Research Article

Yelkal Mulualem Walle*

Hybrid RSA–AES-Based Software-Defined Network to Improve the Security of MANET

https://doi.org/10.1515/opis-2024-0001
received November 05, 2023; accepted February 27, 2024

Abstract: Software-defined networking offers a flexible and programmatically efficient network design. Security in today's ad hoc mobile wireless network is paramount and incredibly challenging. Software-defined network is used to automatically and dynamically manage and control large network devices, network services, traffic paths, network topology, and packet management (quality of service). Recently different attackers are attacking our data when forwarding from one device to another. Therefore, software-defined networking and a Hybrid Rivest, Shamir, and Adelman (RSA)–Advanced Encryption Standard (AES) cryptography algorithm are needed to establish the concept of software-defined networking in mobile ad hoc networks to improve security and routing efficiency. The proposed Hybrid Cryptography Algorithm (HCA)-Based SDN mainly creates strong detection, prevention, and authentication mechanisms for MANET. The proposed secure data channel throughput increased by 0.4%, and the suggested system latency was 3.6% lower than the Normal MANET. It is already proved that the Hybrid cryptography algorithm also generates a key for security faster than RSA (Rivest, Shamir, and Adelman). The performance of the RSA–AES (hybrid) approach for encrypting and decrypting broad data significantly beats the RSA-Blowfish algorithm. In decrypting files, the hybrid approach (RSA–AES) outperforms the RSA-Blowfish method 11.2 times more efficiently when the file size is 32 kB; however, efficiency is increased by 77.1 times when the file size exceeds 4,096 kB. The experimental result shows that as the file size increases the hybrid RSA–AES solution outperforms RSA when the file is only 145 bytes; however, when the file is 6,460 bytes in size, the efficiency is multiplied by 61.3. As file size increases, RSA is less efficient than the hybrid encryption method. This is more preferred to be implemented for different parts of wireless networks like MANET.

Keyword: software-defined network, MANET, RSA, hybrid cryptography algorithm

1 Introduction

In recent years, several computer technologies such as big data, cloud computing, the Internet of Things, and other wireless technologies have grown in popularity. New security threats and vulnerabilities have occurred as a result of the rapid expansion of IT infrastructure and the introduction of new technologies. To accommodate this ever-growing information infrastructure, we use one technology called software-defined network, notably in wireless technologies (SDN) (Mostafaei & Menth, 2018; Rawat & Reddy, 2016; Stancu, Halunga, Suciu, & Vulpe, 2015). SDN has several advantages over traditional networks, including the ability to quickly install and test new ideas, as well as the ability to promote innovative network design (Adere & Murthy, 2010).

It is also a security solution for many security attacks because it implements a security framework through an extensible and programmable Software Design Network together with security applications, resulting in a flexible and effective security protection mechanism (Nishide, Kubo, Shinkuma, & Takahashi, 2012; Stancu et al., 2015).

* **Corresponding author: Yelkal Mulualem Walle,** Department of Information Technology, College of Informatics, University of Gondar, Gondar, Ethiopia, e-mail: yelkalmulualem@gmail.com

Open Access. © 2024 the author(s), published by De Gruyter. This work is licensed under the Creative Commons Attribution 4.0 International License.

Ad hoc network research and development have gained traction due to their wide range of applications. The Mobile Ad Hoc Network (MANET) is a type of wireless network that has attracted the attention of businesses and universities. This network is vulnerable to a variety of external and internal security risks due to its nature (Agarwal & Sejwar, 2015; Modieginyane, Letswamotse, Malekian, & Abu-Mahfouz, 2018; Rubinstein, Moraes, Campista, Costa, & Duarte, 2019). Traditional business models are impacted by the explosive proliferation of multimedia content, the rise of cloud computing, the impact of increased use of mobile devices, and the current economic pressures to reduce costs while keeping revenues stable. SDN's ability to manage the entire network from a single device, as well as its ability to interface with MANET and respond to dynamic changes in the network topology (Rawat & Reddy, 2016), pushes us to learn more. Software Design Network has been limited to a wired network for the past decade. It can be used in both wired and wireless networks, according to a new study. The benefits of SDN and how it can be used in a new wireless network allow for topology changes while increasing network security and performance is required.

Wireless networks, especially networks without infrastructures such as MANETs and cellular networks, are not well served by the current network architecture. Firewalls, antivirus software, security rules, physical security, and other measures that disallow wireless networks in dynamic network topology are among the most common forms of protection mechanisms in current network design. Recently, side-channel attacks have put the operations and implementations of well-known algorithms like Advanced Encryption Standard (AES) and Rivest, Shamir, and Adelman (RSA) at risk (Canto, Kaur, Kermani, & Azarderakhsh, 2023; Mozaffari Kermani, Azarderakhsh, & Mirakhorli, 2016; Kaur, Canto, Kermani, & Azarderakhsh, 2023). Several previous studies used a variety of ways to improve the security of wireless networks. One approach to achieving secure routing in a MANET is to use Software Design Network in conjunction with a cryptographic algorithm. ID-based encryption is a means used with SDN to provide MANET (IBC) security (Sarbhukan & Ragha, 2018; Subasree & Radha, 2014). The ability to secure end-to-end MANET is a key feature of pairing ID-Based Cryptography (IBC) with an SDN controller. Users can generate their public key in IBC using publicly available information such as their email address, telephone number, and IP address. The fact that IBC has built-in key escrow property is its most notable drawback. As a result of the research, the use of IBC for wireless network security in SDN has resulted in several security problems. Identity exposure, key revocation difficulties, key escrow, key management issues, and secure routing are just some of the obstacles to security (Prasad & Ali, 2017; Spooner & Zhu, 2016).

The degree of trust in the private key generator determines whether or not the IBC can be used for security purposes in any system (PGK). Because the PKG owns all the private keys, this security mechanism violates the non-repudiation security service, which requires a higher level of assurance and availability than a certificate authority. Users must have access to the PKG at all times to communicate with their private keys, which makes them more vulnerable to attack. Another major flaw in IBC's security was the lack of revocation of keys. When a user's private key is lost, he cannot access his private key. Consequently, this requires much more attention than just the security level, and a strong secure line of key distribution must be created between the PKG and each user. Therefore, integrating a Hybrid cryptography algorithm with a software-defined network to improve the security and routing efficiency of MANET is important.

Hence, this research work attempted to answer the following research questions:

- How does software-defined networking improve the routing efficiency of the ad hoc mobile network?
- How many Hybrid cryptography algorithms be used to enable safe routing among mobile nodes inside a MANET with the help of SDN?
- How can the suggested security algorithm's strength against security assaults be measured in comparison to existing security algorithms?

The rest of the article is organized as follows. Section 2 explains related works that give an overview of a few recent proposed security algorithms and the drawbacks of those methods. The proposed model the SDN with a hybrid Cryptography algorithm (RSA–AES) is explained in detail in Section 3. The result and discussion of the proposed method is explained in Section 4. Finally, the conclusions of the present algorithm are discussed in Section 5.

