IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 11 November 2023, accepted 19 December 2023, date of publication 25 December 2023,
date of current version 11 January 2024.

Digital Object Identifier 10.1109/ACCESS.2023.3347350

RESEARCH ARTICLE

ITor-SDN: Intelligent Tor Networks-Based SDN for Data Forwarding Management

FOUAD A. YASEEN[1], NAHLAH ABDULRAHMAN ALKHALIDI[2], AND HAMED S. AL-RAWESHIDY[3], (Senior Member, IEEE)

[1]Electronics and Communication Department, College of Engineering, University of Baghdad, Baghdad 10071, Iraq
[2]Computer Science Department, College of Science, University of Baghdad, Baghdad 10071, Iraq
[3]Department of Electronic and Electrical Engineering, College of Engineering, Design, and Physical Sciences, Brunel University London, Uxbridge, UB8 3PH London, U.K.

Corresponding author: Hamed S. Al-Raweshidy (Hamed.Al-Raweshidy@brunel.ac.uk)

ABSTRACT Tor (The Onion Routing) network was designed to enable users to browse the Internet anonymously. It is known for its anonymity and privacy security feature against many agents who desire to observe the area of users or chase users' browsing conventions. This anonymity stems from the encryption and decryption of Tor traffic. That is, the client's traffic should be subject to encryption and decryption before the sending and receiving process, which leads to delay and even interruption in data flow. The exchange of cryptographic keys between network devices plays a pivotal and critical role in facilitating secure communication and ensuring the integrity of cryptographic procedures. This essential process is time-consuming, which causes delay and discontinuity of data flow. To overcome delay or interruption problems, we utilized the Software-Defined Network (SDN), Machine Learning (ML), and Blockchain (BC) techniques, which support the Tor network to intelligently speed up exchanging the public key via the proactive processing of the Tor network security management information. Consequently, the combination network (ITor-SDN) keeps data flow continuity to a Tor client. We simulated and emulated the proposed network by using Mininet and Shadow simulations. The findings of the performed analysis illustrate that the proposed network architecture enhances the overall performance metrics, showcasing a remarkable advancement of around 55%. This substantial enhancement is achieved through the seamless execution of the innovative ITor-SDN network combination approach.

INDEX TERMS Anonymity, blockchain, ML, SDN, Tor networks.

I. INTRODUCTION

People want to communicate freely without worry of being tracked or scrutinized; this leads to rising demand for anonymity and privacy in the digital era. Further, users like to protect their personal information from unauthorized access. Therefore, a variety of technologies are used to protect privacy and anonymity. Tor [1] and SDN networks [2] are different and complementary technologies used to provide secure communications. Tor networks are prominent for protecting users' privacy and anonymity, making these technologies ideal for applications that need to communicate securely. On the other hand, SDN networks are increasingly utilized in networking environments where automated and flexible management is desired and needed, with the SDN controller handling the forwarding process efficiently [3]. Another technology provides a distributed database among the nodes to increase data security by sharing the information of nodes into a domain. A blockchain is a shared database that is distributed the database among the network nodes. Creating a blockchain ensures the security and fidelity of a record of information and generates confidence without the necessity for an authorized third party [4]. Machine Learning (ML) is an application of Artificial intelligence (AI) that allows machines to extract and learn knowledge from data autonomously. AI is the broad conception of enabling

The associate editor coordinating the review of this manuscript and approving it for publication was Zhenliang Zhang.

a machine to act, sense, or adapt like a human. ML is a promising technology for which existing solutions need better optimization or automation. So, ML assists network management, operations, configuration, and supporting proactive and predictive roles for improving network performance [5].

Tor (also known dark web) provides anonymous transmissions utilizing uncommon application layer protocols and approval schemes. Anonymity with the Tor network is achieved by using the Tor domain that involves a variety of network devices such as routers, bridges, relays, entrances, and exits for this domain. These nodes apply the Onion routing protocol for routing the encrypted data [6]. Tor uses the Onion routing protocol to forward data after three coats of data encryption. Applying the principle of Onion routing protocol by the Tor network gives strength and durability to hide the identity of the users. Each node receives Tor traffic a single layer of encryption is drawn, and this process is then repeated in a similar manner for the next hop [7]. The entry node receives the encrypted data traffic from the Tor client and removes the first encrypted layer. In contrast, the exit node removes the last encryption layer and transmits the data traffic to the destination. The intermediate devices in the Tor network domain are unaware of the eventual destination of the Tor traffic, and the user's identity is kept hidden. The exit router will be the sole recipient of knowledge about the target destination upon decrypting the third layer of the Tor traffic.

The SDN has three distinct layers, application, control, and infrastructure layers. The Application Layer (AL) contains services that link the Control Layer (CL) via the Application Programming Interface (API). The CL manages the SDN controller to configure, control, and monitor the infrastructure network. The Infrastructure Layer (IL), also known as (data plane) contains virtual or physical network devices that forward traffic according to the instructions of the SDN controller. Thus, the controller regards the brain of the network system in SDN [8]. The AL links the CL via the Northbound Application Programming Interface (N-API), where the application services transfer APIs from the AL to the CL. In contrast, the CL exchanges the control messages through the Southbound API (S-API) with the IL.

The cooperation between the SDN and Tor technologies makes traffic management flexible, more secure, and hard to track. The SDN can control the traffic forwarding inside the Tor network domain. The SDN controller needs to have the node's information such as device status, traffic pass through the node, edge devices location, and links capacity. This information is shared with the application layer, which utilizes ML to enable the SDN controller to build forwarding and routing tables. That will enhance and automate by applying ML with blockchain at the application layer to handle and monitor Tor network domain nodes.

The summarizing of our contributions is as follows:

- We apply SDN technology to enhance the performance of the Tor network. This procedure improves resilience against powerful Tor traffic management and forwarding.
- We use the blockchain distributed database to be used by the SDN controller to build its tables and provision to forwarding and routing the Tor traffic.
- We use ML to support the SDN to manage and make optimum decisions for building its forwarding tables intelligently. All these technologies were collaborated in our proposal (ITor-SDN) to enhance the data traffic management.

