future internet

MDPI

Article

LSSDNF: A Lightweight Secure Software Defined Network Framework for Future Internet in 5G–6G

Surjit Singh [1,*], Vivek Mehla [1] and Srete Nikolovski [2,*]

[1] Computer Science and Engineering Department, Thapar Institute of Engineering & Technology, Patiala 147004, India
[2] Power Engineering Department, Faculty of Electrical Engineering Computing and Information Technology, 31000 Osijek, Croatia
* Correspondence: surjit.singh@thapar.edu or surjitmehla@gmail.com (S.S.); srete.nikolovski@ferit.hr (S.N.)

Abstract: As information technology advances quickly, so does the 5G–6G network management system, which is moving toward greater integration, decentralization, diversity, and intelligence. As flexibility is a crucial criterion for 5G–6G network architecture, we use the Software Defined Network (SDN) paradigm to make the programmability more flexible. Due to their ability to replace the current TCP/IP architecture with one that separates the control plane and data plane, software-defined networks have gained much popularity. However, they are susceptible to routing attacks. Therefore, this work proposes Lightweight Security Framework that combines blockchain technology with Software-Defined Networking (LSSDNF) to address this problem. The proposed framework adds the routing data that the controller withheld to the multichain blockchain. Here, a mininet network simulator is used to model the proposed framework. The data transfer rate or network throughput, bandwidth variation, and jitter have all been used to assess the performance of single-controller-SDN networks and multi-controller-SDN networks. The results demonstrate that the proposed framework performs better than the conventional single-controller-SDN architecture in terms of throughput, bandwidth fluctuation, and jitter.

Keywords: SDN; blockchain; 6G; 5G; routing; performance modeling; internet of things

Citation: Singh, S.; Mehla, V.; Nikolovski, S. LSSDNF: A Lightweight Secure Software Defined Network Framework for Future Internet in 5G–6G. *Future Internet* **2022**, *14*, 369. https://doi.org/10.3390/fi14120369

Academic Editor: Giovanni Pau

Received: 21 October 2022
Accepted: 4 December 2022
Published: 8 December 2022

1. Introduction

The 5G/6G network is separated into the 5G/6G network architecture sections: the control layer, the access layer, and the forward layer [1]. This architecture separates user data from the control plane from the user plane only through the access and forwarding layers. In contrast, the control layer performs a logically centralized network control role. The access layer covers a wide range of wireless technologies and network configurations. Based on a common hardware platform, the forward layer can ensure high reliability, minimal delay, and average load of the service data flow.

Blockchain-enhanced Software-Defined Networking refers to a network controlled by a central entity or software used by all the network resources. With an SDN, the control is moved from the network layer to the application layer, dividing the data plane from the control plane. It has numerous advantages over a traditional network, including lower costs, improved security, more flexibility, easier configuration, and prevention of vendor lock-in. A secure framework for SDN can allow a network manager to change the way the network devices, i.e., Routers and Switches, handle packets by enabling complete control of the rules set into network devices from a central console. The mobility of devices in wireless networks has an effect on communication services. The controller can solve position changes in horizontal switching by detecting geographic locations, and it can also somewhat enhance vertical mobility and offer load balancing. Although there is no mobility, the controller can nevertheless locate an alternative route in the event of a failure

and balance the traffic between nodes. Some researchers have brought up these issues; however, there are not many workable answers.

Wireless networks have become more virtualized and sophisticated with 5G/6G. While there are significant security flaws in the topology of conventional wireless networks, 5G and 6G promote infrastructure sharing. Even if the widely accepted hypotheses are true, their plausibility is diminished in networks with a large number of adaptive sensing devices. Additionally, the virtualization trend will provide many unknown difficulties.

The variables that affect a network's quality include bandwidth, Jitter, propagation time, throughput, and latency. One of the most significant is Jitter, which typically causes reduced network throughput and, in certain situations, network congestion that sometimes results in total network failure. The market for Secure-SDN technology is expanding as all the major multinational corporations—including Google, Facebook, and Amazon—adapt to it and change their traditional networks. Traditional networks are now more sluggish, risky, and expensive. Software-Defined Networking with blockchain, on the other hand, enhances network connectivity for sales, customer support, internal communications, and document sharing. This job is in high demand since it integrates blockchain and SDN.

SDN enables complete network control, enabling quick responses to change network or business requirements. By converting them to a Secure Framework for Software Defined Networks, this work focuses on improving the scalability, efficiency, and security of the current Traditional Networks. By dividing the network into three layers—the application layer, control layer, and forwarding layer—traditional decentralized networks converted to centralized SDN networks. These layers each play a part in enhancing the scalability, efficiency, and security of the network. Routing attacks change how packets are routed on the SDN by manipulating routing control information. In order to keep the network secure from routing attacks, the data of the SDN network, such as routing information, including IP addresses and MAC addresses, are stored on a blockchain utilizing multichain.

The contributions of the proposed work are as follows:

- This paper first highlights how much faster, more scalable, and less expensive a software-defined network is than a conventional network in 5G–6G SDN.
- Building a private blockchain and uploading the network's routing information makes SDN safe and impossible for anyone to change it. Creation of a private blockchain with the sole purpose of making a network more secure. Therefore, in order to keep the network secure from malicious nodes, the data of the SDN network, such as routing information, including IP addresses and MAC addresses, are stored on a blockchain utilizing multichain.
- Transfer rate, jitter, delay, and capacity are among the network performance measures used to evaluate the network in a mininet network simulator.

The remainder of the manuscript is structured as follows: The associated work is briefly reviewed in Section 2. The proposed work is explained in Section 3. The simulation setup and results are presented in Section 4, and Section 5 outlines the conclusions.

2. Related Work

Computer networks are currently getting faster, smarter, resource-rich, and more complex as a result of rapid progress. For instance, the Internet of Things (IoT) has been steadily embraced by many businesses and networks. According to a Gartner forecast, there are more than 20 billion linked devices worldwide as of 2020. As the control and data plane are traditionally integrated inside the network devices, the IoT architecture frequently contains a large number of devices, servers, and middleboxes, making network management and configuration much more challenging and error-prone. It is also very laborious to add new devices and modify services. Deploying new services without disrupting existing ones is typically a challenging task. As a network becomes bigger, this problem can get worse.

A viable solution to this problem is Software-Defined Networking (SDN), which separates the network control plane from the data plane. In such a network, IT managers

can adjust network regulations from a controller that is software-based without having to modify settings in each switch. That is to say, regardless of the precise connections between a server and devices, the centralized SDN controller can instruct the switches to supply network services under the needs. This centralized management helps ease network administration and lighten configuration work. In reality, SDN still faces numerous security risks. For example, the SDN controller itself could fail in an adversarial environment, and distributed controllers may also experience reliability and reputation problems.

On the other hand, blockchain technology has drawn a lot of interest from academics and industry due to the massive success of the Bitcoin cryptocurrency. A blockchain maintains a record of all data transfers, where the record is also referred to as a "ledger" and the transfer of data as a "transaction". Each new transaction, which can only be added to the blockchain after successful verification, can be verified via a peer-to-peer network. In this situation, it is thought that blockchain would make it possible for unrelated parties to exchange data or information without requiring a reliable intermediary.

Akshatha et al. (2018) [2] suggested a novel approach for conforming the 5G architecture of the Third Generation Partnership Project (3GPP) to the principles of Software Defined Networking (SDN). The gNB base station is transformed into a pure data plane node as a result of moving the control functionality from the 5G Radio Access Network (RAN) to the network core. As a result, the cost of signaling between the RAN and the core network is significantly reduced. Additionally, the performance of the system is enhanced.

The common SDN-5G/6G application scenarios and important concerns are covered by Long et al. (2019) [3]. This work also concentrated on mobile network mobility management strategies. Additionally, three types of software-defined 5G/6G mobility management mechanisms are discussed and contrasted.

Ly et al. (2020) in [4] analyze the dual-channel architecture specified by the software of wireless sensor in 6G/IoE and proposes a reasonable solution to reduce the signal interference so that the related signals can be transmitted more effectively. This is necessary in order to broadly apply 6G/IoE (Internet of Everything) to various scenarios in real life and connect the infrastructure closely related to people through the network. Zebari et al. (2021) in [5] primarily focused on 5G by highlighting the fundamental design principles associated with such technology, such as the infrastructure architecture of 5G, the deployment potential of 5G, the technology employed in 5G, and the services and applications that 5G is capable of offering.

Yang et al. (2022) [6] presented that the next-generation 6G network should be data-driven and sensor-based to enable huge connections and distributed intelligence. Artificial intelligence (AI) is expected to play a significant role in meeting critical needs for 6G networks because the majority of intelligent applications are being deployed at the edge [6]. The authors outlined the need for a 6G network for AI applications before providing a thorough overview of AI toward 6G networks. They focused on the demands, trends, and problems of distributed edge intelligence in upcoming 6G networks. Then, 6G networks are used as inspiration for a discussion of a liquid-specific and adaptable software-defined network architecture for AI applications.

