Software Defined Networking

By: Nisha Ahuja

TABLE

OF

Contents

Traditional Network

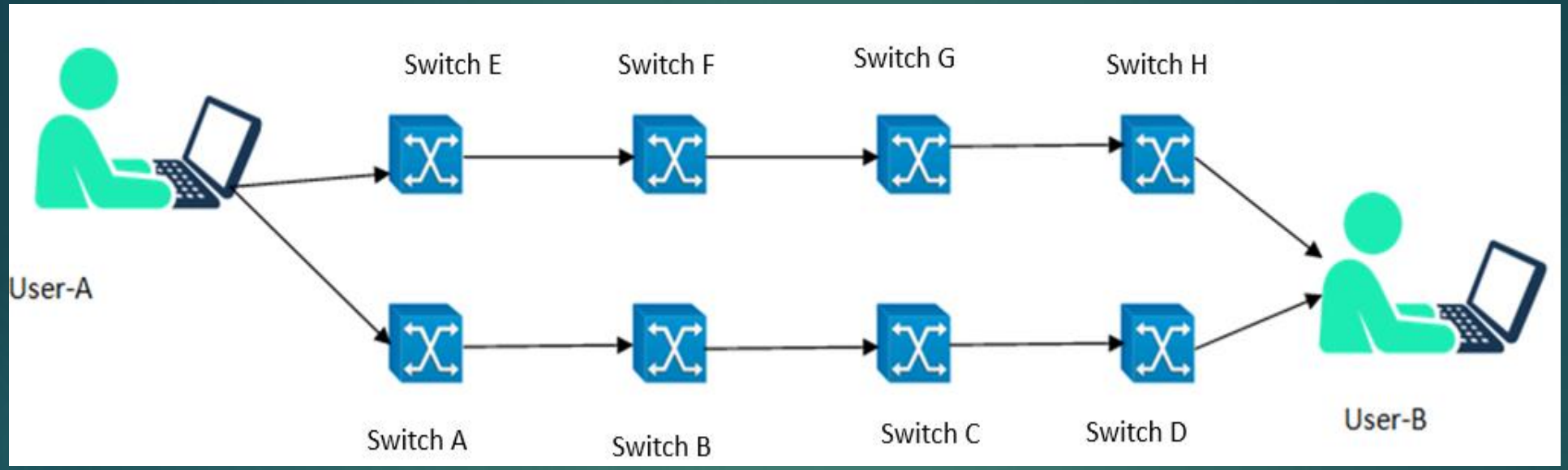


Fig : Traditional Network Architecture

- Traditional Network[1] is based on destination address forwarding.
- Routing is done based on the protocol configured in the switch.(RIP, OSPF, IGRP, EIGRP etc.)
- Switch here are layer 3 switch in which OSPF protocol is executing.
- Every switch forwards packet based on the routing table[4] it posses. Switches does not have global view of the network.

Traditional Network in attack Scenario

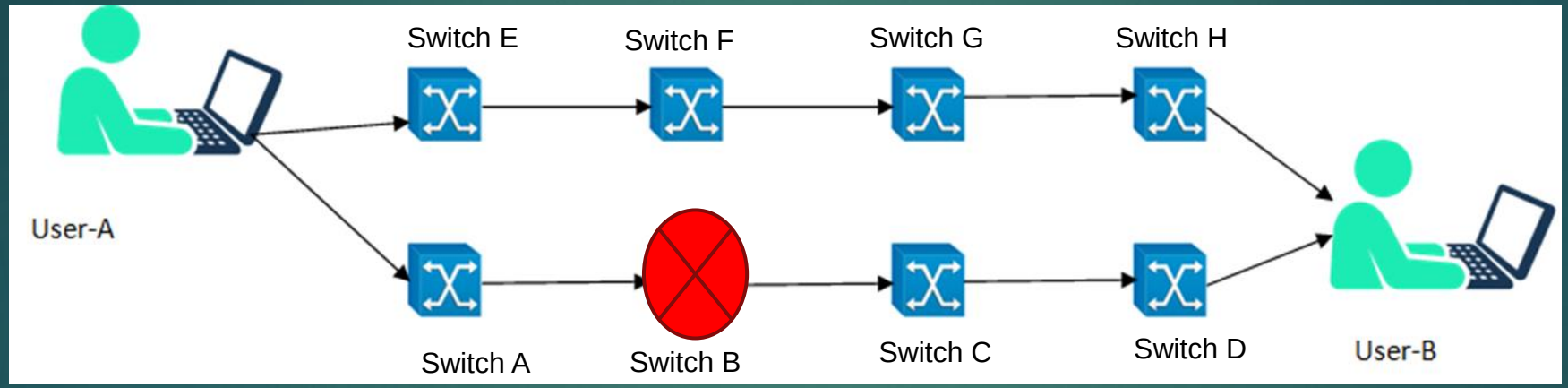


Fig : Attack Scenario in traditional architecture

- Switch C got attacked.
- Need to found other route through which the traffic can be routed.
- But there is no centralized entity which can take care of the rerouting in present network.
- Depending on the protocol the rerouting will occur.

Software-Define Networking (SDN)

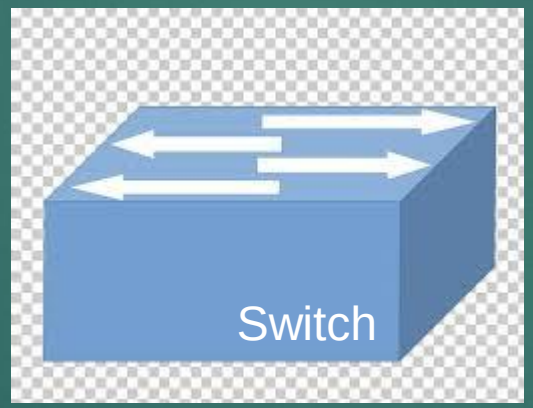


Fig : SDN Concept

- When the switch has no entry in the flow table, the switch contacts the controller.
- Controller add the route to the destination in the switch's flow table [3].
- By using the flow entry switch will send the subsequent requests.

SDN (Continued……..)

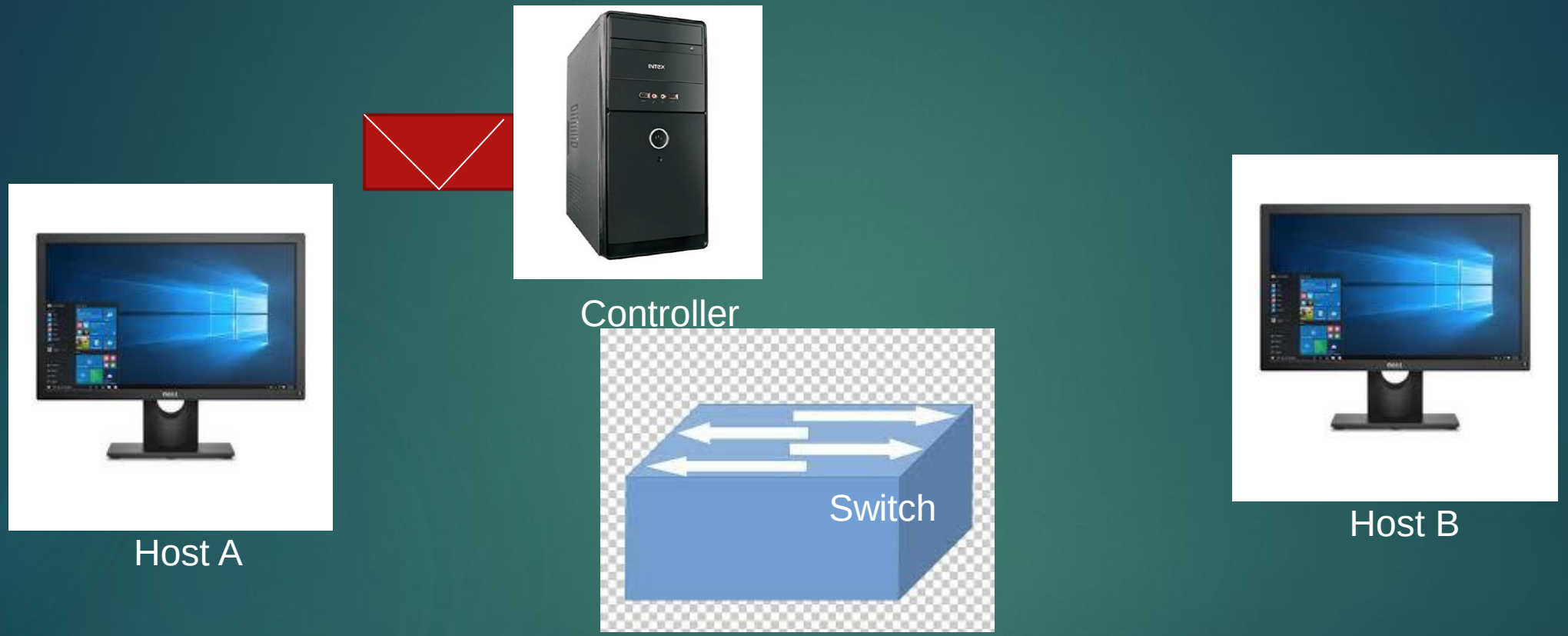


Fig : SDN Concept

- Once the switch's flow table [1] has entry of a route.
- It use that entry and send the packet to the destination host.