The paper is organized as follows: Section II briefly describes related works that discussed anonymous communication, SDN networks, ML, and blockchain technologies. The preliminaries for the Tor network concept, SDN approach, blockchain technique, and the concept of ML for supporting communication networks are in Section III. Section IV presents the proposed network description. Simulation and performance evaluation of our proposal is presented in Section V. Finally, Section VI concludes the paper.

II. RELATED WORKS

Anonymous has been studied from several aspects and integrated with modern networking technologies. Many researchers are interested in anonymous communication, SDN, blockchain database, and ML to provide robust and powerful network performance. Although these technologies have emerged at different times over two decades, they have demonstrated agile approaches to complement and enhance the performance of communication networks.

Diverse anonymous communication web systems are widely deployed worldwide to protect the privacy of users' communications and provide anti-censorship services. Ling et al. [9] applied the SDN switch at the endpoint to receive the routed traffic to the target server and change the window size of the TCP negotiation to alter the traffic speed at the server. Moreover, they modulated a hidden signal inside the traffic that was handled via the anonymous network and got the SDN switch at the end user. According to [10], the SDN has been used to enhance ransomware relief by analyzing the demeanor of widespread ransomware. Two time-dependencies are suggested to isolate the network devices that reacted to the threat without affecting the whole network performance. In [11], suggested a new network architecture that added an additional Tor layer to anonymize the communication between the application and control layers of the SDN. They used an SDN controller, VMware, Wireshark, and GNS3 network to simulate the proposed network architecture. Several technologies have implemented blockchain to provide user authentication, identification, and authorization commonly used on the Internet. The possibility of combining blockchain and the SDN to mitigate the challenges of the energy efficiency and security of the blockchain-enabled-SDN controller, by using a new routing protocol and a cluster structure. The proposed architecture employs private and public blockchains for P2P transmission [12]. The benefit of combining the SDN and blockchain for designing a general framework of blockchain-based

cooperative intrusion detection in the SDN network has been provided by [13]. Zakaria et al. [14] provided several scenarios to exploit the blockchain in SDN network domains to mitigate distributed denial-of-service within inter and intra domains. Moreover, they designed a secure method that permits the collaboration of several SDN domains to share in a distributed manner of the attack information. Guo et al. [15] supported the blockchain by ML to provide the network services integrity with the ML algorithms to execute dynamic adjustment, allocation, and prediction of network resources. Weng et al. [16], offered a distributed deep-learning framework to provide a motivated value-driven according to blockchain to provoke the participants to work perfectly. They were besides, ensuring the solitude of the participant's data with the capability of auditable via the training process. Blockchain and smart contracts were engaged to supply multi-agent networks of different providers and to maintain security and privacy for their data [17]. Authors of [18] presented collaboration of the blockchain and SDN controllers for a consensus global network view. On the contrary, both Blockchain encryption functions and non-encrypted processes can access the same computational server. To optimize the network design energy efficiency, they assigned computational resources and the block size together to be considered as the confidence components of SDN controllers and the resource needs of non-encrypted operations. According to [19], a novel priority management of a network flow had suggested depending on SDN and ML to optimize network traffic with high control by assigning a necessary traffic flow priority. The proposal realized flow prioritizing by selecting specific bits from the packet's header by utilizing the ML to prioritize data forwarding based on the priority levels by controlling the Differentiated Services Code Point (DSCP). To date writing this article, no published research studied the engagement of anonymity, SDN, blockchain, and ML together to improve the network's performance from aspects of security, traffic forwarding, and intelligent decision-making to tackle urgent network failure. Therefore, this issue urges us to provide a study to examine and investigate how different technologies could collaborate to enhance the performance of networks.

III. PRELIMINARIES

In this section, a quick overview of technologies that were utilized to perpetrate our proposal network and simplify the understanding of the principal idea. So, the Tor network, SDN concept, Blockchain, and ML will be curtly reviewed for the recent related studies.

A. TOR NETWORK

Tor, concise for the Onion Routing Protocol, is open-source and free software to support anonymous communication. Tor is a network that depends on Transmission Control Protocol (TCP) that ensures the connection establishment before sending data to the destination [20]. Figure 1 shows a simplified Tor network architecture. Tor network anonymity is built on three encrypted layers that make tracking a user's activity difficult by forwarding the traffic through a chain of routers. Generally, traffic is transferred via three types of nodes: Access Router (*AR*), Intermediate Router (*IR*), and Exit Router (*ER*) [21].

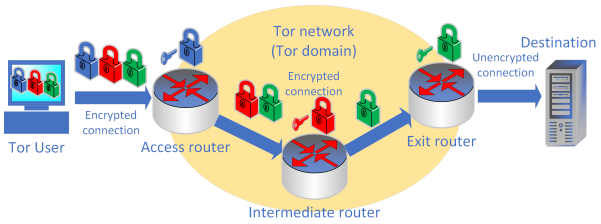

FIGURE 1. Simplified Tor network architecture.

Each of these nodes encloses specific functions:

- *AR*: This is the access gateway to the Tor domain that is selected from a set of onion routers.
- *IR*: The *IR* links the *AR* and *ER*. It serves as a guard to prevent entrance and departure from recognizing each other.
- *ER*: This is the exit point from the Tor network. The *ER* transfers the data to its destination.

Tor anonymity begins with the Tor user by encrypting the user's data based on the concept of the Onion Routing Protocol [22]. This protocol encrypts and encapsulates the data via three layers of encryption and encapsulation to permit the user to communicate anonymously on the Internet. Tor user initiates and establishes communication keys exchange session with each router [23]. The Access Router (*AR*) establishes key exchange through a certified Diffie-Hellman algorithm. Session keys are generated for the *AR* and *ER* by the Diffie-Hellman authenticated tunnel [24]. The data is doubly encrypted with session keys, letting each router peel back it and deliver the traffic to the next hop. That is, the encrypted traffic is forwarded through Tor network devices, and at each device, one of the encryption coatings is peeled. Tor routers direct response traffic in an opposing manner, and then the data is encrypted at every step until getting the Tor client.