Yungaicela-Naula et al. (2022) [7] examine the most recent security automation research projects in SDN environments. Different classes of security solutions with varying degrees of automation and complexity were discovered and ranked. Four clearly defined qualitative parameters—self-healing, self-adaptation, self-configuration, and self-optimization—are used to gauge the degree of automation. The quantity of processing, storage, and implementation requirements serve as indicators of complexity.

The Internet of Things (IoT) of today will become a gigantic IoT of the future as the number of internet-connected gadgets rises [8–12]. On distributed-controller software-defined networking, network policy updates must be committed in a consistent manner. If this does not happen, the network can encounter unanticipated transient configuration states that affect its performance, security, or even proper operation [13]. By providing a comprehensive view of the network, Software-Defined Networking (SDN) makes it possible to administer networks effectively [14]. Although SDN was not specifically designed to address IoT problems, it can nonetheless give the motivation to address complex problems and aid in effective IoT service orchestration. IoT realization is a difficult problem because of the present IoT paradigm's large data generation, complex infrastructures, security risks, and requirements from the recently developed technologies.

Due to its benefits in offering broad access services and the quick development of satellite technology, the space information network has been presented as a solution to address the rising demands of ubiquitous mobile communication [15]. However, the extremely dynamic topology and the limited resource availability of satellites make management and resource use difficult for development. In order to streamline management and boost resource efficiency, some effort has been made to integrate the SDN, also known as the software-defined space information network. Huo et al. (2022) [16] provide a unique lightweight measuring system that operates in the controller in order to lessen the overhead involved in the measurement process and achieve roughly fine-grained measurements. Coarse-grained measurement and interpolation optimization make up the innovative lightweight measurement architecture.

Abid et al. (2022) [17] state that virtualization and Software-Defined Networking (SDN) are two promising technologies for affordably addressing the size and versatility needed for IoT. Following a discussion of traditional IoT networks and the necessity of SDN and Network Function Virtualization (NFV), the authors examined SDN and NFV solutions for various IoT implementation strategies. According to Singh and Sharma (2019) [10], researchers must consider integrating new technologies rather than making minor changes to the LTE specifications since rising user networking demands in industrial automation, precision agriculture, and augmented reality are driving the development of 5G networks. A rapid and effective response must be made in the present world of smart communication, processing, and operation [18]. According to Chica et al. (2020) [19], SDN enables network managers to more flexibly apply security defense mechanisms to its network-wide visibility, monitoring, and flexible network programmability. The core components of the overall system framework are an automated security management function that aids the SDN controller in attack detection and mitigation and a network security monitor function that helps network managers keep tabs on the health of their networks.

According to Bannour et al. (2018) [20], a physically-centralized control plane with a single-controller overseeing the entire network is the ideal option in terms of simplicity. A single-controller system, though, might not be able to keep up with the network's expansion [20]. With an increase in requests and a struggle to maintain the same level of performance, it is likely to become overloaded (a controller bottleneck). A Data Center Network primarily consists of tens of thousands of switching components. One centralized SDN controller could get overloaded by the volume of control events produced by such a large number of forwarding components that can develop quickly. Specifically, WANs and overlay networks are logically dispersed control plane architecture that functions in multi-domain heterogeneous contexts. The most apparent benefit of distributed SDN is the separation of the control plane's intra-domain and inter-domain features, with each feature being carried out by a different component of the distributed SDN architecture.

Routing attacks change how packets are routed on IoT networks by manipulating routing control information [21]. Farris et al. (2018) [21] present that through the generation of fake erroneous routing notifications, hostile attackers can construct routing loops. Routing attacks have been researched using a variety of tactics. A malicious node advertises the shortest path to a destination node in a Black Hole attack, forcing all packets to be

forwarded to that node. The enemy can then drop or modify the arriving packets. In a Sybil attack, an adversary device, or Sybil node, can generate false routing information and claim a genuine identity in the IoT network, which can change the proper forwarding rules of nearby nodes. On recent attacks and defense strategies for distributed denial of service (DDoS) attacks, Javanmardi et al. (2021) [22] highlighted TCP SYN flood attacks in IoT and SDN in particular. Additionally, they summarised a simple method for detecting TCP SYN flood assaults in IoT networks, a predictive model for mitigating SYN TCP attack detection, and bots that resemble Mirai as IoT-based DDoS attacks. Mirai employs bot instances that launch TCP SYN flood attacks on the targeted computing resources.

Customizing blockchains to increase the security and energy efficiency of SDN was presented by Latif et al. in 2022 [23]. IoT's computation and energy bottleneck problem must be rapidly managed if safe and reliable services are to be provided. They concentrated on minimizing a few network-related issues with IoT. They have been suggested as an architecture for IoT networks to increase security and energy efficiency by utilizing the power of AI. They combined two cutting-edge AI-based technologies, blockchain, and SDN, and took advantage of their promise for effective data analysis, data security, and energy management. A cluster-based framework has been established for the IoT network with blockchain integration for the SDN controller. The blockchain is utilized on networks that are both public and private. A brief description of the related work is shown in Table 1.

Table 1. A brief comparison of the existing work.

Sr. No.	References	Technique	Features	Blockchain Used	Benefits	Shortcoming
1.	Javanmardi et al. (2021) [22]	FUPE	Trust degree of nodes	No	Better network utilization	-Not dealing with attack
2.	Latif et al.(2022) [23]	AI-SDN Security	Authentication	Yes	Energy efficiency	-Basic blockchain approach-No routing information added
3.	Daoud et al. (2019) [24]	Task Priority	Resources availability	No	Resource security	-Not dealing with attack

When using traditional networks in the modern era, several problems can occur. They unquestionably no longer exist. We must rethink how we approach networking in light of technologies like cloud computing, remote work, and multi-device management. The amount of data being transmitted grows daily, which increases the number of 5G and 6G devices. When a new network device is added to the system, rewriting is required. All of these gadgets have human controllers. Along with the development of 5G and 6G networks, more devices need to be monitored. Managing these massive 5 G and 6G network infrastructures is difficult. This needs to be performed dynamically in large networks.

SDN facilitates the separation, abstraction, and sharing of network resources, which aids in the virtualized network infrastructure in 5G [25]. NFV is a noteworthy innovation that has profoundly altered networking and service provisioning. While some businesses assert that 5G solutions are already available, there are legitimate worries that we may only see a few SDN point solutions rather than a fully virtualized, SDN-enabled 5G mobile network, given the enormous effort of softwarizing mobile cellular networks [26]. The authors in [27] provided an in-depth analysis and current solutions for 5G network slicing with SDN and NFV. Following a presentation of the 5G service quality and commercial criteria, the authors described the 5G network softwarization and slicing paradigms, outlining key ideas, their background, and many use cases. The management and orchestration of network slices in a single domain are covered, and then a thorough analysis of management and orchestration techniques in 5G network slicing over several domains.

In comparison to current 4G technology, 5G infrastructures are anticipated to offer anywhere network access with notable network improved efficiency, including 1000 times more available bandwidth, 10–100 times higher data rate increases, upwards of 90% effi-

ciency, reduced end-to-end delay with less than 1 ms, and 99.999% ease of access [27,28]. The author in [28] analyzes and contrasts three SDN-based multi-hop routing methods. An SDN-based BS in the routing framework controls multi-hop connectivity between mobile nodes that are within its coverage region. The architecture makes use of two different frequency bands: licensed frequency for cellular and unlicensed frequency for D2D communication. In order to govern data forwarding, the controller uses both proactive and reactive methods. The authors in [29] developed an SDN-enabled network model that supports the FOG networking paradigm in 5G IoT scenarios in order to assess the viability of the Mininet network emulator as a component of the 5G simulation environment.

Hence, it is evident from the linked literature that there has never been a framework for 5G–6G routing information security in SDN utilizing blockchain. Therefore, this work presents a security architecture that combines blockchain technology with Software-Defined Networking in 5G–6G.

3. Proposed Approach

The following discussion covers the many technological ideas connected to the proposed work:

Using software-based controllers or application programming interfaces (APIs) to communicate with the network's underlying hardware architecture and control traffic is known as software-defined networking (SDN). This architecture is distinct from conventional networks, which manage network traffic using specialized hardware (such as switches and routers). Through software, SDN may build and manage a virtual network or manage conventional hardware. SDN has numerous advantages over traditional or conventional networks because it divides the control plane and the data plane and places the control in the application layer rather than the network layer.

