future internet

MDPI

Review

Survey on Secure Scientific Workflow Scheduling in Cloud Environments

Hadeel Amjed Saeed [1], Sufyan T. Faraj Al-Janabi [1], Esam Taha Yassen [1] and Omar A. Aldhaibani [2,*]

[1] College of Computer Science and Information Technology, University of Anbar, Ramadi 31001, Iraq; hadeel.saeed@uoanbar.edu.iq (H.A.S.); sufyan.aljanabi@uoanbar.edu.iq (S.T.F.A.-J.); co.esamtaha@uoanbar.edu.iq (E.T.Y.)

[2] School of Computer Science and Mathematics, Liverpool John Moores University, Liverpool L3 5AH, UK

* Correspondence: o.a.aldhaibani@ljmu.ac.uk

Abstract: In cloud computing environments, the representation and management of data through workflows are crucial to ensuring efficient processing. This paper focuses on securing scientific workflow scheduling, which involves executing complex data-processing tasks with specific dependencies. The security of intermediate data, often transmitted between virtual machines during workflow execution, is critical for maintaining the integrity and confidentiality of scientific workflows. This review analyzes methods for securing scientific workflow scheduling in cloud environments, emphasizing the application of security principles such as confidentiality, authentication, and integrity. Various scheduling algorithms, including heuristics and metaheuristics, are examined for their effectiveness in balancing security with constraints like execution time and cost.

Keywords: scientific workflow; security; scheduling; CIA triad; cloud computing

Academic Editors: Jerry Chou and Wu-Chun Chung

Received: 10 December 2024
Revised: 14 January 2025
Accepted: 16 January 2025
Published: 21 January 2025

Citation: Saeed, H.A.; Al-Janabi, S.T.F.; Yassen, E.T.; Aldhaibani, O.A. Survey on Secure Scientific Workflow Scheduling in Cloud Environments. *Future Internet* **2025**, *17*, 51. https://doi.org/10.3390/fi17020051

1. Introduction

The rapid development of information technology has led to exponential growth in data generation, and the need for processor speed has increased. Therefore, the argument for a storage place capable of storing huge amounts of data increased, so the trend towards cloud services began, and this trend increased the need to maintain data security and simplify it during use. Cloud computing is an essential technology that has arisen to fulfill the growing demand for more cost-effective information technology services. It depicts a concept in which the cloud service provider provides a pool of resources, such as computing power, storage, bandwidth, and so on, in the form of on-demand services that consumers can rent over the Internet [1–3].

In this context, many tools have been utilized to facilitate data processing, including workflow. Workflows are a typical application model in computational science. They define a series of computations that allow for structured and distributed data processing and are often stated as a set of jobs with interdependencies. These apps provide an efficient method of processing and retrieving insight from the ever-growing data generated by increasingly powerful technologies [4,5].

A directed acyclic graph (DAG), typically used to depict scientific workflows, is highly vulnerable to attacks since errors made in the middle will be reflected in the end product. Moreover, the fundamental knowledge in certain scientific domains is frequently hidden in the intermediate data of scientific procedures. Users will suffer enormous damages if their data are taken [6–8]. Adversaries can affect workflow execution in various ways: (i) Attackers can get inside virtual machines (VMs) running workflows and make them go

down. (ii) Instead of disrupting the workflow, the adversaries may aim to tamper with its execution result by manipulating the execution software and workflow intermediate data. (iii) Once inside the virtual machines, the adversaries can also steal the workflow data or implant the backdoor to facilitate the next invasion [6,9,10].

Recently, security has gained popularity as a requirement, and there are numerous ways to implement it. Either (1) separate data into sensitive and non-sensitive groups and send them to a cloud (public or private) or (2) make sure that all data processed in public clouds is safe by providing security services like authentication, integrity verification, and privacy [11]. Reducing the number of tasks in a scientific workflow is an important problem in cloud computing. Some workflow scheduling algorithms apply security services such as authentication, integrity verification, and encryption for sensitive and non-sensitive tasks. However, this approach requires a lengthy implementation period and incurs financial costs.

This paper explores the various methods and algorithms developed to secure scientific workflow scheduling in cloud environments. We analyze the effectiveness of these approaches, focusing on the trade-offs between security, execution time, and cost. The goal is to comprehensively understand the current landscape and highlight key areas for future research in secure cloud-based scientific workflows.

2. Scientific Workflows in the Cloud

Cloud computing provides pay-per-use computer resources over the Internet. The cloud model provides ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (such as networks, servers, storage, applications, and services) that can be quickly provisioned and released with minimal management effort or service provider interaction [2,3]. Figure 1 shows three cloud services and several deployment methods. Software, platform, and infrastructure as a service are the types of services. Basic deployments are public, private, and hybrid. According to NIST, cloud computing has five fundamental qualities: on-demand self-service, broad network access, elastic resource pooling, quick elasticity, and measurable service [12].

The workflow domain orchestrates task sequences and automates processes, considering energy dissipation, virtual machine types, workflow types, enforced function counts, deadline limits, cloud billing charges, acquisition and termination delays, etc. Workflow processing includes problem identification, dataset collection, loading and summarization, data segregation, model assessment, feature scaling, algorithm selection, model training, validation, and prediction. Face recognition, scientific procedures, and object detection systems benefit from workflow scheduling [1,13].

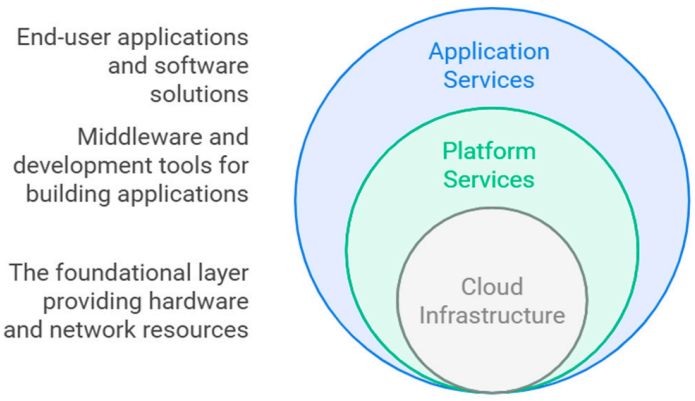

Figure 1. The diagram graphically illustrates the cloud computing model [14].

Data bifurcates workflow classifications into business and scientific categories. First, business workflows are practical workflows depicted as a directed acyclic graph. Second, scientific workflows encompass numerous tasks and necessitate an extensive array of tools for their execution [2,10].

A scientific workflow comprises a series of interdependent computer processes. We categorize the interdependencies between jobs as either data or control-flow dependencies [4,5]. Intricate workflow applications, such as gravitational wave physics, astronomy, and bioinformatics, require significant computational power and utilize cloud resources to analyze large data volumes. Cloud tasks manifest as workflows, predominantly depicted as directed acyclic graphs (DAGs), as seen in Figure 2 [4,5]. A workflow management system receives an abstract process as input and transforms it into an executable format. Scientific workflows are employed to assess cloud workflow scheduling techniques. These include [6,15,16] the following:

1. **CyberShake**: Developed by the Southern California Earthquake Center, CyberShake is designed to assess seismic hazards in a region by using probabilistic seismic hazard analysis (PSHA). It simulates earthquake ground motions by integrating faults, geology, and seismic wave propagation information.
2. **Montage**: Created by NASA, Montage is a workflow application that creates large-scale sky mosaics by stitching together multiple astronomical images. Using input images from various telescopes and data sources, Montage generates high-resolution mosaics that astronomers use to study celestial objects across different wavelengths.
3. **LIGO inspiral**: The Laser Interferometer Gravitational-Wave Observatory (LIGO) uses this workflow to analyze gravitational wave data produced by events such as the merging of binary systems, including black holes and neutron stars.
4. **Sipht**: Developed by Harvard University, Sipht is a workflow used in bioinformatics research to search for small, non-coding RNAs across various bacterial genomes. These small RNAs play a critical role in gene regulation, and identifying them is essential for understanding cellular processes and developing medical applications.
5. **Epigenomics** is in the bioinformatics field, a CPU-intensive application that automates the execution of various genome sequencing operations [3].

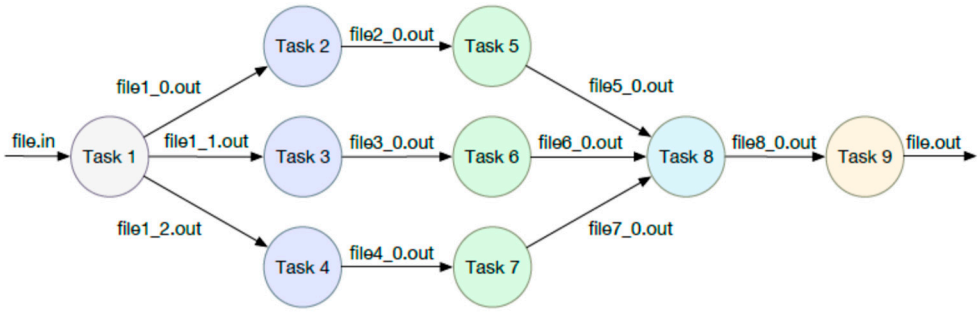

Figure 2. Illustrative workflow with nine activities. The graph nodes signify computing jobs, while the edges denote the data dependencies among these activities [3].

A standard cloud-based scientific workflow system consists of five fundamental modules: the workflow engine, user interface, workflow management module, cloud provider APIs, and cloud resources, as seen in Figure 3. The workflow engine obtains the designated workflow from users by modeling their operations in the user interface. The three main features of the cloud-based scientific workflow system are resource allocation, workflow scheduling, and workflow interpretation. The workflow engine serves as the system's primary component. The resource provisioning module will generate virtual

machines (VMs) according to the resource specifications established by the workflow parser after the submission of the workflow [6,9,17].

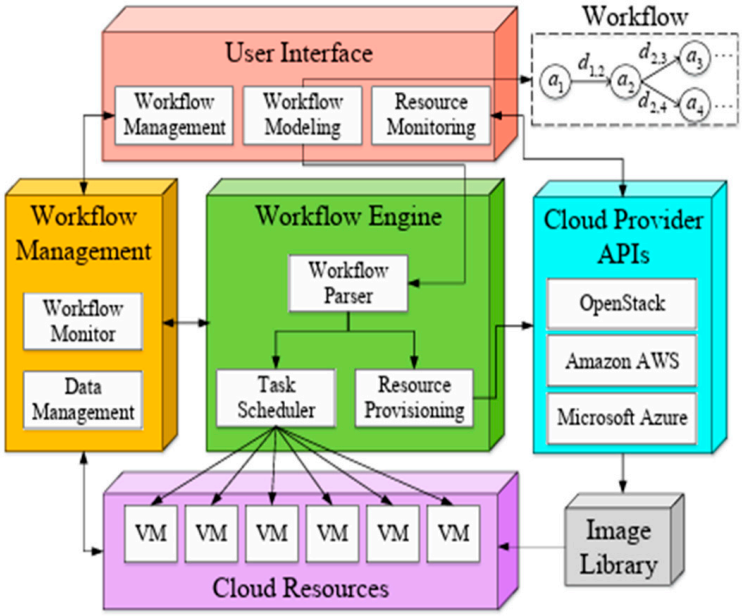

Figure 3. The architecture of a conventional cloud-based scientific workflow system [18].

