International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 12 Issue 06, June-2023

Software Defined Wireless Network: The Rise, The Evolution, The Future

Shubhangi Saha
Dept. of ECE- DSCE, Bangalore

Abstract- **With the advent of computer networks, the world witnessed a growth in communication and data industries. These networks did not take long to take a leap from large scale networks to very large networks. Today, the internet forms the largest network of computers in the world with way more devices connected to it than the human population. This has increased the work flow of the network but also increased the strain on businesses running these networks and also on research institutes to constantly find ways to unburden the current network topologies and their extensions. Thus, SDN comes into play in the twenty-first century. To use the quote "***There is nothing you can do with SDN which cannot be done without. Rather, SDN is about simplification and evolvability"* **would be appropriate**.

Keywords- SDN, OpenFlow, SDWN, SDN-WISE, Wireless Sensor Network.

I. Introduction

SDN *(Software Defined Network)* centralizes the functioning of the control plane of a router or many routers in a network. Every router operates on two level: the data forwarding plane and the control plane. The former is associated with the reception and dismissal of the data packets to its next routing address. The later operates on the former on how this dismissal to the next router will take place (distributed architecture). The control plane of a router analyses the traffic in the adjacent routers and determines the most suitable path for the data towards its destination node. The limitation of this control plane lies in the fact that it can only analyze its immediate next routers and not all routers that must be considered for the path taken by the data to reach its destination. SDN operates by centralizing the control plane from various routers via an application called the controller. Thus, the cost inefficient and inflexible routers are left only with their data plane (almost working like switches that connect the edge, border and control nodes) while this controller gets to connect to routers (all or many), analyze data traffic over the entire network and find the best route for data from source node to destination node.

II. Early SDN And OpenFlow

Any SDN (OpenFlow) have three basic layers in its architecture, viz., the application layer (APIs), the control layer(controller) and the infrastructure layer (the layer of connected network devices). The communication between the controller and the its network devices is called South Bounding Interface and the communication between the controller and the application scripts is called north bounding interface. The physical network of the network devices and their connections with the IP connectivity is called Underlay. Abstracting the underlay is the Overlay (the virtual network). The fabric of the infrastructure layer comprises of the underlay and the overlay as an entirety. The OpenFlow protocol that operates on the south bounding interface of SDN was among the earliest and the most successful to have been created. In OpenFlow's architecture, the forwarding devices are called OpenFlow switches. Each switch has a flow table into which the controller propagates the forwarding rules. This, thereby, makes the device decide the flow of traffic, hence, decoupling of the data plane and the control plane. The flow table constitutes three sections: The Mark Fields (12 tuple of packet header fields to allow IPv6), Counters (to accumulate statistics on the various traffic flows via the forwarding device) and Actions (instructions that handle matched packets) along with a table-miss flow entry that holds action for packets that do not match the requirements of the specific flow entry. These packets can of five types: Type 0- Data packets, Type 1- Beacon packets, Type 2- Report packet, Type 3- Rule/Action Request packet and

Ingress port into OpenFlow switch	Ethernet source address	Ethernet destination address	Ethernet segment type
VLAN identifier	VLAN priority	IP source address	IP destination address
IP protocol	IP type of service	TCP/UDP source port	TCP/UDP destination port

Table 1: OpenFlow Match Fields (from Adam Drescher, A Survey of Software-Defined Wireless Networks. 2014.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 12 Issue 06, June-2023

Type 4- Rule/Action Response packet. On the other hand, no such standardization interface is there for north bound interface. Frequently used tools with OpenFlow are NOX and FlowVisor, being a network operating system and a controller for virtualization of network respectively. FlowVisor aids by achieving an easy traffic isolation whereas NOX provides a global view of the network.

III. SDN IN WIRELESS

The world kept evolving and so did computer networks. WLANs did not take long to take over the traditional LANs. A growing popularity of the wireless connections, both for domestic and multinational network, grew hand in hand. Unlike wired networks, wireless networks had their own domain of limitations. A majority of this limitation are because of the interference that wireless networks are subjected to.

The first SDN architecture resembled the OpenFlow architecture which was first designed for wired networks. As the focus diverged to wireless networks, new SDN functionalities were needed to be developed. The SDN in wireless networks (SDWN) involves following features: Slicing, control strategies and traffic engineering.