B. SDN CONCEPT

The SDN network architecture can be divided into three planes or layers: the application layer, the control layer, and the data layer. The application layer communicates with the control layer via the Northbound Interface (NBI). While the control layer connects to the data layer through the Southbound Interface (SBI). The SDN technique is built based on the separation of the network control data and the user's data traffic. This technique provides dynamic network programmability through the Graphical User Interface (GUI) and Application Programming Interface (API); these interfaces are open-source software that allows to write scripts and automate the network administration to the SDN controller [2], [25]. The application layer contains a set of

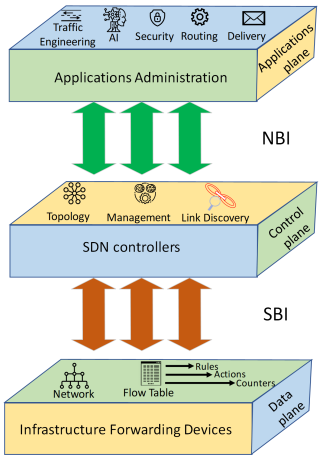

FIGURE 2. SDN network architecture layers.

applications that can be implemented by the SDN controller to configure and manage network services in the data layer. The control plane holds the SDN controller that maintains the networking devices group in the data plane. The SDN controller is regarded as the network brain that executes the demands of the applications and makes decisions, such as data forwarding and network topology infrastructure [26]. The data plane, also known as the forwarding layer has physical open flow switches that carry the data traffic according to a received flow table from the control layer. These switches forward the traffic based on the stored flow tables in its memory [9]. Figure 2 illustrates the SDN architecture layers.

C. BLOCKCHAIN

Emerging blockchain technology [27] allied with cryptocurrencies and online money transfer. Nevertheless, implementing the blockchain exceeds those limits. It has recently applied to boost technologies such as Internet-of-Things (IoT), SDN, ML, and virtualization [28]. Blockchain is a technique to encrypt documenting information that makes it difficult for the system to be changed, manipulated, or hacked. A blockchain is a distributed register that copies and distributes documents across the network's nodes partaking in the blockchain domain. Figure 3 explains the blockchain concept. To manipulate the blockchain, it needs a network that subsists on the Internet. Also, there are certain exchanges within the network for update purposes. These updates are necessary to maintain the distributed information system updated with the latest information block [29].

D. ML

ML gives the spirit to any machine to sense its circumstances and reactions to maximize the possibility of success to attain a goal. In other words, ML provides intelligence and decision-making abilities for devices like humans [30]. Applying ML algorithms in the SDN controller earns precision management to forward data traffic according to

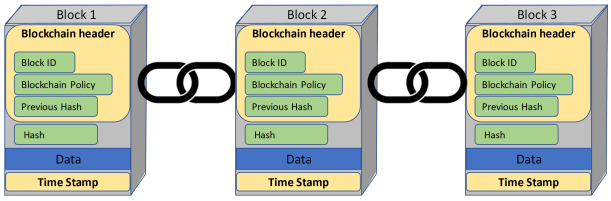

FIGURE 3. Blockchain architecture.

network needs [28]. By incorporating ML and blockchain techniques, ML algorithms can provide access to an exact scenery of a trusted, secure, shared data platform that can be used to host all blockchain records [31], [32], [33]. The blockchain file's record contains information about the public keys, router status, load traffic, and force priority to forward significant traffic of Tor routers according to administrator policy. The SDN controller receives information that the ML modifies through the NBI then the SDN controller makes the decisions that are delivered to the forwarding devices in the data plane via the SBI.

IV. PROPOSED NETWORK DESCRIPTION

The anonymity of the Tor network begins with the client, which should install the Onion Routing browser that encrypts the data through three encryption layers. The encrypted packet is directed to the Tor *AR* over the Internet. The *AR* receives the encrypted packet and removes the first encryption layer, then forwards it to the next *IR* according to the forwarding SDN's table. The *IR* decrypts the second encryption layer and sends the packet to the *ER* based on its flow table created by the SDN controller. The *ER* removes the final layer of encryption before sending the unencrypted packet to its final destination. The Diffie-Hellman algorithm, which generates symmetric keys from asymmetric keys, is used to decrypt data between any directly connected Tor nodes [34]. The generated keys should be sent to the blockchain server, which shares public keys and other information of all nodes with the ML to manage public keys, load balancing, router status, and traffic prioritization to make decisions according to the network forwarding policy. Figure 4 shows the architecture of the proposed Tor network.

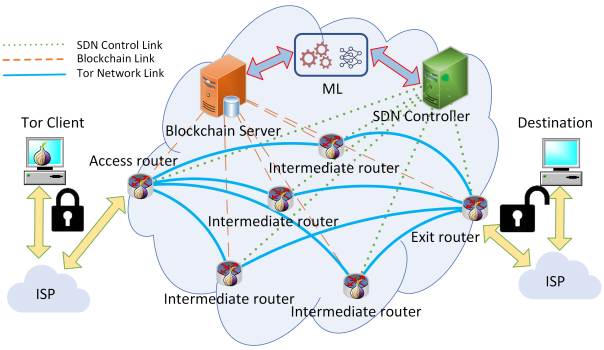

FIGURE 4. Proposed ITor-SDN network architecture.

A. TOR CLIENT

In the initial Onion Routing algorithm, the client receives the public keys of three randomly selected Tor nodes to build a circuit path. Each assigned node should share a public key with the Tor client and blockchain server. The client implements the Encapsulating Security Payload (ESP) [35] protocol to provide data protection. With ESP, a shared key is used to encrypt and decrypt the data transferred between communicating nodes. The client encrypts the data three times based on the received keys before forwarding it.

