Journal of Engineering Science *Vol. XXVII, no. 4 (2020), pp. 128 - 139*
Fascicle *Electronics and Computer Science* *ISSN 2587-3474*
Topic *Computers and Information Technology* *eISSN 2587-3482*

DOI: 10.5281/zenodo.4288305
CZU 004

LEVERAGING BLOCKCHAIN TECHNOLOGY TO ASSURE SECURITY OF SDN

Ali Ameen*, ORCID ID: 0000-0002-5451-8257

Technical University of Moldova, 168 Stefan cel Mare Av., MD-2004 Chisinau, Republic of Moldova
**alisalmanhussein@yahoo.com*

Received: 09. 14. 2020
Accepted: 11. 02. 2020

Abstract. This article presents a potential solution to secure or ensure a better security level for the Software-Defined Networks (SDN) paradigm; by introducing the usage of blockchain technology in a different way than the Marconi protocol technology proposes. Most of the techniques and methodologies proposed by this research are to patch some security issue the SDN presents like the single point of failure, cause as SDN can provide solutions and flexibility for current computer networks; it could also promote some new security threats since it is still a relatively new technology. In this article those algorithms and methodologies are incorporated together as a full suite or a framework that can be applied as a network application for the software-defined network environment and could be implemented in the management or application plane which is the top layer in the architecture of the SDN as we will see later in this article.

Keywords: *application programming interface, control plane, centralization, east-westbound API, smart contracts, secure hash algorithm SHA 256.*

1. Introduction

Software-defined networking is a new way of managing how computer networks behave, it gives more ability to manage the network nodes, facilitates policies enforcement, provides more speed in applying configurations on network nodes [1]. It has one point of controlling which is laying in the control plane represented by the controller which is the only brain for the network which provides flexibility and single point of administration in the same time this situation from a security point of view this could be a single point of failure. So, as restriction of the control in the network by the control plane is a good security feature, it could also open new security challenges and here comes our research secure software-defined networks in order to assure the security of computer networks in general [2]. We propose a whole suite of algorithms and methodologies incorporated with each other in one framework called the Hydra, to help securing the control plane of the SDN represented by the controller or multiple controllers; one of the ideas for proposed for our research is to use blockchain for securing the connection between multiple controllers, since that SDN best features is the centralization and it is also one of the main security challenges in case if the centralization point was jeopardized.

Blockchain is relatively a new technology that provides decentralization and could potentially affect every aspect of our lives in the future, it is fairly known for its biggest

participation in designing cryptocurrencies and implementing them in a relatively secure framework. In this article we try to show how it is nearly possible to implement and use blockchain-based technologies and methods to secure the east-west Application Programming Interface (API) connection which is the line of communication between SDN controllers; we can leverage hashing algorithms and store the data of configuration updates in blockchain-like sequence in order to protect the configuration updates from being tampered with by the attacker, in order to protect the topology of network from some famous attacks including Man In The Middle attack (MITM).

Architecture of Software-defined networks:

First we need to know a little about what structure SDN has, what issues does it have then, explain a bit about the proposed solutions to fix these bugs or problems. Software defined networking has a new way of managing the networks by mainly decoupling the control layer from the data layer that we working coherently in legacy network devices like switches where every switch has its own application specific integrated circuit ASIC brain and forwarding tables to direct and route network data packets so, SDN architecture will be divided into three main layers, the management or application plane, control plane and data plane; where management plane is represented by the network application that fulfills the required network policies by the system administrator where it provides a good level of abstraction.

After that comes the control plane represented by the controller which serves as the network brain which connects to the management or application layer via an application programming interface API and fulfills the commands and configurations specified by the application plane then, as mentioned before comes the data plane or forwarding loop or layer which is connected to the control layer via an SDN protocol mostly using OpenFlow protocol, this layer is simply a forwarding element or elements like a switch but stripped off of its brain which becomes like a simple dumb network node where its sole role is to forward network packets based on the flow rules specified by the controller. Figure 1 shows the general structure of software-defined networks.