2 Related Work

A quick grasp of MANET security design requirements and cryptography solutions, with a focus on security schemas and case studies of cryptography approaches applied to ad hoc networks, are surveyed. Different security techniques to protect our data from potential attackers like symmetric and asymmetric cryptography techniques are described. They select the RSA cryptography algorithm for implementation. Thearticle's flaw is that it does not use an alternative routing efficiency mechanism to exchange data between nodes (Bulla, 2021).

RSA technique to construct secure communication in Manet is used to improve the security of each node like confidentiality and privacy of each node when communication happens between each other from different attackers by using the RSA cryptography algorithm based on Diffie–Hellman Key. The drawback of this investigation is it is only focused on image transmission (Ankush & Gupta, 2015; Ezekiel, Ajibola, & Ebelogu, 2019; Singh, 2013).

An architecture of a Software-defined network for mobile ad hoc network (SDN_MANET) focused on (1) learning route to Software-Defined Networking Controller (SDNC), (2) learning network topology, and (3) sending network routes is introduced (Dusia, 2019). SDN architecture to make it suitable for infrastructureless networks, especially for MANET implemented. This study has certain drawbacks. Any centralized architecture in the system has a single point of failure.

The data transmission and secure routing in MANET protect our data when forwarding data like images, video, audio, and other files from one device to another device by using security parameters implemented. Performance investigation with the AODV protocol and TCP was also conducted to back up their assertion. However, they only used symmetric cryptography to secure the route between each device (Alnumay & Ghosh, 2014).

To improve the security of MANET with the help of SDN to protect different data formats from different attackers like the man-in-the-middle attack, Gray hole attack, black hole assault, sinkhole attack, and other attacks used when passing messages from one node to another is proposed (Demissie, 2020). But ECC algorithm does not work well with multipath routing in a mobile ad hoc network.

For software-defined networks, a cluster-based MANET routing system is developed. A routing protocol called SDN that uses a clustered-based routing protocol to increase routing efficiency and lengthen MANET's life by choosing the optimum path to the target node with the least amount of energy consumption. Therefore, the drawback of this article is that it necessitates the use of a constant central software-defined network controller (Adere & Murthy, 2010; Kadhim, Hosseini Seno, & Shihab, 2018).

3 Proposed Model

3.1 Overview of the Proposed System

The proposed system is an SDN-based MANET scenario that improves routing efficiency and security. Secure routing in MANET is achieved using a unique cryptographic method. The foundation of internet security as we know it today is public key cryptography, which allows two devices (people) to communicate securely without first having to share key confidential information.

Each of the nodes in the proposed work is likely designed to be able to function as a router or final host in networks without any supporting infrastructure. Applications connected to the SDNC using the interfaces must have rules for the SDNC to select network paths and make decisions about security and many other services. We enhance security algorithms based on the concept of a Hybrid cryptography algorithm to use SDNC to protect the data flow between mobile nodes. The reactive routing strategies and how SDN improves routing efficiency in ad hoc mobile networks are looked in detail.

We interface the SDN controller with the ad hoc mobile network using the SDNC open-flow protocol. To create secure end-to-end routing between mobile nodes, we use the principles of the hybrid algorithm, a

cryptographic method. It is possible to build and validate services, as well as encrypt and decrypt data using a limited range of keys. Hybrid encryption is a better choice for devices such as MANET, Vehicular Ad hoc Network (VANET), and other wireless devices used in networks with limited resources, power, and energy (Jaballah, Conti, & Lal, 2019; Saleh & Hosoon, 2019). Because of these properties, the proposed security solution was then implemented using Python with the PyCharm editor, improving the security of ad hoc mobile networks. It is anticipated that network nodes can possess one or more wireless network adapters (WNA). Both in-band control and data communications can be conducted over a single WNA. A flow rule entry is added to the FT to route the control packets to the local controller to distinguish between control packets and data packets.

When connecting to a node, the SDNC frequently uses multiple hops, and no location service is used to maintain track of the nodes' positions. Then, a network mobile node houses the SDN controller. Here, we give an illustration of a node sending a secure data transfer, as shown in Figure 1. Device-A and Device-B from the MANET are intended to have a secure communication channel, and SDN is required to set up this secure line of communication between these two nodes. Even though the SDN controller is aware of the presence of user nodes, there is a substantial difference between controlling and forwarding activities in this case.

The controller can be configured to perform various tasks like sharing resources. The common protocol for implementing this type of architecture with a controller is open flow. RSA-Advanced Encryption Standard (AES; hybrid cryptography) can be used to achieve secure communications in a MANET with an SDN controller is demonstrated (Abdullah, 2017).

However, the network topology can be learned, and network routes can be sent reactively. Each FD in the typical SDN architecture has an agent that installs FT routes and communicates with SDNC via the OpenFlow protocol (Mishra, Dusia, & Sethi, 2018). Each FD in the proposed SDMN architecture has a Local Controller (LC) that performs comparable tasks, but with one important distinction: The OpenFlow protocol is not used in the software-defined network controller (SDNC)-to-LC communication because the size of its messages may be too high for a MANET environment with limited bandwidth. To work with the FT, the LC transforms the routing data sent by SDNC in OpenFlow messages. This is different from the standard SDN architecture. The conversion of the routing information into OpenFlow messages allows the MANET-based SDN architecture to leverage all the benefits of SDN.

3.2 The Proposed Hybrid Cryptography Algorithm (RSA–AES)

The latest hybrid cryptography technique takes advantage of the fundamental arithmetic operations of the RSA and AES algorithms. These two methods each work separately to increase MANET's security when data is transmitted from one device to another. This hybrid strategy, based on the execution of algorithms as well as the theory and practice of encryption and decryption, uses the RSA and AES algorithms. This investigation uses both the stability and key management advantages of the RSA approach and the speed advantages of the AES method to encrypt and decrypt.

The hybrid technique method focused on encrypting and decrypting data and any other file format by using a hybrid algorithm. The AES algorithm uses the same public key and private key for both encryption and decryption. The RSA algorithm uses two different keys for both encryption and decryption.

3.2.1 Proposed Hybrid Cryptography Algorithm (RSA–AES) With SDN

The SDN controller uses a hybrid cryptography algorithm to secure routing in a MANET. Before sending a signed message from the sender to the recipient, both parties must agree on the hybrid algorithm. For example, before transmitting the message to **Device-B**, **Device-A** signs it with its private key to authenticate the conversation. Receiving messages from the source code requires signing messages and verifying messages. These procedures are covered in this section.