SDN Architecture

- Separation of Control & Forwarding Plane
- Programmability & Automation
- Network Device Control Based on Policies.

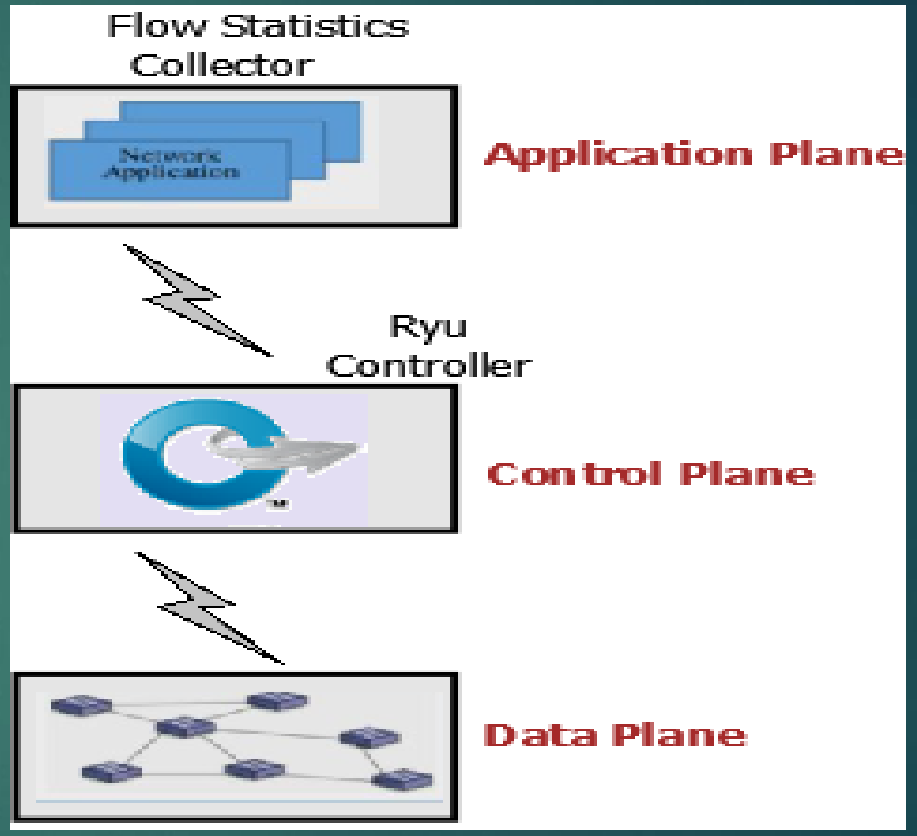


Fig : SDN Architecture

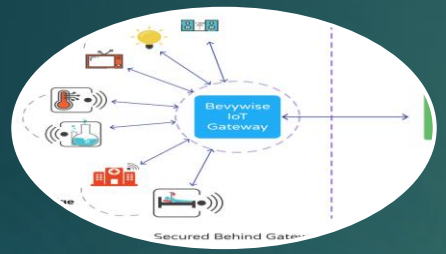
Traditional Network vs SDN

Table-1: Traditional network VS Software defined networking

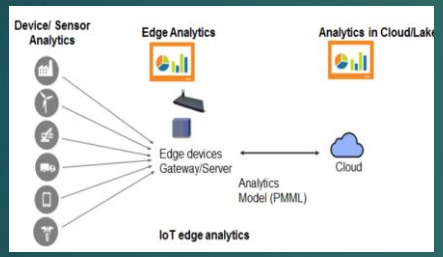
Category	Traditional Networking	SDN
Type of Network	Human intervened	Programmable
Communication Overhead	No	Yes
Vendor Dependent Switches	Yes	No
Error Prone Configuration	Yes	No
Expensive	Yes	No
Network Status Availability	No	Yes

Simplified IOT Architecture

Data collection from sensors . Actuators act upon the data collected and provide some output.



Sensor data aggregation and analog to digital data conversion.



Edge IT System performs enhanced analytics and preprocessing on the data.

Analysis, Management and storage of data.

Challenges of IOT

Real Time Programming of sensor nodes.

Vendor specific architecture of sensor nodes.

Resource Constrained Nodes.

Limited Memory & Distributed Control of Sensor Nodes.

Opportunities in IOT

- Programming the sensor nodes in real time.
- Changing the forwarding path in real time.
- Integrating different sensor nodes in WSN.
- SDN can be used to manage and control IOT network.
- Wireless sensor nodes and network can be controlled using SDN based applications.
- Network performance can be improved significantly using SDN based approaches over traditional approaches.

SDN Integrated with IOT

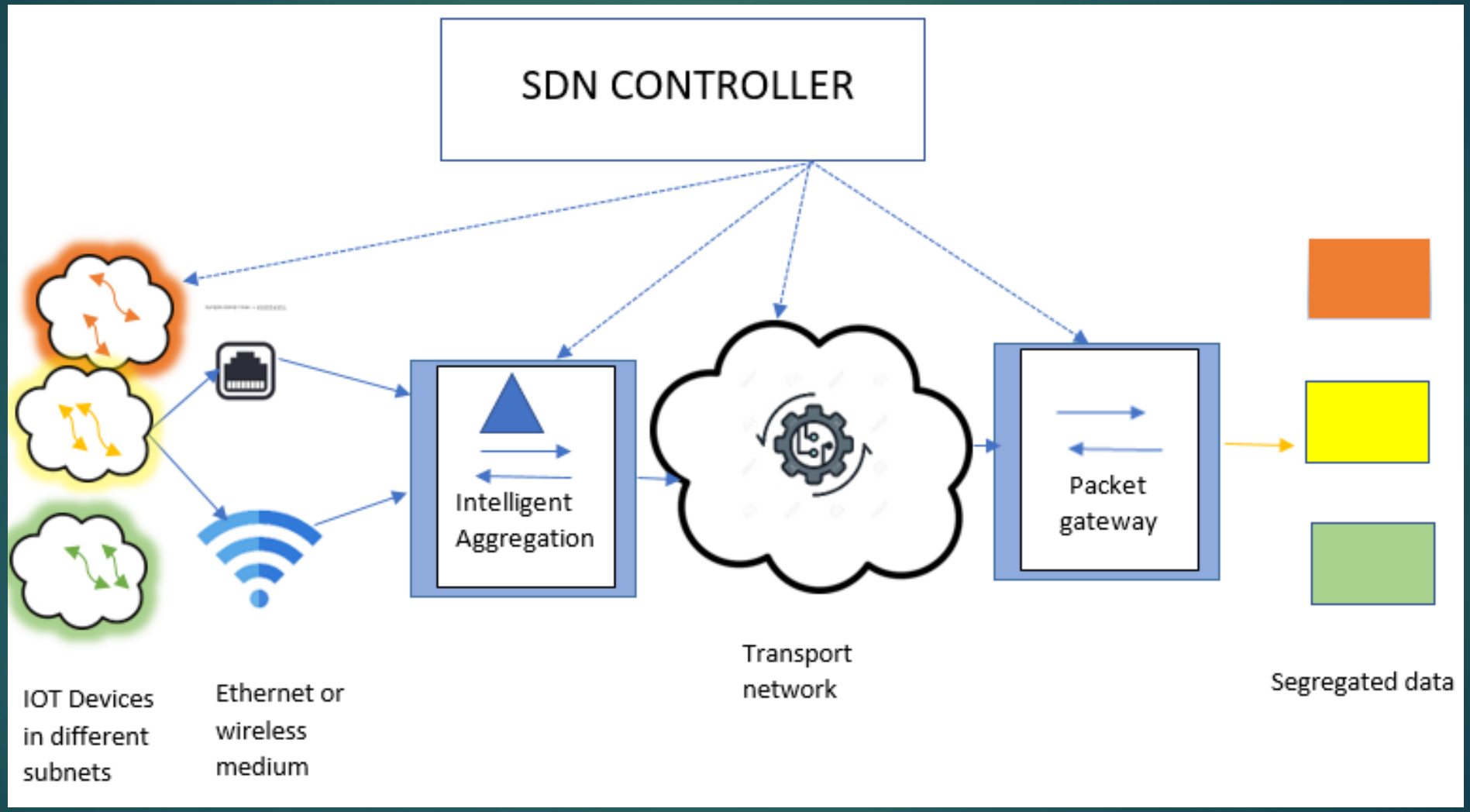


Fig: SDN integrated IOT architecture.

Benefits of integrating SDN to IOT

- Intelligent routing decision
- Simplify Information Collection
- Decision making made simple.
- Intelligent traffic pattern analysis

Soft-WSN: Software-Defined WSN management system for IOT

- Component Based Approach:
  - Device management
    - Sensor Management : Sensors can be used depending on application.
    - Delay Management : Delay for sensing can be set dynamically.
    - Active Sleep Management:
  - Topology Management
    - Node Specific management : Forwarding Logic of a sensor can be modified.
  - Network Specific Management
    - Forward all the traffic of a node.
    - Drop all the traffic of a node.