B. TOR NODES

In our proposal, the Tor nodes (access, intermediate, and exit nodes) are OpenFlow switches [36]. Therefore, the forwarding packet inside the node depends on the flow table that is built by the SDN controller's vision of the Tor network structure. The *AR* (also known as the entry relay) receives the encrypted packet from the client. It decrypts the first layer of the onion encryption, then forwards the encrypted packet to the next *IR* (also known as the bridge relay) according to the SDN flow table. The flow table controls the directing of the incoming encrypted packet from the *AR* to the *IR* and from *IR* to the *ER*, following the vision of the SDN controller of the Tor network topology. That is, the *AR* removes the second encryption layer before sending the packet in agreement with its flow table rules to *ER*. The role of the *ER* is to forward unencrypted data to the destination after removing the last encryption layer.

C. BLOCKCHAIN SERVER

The blockchain collects the information from the Tor nodes to build its records. This information record includes the interface public key, link load capacity, interface traffic status, and packet priority. The blockchain sends this information to the ML that supervises and manages the public key exchange, link load balancing, and data traffic prioritization to make decisions that support and inform the SDN controller to adapt the logical topology (changing the traffic departure interface) of the Tor network under the administrator requirements policy.

D. ML FUNCTIONS

One of the principal benefits of ML is its power to process extensive amounts of data and detect routines that would be impossible for humans to recognize. This advantage has led to incorporating ML technology across numerous areas such as communication networks, healthcare, finance, education, and more. One of the fields where ML has offered a significant and promising role is in the development of communication network performance. The Differentiated Services Code Point (DSCP) in the packet header acts as a node-to-node QoS priority control. The DSCP is the eight bits of the traffic class field that includes six bits of DSCP to manage the priority packet category. The remaining two bits are Explicit Congestion Notification (ECN) priority weights that split into two scopes; non-congestion and congestion control traffic [19]. In our proposal, the semi-supervised machine learning technique based on a simple One Dimension Convolution Neural Network (1D-CNN) [37] receives the public key, link load, router status, and data traffic prioritization as input parameters. Based on these parameters, the ML analyses data from Tor traffic to determine the appropriate decisions about forwarding traffic priority, probability of changing the public key, and link load balancing. The ML forces and controls traffic priority according to the precedence levels by controlling the DSCP bits in maintaining Tor network policies [19]. The lifetime of the public key in the Diffie-Hellman algorithm is 86400 seconds (one day); therefore, the ML generates a new public key by using the Diffie-Hellman algorithm and shares it with the blockchain server, which sends the updated records to the Tor nodes. Figure 5 illustrates the procedure of the proposed system.

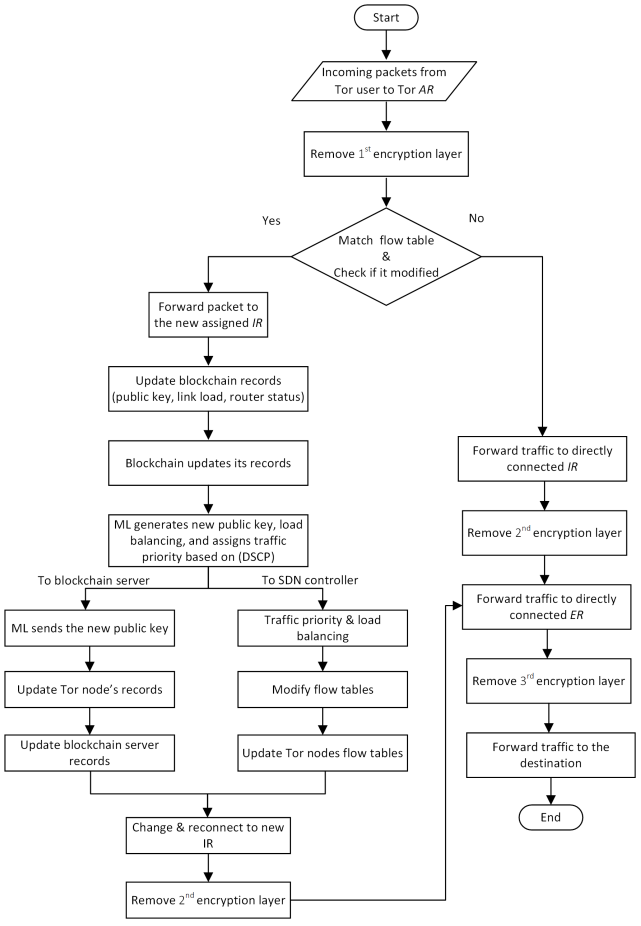

FIGURE 5. Proposed procedure and algorithm.

E. SDN CONTROLLER ROLE

SDN allows network administrators to supervise network services by abstracting lower-level functionality. Moreover, SDN provides a centralized vision for a network to represent, control, and optimize network resources via software

instead of hardware. ML supports SDN for agile and flexible monitoring and optimizing data flows. Furthermore, ML algorithms can be executed on top of SDN to enhance the efficiency and accuracy of network management to analyze data patterns from diverse sources. The SDN controller is responsible for constructing flow tables. It has a wide seen to handle forwarding data across Tor nodes. The Tor nodes (AR, IR, and ER) work under OpenFlow protocol [38]. The OpenFlow protocol is a well-known protocol that links the SDN controller to data-forwarding nodes through the southbound API. The SDN controller builds the flow tables based on its vision for the whole network and sends tables to underneath forwarding devices. Flow table entities are the rules that control how packets are relaid via network devices. These flow table entities have several essential fields, such as source and destination physical and logical addresses, protocol type, network ports, and QoS parameters. Furthermore, the flow table entities can be used to specify and prioritize flow traffic according to network policy and applications or services. Additionally, flow table entities have an action field that specifies which action of packet forwarding should be applied, such as forwarding it to a particular port interface or dropping it.

