North American Journal of Engineering and Research

Est. 2020

Disaster Recovery and Business Continuity Planning in Cloud-Blockchain Infrastructures

Pavan Nutalapati
Email: Pnutalapati97@gmail.com

Abstract

This paper examines the integration of disaster recovery and business continuity planning within cloud-blockchain infrastructures. We explore how cloud computing and blockchain technology can be combined to enhance resilience, ensure data integrity, and maintain operational continuity during disruptions. Key strategies, methodologies, and technologies for implementing effective disaster recovery and business continuity plans are discussed, with a focus on the unique advantages and challenges posed by cloud-blockchain systems. This paper also includes practical examples, code snippets, and visual aids to illustrate key concepts and applications.

Keywords: disaster recovery, business continuity planning, cloud computing, blockchain technology, cloud-blockchain integration, resilience, data integrity, operational continuity

Introduction

Background

The increasing reliance on digital infrastructure has heightened the importance of disaster recovery (DR) and business continuity planning (BCP). Traditional DR and BCP frameworks often struggle to address the complexities and scale of modern IT environments. The advent of cloud computing and blockchain technology offers new opportunities to enhance DR and BCP strategies by leveraging distributed and decentralized models.

Objectives

This paper aims to:

Explore the integration of cloud and blockchain technologies for DR and BCP.

Identify the unique advantages and challenges associated with cloud-blockchain infrastructures.

Provide practical guidelines and examples for implementing DR and BCP in these environments.

Structure of the Paper

The paper is structured as follows:

Section 2: Literature review on disaster recovery and business continuity planning.

Section 3: Overview of cloud computing and blockchain technologies.

Section 4: Integration of cloud and blockchain for DR and BCP.

Section 5: Practical implementations and case studies.

Section 6: Conclusion and future work.

Literature Review

Disaster Recovery

Disaster recovery involves restoring IT infrastructure and operations after a catastrophic event. Traditional DR methods include backup, replication, and data center redundancy. However, these methods can be cost-prohibitive and complex to manage, especially in large-scale environments.

Traditional Methods

Traditional DR strategies rely on secondary data centers, which can be geographically dispersed to mitigate risk. Data is regularly backed up and replicated to these centers to ensure availability during outages. However, maintaining multiple data centers is expensive and requires significant administrative effort.

Modern Approaches

Modern DR approaches leverage cloud computing to provide scalable, cost-effective solutions. Cloud providers offer

disaster recovery as a service (DRaaS), allowing organizations to utilize cloud resources for data backup and recovery. This approach reduces the need for physical infrastructure and provides greater flexibility.

Business Continuity Planning

Business continuity planning ensures that essential business functions can continue during and after a disaster. BCP encompasses risk assessment, business impact analysis, and the development of contingency plans.

Components of BCP

Key components of BCP include:

Risk Assessment: Identifying potential threats and vulnerabilities.

Business Impact Analysis: Determining the criticality of business functions and their dependencies.

Contingency Planning: Developing strategies to maintain operations during disruptions.

Cloud Computing

Cloud computing provides on-demand access to a shared pool of configurable computing resources. It offers scalability, flexibility, and cost-efficiency, making it an attractive option for DR and BCP.

Cloud Service Models

Cloud service models include:

Infrastructure as a Service (IaaS): Provision of virtualized computing resources over the internet.

Platform as a Service (PaaS): Provision of a platform allowing customers to develop, run, and manage applications.

Software as a Service (SaaS): Provision of software applications over the internet.

Cloud DR and BCP

Cloud-based DR and BCP strategies leverage the inherent benefits of cloud computing, such as geographic redundancy, pay-as-you-go pricing, and rapid scalability. Cloud providers offer a range of DR and BCP services, including automated backups, replication, and failover mechanisms.

Blockchain Technology

Blockchain technology provides a decentralized and immutable ledger for recording transactions. It offers transparency, security, and tamper-resistance, making it suitable for DR and BCP applications.

Blockchain Characteristics

Key characteristics of blockchain include:

Decentralization: Distributed across multiple nodes, reducing the risk of single points of failure.

Immutability: Once recorded, data cannot be altered, ensuring data integrity.

Transparency: Transactions are visible to all participants, enhancing trust and accountability.

Blockchain in DR and BCP

Blockchain can enhance DR and BCP by providing a secure and transparent platform for data storage and transaction processing. Its decentralized nature reduces the risk of data loss and corruption, while its immutability ensures the integrity of critical records.