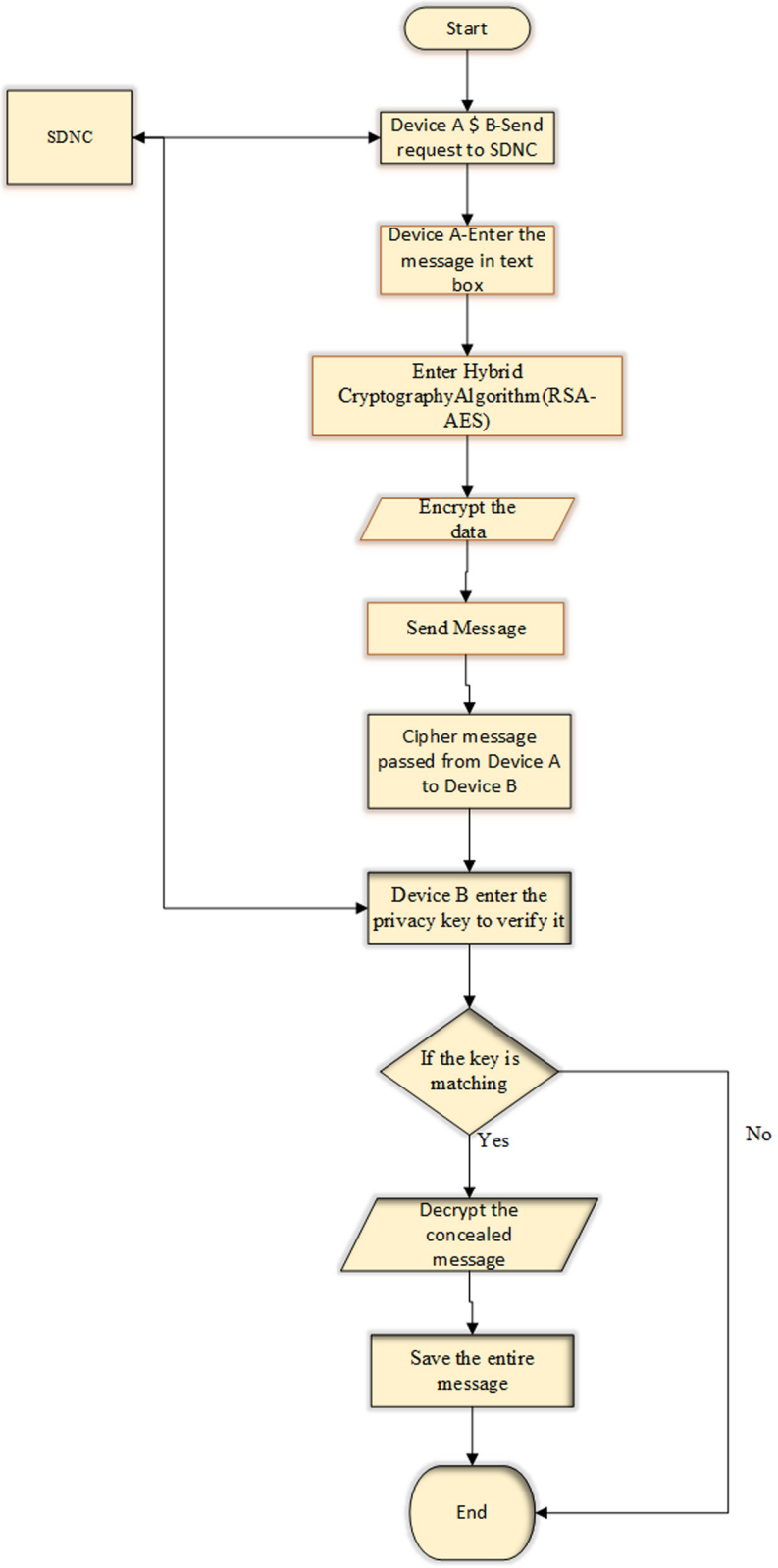

Figure 1: Proposed Framework of MANET-based SDN.

The hybrid technique guarantees the legitimacy of the message and establishes the identity of the sender. However, this does not minimize the importance of the message. Any user on the network can access the message. The post can be edited, but it's hard to tell if anyone has seen it before. Digitally signed documents may contain private information that should not be disclosed to anyone. In this situation, data encryption is essential. Although symmetric encryption is not as secure, creating asymmetric keys takes longer. This gives me the confidence to combine symmetric and asymmetric algorithms. Consider **device A** (the sender) and **device B**, the two users (the recipient). There is a shared secret key that is known only to the sender and the recipient. Each device sends requests to SDNC to obtain its private key and public key. Then, SDNC releases their private key and public key for each device.

Hybrid Implementation Algorithm

```
def main ():
#To encrypt a message addressed to Device-B in a hybrid crypto-system, Device-A does the following:
        Print ("…….. ")
        Print ("How Hybrid cryptography work?.")
        Print ("We'll encrypt and decrypt a message using RSA and AES. ")
```

3.2.1.1 Device-A (Sender)

a. Generate both public and private keys for both AES and RSA cryptography algorithms by Software-defined network controller. RSA private and public keys are used to encrypt and decrypt the AES cryptography algorithm before encrypting and decrypting data by AES public and private keys.
b. Get plain text messages from a user.
c. Encrypt the message using the AES cryptography Algorithm.
d. Next, we encrypt the AES symmetric key with RSA public key and display Cipher text as output.
e. Next, we save the cipher text, Private Key, and public key so that it can be decrypted.
f. At the end, the decrypted message is sent to the Receiver (Device-B)

Encryption Algorithm

```
#To encrypt a message addressed to Device-B in a hybrid crypto-system, Device-A does the following:
#1. Obtains Device-B public key.
Print ("Genering RSA public and private keys ……")
pub,pri = Key Generation ()
#2. Generates a fresh symmetric key for the data encapsulation scheme.
Print ("Genering AES symmetric key.")
key = secrets.token_hex(16)
        print ("AES Symmetric Key: ")
        print(key)
        KeyAES = key.encode ('utf-8')
#3. Encrypts the message under the data encapsulation scheme, using the symmetric key just generated:
        plainText = input
        ("Enter the message:")
        cipherAESe = AES.new(KeyAES, AES.MODE_GCM) nonce = cipherAESe.nonce
        Print ("Encrypting the message with AES ……")
        cipherText = encryptAES(cipherAESe,plainText)
print ("AES cypher text: ")
print (cipherText)
#4. Encrypt the symmetric key under the key encapsulation scheme, using Device-B public key:
        cipherKey = encrypt (pub, key)
```

```
Print (“Encrypting the AES symmetric key with RSA ……”)
print (“Encrypted AES symmetric key”)
        print (cipherKey)
#5 ….Send both of these encryptions to Device-B.
        #Sending…….....
```

3.2.1.2 Device-B (Receiver)

The recipient is in possession of the encoded communication. Additionally, a private copy of the sender’s public key is sent to the receiver. Steps were taken by the recipient to decrypt the communication: -

a. To decrypt cipher text we must first decrypt the AES symmetric key using RSA private key.
b. Once we have the decrypted symmetric key, we can then use this to decrypt the cipher text.