Figure 5 illustrates the proposed scheme applied to the ITor-SDN network. The income traffic from the Tor user subjects to remove the first encryption layer by the *AR* to know to which *IR* should forward the traffic after checking its flow table. If there is no change or modification in the flow table, the *AR* will forward the traffic to the *IR* that is already connected. Otherwise, deliver the traffic to the new assigned *IR* after accomplishing the flow table updating by the SDN controller in accordance with the ML decision. The blockchain server will register the up-to-date flow table in records (the public key, ingress port, egress port, load balance, and traffic priority). The possible changes in interfaces may have occurred between the *AR* - *IR* and/or *IR* - *ER*.

V. SIMULATION AND PERFORMANCE EVALUATION

To implement and evaluate our proposal, we used the Mininet [39] and Shadow emulators [40] to perform the proposed algorithm and Tor network based on the SDN technique. For combining the Mininet and Shadow, we programmed the API to control and exchange data between the proposed entities of our network. The Mininet environment creates a virtual network based on SDN, while the Shadow platform supports the Tor network. The host physical machine with specifications: 64GB DDR4 RAM, 8 Core 1.7GHz Intel Xeon Bronze 3106 CPU, and 3.8TB Intel SSD data storage. To perpetuate and execute the proposed network, we built the network by using the Mininet to emulate the physical networking devices. Moreover, the Mininet enables the creation of mixed network topologies, the configuration of SDN controllers, OpenFlow Switches (OFS), Blockchain servers, and hosts. The scenario of the proposed SDN network consisted of six OFSs that represent (Access, Intermediate, and Exit routers), an exterior SDN controller (we used the HPE-VAN controller), a database server (Blockchain server), and two hosts. The Application Program Interface (API) was programmed using Python programming code to enable communication with the SDN controller, ML, and Blockchain server.

Figure 6 refers to the network availability with increasing the number of flows injected into the proposed system. The performance of the Tor network was enhanced using the SDN technique, even with the growing number of flows. While, the highest performance was delivered by the Tor network when the combination of blockchain, SDN, and ML technologies managed the proposed network. This enhancement in the Tor network is due to the SDN controller rerouting the traffic path (routers) for the passing flows through it. The ML and blockchain server speeds up the distribution and configuration of the public keys between the routers. As we can see from Figure 6, the Tor network performance decreases with increasing the number of flows in the range (10 - 40 flows), the performance of the Tor network (18.5% −14%) and Tor-SDN (30%-28.5%) decreases also. Whilst, the ITor-SDN approach scored (56%-53%). That is, by increasing the number of flows in the proposed network, the efficiency of traffic management recorded the highest value.

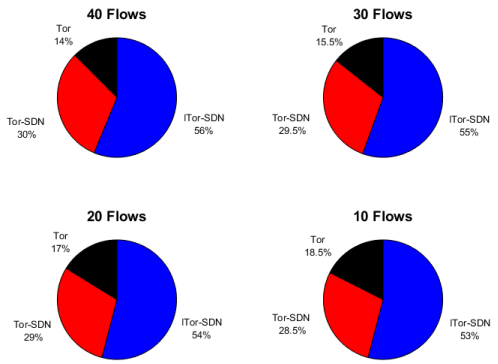

FIGURE 6. Network availability against increase the number of flows.

Figure 7 shows the average delay time required to negotiate and configure the public key between Tor network devices. We forced the proposed network to change the public key 16 times to examine the efficiency of the network to perform the decryption for transferring data traffic. Tor took a long time to decrypt and fulfill the public key changes to know the next hop for routing data traffic. Tor-SDN improves the process of decrypting and changing the public key due to the centralized SDN controller's monitoring of the entire Tor network. Furthermore, the provision of handling the data traffic within Tor-SDN gives high flexibility to make circuit-switched paths to pass flow data. The Tor-SDN efficiency increased by about 50%. Adding a blockchain server and SDN approach underneath ML enhanced the Tor network efficiency by 95% for managing data traffic and assigning the best path for incoming flows into the Tor network.

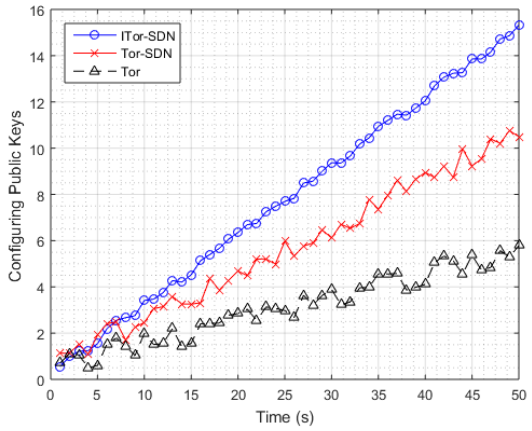

FIGURE 7. Average delay time to change the public key.

Figure 8 indicates the network access availability to be utilized by Onion clients. The availability is high when a few flows enter the network. However, with an increase in the number of flows, the suggested network shows a high performance in managing the income traffic. The Tor-SDN approach presents an acceptable improvement in traffic flow management because the SDN controller and ML handle the forwarding data flows across the Tor network according to administrated flow priority. Using the blockchain server elevated the performance of the proposed network by approximately 42% and 8% over the Tor and Tor-SDN networks, respectively. In other words, the ITor-SDN combination accelerated data transmission over the proposed network, and this improvement resulted from the proactive processing of Tor network management information by the ML and SDN controller with the support of the blockchain server.

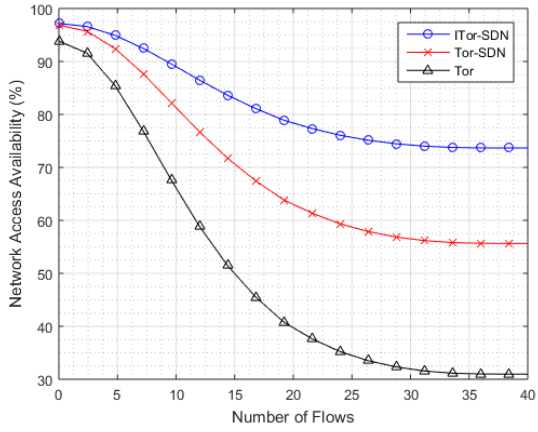

FIGURE 8. Network access availability.