Additionally, the workflow parser is capable of transforming each abstract workflow into an internal executable representation. The task scheduler will assign the workflow sub-tasks to virtual machines (VMs) for execution employing specified scheduling methodologies. Users can select from several scheduling strategies, including minimizing workflow makespan, reducing the financial expenses of workflow execution, and enhancing security according to their requirements. The workflow management module will supervise the generated intermediate data and track the execution status of the workflow during its operation. This information will be shown on the user interface [6,9].

3. Scientific Workflow Scheduling in the Cloud

Cloud computing can boost data science workflows by providing data availability and accessibility, data processing and analysis, data visualization and presentation, and data security and compliance. Cloud computing can help data scientists work faster, smarter, and more efficiently on their data science projects. Scheduling of scientific workflows in cloud computing introduces the following challenges [3,8,19,20]:

1. Mapping task classes to virtual resources results in a significant make-span, and the challenge is in identifying a minimal set of ideal schedules that maximize performance according to user-defined quality of service parameters, such as cost and speed.
2. A user-controlled scheduler assigns resources in a cloud environment. The challenge lies in determining the types and quantities of resources required for the workflow application to function effectively. Resource overprovisioning enhances performance but escalates costs, while resource under-provisioning detrimentally affects efficiency.
3. Dependencies on data and control flow between tasks increase the wait time before a task is ready to start, which lengthens the makespan.

Figure 4 below illustrates numerous classifications of methods and techniques for workflow scheduling [16,21].

Reinforcement learning is a machine learning methodology that addresses decision-making in dynamic contexts [22,23]. Metaheuristics are more computationally intensive

algorithms that can be utilized to address a wide range of optimization problems. They also attempt to find more necessary schedules by investigating various options and using guided search. Metaheuristics can be divided into two primary categories: population-based search and single-solution search [20,24].

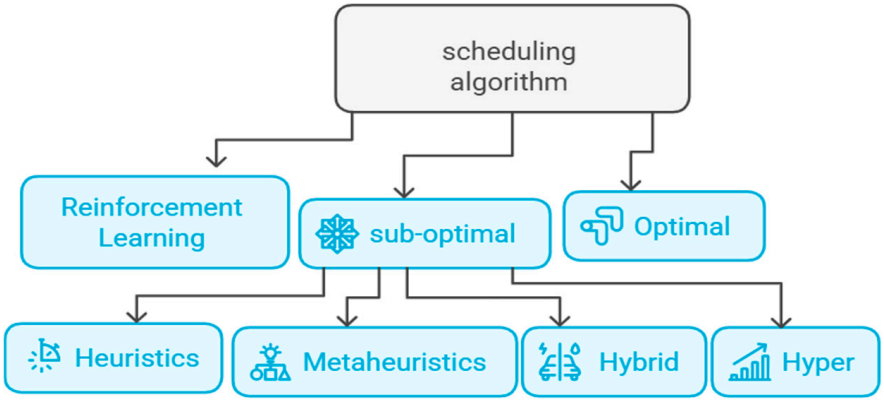

Figure 4. Categories of workflow scheduling methods based on optimization strategies.

A heuristic is a collection of guidelines for solving a particular problem in a reasonable amount of time. By integrating a workflow application and understanding cloud characteristics, the heuristic approach finds a timetable that meets the user's requirements [16,18,24].

Hybrid combines two heuristic algorithms, PSO and GA, or hyper with reinforcement learning or another way [25–27]. Hyper-heuristic is an automated methodology for selecting or generating heuristics to solve hard computational search problems [27]. Table 1 shows the differences, benefits, and limitations of each approach used in scheduling.

Table 1. The comparison between methods.

Approach	Differences	Benefits	Limitations
Reinforcement Learning	- Uses agents to learn optimal scheduling policies through trial and error.	- Adaptive to dynamic environments. - Can optimize long-term objectives. - Learned from experience.	- Requires large training time and data. - May struggle with complex, high-dimensional state spaces. - Computationally intensive.
Metaheuristics	- High-level strategies guiding other heuristics to explore the search space (e.g., genetic algorithms, particle swarm optimization).	- Capable of finding near-optimal solutions. - Flexible for various problem types. - Can escape local optima.	- No guarantee of finding the optimal solution. - May require fine-tuning of parameters. - Often computationally expensive.
Heuristics	- Problem-specific algorithms that use domain knowledge to find good enough solutions quickly.	- Fast and efficient for specific problems. - Easy to implement. - Requires less computational resources.	- May not find optimal solutions. - Can be inflexible for different problem types. - Susceptible to getting stuck in local optima.

Table 1. *Cont.*

Approach	Differences	Benefits	Limitations
Hybrid Approaches	- Combine multiple techniques (e.g., metaheuristics with heuristics) to leverage their strengths.	- Can balance exploration and exploitation. - Often achieves better performance than individual methods. - Versatile.	- Increased complexity in design and implementation. - May require more computational resources. - Complex tuning process.
Hyper-heuristics	- A higher-level approach that selects or generates heuristics to solve problems. - Can operate on a set of low-level heuristics or rules.	- Adaptable to different problem domains. - Automated heuristic design and selection. - Generalizes well.	- It may be less effective than specialized heuristics. - High computational overhead. - Requires careful design of the heuristic set.

4. Scheduling Objectives

All surveyed algorithms share the fundamental characteristic of cost awareness. Along with this goal, most algorithms look at performance indicators, such as the total time the system takes to run and the number of operations it does. Moreover, several cutting-edge algorithms integrate energy consumption, dependability, and security into their aims, as shown in Figure 5 [1,2,24,28]. Choosing less expensive resources, such as less energy-efficient servers or data centers, is a common strategy for cost reduction. Compared to more expensive, energy-efficient options, these resources may use more energy per job or computing unit.

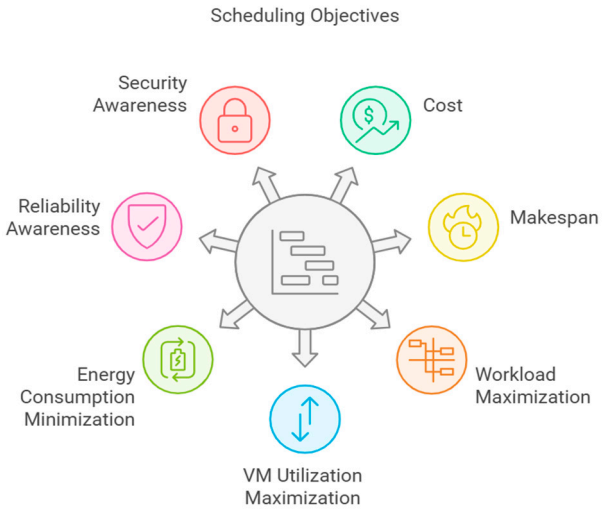

Figure 5. Types of scheduling objectives.

4.1. Cost

Algorithms designed for cloud platforms must consider the cost of leasing the infrastructure. Failure to comply may result in significantly elevated expenses associated with renting virtual machines, data transport, and cloud storage utilization. This objective is included in algorithms by either attempting to reduce its value or imposing a limit on resource expenditure (i.e., budget). All analyzed algorithms balance cost with supplementary performance or non-functional requirements, such as security, reliability, and energy consumption in cloud environments. For example, the predominant quality of service (QoS) demand is to minimize total costs while adhering to a user-specified deadline restriction [4,24].

4.2. Makespan

Most evaluated algorithms focus on the duration required to execute the process or the makespan. It is incorporated into the scheduling objectives by either striving to minimize its value or establishing a time constraint or deadline for workflow execution [4,25].

4.3. Workload Maximization

Workload maximization in cloud-based scientific workflows pertains to executing the maximum number of workflows within specified constraints, such as budget or deadline; thus, strategies in this domain focus on optimizing workflow execution within the allocated financial resources or designated time limits [4,25].

4.4. VM Utilization Maximization

Maximizing virtual machine (VM) utilization is essential in cloud-based scientific workflow scheduling. Most algorithms implicitly pursue this objective by being cost-conscious. Unutilized time slots in leased virtual machines are considered a financial inefficiency, prompting algorithms to circumvent them in their scheduling. Nonetheless, it is rare for these unutilized time intervals to emerge from workflow execution, mostly due to task dependencies and performance requirements. Certain algorithms focus on minimizing idle time slots and maximizing resource utilization, benefiting customers through cost reduction and providers through decreased energy consumption, increased profit, and more effective resource usage [4].

4.5. Energy Consumption Minimization

Individuals, organizations, and governments globally have heightened their interest in minimizing carbon footprints to mitigate environmental effects. This topic, while not exclusive to cloud computing, has garnered attention. Recently, researchers have developed several algorithms that consider energy consumption during process execution. They evaluate a synthesis of conflicting scheduling objectives while seeking a compromise among energy consumption, performance, and cost [4].

4.6. Reliability Awareness

Reliability is an important goal for algorithms, and they have ways to ensure workflow execution stays within users' QoS limits, even if a resource or task fails. Algorithms designed for unstable virtual machine instances prone to failure (e.g., Amazon EC2 spot instances) must have policies to ensure dependability. Common methodologies encompass duplicating essential work and utilizing checkpoints to reorganize unsuccessful processes. Algorithms must consider the supplementary expenses related to task replication and data storage for checkpointing. Moreover, it is essential to recognize that most scientific workflows are legacy software lacking checkpointing facilities, making reliance on this assumption potentially unrealistic [4].

4.7. Security Awareness

Certain scientific applications may necessitate the safe management of input or output data. Furthermore, certain activities may include sensitive computations that require security measures. Algorithms addressing these security concerns may utilize various security services IaaS vendors provide. They may secure data by classifying it as immovable or employing resources or providers with superior security credentials to perform and store sensitive tasks and data. Taking these security measures into account influences scheduling decisions since functions may need to be rescheduled near fixed datasets, and the overhead associated with additional security services may need to be factored into time and cost estimations.

A secure workflow system necessitates the consideration of many security services for modeling security-sensitive applications, including authentication, integrity, and secrecy, as elaborated below [8,15,27] and illustrated in Figure 6.