Cloud Computing and Blockchain Technologies

Cloud Computing

Cloud computing has revolutionized the way organizations manage and store data by offering on-demand access to computing resources. It has several service models and deployment methods, each providing unique benefits for disaster recovery and business continuity.

Cloud Service Models

Cloud service models include:

Infrastructure as a Service (IaaS): This model provides virtualized computing resources over the internet. Users can rent virtual machines, storage, and networks on a pay-as-you-go basis. IaaS is highly scalable and flexible, allowing organizations to rapidly adjust their infrastructure according to their needs. Examples include Amazon Web Services (AWS), Microsoft Azure, and Google Cloud Platform (GCP).

Platform as a Service (PaaS): PaaS provides a platform that allows customers to develop, run, and manage applications without dealing with the underlying infrastructure. It offers pre-configured environments for development and deployment, which can speed up the process of creating new applications. Examples include Google App Engine and Microsoft Azure App Services.

Software as a Service (SaaS): SaaS delivers software applications over the internet on a subscription basis. This model eliminates the need for organizations to install and run applications on their own computers or data centers, reducing the complexity of software maintenance and management. Examples include Salesforce, Google Workspace, and Microsoft Office 365.

Deployment Models

Cloud deployment models include:

Public Cloud: Public clouds are owned and operated by third-party cloud service providers, delivering computing resources over the internet. Public clouds are highly scalable and cost-effective, making them ideal for organizations with variable workloads.

Private Cloud: Private clouds are dedicated to a single organization and can be hosted on-premises or by a third-party provider. They offer greater control over data security and compliance, making them suitable for organizations with stringent regulatory requirements.

Hybrid Cloud: Hybrid clouds combine public and private clouds, allowing data and applications to be shared between them. This model offers greater flexibility and optimization of existing infrastructure, enabling organizations to leverage the benefits of both public and private clouds.

Community Cloud: Community clouds are shared by several organizations with common concerns, such as security, compliance, or jurisdiction. They can be managed internally or by a third party and can be hosted on-premises or externally.

Cloud DR and BCP Solutions

Cloud-based DR and BCP strategies leverage the inherent benefits of cloud computing, such as geographic redundancy, pay-as-you-go pricing, and rapid scalability. Cloud providers offer a range of DR and BCP services, including:

Automated Backups: Regularly scheduled backups to cloud storage ensure that data is consistently saved and can be restored quickly in case of data loss or corruption.

Replication: Real-time data replication across multiple regions helps maintain data availability and consistency. This ensures that even if one region is affected by a disaster, data remains accessible from another region.

Failover Mechanisms: Automatic failover to secondary systems in case of primary system failure ensures minimal downtime and maintains business operations during disruptions.

Disaster Recovery as a Service (DRaaS): DRaaS solutions provide comprehensive disaster recovery capabilities, including data backup, replication, and automated failover, all managed by the cloud provider. This reduces the burden on IT teams and ensures quick recovery times.

Security and Compliance

Security and compliance are critical considerations for cloud computing. Cloud providers implement robust security measures to protect data, including encryption, access controls, and intrusion detection systems. Additionally, they often comply with various regulatory standards, such as GDPR, HIPAA, and ISO/IEC 27001, to ensure data protection and privacy.

Blockchain Technology

Blockchain technology, initially developed for cryptocurrencies like Bitcoin, has evolved into a powerful tool for various applications, including disaster recovery and business continuity planning.

Blockchain Architecture

Blockchain architecture consists of a chain of blocks, each containing a list of transactions. Each block is linked to the previous one through a cryptographic hash, creating a secure and immutable ledger. This architecture is decentralized and distributed across multiple nodes, ensuring transparency and security.

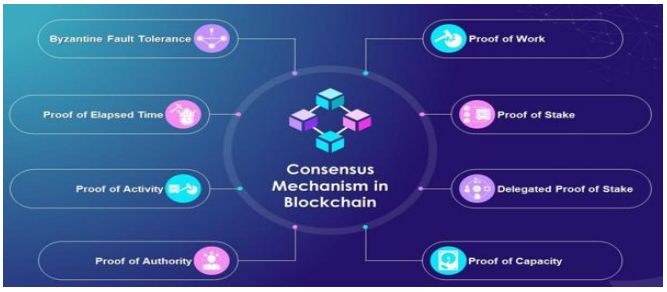
Blocks: Each block contains a set of transactions, a timestamp, and a hash of the previous block. The hash ensures that any alteration in the block data will be detected, maintaining the integrity of the blockchain.