Figure 9 describes the usage of network bandwidth to manage and implement the public keys exchange. The ITor-SDN utilizes the minimum bandwidth to complete public key exchange between any pair of routers because

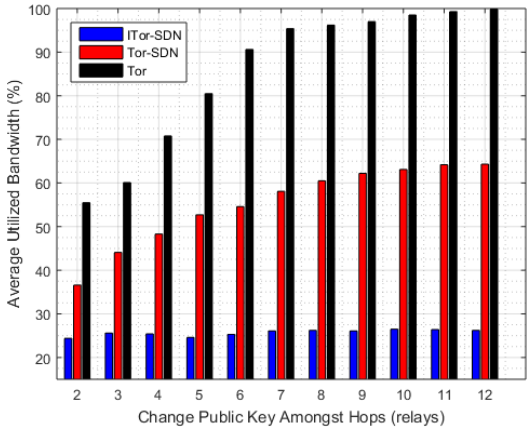

FIGURE 9. Utilized bandwidth for public keys exchange.

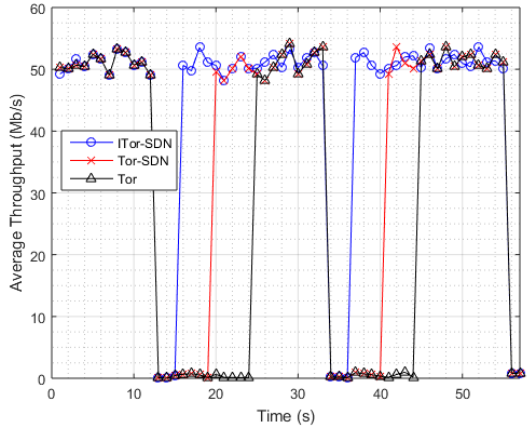

FIGURE 10. Throughput comparison for Tor, Tor-SDN, and ITor-SDN approaches.

there is no need for prior negotiation between routers before public key exchange. We administrate the network to change the public keys for more than three pairs of Tor network devices simultaneously. Although Tor-SDN reduced network bandwidth usage to perform this task, it consumed a relatively large amount (65%) of network bandwidth. While the proposed network(ITor-SDN) consumed a moderately small amount (nearly 27%) of network bandwidth.

Figure 10 illustrates the throughput comparison for proposed networks (Tor, Tor-SDN, and ITor-SDN). We measured the average throughput of the three network topologies that can be delivered to the Tor client during the change of the public key. The results indicate that the modified Tor network supported by SDN, ML, and blockchain server achieves flow continuity improvement when the traffic flow path is changed (public key exchange processing delay time). The data flow suffers an interruption due to a change or switch in the data flow path.ITor-SDN took the least time to achieve the path switching (just under 2 seconds). While Tor-SDN performs this process (about 6 seconds). In contrast, Tor had the longest time (nearly 11 seconds) to achieve the route change.

VI. CONCLUSION

Tor, The Onion Routing network, was designed to provide users with a way to browse the Internet without being identified. It is well known for its security features, which provide users anonymity and privacy against a third party trying to track their online activities. The anonymity of Internet users necessitates an exceptional process for data traffic, such as data cryptography. However, this brings about data-forwarding latency in Tor network devices. This trade-off between anonymity and data forwarding delay leads to discontinuity in the data flow. Encryption and decryption are crucial security measures; however, they can result in performance overhead, causing delays and interruptions in the data traffic flow. Through the efficient utilization of proactive processing techniques of the public keys for the Tor network devices, we were able to facilitate the rapid exchange of public keys between devices. This innovative approach allowed for a seamless and secure transmission of cryptographic information with less processing time. That is, enhancing the overall efficiency and effectiveness of public key exchanging procedures and keeping data traffic continuity. To solve this concern, we suggest the ITor-SDN network to ensure user anonymity and traffic flow continuity. Exploiting SDN, ML, and BC technologies that bolster the Tor network to expedite the exchange of public keys through the prescient processing of Tor network data. As such, the combination network (Tor, SDN, BC, and ML) guarantees consistent data flow to a Tor client. We simulated and emulated the suggested network through Mininet and Shadow simulations. The obtained results confirm that the performance of the proposed ITor-SDN network achieves the integrity of anonymity and continuity of traffic flow.

REFERENCES

[1] L. Basyoni, N. Fetais, A. Erbad, A. Mohamed, and M. Guizani, "Traffic analysis attacks on tor: A survey," in *Proc. IEEE Int. Conf. Informat., IoT, Enabling Technol. (ICIoT)*, Feb. 2020, pp. 183–188.

[2] F. A. Yaseen and H. S. Al-Raweshidy, "Smart virtualization packets forwarding during handover for beyond 5G networks," *IEEE Access*, vol. 7, pp. 65766–65780, 2019.

[3] M. Bernaschi, A. Celestini, M. Cianfriglia, S. Guarino, F. Lombardi, and E. Mastrostefano, "Onion under microscope: An in-depth analysis of the tor web," *World Wide Web*, vol. 25, no. 3, pp. 1287–1313, May 2022.

[4] L. Hellebrandt, I. Homoliak, K. Malinka, and P. Hanácek, "Increasing trust in tor node list using blockchain," in *Proc. IEEE Int. Conf. Blockchain Cryptocurrency (ICBC)*, May 2019, pp. 29–32.

[5] B.-S. P. Lin, "Toward an ai-enabled SDN-based 5G & IoT network," *Netw. Commun. Technol.*, vol. 5, no. 2, pp. 1–7, 2021.