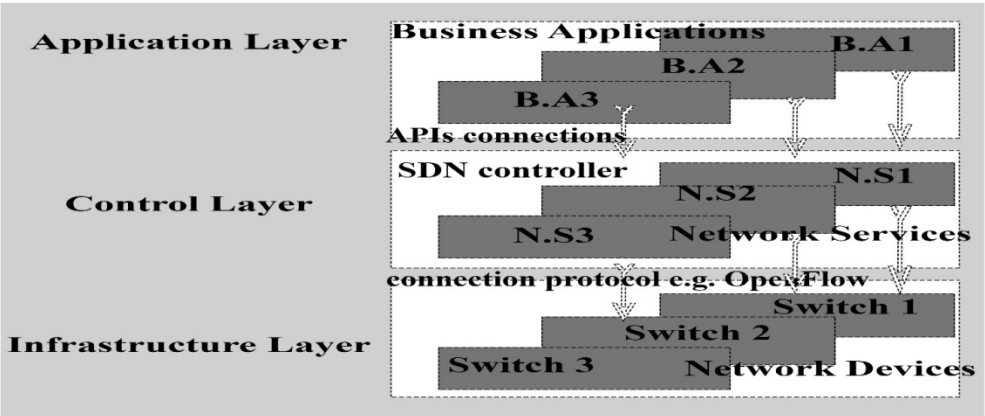

Figure 1. Structure of Software-Defined Networks.

Vulnerabilities of Software-Defined Networks:

Despite that software-defined networking is itself a solution for network cyber security issues and network management obstacles etc. but this new approach is a new

technology challenge itself and poses some new security threats. This article talks about a part of our research which focuses on two main weakness points which are:

- Centralization: now despite that centralization of SDN architecture is one of the main positive features of SDN and an advantage in SDN over the classical architecture in one hand but on the other hand it represents a potential threat itself in the same time by creating a single point of failure. Where we have a single controlling entity represented by the controller, whether it was software-based or hardware-based and as known the controller is responsible for managing the network so, any kind of attack or disruption that could stop the controller will have deep negative implications.
- East-westbound API: in case of using multiple controllers in the network topology to solve the previous problem, there could be a connection between those multiple controllers and that communication channel is called the east-westbound application programming interface API and to the best of our knowledge, there is not much concentration on securing it, and it could be vulnerable to some cyber-attacks like Man In The Middle attack (MITM), Denial of Service (DoS) or Distributed Denial of Service (DDoS). In case of MITM where a perpetrator listens to the information exchanged in the channel and commits eavesdropping to steal delicate information or in case of DoS/DDoS attack that could stop or disrupt the communication channel which turns the called controller/ server that is getting requests or receiving information, unreachable.

2. Proposed solutions

To address the above issues, we proposed in our research a suite of algorithms and network specific controller topologies to in order to solve or mitigate the threats posed that could exploit the existing weakness points. Our suite is called the HYDRA since it has kind of hydra heads behavior where you cut one head and you have 2 others replacing it; our suite has 3 different types of SDN controller topologies where we have back controllers to replace the main controller or support each other in case if the controllers were working simultaneously. Those topologies implement specific algorithms used for assuring the security of connection between the controllers themselves. The main proposed algorithms are:

2.1 HYDRA: this proposes method is the sum of all below mentioned methods and the result of their interoperability. Also, in this algorithm we propose a counter attack measurement where every single node in the software-defined network will be provided by the SDN controller with a botnet software; so that in case of any DoS/DDoS attack on any controller of the controllers' topology, those controllers especially the none infected ones will be able to conduct a DDoS attack on the attacker based on the IP of the attacking source or sources then and only then our controllers will block that IP using any kind of blocking technique like Access Control List (ACL).

2.2 VPN: which stands for Virtual Private Network, we propose to use a virtual private network technology to create a secure end-to-end channel between every two controllers in order to prevent any attempts of MITM attacks to jeopardize the east-westbound API.

2.3 Double RSA: The Rivest-Shamir-Adleman famous security algorithm known for its participation in the advancement of cryptography and securing connections can be used but in a double-sided way, where the party 1 uses PUblic Key 1 (PUK1) alongside PRivate Key 1 (PRK1) to encrypt a message then the party 2 uses (PUK1) alongside PRivate

Key 2 (PRK2) to decrypt the message then, if party 2 needed to send a message and become the sender, it will use the PUblic Key 2 (PUK2) which is totally different public key and PRivate Key 3 (PRK3) which is also different from PRK2 to encrypt the message required to be sent while part 1 which becomes a receiver this time; uses PUK2 and PRivate Key 4 (PRK4) which is also different from PRK1 to decrypt the received message from party 2 or node 2. So, in other words we'll have two secure channels between every 2 nodes and that means every node will have a secure channel one for sending an encrypted message and one for receiving a message that will be decrypted later in the other end of the channel, as shown in the figure 2 below.