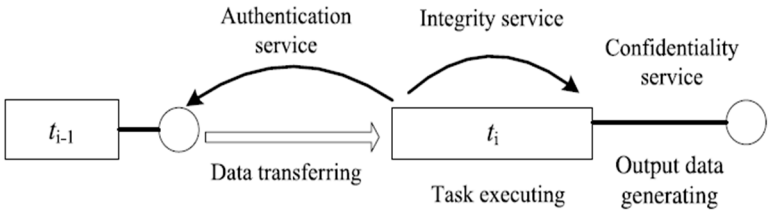

Figure 6. The structure of a typical cloud-based scientific workflow system [27].

4.7.1. Authentication

It concerns job execution agent identity verification dependability. AAA is a security architecture for authentication, authorization, and accounting. AAA checks a user's authentication credentials when they access cloud resources via a CSP. AAA checks system access after authentication. Users are authenticated using HMAC (MD5, SHA-1, and others).

4.7.2. Integrity

Integrity services protect data and applications in the IaaS cloud. Data integrity is compromised when an attacker modifies it. Hash functions like Tiger, RIPEMD-160, SHA-1, and others provide integrity.

4.7.3. Confidentiality

Users need confidentiality to store critical cloud resources safely. Confidentiality prevents eavesdropping and other passive risks to cloud resources. Passive attackers might disclose unsecured or unencrypted data transmission. Encryption algorithms like IDEA, DES, and others provide confidentiality.

5. Cloud Security

The protection mechanism against unauthorized access, use, and modification of cloud resources is called security in cloud computing. We employ various technologies, including rules, processes, and controls, to safeguard the cloud-based system's infrastructure from potential threats [29,30].

Security risk is related to many types of attacks, threats, vulnerabilities, and other issues. Therefore, it is important to take care when selecting or building a system from a vendor or customer. Thus, the cloud provider uses unique security standards, methods, and models to satisfy the client's requirements.

Cloud security protects data, applications, and infrastructure inside cloud computing environments. It encompasses an array of methodologies, instruments, and legislation aimed at safeguarding cloud-based systems from intrusions, data breaches, and various security risks [31,32]. Snooping, data manipulation, and spoofing are three prevalent risks in cloud environments. Snooping attacks entail an unauthorized entity intercepting data transmission between two network hosts. An unauthorized entity can access all traffic data if data transmission lacks encryption [10,17].

Data tampering refers to the unauthorized modification (editing, deletion, or manipulation) of data. In a data transmission context lacking protection, an unauthorized individual may intercept the data packet, alter its contents, and redirect its destination. Malicious refers to altered data, including a script to compromise user personal information. Finally, spoofing is an assault executed by a nefarious individual who impersonates another to infiltrate a third system, expropriate data, misappropriate funds, or disseminate malware.

Numerous spoofing attacks encompass email spoofing, caller ID spoofing, man-in-the-middle attacks, IP spoofing, and website spoofing [30,31]. Various measures are employed to mitigate these assaults in the cloud, including encryption, intrusion detection systems, zero-trust architecture, antivirus software, and others.

The Confidentiality, Integrity, and Availability (CIA) triad remains the most widely used framework for defining security vulnerabilities in traditional information systems. The primary goal of this section is to extend these security principles to the emerging cloud infrastructure. Figure 7 illustrates the key components of cloud computing data security, highlighting potential threats and corresponding countermeasures [32,33]. By adapting the CIA model to cloud environments, we can better understand and address the unique challenges posed by distributed systems, multi-tenancy, and remote access inherent in cloud computing.

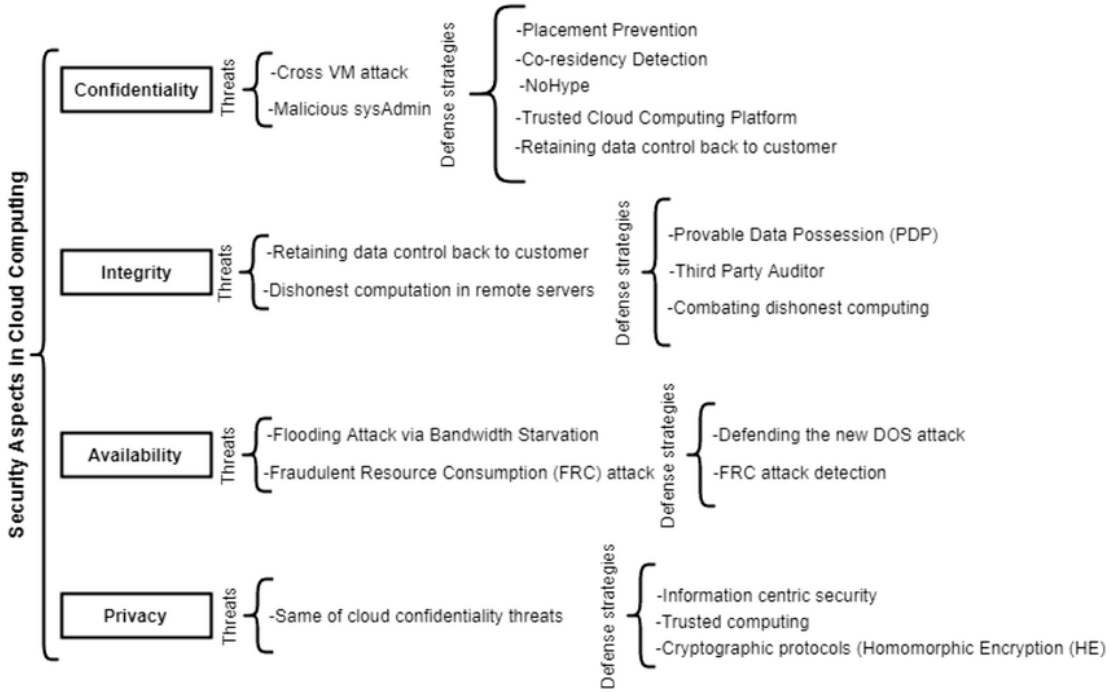

Figure 7. Components of data security in cloud computing [32].

6. Threats of the Scientific Workflows in Clouds

Numerous scientific procedures are associated with significant computer activities, including atmospheric science, bioinformatics, high-energy physics, etc. Process intermediate data contains vital secrets in some fields, and its theft would result in significant losses. However, one of the major cloud risks is data leakage because of several tenants [12,18]. Virtual machines (VMs) execute workflows in cloud-based scientific workflow systems.

After completing each workflow sub-task, the intermediate data produced in the virtual machines is employed to execute future sub-tasks. These intermediate data are generally unencrypted as they are temporarily retained in the virtual machine (VM). These intermediary data can be readily appropriated if the attacker can access the virtual computers. In a scientific process, virtual machines (VMs) generally exist within the same tenant network due to the necessity of regular data transfer. The tunneling protocol segregates distinct tenant networks; a failure of this protocol results in the visibility of all tenant communications.

Consequently, attackers can infiltrate the networks of other tenants by compromising the tunneling protocol. If an attacker has infiltrated a tenant network hosting numerous VMs conducting sub-tasks of scientific workflows, he can employ a scanning tool to

acquire information regarding the operating systems of these VMs. Three common threats could affect cloud workflow intermediate data: data loss, malicious media, and traffic eavesdropping [23,34,35]. The three threats are explained as follows [35]:

1. **Data of Loss:** VMs typically store workflow intermediate data. The availability of the intermediate data will be jeopardized if these virtual machines fail, as the intermediate data will be lost.
2. **Traffic Eavesdropping**: This could make it easier for attackers to obtain information transferred over the network unlawfully. Numerous workflows are associated with significant scientific computing activities, including atmospheric science, bioinformatics, high-energy physics, and so forth [25]. Because process intermediate data frequently contains key secrets in some domains, data theft would result in significant losses. The confidentiality of intermediate data will be in jeopardy due to this assault.
3. **Malicious Medium:** This threat involves intercepting and modifying data while it is being transmitted across the network. In some cases, adversaries may introduce malicious content to compromise the security of the data. Such actions can corrupt the workflow by altering or injecting harmful elements into the intermediate data. As a result, the integrity of the data is compromised, leading to inaccurate results and potentially rendering the entire workflow unreliable.

Before talking about the need for security, scientific workflow systems must know what challenges can be exposed. Four primary challenges exist [35,36] and are presented as follows:

1. To verify that each sub-task can be executed without any VM failures, the systems must assess the average earliest finish time of the virtual cluster about the subtask sub-deadline.
2. Systems must be capable of (i) evaluating the accuracy of sub-task results by analyzing the confidence of intermediate data across all copies and (ii) rectifying modified outputs to safeguard the system from the third type of assault by re-executing the current task.
3. The system must possess sufficient strength to endure the fourth type of attack by eliminating latent threats and purging executors through resource recycling.
4. Preserving system efficiency while implementing security measures, guaranteeing that the fault-intrusion-tolerant method does not adversely affect workflow performance.

As discussed earlier, scientific workflows face numerous attacks and challenges that can compromise their progress. Here, we explore the reasons why security is crucial, even for data that may not initially appear sensitive. Security is often associated with personal and financial data, but it is equally vital in scientific research for the following reasons:

1. Protecting Intellectual Property (IP): Researchers want to protect their property from unauthorized access or modification.
2. Maintaining Integrity in Workflow: Unauthorized interference or manipulation of the scientific process is prevented by a secure workflow, which guarantees that only permitted actions are carried out in the correct order.
3. Preserving Reproducibility: A fundamental aspect of scientific inquiry is reproducibility. Because the workflow is not changed, secure workflows contribute to the assurance that others may precisely repeat experiments.
4. Securing Resources: Scientific workflows often rely on computational resources like cloud platforms. Security measures protect these resources from unauthorized access, abuse, and exploitation, ensuring that they are available and functioning correctly when needed.

7. Secure Scientific Workflow in the Cloud

Moreover, cloud system security provides numerous advantages, such as centralized protection, decreased expenses and management, and enhanced reliability. In IaaS cloud workflow execution, a workflow management system (WMS) assigns workflow tasks to secure cloud resources to ensure failure-free execution. A secure workflow system necessitates consideration of several security services for modeling security-sensitive applications, including authentication, integrity, and confidentiality, as elaborated in Section 4.7 [37,38]. Investigations have been conducted about security concerns related to scheduling. Data security, data center security, and infrastructure security are the three categories into which the authors divided models after identifying the many security restrictions [39,40]. Our paper divided the security scientific workflow into data security, data center security, and allocation security, as shown in Figure 8, which shows each level.