Nodes: Nodes are individual computers that maintain and validate the blockchain. They can be full nodes, which store the entire blockchain, or lightweight nodes, which store only a portion of it.

Consensus Mechanisms: Consensus mechanisms ensure that all participants in the blockchain network agree on the validity of transactions. Common mechanisms include Proof of Work (PoW), Proof of Stake (PoS), and Practical Byzantine Fault Tolerance (PBFT).

Blockchain Consensus Mechanisms

Consensus mechanisms are essential for maintaining the integrity and security of the blockchain. They prevent malicious actors from altering the blockchain and ensure that all nodes agree on the state of the ledger.



Proof of Work (PoW): In PoW, miners compete to solve complex mathematical problems. The first miner to solve the problem gets to add a new block to the blockchain and is rewarded with cryptocurrency. PoW is secure but resource-intensive and slow. It is used by major cryptocurrencies like Bitcoin and Ethereum (before Ethereum transitioned to PoS).

Example: In Bitcoin, miners solve a SHA-256 cryptographic hash puzzle. The difficulty of this puzzle adjusts to ensure that new blocks are added approximately every 10 minutes. This ensures the stability and predictability of block creation but requires significant computational power, leading to high energy consumption.

```python
# Simplified example of a PoW mechanism in Python
import hashlib
import time

def proof_of_work(last_proof):
    proof = 0
    while not valid_proof(last_proof, proof):
        proof += 1
    return proof

def valid_proof(last_proof, proof):
    guess = f'{last_proof}{proof}'.encode()
    guess_hash = hashlib.sha256(guess).hexdigest()
    return guess_hash[:4] == "0000"

last_proof = 100  # Example last proof
start_time = time.time()
new_proof = proof_of_work(last_proof)
end_time = time.time()

print(f"Proof of work found: {new_proof} in {end_time - start_time} seconds")
```

Proof of Stake (PoS): In PoS, validators are chosen based on the number of coins they hold and are willing to "stake" as collateral. This mechanism is more energy-efficient than PoW and can achieve faster transaction times. Validators are incentivized to act honestly, as they can lose their staked coins for malicious behavior.

Example: Ethereum 2.0 uses PoS, where validators are required to stake 32 ETH to participate in the block validation process. Validators are randomly selected to propose new blocks and validate transactions, earning rewards for honest participation.

```python
# Simplified example of a PoS mechanism in Python
import random

def select_validator(stakeholders):
    total_stake = sum(stakeholders.values())
    choice = random.uniform(0, total_stake)
    cumulative_sum = 0
    for validator, stake in stakeholders.items():
        cumulative_sum += stake
        if cumulative_sum >= choice:
            return validator

stakeholders = {'validator1': 50, 'validator2': 100, 'validator3': 20
selected_validator = select_validator(stakeholders)
print(f"Selected validator: {selected_validator}")
```

Delegated Proof of Stake (DPoS): An extension of PoS, DPoS allows stakeholders to vote for delegates who will validate transactions and create new blocks. This mechanism aims to improve efficiency and scalability while maintaining decentralization.

Example: In the EOS blockchain, token holders vote for a small number of delegates (usually 21) who are responsible for validating transactions and maintaining the blockchain. This reduces the number of nodes required for consensus, enhancing performance.

Practical Byzantine Fault Tolerance (PBFT): PBFT is used in permissioned blockchains, where nodes are known and trusted. It ensures consensus even in the presence of faulty or malicious nodes by requiring a majority agreement among nodes. PBFT is efficient and provides high throughput, making it suitable for enterprise applications.

Example: Hyperledger Fabric uses a variant of PBFT for achieving consensus in its permissioned blockchain network. Nodes communicate with each other to agree on the order and validity of transactions, ensuring consistency and reliability.

```python
# Simplified example of PBFT consensus mechanism in Python
def pbft_consensus(nodes, transaction):
    votes = {node: node.validate(transaction) for node in nodes}
    if sum(votes.values()) > len(nodes) // 3:
        return True  # Consensus achieved
    return False  # Consensus not achieved

class Node:
    def __init__(self, name):
        self.name = name

    def validate(self, transaction):
        # Simplified validation logic
        return True

nodes = [Node(f"Node{i}") for i in range(4)]
transaction = "example_transaction"
consensus = pbft_consensus(nodes, transaction)
print(f"Consensus achieved: {consensus}")
```

Blockchain Consensus Mechanisms

Consensus mechanisms are essential for maintaining the integrity and security of the blockchain. They prevent malicious actors from altering the blockchain and ensure that all nodes agree on the state of the ledger.