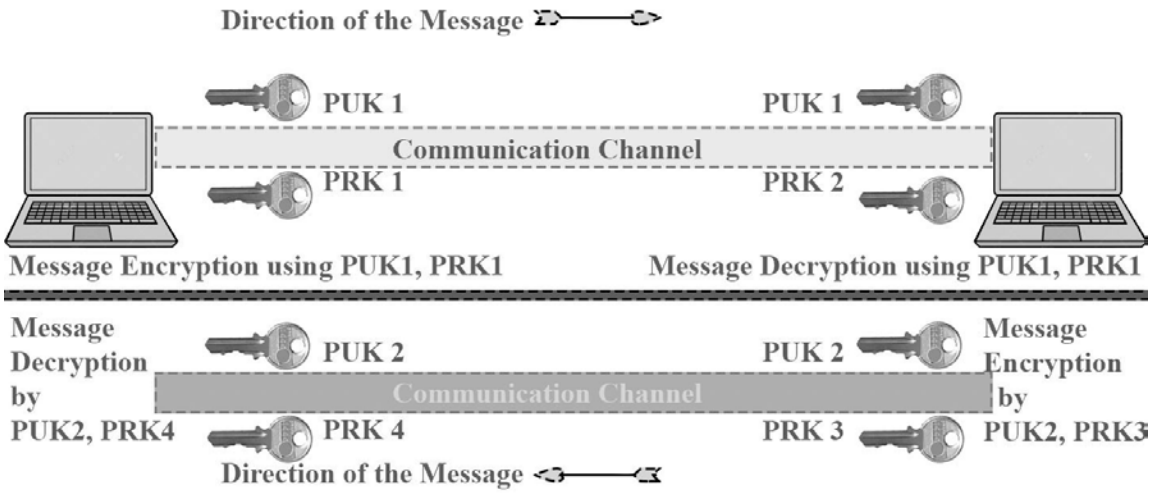

Figure 2. Double RSA algorithm & how it works.

2.4 Blockchain: this article focuses on the usage of blockchain and the ability to incorporate this algorithm into the SDN concept for a more secure software-defined networks, since that blockchain has many advantages and it creates a revolutionary change in any field it's used in. but, first let's take a look to blockchain technology.

A blockchain, originally block chain, is a ledgers' list that can grow as much as needed, called blocks, that are linked using enciphering. Each block contains (an enciphered form of data that is called a hash) of the previous block, a transaction data and the time data referred to as the timestamp.

By design, a blockchain is resistant to modification of the data. It is (an open, distributed ledger that can record transactions between two parties efficiently and in a verifiable and permanent way). For use as a record that can be distributed, a blockchain is usually administrated and run using the point-to-point or peer-to-peer network basis, complying with the inter-node communication protocol and validating new blocks. Once recorded, the data in any given block cannot be altered retroactively without alteration of all subsequent blocks, which needs consensus of the network majority. Although blockchain records are not unalterable, blockchains has a safe design methodology, structure and exemplify a high Byzantine fault tolerance-based distributed computing system. Therefore, Blockchain has the Decentralized consensus [3].

It is thought that Blockchain was developed or invented in 2008 by a person or more than one person that has the nickname or name Nakamoto to be used as a public transaction record for the cryptocurrency named bitcoin which is a one of many types of this new emerging technology and that is a virtual currency in the virtual world of internet,

which could be the future way of conducting monetary transactions and deals. But, Satoshi Nakamoto is unknown. The creation of the blockchain for bitcoin helped to make it as the first digital currency that can solve some prominent issues like the double-spending issue with no need for a central managing or controlling or monitoring entity, trusted authority or party. The bitcoin design has inspired other fields of technology, applications, and blockchains which are readable by the public and usually used as the basis for this virtual enciphered coins named cryptocurrencies.

Blockchain is a promising technology and works as mentioned before as a public ledger which works like a log by keeping a record of all transactions in a chronological order, secured by an appropriate consensus mechanism and providing an immutable record [4] so, we can say that the blockchain is a decentralized ledger of all transactions across a peer-to-peer network. Using this technology, participants can confirm transactions without the need for a central certifying authority.

