which is being praised as the most introduction of new technologies to concisely allow the dreamt digital transformation, is getting a ton of traction these times. The cloudification era had also definitely decided to kick in strongly and is on the right path to becoming a prominent and vital component and a single aspect of the amusing and intelligent IT era. Business enterprises, IT groups, and cloud service providers (CSPs) are working together to have a range of company-specific and common public clouds by using excellentlydefined and built cloud processes, products, and trends. A host of technical solutions identify and overcome the high-priority constraints and concerns. The respected industry analysts and consulting firms have estimated that by 2020, about 80% of the company systems currently operating will be officially overhauled and migrated to multiple clouds. Clouds are emerging as one of the most structured and organized IT ecosystems for an increasing multitude of private, social, and technical programs to reside and operate. The IT area fermentation patterns tell immediately us that the recognition and acceptance levels of the Cloud are quickly going to pick up. One positive change is that companies around the world are linking up their own IT areas / classical storage arrays with one or even more public clouds to have the cloud paradigm's different and clear advantages. Thus, a close connection between typical IT and new IT is formed and maintained, which is predominantly defined by the business model Cloud. This dimension is called Hybrid IT. This fresh and integrated operations strategy delivers the agility, flexibility, and creativity necessary to propel any company forward. A hybrid IT model allows IT to achieve important business objectives:

- Strengthen customer engagement, market shares, satisfaction, and trust.
- Develop new fields of productive and differentiated production.
- Rising risk and lower production costs by speeding up and improving performance.

Hybrid Clouds are yet another thrilling occurrence., it is called the hybrid cloud to integrate private clouds with one or even more public clouds. With this modern trend, some technically sound benefits are being bandied around. In simple, we're hiking rapidly and attempting toward a multi-cloud era. With the distributed computing paradigm being predicted as a feasible and traditional platform for the future age of information infrastructure and intelligent applications, there is no further gap in adopting hybrid cloud innovations. With standard cloud management and orchestration services and products available on the marketplace, hybrid cloud activities including and managing many, geographically dispersed and fragmented clouds are on board. The position and obligation of multi-cloud environments are set to increase in the coming days for the establishment of electronically disrupted and changed companies and communities.

### 2.1 Challenges in Multi-cloud computing.

The word "multi-clouds" is parallel to the words "inter-clouds" or "cloud-of-clouds" said Vukolic[5]. These normally computed cloud computing is not supposed to end in a single cloud. Using their example, a cloudy sky combines various cloud colors and patterns that lead to specific implementation and operational occupations. Recent studies have focused on the environment of multi-cloud [8],[6] that controls multiple clouds and prevents heavy reliance on just about anyone cloud. Cachin et al.[6] recognize two layers within the multi-cloud atmosphere: the inner cloud is the lower part, whereas the inter-cloud is the second level. The Byzantine attitude to criticism takes its location in the inter-cloud. We will sum up the preceding Byzantine procedures throughout the last three decades initially. In cloud computing, some software or hardware defects are classified as Byzantine defects which typically apply to unprofessional conduct and sensitivity to invasion. It also contains random and malfunctioning faults[5]. Since the first implementation, much interest has been directed to Byzantine fault-tolerant (BFT). While much attention has been paid to BFT science, it still struggles with the constraints of functional adoption and appears marginal in integrated systems[5]. BFT 's association with cloud computing has been studied, yet many say that it has been deemed one of the main positions on the distributed network agenda in recent years. In addition, several characterize BFT to be of pure "academic interest" for a service of the cloud [9]. This loss of enthusiasm in BFT is quite distinct from the amount of enthusiasm seen in processes used during large-scale systems for accepting crash weaknesses. Reasons that decrease BFT adoption include problems in designing, implementing, or understanding BFT protocols[5], for example. BFT protocols, as already mentioned, really aren't suitable for individual clouds. Vukolic[5] argues that one of BFT 's drawbacks for the inner cloud is that BFT demands a high degree of independence from failure, just as all fault-tolerant routers do[10]. If Byzantine failure occurs in a single node in the cloud, having a different operating system, separate application, and different hardware is fair to ensure that such breakdown does not propagate to other domains in the very same cloud. Furthermore, if an attack occurs on a specific cloud, it may allow attackers to subvert the specific transportation system within the cloud[5].

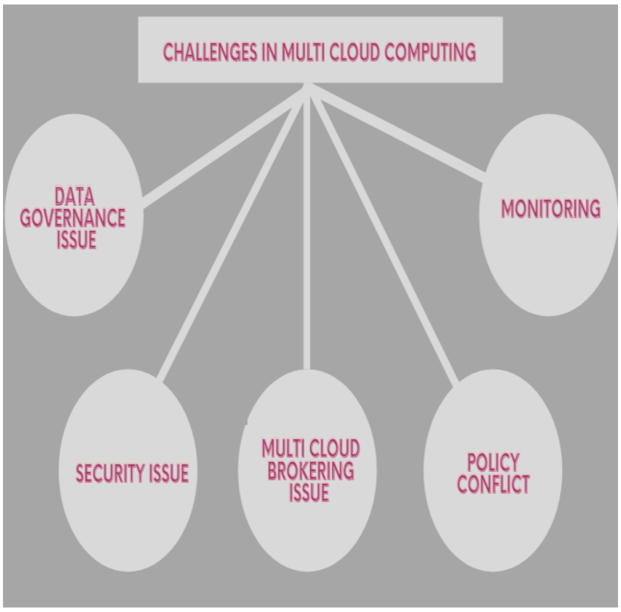
CHALLENGES IN MULTI CLOUD COMPUTING
DATA GOVERNANCE ISSUE
MONITORING
SECURITY ISSUE
MULTI CLOUD BROKERING ISSUE
POLICY CONFLICT

**Figure 2.** Challenges in Multi-cloud computing

### 2.2 Data Governance and Compliance:

Multiple clouds and cloud service circumstances can give excellent versatility and conformance in distinct capitals of the world; even then, this will not come instantly. The main obstacle is to recognize where the data practically dwells this problem may be more challenging for small to mid-sized businesses. Provided the usability of the multi-cloud environment, making mistakes and ending up trying to run an implementation in an unauthorized atmosphere might be simple. There are many dozens of guidelines set, particularly, especially GDPR, which may result in heavy penalties being collected when broken; in an attempt to stem this inherent problem, IT supervisors may need to start preparing the appropriate equipment to gain visibility to track the magnitude of their burdensome regulations [11].