[6] Y. Yang, L. Yang, M. Yang, H. Yu, G. Zhu, Z. Chen, and L. Chen, "Dark web forum correlation analysis research," in *Proc. IEEE 8th Joint Int. Inf. Technol. Artif. Intell. Conf. (ITAIC)*, May 2019, pp. 1216–1220.

[7] L. Basyoni, A. Erbad, M. Alsabah, N. Fetais, A. Mohamed, and M. Guizani, "QuicTor: Enhancing tor for real-time communication using QUIC transport protocol," *IEEE Access*, vol. 9, pp. 28769–28784, 2021.

[8] H. D. Hoang, P. T. Duy, and V.-H. Pham, "A security-enhanced monitoring system for northbound interface in SDN using blockchain," in *Proc. 10th Int. Symp. Inf. Commun. Technol. (SoICT)*, 2019, pp. 197–204.

[9] Z. Ling, J. Luo, D. Xu, M. Yang, and X. Fu, "Novel and practical SDN-based traceback technique for malicious traffic over anonymous networks," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, Apr. 2019, pp. 1180–1188.

[10] K. Cabaj and W. Mazurczyk, "Using software-defined networking for ransomware mitigation: The case of CryptoWall," *IEEE Netw.*, vol. 30, no. 6, pp. 14–20, Nov. 2016.

[11] O. Ahmed and M. H. MohdYusof, "Architecture based on tor network for securing the communication of northbound interface in sdn," *Compusoft*, vol. 9, no. 7, pp. 3755–3761, 2020.

[12] A. Yazdinejad, R. M. Parizi, A. Dehghantanha, Q. Zhang, and K. R. Choo, "An energy-efficient SDN controller architecture for IoT networks with blockchain-based security," *IEEE Trans. Services Comput.*, vol. 13, no. 4, pp. 625–638, Jul. 2020.

[13] W. Li, Y. Wang, W. Meng, J. Li, and C. Su, "BlockCSDN: Towards blockchain-based collaborative intrusion detection in software defined networking," *IEICE Trans. Inf. Syst.*, vol. E105.D, no. 2, pp. 272–279, 2022.

[14] Z. Abou El Houda, A. S. Hafid, and L. Khoukhi, "Cochain-SC: An intra- and inter-domain ddos mitigation scheme based on blockchain using SDN and smart contract," *IEEE Access*, vol. 7, pp. 98893–98907, 2019.

[15] S. Guo, Y. Qi, P. Yu, S. Xu, and F. Qi, "When network operation meets blockchain: An artificial-intelligence-driven customization service for trusted virtual resources of IoT," *IEEE Netw.*, vol. 34, no. 5, pp. 46–53, Sep. 2020.

[16] J. Weng, J. Weng, J. Zhang, M. Li, Y. Zhang, and W. Luo, "DeepChain: Auditable and privacy-preserving deep learning with blockchain-based incentive," *IEEE Trans. Depend. Secure Comput.*, vol. 18, no. 5, pp. 2438–2455, Sep. 2021.

[17] M. A. Ferrag, M. Derdour, M. Mukherjee, A. Derhab, L. Maglaras, and H. Janicke, "Blockchain technologies for the Internet of Things: Research issues and challenges," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2188–2204, Apr. 2019.

[18] J. Luo, Q. Chen, F. R. Yu, and L. Tang, "Blockchain-enabled software-defined industrial Internet of Things with deep reinforcement learning," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 5466–5480, Jun. 2020.

[19] F. A. Yaseen, N. A. Alkhalidi, and H. S. Al-Raweshidy, "SHE networks: Security, health, and emergency networks traffic priority management based on ML and SDN," *IEEE Access*, vol. 10, pp. 92249–92258, 2022.

[20] I. Karunanayake, N. Ahmed, R. Malaney, R. Islam, and S. K. Jha, "De-anonymisation attacks on tor: A survey," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 4, pp. 2324–2350, 4th Quart., 2021.

[21] V. V. Lapshichyov, "TLS certificates of the tor network and their distinctive features," *Int. J. Syst. Softw. Secur. Protection*, vol. 10, no. 2, pp. 20–43, Jul. 2019.

[22] R. Gupta, S. Tanwar, and N. Kumar, "B-IoMV: Blockchain-based onion routing protocol for D2D communication in an IoMV environment beyond 5G," *Veh. Commun.*, vol. 33, Jan. 2022, Art. no. 100401.

[23] R. Jansen, J. Tracey, and I. Goldberg, "Once is never enough: Foundations for sound statistical inference in tor network experimentation," in *Proc. USENIX Secur. Symp.*, 2021.

[24] D. Adrian, K. Bhargavan, Z. Durumeric, P. Gaudry, M. Green, J. A. Halderman, N. Heninger, D. Springall, E. Thomé, L. Valenta, B. VanderSloot, E. Wustrow, S. Zanella-Béguelin, and P. Zimmermann, "Imperfect forward secrecy: How Diffie–Hellman fails in practice," in *Proc. 22nd ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2015, pp. 5–17.

[25] F. Bannour, S. Dumbrava, and D. Lu, "A flexible GraphQL northbound API for intent-based SDN applications," in *Proc. NOMS IEEE/IFIP Netw. Oper. Manage. Symp.*, Apr. 2022, pp. 1–5.

[26] R. K. Das, N. Ahmed, A. K. Maji, and G. Saha, "Nx-IoT: Improvement of conventional IoT framework by incorporating SDN infrastructure," *IEEE Internet Things J.*, vol. 10, no. 3, pp. 2473–2482, Feb. 2023.

[27] V. Merlo, G. Pio, F. Giusto, and M. Bilancia, "On the exploitation of the blockchain technology in the healthcare sector: A systematic review," *Expert Syst. Appl.*, vol. 213, Mar. 2023, Art. no. 118897.

[28] M. Ebrahim, A. Hafid, and E. Elie, "Blockchain as privacy and security solution for smart environments: A survey," 2022, *arXiv:2203.08901*.