Usage fields:

The Blockchain technology could be incorporated with multiple areas or integrated into other diverse fields of technology. Nowadays the blockchains are mainly used as a distributed ledger for cryptocurrencies, and bitcoin can be considered as the most notable example.

- **Cryptocurrencies**

Usually most of cryptocurrencies leverage the blockchain technology for transactions recording. Some of the examples of that are the networks of Ethereum and Bitcoin where they are both based on the blockchain.

- **Financial services**

Nowadays many aspects and fields of the financial and banking establishments are conducting banking operations using distributable lists of records that can grow. The Banks are showing a great deal of interest in this technology due to its ability to make the settlement systems of the back office work faster.

- **Smart contracts**

A smart contract can be considered as a protocol procedure within a computer program that conducts operations intended for digitally verifying facilitating, or enforce the performance or the negotiation of a deal that could be between two parties like a contract. Smart contracts allow the performance of credible transactions without third parties. These transactions are trackable and irreversible [5]. The smart-contracts that work on the basis of blockchain could be executed or implemented partially or fully or could be implemented without any human intervention or interaction. The smart contract has many objectives including automated escrow.

- **Supply chain**

There are many attempts and efforts working continuously to employ blockchains in the strategic planning of chain of supply and supply chain management industry. supply chain is just a system or a line of anything related to moving a thing, developing a step or a service, a product or any merchandise whether it was an organization, people, activity, resources or information from the manufacturer to the end user. The activities of supply chain might need the transformation of some raw resources, natural elements, materials or components into another new state as the last prepared product that is ready to be

delivered and consumed by the consumer. one of the main clients of tracking service delivered by IBM and based on blockchain technology is the Everledger.

- **Some other usage fields**

There are many other various ways and fields that could leverage Blockchain technology as the basis of their performance for example it could be used for creating a public, transparent, permanent ledger system for tracking digital use and payments sent to content creators, compiling data on the sales. Nowadays there are new distribution methods that exist and used by the industry of insurance like microinsurance. As mentioned before blockchains could be integrated and incorporated with many other fields because it's a promising technology that's why we have to notice its effect on the IoT technology. Also, another application of blockchain is its application in online voting.

Block chains' types:

There are 4 known categories of blockchain networks and they're: Hybrid blockchains, private blockchains, public blockchains, and consortium blockchains.

- **Hybrid blockchains**

As stated from its name the hybrid blockchain is simply a combination of both public and private blockchains' characteristics. One of the main features of this type is that it permits users of the blockchain APIs to decide what information remains private and what information becomes public. Figure 3 shows a depiction of the general idea and how the structure of blockchain looks like.

- **Private blockchains**

The private blockchain obviously has an access permission. That's why, if someone wants to join it he has to get access by invitation from the network administrators. So, both the participant and validator access is restricted.

- **Public blockchains**

As mentioned before public blockchain has no restrictions of access which means that anyone with just an internet connection may participate in the transaction process or be a validator which of course means he becomes a part of the consensus protocol execution. Now usually such networks offer some kind of economic incentive for those who add or provide a better security for it and leverage any of the existing proof of work or proof of stake algorithms. As known to many people one of the biggest public blockchains examples is the public bitcoin blockchain. Not to forget to mention that Proof Of Work (PoW) itself is a defense measurement against some cyber-attacks like denial of service attacks and other abuses of service like spams sent to a network conducted by submerging it with requests of some services by the requester of the service, which usually means costing a computer a processing time. But, consensus based on PoW is vulnerable to 51%" attacks, a 51% attack may occur when a single miner node, which happens to have exceptionally more computational resources than the rest of the network nodes, dominates the verification and approval of transactions and controls the content of a blockchain [6]. So, it is possible to say that proof of work some kind of an implementation of consensus algorithm which is the process of having nodes accept a new version of the records list [7] and proof of stake can be considered as another type of consensus algorithm that allows a network of a blockchain to conduct distributed consensus. In Proof Of Stake (PoS)-based cryptocurrencies the issuer of the next block is chosen using various combinations of random selection and wealth or age (i.e., the stake). Unlike the cryptocurrencies that use

PoW like bitcoin which leverages the mining strategy; which is, an intensive puzzle that requires the nodes to solve it computationally to validate transactions and create new blocks [8].