### 2.3 Security issues.

Experts and industry professionals have identified many security concerns in cloud computing, namely separation management, data disclosure and privacy, virtual OS protection, confidence and enforcement, and project assurance.8 Throughout decentralized management and exchange through multiple clouds, new security concerns are emerging. Issues relating to trust, regulation, and privacy are particular issues throughout multi-cloud computing environments. Settling confidence and stable coordination. As with other IT structures, protection in clouds is highly dependent on creating relationships based on trust between the individuals involved. The need for trust exists when a customer waives the exclusive protection of the confidentiality and protection of his properties of cloud services provider (CSP). Doing so introduces the properties of a company to new threats otherwise reduced or unpreventable within an organizational structure. These dangers involve informant potential threats, weakening data property rights, syntactic emotional issues in polymer cloud services of third-party providers, and weakened system security oversight.8 A consumer should impart a high degree of satisfaction on a CSP concerning its power to deliver effective asset-securing tools and procedures. Therefore, by using cloud-based services a customer should be able to embrace the higher rates of threat. Using proxy takes the trust barrier one step even farther: customers and CSPs now have to create a mutual trust with proxy, including acknowledging security, reliability, quality, and assurances of disaster recovery from a proxy. In addition, CSPs going to respond to queries for service that a server helps make on behalf of the company or other CSPs should believe the proxy to intervene on behalf of the ability to request organization legally. For example, the proxy should provide a secure computing network to secure rest and in transit and location of data that prohibits malicious programs from assuming charge and damaging confidential user and cloud application information. They also must ensure the confidentiality of the data all through transmitting via the proxy system, potentially through using requirements including the transmission control protocol (TCP Protocol. Additionally, customers, clouds, and proxy servers have to incorporate a mechanism to ensure a secure committee, which involves: on-the-fly contracts. Deferring to a proxy shall, on the fly, establish an actual statement between the delegator and the proxy allowing the proxy to intervene on behalf of the delegate. Technologies for delegating to a proxy should include processes that constrain the actions of the proxy, such as information and access to resources, to meet the limitations indicated by the strategic thinker.

### 2.4 Multi-cloud Brokering:

Brokerage technologies and products in the period of linked and federalized Clouds are quite essential. These cloud service vendors conduct connectedness, facilitation, and other enhancement and facilitation functionality. To create a provided immediate among public and private clouds, there are many adaptors, connectors, operators, and other alternatives. Bridge alternatives are designed to establish excellent access between public Clouds. Therefore, with several clouds and providers providing specific SLAs, cloud brokerage positions and responsibilities are expected to intensify in the upcoming days. Highly developed CMPs come with brokerage tools and turbochargers.

### 2.5 Policy heterogeneity and conflicts:

Diverse security protocols can be the origin of issues cause resulting in security vulnerabilities when proxies allow vibrant collaboration between different CSPs. VPs must evaluate and protect themselves against these violations. Although established regulatory assessment mechanisms can verify personal domain initiatives, security breaches can happen easily all through integration.12 services provided can keep driving vibrant, temporary, and costly interaction of different application components in multi-cloud collaborative efforts utilizing proxy servers. Thus, the strategy interoperability assignments of a VPN service must overcome problems such as semantic heterogeneity, safe integration, and strategic planning of communication links. Designing security controls for multi-cloud communication must allow proxies to handle carefully while maintaining that effective integration does not take place They contribute to breaches of security. Public policy usually includes real estate confirmation and context information, and also an assessment of policy edition distinctions. A proxy shall deal with many registration services from different locations and also proxies for a cooperative service. That requires different proxies to undertake the implementation and decomposing of policies regionally. The integration of policies seeks to produce distributed access agreements by each involved party in a cooperative effort. A strategy strategic alignment for patrols between the various apartment cooperation must deal with possible collisions and resolve problems methodically**.**

### 2.6 Distributed Denial of Service (DDoS):

Distributed Denial of Service (DDoS) is becoming an excellently-known security problem in the cloud environment and poses a possible hazard of significant negative effects on decentralized clouds[12]. Malicious users and attackers aim to leverage the weaknesses in cloud infrastructure undergirded by virtualization software, processes for provisioning, and multi-tenancy. The assault usually absolves other customers of funds and processing power on the cloud, thus causing them to accrue more financial value with less optimum design. It in turn harms cloud service provider customer confidence and cloud foster care. Large-scale attacks recorded in mass media also have both business and political motivations. Three specific frameworks are commonly proposed for resolving these assaults, including avoidance, identification, and mitigating. Even so, the creation and implementation of practical cloud-based approaches are still in their adolescence. Vulnerability is more day with the adoption of distributed cloud systems, and with more devices and applications linked to the cloud. Attacks of prevention, identification, and prevention will need to be more evolved following the cloud-enabled structure innovations.

### 2.7 Monitoring*:*

When testing several clouds, it is difficult to determine performance and security issues. Finding a good cloud surveillance tool can prevent you from getting this problem. Many other APM alternatives can support many other clouds. This gives the organization a variety of options to find the correct measurement system. Effective organizational performance for multi-clouds does, however, include making sure that the tool acknowledges how the volume of work of the cloud works. To warn you about incoming issues, the tracking tool must understand that the dual workloads are operating in separate clouds, although they are intertwined and based on each another.