Decryption Algorithm

```
# To decrypt this hybrid cipher-text, Device-B does the following:
#1. Uses her private key to decrypt the symmetric key contained in the key encapsulation
segment.decriptedKey =”.join(decrypt(pri,cipherKey))
Print (“Decrypting the AES Symmetric Key. With RSA private key••”)
Print(“AES Symmetric Key:”)
Print (decriptedKey)
#2. Uses this symmetric key to decrypt the message contained in the data encapsulation
segment.decriptedKey = decriptedKey. Encode(‘utf-8’)
cipherAESd = AES.new(decriptedKey,AES.MOOE_GCH, nonce = nonce)
decrypted = decryptAES (cipherAESd, cipherText)
print(“Decrypting the message using the AES symmetric key•••••”)
print(“decrypted message: ”)
print(decrypted)
```

In general, to secure communications while allowing receiver and sender identity verification, a hybrid cryptography algorithm operates. Nothing can read the data because it is encrypted. The investigation’s outcome is the secure transport of data. Data integrity and non-repudiation are the conclusions of this inquiry. Even if there are several security dangers and issues in the network, the provided strategy would help to safely transfer the data and protect our data from numerous attackers like Man in the Middle, Sink Hole, Black Hole, and others.

4 Implementation and Result

The POX controller is already pre-installed on the Linux OS with the help of Python language. Using the POX controller, you can turn straightforward OpenFlow devices into hubs, switches, load balancers, and firewalls. Implementing OpenFlow/SDN experiments is made simpler by the truth. Depending on the real-world or hypothetical topologies, it can be changed to POX in a variety of ways, allowing you to conduct tests on the NS3 simulator. Furthermore, the OpenFlow compatibility of the switches facilitates the development of OpenFlow-based applications used in the SDN environment. It also provides an extendable Python API for creating the network. UDP packets are used for both data and control communication. An end-to-end secure routing for MANET-based SDN architecture and run on NS3 through API that implements security using Python.

4.1 Experiment and Result Analysis for Hybrid Cryptography Algorithm

The experiment files have a “txt” extension and range in size from 32 to 4,096 kB. Due to the file size of experience, memory file mapping is used to improve productivity, reduce disk demands, and allow frequent access to data stored in large files. In this experiment, the same batch of data was encrypted and decrypted using three different techniques. Because the method of combining data from multiple runs is used and the value with large error is removed, the experimental results are more reliable. RSA–AES and RSA-Blowfish encryption times are compared in light of the results. Among them, in various file sizes, the scheduling of the two is shown in Figure 2.

As shown in Figure 2, the RSA-Blowfish encryption methods encrypt data in approximately the same amount of time. The hybrid approach (RSA–AES) outperforms the RSA-Blowfish method 8.9 times more efficiently when the file size is 32 kB; however, efficiency is increased by 75.5 times when the file size is 4,096.

The RSA-Blowfish algorithms take longer to decrypt files as their sizes increase, according to experimental investigations shown in Figure 3. The hybrid approach (RSA–AES) outperforms the RSA-Blowfish method 11.2 times more efficiently when the file size is 32 kB, however, efficiency is increased by 77.1 times when the file size exceeds 4,096 kB. The performance of the RSA–AES (hybrid) approach for encrypting and decrypting broad data significantly beats the RSA-Blowfish algorithm, as shown by the comparison of encryption and decryption in Figures 2 and 3, respectively.

Plaintext size (KB)	RSA+AES Encryption Time	RSA+Blowfish Encryption Time	RSA+AES decryption Time	RSA+Blowfish decryption Time
32	120	3456123	140	1881211
64	126	3961234	131	2213234
128	127	4813452	159	2934334
256	133	5376246	143	3398716
512	138	6523544	173	4126456
1024	145	8124567	171	5234546
2048	146	8825251	199	5913234
4096	188	9934535	222	7142456

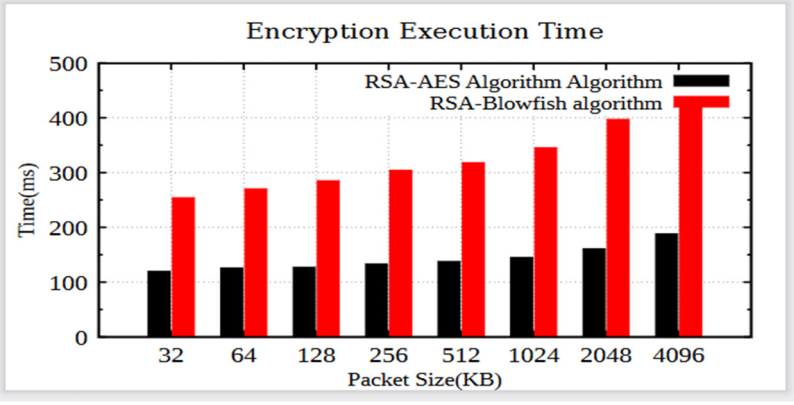

Figure 2: Compare RSA-Blowfish and RSA–AES with its encryption time.

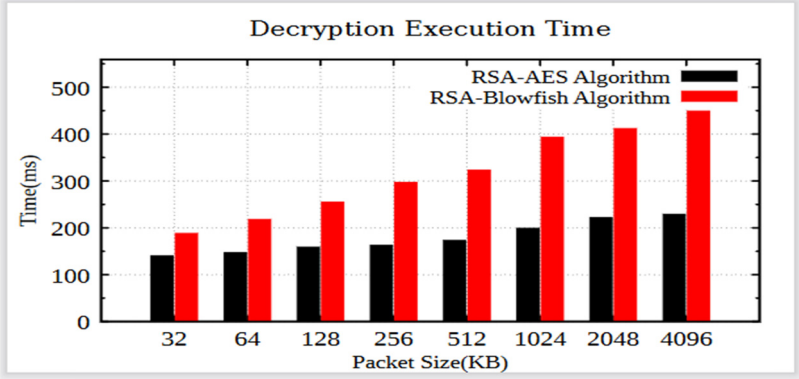

Figure 3: Compare RSA-Blowfish and RSA–AES with decryption time.

5 Result and Discussion

5.1 Measure Performance of Manet Based SDN

5.1.1 Small Networks

This study evaluates the effectiveness of the suggested strategy for networks with up to 60 nodes. The metrics end-to-end delay, packet delivery rate, and average throughput stated below are used to assess a network within this range.

Number of Node	PDR		AT		Ae2ed	
	Manet-SDN	Normal Manet	Manet-SDN	Normal Manet	Manet-SDN	Normal Manet
15	0.39	0.318	110.92	80	0.0041	0.0041
30	0.595	0.532	206.8	190	0.0061	0.0061
45	0.796	0.729	334.9	300	0.01	0.0089
60	0.99	0.909	448.89	393	0.0125	0.0101

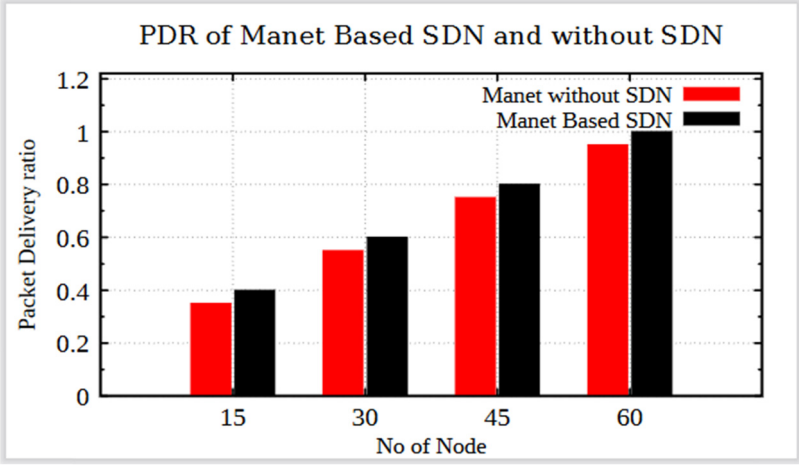

Figure 4: Performance measurement for a small network and Packet Deliver ratio of MANET-based SDN and MANET without SDN.