Software-Defined Sensor network Architecture

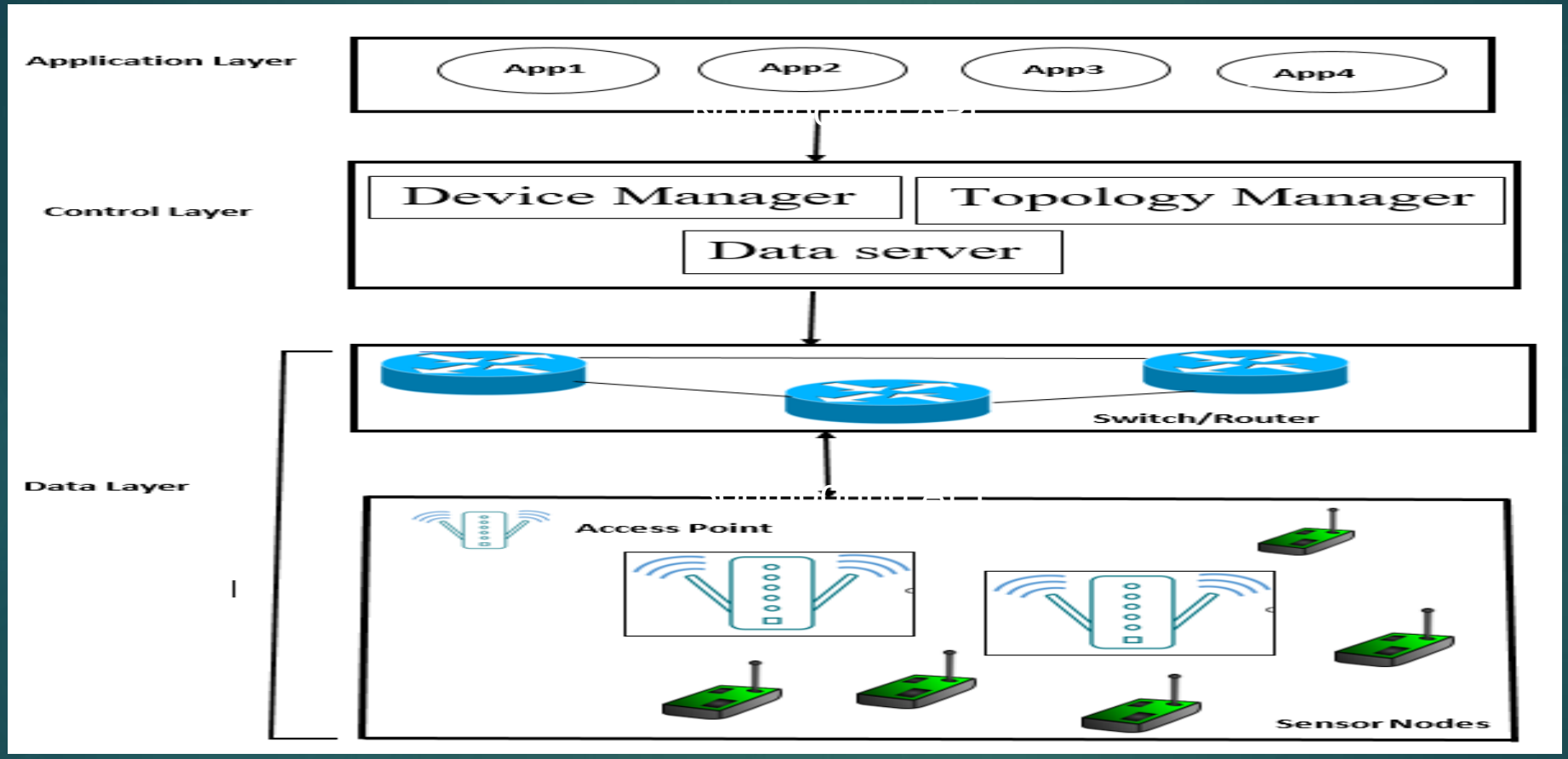
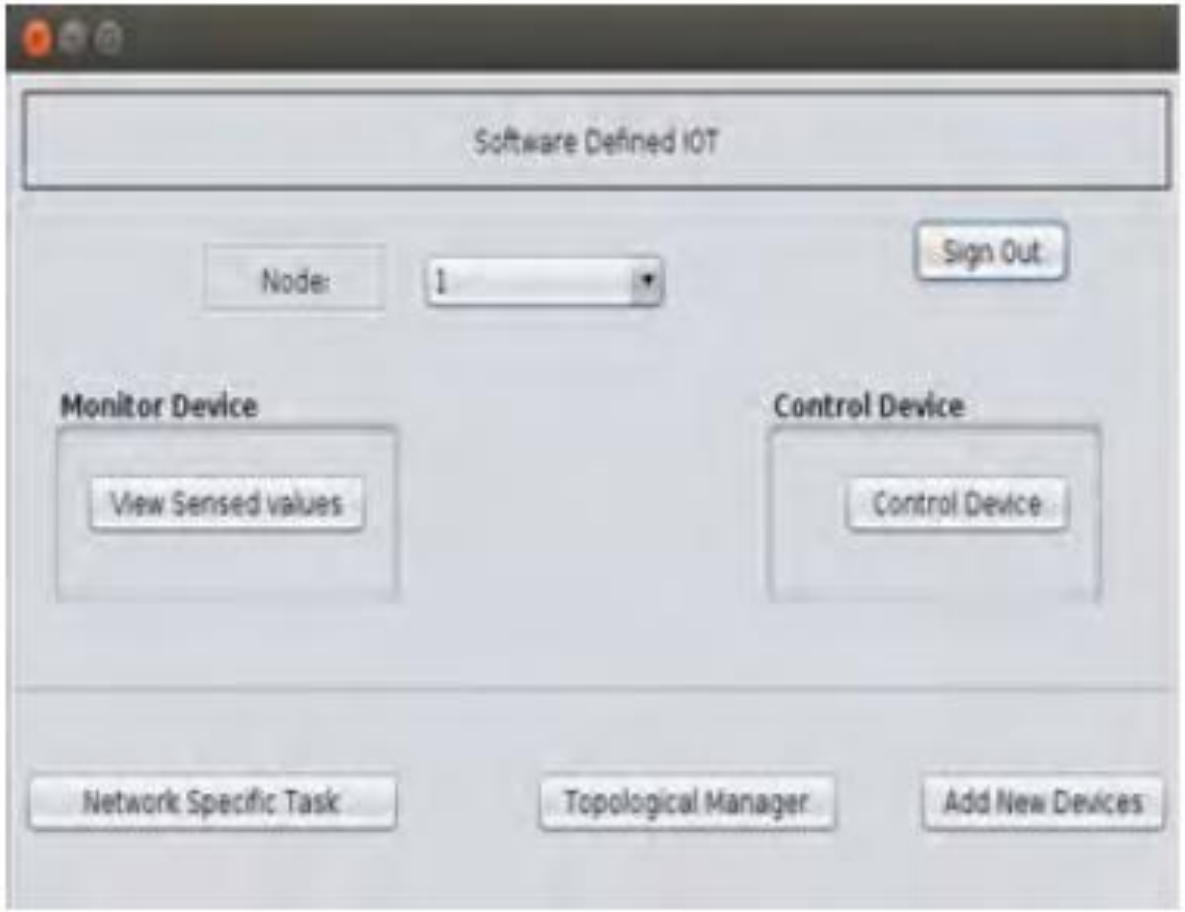
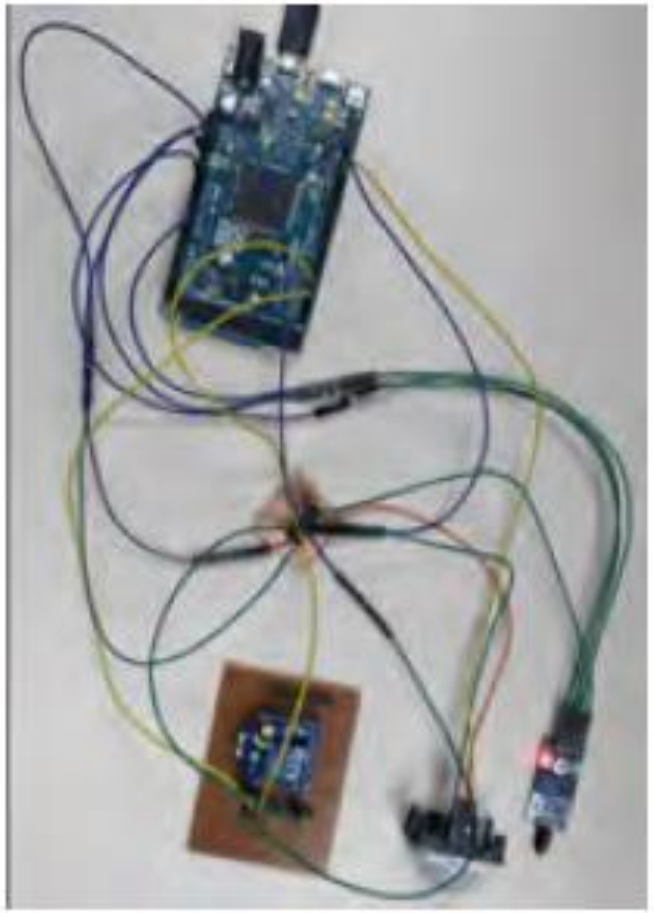


Fig: SDN-WSN Architecture

Schematic diagram of Developed Application





Results Achieved

1. Packet delivery ratio is larger in case of Soft-WSN due to intelligent routing by controller as compared to traditional WSN.