Blockchain data is stored in such a way that system changes, hacking, and cheating is made difficult or impossible. A blockchain is basically a network of nodes that replicates and distributes a digital record of transactions throughout the network. Each blockchain block is made up of numerous transactions, and each individual's ledger acquires a copy of every new transaction added to the blockchain. The term Distributed Ledger Technology (DLT) refers to a decentralized database maintained by many people. On a blockchain, transactions are stored with an unchangeable cryptographic fingerprint known as a hash. As a result, if a block occurred in only one chain, it would be obvious that it had been updated. To breach a blockchain system, hackers would need to change every block in the chain across all decentralized versions of the chain.

Smart Contracts—A smart contract is a software program that, under specific conditions, controls the exchange of digital wealth between parties directly and autonomously. A smart contract, like any other contract, functions with automated enforcing compliance. Smart contracts are computer programs that run precisely how they are configured (coded, programmed) by their authors. A smart contract is enforceable by code, just way a conventional contract is by law.

iPerf: The maximum possible bandwidth on IP networks can be actively measured using the iPerf. It allows numerous fine-tuning timing, buffer, and protocol-related factors (TCP, UDP, SCTP with IPv4 and IPv6). It provides parameters such as bandwidth, loss, and others for each test.

Multichain—Multichain technology allows users to create specific private blockchains that businesses can utilize for financial transactions. Multichain offers us a command-line interface and a straightforward API. This aids in establishing the chain. The multichain platform cuts the time required for blockchain development by up to 80%.

The steps involved in the proposed smart contract for storing routing information in the multichain are described in Algorithm 1 as follows:

Algorithm 1: *Routing Information Registering: MAC Address, IP Address, and Nest Node ID in Multichain*

Procedure: Route(list, macAddress, IP, NextNode)
Step1*: **if***(device[macAddress]! = True]).
Step1.1*: **then*** mac_list.push(macAddress).
Step1.2*:* device[macAddress] = True
Step2*: **if***(device[IPAddress]! = True]).
Step2.1*: **then*** ip_list.push(IPAddress).
Step2.2*:* device[IPAddress] = True
Step3*: **if***(device[NextNode]! = True]).
Step3.1*: **then*** NextNode_list.push(NextNode).
Step3.2*:* device[NextNode] = True
Step6*: **end procedure Route***

The steps involved in the proposed ***LSSDNF*** approach are described in Algorithm 2 as follows:

Algorithm 2: *Proposed approach: LSSDNF*

The steps involved in the proposed ***LSSDNF*** approach are described as follows:
Step1*:* With the aid of the mininet network simulator, the new topology is set up. The topology components are computers, Openflow switches, and an Openflow remote SDN controller.
Step2*:* The Openflow 1.0 protocol is then used to implement the aforementioned architecture in the mininet.
Step3: Execute several instances of the cli within the same display when the command-line interface is provided in a window using the Xterm. Use Xterm to access the shells of the two network computers for testing.
Step4*:* Configured one of the two chosen devices as a server and the other as a client using the Iperf utility.
Step5*:* In order to accept incoming packets of the same protocol type, the client creates a TCP or UDP window.
Step6*:* Using the client's IP address as a connection point, the server begins transmitting data packets to the client.
Step7*:* In addition to receiving the data packets, the client stores each network characteristic in a separate file, such as bandwidth, jitter, and transfer rate.
Step8*:* For each of the two types of topologies, steps 1 through 6 are repeated three times.
Step10: Out of the topologies compared above, pick the best one and obtain its routing data from the SDN controller.
Step11: Using Multichain, build a private Blockchain and store the routing data.

4. Test System, Assumptions, and Result Analysis

This section provides concrete evidence of how SDN outperforms traditional networks. Here, we tested two distinct network types while controlling the number of devices connected to the network, its bandwidth, and the testing period for each network.

4.1. Simulation Setup

To simulate our network topologies, we used the mininet network emulator. Mininet runs actual kernel, switch, and application code on a single machine to build a realistic virtual network. In this work, we created two topologies using mininet: one with 40 devices, including 30 computers, 9 switches, and 1 SDN controller; the other, with 42 devices, including 30 computers, 9 OpenFlow switches, and 3 SDN controllers.

The multi-controller topology serves only to make the single-controller topology crash-protected. For example, in a single-controller topology, everything depends on the controller, and the entire network will also crash if it crashes. Here, we used a multi-controller topology in a master-slave configuration, where one controller is the master and all the others are slaves; if the master crashes, the slave wakes up and begins acting like the master, and our entire network does not crash.

The following tools and technologies are used to simulate the networks:

- Mininet and the Pox SDN controller are used to construct the topology that will be evaluated [30,31].
- iPerf is employed in order to test the network [32].
- For storing routing information, a multichain private blockchain is used [33].
- Smart contracts are created using the Solidity programming language.

- For evaluating the behavior of the smart contract under various circumstances, the Mocha testing framework is used.
- Ganache-cli for creating a test wallet and blockchain for testing [34].
- Truffle HD Wallet Provider for connecting our smart contract to the user's Metamask wallet in the web3 framework using Remix IDE [34,35].

The parameters used to measure the network performance are given in Tables 2–4. The description of the test network is as follows:

- **In the case of TCP**: With a network connection established between the two end devices and a TCP bandwidth restriction of 26 Mbps, data packets were transferred from device 1 to device 2 to test the network.
- **In the case of UDP**: Set the bandwidth restriction to 2 Mbps, and send several UDP data packets from one end device to another end device.
- **Routing Information**—Using the Multichain blockchain development platform, a private blockchain is created and put the routing data (IP, MAC, etc.) to the controller in the multichain.

Table 2. Parameters used in Single-controller topology.

S.No.	Parameters	Values
1	TCP Bandwidth	26 Mbps
2	TCP Time (s)	180 s
3	UDP Bandwidth	2 Mbps
4	TCP Time (s)	180 s
5	No of Computers	23
6	No. of SDN Controller	1
7	Switches	8

Table 3. Parameters used in Multi-controller topology.

S.No.	Parameters	Values
1	TCP Bandwidth	26 Mbps
2	TCP Time (s)	180 s
3	UDP Bandwidth	2 Mbps
4	TCP Time (s)	180 s
5	No of Computers	23
6	No. of SDN Controller	3
7	Switches	8

Table 4. Blockchain Parameters.

S.No.	Parameter Name	Values
1	Blockchain platform	Multichain
2	No. of transactions	Not-fixed
3	Block Header Size	80 Bytes

The test networks with a single-controller and multi-controller topologies are shown in Figures 1 and 2, respectively.

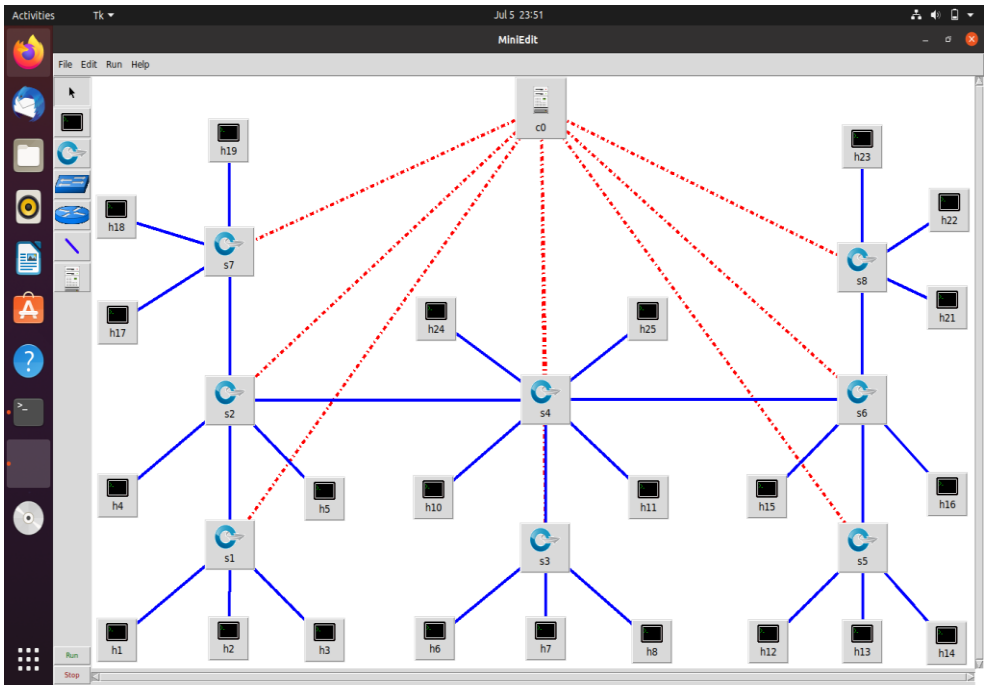

Figure 1. Single-controller SDN Test Network.