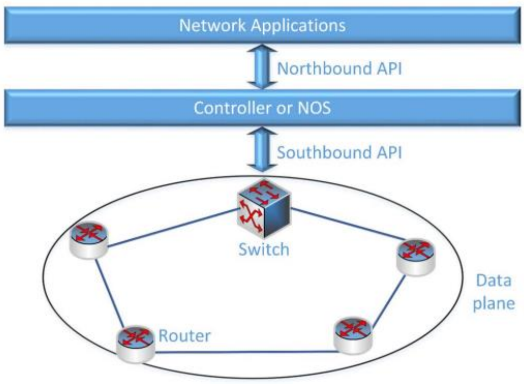

Figure 1: SDN Architecture (from Israat Tanzeena Haque, Nael Abu-Ghazaleh. Wireless Software Defined Networking: A Survey and Taxonomy. 2016.)

	One Controller (Centralized)	Many Controllers (Distributed)
Cost	✓	
Complexity	✓	
Reliability		✓
Response Time		✓
Amount of Control Traffic	✓	

Table 2: Comparison between Centralized and Distributed Control Strategies (from Adam Drescher, A Survey of Software-Defined Wireless Networks. 2014.)

Slicing allows separation of flow of traffic into distinct subspaces. Slices share network resources virtually so that many slices can run on the same hardware. There can be equal or a greater number of slices and channels. There is a need for multiplexing in case of more than one slice using a channel that is inclusive of guard intervals or fine-grained timing depending on whether TDMA or FDMA is used.

Control strategies are useful when the network deals with more than one SDWN controller. Usually, the use of single controller is promoted as it is simple to deploy (in both Brownfield deployment or Greenfield deployment) and cost effective but this type of network has low reliability as the entire network operation gets dependent on the single SDWN controller. Therefore, multiple controllers can be deployed to the network in order to increase the reliability of the network. In such cases, there must be protocols via which the various controllers can communicate amongst themselves without any hindrance. Such cases are handled by the network operator.

Traffic Engineering in SDN is purposed for load balancing. If a certain traffic threshold for traffic is reached, the controller initiates a new set of rules (actions) in order to unburden the current loaded link and delegate the excessive load to other routes. This is also referred to as the flow-based routing.

SDWN is required to have provisions to support duty cycles, in-network data aggregation and flexible definition of rules (A rule is a description of the characteristics which are featured by packets belonging to a flow and that must be treated by the node in the same way) along with being robust to topology changes, robust to packet losses and node faliures. The SDWN architecture specifies two types of nodes: the generic and the sink node. Forwardinglayer and Aggregation Layer (follows Aggregation Equivalent Flow) are inclusive to the generic node. The Adaptation Layer and the Virtualizer Layer are inclusive to the sink node. The

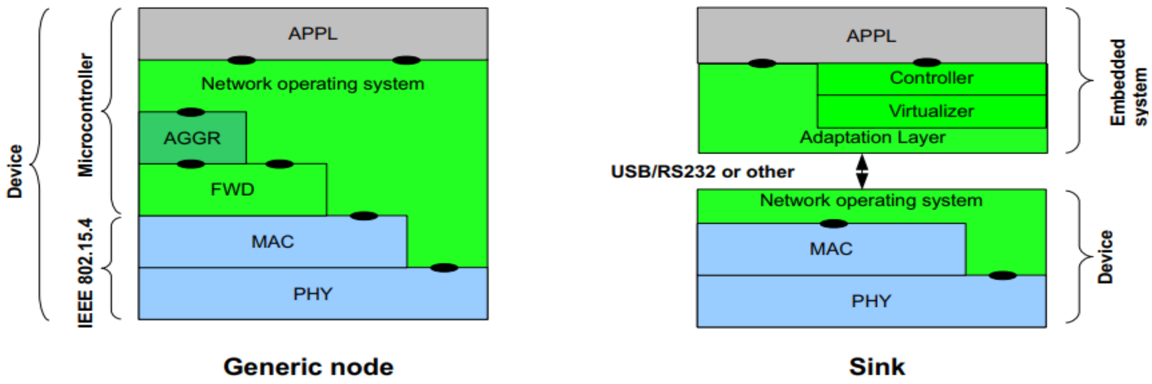

Figure 2: Protocol Architecture (from Salvatore Costanzo, Laura Galluccio, Giacomo Morabito, Sergio Palazzo. Software Defined Wireless Networks: Unbridling SDNs. European Workshop on Software Defined Networking. 2012.)