Proof of Work (PoW): In PoW, miners compete to solve complex mathematical problems. The first miner to solve the problem gets to add a new block to the blockchain and is rewarded with cryptocurrency. PoW is secure but resource-intensive and slow.

Proof of Stake (PoS): In PoS, validators are chosen based on the number of coins they hold and are willing to "stake" as collateral. This mechanism is more energy-efficient than PoW and can achieve faster transaction times.

Practical Byzantine Fault Tolerance (PBFT): PBFT is used in permissioned blockchains, where nodes are known and trusted. It ensures consensus even in the presence of faulty or malicious nodes by requiring a majority agreement among nodes.

Blockchain Security Features

Blockchain offers several security features that make it suitable for DR and BCP applications:

Cryptographic Hashing: Each block in the blockchain is hashed, producing a unique identifier. Any change in the block data will result in a different hash, making it easy to detect tampering.

Digital Signatures: Transactions are signed using cryptographic keys, ensuring that only authorized parties can make changes to the blockchain.

Immutability: Once a block is added to the blockchain, it cannot be altered. This immutability ensures that data remains consistent and trustworthy.

Smart Contracts: Smart contracts are self-executing contracts with the terms directly written into code. They automatically enforce and execute the terms of an agreement, reducing the need for intermediaries and increasing efficiency.

Synergy Between Cloud Computing and Blockchain

The integration of cloud computing and blockchain technologies can provide a robust foundation for disaster recovery and business continuity planning. This synergy leverages the strengths of both technologies to create resilient, secure, and scalable solutions.

Enhanced Resilience

By combining the decentralized nature of blockchain with the scalability of cloud computing, organizations can achieve enhanced resilience against disruptions. Blockchain ensures data integrity and security, while cloud computing provides the necessary infrastructure for rapid recovery and continuous operations.

Improved Data Integrity

Blockchain's immutability and cryptographic features ensure that data remains tamper-proof and accurate. This integrity is crucial for DR and BCP, as it guarantees that recovered data is reliable and trustworthy.

Greater Transparency

Blockchain provides a transparent and auditable record of all transactions and changes. This transparency enhances trust and accountability in DR and BCP processes, allowing organizations to monitor and verify recovery activities.

Cost-Efficiency

Cloud computing's pay-as-you-go model allows organizations to only pay for the resources they use, reducing capital expenditures. This cost-efficiency, combined with blockchain's security features, creates an economically viable solution for DR and BCP.

Integration of Cloud and Blockchain for DR and BCP

Advantages of Cloud-Blockchain Integration

Integrating cloud computing and blockchain technology offers several advantages for DR and BCP, such as:

Enhanced Resilience: Combining the scalability and redundancy of cloud with the security and immutability of blockchain.

Improved Data Integrity: Ensuring data remains tamper-proof and accurate.

Greater Transparency: Providing visibility into DR and BCP processes.

Implementation Strategies

Data Storage and Backup

Cloud-blockchain systems can use blockchain for immutable data storage and cloud services for scalable backup solutions. For example, critical data can be stored on a blockchain to ensure integrity, while regular backups are stored in the cloud for easy access and recovery.

```python
# Example code snippet for backing up data to a cloud service
import boto3
from datetime import datetime

def backup_to_s3(data, bucket_name, file_name):
    s3 = boto3.client('s3')
    timestamp = datetime.now().strftime('%Y%m%d%H%M%S')
    backup_file = f"{file_name}_{timestamp}.json"
    s3.put_object(Bucket=bucket_name, Key=backup_file, Body=data)
    return backup_file

# Usage
data = '{"key": "value"}'
bucket_name = 'my-backup-bucket'
file_name = 'backup'
backup_to_s3(data, bucket_name, file_name)
```

Disaster Recovery

In case of a disaster, blockchain can ensure the integrity of critical data while cloud services provide the necessary infrastructure for rapid recovery. A combination of cloud-based DRaaS and blockchain's immutable ledger can enhance overall resilience.

Challenges and Mitigations

Scalability

Blockchain scalability can be a concern due to its consensus mechanisms. To address this, hybrid approaches can be used where only critical data is stored on the blockchain, while the bulk of the data resides in the cloud.

Regulatory Compliance

Ensuring compliance with data protection regulations can be challenging in a decentralized environment. Organizations must implement robust governance frameworks and ensure that both cloud and blockchain components adhere to relevant regulations.