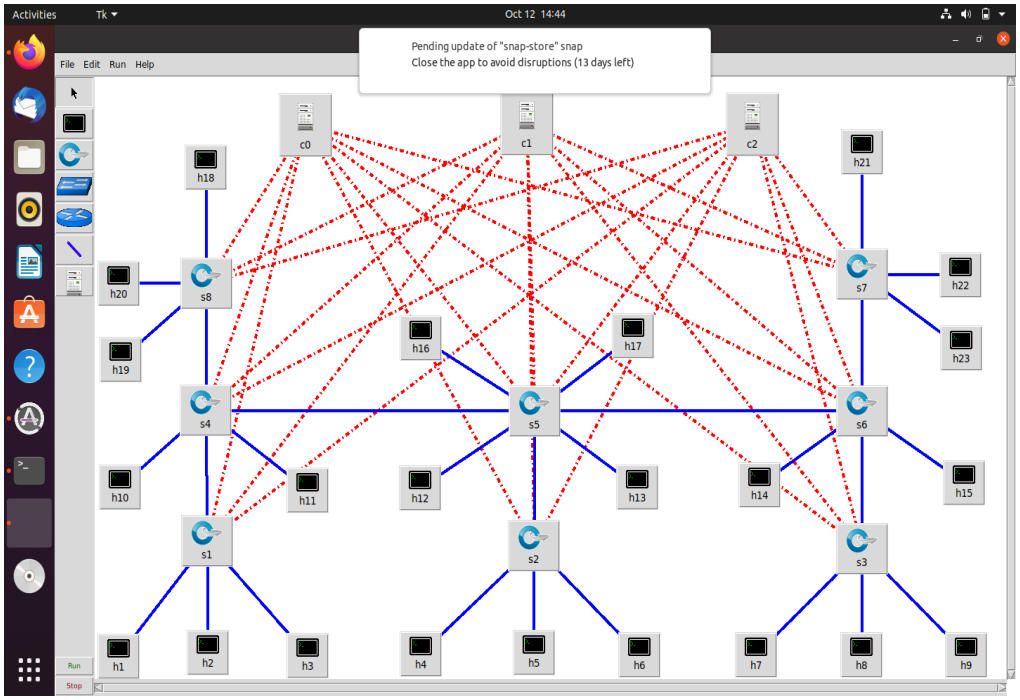

Figure 2. Multi-controller SDN Test Network.

Figures 3 and 4 display the state of smart contracts that have been deployed and the cost of transactions on the Ethereum blockchain. The transaction cost in this instance is 589,671 gas, as can be observed. Therefore, creating a private blockchain in the SDN would be economical. With the aid of the Multi-chain blockchain development platform, a private blockchain for network security was created. By uploading our networking routes and controller information to the blockchain, followed by regular checks to see if the network routes have changed or not, the network is made more secure by performing security checks. Therefore, we only need to isolate that one malicious node from the network

and avoid performing security checks on the entire network. Here, an attacker tries to provide bogus information to the controller in order to change the routing information. Routing attacks change how packets are routed on SDN by manipulating routing control information [21]. As a result, multi-controller nodes employing multichain can detect a malicious node and an intrusion attempt to change the routing information.

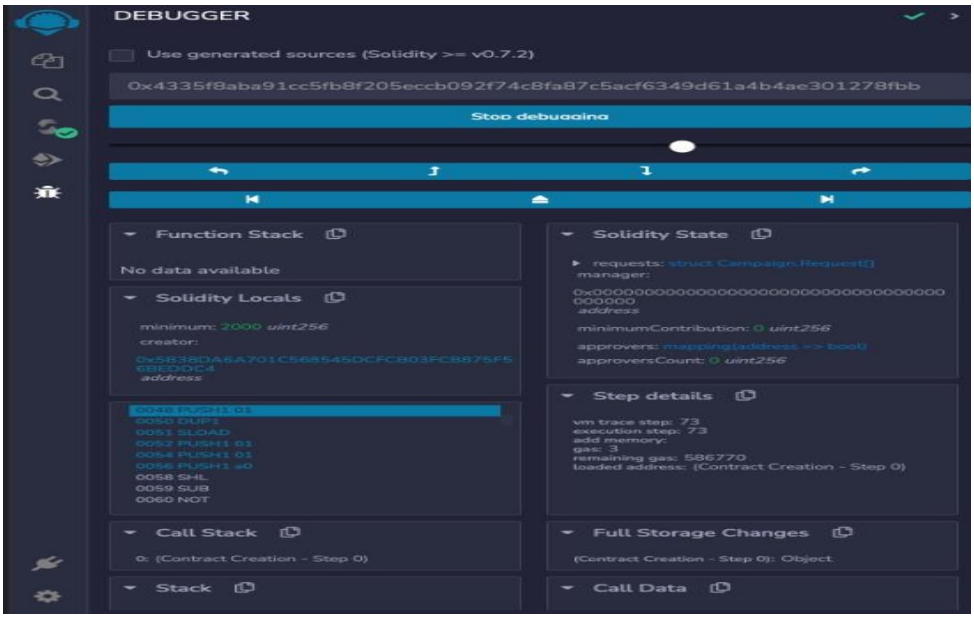

Figure 3. Smart contract deployment.

Figure 4. Transaction cost.

4.2. Simulation Metrics

A description of the test parameters used is provided below:

- **Bandwidth**
- **Throughput**
- **Latency (Delay)**
- **Jitter**

Any network's performance is determined by the bandwidth and the protocols being utilized. After adjusting some basic variables, including the number of devices in the network, the bandwidth, and the amount of time, the network was tested. The iPerf network tool is used to identify the network characteristics for two distinct types of networks. The aforementioned parameters are applied with the following constraints:

- **Time:** A brief test period of 180 s for each of the three networks was sufficient to produce the desired findings.
- **Bandwidth:** For all three networks, the bandwidth used to extract the TCP and UDP network packet characteristics was set to a constant value of 26 Mbps and 2 Mbps, respectively.
- **No of Devices:** The controller in topology one was the only network device modified. In the first type of architecture, the single-controller topology, we only need a single SDN controller. Specifically, we use three SDN controllers in a master–slave configuration for the multi-controller SDN topology.

4.3. Experimental Results

Traditional TCP Network vs. SDN TCP Network

We evaluated both networks for 180 s while restricting the bandwidth to 25 Mbps and only using TCP protocol packets. Below is a graph of the network properties.

Figures 5 and 6 make it very evident that SDN offers more bandwidth stability than traditional networks.

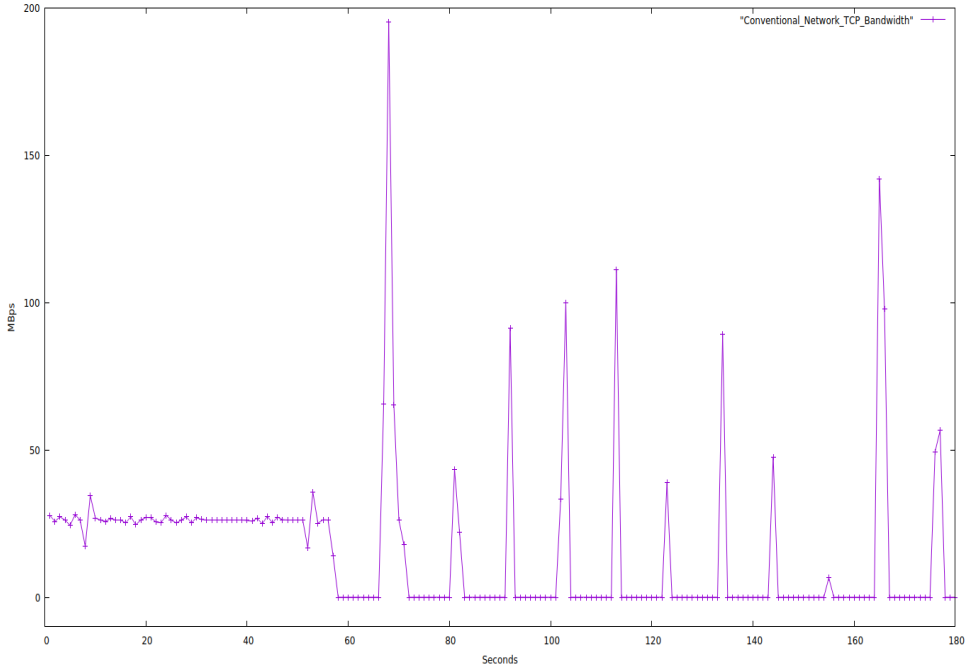

Figure 5. Traditional Network TCP Bandwidth Fluctuation.