2. Energy Consumption is reduced in case of Soft-WSN due to central coordination as compared to traditional WSN. Due to which the lifetime of network also increases.

3. Message Overhead is reduced in case of Soft-WSN.

Sensor Open flow: Enabling SDN enabled wireless Network.

- WSN was thought to be of as application specific field.
- Policy changes because of changing business needs in WSN is a manual process.
- But if the applications are configurable then WSN will be versatile in nature.
- WSN is hard to manage because of distributed nature.
- The whole idea is to make data plane programmable by manipulating the flow table on each sensor node.
- Controller perform intelligent routing and QoS.

Luo et. al. Communication Letters' 12

Some Ideas to integrate SDN with IOT

- MQTT protocol is used in the various IOT devices.
- To simulate data from MQTT protocol Mosquitto is required to be installed on mininet wifi/raspberry pi which help communication between the IOT devices.
- IOT devices will connect to the SDN controller on mininet wifi using mosquitto broker.
- MQTT running on mininet wifi is sending the distance information to the smartphone.

Different Attacks in SDN

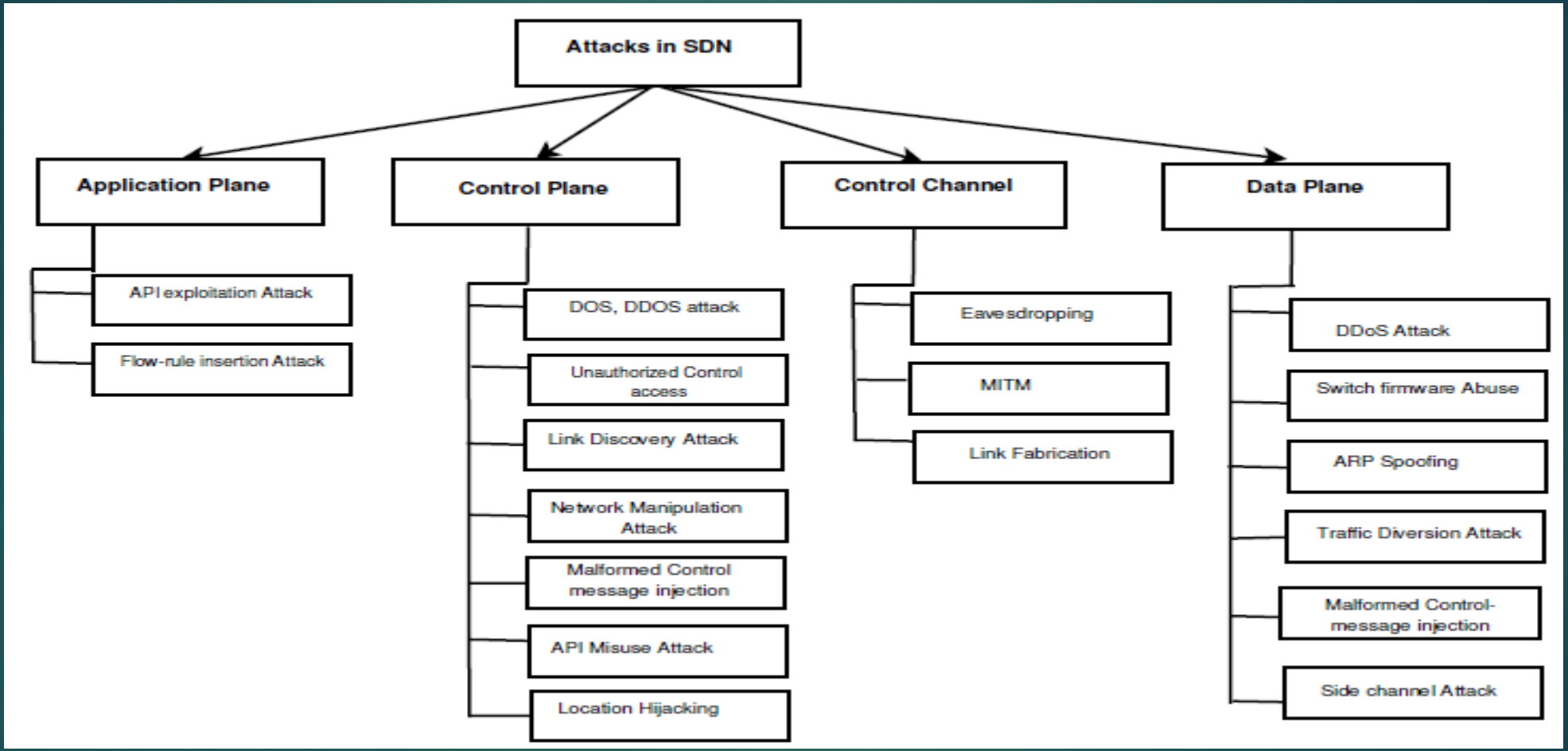


Fig : Attack Taxonomy in SDN

Different Attacks in SDN

There are various type of attacks possible in SDN. Some of them are:

1. Attacks at Data Plane Layer:

    a) Denial of Service (DoS) attack: Attacker attempts to make a network resource unavailable to its intended users.

    b) Spoofing &Tampering: Attacker successfully enter new flows by spoofing either the Northbound API or Southbound API [3].

    c) Traffic Diversion: This attack occurs to the network element at the data plane. This attack compromises a network element to divert the traffic [4].

Different Attacks in SDN

2. Attacks at Controller Layer:

a) Spoofing & Tampering attack: Attacker impersonate as controller and tamper with the flow table entries of different hosts [5].

b) Distributed Denial of Service (DDoS) attack: Attackers perform resource consumption attacks on the controller to bog it down and cannot cater to the benign traffic.

Different Attacks in SDN

3. Attacks at Control-Link Channel

a) Man-in-the-Middle Attack (MITM): Attacker maliciously steal the information exchange between two genuine users and eavesdrop the communication between them [6]. One of the way to perform MITM attack is through ARP poisoning.

b) Replay attacks: By accessing the timestamp field of the message, it may lead to attack in which benign traffic is maliciously delayed and is known as replay attack.

DDOS attack detection in SDN

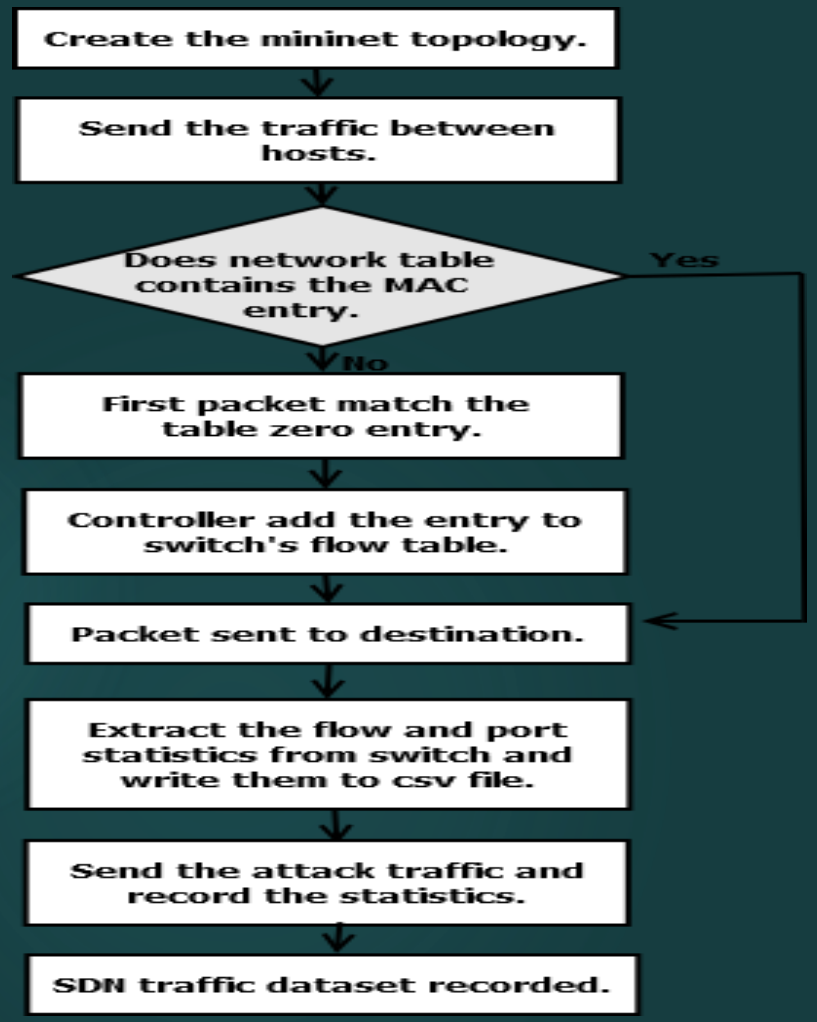


Fig : Flowchart of dataset Creation.

1. Implemented an application to record the various statistics of traffic running in the network.
2. Application collects the various flow statistics and port statistics from various switches.
3. Based on the collected statistics, other parameters are calculated (Number of Flows, Number of Packet_ins, Packet Rate, Average packet per flow, Average Byte per flow)
4. All are compared with a predefined threshold limit. If the parameters exceeds the limit, an attack is detected and thus the record is annotated as Attack else normal.