newness of the SDWN reflects upon its various other features too like the use of wireless medium, the expansion of functionality of the data plane and dynamicity of a greater scale. Wireless connections, by convention, tend to have lower connecting speed than wired networks. Businesses tend to switch to small cellular networks for achieving a higher speed which in turn leads to higher levels of interference along with decreased transmission range of signal and in most cases, an oversaturated network. As a result, network designers choose to have larger channel size that has low transmission range. These channels need several access points to function. On the basis of the density of access points in a network, wireless networks are either Infrastructure-heavy wireless networks or Infrastructure-light or Zero-Infrastructure wireless networks. Therefore, SDWN is widely classified as follows:

A. Infrastructure-heavy SDWN

WiFi and WiMAX, two handover algorithms that paved the way for wireless network testing using OpenFlow. By apply SDN to "extremely dense wireless networks, the CROWD project came up with solutions for emerging WLANs and Cellular networks. Infamous issues with the dense wireless networks like high energy consumption and higher interference by twitching a specific few 802.11 parameters and an approach called Almost Blank Subframe Paradigm wherein subframes that lead to higher interference are put on mute in the network. An SDN framework with peak application in WLANs was Odin that provided more flexibility and extensive services than traditional WLANs. On the other hand, cellular networks used fine-grained subcarrier coordination in order to reduce inter-cell interference. Cellular networks that used packet gateways (P-GW) in the network achieved a higher distributed architecture, higher scalability, higher virtualization along with cheaper network switches when switched to SDWN. SDN also ensured vendor neutrality in both networking businesses and research institutes. Sub-categories of interests developed like the Soft-RAN (Software Defined Radio Access Network) and Soft-Air (Software Defined Networking Architecture for 5G), etc.

B. Infrastructure-Light SDWN

These are networks that are not constructed with a definite number of network device. These networks (usually consumer-based networks) tend to evolve in numbers and in topologies along with unpredictable emergence. These can be of types like the wireless mesh networks (WMNs), home networks and wireless internet service providers. The growth pattern of these networks is uncertain which imposes a serious problem to the existing network about its deployment of new devices without disrupting the existing one and not compromising on the quality of service. Though these are cost efficient services, they put strain on the network to evolve constantly. An approach to increase this flexibility is the implementation of SDN-WISE.

IV. SDWN TO SDN-WISE

SDN-WISE (Software Defined Wireless Sensor Network), comprise of a set of resource limited sensor that are installed to monitor a specific target. These have to be energy efficient, coverage and topology management efficient and application specific to function properly. The various sensors in the wireless sensor network have a specific target during a process. The target sends data to the sink node which thereby is accessed by the end user via the Internet. This implements multi-hop mechanisms, many to one and one to many communications with energy and processing power limitations. This type of network is prone to attacks, failures and unreliability. The solutions to these have to be network specific. The architecture of SWN

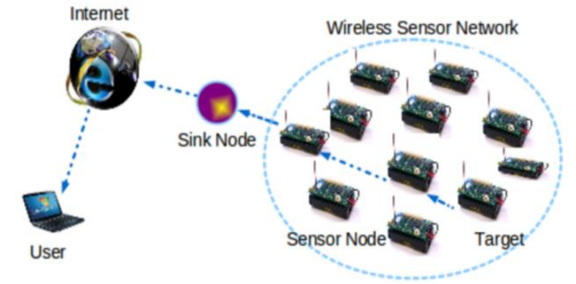

Figure 3: SDN-WISE (from Giacomo Morabito. Wireless Software Defined Networks, International Conference on Information Networking – Chiang Mai, Thailand. 2018.)