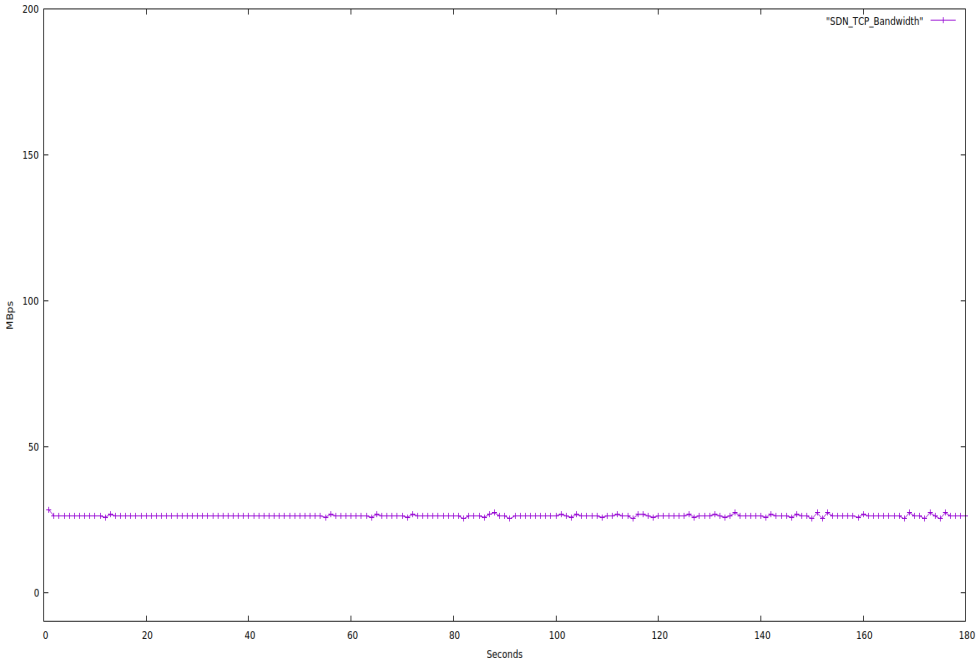

Figure 6. SDN TCP Bandwidth Fluctuation.

Figure 7 clearly shows that throughput nearly stays constant for a constant bandwidth in an SDN network. In contrast, there is so much fluctuation in a traditional network that the transfer rate occasionally drops to zero and stays there for a while. We also noticed that SDN could transfer more data than the conventional network for the same period.

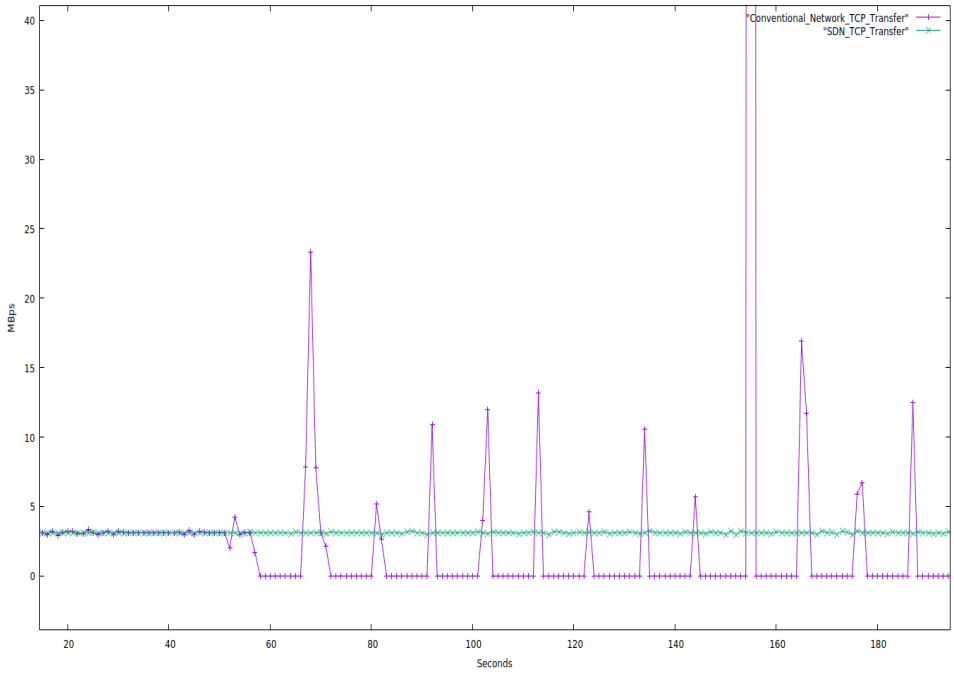

Figure 7. Transfer Rate/Throughput of TCP SDN vs. Traditional Network.

4.4. Traditional UDP Network vs. SDN UDP Network

Both networks are evaluated for 180 s while restricting the bandwidth to 1.05 Mbps and solely using UDP protocol packets. The characteristics of the network are depicted in Figures 8–11.

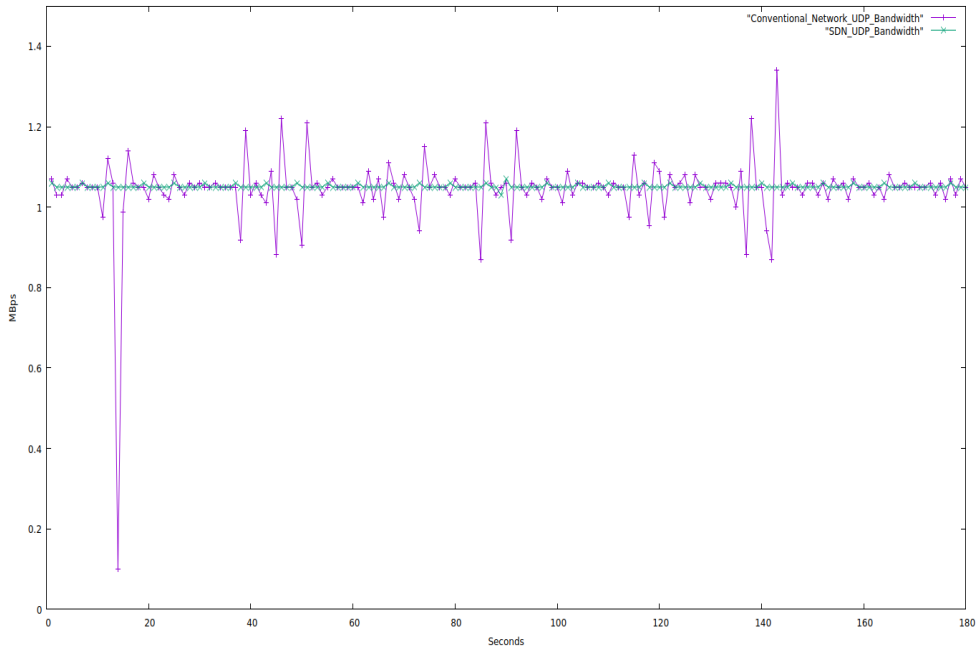

Figure 8. UDP Bandwidth Fluctuation of SDN vs. Traditional Network.

We fixed the bandwidth for the aforementioned tests, but how the network uses the bandwidth is up to it. With SDN, the bandwidth remains constant. SDN, i.e., supplies or provides us with a stable network. However, the traditional network's bandwidth fluctuates significantly, demonstrating that conventional networks are not stable.

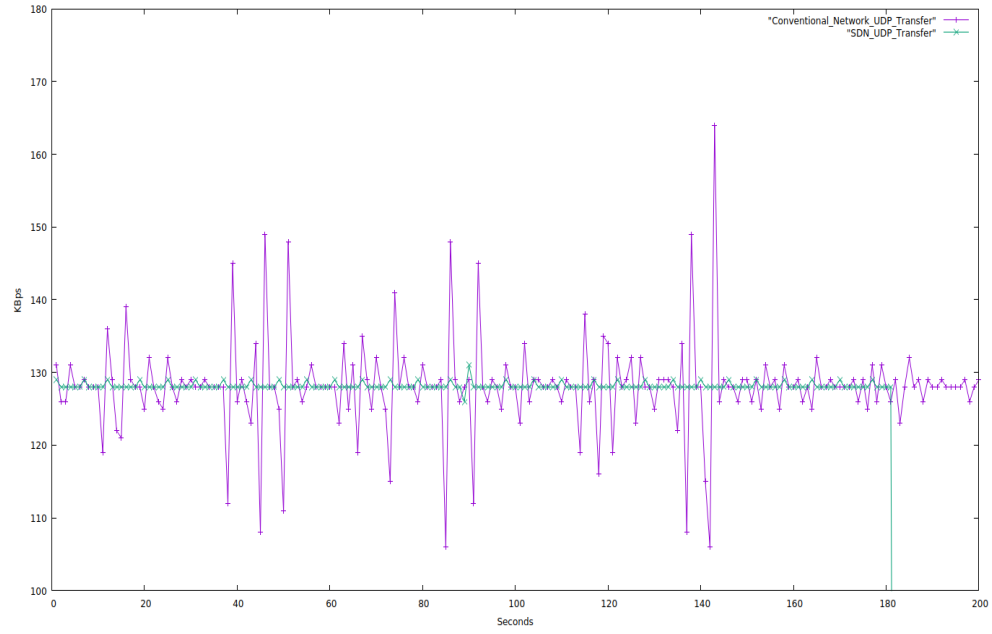

Figure 9. Transfer Rate/Throughput of SDN vs. Traditional Network.

