Volume 9, Issue 5, May – 2024 International Journal of Innovative Science and Research Technology
ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY148

# Blockchain-Based Solution for Supply Chain Data Integrity

Kakumanu Sai Dasarath
Department of Computer Science and Engineering
Koneru Lakshmaiah Education Foundation
Vaddeswaram, Andhra Pradesh, India

Kotti Durga Sai Pranith
Department of Computer Science and Engineering
Koneru Lakshmaiah Education Foundation
Vaddeswaram, Andhra Pradesh, India

Kotra Leela Balaji
Department of Computer Science and Engineering
Koneru Lakshmaiah Education Foundation
Vaddeswaram, Andhra Pradesh, India

B.V.A Bheema Sena Reddy
Department of Computer Science and Engineering
Koneru Lakshmaiah Education Foundation
Vaddeswaram, Andhra Pradesh, India

Dr. Garikapati Bindu
Department of Computer Science and Engineering
Koneru Lakshmaiah Education Foundation
Vaddeswaram, Andhra Pradesh, India

**Abstract:- Data plays a crucial role in today's world, guiding business decisions across various computer-assisted activities. Maintaining the integrity of data is vital, as any tampering could have serious consequences for important business decisions. This concern is particularly significant in cloud computing settings, where data owners have limited control over key aspects like physical storage and access control. Blockchain technology has emerged as an intriguing solution to address data integrity concerns. With its inherent properties, blockchain offers promising avenues for ensuring data integrity. However, some challenges need to be overcome, such as low throughput, high latency, and stability issues, which currently limit the practical implementation of blockchain-based solutions. we focus on a case study from the European SUNFISH project, which aims to develop a secure cloud federation platform for the public sector. We examine the specific data integrity requirements in cloud computing environments and identify the research questions that need to be addressed to implement blockchain-based databases effectively. We start by outlining the open research questions and the inherent difficulties associated with addressing them. Then, we propose a preliminary design for a blockchain-based database tailored to cloud computing environments. This design aims to leverage the strengths of blockchain technology while addressing the challenges unique to cloud computing. By addressing these research questions and proposing practical solutions, we aim to pave the way for the adoption of blockchain-based databases in cloud computing environments. This has the potential to enhance data integrity and security, ultimately benefiting organizations operating in the cloud.**

***Keywords***:- *Cloud Computing, Blockchain.*

## I. INTRODUCTION

Data has become an essential part of our lives today. It's not just important for businesses; it's also crucial in various fields like finance, insurance, healthcare, education, and government services. With the increasing reliance on computers to assist us in our daily tasks, the trustworthiness of data has become more important than ever. Because data plays such a critical role, it has also become a prime target for cyber-attacks. These attacks aim to disrupt the basic properties that data should have to be considered trustworthy: confidentiality, integrity, and availability. In simpler terms, data needs to be kept private, accurate, and accessible when needed. When these properties are compromised, it can have serious consequences for individuals and organizations alike. Cyber-attacks can seriously mess with our trust in data. Let's break it down: when an attack messes with the availability of data, it's like putting them on hold temporarily. You can't get to them right away, but once things are back to normal, you can resume your operations. If an attack breaches confidentiality, it's like spilling the beans on private stuff. Once that happens, there's no going back. The original data might still be there, but now everyone knows your secrets. This can lead to all sorts of problems, like financial losses for businesses if sensitive information gets out. But messing with the integrity of data is the worst of the bunch. It's like someone sneaking in and messing with the information itself. They might delete important stuff to cover their tracks or change things to trick people into doing something they shouldn't. Back in 2015, there was a huge cyber-attack on more than 100 banks around the world. The attackers managed to steal around $1 billion by messing with the account balances. This shows just how serious these attacks can be and why we need to protect our data at all costs. When data integrity is compromised, it's like losing something valuable forever. There's no way to bring back the original