- **Consortium Blockchain**

To understand and get a better comprehension about consortium blockchains it is better to compare it with public blockchain, public blockchain possesses no access restriction, which means that anyone with an internet connection can be a participant in a public blockchain. So, anyone in the world is capable of reading the data included in the blockchain, and is permitted to conduct or implement transactions on a public blockchain [9]. Also, there is no access restriction on the participation of blockchains' consensus process, this process determines the entity or individual that can add a new block to the blockchain. Public blockchains are known as fully decentralized blockchains, and they have control over the blockchain for making it out of the hands or control of any single individual or entity.

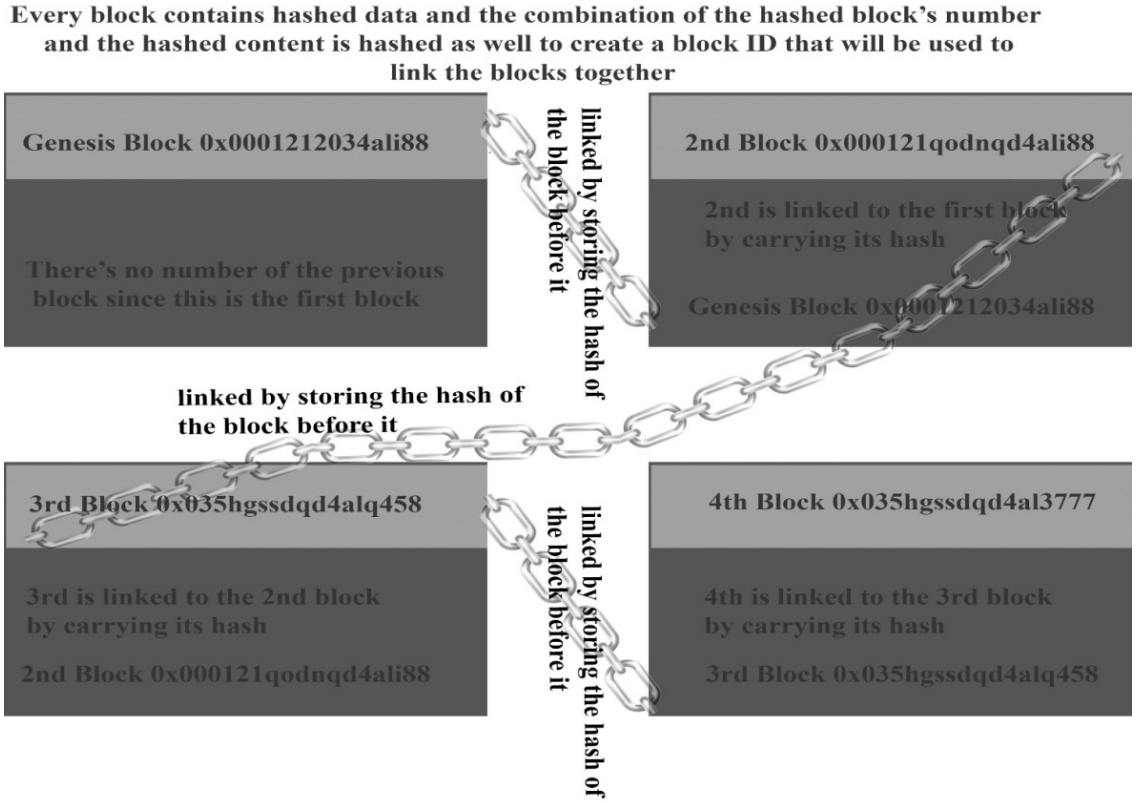

Figure 3. Blockchain Technology structure.

While on the other hand consortium blockchains differ from their counterpart in the access or permission feature since they are permissioned, hence, not anyone who has an internet connection could gain access to the consortium blockchain.

That's why it's possible to consider consortium blockchains as semi-decentralized. The control over the consortium blockchain is not granted or given to a single individual or entity, but to a group of approved individuals instead.

The consensus process of the consortium blockchain is different usually from that of a public blockchain. Where we can see that participants of the consortium blockchain's process are usually forming a group of pre-approved nodes in the network rather than making everyone allowed to participate in the procedure. That's why, the consortium

blockchains inherited security features from public blockchains, while providing a greater degree of network control as well.