The throughput is shown in the above figure, and as we can see, it is constant for the entire testing period for SDN, but it abruptly drops to zero at 182 s. In contrast, we can see that for Traditional Network, the throughput fluctuates and is not even close to being a constant value. We can also see that it still transfers data after 182 s, i.e., it continues to transfer data even after the testing period has ended. This is concrete evidence that SDN is significantly faster than traditional networks.

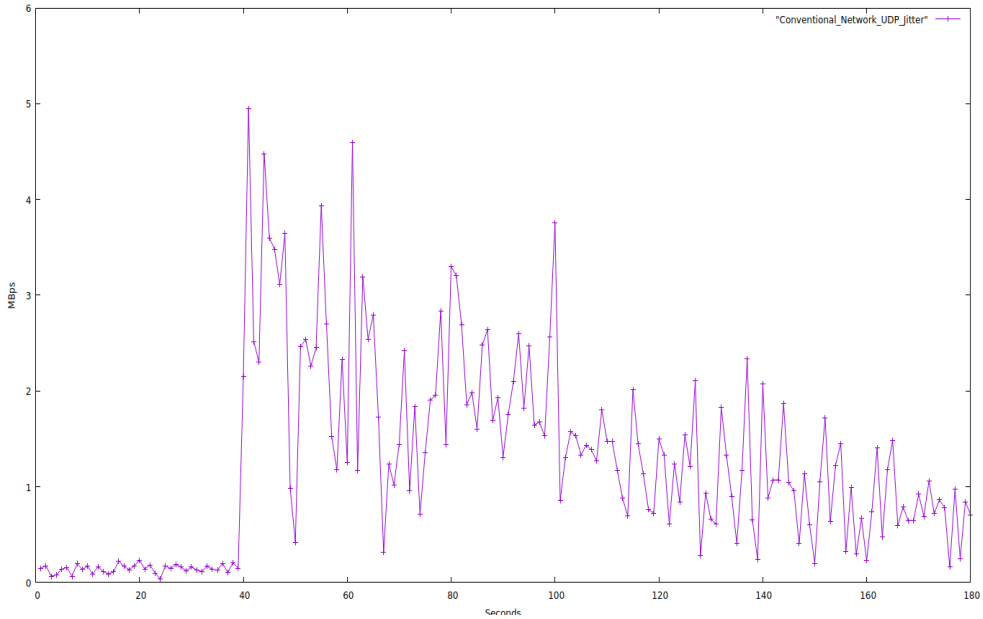

Figure 10. Jitter variation in Traditional Network.

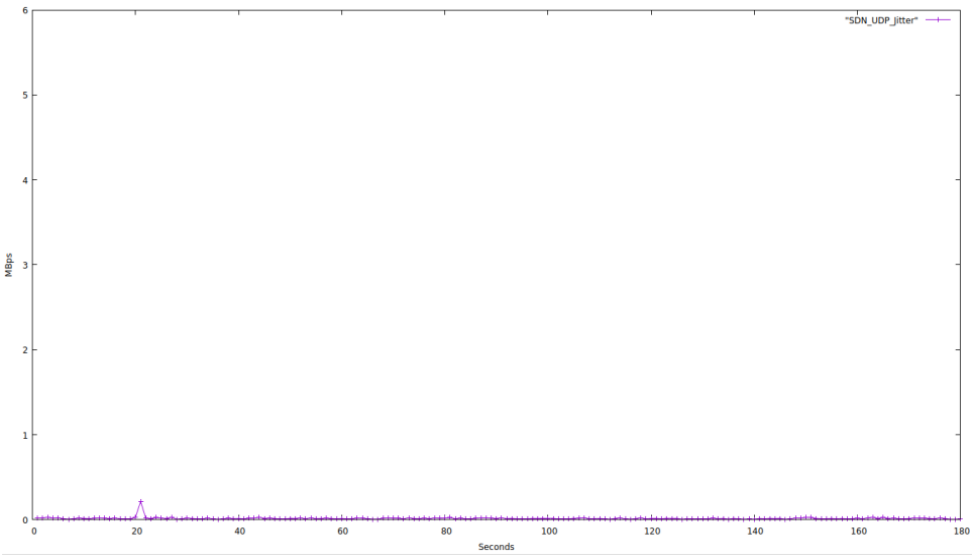

Figure 11. Jitter variation in SDN.

According to our testing, the average jitter for SDN networks is 0.112 Mbps, which is a meager value. In contrast, the average jitter for Traditional networks was found to be much higher than the corresponding SDN networks. Jitter is the effect we experience when there is a delay in data transfer in a network. For a network to be good, the jitter should be as low as possible.

4.5. Single-Controller TCP SDN vs. Multi Controller TCP SDN

In this part, we contrast a topology with just one SDN controller with one with several (in this case, three) SDN controllers. This topology has 34 network devices, including 23 computers, 8 OpenFlow switches, and 3 SDN controllers. The single-controller topology has 32 network devices in total, including 23 computers, 8 OpenFlow switches, and one SDN controller. We tested both networks for 180 s while setting the bandwidth to 26.2 Mbps and using TCP protocol packets. The network characteristics are displayed next in Figures 12–16.

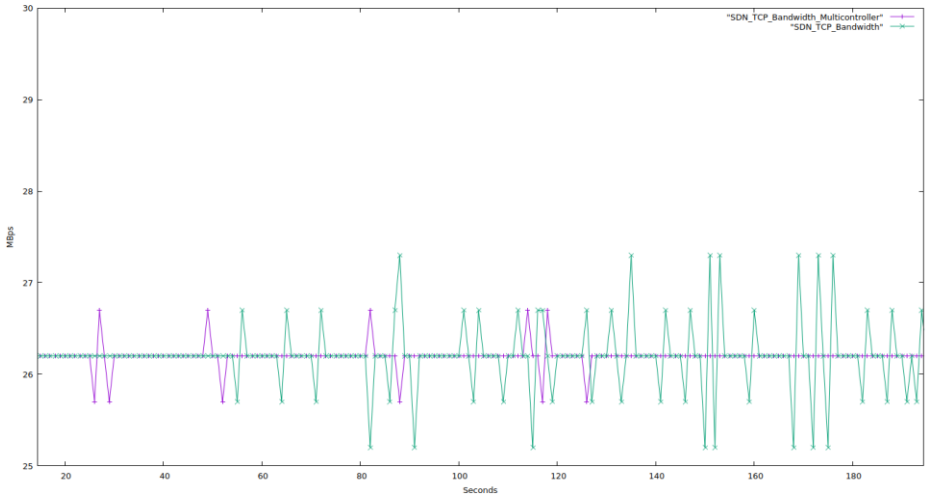

Figure 12. Bandwidth Fluctuation TCP single-controller SDN vs. TCP Multi-controller SDN.

Figure 12 shows that both networks' bandwidth fluctuates in a manner that is essentially the same, with no topology experiencing more significant fluctuations than the others.

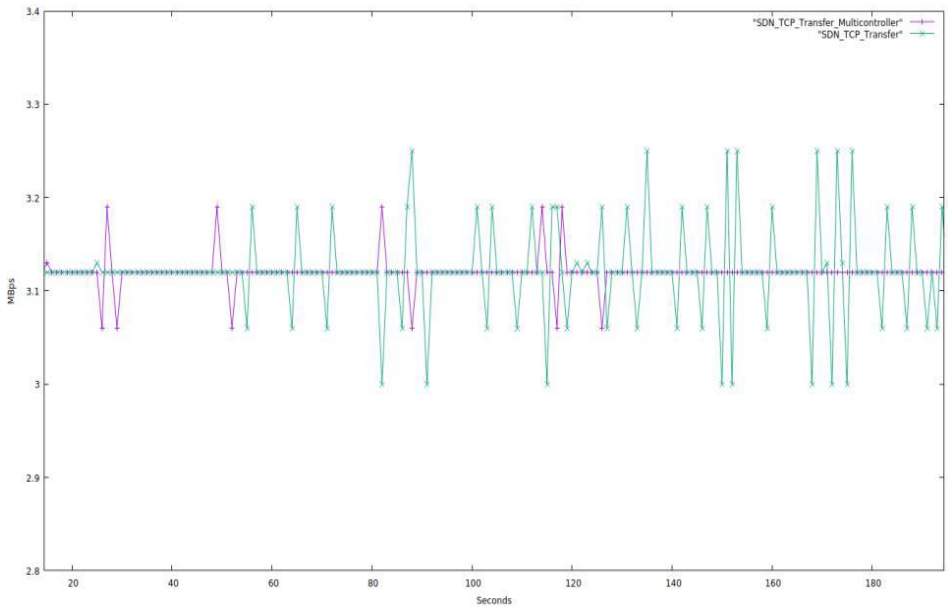

Figure 13. Transfer Rate/Throughput of TCP single-controller SDN vs. TCP Multi-controller SDN.