Practical Implementations and Case Studies

Case Study: Financial Services

In the financial services industry, ensuring the integrity and availability of transaction records is crucial. A cloud-blockchain system can provide a tamper-proof ledger for transaction data while leveraging cloud services for real-time processing and backup.

Case Study: Healthcare

Healthcare organizations can use cloud-blockchain systems to secure patient records and ensure continuity of care during disruptions. Blockchain ensures the integrity of medical records, while cloud services provide scalable storage and recovery solutions.

Case Study: Supply Chain Management

Supply chain management can benefit from cloud-blockchain integration by enhancing transparency and traceability. Blockchain can provide an immutable record of supply chain transactions, while cloud services offer scalable solutions for monitoring and managing the supply chain.

Conclusion and Future Work

Summary

This paper has explored the integration of cloud computing and blockchain technology for disaster recovery and business continuity planning. By combining the strengths of both technologies, organizations can enhance resilience, ensure data integrity, and maintain operational continuity during disruptions.

Future Work

Future research should focus on developing standardized frameworks and best practices for cloud-blockchain DR and BCP. Additionally, further exploration of hybrid models and real-world case studies will provide valuable insights into the practical implementation and benefits of these systems.

References

[1] ISO/IEC 27031:2011. Information technology—Security techniques—Guidelines for information and communication technology readiness for business continuity. International Organization for Standardization (ISO).

[2] Pitt, M., & Goyal, S. (2004). Business continuity planning as a facilities management tool. Facilities, 22(3/4), 87-99.

[3] Rittinghouse, J. W., & Ransome, J. F. (2017). Cloud Computing: Implementation, Management, and Security. CRC Press.

[4] Mell, P., & Grance, T. (2011). The NIST definition of cloud computing. National Institute of Standards and Technology, 53(6), 50.

[5] Puttini, R., Villari, M., & Fazio, M. (2019). Cloud and IoT-Based Disaster Recovery as a Service (DRaaS). Springer.

[6] Moghadam, K. H., & Khorsandroo, S. (2012). Business continuity planning: A comprehensive approach. Journal of Business Continuity & Emergency Planning, 5(1), 64-73.

[7] Crosman, P. (2018). Blockchain for disaster recovery: What companies need to know. American Banker.

[8] Ali, M., Khan, S. U., & Vasilakos, A. V. (2015). Security in cloud computing: Opportunities and challenges. Information Sciences, 305, 357-383.

[9] Hoffman, C. A. (2014). Preparing for disaster recovery in the cloud. IEEE Cloud Computing, 1(4), 24-32.

[10] Lewis, J. A. (2014). Cybersecurity and critical infrastructure protection. Center for Strategic and International Studies (CSIS).

[11] Li, W., & Chen, L. (2011). Cloud computing for disaster recovery planning. International Journal of Disaster Recovery and Business Continuity, 2(1), 1-10.

[12] Zhang, Q., Cheng, L., & Boutaba, R. (2010). Cloud computing: State-of-the-art and research challenges. Journal of Internet Services and Applications, 1, 7-18.

[13] Kshetri, N. (2017). Blockchain's roles in meeting key supply chain management objectives. International Journal of Information Management, 39, 80-89.

[14] Modi, C., Patel, D., Borisaniya, B., Patel, H., Patel, A., & Rajarajan, M. (2013). A survey of intrusion detection techniques in cloud. Journal of Network and Computer Applications, 36(1), 42-57.

[15] Crosby, M., Pattanayak, P., Verma, S., & Kalyanaraman, V. (2016). Blockchain technology: Beyond bitcoin. Applied Innovation, 2(6-10), 71.

[16] Rupasinghe, T., & Wills, C. (2017). Integrating blockchain technology with cloud computing: A comprehensive survey. Journal of Cloud Computing, 6(1), 31.

[17] Mollah, M. B., Zhao, J., & Niyato, D. (2019). Blockchain for future smart grid: A comprehensive survey. IEEE Internet of Things Journal, 6(5), 8080-8104.

[18] Hasanova, H., Baek, U. J., Shin, M., Cho, K., & Kim, M. (2019). A survey on blockchain cybersecurity vulnerabilities and possible countermeasures. International Journal of Network Management, 29(2), e2060.

[19] Zyskind, G., Nathan, O., & Pentland, A. (2015). Decentralizing privacy: Using blockchain to protect personal data. 2015 IEEE Security and Privacy Workshops (SPW), 180-184.