## 3. CLOUD-NATIVE

Web-native computing takes full advantage of several modern technologies, namely microservices, PaaS, agile methodology, multicolored containers, CI / CD, and DevOps in specific use, "web-native" is a methodology for designing and operating applications that incorporate the benefits of the computing delivery method. "Cloud-native" about how to generate and deploy applications, and not where. As compared to an on-premises data center, it indicates that the applications live throughout the public cloud. [5] It is basic decency for cloud-based applications to be purposely established for the cloud. There's no specific meaning, although, that describes exactly what it means. Our evaluation of quality assurance in higher on cloud-native applications has started to turn out that evaluated papers have a common but disoriented comprehension. Cloud-native applications (CNA) should be designed according to the relevant CNA concepts (controlled on the automated potential opportunity for the industry infrastructure and application considering issues of transition and compatibility). Such principles allow the construction of CNA architectures with special and unique CNA characteristics (horizontal parallelization, suppleness, durability, and independent structures, to name only a few of the most widely described). Methods surrounding CNA have mostly developed mechanisms. Fehling et al. are proposing IDEAL ought to be a cloud-native implementation. It will have a separate nation, is transmitted in its natural world, is horizontally elastic, is controlled via a computerized management platform, and should be highly interconnected with its components [13]. As per Stine, cloud-based technology architectures [14] like to provide software-based solutions. Incentives Alternatives faster (speed), extra responsibility to fix-isolating, fault-tolerant, and instant (protection) recovery, enabling horizontal (rather than vertical) implementation scalability (magnitude), and eventually dealing with a wide range of (phone) systems and legacy software (customer variability). Multiple application designs and network strategies tackle these specific motivations [15]: microservices reflect the absorption of homogenous (business) structures into comprising of individual services that do "one thing well"[16]. In cloud-native software development, the main form of transport among services is through reviews and services to users' APIs (API-based collaborative project). Often those APIs implement the HTTP REST-style dependency injection with JSON but some procedures and recompilation configurations can also be used. Single architectural design installation components are designed and interrelated based on a compilation of cloud-focused trends such as the twelve-factor app collection, the power supply pattern, or cloud services patterns [13]. And eventually, agile management architectures with self-service are used to deploy and run these web services through self-contained delivery units (containers). These systems offer extra operational capacity in addition to IaaS infrastructures such as computerized as well as on-demand scalability of exploring the possibilities, managed care applications, route information and load balancing, and ass accumulation of log files and methodologies

### 3.1 Developed with best-of-breed languages and frameworks:

Each cloud-native applications service is designed with the programming language perfectly suited to the application. Implementations in the cloud are polyglot; service providers use a range of words, third-party tools, and frameworks. For instance, developers can build a WebSocket-based, real-time streaming service, formed in Node.js, while selecting Python and Flask to expose the API. The granular method of trying to develop microservice architecture allows them to select the best framework for a particular job.

### 3.2 Designed as loosely coupled microservices:

Services belonging to the very same implementation uncover each other during the runtime of the implementation. They do exist regardless of other services. When implemented fully, stretchy infrastructure and software architectures can be adjusted-out with accuracy and greater productivity.

### 3.3 Isolated from the server and operating system dependencies:

Cloud-native applications have no connection to any specific operating system or computer. They work at an elevated level of abstraction. The exception is when a microservice requires other features, like solid-state drives (SSDs) and graphics processing units (GPUs), that a group of computers can specifically provide.

### 3.4 Managed through agile DevOps processes:

That cloud-native technology connection goes via an individual development cycle, which is handled by an adaptive DevOps process. Different configuration management / continuous deployment pipelines (CI / CD) can function together to install and manage a cloud-based application.

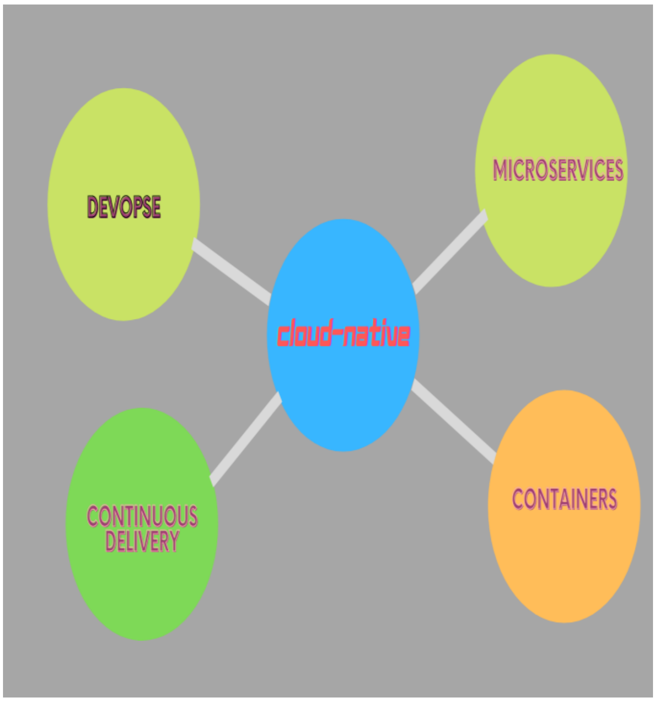

**Figure3.** Cloud-Native

### 3.5 Automated capabilities:

Applications located in the cloud can be fully automated. They go along well with the technology principle as code. Nonetheless, handling such massive and complicated applications requires a high level of optimization.