Figure 13 shows that both topologies' data transfer rates are similar and that the multi-controller architecture performs somewhat better than the single-controller topology.

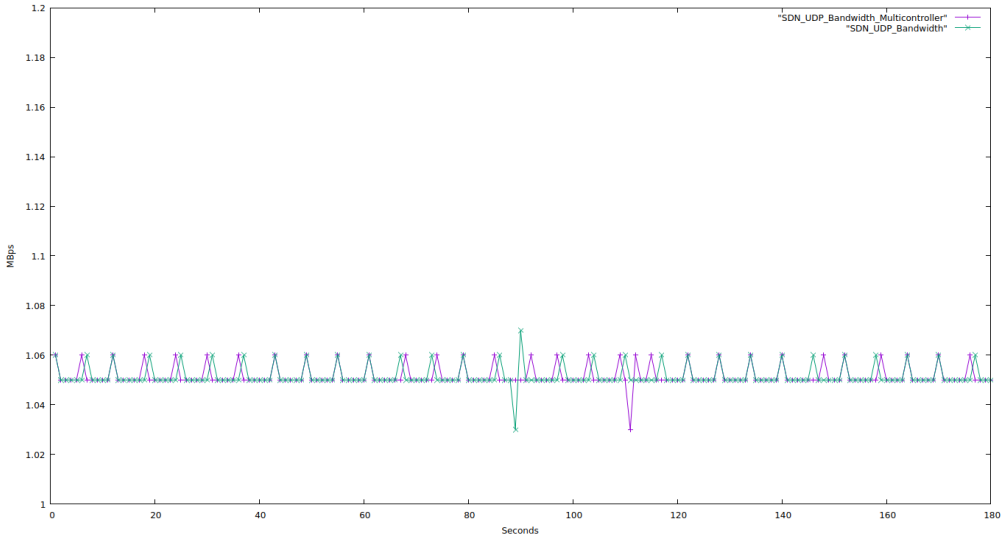

Figure 14. UDP bandwidth fluctuation of single-controller SDN vs. Multi-controller SDN.

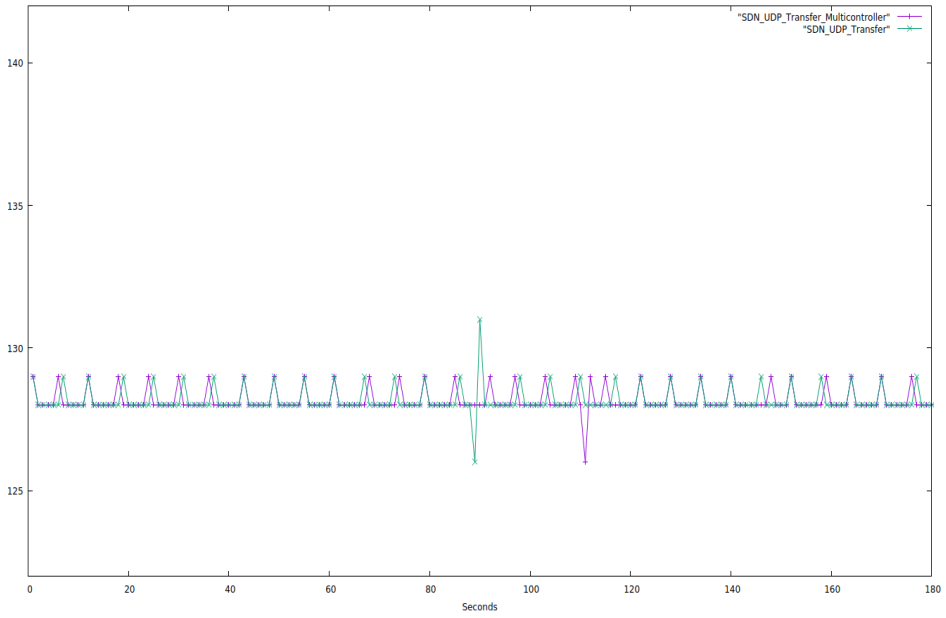

Figure 15. Transfer rate/Throughput of the UDP single-controller SDN vs. Multi-controller SDN.

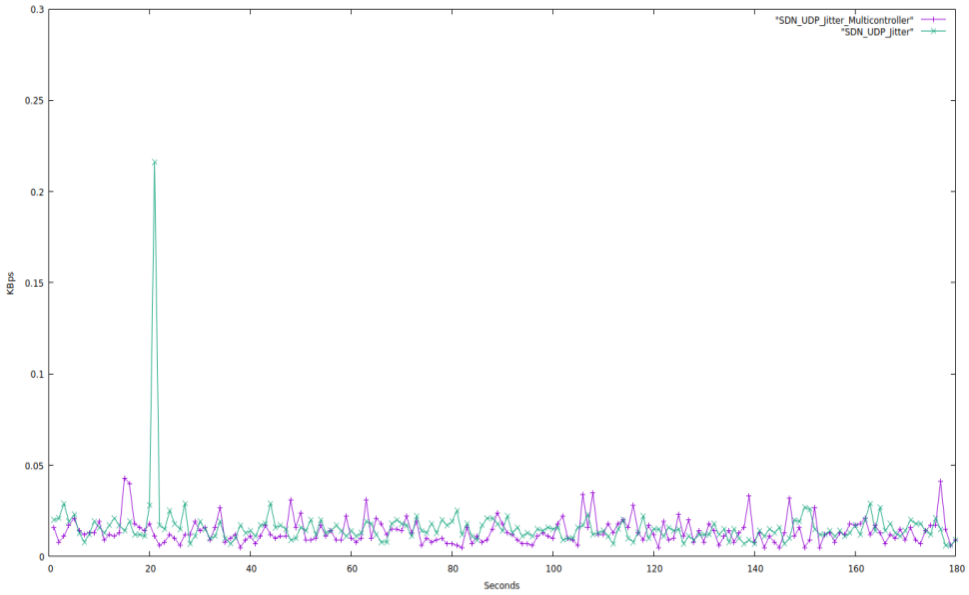

Figure 16. Jitter fluctuation single-controller SDN vs. Multi-controller SDN.

4.6. Single-Controller UDP SDN vs. Multi Controller UDP SDN

In this section, we contrast a topology with a single SDN controller with one with multiple (in this example, three) SDN controllers. In this topology, there are 41 network devices total, including 30 computers, 8 OpenFlow switches, and 3 SDN controllers. In the single-controller topology, there are 39 network devices total, including 30 computers, 8 OpenFlow switches, and one SDN controller. We tested both networks for 180 s while setting the bandwidth to 1.05 Mbps and exclusively using UDP protocol packets.

Figure 14 shows that both topologies—single-controller and multi-controller—show an average fluctuation of 0.01 Mbps, a very negligible fluctuation. As a result, we can conclude that both topologies perform exactly as they should. When compared to the single-controller test network, the multi-controller network performs somewhat better. It follows that a very big network will benefit from the multi-controller's superior performance.

Figure 15 shows that there is little to no difference in the throughput of the two topologies, indicating that they are both transmitting data at nearly the same pace. This suggests that the two topologies behave at the same pace. Although when compared to the single-controller test network, the multi-controller network performs somewhat better. It follows that a very big network will benefit from the multi-controller's superior performance.

It is evident from the aforementioned graph that the jitter for both topologies varies, but the average value of jitter for both topologies equals 0.02 Kbps for single-controller SDN and 0.0202 Kbps for multiple controller SDN, which is almost the same. The multi-controller routing information is added to the blockchain. As a result, we can state categorically that multi-controller network topology is superior to a single controller. However, the multi-controller performs better in terms of jitter fluctuation in the event of any breakdown. However, if a single controller quits functioning, the entire network fails.