Practice:

To the best of our knowledge despite the existence of the proposed Marconi protocol, blockchain is not regularly used to secure software-defined network communications yet and its usage could be of a great deal of advantage to the cyber security field [10].

Blockchain can track information and store data in a chronological fashion and if there's a change in the data stored, its block will not be tampered with or changed but a new block will be created instead, containing information about the adjusted or changed data for example adding a new block which states that x changed to y.

before a block gets added to the chain of block a few things must happen:

- A cryptographic puzzle must be solved and that will create the block.
- Proof-of-work, which is the process of sharing the solution of the puzzle by the computer that solved it with all the nodes of the network.
- Then the network verifies this proof-of-work and if it was correct then, the newly created block will be added to the chain.

Our framework will leverage blockchain by mining a special blockchain for the network environment and creating blocks of hashes. A hash is just a hexadecimal number created with a fixed length using a hash function like SHA 256 which is a one-way cryptography meaning that unlike usual two-way cryptography that can be encrypted and decrypted, the hashing can't be decrypted. To verify if a block's hash has the right value and the block can be added to the chain or not we set a limit or a target number that will be like the limit that the hash value should be less than and not to exceed it.

Then we create a value that represents the difficulty but, what the difficulty value will do? We'll subtract it from the target number which means that the bigger the difficulty number is the less target number we'll have which means it will be harder for the computer/controller to calculate the block hash value since it has to be less than the target number and that means that the controller will take longer time to guess the right answer hence, it will create more hashes for a block because number of hashes represents the number of attempts done by the device to figure out the right block hash, and that is the meaning of the mining difficulty. Our proposed framework uses blockchain technology to create a secure channel in one of the following ways:

- Creating blocks of hashes between each 2 controllers to authenticate and validate the connection channel between them.
- We could use blockchain to start an authenticated session so, it will be only used at the beginning of a session which means it will not consume much resources and it will not be repeated along the session.
- if we wanted a more robust but time and resources consuming approach we could add a hash in every packet and incorporate it with the packet's head information, which means using them along the session till the end of the session to verify the integrity of the packets and prevent them from being tampered with because in case of any attempt of injecting new information in the packet or adjustment or change in the packet's head, it might require changing the information in the packet's head where the hash info resides and that will change the hash info so, it will be detected as a packet that is tampered with

hence, it will not be accepted and dropped. this approach is also not feasible due to its heavy load on the network bandwidth and not to forget that it consumes the time and other network resources.

- Another approach would be using blockchain as hashes for updates of configuration every 10 seconds. So, Blockchain could be leveraged for connection authentication.
- For storing the data of updates in blockchain we mainly can store the configuration updates in a distributed network like InterPlanetary File System (IPFS) where the data will be stored on multiple distributed network nodes then will get the hash of that data and store it in the blockchain, once we need to retrieve the data, we get the hash from the blockchain which is guaranteed to be safe and tamper-proof and use it to restore the data from IPFS.
- Since blockchain is an innovative type of database so, there's a chance to use it for both recording and encryption. Every 10 seconds there will be an update of configuration broadcasted from the main controller to the backup one, regardless of the type of topology; the data sent from the main controller will be hashed and the data will be sent alongside with its hash so that the backup controller will be prepared to compare the hashes coming from the main controller next time and check for two things:

1. if the next hashes fit the type of hashes it has or not.
2. since any slight change in the data means new hash, if the main controller tried to change the data or update a data of a specific node again and it was with a different length or type of hashes, then it will be rejected by the backup controller after being compared with the previous blocks of hashes and the backup controller will be alerted of a breach in the main controller.
3. Configuration updates have a timeline, for instance at hour 15:00 the ACL or access control list was used to block network 10.10.12.0, at hour 16:00 another network was added to the list let's say 11.11.14.0; by using blockchain which is a distributed ledger that keeps the time order of information and data stored in it and prevents everybody from tampering with it, we can protect the data from being tampered with by the perpetrator who might try to change the time sequence of those configuration updates; even if he tried to insert a code to manipulate a block let's say block of index 2 and that block contains information about blocking a specific network and changed it into granting access to that network, his new block will not match the chain because it will have different hash from the original one and it will have to have the hash of the previous block and of course, the previous hash contained in this new block is different from the original one since it's a new block that has been tampered with and for those reasons and others, this new modified block will be considered as an invalid block so it will not be accepted by the chain. The configuration time period which is 10 seconds will force us to use the simplest aspect of blockchain cryptography in our proposed framework for that it's a short period of time which means that the difficulty will have to be not so high but effective in the same time, why not so high? Because our goal is to protect the network information from corruption and make the data tamper-proof not create high efficient cryptocurrencies because higher difficulty means more load on the network and higher consumption of its resources and that will make the network more DoS/DDoS attacks - prone which will take us back to where we started; because we have to make a balance between our needs of security and the load on the network nodes. So, in other words we can simply say that we'll make a puzzle that nodes of the network, in our case the controllers will have to find its result in