### 3.6 Cloud-Native architecture:

There are several potential solutions to the Redundant duplication issue when the distributed systems are operating in the cloud. Generally speaking, we can't manage redundancy in the cloud service because the cloud service provider manages those resources. We can, therefore, adjust how we reproduce the application stage. One easy step is to eliminate duplication at the application level. Because the data storage already offers error tolerance through replication, the application does not need to replicate for fault tolerance. The downside to this strategy is that it causes the service to lose quality. If the primary operating case, e.g. due to system failure, system malfunction, or server partition, is unattainable, then the entire production is not usable. An alternative approach is to share a common main version of the information on the cloud storage server with several copies of the database. There is a range of possible solutions to the issue of distributed redundancy when operating distributed systems in the cloud. Generally speaking, we can't manage redundancy in the cloud service because the cloud service provider manages those facilities. We can, therefore, adjust how we reproduce the application stage. One easy step is to eliminate duplication at the application level. Since the data storage already offers error tolerance through duplication, the application does not need to duplicate for fault-tolerant. The downside to this strategy is that it causes the service to lose quality. If the single working case, e.g. due to a system crash, system malfunction, or network division, is unattainable, then the entire production is not usable. An alternative approach is to share a single default copy of the information on the cloud storage system with several copies of the database We note that main-delta structures have the desired properties that can be utilized in cloud services: they have a broad publish-only primary data fragment, which is reconstructed regularly to add a bunch of deltas into a procedure known delta merge. As the information in central is read-only, it'd be able to transfer a single key for numerous incidents of a distributed service, without adding the same argument mentioned above of numerous incidents creating a common main backup of the information. It would only require coordination of the periodic delta-merge operation. On the other hand, repeated delta mergers hold deltas fairly low and erase them after the merging takes place. By using the application-level duplication method each replica will retain its delta. Delta may be stored on a disk drive, in the cloud storage service's private region, or in storage dependent on the system's setting and reliability assurances. Decide which node merges the delta to a replica collection

The key relies on the app's replication strategy. Therefore, it is crucial to understand the potentiality to actuality used in action before giving relevant data about the delta-merge technique.

## 4. CLOUD-NATIVE APPLICATIONS

It is basic decency to involuntarily build cloud-native apps for the cloud. There's no proper definition, though, that describes exactly how this works. Our analysis of literature1 on the cloud-native application has found that all analyzed papers have a specific yet subconscious interpretation. Cloud-native applications (CNA) should be designed as per the correlating CNA concepts (controlled on automated test" processes infrastructure and application, considering elements of population movement and integration). Such features allow for the construction of CNA systems with unique and desirable CNA properties (horizontal parallelization, permeability, durability, and independent systems, to mention several of the most frequently mentioned). Techniques surrounding CNA have mostly developed mechanisms. The following observations are the most important from the viewpoint of the developers Fehling et al. recommend that IDEAL ought to be a cloud-native program. It will have an independent state, is distributed in its design, is horizontally elastic, is controlled through an automated control system, and therefore should be highly interconnected with its components. And according to Stine, cloud-based application architecture provides software-based solutions are popular motivations alternatives faster (speed), further responsibility to fix-isolating, fault-tolerant, and automatic (safety) recovery, allowing horizontal (instead of vertical) device scale (magnitude), and eventually dealing with a wide variety of (mobile) systems and legacy software (client diversity). Many other application designs and network approaches tackle these specific motivations: microservices reflect the decay of centralized (company) structures into comprising of individual services that do "one element well". Within a cloud-native client application, the key form of transport among providers is through released and evanescent APIs (API-based collaborative projects). These APIs mostly adopt the Long - term sleep-style with JSON continuations, but can also use other protocols and dependency injection formats. Single architecture implementation systems are planned and integrated based on a set of cloud-focused trends such as the twelve-factor device array, the circuit pattern, or cloud services patterns. And eventually, agile management architectures with self-service are being used to deploy and run these web services through self-contained delivery units (containers). Such systems have significant additional features in addition to IaaS infrastructures such as automatic as well as on-demand application example balancing, application patient care, route information, and enables seamless, and also log and measurement accumulation. Such getting chosen us all to the concept of.