[29] T. Alladi, V. Chamola, N. Sahu, and M. Guizani, "Applications of blockchain in unmanned aerial vehicles: A review," *Veh. Commun.*, vol. 23, Jun. 2020, Art. no. 100249.

[30] N. Burkart and M. F. Huber, "A survey on the explainability of supervised machine learning," *J. Artif. Intell. Res.*, vol. 70, pp. 245–317, Jan. 2021.

[31] K. Salah, M. H. U. Rehman, N. Nizamuddin, and A. Al-Fuqaha, "Blockchain for AI: Review and open research challenges," *IEEE Access*, vol. 7, pp. 10127–10149, 2019.

[32] R. Amin, E. Rojas, A. Aqdus, S. Ramzan, D. Casillas-Perez, and J. M. Arco, "A survey on machine learning techniques for routing optimization in SDN," *IEEE Access*, vol. 9, pp. 104582–104611, 2021.
[33] S. Choudhary and S. Dorle, "A quality of service-aware high-security architecture design for software-defined network powered vehicular ad-hoc networks using machine learning-based blockchain routing," *Concurrency Comput., Pract. Exper.*, vol. 34, no. 17, Aug. 2022, Art. no. e6993.
[34] L. Øverlier and P. Syverson, "Improving efficiency and simplicity of tor circuit establishment and hidden services," in *Proc. Int. Workshop Privacy Enhancing Technol.*, Jun. 2007, pp. 134–152.
[35] D. Migault and T. Guggemos, *Minimal IP Encapsulating Security Payload (ESP)*, document RFC 9333, 2023.
[36] J. Miguel-Alonso, "A research review of OpenFlow for datacenter networking," *IEEE Access*, vol. 11, pp. 770–786, 2023.
[37] N. Alkhalidi and F. Yaseen, "FDPHI: Fast deep packet header inspection for data traffic classification and management," *Int. J. Intell. Eng. Syst.*, vol. 14, no. 4, pp. 373–383, Aug. 2021.
[38] *Standard, Open Networking Foundation. 2015*, document Version 1.5. 1, O. S. Specification, 2017.
[39] N. Gupta, M. S. Maashi, S. Tanwar, S. Badotra, M. Aljebreen, and S. Bharany, "A comparative study of software defined networking controllers using mininet," *Electronics*, vol. 11, no. 17, p. 2715, Aug. 2022.
[40] R. Jansen, J. Newsome, and R. Wails, "Co-opting Linux processes for $high − performance$ network simulation," in *Proc. USENIX Annu. Tech. Conf. (USENIX ATC)*, 2022, pp. 327–350.

FOUAD A. YASEEN received the B.Sc. degree in electronics and communication engineering from the University of Technology, Baghdad, Iraq, in 1994, the M.Sc. degree in electronics and communication engineering from the University of Baghdad, Baghdad, in 2010, and the Ph.D. degree from Brunel University London, U.K., in 2019. He is currently a member of the Iraqi Engineers Association, and he is also an Instructor Member of the CISCO Academy, University of Baghdad/Computer Center. He has authored or coauthored 12 papers in international journals and refereed conferences. His research interests include computer networks, wireless communication networks, mobile communication systems, SDN and cloud networks, and artificial intelligence. Besides designing and implementing industrial electronic circuits. He was the Head of many teams for designing and installing computer networks for the Iraqi Ministry of High Education and Scientific Research, and also the Iraqi Ministry of Communications/The High Institution for Communications.

NAHLAH ABDULRAHMAN ALKHALIDI received the B.Sc. degree in control and computer engineering from the University of Technology, Baghdad, Iraq, in 1993, and the M.Sc. degree in control and computer engineering from the University of Baghdad, Baghdad, in 2013. She is currently an Instructor Member of the CISCO Academy, University of Baghdad/College of Science. She is a member of the Iraqi Engineers Association. She has authored or coauthored eight papers in international journals and refereed conferences. Her research interests include computer networks, wireless communication networks, fiber optics communication systems, SDN and cloud networks, and artificial intelligence.

HAMED S. AL-RAWESHIDY (Senior Member, IEEE) received the Ph.D. degree from Strathclyde University, Glasgow, U.K., in 1991. He is currently a Professor in communications engineering. He was with the Space and Astronomy Research Centre, Iraq, PerkinElmer, USA, Carl Zeiss, Germany, British Telecom, U.K., Oxford University, Manchester Metropolitan University, and Kent University. He is also the Group Leader of the Wireless Networks and Communications Group (WNCG) and the Director of PG studies (EEE) with Brunel University, London, U.K. He is the Co-Director of the Intelligent Digital Economy and Society (IDEAS); the new research center which is a part of the Institute of Digital Futures (IDF). He is an Editor of the first book in *Radio over Fiber Technologies for Mobile Communications Networks*. He acts as a Consultant and involved in projects with several companies and operators, such as Vodafone, U.K.; Ericsson, Sweden; Andrew, USA; NEC, Japan; Nokia, Finland; Siemens, Germany; Franc Telecom, France; Thales, U.K. and France; and Tekmar, Italy, Three, Samsung and Viavi Solutions—actualizing several projects and publications with them. He is a Principal Investigator for several EPSRC projects and European Project, such as MAGNET EU Project (IP) 2004-2008. He has published more than 450 journals and conference papers and his current research interests include 6G with AI and the IoT. He is also an External Examiner for the Beijing University for Posts and Telecommunications (BUPT)—Queen Mary University of London. Further, he was an External Examiner for a number of the M.Sc. communications courses with Kings College London, from 2011 to 2016. He has also contributed to several white papers. Specifically, he was an Editor of *Communication and Networking* (White Paper), which has been utilized by the EU Commission for research. He has been invited to give presentations at the EU workshop and delivered two presentations at Networld2020, and being the Brunel Representative for NetWorld2020 and WWRF (for the last 15 years).