order to validate one block in the chain which will contain the data of network configuration and that will be done to each block hence, we can say that each update of configuration will be a block in the blockchain of update flowing through the network structure. The puzzle is done by including the configuration information into the blockchains and that's by adding a number to the data that the hash number resulting from mixing both the number and the data together will have a specific range of zeros and that is the difficulty we choose as per our needs in blockchain technology and that will make the node try guessing the right number by choosing different numbers randomly till finding the right number which won't be so hard or so easy but it won't take long for the reasons mentioned before but, it will be in 10 seconds or less than 10 seconds and we'll program the app to wait for 10 seconds before sending the next update message even if the node was able to guess the right number and create a hash of a valid block in 2 or 5 seconds, so that it will be of a steady consistent pace; this number is called a "nonce", which is a concatenation of number used once. In the case of bitcoin, the nonce is an integer between 0 and 4,294,967,296; that kind of guessing will be the proof of work-like for our framework and that will make it hard to tamper with the data because if the attacker tried to add some blocks of data of his own, these blocks will have to have the right required number to be added to the data of network and passed to the hash function to create a specific hash number with starting specific number of zeros and that's the real puzzle for network's node and to make his block contain the hash of the previous block, all that has to be done in order for him to be able to add a valid block of data and tamper with the information stored in the chain; it will also require the perpetrator more time and create more obstacles in his way in order to manipulate the data mean while our framework and the already built-in techniques will detect him so fast maybe before he even can start guessing. Of course, this is merely for research purposes just to prove the ability to use blockchain technology to insure the security of the data flowing and broadcast messages of software-defined network. Finally, we can use the Marconi protocol to achieve some or all of our ideas but, what is Marconi protocol? It's a new protocol specialized to help securing Ethernet-based networks using some blockchain-based technology by targeting layer 2 of the Open System Interconnection (OSI) model of networks which is the datalink layer and creating programmable packets, it allows developers to create and deploy intelligent, decentralized networking applications that can be run by nodes or end users. Blockchain projects, private institutions, and enterprises can utilize the network and the platform it's built on to manage their infrastructure and develop smart distributed networking and cybersecurity services. also, as an example of applications that could be developed by Marconi include Software-Defined Networking (SDN), anti-malware and anti-virus protection, intrusion detection and prevention systems (IDS/IPS), Virtual Private Networks (VPN), Content Delivery Networks (CDN), and some new blockchain protocols. But there are some differences between our proposed framework and the Marconi protocol for instance, our framework works on Transport Layer Security TLS layer, since that it uses the RSA asymmetric cryptography algorithm and openflow SDN protocol that connects the controller with the switch and both of them work on that layer. also, our proposed framework works on the network layer since it uses the IPsec algorithm that is the basis of VPN technology which is used by the network layer while, Marconi uses datalink as its work environment so, our framework targets different layers and even if it might somehow work on datalink then, it will not be the only layer that it targets. Also, it uses smart contracts for defining how long the exchange of

data will be and how much data can be exchanged, and at what price of fuel while ours, could use smart contracts only for authentication of the connecting parties. Also, the network operators, individuals, and Internet Service Providers or for short (ISPs) can contribute by donating some of their assets like compute resources, bandwidth to the network. For that contribution the participants periodically gain network tokens known as marcos. a marco is just a name for the main measurement unit of the fuel consumed for network usage, computing and distributed networking, administration, and processing of smart contract; while our framework doesn't do that, it is merely using an aspect of the blockchain technology and leveraging its cryptography for securing some of the information exchanged between the controlling nodes in the control plane of SDN. End users can utilize the Marconi network to access the internet or nearby compute power, either by procuring marcos or by mining them through operating a contributing node, while ours doesn't have that and it doesn't have that kind of complexity, these differences and others are differentiating our proposed framework from the Marconi protocol.