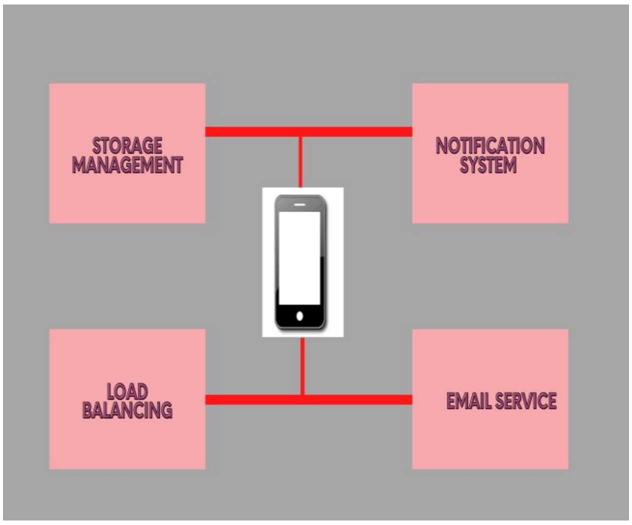

**Figure 4.**Cloud-native applications

## 5. CLOUD-NATIVE DESIGN PATTERNS TO SECURITY-AS-A-SERVICE APPLICATIONS

### 5.1 Cloud-Native Architectures Design Principles and Motivations for Migration.

Cloud technology implementation of applications is typically driven by the benefits the cloud provides. Some implementations, however, use methods specially developed for traditional information centers [17]. A classic example is a practice of bundling application server stacks into cloud VM, e.g. a three-tiered web application. In terms of capacity, the program is inflexible, and scalability is feasible but constrained by certain variables, e.g. the RAM capacity of the VM. Conversely, the application may scale horizontally, but this is costly as it requires replication of VM for each additional case. Although these methods were widely used, organizations used modern methods, e.g. Netflix is stronger [14]. Such different initiatives collectively referred to as cloud-native process models, were driven by a desire for fast technology development, massively scalable resources, broad scalability, and diversity of customers. Cloud-native design patterns aim at assignment writers to monitor continuously the cloud benefits. Because cloud platforms have unique features, such features must be considered throughout application creation and deployment. For cloud environments, for example, mistakes are normal due to the complexity of distributed systems, thus the capacity of an application to detect and handle failures enhances resilience. This is done through the implementation of the circuit breaker pattern.1 Cloud-native upregulation is still constantly changing; hence, a clear understanding of the core ideas is far beyond the purpose of this document. In scientific and corporate contexts, though, we point to sources. Moving on from Fehling et. Cloud-native design patterns (all [18] and Kratzke et al) involve durability, usability, secluded provinces, permeability, and weak couplings. Another common definition of CNAs feature is given in Twelve-Factor apps, a Heroku cloud engineering group construction. 2 These considerations include microservices, adaptable self-service infrastructures, collaborative, and anti-fragility-based APIs [14]. We include finer aspects in Section V about how we enforced these characteristics in our prototype.

### 5.2 Use Case of a Cloud Native SecaaS - Security Integration in DevOps.

The DevOps methodology aims to enhance cooperation among enterprise development and operations teams [20]. This means improved coordination and shared decision-making among these groups, and essentially DevOps provides energy generated for shorter development phases of software. Nonetheless, security teams are frequently overlooked which hurts software development, e.g. security vulnerability introduction [19]. Consequently, concerted attempts to incorporate safety checking into CI and/or CD i.e.DevSecOpsare indispensable for balancing speed and health. One potential solution is to train DevOps on safety roles. But that solution is not ideal and may be limited provided the security knowledge requirement. Therefore, another solution is to recruit safety professionals to work directly with DevOps. Added expenses are a weakness with this strategy, especially for small businesses, e.g. entrepreneurs. Yet another alternative is for SecaaS providers to outsource protection. To address the above problems, SecaaS may be built into CD pipelines. SecaaS API calls can optimize security operations, like safety analysis and reporting mechanisms.

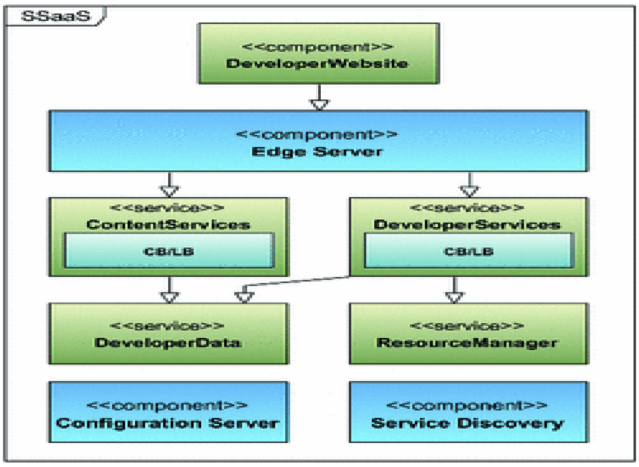

**Figure 5.** Migration of Cloud-native

### 5.3 Securityrequirementsfor migrationtoCNA and CAVAS system model.

To examine SecaaS migrations for CNA, we have used CAVAS as the goal framework. CAVAS was initially a monolithic framework to perform security tests for Dynamic user acceptance testing (DAST). We built it for prior cloud risk analysis research[20]. Given our current expertise and working understanding of the program, CANVAS was chosen as the source program for this job. Familiarity with software for migration improves the chances of success due to the enormous code re-factoring process. We first address the security criteria for CNA migrations in the next sections and include information on the measures taken throughout CAVAS relocation. Afterward, we define the device model of the transformed application.

### 5.4 SecurityRequirements andConsiderations.

Some requirements for migration to CNA include:

- **Application Security:**

The process of breaking down a framework into small, independent parts frequently involves significant code refactoring, and even new software creation. For programming languages/systems each microservice can be built locally. This variety of technologies implies additional security overheads, e.g. vulnerability evaluations will address such disparate technologies in cloud architecture as opposed to narrow technologies.

- **Network Security:**

The protection of the target migration environment needs an evaluation of the security. Especially on cloud platforms, because protection is a mutual obligation among cloud service providers (CSP) and renters, this is crucial. Security detectors for instance can be used to search target VMs for vulnerability tests. In addition, firewalls, protection groups, and other security mechanisms for cloud networking should be designed in advance. This security assessment, therefore, depends on the implementation model of the cloud. Above that,the initiatives are unique to IaaS, and similar efforts on PaaS may not be appropriate[21].

- **Data Security:**

Information storage is an essential security function, which may severely affect the effect of information leaks or breaches during the transition. Data is basically in motion throughout the transition and subjected to many vectors of assault, like attacks by MiTM. A potential database migration technique is a staggered method, in which servers are transferred over several phases. There are 2 key database models used in CNA, a file-sharing model in which many microservices have used a central vocabulary, and a -per-service database model where each microservice element retains its very own database [21]. The subsequent model entails more complexity and thus extra security effort is needed. Regardless of the strategy, however, effective steps such as data encryption may be regarded to minimize safety risks.

- **Security Monitoring:**

According to the trend of observability, software monitoring has been embedded in cloud-native architecture. Correctly, most frameworks to apply CNA incorporate the collection functionality of logs and statistics. But these reports and reports are designed primarily for quality debugging and daily medical checks. In most CNA frameworks and systems, physical security isn't accessible. Security monitoring may help to identify security risks and secure expanded surfaces of assault.