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY148

data once it's been tampered with. That's why we're focusing on keeping data intact in this paper, rather than just keeping it secret or available. In cloud computing, where data is stored and accessed remotely, things get even trickier. Data owners don't have much control over where their data goes or who gets to see it. However, despite the risks, many organizations are outsourcing their data to the cloud to save on maintenance and storage costs. This makes ensuring data integrity in the cloud more important than ever. To keep data safe, we use two main tools: cryptographic techniques and data replication. Cryptographic tools like digital signatures help verify the authenticity of data. If someone tries to tamper with data, these signatures can quickly catch the changes. But if attackers manage to break through the cryptographic defences, their changes can be hard to spot. That's where data replication comes in. By storing copies of data on multiple nodes, we make it much harder for attackers to mess with data integrity. Even if they compromise one copy, they'd have to do the same to all the others without getting caught. This strategy is commonly used in cloud computing, where there are plenty of distributed storage resources to work with. Relying solely on replication isn't foolproof. Cloud providers themselves could team up with attackers to undermine data integrity. To combat this threat and avoid blindly trusting cloud providers, we propose using blockchain technology. Blockchain, the technology behind cryptocurrencies like Bitcoin, is all about transparency and security. By using blockchain, we can create a distributed, secure database for cloud computing environments. This database would be resistant to tampering and collusion, making it a powerful tool for ensuring data integrity in the cloud. Imagine a blockchain as a giant, shared database spread out across thousands of computers owned by different people or groups. Initially, it was created as a public ledger to keep track of Bitcoin transactions. But lately, it's been getting a lot of attention for its amazing features, like making sure everyone agrees on the same information, keeping data safe and permanent, and making it impossible to deny past actions. One of the most important things about blockchain is that once information is stored in it for a certain amount of time, it becomes nearly impossible to change. But here's the tricky part: that "certain amount of time" isn't set in stone. For example, with Bitcoin transactions, it's considered safe after about an hour. But for things like cloud computing, waiting that long just isn't practical. Even though this time issue makes blockchain tricky to use for some things, its built-in features like replication and distribution make it useful in cloud settings. In this paper, we're going to talk about the challenges of using blockchain as it is, and then we'll suggest some ways to make it work better for cloud computing. we'll talk about the problems with using blockchain right out of the box, like how long it takes to guarantee data integrity, how it can slow things down, and how it's not always stable. Then, we'll introduce a new kind of blockchain-based database that balances strong data integrity with good performance and stability. Picture a massive, interconnected database spread across thousands of computers, each belonging to different individuals or organizations. Originally, this colossal network was conceived as a ledger to record Bitcoin transactions. However, in recent times, it has captured widespread attention for its remarkable capabilities, such as ensuring consensus among users, safeguarding data in an immutable manner, and rendering past actions irrefutable. A fundamental aspect of blockchain technology lies in its ability to preserve data integrity over time. Once information is recorded within the blockchain, altering or tampering with it becomes exceedingly difficult. However, there's a catch: the duration for which data must remain untouched to achieve this level of security isn't fixed. Take Bitcoin transactions, for instance; they're typically deemed secure after approximately an hour. Yet, in contexts like cloud computing, waiting for such extended periods isn't practical or feasible. Despite this temporal challenge, the inherent characteristics of blockchain, including replication and distribution, render it exceptionally advantageous in cloud environments. Within the pages of this paper, we aim to delve into the obstacles associated with deploying blockchain in its raw form, while also proposing innovative solutions to enhance its suitability for cloud computing applications. Our discussion will revolve around the inherent limitations of using blockchain in its conventional state. We'll examine issues such as the time required to ensure data integrity, the potential performance bottlenecks it introduces, and its occasional lack of stability. Subsequently, we'll introduce a novel approach—a revamped blockchain-based database—that strikes a harmonious balance between robust data integrity, optimal performance, and steadfast stability.

## II. RELATED WORK

Ensuring data integrity is a significant concern in the realm of computing systems, particularly within cloud environments where users entrust their data to external servers. For individuals with limited computing resources, the task of verifying data integrity can be daunting, especially when dealing with vast amounts of data that need to be downloaded for inspection. A groundbreaking model introduced by Ateniese et al. [1] offers a solution by allows a user to verify the integrity of their outsourced data on a single server without the need to retrieve them—a significant advancement in cloud computing. Remote Data Auditing (RDA) emerges as another viable solution for cloud environments, offering users the ability to audit their outsourced data with the assistance of a trusted third party. This approach effectively alleviates the computational burden on the user, ensuring both security and efficiency in data integrity verification processes. Various RDA techniques have been proposed to enhance security and efficiency, as highlighted in the survey conducted by Sookhak et al. [11]. However, these methodologies operate under the assumption that the third-party intermediary is inherently trustworthy. Should this trust be violated, the integrity of the data cannot be guaranteed, posing a significant risk to users. To address this challenge, our solution leverages the immutable nature of blockchain technology, specifically utilizing Proof of Work (PoW) consensus mechanisms. By harnessing the blockchain's inherent immutability feature, we can ensure data integrity even in a trustless environment. This innovative approach offers a robust solution to the problem of data integrity, providing users with confidence in the security and reliability of their outsourced data, regardless of the