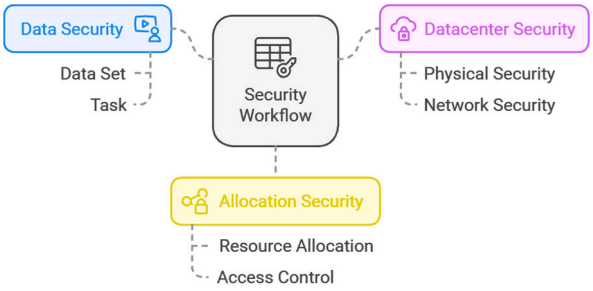

Figure 8. Types of security scientific workflow.

7.1. Data Security

The scientific workflow is a data-intensive application consisting of tasks and datasets, where a task may be related to multiple datasets and a dataset may also be related to consisting of tasks. There is a data dependency relationship between the tasks, where the output datasets of a task may be multiple tasks.

The tasks are very importantand have different processes, such as entrance, end, gathering, scattering, and both gathering and scattering. Every task type carries out particular tasks and has repercussions if its data are stolen. Offer five task selection policies that can be selected based on security needs. The Entry = End policy, which ensures the security of all data handled in both entry and end operations, is the initial policy. It can be selected for workflows that must secure both the source and finished data. Tasks that transfer sensitive data to numerous tasks or receive sensitive data from several tasks can be protected by the Gather and Scatter rules, respectively [37,41].

Tasks with multiple parent tasks and numerous child tasks should be gathered and scattered to ensure security. Lastly, every policy is applied when the data associated with all jobs requires secure handling. All five policies use encryption, integrity verification, and authentication services to secure crucial tasks. The workflow management system (WFMS) may provide policies to secure workflow execution [38].

7.2. Datacenter Security

Data interchange occurs during job scheduling, either within a data center or between data centers. Many communities have taken a great interest in it because of its fault tolerance, scalability, and flexibility [9,42,43]. Cloud service providers use their data centers (DCs) to house various IT equipment, including servers, storage, and network devices. This practice results in high power consumption and a greater environmental carbon footprint. For example, DCs account for around 3% of the world's power generation and have an estimated cost of over USD 30 billion. Calculations show that despite the volume of operations, DC power consumption is rising at a rate of 15% to 20% annually [5].

It is challenging to lower electricity costs when scheduling process operations since data are located in globally dispersed cloud data centers (GD-CDCs) to improve energy efficiency and system dependability [43,44]. So, it is very important to take a placement strategy for tasks to guarantee security.

7.3. Allocation Security

The relationship that exists between the scheduling of workflows and the allocation of resources (workflow specification and scheduling in hybrid clouds with security constraints) applications requests with computing and networking requirements are received by the scheduler, who also specifies which kind of virtual machine (VM) each workflow component will run on. The resource allocator then chooses which physical servers can house these VMs and which set of links will be used for communication between them [44]. We must specify when the resource allocator must be called by the scheduler. The scheduler's next task is to determine the locations for each workflow component's execution after receiving the workflow. It needs two types of data to accomplish this: the cloud infrastructure specification and the concrete workflow specification [22,45].

The scheduler's goal is to keep the workflow deadlines intact while limiting the renter's financial outlays. Workflow components for virtual machines in four distinct states can be directed by the scheduling output: (i) virtual machines that are currently assigned and operating in the private (ii) unallocated VMs at the private cloud; (iii) already allocated VMs in the public cloud; and (iv) unallocated VMs in the public cloud [45,46].

On the other hand, if the scheduler concludes that new, unallocated virtual machines (VMs) are required to ensure quality of service (QoS), the subsequent action is to identify the physical machines on which these VMs can be installed. Because of this, every process submission is scheduled separately, and the scheduler notifies the resource allocator of the list of virtual machines (VMs) that need to be constructed, the bandwidth that these VMs require to communicate with one another, and the tenant relationships that need to be trusted [45,47,48].

Generally speaking, the scheduler handles each workflow's allocation decision on its own. It can be added to the hybrid cloud tenant's local or personal context. The trust that exists between users is reflected in how dependent one user is on the others. To mitigate malicious and self-serving attacks in the intra-cloud network, the resource allocation strategy, which is based on the trust relationships between tenants, uses the workflow scheduler's information to allocate resources (e.g., the consumption of an unfair share of the network to complete tasks in a shorter amount of time) [47–49].

8. Survey of Secure Workflow Scheduling Approaches in Cloud Environments

Early in the 1980s, workflow technology was developed. Business workflows, grid workflows, and cloud workflows have all emerged with the growth of distributed computing. Systems for cloud computing use virtualization to provide flexible resource management. Cloud workflows are more productive and adaptable than previous workflow solutions. Consequently, cloud workflow solutions have emerged as a popular topic for research in recent years. In scientific workflows, scheduling in the cloud started many years ago, almost between 2009 and 2010. This section discusses various existing approaches to scheduling workflows in cloud environments, particularly those considering security aspects.

L. Zeng [50] proposed a security-conscious and cost-effective workflow scheduling technique (SA-BA) that ensures an affordable allocation of activities among available Cloud Service Providers (CSPs) to deliver consumers reduced makespan and enhanced security

services. They performed comprehensive simulation tests utilizing six distinct workflows from both real-world and synthetic applications, significantly enhancing resource utilization by concurrently managing dynamic data transfer and execution.

Z. Li et al. [42] This study presents five policies for selecting sensitive data tasks. Five task selection policies were available depending on security needs. The first policy, entry-end, secures all entry and end data. Workflows with sensitive beginning and end data can be used. gather, and scatter rules protect tasks that receive or send sensitive data from various sources. Cleaned up activities with various parent–child roles. When all task data needs secure handling, all policies are applied. All five policies safeguard critical processes via authentication, integrity verification, and encryption. They suggested a multi-population genetic algorithm workflow scheduling system to save costs and meet deadlines. Four process application experiments show the idea reduces make-span and cost.

A. R. Arunarani's [51] provides a cost-effective and secure scheduling method for various jobs in a scientific workflow in the Cloud. The suggested algorithm is based on the hybrid optimization technique, which incorporates the Firefly and Bat algorithms. The coding technique aims to achieve the time and risk rate limitations while minimizing the overall execution cost. A multi-objective function is used in the proposed system, and the results show that the approach consistently performs better than conventional techniques.

H. Y. Shishido and all [41] use coding to lower execution costs while meeting timing and risk rate constraints. The multi-objective function system routinely outperforms earlier methods. Five policies for selecting sensitive data tasks are suggested. Their workflow scheduling method employs a multi-population genetic algorithm to cut expenses and stay on time. Using four workflow applications in experiments shows that reducing time and money while keeping sensitive data safe is possible compared to a different method.

H. Y. Shishido and others' [52] paper includes workflow simulators incorporating the added labor of establishing security services for sensitive data. This study advises adding security services to the workflow simulator. These found seven workflow execution security overhead evaluation methodologies. They tested the extension using a process that used authentication, integrity verification, and encryption. By adding security services to private data and investigating how security affected time, cost, and security metrics, the add-ons emulated process execution.

Y. Wang and all [35], they proposed (ACISO) system to increase the availability, confidentiality, and integrity of the intermediate data, thereby securing it. Hash functions, encryption algorithms, and erasure codes create availability, confidentiality, and integrity strategy pools. Next, they presented the Security Strategy Optimal Allocation Model (SSOA), which maximizes intermediate data security while complying with workflow makespan and storage overhead limits.

A. Abdali and S. M. Nia's [53] study aims to introduce a novel, robust algorithm for managing the scheduling of numerous workflows by utilizing various quality of service (QoS) criteria. It does this by combining the CPSO and GA metaheuristic algorithms. However, it should be mentioned that the most immediate performance measure used to manage the scheduling process is the combination of three separate QoS: execution cost, load balancing, and security. The execution cost of this algorithm was minimized while meeting deadline and risk rate requirements, according to the conclusion drawn from it. Verifying the suggested algorithm using alternative algorithms was considered.

Y. Wang et al. [18] suggest switching defensive methods throughout workflow execution to reduce network scans and turn workflow security into an attack–defense game. Calculating the attack–defense game model's Nash Equilibrium yields the probability distribution of the best-mixed defense options. Based on likelihood, workflow execution uses differentiated virtual machines. A dynamic HEFT (heterogeneous earliest finish time)

task-VM mapping mechanism is included to boost workflow efficiency and defense strategy switching. The studies are carried out in both a simulated and real-world setting. The findings show that, in comparison to alternative algorithms, the suggested algorithm can cut the benefits to the attacker by about 15.23% and lower the time expenditures of the algorithm by around 7.86%.

S. Hammouti et al. [54] argue that all workflow simulators ignore the overhead of installing security services for sensitive data. This study suggests adding security services to the workflow simulator. Workflow execution security overheads may be assessed in seven ways. The extension was tested utilizing authentication, integrity verification, and encryption. The software successfully emulated a process by including security services for sensitive data and investigating how security measures affected timeliness, cost, and security metrics.

M. Farid et al. [55] suggested using the fault and intrusion-tolerant process scheduling method (FITSW) to improve process dependability. The proposed workflow system utilizes task executors of many virtual computers to accomplish workflow operations. FITSW employs a deadline partitioning technique to determine sub-deadlines for each sub-task following the duplication of each sub-task three times and the implementation of an intermediate data decision-making process. Thus, job scheduling with resource flow attains dynamism. The proposed approach enhances efficiency, preserves an organized workflow, and produces or reuses task executors. WorkflowSim tests and evaluates task completion rate, success rate, and completion time, which were conducted to examine the efficacy of FITSW. The data indicate that FITSW improves success rates by around 12%, reduces completion time by about 15.6% compared to the intrusion-tolerant scientific workflow ITSW system, and boosts the task completion rate by 6.2%.

The scheduling technique presented by H. Y. Shishido and colleagues [11] takes user annotation of workflow tasks depending on their sensitivity. They optimize scheduling with a multi-population genetic algorithm to save money and fulfill deadlines. Three workflow applications with sensitive job-to-data size ratios were extensively tested for cost, makepan, risk, and wastage. The methodology secured critical jobs more effectively and cheaply than earlier techniques in the literature.

S. Shahul Hammed and B. Arunkumar's [56] paper suggests an effective way to manage workflow while taking information value into account by classifying high-value and low-value information and constructing an algorithm that acts as a scheduler using the parallel implementation in the natural process of genetic algorithms (GA) with a secured framework for high-value information. The researchers say the suggested job performs significantly better than traditional methods in terms of execution time and overall cost.

M. Alam et al.'s [57] paper presented the Security Prioritized Heterogeneous Early Finish Time (SPHEFT) algorithm to maximize the security overhead and guarantee ratio of the workflow activities in cloud systems. In this case, SPHEFT gives jobs with higher security requirements, which are assigned to virtual machines with higher reliability and higher priorities. The traditional HEFT algorithm and SPHEFT are evaluated experimentally for various tasks. Experimental findings demonstrate that SPHEFT performs better regarding security overhead and has superior efficiency in raising the task guarantee ratio.