### 5.5 Justification for Migration to CNA and Redesign Efforts.

Efficient auto-scaling was a requirement for Efficient auto-scaling was a prerequisite for CAVAS due because of the need to manage volatile spikes in scan demands effectively. A feasible approach for managing overload demands is to auto-scale the overloaded modules horizontally[22]. Monolithic implementations, nevertheless, are mostly vertically scale-up and are now in-flexible and semi-fault tolerant[21]. On the opposite, CNA provides horizontal scaling possibilities, per web service. Furthermore, vulnerability management activities are typically decades-long tasks and Processor-intensive, for instance, a thorough scan of a simple website could require 15-20 minutes. Therefore, horizontal scaling is ideal for numerous calls to be handled in parallel. Similarly, a magnitude down is useful if there are very few requests since this great way to turn, i.e. only a minimum amount of Virtual Machine (VM) clouds will remain active. The CAVAS redesign accompanied several moves and relevant factors. Initially, the latest framework was revamped and introduced with a microservice method. As seen in Figure 3 the elements were decayed as separate entities. Next, we implemented cloud-focused sequence diagrams such as load-balancing, network configuration, externally imposed specification, and API gateway[5], specifics of execution are provided in Section V. Three approaches were considered for deployment strategy: 1) intermodal surroundings like Docker 2) multiple clouds i.e. (PaaS or IaaS) 3) a combination of the retired two. We opt for the hybrid solution, i.e.An OpenStack server and a Docker cloud. OpenStack also incorporates many principles of cloud-native nature such as scalability and speedy permeability. The preference for container integration was due to the intrinsic performance and scalability of the container technologies. It might also be simpler and more convenient to control non-Open Stack elements when implemented in containers than to use virtual OpenStack occurrences. Notice the improvements made to the network interface for information, the scanning system offers the database server strategy[21] to store information, e.g. scanner configuration. In the User Server, a Redis database is installed to maintain session details, and outsource user status from the UI System i.e. statelessness. This solution offers greater scale versatility and simple commodification i.e. an unsafe system can be quickly substituted by another one.

### 5.6 System Model.

### CAVAS can be implemented in 2 ways:

As either an integrated software pipeline module or an auto testing procedure. A user provides the hostname or IPaddress of a targeted tool for vulnerability scanning in a later style. The test request is forwarded to the UI Server through the API Gateway, where the user is authorized. If confirmation of the desired resource ownership is required, the information is made to the real-time process. Many specifics of the verification method we are implementing can be found in Section V-E. After successful verification, the application is moved on to the Controller Service, allowing an API call to the Scanner provider. The query is further addressed to the corresponding scanning machine depending on the form of query i.e. web app test queries are addressed to either Arachni[18] or OWASP ZAP[18]. The scan operation is started, and the document is transmitted to any instances of the UI System where it is delivered to the consumer.

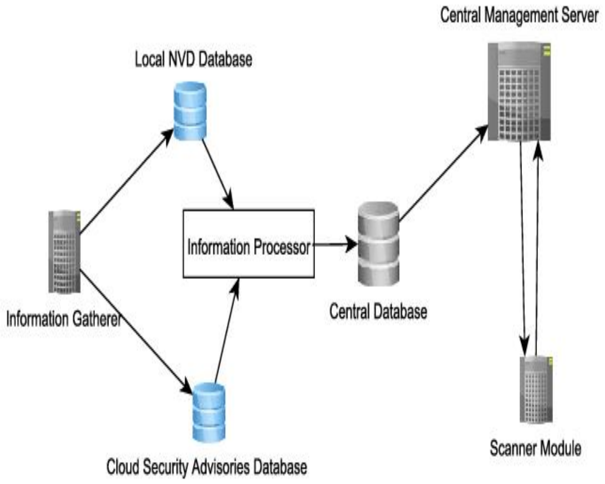

**Figure.6** High-level architecture of CAVAS

## 6. DIGITAL TRANSFORMATION

Cloud transformation has been described as the method of explicitly or implicitly distributing the digital properties, products, IT assets, or cloud applications of an organization. Migration to the cloud computing platform will trigger supply disruptions. Thus, well-developed approaches must be adopted for decision-making and also migration law. This segment provides for the analysis of existing Policy Structure and Migration procedures to work in the cloud. The six R's are the first high top Muster. Those R's are Re-host, Repurchase, Re-Platform, Retain, recompile, and Withdraw. The rehost plan takes over the current Applications and primarily runs it on the cloud service the business model type IaaS. Strategy for Re-Migration Program guarantees improvements to the host application platform for the use of the PaaS payment system and optimizes it. Buyback

the approach allows for the acquisition of new SaaS products. Refactoring alternatives would need redevelopment of the facilities usage of the native cloud apps. For certain instances, it could be wise to maintain such on-premise network facilities. Whereas there may not even be other programs and technologies.