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY148

trustworthiness of intermediaries. The first-layer blockchain we're utilizing to boost performance draws inspiration from Bitcoin-NG [5], a modified version of the Bitcoin protocol designed to enhance efficiency. However, this optimization comes at the cost of certain security guarantees, as data integrity is only assured under the assumption of a majority of honest miners. Similar approaches to blockchain-based databases have been explored in existing literature. One such example is BigchainDB [7], which offers a NoSQL-like storage solution built on blockchain technology, incorporating its consensus mechanism. Like Bitcoin-NG, the primary objective of BigchainDB is to enhance performance by sacrificing some security assurances. In scenarios where a majority of miners are malicious, both Bitcoin-NG and BigchainDB fail to uphold data integrity. In contrast, our solution distinguishes itself by ensuring data integrity even when faced with a majority of malicious miners. As demonstrated in Section 5.3, our system can maintain integrity even when all but one miner are colluding maliciously, indicating resilience against attacks from weaker adversaries, such as scenarios where a majority of miners (potentially fewer, assuming at least three miners) are malicious but not cooperating among themselves. Another notable endeavor aimed at improving performance while maintaining security is RSCoin [3], a cryptocurrency framework introducing a certain degree of centralization. In this setup, a central bank retains full control over the monetary supply, augmented by mintettes—a distributed group of authorities tasked with preventing double-spending. The authors of RSCoin illustrate how their approach, underpinned by a robust consensus algorithm, achieves improved performance and integrity assurances. However, compared to our solution, RSCoin exhibits two key limitations: a degree of centralization and integrity guarantees contingent upon a majority of honest mintettes. While RSCoin offers notable advancements in performance and security, its reliance on centralization and the need for a majority of honest mintettes to ensure integrity present notable drawbacks when compared to our approach.

- *Problem Statement*

In the world of cloud federation, there are numerous threats to data integrity that we need to watch out for. Specifically, we're concerned about the database holding the crucial governance data of a federation. Any corruption in this data could have serious consequences for the entire federation's security. The threats we're looking at range from intentional changes to the data by malicious actors, to updates being made without all the relevant members being notified. It's essential to address these threats to ensure the stability and security of the federation as a whole.

- *Data Collection*

In today's rapidly evolving business landscape, both public and private companies are increasingly seeking ways to enhance interoperability and collaboration among their existing cloud systems. This trend is highlighted in reports like the one from ENISA [4], which underscores the importance of federating different cloud systems to achieve common goals. However, alongside the technical challenges involved in creating and managing these federations, there are significant security concerns that must be addressed, particularly regarding the protection of sensitive data and the assurance of data integrity. To tackle these security challenges head-on, the EU SUNFISH project is working on developing a distributed, democratic cloud federation platform designed to prioritize data security from the outset. This platform, known as Federation-as-a-Service (FaaS) [9], offers a novel approach to securely creating and managing cloud data and services. With advanced data security features and innovative governance principles, FaaS aims to establish a secure and transparent environment for cloud federation. While the specifics of the data security services offered by FaaS are detailed elsewhere [12, 13], our focus here is on the critical role of data integrity in governing federations. At the core of cloud federations is the concept of sharing services among members through regulated, secure interactions across different clouds. These interactions are governed by specific contracts that outline the terms of service usage. For example, a service provider may stipulate that only certain consumers can access their service, and that any outputs must be anonymized for privacy protection. Given the highly sensitive nature of the data managed within cloud federations, such as personal and medical data in the public sector, FaaS must provide robust assurances regarding contract compliance. In addition to enforcing these contracts at runtime, FaaS must ensure the integrity of the contracts themselves, ensuring they remain tamperproof and that all relevant members are aware of their terms. Furthermore, to establish irrefutable evidence of contract enforcement, FaaS must monitor all intercloud interactions and maintain logs with strong integrity guarantees. These measures are essential for maintaining trust and accountability within cloud federations, safeguarding both the data and the integrity of the federation as a whole.