Ahuja, N., Singal, G., Mukhopadhyay, D., & Kumar, N. (2021). Automated DDOS attack detection in software defined networking.,Journal of Network and Computer Applications, 103108

DDOS attack detection in SDN

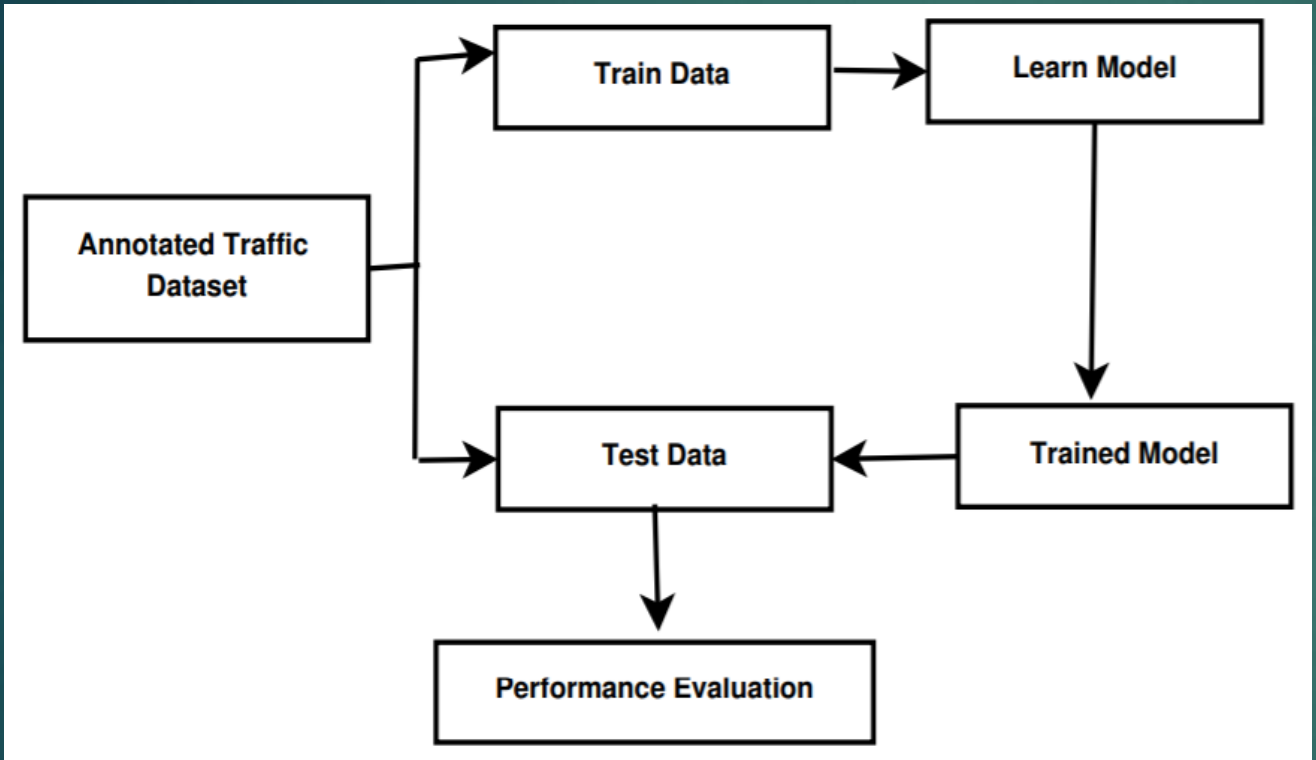


Fig : Machine Learning Algorithm applied to the dataset Created.

Dataset is created with 1,04,345 rows.

After the dataset is recorded, the various machine learning and deep learning[10] algorithms are trained and classification of traffic is done.

Ahuja, N., Singal, G., Mukhopadhyay, D., & Kumar, N. (2021). Automated DDOS attack detection in software defined networking. Journal of Network and Computer Applications, 103108

Taxonomy of Topology Attacks

Topology tampering [18] can be possible by following attacks:

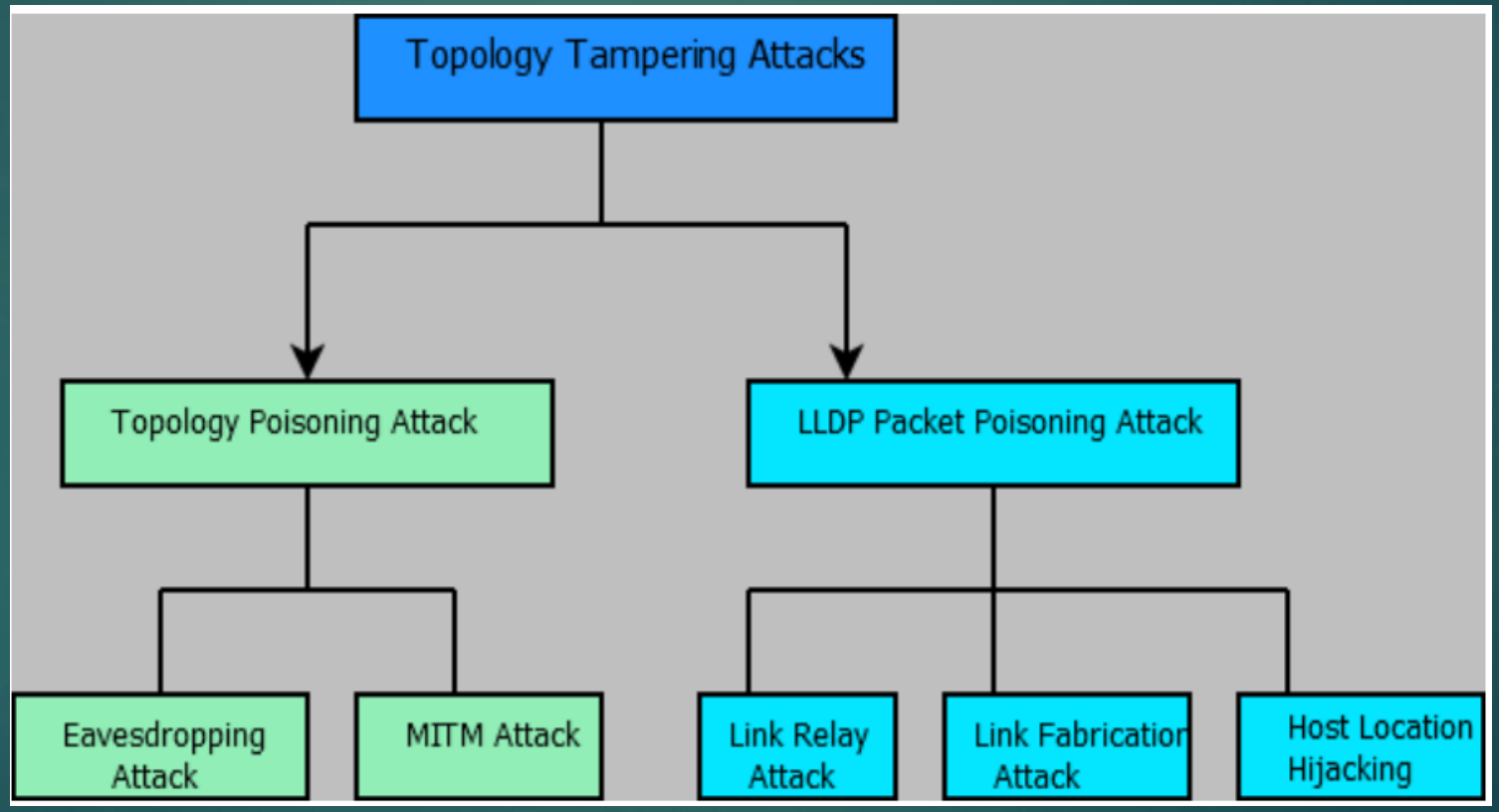


Fig : Topology attacks taxonomy.

Topology Discovery Process

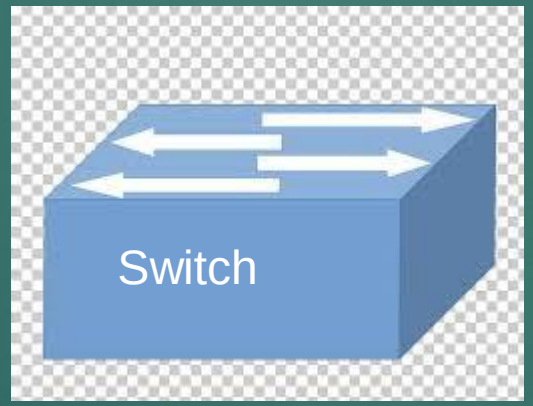
Controller

A's ARP Table

10.0.0.2:0:0:0:2

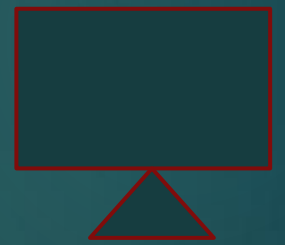
Host A

10.0.0.1- 00:00:00:01



B's ARP Table

10.0.0.1:0:0:0:1



Host B

10.0.0.2: 00:00:00:02

Fig: Concept of Topology poisoning attack.