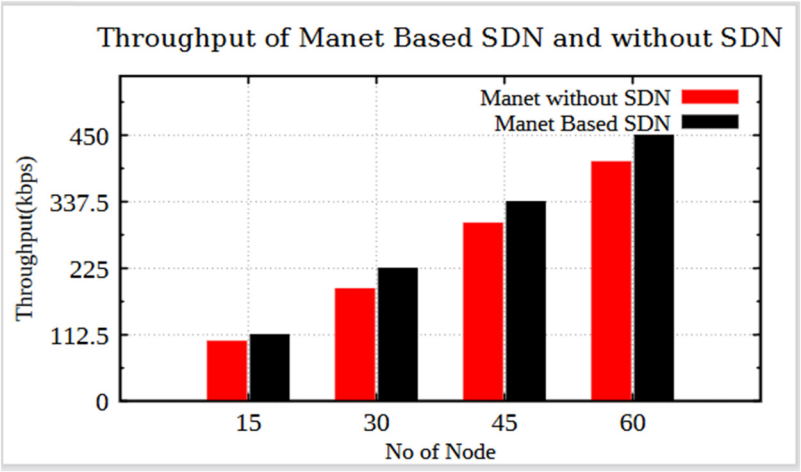

Figure 5: Throughput of Manet-Based SDN $ Manet without SDN.

Figure 4 illustrates how the MANET-Based SDN outcomes are superior to the selected method. A Hybrid Algorithm with MANET-based SDN outperforms MANET without SDN in networks with fluctuating node speeds. We infer that MANET-Based SDN is superior to MANET without SDN.

Analysis of Figure 5 shows that Manet with SDN outperforms Manet without SDN. Another feature that has become clear is that throughput is mostly unaffected by stop time and is solely governed by network density (Figure 6).

Ae2eD are lower for Manet-based SDN than in Normal Manet. The lower overhead is a result of the route update encapsulation technique used by the Manet-based SDN. On the other hand, Manet-based SDN incurs significant overhead due to the constant transfer of routing data.

In general, the simulation experiment shows that the average throughput and packet delivery ratios of MANET with SDN and MANET without SDN are nearly comparable. A MANET-based SDN has a shorter end-to-end delay than a Normal mobile ad hoc network. The MANET-based SDN controller reduces delays because it has a centralized global view of the network and can select shorter paths accordingly.

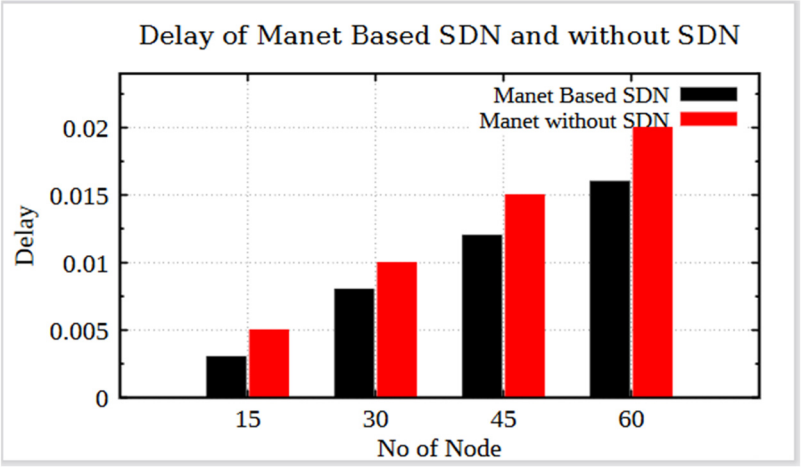

Figure 6: Delay of Manet Based SDN $ Manet without SDN.

5.1.2 Large Networks

Network sizes range from 50 to 200 nodes.

The difference between Manet with SDN and Manet without SDN increases for both PDR and AT as the network size approaches 200. However, compared to Manet without SDN, Ae2eD is even weaker with Manet-based SDN. However, as the total number of control packets in the network increased, the PDR and AT decreased in normal Manet.

The network performance of the suggested design was then evaluated and compared with the current architecture, considering the UDP throughput and latencies in the above (Figure 7). The target secure data channel throughput increased by 0.4% as shown in Figure 8. The suggested system latency was 3.6% lower than the Normal MANET. In this case, a hybrid approach was used to increase the speed of the network by adding an extra layer of encryption and decryption in addition to the selected secure data transport (Figure 9).

5.2 Measurement of Security Strength and Other Performance

The proposed hybrid system protects MANET-Based SDN from routing attacks such as spoofing attacks, wormhole attacks, flooding, and sinkhole attacks by applying powerful detection, protection, and authentication approaches.

Number of Node	PDR		AT		Ae2ed	
	Manet-SDN	Normal Manet	Manet-SDN	Normal Manet	Manet-SDN	Normal Manet
50	0.248	0.19	131.1	97.2	0.005	0.0041
100	0.493	0.43	262.7	213.04	0.0075	0.0063
150	0.746	0.661	391.6	322	0.01	0.0089
200	0.989	0.901	526.04	508.9	0.0.125	0.019

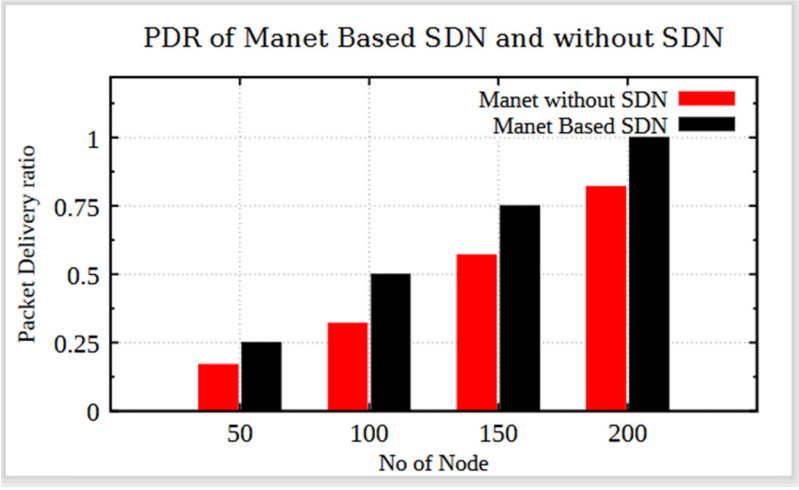

Figure 7: Packet delivery ratio of Manet-based SDN to Manet without SDN.