J. Lei et al.'s [58] paper proposed two scheduling techniques: simulated annealing (PSSA) and privacy and security-aware list scheduling (PSLS), aimed at addressing the constrained optimization problem at hand. PSLS allocates a user-defined deadline to each work and assigns them to a hybrid cloud resource that meets the specifications while minimizing costs. PSSA employs PSLS and simulated annealing to rearrange work lists for iterative enhancement. Simulation tests conducted reveal that PSLS surpasses four

existing algorithms in financial cost optimization overall, whereas PSSA exceeds PSLS further, although with increased runtime costs.

M. Farid et al. [59] proposed the approach to decision-making known as minimum weight optimization (MWO). Multi-objective algorithms use this technique to choose a set of permutations that offer the optimal compromise between conflicting objectives. By comparing several weights, (MWO) seeks to identify the optimum option and refines the search for the ideal answer iteratively. The study compared the suggested technique to Pareto dominance, multi-criteria decision-making (MCDM), linear normalization I and II, and weighted aggregated sum product evaluation using common scientific workflows with conflicting goals. MWO outperforms these methods.

M. Alam et al.'s [60] research uses the security-prioritized mapping approach for infrastructure as a service cloud computing to create an SPMWA model. Implementing a security-prioritized allocation technique under precedence limits should enhance workflow processing in dangerous situations. This strategy prioritizes secure workloads and allocates resources to more dependable virtual machines to reduce cloud system failure. Decreased task failures mean assigning tasks to dependable virtual machines with a high trust level for a comparative examination of task failures, failure probability, and makespan.

M. Alam et al. [61] improved the security measures in the cloud system by integrating security services into the workflow allocation. Consequently, they suggested a multi-constraint workflow allocation technique for cloud computing's heterogeneous activities. The recommended course of action is to satisfy deadlines and financial restrictions while minimizing the likelihood of risk and the expense of execution. The results demonstrate that the method consistently outperforms the current multi-constraint methods in the suggested system.

N. Soveizi et al. [22] suggested a method that centers on monitoring networks and cloud services to find security breaches that occur when workflows are being executed. This method chooses the best adaptation step to minimize the impact on the workflow after detection. He uses adaptive learning to identify the best adaptation action to reduce the uncertain cost of such adaptations and their possible effects on other jobs in the workflow. The metrics used to assess this technique are the efficacy of the detection process and the effects of the chosen modifications on the processes.

S. Mangalampalli et al. [62] introduced a deep reinforcement learning-based multi-objective workflow scheduling system. Dependencies determined all workflow priorities before mapping processes to VMs. Next, data center electricity costs influenced VM priority. The scheduler uses the Deep Q-Network architecture to schedule tasks dynamically based on VM and job priority. Simulations of real-time scientific procedures (LIGO, CyberShake, Montage, Epigenomics). MOPWSDRL was compared to the most sophisticated approaches, including ant colony optimization, heterogeneous earliest first deadline, and cat swarm optimization. The recommended MOPDSWRL outperformed state-of-the-art algorithms in makespan and energy use.

Hao Liang et al.'s [63] paper introduces an SMWE focus that harmonizes security and makespan by allocating functions to operational contexts and selectively implementing secure methodologies. Comprehensive assessments indicate that SMWE markedly enhances the security of serverless processes with minimal makespan expenditure.

Alper Alimoğlu and Can Özturan's [64] paper presented a scientific workflow execution manager based on the Ethereum blockchain that allocates workflows to cluster computing providers utilizing the Slurm workload manager. The solution enhances the eBlocBroker autonomous resource broker, a DAO-based decentralized coordinator, to facilitate distributed workflow execution through blockchain technology. This novel methodology is intended for e-Science, where scientific operations are extensively utilized.

Ginavanee A. and Dr. S. Prasanna's [65] paper proposes a Hybrid Healthcare Data Management System (HDMS) that integrates blockchain and cloud computing technology for secure health data management. The system uses the Ethereum blockchain for data integrity assurance, blockchain anchoring for scalable storage, Google Cloud integration, and compliance with Health Level 7 formatting criteria. It also uses decentralized identifiers and homomorphic encryption for secure computations. The system provides redundancy and resilience, outperforming the Optimized Blowfish Algorithm in encryption and decryption times. The goal is to improve patient outcomes through proper data handling, storage, analysis, and use. Table 2 shows a general analysis of selected papers.

Table 2. General analysis of selected papers.

Ref.	Year	Object/Aim	Algorithm	Advantage/ Contribution	Type of Security	Parameters/ Strategy	Limitations
[50]	2015	Optimizing for security requirements while adhering to budget constraints in cloud environments.	Security-Aware and Budget-Aware (SABA)	Balances security requirements with budget constraints.	Confidentiality, Integrity, and Availability (CIA)	Allocation of resources, balancing cost against security needs, security level, dictating.	The algorithm's effectiveness may decrease with an increase in the number of tasks or complexity of workflows.
[42]	2016	Minimize total cost while fulfilling timeline and risk rate restrictions.	PSO (particle swarm optimization)	Reduce the total workflow execution cost.	CIA	Cost, the deadline, and risk rate.	Limited scalability for large workflows.
[51]	2017	Reduce execution costs while meeting the deadline and risk rate requirements.	FFBAT (Firefly and Bat) algorithm	The proposed algorithm is based on the hybrid optimization approach, which combines the the Firefly and Bat algorithms.	Confidentiality (SEAL, RC4, Blowfish, Khufu/Khafre RC5, Rijndael DES, IDEA)	Cost, deadline, risk rate, security overhead.	High computational overhead.
[41]	2018	For minimizing workflow execution cost, preserving the privacy of critical tasks while respecting the deadline.	Hybrid Meta-heuristic	Optimized for privacy and execution costs.	CIA+ specific encryption protocols	Make-span, cost, and security risk.	Limited flexibility with heterogeneous workflows.

Table 2. *Cont.*

Ref.	Year	Object/Aim	Algorithm	Advantage/ Contribution	Type of Security	Parameters/ Strategy	Limitations
[52]	2018	Evaluate scheduling algorithms that preserve the security of sensitive data.	Meta-heuristic algorithms.	An extension for workflow simulators to support security services.	Encryption (SEAL, RC4, Blowfish, Knufu/Khafre, RC5, Rijndael, DES, IDEA) Integrity (MD4, MD5, RIPEMD, RIPEMD128, SHA1, RIPEMD160, TIGER) Authentication HMAC (MD5, SHA1), CBC, MAC-AES	Makespan, monetary cost, reliability, energy consumption, and risk.	There is another approach for securing workflow execution. It involves assigning sensitive tasks to private clouds and non-sensitive chores to public clouds.
[35]	2019	To find the best solution by comparing alternative weights, narrowing the search for an optimal solution through iterative refinement.	Multi-objective FR-MOS-MWO algorithm that combines FR-MOS and the minimum weight optimization method	A user-preference-based minimal weight optimization (MWO) method chooses and shows a feasible solution using the Pareto front's optimum set. The MWO-based multi-objective algorithm is compared to five standard workflow scheduling decision-making methods.	CIA	Reliability, cost, utilization of resources, risk probability, and time makespan.	Optimizing workflow scheduling using more than five QoS criteria. They will expand our energy-saving method to achieve fault tolerance while scheduling workflow in a hybrid environment.
[53]	2019	The goal is to optimize scheduling performance, minimize total execution costs, and balance resource load while adhering to constraints regarding deadlines and risk rates.	CPSO and GA	Assess the suggested algorithm in the context of extensive scientific procedures. Utilizing such tools has led to numerous significant discoveries. To examine the validity, integrity, or fallacy of the proposed procedure.	CIA	Makespan, risk rate, cost, load balance.	Schedules that optimize the entire budget may attract interest. Thus, both makepan and reliability can also be minimized. Using more effective optimization procedures will be beneficial in educating the reader about the latest trends in technique acquisition to address the task-resource scheduling problem.

Table 2. *Cont.*

Ref.	Year	Object/Aim	Algorithm	Advantage/ Contribution	Type of Security	Parameters/ Strategy	Limitations
[18]	2020	He proposes CLOSURE to enhance the challenges for attackers attempting to infiltrate virtual machines executing workflow sub-tasks.	HEFT	Propose the dynamic recycling and redeployment of VMs to alternate defense strategies during workflow execution; a task scheduling technique based on dynamic HEFT is introduced to enhance the speed of defense strategy transitions and improve workflow efficiency.		Reduce the attacker's benefits, decrease the time and costs.	A multiplayer game model is needed if there are multiple attackers.
[54]	2020	This module aims to safeguard sensitive data designated for storage in the cloud, their associated tasks, and data transferring between the (public and private) clouds.		The pre-scheduler designates each job or dataset for execution or storage in the "private or public" cloud. The security improvement module focuses on incorporating the necessary security services for the dataset while minimizing the expenses and overhead generated by these services. Post-Scheduler allocates each job or dataset for execution or storage in an appropriate virtual machine (VM) while adhering to budgetary and temporal limitations.	Confidentiality	Cost · security · budget · deadline.	The system can concurrently tackle the incorporation of security services at both the data and task levels, devise a more economical cryptographic method to fulfill security requirements, and pinpoint a scheduling plan that incorporates extra parameters and constraints, such as energy concerns.

Table 2. *Cont.*

Ref.	Year	Object/Aim	Algorithm	Advantage/ Contribution	Type of Security	Parameters/ Strategy	Limitations
[55]	2021	The goal is to create a FITSW workflow scheduling algorithm that enhances system failure and intrusion tolerance.	FITSW	Propose the fault and intrusion-tolerant workflow scheduling algorithm (FITSW).			
[11]	2021	This work aims to enhance the security needs of workflow tasks while minimizing the cost and makespan of workflows in public clouds.	MPGA	(1) Examination of the effects of implementing security services only for sensitive jobs using a task annotation methodology; (2) a scheduling algorithm that enhances task-VM allocation; and (3) metrics for calculating the ratio-risk and the inefficiency in guarding non-sensitive tasks.	CIA	Risk rate, cost, time, makespan.	
[56]	2022	The security provisioning in terms of validation, verification, and encryption allotted only to the sensitized tasks even reduces the cost and the time to a certain level compared to the other methods for scheduling.	GA	The authors describe a useful method for managing workflows that differentiates between high-value and low-value data. They also develop an algorithm that works as a scheduler by implementing it in parallel with the natural processes of a genetic algorithm (GA), ensuring that high-value information is kept safe.	CIA	Time of execution and total cost.	