- ARP request is broadcasted to all the remaining hosts in the topology.
- Destination host update the Source MAC in its ARP table and reply with ARP reply.
- When the Host A get the reply packet, it also updates its ARP table with Host B Mac address.
- In this way the ARP table of both the hosts get updated [17] and normal communication take place.

Topology Poisoning Attack

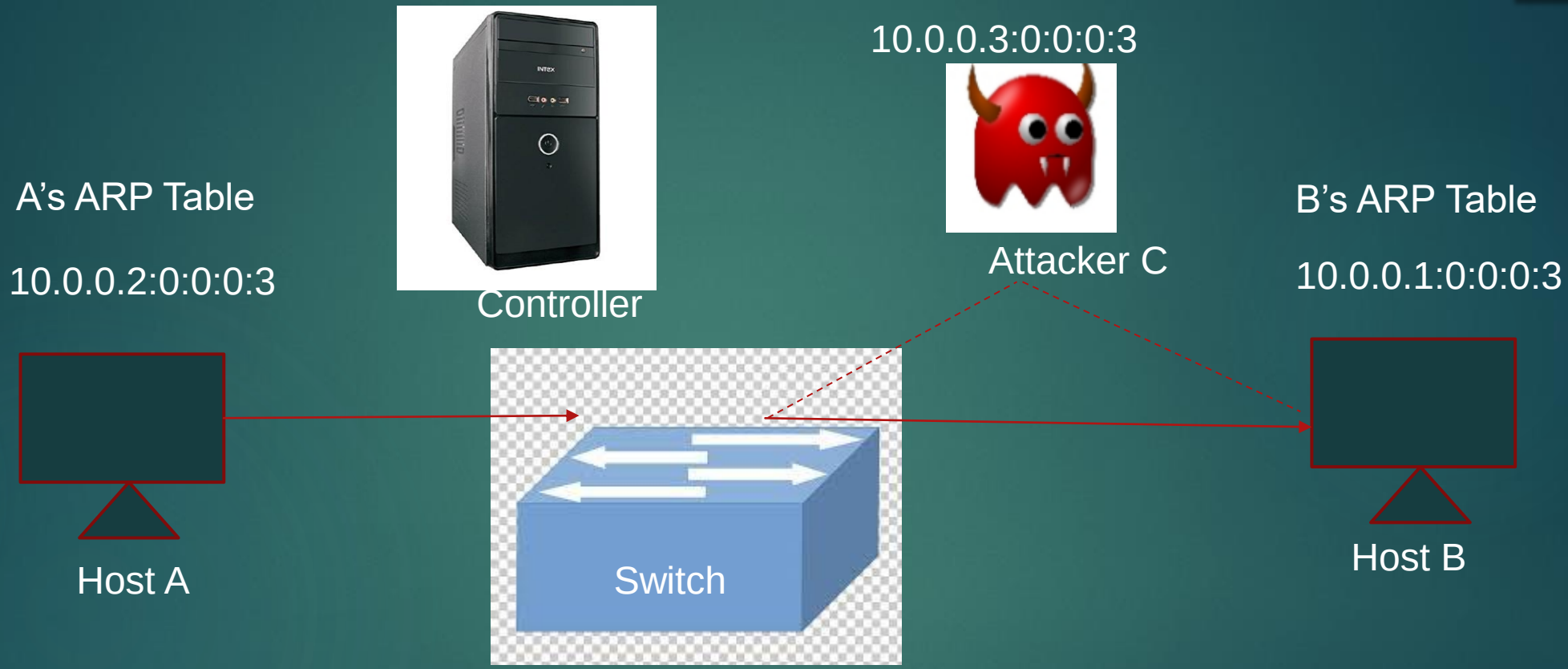


Fig: Concept of Topology poisoning, Flood attack.

- Attacker initiate the ARP request as "who is 10.0.0.1 tell 10.0.0.2" with some random MAC address.
- Attacker keeps on sending such requests from multiple MAC addresses and thus overflow the ARP table of genuine host with number of such requests.
- This way ARP table is flooded with random MAC addresses. Now if any genuine host want to communicate with Host A, it is not fulfilled as the ARP table cannot accommodate more entries.

Dataset Description

Table 3: Features Used in the created Dataset.

S.No.	*Features Used*
1	Source MAC Address at ethernet
2	Source MAC address at ARP header
3	Sender IP Address in ARP request.
4	Target IP Address in ARP request.
5	Protocol Code
6	Destination MAC Address at ARP
7	Destination MAC Address at Ethernet
8	Time to live (TTL)
9	Switch-ID.
10	Ping statistics.
11	in_port, out_port.
12	Operation Code.
13	Round trip time.
14	Packet loss
15	Number of Packet_in messages

Table 4: Message wise Traffic Analysis of the dataset

Traffic class	Benign	ARP Poison	ARP Flooding
ARP request	30570	13345	76186
ARP reply	5315	1535	138
ICMP request	4570	235	57
ICMP reply	1893	57	39

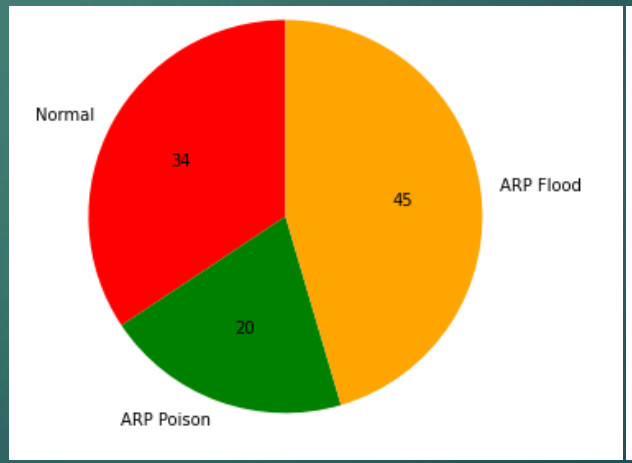


Fig : Traffic Analysis in the complete Dataset

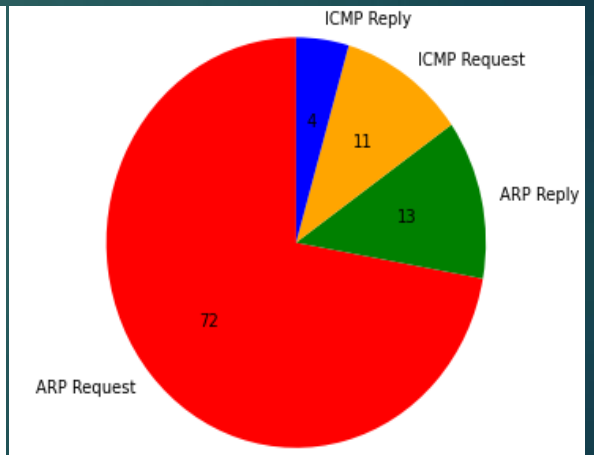


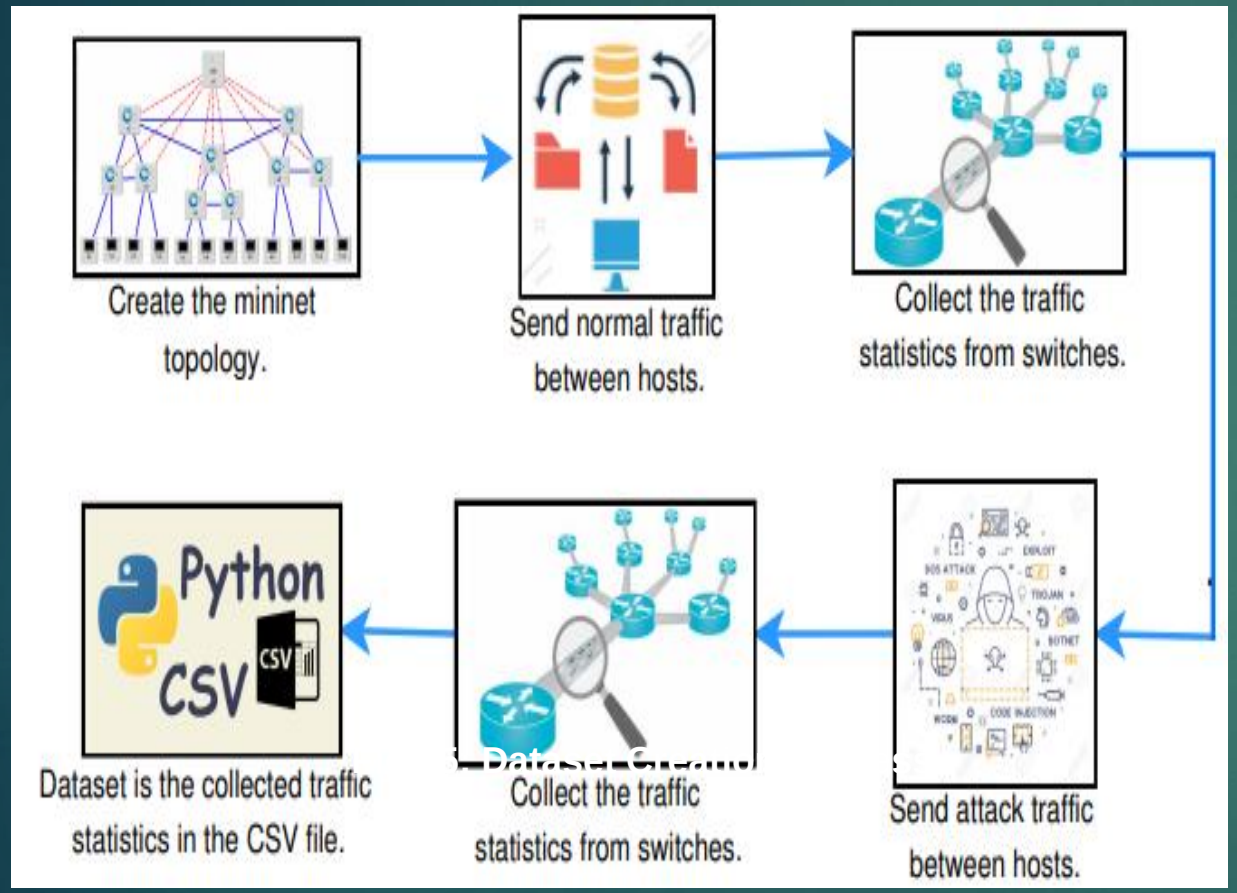
Fig : Message wise Traffic Analysis of ARP Poison Traffic.