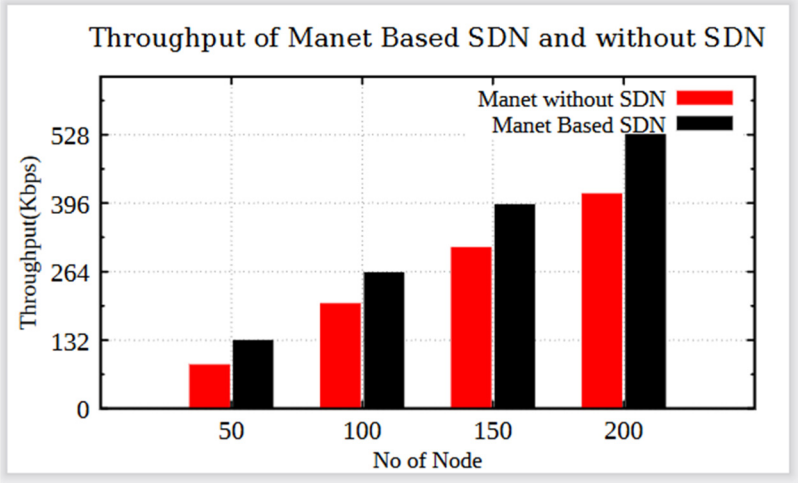

Figure 8: Throughput of Manet-based SDN and Manet without SDN.

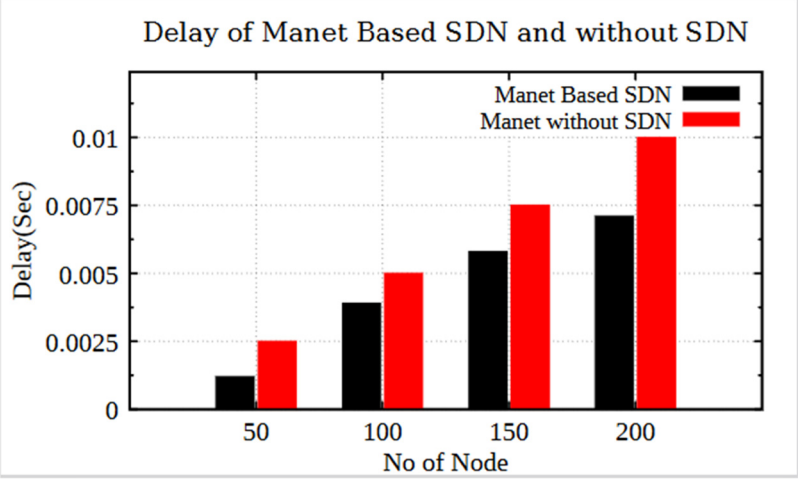

Figure 9: Delay of Manet-based SDN and Manet without SDN.

5.2.1 Computational Time

The performance of hybrid encryption-based SDN is evaluated in terms of time function and compared with RSA and AES in encrypting and decrypting data packets transmitted between mobile nodes in a network. This is supported by the comparison shown in Figure 10.

The experimental result in Figure 10 shows that as the file size increases, the execution time of the RSA algorithm regularly doubles, while the AES hybrid algorithm technique now encrypts data faster but is unstable and poses key management problems. The hybrid solution outperforms RSA when the file is only 145 bytes; however, when the file is 6,460 bytes in size, the efficiency is multiplied by 61.3. As file size increases, RSA is less efficient than the hybrid encryption method.

Experimental studies show that the speed of the RSA algorithm for decoding data depends on the file size. The growth of the RSA method is considerably more linear than that of the hybrid method. When dealing with huge amounts of data, better decryption performance than the RSA approach has a significant influence. When the file size reached 4,320 bytes, the decryption performance of the hybrid encryption algorithm increased by a factor of approximately 4.7, but at 46,500 bytes the efficiency increased by a ratio of 54.1.

Size(Bytes)	AES	RSA	Hybrid
145	32	180	49
1410	177	630	302.9
3230	230	1323	493.5
6460	940	2465	1401

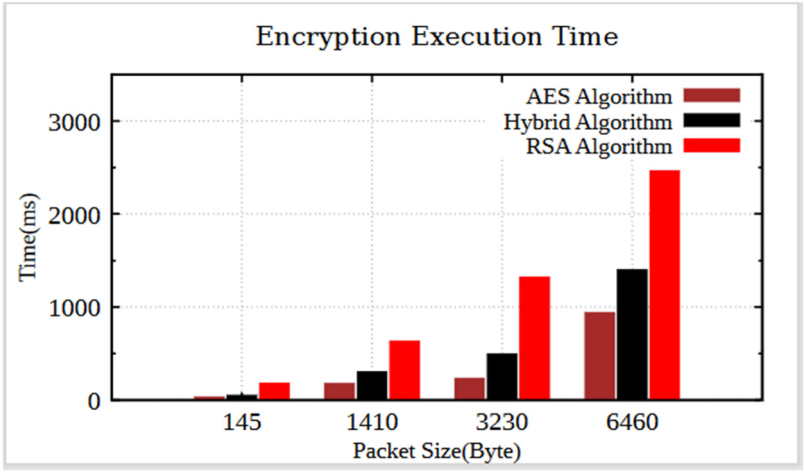

Figure 10: Encryption execution time.

Size(Bytes)	AES	RSA	Hybrid
4320	1419	1848	1630
12600	3381.3	4941	4112.3
20800	5128	8139	6921.5
46500	13842.9	17957	15931.7

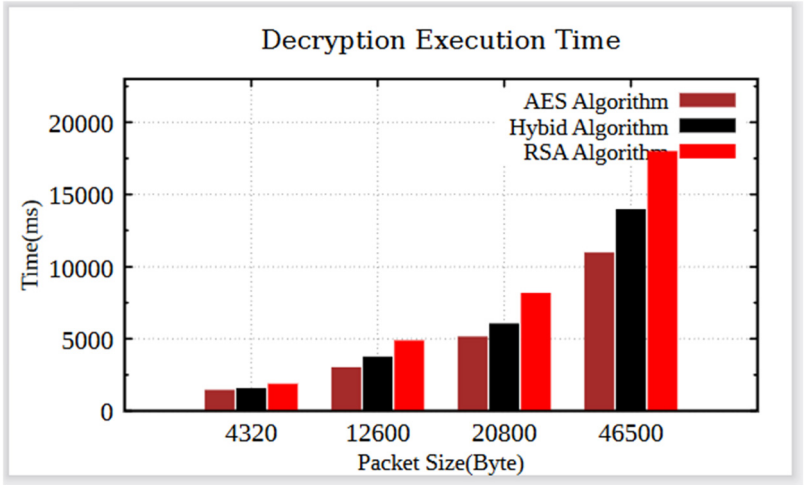

Figure 11: Decryption execution time.

Based on the encryption and decryption comparisons offered (Figures 10 and 11), the hybrid technique significantly outperforms RSA and AES in encryption efficiency when encrypting and decrypting large files. Double-layer encryption makes it more difficult to decrypt a file once it's open. The hybrid strategy solves the problem where security has been compromised due to the theft of the AES algorithm key.