Table 2. *Cont.*

Ref.	Year	Object/Aim	Algorithm	Advantage/ Contribution	Type of Security	Parameters/ Strategy	Limitations
[57]	2022	The algorithm aims to enhance both security and efficiency. Ultimately, the goal is to create a reliable and secure cloud system for workflow execution.	SPHEFT algorithm	The proposed SPHEFT, which integrates security awareness into workflow scheduling by considering security priorities during task allocation.	Confidentiality	Security overhead, deadline.	
[58]	2022	We employ a three-level privacy and security model and use encryption methods and hash functions to guarantee the security of cross-platform data transmission.	PSLS + PSSA (Simulated Annealing)	Handles privacy, minimizes cost, maintains deadlines	Confidentiality and integrity (IDEA, SHA, Blowfish).	Costs and performance, deadline.	Runtime increases with workload size.
[59]	2023	Reduce makespan, cost, and risk probability, and maximize resource utilization and dependability, which are concurrently considered alongside the interests of service providers and customers.	FR-MOS--MWO	Offers superior solutions relative to the extended Pareto dominance and alternative decision-making techniques utilizing the FR-OS algorithm.	CIA	Makespan, cost resource utilization, reliability, risk propalitty.	One may contemplate incorporating over five QoS criteria to enhance workflow scheduling. One may broaden this technique to diminish energy consumption and achieve fault tolerance while orchestrating the workflow in a hybrid environment.

Table 2. *Cont.*

Ref.	Year	Object/Aim	Algorithm	Advantage/ Contribution	Type of Security	Parameters/ Strategy	Limitations
[60]	2023	To optimize the failure probability and number of task failures as per the requirements of the cloud users.	(SPMWA) Security-Prioritized Workflow Allocation	(SPMWA) A paradigm for the IaaS cloud computing environment is suggested by incorporating the security-priori mapping scheme. Workflow processing performance in risky contexts is projected to be improved by implementing a security-prioritized allocation strategy under precedence restrictions. This model assigns jobs requiring high levels of security to more reliable virtual machines, thereby reducing the likelihood of cloud system failure.	CIA	Task failure, failure probability, and makespan.	High dependency on security metrics.
[61]	2024	To optimize the risk probability while satisfying the precedence constraints in workflow applications and to solve the allocation problem.	SCEDA	Propose a multi-constraints workflow allocation strategy for heterogeneous tasks in cloud computing.	Authentication	Risk probability and the execution cost, budget, and deadline constraints.	Future research related to this contribution will take into account the VM's termination delay, VMs located in various countries, and other security services offered by CSPs. Moreover, the extended work can include more than one objective with many constraints.

Table 2. *Cont.*

Ref.	Year	Object/Aim	Algorithm	Advantage/ Contribution	Type of Security	Parameters/ Strategy	Limitations
[22]	2024	The methodology emphasizes the surveillance of cloud services and networks to identify security breaches during workflow operations.	RL and MDP	They have proposed two ways to determine the optimal action to mitigate the consequences of such infractions. The initial technique identifies the most economical adaptation measure, whilst the subsequent one utilizes adaptive learning from previous responses.	CIA	Attack score and cost.	They will broaden their research to encompass additional possible enemies, including renters and their users, and implement security measures to counter these threats.
[62]	2024	Minimizing makespan and energy consumption.	MOPWSDRL	A prioritized multi-objective workflow scheduling algorithm was developed using a deep Q-learning network model.	Priorities of both tasks and VMs	Cost and makespan.	Particular attributes should be retrieved to enhance parameters, rendering the scheduler more resilient and efficient for various operations. A trust-based scheduling mechanism must be created in a multi-cloud context utilizing reinforcement learning techniques.
[63]	2024	Secure and makespan-oriented workflow execution in serverless computing.	SMWE (secure and makespan workflow execution)	Enhances security and reduces makespan.	Confidentiality, integrity	Selection based on task sensitivity and dynamic resource allocation.	Applicability to highly heterogeneous workflows.

Table 2. *Cont.*

Ref.	Year	Object/Aim	Algorithm	Advantage/ Contribution	Type of Security	Parameters/ Strategy	Limitations
[64]	2024	Autonomous blockchain-based workflow execution broker for e-science.	Autonomous blockchain workflow broker	Facilitates trustless collaboration in e-science environments using blockchain.	Confidentiality, integrity, non-repudiation	Integration of blockchain for trustless workflow execution; use of smart contracts for workflow orchestration.	High latency and resource demands of blockchain technology; scalability issues with large scale.
[65]	2024	Integration of Ethereum blockchain with cloud computing for secure healthcare data management.	Ethereum blockchain integration	Ensures secure healthcare data management by integrating Ethereum blockchain with cloud computing; provides immutability and decentralized security.	Confidentiality, integrity, availability (CIA)	Smart contract-enabled data access control; decentralized storage mechanisms to enhance security and prevent unauthorized access.	Ethereum's transaction throughput and high gas fees.

9. Discussion

The reviewed methods demonstrate significant advancements in cloud-based secure scientific workflow scheduling. Various strategies, such as multi-objective optimization, hybrid algorithms, game theory, and deep learning, have enhanced security, cost-efficiency, and performance. The integration of these techniques has led to better resource allocation, improved data protection, and optimized task execution in complex and dynamic cloud environments.

Despite these advancements, there remain areas for future research, particularly in integrating these approaches, addressing emerging security challenges, improving efficiency in diverse and dynamic cloud settings, and exploring novel algorithms. The literature review primarily focuses on scheduling workflows in cloud environments, emphasizing security. Most of the reviewed papers incorporate the CIA triad (confidentiality, integrity, and availability) to bolster the security aspects of workflow scheduling. Below is a breakdown of the key contributions from the literature:

- References [1–10]: These studies explore scientific workflow scheduling in cloud environments, covering essential topics like scheduling algorithms, challenges, and tools. They lay the foundation for understanding the complexities of workflow management in the cloud.
- References [11,12,15–18,21–24,46]: These papers delve into task scheduling with a focus on optimization techniques, including genetic algorithms, particle swarm optimization (PSO), and hyper-heuristics. They emphasize optimizing workflow performance and efficiency.
- References [13,14,26–29,31–38,40–45,47–53,55–59]: These sources address security concerns in workflow scheduling, highlighting the development of security-aware algorithms, cost-effectiveness strategies, and privacy-preserving techniques. They underscore the importance of integrating security measures into scheduling algorithms.
- References [19,30,39,56,57,63]: These papers explore broader cloud computing resource management aspects, such as multi-cloud environments, resource allocation strategies, scalability issues, and task priority.

- References [60–62]: This recent work investigates multi-objective workflow scheduling using deep reinforcement learning, a promising approach that adapts dynamically to evolving cloud environments and security threats.
- References [64,65]: This research represents an integrated blockchain platform scientific dataset such as healthcare, montage, epigenomic, and others. This level of security provided access management, transparency, and integrity.

The evaluated techniques exhibit considerable progress in safe cloud-based scientific workflow scheduling. Diverse methodologies, including multi-objective optimization, hybrid algorithms, game theory, and deep learning, have improved security, cost-effectiveness, and performance. These techniques emphasize resource allocation, data security, and efficient job execution in intricate and evolving cloud settings. A crucial element of these methodologies is their conformity to the CIA trinity (confidentiality, integrity, and availability), which is vital for safeguarding scientific processes. The triangle establishes the basis for assessing and formulating safe workflow scheduling methods, as outlined below:

- Confidentiality: Safeguarding sensitive workflow data from unauthorized access is essential, especially in private or classified data processes. Heuristic methods, including encryption techniques, offer rapid answers but may be deficient in scalability. Metaheuristic techniques, such as particle swarm optimization (PSO) and hybrid strategies like SPHEFT, include robust encryption protocols and allocate jobs to high-security virtual machines to maintain confidentiality while preserving efficiency.
- Integrity: Data integrity guarantees that workflow data remains unchanged during execution. Heuristic approaches frequently use hashing algorithms such as MD5 or SHA-256 to ensure data integrity. Metaheuristic methods, like genetic algorithms (GA), use sophisticated integrity-checking systems in optimization procedures.
- Availability: Workflow availability guarantees continuous access to resources and the prompt execution of tasks. Heuristic methods depend on fundamental failover mechanisms, but metaheuristics employ dynamic resource allocation to ensure availability during system failures. Hybrid methodologies such as ACISO employ resource redundancy and load balancing to guarantee the uninterrupted operation of processes in the face of disturbances.

This study examines the trade-offs between security and performance by linking the CIA principles with the analyzed algorithms. Heuristic methods offer efficiency and speed but may falter in complex scenarios. Metaheuristic approaches provide adaptation and flexibility; nevertheless, hybrid models balance security and efficiency. These techniques address the intricate challenges of secure process scheduling in cloud systems.

As illustrated in Figure 9, these studies collectively aim to minimize execution costs, meet critical deadlines, and maximize security within the framework of scientific workflows in cloud computing. The figure visually represents the trends and connections between the different approaches discussed in this survey.

From Section 8, each paper has limitations. These limitations are explained below in detail to provide a comprehensive understanding:

1. Scalability limitations: Numerous paper approaches encounter challenges in scalability when implemented in extensive, intricate operations. As the work quantity rises, expenses escalate considerably, resulting in possible inefficiencies.
2. Resource Heterogeneity: Although many methodologies tackle resource allocation and security levels of datasets, they frequently neglect the heterogeneity inherent in cloud systems. This issue may lead to inefficient work distribution and increased operational expenses.

3. Dynamic Threats: Several papers emphasize static security measures but lack adaptability to dynamic and evolving threats, such as zero-day vulnerabilities or other threats.
4. Energy Efficiency: The Day World focuses on energy consumption, a critical metric. So, few algorithms integrate energy-aware scheduling. This omission could lead to unsustainable cloud operations.

To effectively address these gaps, future research should integrate dynamic security models, advanced resource management techniques, and energy-efficient algorithms.

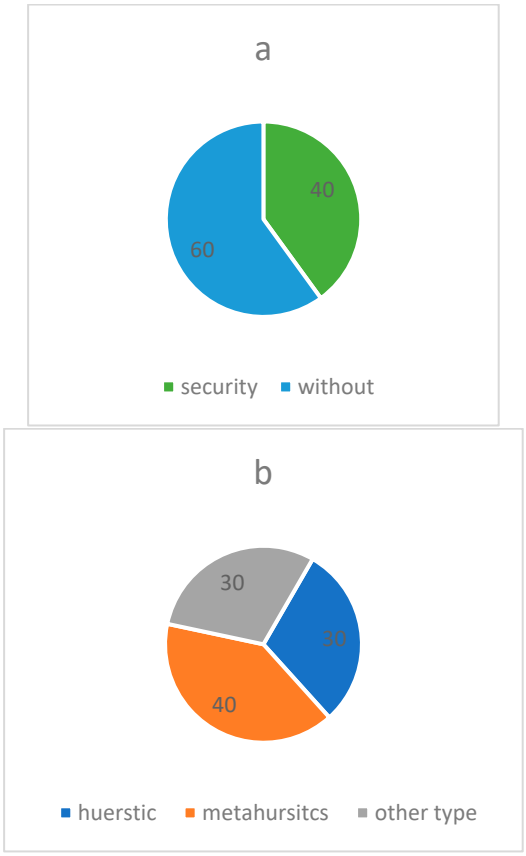

Figure 9. (**a**). Percentage of security and without security. (**b**) Percentage of methods used.