Dataset Snapshot

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	switch	switch_	in_port	outport	src_mac_addr(eth	src_mac_addr(a	dst_mac_addr(eth	dst_mac_addr	src_ip(arp	dst_ip(arp	op_code(a	packet_in	Protocol	Pkt loss	rtt	ttl	Label
2	5	5	1	4.29E+09	00:00:00:00:00:07	00:00:00:00:00:0	ff:ff:ff:ff:ff:ff	00:00:00:00:00:	10.0.0.7	10.0.0.12	1	1707	0	0	0	64	0
3	2	2	3	4.29E+09	00:00:00:00:00:07	00:00:00:00:00:0	ff:ff:ff:ff:ff:ff	00:00:00:00:00:	10.0.0.7	10.0.0.12	1	1708	0	0	0	64	0
4	5	4	4	4.29E+09	00:00:00:00:00:07	00:00:00:00:00:0	ff:ff:ff:ff:ff:ff	00:00:00:00:00:	10.0.0.7	10.0.0.12	1	1709	0	0	0	64	0
5	4	3	4	4.29E+09	00:00:00:00:00:07	00:00:00:00:00:0	ff:ff:ff:ff:ff:ff	00:00:00:00:00:	10.0.0.7	10.0.0.12	1	1710	0	0	0	64	0
6	3	1	1	4.29E+09	00:00:00:00:00:07	00:00:00:00:00:0	ff:ff:ff:ff:ff:ff	00:00:00:00:00:	10.0.0.7	10.0.0.12	1	1711	0	0	0	64	0
7	1	6	4	4.29E+09	00:00:00:00:00:07	00:00:00:00:00:0	ff:ff:ff:ff:ff:ff	00:00:00:00:00:	10.0.0.7	10.0.0.12	1	1712	0	0	0	64	0
8	6	10	4	4.29E+09	00:00:00:00:00:07	00:00:00:00:00:0	ff:ff:ff:ff:ff:ff	00:00:00:00:00:	10.0.0.7	10.0.0.12	1	1713	0	0	0	64	0
9	10	8	4	4.29E+09	00:00:00:00:00:07	00:00:00:00:00:0	ff:ff:ff:ff:ff:ff	00:00:00:00:00:	10.0.0.7	10.0.0.12	1	1714	0	0	0	64	0
10	7	7	4	4.29E+09	00:00:00:00:00:07	00:00:00:00:00:0	ff:ff:ff:ff:ff:ff	00:00:00:00:00:	10.0.0.7	10.0.0.12	1	1715	0	0	0	64	0
11	11	9	4	4.29E+09	00:00:00:00:00:07	00:00:00:00:00:0	ff:ff:ff:ff:ff:ff	00:00:00:00:00:	10.0.0.7	10.0.0.12	1	1716	0	0	0	64	0
12	9	12	4	4.29E+09	00:00:00:00:00:07	00:00:00:00:00:0	ff:ff:ff:ff:ff:ff	00:00:00:00:00:	10.0.0.7	10.0.0.12	1	1717	0	0	0	64	0
13	5	11	4	4.29E+09	00:00:00:00:00:07	00:00:00:00:00:0	ff:ff:ff:ff:ff:ff	00:00:00:00:00:	10.0.0.7	10.0.0.12	1	1718	0	0	0	64	0
14	2	13	4	4.29E+09	00:00:00:00:00:07	00:00:00:00:00:0	ff:ff:ff:ff:ff:ff	00:00:00:00:00:	10.0.0.7	10.0.0.12	1	1719	0	0	0	64	0
15	3	7	3	4	00:00:00:00:00:0c	00:00:00:00:00:0	00:00:00:00:00:07	00:00:00:00:00:	10.0.0.12	10.0.0.7	2	1720	0	0	0	64	0
16	4	6	1	4	00:00:00:00:00:0c	00:00:00:00:00:0	00:00:00:00:00:07	00:00:00:00:00:	10.0.0.12	10.0.0.7	2	1721	0	0	0	64	0
17	1	1	2	1	00:00:00:00:00:0c	00:00:00:00:00:0	00:00:00:00:00:07	00:00:00:00:00:	10.0.0.12	10.0.0.7	2	1722	0	0	0	64	0
18	10	2	4	3	00:00:00:00:00:0c	00:00:00:00:00:0	00:00:00:00:00:07	00:00:00:00:00:	10.0.0.12	10.0.0.7	2	1725	0	0	0	64	0
19	6	5	4	1	00:00:00:00:00:0c	00:00:00:00:00:0	00:00:00:00:00:07	00:00:00:00:00:	10.0.0.12	10.0.0.7	2	1727	0	0	0	64	0
20	13	7	3	4	00:00:00:00:00:0c	00:00:00:00:00:0	00:00:00:00:00:07	00:00:00:00:00:	10.0.0.12	10.0.0.7	1	2180	0	0	0	64	0
21	11	6	1	4	00:00:00:00:00:0c	00:00:00:00:00:0	00:00:00:00:00:07	00:00:00:00:00:	10.0.0.12	10.0.0.7	1	2181	0	0	0	64	0
22	12	1	2	1	00:00:00:00:00:0c	00:00:00:00:00:0	00:00:00:00:00:07	00:00:00:00:00:	10.0.0.12	10.0.0.7	1	2182	0	0	0	64	0
23	9	2	4	3	00:00:00:00:00:0c	00:00:00:00:00:0	00:00:00:00:00:07	00:00:00:00:00:	10.0.0.12	10.0.0.7	1	2183	0	0	0	64	0

Fig: Dataset for ARP Poison and ARP Flood attack with 15 features.

ARP Poison Attack Detection



- **Fig: Dataset creation for ARP based attacks in SDN.**

Steps to create Dataset creation:

- A python application is created to extract the different features of the traffic at ARP and IP layer.
- ARP and IP related features are written into respective CSV files.
- Normal traffic and attack traffic is run and the features are extracted.
- Both the files are combined based on the common Date Time field and Dataset is created.