5.2.2 Data Security

In this scenario, identical text files are encrypted and decrypted using two different techniques. The hybrid algorithm is the best choice when comparing the encryption and decryption times of RSA and hybrid algorithms because it provides the highest level of security and reliability for the data provided. The encryption-decryption times for the two with different file sizes are shown in Figure 12. The total processing time is shown in Figure 12 in seconds. After a comparison of the two procedures with text files of different sizes, the results presented in Figure 12 demonstrate that the hybrid strategy is superior to the RSA algorithm. The hybrid encryption algorithm offers the highest level of data protection compared to RSA techniques and is significantly faster than the RSA algorithms.

6 Discussion

Software-Defined network is one of the powerful computing technologies that provide flexibility and programmability to network infrastructure by separating the control plane from the data plane. Software-defined network Controller is the strategic point Of SDN. SDN improves the routing efficiency of Mobile ad hoc networks by discovering the transferred data to it instead of sending them to the Receiver one or the destination via neighbor in a Multihop manner with the help of SDNC. The SDN controller, which allows

Files size(MB)	RSA total time	Hybrid total time
1.19	0.64	0.535
3.57	0.97	0.62
7.14	1.52	0.8
10.7	2.07	0.88

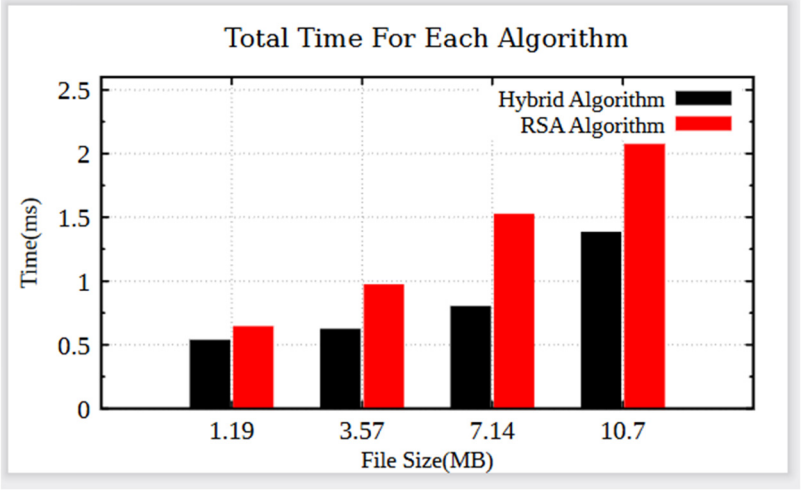

Figure 12: Total time for each algorithm in the second Based SDN.

devices to perform a variety of functions, makes networking efficient and keep a log of all prohibited operations and equipment.

A hybrid cryptography algorithm is one of the cryptography techniques that are used to protect our data from an attacker and any intruder by implementing a security mechanism to achieve end-to-end security of data when forwarding it from one device to another. We used Hybrid Cryptography algorithm (RSA–AES) to secure our data before transmitting it to the receiver (Destination) by encrypting the in-boxed data format may be audio, video, image, and another file format into text box using both RSA and AES public key at the next to that and generate the cipher message or encrypted data. At the end, the cipher message is forwarded to the destination.

We have used the three parameters to analyze the security of the proposed algorithm and an existing system like Packet delivery ratio, Average throughput, and average end-to-end delay to compare MANET-Based Software-defined network and Normal MANET. In this scenario, MANET-Based SDN outperforms Normal MANET, and due to Normal MANET Used Broadcast transmission during network simulation, retransmission of data and poor performance happen.

Additionally, the proposed algorithm and the existing system were also analyzed by Computation Time, data security, Memory requirement, and Interference by adding a DOS attack on each device for both Normal MANET and MANET-Based SDN to check the security strength of both algorithms. At the end of the proposed algorithm, the target secure data channel throughput increased and the suggested system latency was lower than the Normal MANET. Additionally, the emergence of quantum computers has made post-quantum cryptography, which aims to defend encrypted data based on classical encryption against quantum attacks, imperative. Existing cryptographic methods like RSA and ASE are at risk of these attacks (Cintas-Canto, Mozaffari-Kermani, Azarderakhsh, & Gaj, 2022; Canto et al., 2023; Elkhatib, Azarderakhsh, & Mozaffari-Kermani, 2021; Kermani & Azarderakhsh, 2016; Sarker, Kermani, & Azarderakhsh, 2022). Some studies indicate that lightweight cryptosystems are suitable for devices with limited capabilities to reduce computation power and improve cryptography performance for resource-constrained devices. It is essential to look into how lightweight RSA and AES algorithms might be used in a mathematical model to create a lightweight cryptographic scheme for different applications (Cintas-Canto, Kermani, & Azarderakhsh, 2022; Cintas-Canto, Kaur, Mozaffari-Kermani, & Azarderakhsh, 2023; Mozaffari-Kermani, Azarderakhsh, Ren, & Beuchat, 2016; Niasar, Azarderakhsh, & Kermani, 2020).

7 Conclusion and Recommendation

In recent years, the size and demand of the network have increased significantly. Modern wireless networks such as MANET, VANET, WSN, and others wireless networks face significant security challenges. Any network architecture based on a software-defined network provides a mechanism to more efficiently manage the network infrastructure and programmatically configure it while they are running. Users can create many logical network topologies on a single device and connect and modify them as needed using an SDN controller. SDN provides a complete overview of the entire network and its status. In terms of power consumption, network management, configuration, routing, mobility and location, and communication with all wireless networks, SDN offers a wide range of benefits. It is also used as a security solution to reduce an attacker.

The application of SDN on wireless networks what we call MANET is examined. The Software-defined network can help improve routing efficiency and security. However, many security attacks emerge when we use SDN for various reasons. Therefore, one way to enhance secure routing is to embed a cryptographic system in the SDNC. The proposed SDN controller based on a hybrid cryptography algorithm for secure routing for ad hoc mobile networks improves the security of MANET. This proposed approach provides a strong authentication mechanism to circumvent the disadvantages of IBC used with MANET-based SDN in legacy systems.

The use of different security algorithms with software-defined networking in providing security service left us additional work for a wireless network like MANET, VANET, and WSN since the use of SDN is in its initial stage. We tried to use a Hybrid cryptography algorithm (RSA–AES)-based SDN to establish secure

routing among mobile nodes. Applying and testing the proposed algorithm on WSN in future work will be recommended.

Funding information: This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflict of interest: The author states no conflict of interest.

References

Abdullah, A. M. (2017). Advanced encryption standard (AES) algorithm to encrypt and decrypt data. *Cryptography and Network Security*, *16*(1), 11.

Adere, K., & Murthy, G. R. (2010, September). Solving the hidden and exposed terminal problems using directional-antenna based MAC protocol for wireless sensor networks. In *2010 Seventh International Conference on Wireless and Optical Communications Networks-(WOCN)* (pp. 1–5). Colombo, Sri Lanka: IEEE.

Agarwal, K., & Sejwar, V. (2015). Avoidance of Hidden Terminal and Exposed terminal problem Using Directional MAC Protocol. *International Journal of Future Generation Communication and Networking*, *8*(4), 231–238.