## III. METHODOLOGY

The blockchain is a relatively new technology that has emerged in recent years, initially gaining prominence as the underlying system for the Bitcoin cryptocurrency [8]. Essentially, it operates as a digital ledger comprising a series of interconnected blocks, each containing transaction records. These blocks are duplicated across multiple nodes within a peer-to-peer network. Transactions within the blockchain typically involve the exchange of assets, such as cryptocurrencies like Bitcoin, between pseudonymous parties. The process of recording transactions and adding them to the chain is decentralized, with specific nodes known as miners responsible for validating and incorporating new blocks into the blockchain. Miners employ specialized algorithms, known as the mining process, to achieve consensus among themselves regarding the validity of newly created blocks. In the case of Bitcoin and Ethereum [14], the original mining process relies on a mechanism called proof of work (PoW). This involves performing computationally intensive cryptographic calculations, regulated by the blockchain difficulty parameter, to validate transactions and create new blocks. Once a miner successfully creates a new block, it is broadcasted to all other nodes in the network, who then recognize it as the latest addition to the blockchain. Subsequently, miners commence the process of mining new

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY148

blocks to append to the chain. It's important to note that in a permissionless blockchain, such as Bitcoin, any node can participate as a miner without requiring authorization. Conversely, in a permissioned blockchain, there exists an authentication and authorization layer governing which nodes are eligible to participate as miners. In practical terms, the process unfolds as follows, when a miner successfully creates a new block, it becomes part of the blockchain. However, in situations where multiple miners concurrently add blocks, a temporary fork may occur, resulting in the creation of parallel chains. However, due to the inherent design of blockchain protocols, miners consistently prioritize the longest chain, swiftly resolving any temporary forks and ensuring the integrity and continuity of the blockchain. The blockchain, a relatively new technology that emerged in recent years, initially gained prominence as the public ledger supporting Bitcoin cryptocurrency transactions [8]. At its core, it comprises a series of interconnected blocks containing transaction records, which are duplicated across nodes in a peer-to-peer network. These records serve as proof of transactions conducted between anonymous entities, involving cryptocurrencies like Bitcoin or other types of assets. The decentralized network relies on designated nodes, known as miners, to manage the collection of transactions and their incorporation into chain blocks. Miners employ specific block construction methods, referred to as the mining process, to achieve consensus among all participants regarding the creation of new blocks. Bitcoin operates as a permissionless blockchain, meaning any node can participate in the mining process without restrictions. Conversely, permissioned blockchains incorporate an authentication and authorization layer for miners. The original mining process, employed by Bitcoin and Ethereum [14], is based on the proof of work (PoW) concept. This involves computationally intensive hashing tasks regulated by blockchain difficulty, which controls the average time required for miners to create a new block. Once a miner successfully creates a new block, it is broadcasted to all other miners, who recognize it as the latest addition to the chain and begin mining subsequent blocks. The PoW-based blockchain offers numerous captivating features related to data integrity, stemming from the mining process and the full replication of the blockchain across a vast number of nodes. When a block becomes part of the chain, consensus among miners ensures agreement on its contents, rendering it practically non-repudiable and persistent. This means that unless an attacker controls the majority of miners' hash power, allowing them to create a fork in the chain, transactions included in the blockchain are considered secure. Assuming the majority of hash power is in the hands of honest miners, the probability of a fork occurring decreases exponentially with its depth. For users, this instils confidence that waiting for a short period, typically six blocks in Bitcoin, ensures their transactions are permanently recorded with high certainty. One big problem with PoW-based blockchains is that they're just not very fast. This slowness comes from two main things. First, imagine you're in a huge crowd and you want to tell everyone some news. It's going to take time for your message to spread to every person in that crowd, right? That's kind of how it works with new information, like transactions, on a blockchain. It takes time for that information to spread to all the computers in the network. Second, there's this thing called Proof of Work (PoW). It's like a big puzzle that computers have to solve to confirm transactions. But solving this puzzle takes a lot of computing power and time. So, when you want to make a transaction on a blockchain, it can take a pretty long time before it's confirmed and officially added to the chain. For example, with Bitcoin, it takes about 10 minutes for an average transaction to get confirmed, and the system can only handle about 7 transactions per second. whether it's stable over time. Think of it like a roller coaster ride. Bitcoin has been running okay so far, but experts aren't exactly sure why. They're also not sure if it'll keep working well in the future or how long it will last. This uncertainty makes it hard to know if we can rely on blockchain technology in the long run. Plus, the way PoW-based blockchains use cryptocurrencies to reward people for helping keep the system running means they're really sensitive to changes in the market. It's like if the price of your favorite snack kept going up and down all the time – it would be hard to plan your snacks for the week, right? Similarly, if the value of these cryptocurrencies goes up and down a lot, it can cause big problems for how well the blockchain works over time. The first layer is all about speed and efficiency. We're using a lightweight system that can quickly store evidence of every little thing that happens in our database. It's like having a really fast secretary jotting down every detail of a meeting. But here's the catch: this layer isn't super strong when it comes to making sure the data stays safe and unchanged. That's where the second layer comes in. Think of it as the security guard of our database. It's based on something called Proof of Work, which is like a tough puzzle that needs to be solved before anything can be added to the database. This layer is all about making sure the evidence stored in the first layer is super secure and can't be tampered with. But, because it's so focused on security, it's not as speedy as the first layer. Now, here's where things get interesting. By cleverly combining these two layers, we get the best of both worlds. We get the speed and efficiency of the first layer, along with the rock-solid security of the second layer. It's like having a speedy secretary who's backed up by a super-strong security guard. This way, we not only improve overall performance but also make sure our data is as safe as can be. In our plan for a better database, we're bringing in blockchain technology to make sure everything stays safe and reliable. Blockchain isn't just about cryptocurrencies like Bitcoin – it can also be used to keep track of important data and make sure it's not tampered with. With blockchain, we can store evidence of every operation on the database in a way that can't be denied or changed. This means we can guarantee the integrity of the database and have full control over the data, all without relying on a central authority. Our database proposal is designed to work seamlessly within FaaS federations, where different cloud systems come together to share services and resources. Picture it like a team of clouds working together to keep a big database running smoothly. Each cloud member sends commands to the database through something called the Database Interface. These commands are first recorded by the first-layer blockchain, which acts like a digital notary, making sure everything is documented properly. Then, the commands are carried out on the distributed database replicas, which are like copies of the main database spread