- In the future we could develop this method into a new approach by sharing the blockchain across the whole network nodes including PCs, switches, routers etc.

Our app could be developed in the future to serve deeper purposes and to have higher and better features, we could even make the time period of the configuration updates shorter or longer as per our needs. Of course these methods and approaches will be filtered in third chapter by the programming of the application that will implement the proposed framework and show more details about the limits of our methodology. Also, in the future some issues might emerge and have to be addressed like the running out numbers guessed to create valid blocks which means running out of blocks and that could be after a really long time; just like what will happen to bitcoin cryptocurrency in the future, where it is estimated that mining the available 21 million bitcoins will reach its maximum peak and end of course by the year 2140 so, getting back to our situation if this case will be faced by the network administrator then it is simply possible tweak some features in the blockchain created and distributed among the controllers of the software-defined network, one of the features that could be changed is the difficulty which means that the nodes will create a totally new blockchain containing the rest of configuration updates. And that could mean restarting the app and getting back to work but, it won't take a while and if needed it will be done after a long time as mentioned before.

In a whole the general shape or form of the suite of algorithms applied in the framework will be like what is shown in the figure 4 below:

Figure 4. The architecture of Hydra suite containing the proposed algorithms.

3. Conclusions

This article gave a description for the general structure of the software-defined networks, with it came a simple explanation for the main weakness points from the research's point of view and the proposed solutions to address these issue comes afterwards. Then, comes after that an explanation of the integration of all the proposed solutions into one suite or a framework named HYDRA. This article focuses mainly on one of the suggested algorithms which is already used in other technologies like cryptocurrencies and also in a less way in networking but not as it is proposed here in the article and this technology or algorithm is the blockchain which is already getting more and more attention in different fields of technology due to its conceptual ability to provide a better level of security to any field it's integrated with. In the bottom line we can get the following results:

- Software-defined network is a great technology that can solve some security issues but poses some other ones in the same time.
- There is a need to suggest a solution for those issues and that's done by proposing a whole suite of network controllers' topologies and algorithms that can help solve or at least mitigate some security issues like MITM, DoS/DDoS attacks that might disrupt the east-westbound API or exploit the centralization of SDN structure.
- Blockchain is another great technology that can be incorporated with any exiting technology to push it further into a better situation.
- We have proposed to use blockchain for encrypting the configuration updates between multiple controllers in the SDN environment to secure them from being tampered with.

References

1. Reijers W, O'Brolcháin F, and Haynes P. 2016. Governance in Blockchain Technologies & Social Contract Theories. Ledger 1: 134-151. doi: https://doi.org/10.5195/ledger.2016.62.
2. Vyacheslav Kunev. Extended RSA-M algorithm as a way of increase computational complexity of cryptosystems. Journal of Engineering Science, Vol. XXV(2) (2018), pp. 45-56.
3. Blockchain. [online]. [accessed: 25.07.2019]. available: https://en.wikipedia.org/wiki/Blockchain.
4. Puthal D., Movants S. P. Everything you wanted to know about Blockchain: its Promise, Components, Processes and Problems, 2018, pp.1-9.
5. Smart contract. [online]. [accessed: 25.07.2019]. available: https://en.wikipedia.org/wiki/Smart_contract.
6. Xu J. J. Are blockchains immune to all malicious attacks, 2016, pp. 1-9.
7. Del Rio, C. A. Use of distributed ledger technology by central banks: A review, 2017, 8 (5), pp. 1-13.
8. Proof of stake. [online]. [accessed: 27.07.2019]. available: https://en.wikipedia.org/wiki/Proof_of_stake.
9. IBM. 2016. Hyperledger Architecture Working Group. Available: https://www.hyperledger.org/wp-content/uploads/2017/08/Hyperledger_Arch_WG_Paper_1_Consensus.pdf.
10. Charles Brett. Marconi adds Diffie and launches its Mainnet for scaling complex networks. May 17, 2019. Available: https://www.enterprisetimes.co.uk/2019/05/17/marconi-adds-diffie-and-launches-its-mainnet-for-scaling-complex-networks/.