Alnumay, W. S., & Ghosh, U. (2014). *Secure routing and data transmission in mobile ad hoc networks*. arXiv preprint arXiv:1402.2108.

Ankush, J., & Gupta, S. (2015). A secure communication using RSA cryptography in mobile ad-hoc networks. *International Journal for Rapid Research in Engineering Technology and Applied Science*, *1*(1), 1–5.

Bulla, S. (2021). A comprehensive survey on cryptography evaluation in mobile (MANETs). *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, *12*(2), 3406–3416.

Canto, A. C., Kaur, J., Kermani, M. M., & Azarderakhsh, R. (2023). *Algorithmic security is insufficient: A comprehensive survey on implementation attacks haunting post-quantum security*. arXiv preprint arXiv:2305.13544.

Cintas-Canto, A., Kaur, J., Mozaffari-Kermani, M., & Azarderakhsh, R. (2023). *ChatGPT vs Lightweight Security: First Work Implementing the NIST Cryptographic Standard ASCON*. arXiv preprint arXiv:2306.08178.

Cintas-Canto, A., Kermani, M. M., & Azarderakhsh, R. (2022). Reliable architectures for finite field multipliers using cyclic codes on FPGA utilized in classic and post-quantum cryptography. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, *31*(1), 157–161.

Cintas-Canto, A., Mozaffari-Kermani, M., Azarderakhsh, R., & Gaj, K. (2022, October). CRC-oriented error detection architectures of post-quantum cryptography niederreiter key generator on FPGA. In *2022 IEEE Nordic Circuits and Systems Conference (NorCAS)* (pp. 1–7). Oslo, Norway: IEEE.

Demissie, K. (2020). *Improving security of mobile ad hoc networks using elliptic curve cryptography based software defined networking (ECC_SDN)*. Debre Berhan University.

Dusia, A. (2019). *Software-defined architecture and routing solutions for mobile ad hoc networks*. University of Delaware. https://udspace.udel.edu/handle/19716/25641.

Elkhatib, R., Azarderakhsh, R., & Mozaffari-Kermani, M. (2021, June). Accelerated risc-v for sike. In *2021 IEEE 28th Symposium on Computer Arithmetic (ARITH)* (pp. 131–138). IEEE.

Ezekiel, B., Ajibola, A., & Ebelogu, C. (2019). Hybrid data encryption and decryption using RSA and RC4. *International Journal of Scientific & Engineering Research*, *10*(10), 156–165.

Jaballah, W. B., Conti, M., & Lal, C. (2019). Software-defined VANETs: Benefits, challenges, and future directions. *IEEE Commun*, 1–17. https://arxiv.org/pdf/1904.04577.pdf.

Kadhim, A., Hosseini Seno, S. A., & Shihab, R. A. (2018). Routing protocol for SDN-cluster based manet. *Journal of Theoretical and Applied Information Technology*, *96*, 5398–5412.

Kaur, J., Canto, A. C., Kermani, M. M., & Azarderakhsh, R. (2023). *A comprehensive survey on the implementations, attacks, and counter-measures of the current NIST lightweight cryptography standard*. arXiv preprint arXiv:2304.06222.

Kermani, M. M., & Azarderakhsh, R. (2016, December). Lightweight hardware architectures for fault diagnosis schemes of efficiently-maskable cryptographic substitution boxes. In *2016 IEEE International Conference on Electronics, Circuits and Systems (ICECS)* (pp. 764–767). Monte Carlo, Monaco: IEEE.

Mishra, V. K., Dusia, A., & Sethi, A. (2018). *Routing in software-defined mobile ad hoc networks (sd-manet)*. US Army Research Laboratory Aberdeen Proving Ground United States. https://apps.dtic.mil/sti/pdfs/AD1059388.pdf.

Modieginyane, K. M., Letswamotse, B. B., Malekian, R., & Abu-Mahfouz, A. M. (2018). Software defined wireless sensor networks application opportunities for efficient network management: A survey. *Computers & Electrical Engineering*, *66*, 274–287.

Mostafaei, H., & Menth, M. (2018). Software-defined wireless sensor networks: A survey. *Journal of Network and Computer Applications*, *119*, 42–56.

Mozaffari Kermani, M., Azarderakhsh, R., & Mirakhorli, M. (2016). Multidisciplinary approaches and challenges in integrating emerging medical devices security research and education. *American Society for Engineering Education 123rd Annual Conference & Exposition*, 1–15.

Mozaffari-Kermani, M., Azarderakhsh, R., Ren, K., & Beuchat, J. L. (2016). Guest Editorial: Introduction to the special section on emerging security trends for biomedical computations, devices, and infrastructures. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, *13*(3), 399–400.

Niasar, M. B., Azarderakhsh, R., & Kermani, M. M. (2020). Optimized architectures for elliptic curve cryptography over Curve448. *Cryptology ePrint Archive*, Paper 2020/1338, 1–19.

Nishide, K., Kubo, H., Shinkuma, R., & Takahashi, T. (2012). Detecting hidden and exposed terminal problems in densely deployed wireless networks. *IEEE Transactions on Wireless Communications*, *11*(11), 3841–3849.

Prasad, B. V., & Ali, S. S. (2017). Software–defined networking based secure rout-ing in mobile ad hoc network. *International Journal of Engineering & Technology*, *7*(1.2), 229.

Rawat, D. B., & Reddy, S. R. (2016). Software defined networking architecture, security and energy efficiency: A survey. *IEEE Communications Surveys and Tutorials*, *19*, 325–346. doi: 10.1109/COMST.2016.2618874.

Rubinstein, M. G., Moraes, I. M., Campista, M. E. M., Costa, L. H. M., & Duarte, O. C. M. (2019, August). A survey on wireless ad hoc networks. In *IFIP International Conference on Mobile and Wireless Communication Networks* (pp. 1–33). Boston, MA: Springer US.

Saleh, H. H., & Hosoon, S. T. (2019). A survey on VANETs: Challenges and solutions. *International Journal of Engineering & Technology*, *8*(4), 266–274.

Sarbhukan, V. V., & Ragha, L. (2018). A review: Various security mechanisms used in MANET. *IOSR Journal of Engineering (IOSRJEN)*, *11*, 6–10.

Sarker, A., Kermani, M. M., & Azarderakhsh, R. (2022). Efficient error detection architectures for postquantum signature falcon's sampler and KEM SABER. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, *30*(6), 794–802.

Singh, G. (2013). A study of encryption algorithms (RSA, DES, 3DES and AES) for information security. *International Journal of Computer Applications*, *67*(19), 33–38.

Spooner, J., & Zhu, S. Y. (2016). A review of solutions for SDN-exclusive security issues. *International Journal of Advanced Computer Science and Applications*, *7*(8), 113–122.

Stancu, A., Halunga, S., Suciu, G., & Vulpe, A. (2015, April). An overview study of software defined networking. In *2015 14th International Conference on Informatics in Economy (IE 2015), Bucharest* (pp. 50–55).

Subasree, S., & Radha, S. (2014). Enhanced security key management scheme for MANETS. *WSEAS Transactions on Communications*, *13*, 15–25.