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY148

out across different clouds. In our setup, each member cloud has its own dedicated miner, which is like a digital guardian responsible for verifying and recording transactions on the blockchain. This firstlayer blockchain is permissioned, meaning only approved participants can take part. This setup ensures that every operation on the database is securely logged and verified, giving us peace of mind knowing that our data is safe and trustworthy. In our setup, we make sure everyone agrees on the transactions using something called the mining rotation consensus mechanism. Here's how it works: we divide time into rounds, and for each round, we pick one miner to be the leader. This leader's job is to take in new transactions, put their digital signature on them, and then share them with the other miners. Once all the miners have signed off on the transactions, they become part of the blockchain. Each miner then adds these transactions to their own record of transactions and applies them to their copy of the database. We do this using a technique called blockchain anchoring. This technique is like tying a rope between the two layers to keep them connected. At regular intervals, we take a snapshot of the first-layer blockchain up to the current point, kind of like taking a picture of a puzzle at different stages of completion. Then, we send this snapshot, called a witness transaction, to the second-layer blockchain, where it's stored permanently and can't be changed. These snapshots act as evidence that proves the integrity of the data stored in the first-layer blockchain, like breadcrumbs that show us where we've been and how we got there.

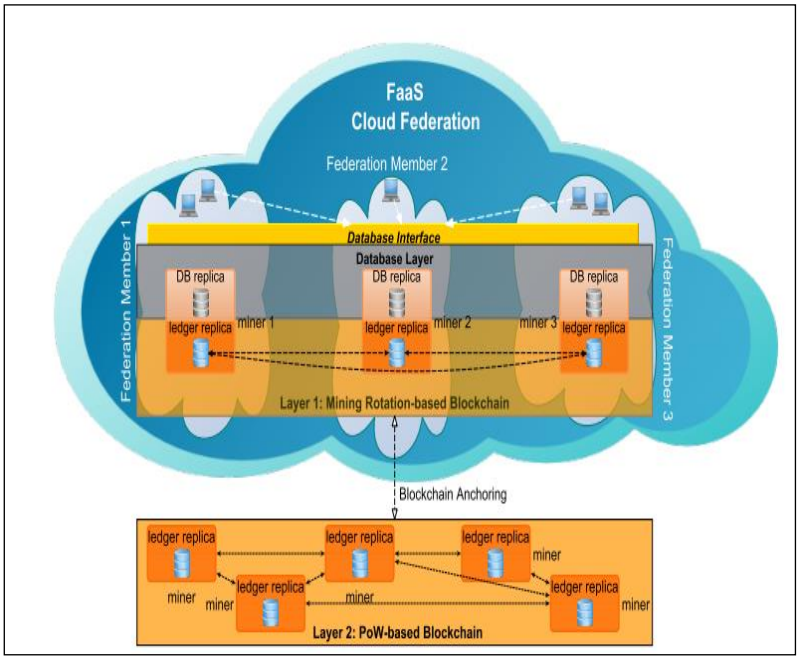


Fig 1 FaaS Cloud Federation

## IV. CONCLUSION

In our paper, we've pinpointed what's needed and what questions need answers to create a blockchain-powered database for cloud computing setups. We've based our findings on actual needs identified in the SUNFISH project in Europe. Our big idea here is offering up a top-level plan that tackles these questions head-on. This plan sets the stage for crafting a blockchain-driven database that can deliver the right assurances on keeping data safe, making sure things run smoothly, and staying reliable over time. This project sets the stage for exciting future endeavors. Our proposed direction can be explored further by creating a working model to test how well our solution performs in terms of speed and capacity. Additionally, we need to delve deeper into understanding the balance between performance and ensuring data stays intact to prove that our design is effective against potential threats. It's important to note that our preliminary design for the blockchain-based database relies on a complete consensus mechanism. While this approach ensures integrity among the distributed copies of data and simplifies handling potential issues, it could face challenges if even just one miner misbehaves. As we move forward with implementing such a database, we're developing a reliable consensus algorithm that can maintain both integrity and availability, even in the face of faults. Lastly, it's crucial to explore ways to create more stable blockchains to make them viable options for storing data, especially in cloud computing setups.