(Sensor Wireless Network) is focused upon one or multiple centralized base station or sink to formulate and execute the task of gathering data. This offers the controller a global network view that offers proper resource allocation along with management of the network. Factors like network slicing and abstraction, data plane programmability and protocol evolution can be carried out easily with the amalgamation of SDN in WSN.

Two major requirements of SDN-WISE are multi-tenancy and statefulness. Multitenancy is achieved through an abstraction tool called Wise-Visor. It abstracts the real network by creating different views for various tenants to use a common packet. In the statefulness parameter, a buffer memory is reserved for the state information like rules and actions. Use of the state information should be for conditional forwarding, multipath Routing and support of QoS. The two major problems faced being reqirement of platform dependent implementations and time management due to statfulness is achieved by the use of a network OS or NOS.

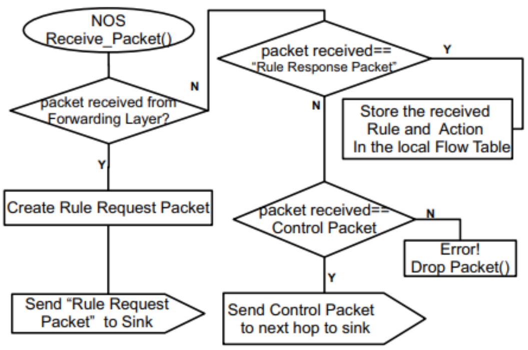

Figure 4: Flow Chart of NOS Operations (from Salvatore Costanzo, Laura Galluccio, Giacomo Morabito, Sergio Palazzo. Software Defined Wireless Networks: Unbridling SDNs. European Workshop on Software Defined Networking. 2012.)

The standardization of these various aspects of SDWN has been taken up by the Open Networking Foundation (ONF). Along with the core standards, the body also aims on standardizing testing specifications and extended specifications for future use.

V. CONCLUSION

This paper highlights the key features of SDN and SDWN in specific. It revolves around the idea of how SDN is not doing something impossible but giving exemplary solution to things in order to make it better. The technology does have some current limitations but with the current pace of advancement in computer networks, it would either improve or give way to an even better approach. Nonetheless, SDWN has proven to be a revolutionary aid to solve network controllability issues. The paper focuses on the milestone that SDWN has achieved in the past decades and a few problems whose solutions are being sought for by the extensive research community that it interests.

REFERENCES

[1] Adam Drescher, A Survey of Software-Defined Wireless Networks. 2014.

[2] Giacomo Morabito. Wireless Software Defined Networks, International Conference on Information Networking – Chiang Mai, Thailand. 2018.

[3] Salvatore Costanzo, Laura Galluccio, Giacomo Morabito, Sergio Palazzo. Software Defined Wireless Networks: Unbridling SDNs. European Workshop on Software Defined Networking. 2012.

[4] Carlos J. Bernardos, Antonio de la Oliva, Pablo Serrano, Albert Banchs, Hao Jin, Luis M. Contreras, Juan Carlos Zúñiga. An Architecture for Software Defined Wireless Networking.

[5] Israat Tanzeena Haque, Nael Abu-Ghazaleh. Wireless Software Defined Networking: A Survey and Taxonomy. 2016.

[6] Ahmed Abdelaziz, Abdullah Bin Gani, Faiz Alotaibi. On Software-Defined Wireless Network (SDWN) Network Virtualization: Challenges and Open Issues. Article in The Computer Journal. October. 2017.

[7] Musa Ndiaye, Gerhard P. Hancke and Adnan M. Abu-Mahfouz. Software Defined Networking for Improved Wireless Sensor Network Management: A Survey. 2017.

[8] Faridullah Amarkhil1, Prashansa Taneja. A Research Paper of Security Enforcement Policy for (SDN) (WLAN) Software Defined Network. International Journal of Engineering Research & Technology (IJERT). Vol. 9 Issue 06, June-2020.

[9] José Quaresma Filho, Nailson Cunha, Robertson Lima, Eudisley Anjos , Fernando Matos. A Software Defined Wireless Networking Approach for Managing Handoff in IEEE 802.11 Networks. Hindawi Wireless Communications and Mobile Computing Volume 2018.

[10] Claude Chaudet, Yoram Haddad. Wireless Software Defined Networks: Challenges and opportunities. Conference Paper · October 2013