10. Conclusions and Future Work

Reviewing secure scientific workflow scheduling methods in cloud environments has highlighted significant advancements across various approaches. Researchers have successfully integrated multi-objective optimization, hybrid algorithms, game theory, and deep learning techniques to enhance security, cost-efficiency, and performance in workflow scheduling.

10.1. Conclusions

1. **Security Integration**: Most reviewed approaches incorporate security considerations such as confidentiality, integrity, and availability directly into the scheduling process. This integration ensures that workflows are executed securely without compromising the performance of cloud resources.
2. **Adaptability and Efficiency**: The use of diverse algorithms—from heuristic and metaheuristic approaches to advanced deep learning models—demonstrates the flexibility and robustness required to manage cloud environments' dynamic and often unpredictable nature.

3. **Trade-offs in Optimization**: These methodologies focus on balancing the trade-offs between performance metrics such as makespan and security measures. This balance is crucial in scientific workflows, where execution time and data protection are of paramount importance.
4. **Resource Management**: Techniques like VM utilization maximization and energy-aware scheduling address resource efficiency and environmental sustainability goals, catering to user needs and global ecological concerns.

10.2. Future Work

Future research should focus on further integration of these methodologies to address emerging security challenges. Areas for future exploration include:

- **Advanced AI Models:** The development of AI-driven models, particularly in reinforcement learning, that can dynamically adapt to new threats while optimizing workflows.
- **Multi-Cloud Interoperability:** Research into secure workflow scheduling across multiple cloud platforms, addressing interoperability issues while maintaining robust security standards.
- **Trust-Based Scheduling:** We are further developing trust-based models that evaluate the reliability and security history of cloud resources, allowing for more accurate and secure task assignments.

These areas hold the potential to significantly advance the field of secure scientific workflow scheduling, offering more robust, efficient, and scalable solutions for complex cloud environments.

Author Contributions: Author Contributions: Conceptualization, H.A.S., S.T.F.A.-J., E.T.Y. and O.A.A.; Methodology, H.A.S.; Validation, H.A.S., S.T.F.A.-J., E.T.Y. and O.A.A.; Formal Analysis, H.A.S., S.T.F.A.-J., E.T.Y. and O.A.A.; Investigation, H.A.S. and S.T.F.A.-J.; Resources, H.A.S.; Data Curation, H.A.S.; Writing—Original Draft Preparation, H.A.S.; Writing—Review and Editing, S.T.F.A.-J., E.T.Y. and O.A.A.; Visualization, S.T.F.A.-J. and E.T.Y.; Supervision, S.T.F.A.-J., E.T.Y. and O.A.A.; Project Administration, S.T.F.A.-J. and E.T.Y.; Funding Acquisition, O.A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: No new data were created or analyzed in this study.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Tarafdar, A.; Karmakar, K.; Das, R.K.; Khatua, S. Multi-criteria scheduling of scientific workflows in the Workflow as a Service platform. *Comput. Electr. Eng.* **2023**, *105*, 108458. [CrossRef]
2. Menaka, M.; Kumar, K.S. Workflow scheduling in cloud environment—Challenges, tools, limitations & methodologies: A review. *Meas. Sens.* **2022**, *24*, 100436. [CrossRef]
3. Rodriguez, M.A.; Buyya, R. Scientific Workflow Management System for Clouds. In *Software Architecture for Big Data and the Cloud*; Elsevier: Amsterdam, The Netherlands, 2017; pp. 367–387. [CrossRef]
4. Alejandra, M.; Sossa, R. Resource Provisioning and Scheduling Algorithms for Scientific Workflows in Cloud Computing Environments. Doctoral Dissertation, Department of Computing and Information Systems, University of Melbourne, Melbourne, Australia, 2016.
5. Ahmed, S.; Omara, F.A. A Modified Workflow Scheduling Algorithm for Cloud Computing Environment. *Int. J. Intell. Eng. Syst.* **2022**, *15*, 336–352. [CrossRef]
6. Wang, Y.; Guo, Y.; Guo, Z.; Liu, W.; Yang, C. Protecting scientific workflows in clouds with an intrusion tolerant system. *IET Inf. Secur.* **2020**, *14*, 157–165. [CrossRef]
7. Wang, Y.-W.; Wu, J.-X.; Guo, Y.-F.; Hu, H.-C.; Liu, W.-Y.; Cheng, G.-Z. Scientific workflow execution system based on mimic defense in the cloud environment. *Front. Inf. Technol. Electron. Eng.* **2018**, *19*, 1522–1536. [CrossRef]

8. Hammouti, S.; Yagoubi, B.; Makhlouf, S.A. Parametric Scientific Workflow Scheduling Algorithm in Cloud Computing. In Proceedings of the 2022 International Symposium on iNnovative Informatics of Biskra, ISNIB 2022, Biskra, Algeria, 7–8 December 2022; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2022. [CrossRef]
9. Shishido, H.Y.; Estrella, J.C.; Toledo, C.F.M.; Arantes, M.S. Genetic-based algorithms applied to a workflow scheduling algorithm with security and deadline constraints in clouds. *Comput. Electr. Eng.* **2018**, *69*, 378–394. [CrossRef]
10. Alouffi, B.; Hasnain, M.; Alharbi, A.; Alosaimi, W.; Alyami, H.; Ayaz, M. A Systematic Literature Review on Cloud Computing Security: Threats and Mitigation Strategies. *IEEE Access* **2021**, *9*, 57792–57807. [CrossRef]
11. Shishido, H.Y.; Estrella, J.C.; Toledo, C.F.M.; Reiff-Marganiec, S. Optimizing security and cost of workflow execution using task annotation and genetic-based algorithm. *Computing* **2021**, *103*, 1281–1303. [CrossRef]
12. Sarra, H. Job Scheduling in Cloud Computing Environment. Master's Thesis, University Mohamed Boudiaf-M'sila, M'Sila, Algeria, 2020.
13. Farid, M.; Latip, R.; Hussin, M.; Wati, A.; Hamid, A.; Senthil Kumar, A.V. Comparative Analysis of PSO-derived Workflow Scheduling Algorithms in Cloud Computing based on QoS Requirements. *Int. J. Adv. Res. Eng. Technol.* **2020**, *11*, 1387–1399. [CrossRef]
14. Kour, R.; Koul, S.; Kour, M. A Classification Based Approach for Data Confidentiality in Cloud Environment. In Proceedings of the 2017 International Conference on Next Generation Computing and Information Systems (ICNGCIS), Jammu, India, 11–12 December 2017; pp. 24–28. [CrossRef]
15. Pasha, F.; Natarajan, J. Research on secure workload execution scheme in heterogeneous cloud environment. *Indones. J. Electr. Eng. Comput. Sci.* **2023**, *29*, 1047–1054. [CrossRef]
16. Kaur, S.; Bagga, P.; Hans, R.; Kaur, H. Quality of Service (QoS) Aware Workflow Scheduling (WFS) in Cloud Computing: A Systematic Review. *Arab. J. Sci. Eng.* **2019**, *44*, 2867–2897. [CrossRef]
17. Rodriguez, M.A.; Buyya, R. A taxonomy and survey on scheduling algorithms for scientific workflows in IaaS cloud computing environments. *Concurr. Comput. Pract. Exp.* **2017**, *29*, e4041. [CrossRef]
18. Wang, Y.; Guo, Y.; Guo, Z.; Baker, T.; Liu, W. CLOSURE: A cloud scientific workflow scheduling algorithm based on attack–defense game model. *Future Gener. Comput. Syst.* **2020**, *111*, 460–474. [CrossRef]
19. Yang, L.; Xia, Y.; Ye, L.; Gao, R.; Zhan, Y. A Fully Hybrid Algorithm for Deadline Constrained Workflow Scheduling in Clouds. *IEEE Trans. Cloud Comput.* **2023**, *11*, 3197–3210. [CrossRef]
20. Alkhanak, E.N.; Lee, S.P. A hyper-heuristic cost optimisation approach for Scientific Workflow Scheduling in cloud computing. *Future Gener. Comput. Syst.* **2018**, *86*, 480–506. [CrossRef]
21. Jihad, A.A.; Al-Janabi, S.T.F.; Yassen, E.T. A survey on provisioning and scheduling algorithms for scientific workflows in cloud computing. *AIP Conf. Proc.* **2022**, *2400*, 020019. [CrossRef]
22. Soveizi, N.; Karastoyanova, D. Enhancing Workflow Security in Multi-cloud Environments Through Monitoring and Adaptation upon Cloud Service and Network Security Violations. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; Springer Science and Business Media Deutschland GmbH: Berlin/Heidelberg, Germany, 2024; pp. 157–175. [CrossRef]
23. Muñoz, A.; Maña, A.; González, J. Dynamic security properties monitoring architecture for cloud computing. In *Security Engineering for Cloud Computing: Approaches and Tools*; IGI Global: Hershey, PA, USA, 2012; pp. 1–18. [CrossRef]
24. Singh, P.; Dutta, M.; Aggarwal, N. A review of task scheduling based on meta-heuristics approach in cloud computing. *Knowl. Inf. Syst.* **2017**, *52*, 1–51. [CrossRef]
25. Manasrah, A.M.; Ali, H.B. Workflow Scheduling Using Hybrid GA-PSO Algorithm in Cloud Computing. *Wirel. Commun. Mob. Comput.* **2018**, *2018*, 1934784. [CrossRef]
26. Solanki, S.V. An Approach for Task Scheduling in Cloud Computing Using Hybrid Algorithm. Available online: http://www.webology.org (accessed on 23 May 2024).
27. Yin, L.; Sun, C.; Gao, M.; Fang, Y.; Li, M.; Zhou, F. Hyper-Heuristic Task Scheduling Algorithm Based on Reinforcement Learning in Cloud Computing. *Intell. Autom. Soft Comput.* **2023**, *37*, 1587–1608. [CrossRef]
28. Medara, R.; Singh, R.S. A Review on Energy-Aware Scheduling Techniques for Workflows in IaaS Clouds. *Wirel. Pers. Commun.* **2022**, *125*, 1545–1584. [CrossRef]
29. Kołodziej, J.; Khan, S.U.; Wang, L.; Kisiel-Dorohinicki, M.; Madani, S.A.; Niewiadomska-Szynkiewicz, E.; Zomaya, A.Y.; Xu, C.-Z. Security, energy, and performance-aware resource allocation mechanisms for computational grids. *Future Gener. Comput. Syst.* **2014**, *31*, 77–92. [CrossRef]
30. Prakash, V.; Bawa, S.; Garg, L. Multi-Dependency and Time Based Resource Scheduling Algorithm for Scientific Applications in Cloud Computing. *Electronics* **2021**, *10*, 1320. [CrossRef]
31. Ahmad, S.; Mehfuz, S.; Urooj, S.; Alsubaie, N. Machine learning-based intelligent security framework for secure cloud key management. *Clust. Comput.* **2024**, *27*, 5953–5979. [CrossRef]