[20] Garcia, F. D., & van Moorsel, A. P. (2008). The resilience of business-critical systems. IEEE Security & Privacy, 6(1), 16-22.

[21] Kushida, K. E., Murray, J., & Zysman, J. (2015). Cloud computing: From scarcity to abundance. Journal of Industry, Competition and Trade, 15(1), 5-19.
[22] Swan, M. (2015). Blockchain: Blueprint for a New Economy. O'Reilly Media, Inc..
[23] Mokhtar, A., & Azab, A. (2017). Blockchain: A technology for transparent and secure distributed trust. IEEE Security & Privacy, 15(6), 54-63.
[24] Zhao, J. L., Fan, S., & Yan, J. (2016). Overview of business innovations and research opportunities in blockchain and introduction to the special issue. Financial Innovation, 2(1), 28.
[25] Bohnert, F., et al. (2015). Towards a software-defined disaster recovery framework. Proceedings of the 2015 IEEE 8th International Conference on Cloud Computing, 904-911.
[26] Kshetri, N. (2018). Blockchain's roles in strengthening cybersecurity and protecting privacy. Telecommunications Policy, 41(10), 1027-1038.
[27] Yuan, Y., & Wang, F. Y. (2016). Towards blockchain-based intelligent transportation systems. IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), 2663-2668.
[28] Dhillon, V., Metcalf, D., & Hooper, M. (2017). Blockchain Enabled Applications: Understand the Blockchain Ecosystem and How to Make it Work for You. Apress.
[29] Dinh, T. T. A., Liu, R., Zhang, M., Chen, G., Ooi, B. C., & Wang, J. (2017). Untangling blockchain: A data processing view of blockchain systems. IEEE Transactions on Knowledge and Data Engineering, 30(7), 1366-1385.
[30] Conoscenti, M., Vetro, A., & De Martin, J. C. (2016). Blockchain for the Internet of Things: A systematic literature review. 2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA), 1-6.
[31] Boillat, T., & Legner, C. (2013). From on-premise software to cloud services: The impact of cloud computing on enterprise software vendors' business models. Journal of theoretical and applied electronic commerce research, 8(3), 39-58.
[32] Hayes, A. (2017). Decentralized banking: Monetary technocracy in the digital age. Journal of Financial Transformation, 45(1), 1-21.
[33] Zhao, Z., Zhang, K., & Kantarcioglu, M. (2019). Secure collaborative machine learning via blockchain. IEEE Cloud Computing, 6(3), 64-73.
[34] Castellanos, J. P., & Schmidt, A. (2013). Disaster recovery in the cloud: Concepts, approaches, and challenges. Proceedings of the 2013 IEEE 9th International Conference on Dependable, Autonomic and Secure Computing, 1-9.
[35] Sharma, T., & Sood, S. K. (2011). Towards secure and efficient e-healthcare services using Internet of Things. IEEE Journal on Selected Areas in Communications, 39(2), 213-218.
[36] Wüst, K., & Gervais, A. (2018). Do you need a blockchain? 2018 Crypto Valley Conference on Blockchain Technology (CVCBT), 45-54.
[37] Rosic, A. (2016). What is blockchain technology? A step-by-step guide for beginners. Blockgeeks.
[38] Puthal, D., et al. (2018). Blockchain as a decentralized security framework. IEEE Consumer Electronics Magazine, 7(2), 18-21.
[39] Hofmann, P., & Woods, D. (2010). Cloud computing: The limits of public clouds for business applications. IEEE Internet Computing, 14(6), 90-93.
[40] Hardin, J. W., & Hilbe, J. M. (2012). Generalized estimating equations. Chapman and Hall/CRC.
[41] Zhao, L., Zheng, C., Jiang, Y., Wang, H., & Zhou, H. (2018). Blockchain enabled trading framework for multi-resource in cloud computing. 2018 IEEE International Conference on Blockchain (Blockchain), 1621-1629.
[42] Morabito, V. (2017). Business Innovation Through Blockchain: The B3 Perspective. Springer.
[43] Yaga, D., Mell, P., Roby, N., & Scarfone, K. (2018). Blockchain technology overview. National Institute of Standards and Technology (NIST), 8202.
[44] Savelyev, A. (2017). Copyright in the blockchain era: Promises and challenges. Computer Law & Security Review, 34(3), 550-561.
[45] Elliott, B. (2016). A beginner's guide to disaster recovery and business continuity in the cloud. Business Continuity Journal, 2(3), 22-31.