5. Conclusions

This work presented a Lightweight Secure Framework that blends Software-Defined Networking with blockchain technology. The mininet network simulator is used to model the proposed SDN framework. The data transfer rate or network throughput, bandwidth variation, and jitter have all been used to assess the performance of single-controller-SDN and multi-controller-SDN networks. The results demonstrate that the suggested framework performs better than the conventional single-controller-SDN architecture in terms of throughput, bandwidth fluctuation, and jitter. When three SDN controllers fail in

a master-slave configuration in the suggested framework, the multi-controller performs better in terms of bandwidth, transfer rate, and jitter fluctuation. Hence, single points of failure are protected by this dual refrigeration system. If the master controller shuts down for any reason, the following slave controller steps in to act as the master controller until the previous master controller comes back online and begins acting as planned. This ensures that the entire topology is not affected. The network so continues to operate normally as if nothing unexpected had occurred.

Author Contributions: Conceptualization, S.S. and V.M.; methodology, S.S. and V.M.; audit parameters, S.S., V.M., and S.N.; validation, S.S. and V.M.; formal analysis, S.S. and V.M.; investigation, S.S. and V.M.; resources, S.S. and V.M.; writing—original draft preparation, S.S.; writing—review and editing, S.S., V.M., and S.N.; supervision, S.S. and S.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: This research was supported by the Thapar Institute of Engineering and Technology, Patiala, under the seed money research project grant scheme with Ref. TU/DORSP/7/2/2022.

Conflicts of Interest: The authors declare that they have no conflicts of interest.

References

1. Yang, F.Y.; Zhang, J.M.; Xie, W.L.; Wang, M.; Wang, H. Analysis of 5G cellular network architecture. *Telecommun. Sci.* **2015**, *31*, 46–56.
2. Akshatha, N.M.; Jha, P.; Karandikar, A. A centralized SDN architecture for the 5G cellular network. In Proceedings of the 2018 IEEE 5G World Forum (5GWF), Silicon Valley, CA, USA, 9–11 July 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 147–152.
3. Long, Q.; Chen, Y.; Zhang, H.; Lei, X. Software defined 5G and 6G networks: A survey. *Mob. Netw. Appl.* **2022**, *27*, 1792–1812. [CrossRef]
4. Lv, Z.; Kumar, N. Software defined solutions for sensors in 6G/IoE. *Comput. Commun.* **2020**, *153*, 42–47. [CrossRef]
5. Zebari, G.M.; Zebari, D.A.; Al-zebari, A. Fundamentals of 5G cellular networks: A review. *J. Inf. Technol. Inform.* **2021**, *1*, 1–5.
6. Yang, T.; Qin, M.; Cheng, N.; Xu, W.; Zhao, L. Liquid software-based edge intelligence for future 6G networks. *IEEE Netw.* **2022**, *36*, 69–75. [CrossRef]
7. Yungaicela-Naula, N.M.; Vargas-Rosales, C.; Pérez-Díaz, J.A.; Zareei, M. Towards security automation in software defined networks. *Comput. Commun.* **2022**, *183*, 64–82. [CrossRef]
8. Singh, S.; Delia Jurcut, A. (Eds.) *Revolutionary Applications of Blockchain-Enabled Privacy and Access Control*; IGI Global: Hershey, PA, USA, 2021.
9. Singh, S.; Sharma, R.M. Heuristic based coverage aware load balanced clustering in WSNs and enablement of IoT. *Int. J. Inf. Technol. Web Eng. IJITWE* **2018**, *13*, 1–10. [CrossRef]
10. Singh, S.; Mohan Sharma, R. (Eds.) *Handbook of Research on the IoT, Cloud Computing, and Wireless Network Optimization*; IGI Global: Kurukshetra, India, 2019.
11. Singh, S.; Kumar, P. MH-CACA: Multi-objective harmony search-based coverage aware clustering algorithm in WSNs. *Enterp. Inf. Syst.* **2020**, *14*, 1325–1353. [CrossRef]
12. Pokhrel, S.R.; Singh, S. Compound TCP performance for industry 4.0 WiFi: A cognitive federated learning approach. *IEEE Trans. Ind. Inform.* **2020**, *17*, 2143–2151. [CrossRef]
13. Mattos, D.M.F.; Duarte, O.C.M.B.; Pujolle, G. A lightweight protocol for consistent policy update on software-defined networking with multiple controllers. *J. Netw. Comput. Appl.* **2018**, *122*, 77–87. [CrossRef]
14. Rafique, W.; Qi, L.; Yaqoob, I.; Imran, M.; Rasool, R.U.; Dou, W. Complementing IoT services through software defined networking and edge computing: A comprehensive survey. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 1761–1804. [CrossRef]
15. Xue, K.; Meng, W.; Zhou, H.; Wei, D.S.; Guizani, M. A lightweight and secure group key based handover authentication protocol for the software-defined space information network. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 3673–3684. [CrossRef]
16. Huo, L.; Jiang, D.; Lv, Z. A software—defined networks—based measurement method of network traffic for 6G technologies. *Trans. Emerg. Telecommun. Technol.* **2022**, *33*, e4172. [CrossRef]
17. Abid, M.A.; Afaqui, N.; Khan, M.A.; Akhtar, M.W.; Malik, A.W.; Munir, A.; Ahmad, J.; Shabir, B. Evolution towards smart and software-defined internet of things. *AI* **2022**, *3*, 100–123. [CrossRef]

18. Singh, S.; Nikolovski, S.; Chakrabarti, P. GWLBC: Gray Wolf Optimization Based Load Balanced Clustering for Sustainable WSNs in Smart City Environment. *Sensors* **2022**, *22*, 7113. [CrossRef]
19. Chica, J.C.C.; Imbachi, J.C.; Vega, J.F.B. Security in SDN: A comprehensive survey. *J. Netw. Comput. Appl.* **2020**, *159*, 102595. [CrossRef]
20. Bannour, F.; Souihi, S.; Mellouk, A. Distributed SDN control: Survey, taxonomy, and challenges. *IEEE Commun. Surv. Tutor.* **2017**, *20*, 333–354. [CrossRef]
21. Farris, I.; Taleb, T.; Khettab, Y.; Song, J. A survey on emerging SDN and NFV security mechanisms for IoT systems. *IEEE Commun. Surv. Tutor.* **2018**, *21*, 812–837. [CrossRef]
22. Javanmardi, S.; Shojafar, M.; Mohammadi, R.; Nazari, A.; Persico, V.; Pescapè, A. FUPE: A security driven task scheduling approach for SDN-based IoT–Fog networks. *J. Inf. Secur. Appl.* **2021**, *60*, 102853. [CrossRef]
23. Latif, S.A.; Wen, F.B.X.; Iwendi, C.; Li-li, F.W.; Mohsin, S.M.; Han, Z.; Band, S.S. AI-empowered, blockchain and SDN integrated security architecture for IoT network of cyber physical systems. *Comput. Commun.* **2022**, *181*, 274–283. [CrossRef]
24. Daoud, W.B.; Obaidat, M.S.; Meddeb-Makhlouf, A.; Zarai, F.; Hsiao, K.-F. Tacrm: Trust access control and resource management mechanism in fog computing. *Hum.-Cent. Comput. Inf. Sci.* **2019**, *9*, 28. [CrossRef]
25. Blanco, B.; Fajardo, J.O.; Giannoulakis, I.; Kafetzakis, E.; Peng, S.; Pérez-Romero, J.; Trajkovska, I.; Khodashenas, P.S.; Goratti, L.; Paolino, M.; et al. Technology pillars in the architecture of future 5G mobile networks: NFV, MEC and SDN. *Comput. Stand. Interfaces* **2017**, *54*, 216–228. [CrossRef]
26. Zaidi, Z.; Friderikos, V.; Yousaf, Z.; Fletcher, S.; Dohler, M.; Aghvami, H. Will SDN be part of 5G? *IEEE Commun. Surv. Tutor.* **2018**, *20*, 3220–3258. [CrossRef]
27. Barakabitze, A.A.; Ahmad, A.; Mijumbi, R.; Hines, A. 5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges. *Comput. Netw.* **2020**, *167*, 106984. [CrossRef]
28. Ashtari, S.; Abdollahi, M.; Abolhasan, M.; Shariati, N.; Lipman, J. Performance analysis of multi-hop routing protocols in SDN-based wireless networks. *Comput. Electr. Eng.* **2022**, *97*, 107393. [CrossRef]
29. Persia, S.; Ferranti, L.; Salvo, P.; D'Alterio, F.; Matera, F.; Rea, L.; Lavacca, F.G. Ns3 and Mininet Codes to Investigate Complete 5G Networks. In Proceedings of the 2021 AEIT International Annual Conference (AEIT), Milan, Italy, 4–8 October 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–6.
30. Available online: www.mininet.org (accessed on 20 October 2022).
31. Available online: www.noxrepo.github.io (accessed on 20 October 2022).
32. Available online: www.iperf.fr (accessed on 20 October 2022).
33. Available online: www.multichain.com (accessed on 20 October 2022).
34. Available online: https://truffleframework.com/ganache (accessed on 20 October 2022).
35. Available online: www.metamask.io (accessed on 20 October 2022).