Therefore, retirement them best of any use. Amazon gave out six phased approachesto cloud migration execution Amazon Web Services (AWS) include cloud analysis, data migration, code migration, feasibility study; leverage, and optimization of the cloud[23]. Cloud migration should not be based solely on financial and technical benefits.Organizational aspects need to be learned, too. Impact research has to be carried out by stakeholders proposed to assess Cloud benefits and weaknesses of app business migration[24]. In one of those instances, the organization has identified the following benefits: the ability to manage incomes & outgoings, deliver new incentives products/services, enhanced standing, boring job canceled, boost job satisfaction, and create new opportunities skills, and corporate growth prospects. Likewise, the following threats have been recognized: Customer service degradation & quality of service, increased reliance on third-party external, reduction of job satisfaction, individual departments downsizing, confusion over emerging technologies, lack of funding, and lack of cloud comprehension. The cloud migrationreference model Cloud-RMM, developed from a literature review, exists to migrate legacy frameworks into the cloud. [24]. This recognizes three moves time and compares different activities, like preparation, implementation appraisal, and cross-cutting issues. This alsoaddresses researched regions, such as the shortage of migration aid resources, aid for spatial change, and self-adaptive cloud installations. This describes 4 distinct data collection; such as removing (one or even more Cloud service components), migration partly (one or more layers of framework or collection of IEEE 5th annual meeting on engineering 2018 Architectural elements of one or more strata & Applied Sciences, 22-23 November 2018, Bangkok Thailand introducing special cloud functionality), migrate the entire stack (VM-encapsulated program and running on the cloud and cloudy (complete implementation of applications through cloud utilities). The Cloudstep is a step-by-step decision process about cloud migration for traditional applications [25]. Uses the prototype definition profiles focused on three individuals, such as the digital transformation organization,deployment framework, and cloud service provider. The restrictions are similarly defined in each organization and speech. On the efficient resolution of migration constraints, the strategy is established, and the transition is finally carried out. This stresses still in the phase of pilot migration. Excellent performance migrates to the cloud to attain software cost objectives and quality is based on seven tasks and four strategic actions linked in networked versus hierarchical fashion dress[26]. Those conditions for judgment are added distribution, services and cloud service providers, multitenancy criteria, and elasticity strategy. The job to be completed carried out are monitoring work-load, enforcement, safety issues found, appropriately found level QoS, forecast results, including cost, and commitment estimate. For one analysis five approaches for the legacy were established migration of the program to IaaS, transfer to PaaS, the substitution of SaaS, SaaS-based revising, and SaaS reengineering[27]. The migration legacy program for SaaS needs to be restructured n SOA from the cloud deployment model first as well as SOA leads. Initially, the process of recovery helps in extracting Legacy System Architecture throughexperience exploration and geometric modeling. Migration is the next operation making use of SOA and cloud trends, replacement or covering of and redesign of existing components in construction, by design in operation. Now, the cloud at SaaS implementation is through model-driven architectures (THE MDA). Multi-cloud migration is the rule, because of many problems such as managerial, or technological enforcement. VPAM (Variability-based, Pattern-drivenArchitecture Migration) enables the collection and customization of migration patterns forapplications. The pattern defines the required architectural adjustment (refactoring) within the re-engineering of systems and parameterization of implementation. and the pattern variance is implemented in three such dimensions include access, functionality, and platform. It has it 15 Migration patterns described coded as MP1 to MP15 classified into five core trends, for example, redeployment, relocation, refactoring multi-cloud, multi-cloud rebinding, replacement, and variations thereof. Enterprise IT market leaders have set up their cloud on the same lines to deliver approaches, tools, andtechniques to move the IT infrastructures of their clients into their cloud. Orakel providesan EBS platform for business cloudoptimization suites for using the cloud world and lift and transfer software from clients to oracle cloud. Moreover, they have one JD Edwards clip including tool applications and JD provisioning tool with one button applications in Edwards. SAP likewise provides a range of choices to move ABAP-based SAP systems to the SAP HANA Cloud application. Application. They bring thetransformation of the SAP landscape migration applications and network-providing tools [28]. SAP also features thousands of SAP-builtapplications and PaaS platform partners in creating new technologies. Microsoft's Azure is a platform for cloud computing that provides tools to analyze, transition, and automate in all three stages. This comprises three migration strategies to the Azure zone, namely the rehost, refactor, rearchitect and reconstruct[29]. Re-host means raise and rehost the change strategy, the refactor strategy, the PaaS services, the rearchitect needs code transformation into micro-services, and restoring approach includes restoring the cloud app platform local. Strategies and procedures for cloud migration can be divided into five levels, namely market assessments, technical assessments evaluations, Migration Policy, and migration strategy execution and development critical company assignments evaluations are evaluations of the workloads, workloads quality audit evaluation, resolving protection concerns, quality of service rates calculated, predictions on results, cost analysis, an estimate of effort and readiness for the organization. Production in this level of workload the unit made up of several common programs, System photos , and customers, and company components should be allocated for easy to priories [30].To use cloud-based services such as the PaaS database plan service. Institutions of the SaaS business model need to adopt a policy for the rebuy. Approach component or cloudy converts a framework into the design of microservices using the full range of cloud computing, for example, isolation; the parallelization and permeability can be performed in containers. Developing the choice to use the enterprise cloud platform requires complete access redesign and application development from the start. Institutions should benefit from the lowest risk rehosting methods, re-platform, repurchase, process restore refactor when the program is designed in the optimization stage transitioned for threat reduction. Cloud transformation has been described as the method of explicitly or implicitly distributing the digital properties, products, IT assets, or cloud applications of an organization. Migration to the cloud computing platform will trigger supply disruptions. Thus, well-developed approaches must be adopted for decision-making and also migration law.

## 7. CONCLUSION

Customized cloud solutions and integration plans should be prepared to adopt the cloud, taking into account the size of data, regulation concerns, cloud preparedness of business applications, leisure time costs and SLA necessity, and dataand application movement of people in the event of cloud provider transformation. In other words, organizations must have a consistent plan, architecture of the application, and model of governance for a move over to the cloud.This paper provides a brief overview of cloud computing as well as a cloud-native framework at the top standard. While cloud computing usage has increased exponentially, cloud computing protection has still been considered the largest problem in the cloud computing world. Additionally, the loss of the level of information has recently created major issues for a lot of customers. Additionally, data encroachment causes a lot of problems for cloud computing consumers. We discovered that a lot of studies have been carried out to maintain the protection of single cloud and cloud processing while less attention was paid to multi-clouds in the secure area.There are a variety of fields that will affect both the growing cloud infrastructure and the evolving computing infrastructure. SecaaS systems use the SaaS cloud delivery model to provide top-quality security services. This security solution, despite the availability of simple and cheap exposure to security without the need for complicated installations and configurations, provides reliable protection for businesses experimentally. Future interesting work involves automating CNA migrations with appropriate pre-migration security assessments. A customized cloud strategy and migration plan should be in place to implement the cloud, taking into account data size, regulatory concerns, cloud readiness for business applications, and the cost of downtime. This paper provides a brief overview of cloud computing and cloud-native platform at the top standard. Leading enterprise technology vendors have built cloud infrastructures and deliver their technology the best in class migration solutions and strategiesand additional support to third-party vendors.