32. Ahmad, S.; Mehfuz, S.; Mebarek-Oudina, F.; Beg, J. RSM analysis based cloud access security broker: A systematic literature review. *Clust. Comput.* **2022**, *25*, 3733–3763. [CrossRef] [PubMed]
33. Sotiriadis, S.; Bessis, N.; Antonopoulos, N.; Anjum, A. SimIC: Designing a New Inter-cloud Simulation Platform for Integrating Large-Scale Resource Management. In Proceedings of the 2013 IEEE 27th International Conference on Advanced Information Networking and Applications (AINA), Barcelona, Spain, 25–28 March 2013; pp. 90–97. [CrossRef]
34. Albtoush, A.; Yunus, F.; Almi'ani, K.; Noor, N.M.M. Structure-Aware Scheduling Methods for Scientific Workflows in Cloud. *Appl. Sci.* **2023**, *13*, 1980. [CrossRef]
35. Wang, Y.; Guo, Y.; Guo, Z.; Liu, W.; Yang, C. Securing the Intermediate Data of Scientific Workflows in Clouds with ACISO. *IEEE Access* **2019**, *7*, 126603–126617. [CrossRef]
36. Ma, J.; Cao, J.; Zhang, Y. Efficiently supporting secure and reliable collaboration in scientific workflows. *J. Comput. Syst. Sci.* **2010**, *76*, 475–489. [CrossRef]
37. Wang, Y.; Guo, Y.; Wang, W.; Liang, H.; Huo, S. INHIBITOR: An intrusion tolerant scheduling algorithm in cloud-based scientific workflow system. *Future Gener. Comput. Syst.* **2021**, *114*, 272–284. [CrossRef]
38. Francis, A.O.; Emmanuel, B.; Zhang, D.; Zheng, W.; Qin, Y.; Zhang, D. Exploration of Secured Workflow Scheduling Models in Cloud Environment: A Survey. In Proceedings of the 2018 6th International Conference on Advanced Cloud and Big Data, CBD 2018, Lanzhou, China, 12–15 August 2018; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2018; pp. 71–76. [CrossRef]
39. Soveizi, N.; Turkmen, F.; Karastoyanova, D. Security and privacy concerns in cloud-based scientific and business workflows: A systematic review. *Future Gener. Comput. Syst.* **2023**, *148*, 184–200. [CrossRef]
40. Sheikh, A.; Munro, M.; Budgen, D. Systematic Literature Review (SLR) of Resource Scheduling and Security in Cloud Computing. 2019. Available online: www.ijacsa.thesai.org (accessed on 3 May 2024).
41. Shishido, H.Y.; Estrella, J.C.; Toledo, C.F.M.; Reiff-Marganiec, S. (WIP) Tasks Selection Policies for Securing Sensitive Data on Workflow Scheduling in Clouds. In Proceedings of the 2018 IEEE International Conference on Services Computing, SCC 2018—Part of the 2018 IEEE World Congress on Services, San Francisco, CA, USA, 2–7 July 2018; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2018; pp. 233–236. [CrossRef]
42. Li, Z.; Ge, J.; Yang, H.; Huang, L.; Hu, H.; Hu, H.; Luo, B. A security and cost aware scheduling algorithm for heterogeneous tasks of scientific workflow in clouds. *Future Gener. Comput. Syst.* **2016**, *65*, 140–152. [CrossRef]
43. Hussain, M.; Wei, L.-F.; Rehman, A.; Abbas, F.; Hussain, A.; Ali, M. Deadline-constrained energy-aware workflow scheduling in geographically distributed cloud data centers. *Future Gener. Comput. Syst.* **2022**, *132*, 211–222. [CrossRef]
44. Chen, Z.; Zhao, X.; Lin, B. Fuzzy Theory-Based Data Placement for Scientific Workflows in Hybrid Cloud Environments. *Discret. Dyn. Nat. Soc.* **2020**, *2020*, 8105145. [CrossRef]
45. Marcon, D.S.; Bittencourt, L.F.; Dantas, R.; Neves, M.C.; Madeira, E.R.M.; Fernandes, S.; Kamienski, C.A.; Barcelos, M.P.; Gaspary, L.P.; da Fonseca, N.L.S. Workflow specification and scheduling with security constraints in hybrid clouds. In Proceedings of the 2013 IEEE Latin America Conference on Cloud Computing and Communications (LatinCloud), Maceio, Brazil, 9–10 December 2013; pp. 29–34. [CrossRef]
46. Sujana, J.A.J.; Revathi, T.; Priya, T.S.S.; Muneeswaran, K. Smart PSO-based secured scheduling approaches for scientific workflows in cloud computing. *Soft Comput.* **2019**, *23*, 1745–1765. [CrossRef]
47. Singh, H.; Tyagi, S.; Kumar, P.; Gill, S.S.; Buyya, R. Metaheuristics for scheduling of heterogeneous tasks in cloud computing environments: Analysis, performance evaluation, and future directions. *Simul. Model. Pract. Theory* **2021**, *111*, 102353. [CrossRef]
48. Manzoor, M.F.; Abid, A.; Farooq, M.S.; Azam, N.A.; Farooq, U. Resource Allocation Techniques in Cloud Computing: A Review and Future Directions. *Elektron. Elektrotechnika* **2020**, *26*, 40–51. [CrossRef]
49. Boroumand, A.; Shirvani, M.H.; Motameni, H. A heuristic task scheduling algorithm in cloud computing environment: An overall cost minimization approach. *Clust. Comput.* **2025**, *28*, 137. [CrossRef]
50. Zeng, L.; Veeravalli, B.; Li, X. SABA: A security-aware and budget-aware workflow scheduling strategy in clouds. *J. Parallel Distrib. Comput.* **2015**, *75*, 141–151. [CrossRef]
51. Arunarani, A.R.; Manjula, D.; Sugumaran, V. FFBAT: A security and cost-aware workflow scheduling approach combining firefly and bat algorithms. In *Concurrency and Computation: Practice and Experience*; John Wiley and Sons Ltd.: Hoboken, NJ, USA, 2017. [CrossRef]
52. Shishido, H.Y.; Estrella, J.C.; Toledo, C.F.M.; Reiff-Marganiec, S. A CloudSim Extension for Evaluating Security Overhead in Workflow Execution in Clouds. In Proceedings of the 2018 Sixth International Symposium on Computing and Networking (CANDAR), Takayama, Japan, 27–30 November 2018; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2018; pp. 174–180. [CrossRef]
53. Abdali, A.; Nia, S. A new optimization method for security-constrained workflow scheduling. *Indian J. Comput. Sci. Eng.* **2019**, *10*, 8–25. [CrossRef]

54. Hammouti, S.; Yagoubi, B.; Makhlouf, S.A. Workflow Security Scheduling Strategy in Cloud Computing. In *Modelling and Implementation of Complex Systems*; Springer: Cham, Switzerland, 2020.
55. Farid, M.; Latip, R.; Hussin, M.; Hamid, N.A.W.A. A fault-intrusion-tolerant system and deadline-aware algorithm for scheduling scientific workflow in the cloud. *PeerJ Comput. Sci.* **2021**, *7*, e747. [CrossRef]
56. Hammed, S.; Arunkumar, B. Efficient workflow scheduling in cloud computing for security maintenance of sensitive data. *Int. J. Commun. Syst.* **2022**, *35*, e4240. [CrossRef]
57. Alam, M.; Shahid, M.; Mustajab, S. Security Prioritized Heterogeneous Earliest Finish Time Workflow Allocation Algorithm for Cloud Computing. In *Congress on Intelligent Systems*; Lecture Notes on Data Engineering and Communications Technologies, Volume 114; Springer Science and Business Media Deutschland GmbH: Berlin/Heidelberg, Germany, 2022; pp. 233–246. [CrossRef]
58. Lei, J.; Wu, Q.; Xu, J. Privacy and security-aware workflow scheduling in a hybrid cloud. *Future Gener. Comput. Syst.* **2022**, *131*, 269–278. [CrossRef]
59. Farid, M.; Lim, H.S.; Lee, C.P.; Latip, R. Scheduling Scientific Workflow in Multi-Cloud: A Multi-Objective Minimum Weight Optimization Decision-Making Approach. *Symmetry* **2023**, *15*, 2047. [CrossRef]
60. Alam, M.; Shahid, M.; Mustajab, S. Security prioritized multiple workflow allocation model under precedence constraints in cloud computing environment. *Clust. Comput.* **2023**, *27*, 341–376. [CrossRef]
61. Alam, M.; Shahid, M.; Mustajab, S.; Ahmad, F.; Haidri, R.A. Security Driven Cost-Effective Deadline Aware Workflow Allocation Strategy in Cloud Computing Environment. In Proceedings of the 2023 3rd International Conference on Technological Advancements in Computational Sciences (ICTACS), Tashkent, Uzbekistan, 1–3 November 2023; pp. 1–7. [CrossRef]
62. Mangalampalli, S.; Hashmi, S.S.; Gupta, A.; Karri, G.R.; Rajkumar, K.V.; Chakrabarti, T.; Chakrabarti, P.; Margala, M. Multi Objective Prioritized Workflow Scheduling Using Deep Reinforcement Based Learning in Cloud Computing. *IEEE Access* **2024**, *12*, 5373–5392. [CrossRef]
63. Liang, H.; Zhang, S.; Liu, X.; Cheng, G.; Ma, H.; Wang, Q. SMWE: A Framework for Secure and Makespan-Oriented Workflow Execution in Serverless Computing. *Electronics* **2024**, *13*, 3246. [CrossRef]
64. Alimoğlu, A.; Özturan, C. An autonomous blockchain-based workflow execution broker for e-science. *Clust. Comput.* **2024**, *27*, 10235–10244. [CrossRef]
65. Prasanna, G.A.S. Integration of Ethereum Blockchain with Cloud Computing for Secure Healthcare Data Management System. *J. Electr. Syst.* **2024**, *20*, 111–124. [CrossRef]