## REFERENCES

[1]. Giuseppe Ateniese, Randal Burns, Reza Curtmola, Joseph Herring, Lea Kissner, Zachary Peterson, and Dawn Song. Provable data possession at untrusted stores. In Proceedings of the 14th ACM conference on Computer and Communications security, pages 598–609. ACM, 2007.

[2]. Joseph Bonneau, Andrew Miller, Jeremy Clark, Arvind Narayanan, Joshua A Kroll, and Edward W Felten. Sok: Research perspectives and challenges for bitcoin and cryptocurrencies. In 2015 IEEE Symposium on Security and Privacy, pages 104–121. IEEE, 2015.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY148

[3]. George Danezis and Sarah Meiklejohn. Centrally Banked Cryptocurrencies. In 23rd Annual Network and Distributed System Security Symposium, NDSS, 2016.
[4]. ENISA. Security Framework for Governmental Clouds, 2015. Available at https://www.enisa.europa.eu/activities/Resi lience-and-CIIP/cloud-computing/ governmental-cloud-security/securityframework-for-govenmental-clouds.
[5]. Ittay Eyal, Adem Efe Gencer, Emin G¨un Sirer, and Robbert Van Renesse. Bitcoin-NG: A scalable blockchain protocol. In 13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16), pages 45–59, 2016.
[6]. Juan Garay, Aggelos Kiayias, and Nikos Leonardos. The Bitcoin Backbone Protocol: Analysis and Applications, pages 281–310. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.
[7]. Trent McConaghy, Rodolphe Marques, Andreas M¨uller, Dimitri De Jonghe, Troy McConaghy, Greg McMullen, Ryan Henderson, Sylvain Bellemare, and Alberto Granzotto. BigchainDB: A Scalable Blockchain Database (DRAFT). 2016.
[8]. Satoshi Nakamoto. Bitcoin: A peer-topeer electronic cash system, 2008. Available at https: //bitcoin.org/bitcoin.pdf.
[9]. Francesco Paolo Schiavo, Vladimiro Sassone, Luca Nicoletti, and Andrea Margheri. FaaS: Federation-as-a-Service, 2016. Technical Report. Available at https://arxiv.org/abs/1612. 03937.
[10]. Mehdi Sookhak, Abdullah Gani, Hamid Talebian, Adnan Akhunzada, Samee U. Khan, Rajkumar Buyya, and Albert Y. Zomaya. Remote Data Auditing in Cloud Computing Environments: A Survey, Taxonomy, and Open Issues. ACM Comput. Surv., 47(4):65:1–65:34, May 2015.
[11]. Mehdi Sookhak, Hamid Talebian, Ejaz Ahmed, Abdullah Gani, and Muhammad Khurram Khan. A review on remote data auditing in single cloud server: Taxonomy and open issues. Journal of Network and Computer Applications, 43:121–141, 2014.
[12]. Bojan Suzic, Bernd Pr¨unster, Dominik Ziegler, Alexander Marsalek, and Andreas Reiter. Balancing Utility and Security: Securing Cloud Federations of Public Entities. In C&TC, volume 10033 of LNCS, pages 943–961. Springer, 2016.
[13]. Mor Weiss, Boris Rozenberg, and Muhammad Barham. Practical Solutions For Format-Preserving Encryption. CoRR, abs/1506.04113, 2015.
[14]. Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger. Ethereum Project Yellow Paper, 2014.