## References:

1. Georgiadis G, Poels G. Towards a privacy impact assessment methodology to support the requirements of the general data protection regulation in a big data analytics context: A systematic literature review. Computer Law & Security Review. 2022 Apr 1;44:105640.
2. Abraham, I., et al., *Byzantine disk paxos: optimal resilience with byzantine shared memory*. Distributed Computing, 2006. **18**(5): p. 387-408.
3. Rahmani P, Fakhrahmad SM, Taheri M. New attacks on secret sharing-based data outsourcing: toward a resistant scheme. The Journal of Supercomputing. 2022 Apr 25:1-37.
4. Ateniese, G., et al. *Provable data possession at untrusted stores*. in *Proceedings of the 14th ACM conference on Computer and communications security*. 2007.
5. Vukolić, M., *The Byzantine empire in the intercloud.* ACM Sigact News, 2010. **41**(3): p. 105-111.
6. Cachin, C., R. Haas, and M. Vukolic, *Dependable storage in the intercloud.* IBM research, 2010. **3783**: p. 1-6.
7. Bowers, K.D., A. Juels, and A. Oprea. *HAIL: A high-availability and integrity layer for cloud storage*. in *Proceedings of the 16th ACM conference on Computer and communications security*. 2009.
8. Bessani, A., et al., *DepSky: dependable and secure storage in a cloud-of-clouds.* Acm transactions on storage (tos), 2013. **9**(4): p. 1-33.
9. Birman, K., G. Chockler, and R. van Renesse, *Toward a cloud computing research agenda.* ACM SIGACt News, 2009. **40**(2): p. 68-80.
10. Schneider, F.B. and L. Zhou, *Implementing trustworthy services using replicated state machines.* IEEE Security & Privacy, 2005. **3**(5): p. 34-43.
11. Raj, P. and A. Raman, *Multi-cloud management: Technologies, tools, and techniques*, in *Software-Defined Cloud Centers*. 2018, Springer. p. 219-240.
12. Yu, S., et al., *Can we beat DDoS attacks in clouds?* IEEE Transactions on Parallel and Distributed Systems, 2013. **25**(9): p. 2245-2254.
13. Fehling, C., et al., *Cloud computing patterns: fundamentals to design, build, and manage cloud applications*. 2014: Springer.
14. Stine, M., *Migrating to cloud-native application architectures*. 2015: O'Reilly Media.
15. Balalaie, A., A. Heydarnoori, and P. Jamshidi. *Migrating to cloud-native architectures using microservices: an experience report*. in *European Conference on Service-Oriented and Cloud Computing*. 2015. Springer.
16. Dmitry, N. and S.-S. Manfred, *On micro-services architecture.* International Journal of Open Information Technologies, 2014. **2**(9).
17. Leymann, F., et al. *Native cloud applications: why monolithic virtualization is not their foundation.* in *International Conference on Cloud Computing and Services Science*. 2016. Springer.
18. Torkura, K.A., et al. *Leveraging cloud native design patterns for security-as-a-service applications*. in *2017 IEEE International Conference on Smart Cloud (SmartCloud)*. 2017. IEEE.
19. Rahman, A.A.U. and L. Williams. *Software security in devops: Synthesizing practitioners' perceptions and practices*. in *2016 IEEE/ACM International Workshop on Continuous Software Evolution and Delivery (CSED)*. 2016. IEEE.

20. Torkura, K.A. and C. Meinel. *Towards vulnerability assessment as a service in openstack clouds*. in *2016 IEEE 41st Conference on Local Computer Networks Workshops (LCN Workshops)*. 2016. IEEE.

21. Wettinger, J. *Native Cloud Applications: Why Monolithic Virtualization Is Not Their Foundation*. in *Cloud Computing and Services Science: 6th International Conference, CLOSER 2016, Rome, Italy, April 23-25, 2016, Revised Selected Papers*. 2017. Springer.

22. Lorido-Botran, T., J. Miguel-Alonso, and J.A. Lozano, *A review of auto-scaling techniques for elastic applications in cloud environments.* Journal of grid computing, 2014. **12**(4): p. 559-592.

23. Varia, J., *Migrating your existing applications to the aws cloud.* A Phase-driven Approach to Cloud Migration, 2010.

24. Khajeh-Hosseini, A., D. Greenwood, and I. Sommerville. *Cloud migration: A case study of migrating an enterprise it system to iaas*. in *2010 IEEE 3rd International Conference on cloud computing*. 2010. IEEE.

25. Beserra, P.V., et al. *Cloudstep: A step-by-step decision process to support legacy application migration to the cloud*. in *2012 IEEE 6th international workshop on the maintenance and evolution of service-oriented and cloud-based systems (MESOCA)*. 2012. IEEE.

26. Andrikopoulos, V., S. Strauch, and F. Leymann, *Decision support for application migration to the cloud.* Proceedings of CLOSER, 2013. **13**: p. 149-155.

27. Zhao, J.-F. and J.-T. Zhou, *Strategies and methods for cloud migration.* international Journal of Automation and Computing, 2014. **11**(2): p. 143-152.

28. Missbach, M., et al., *SAP on the Cloud*. 2016: Springer.

29. Ahmad, N., Q.N. Naveed, and N. Hoda. *Strategy and procedures for Migration to the Cloud Computing*. in *2018 IEEE 5th International Conference on Engineering Technologies and Applied Sciences (ICETAS)*. 2018. IEEE.

30. Council, C., *Migrating applications to public cloud services: roadmap for success*. 2016.