ARP Poison Attack Detection

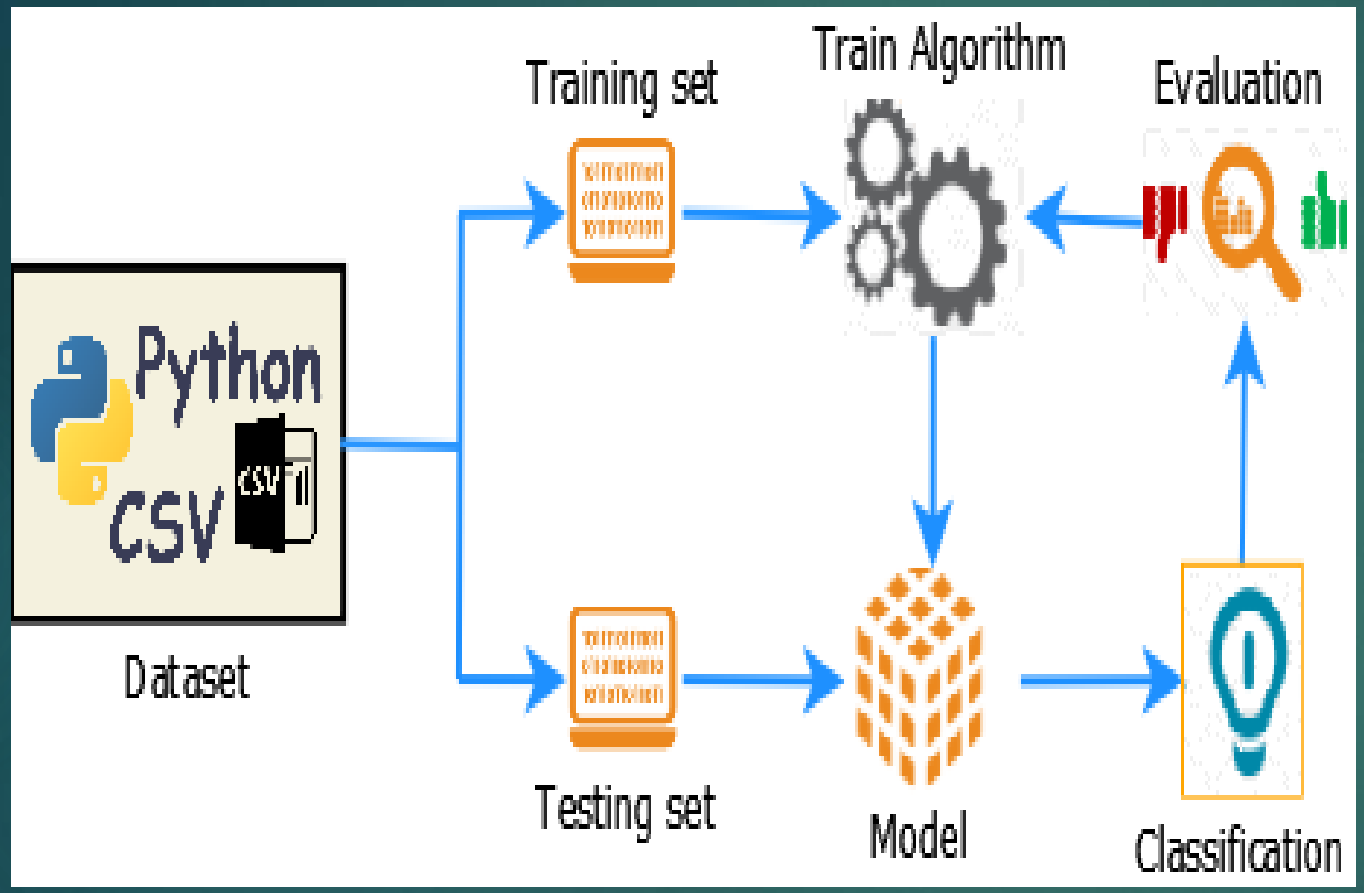


Fig: Traffic classification using Machine Learning

- A Dataset is created with 1,34,000 records in a CSV file.
- Different Machine Learning [12] and Deep Learning algorithms are applied for traffic classification.

Blockchain Introduction

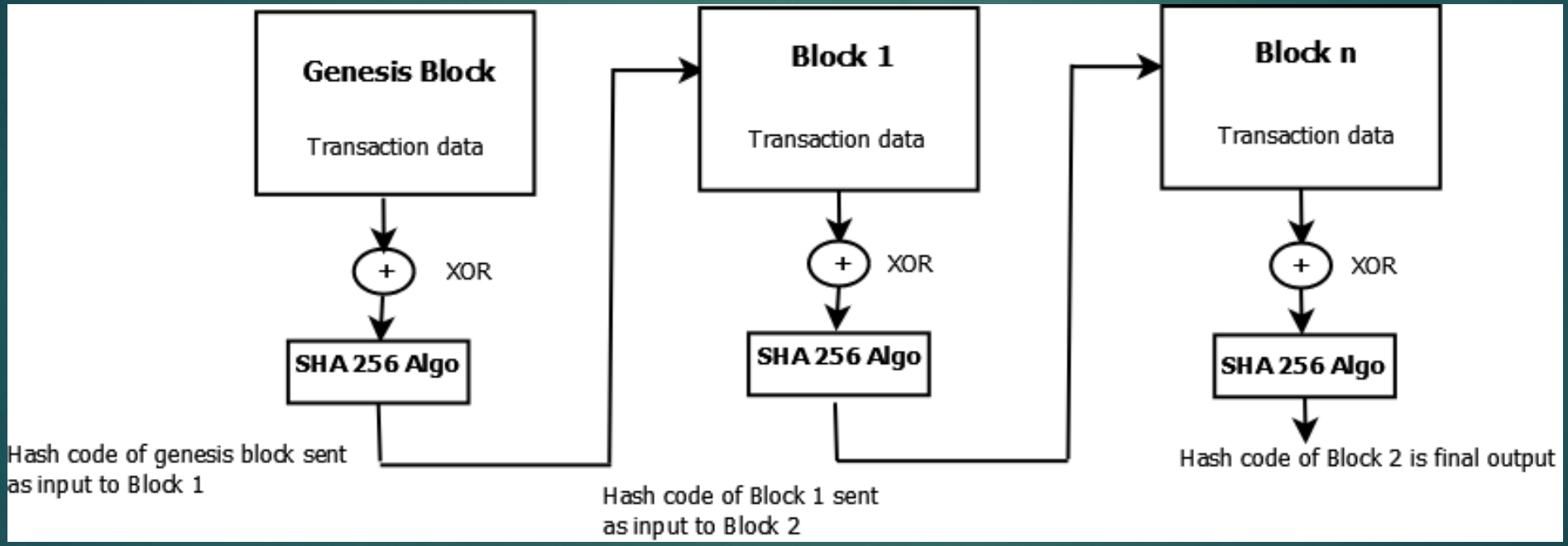


Fig: Diagram depicting the Concept of Blockchain

Blockchain Principles

- Trust: Network elements work upon updation of table entries together but each updating the entries together rather than each one separately.
- Integrity: To update or read the block data, one needs access rights which makes the blockchain an important security feature.
- Resiliency: In Blockchain every block has its importance since the information is stored in all the blocks.
- Privacy: Blockchain permanently stores the data of older generated blocks to maintain records of all the transactions. This way the new blocks can be checked for their consistency and authenticity by looking into the previous records.

ARP Poisoning Attack

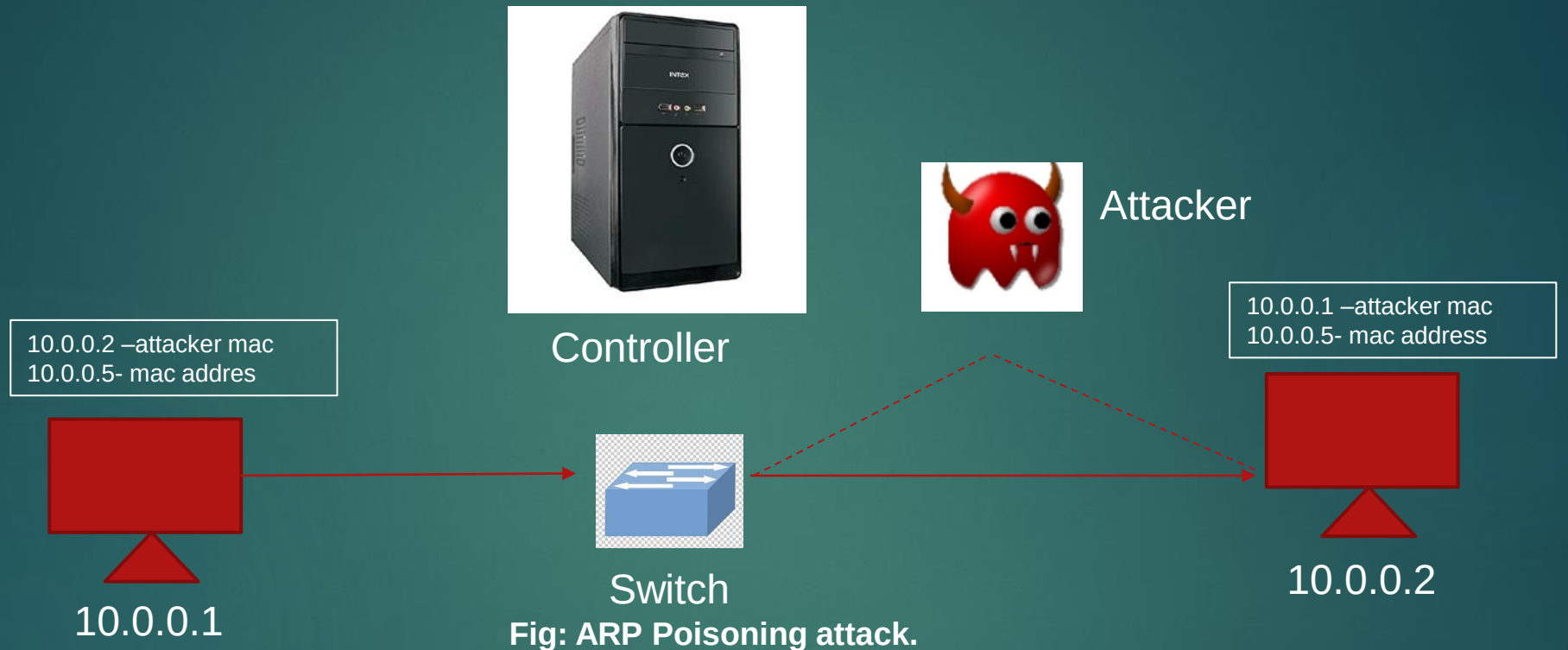


Fig: ARP Poisoning attack.

1. Attacker initiate the ARP request as"who is 10.0.0.1 tell 10.0.0.2" with MAC address as attacker Mac address and also "who is 10.0.0.2 tell 10.0.0.1" with MAC address of attacker.
2. Host A and B update their ARP table with 10.0.0.2 with MAC address of attacker and Host B with 10.0.0.1 with MAC address of attacker.
3. This way ARP table is poisoned. Now any communication between Host A and B is going through attacker C.

Blockchain Perspective

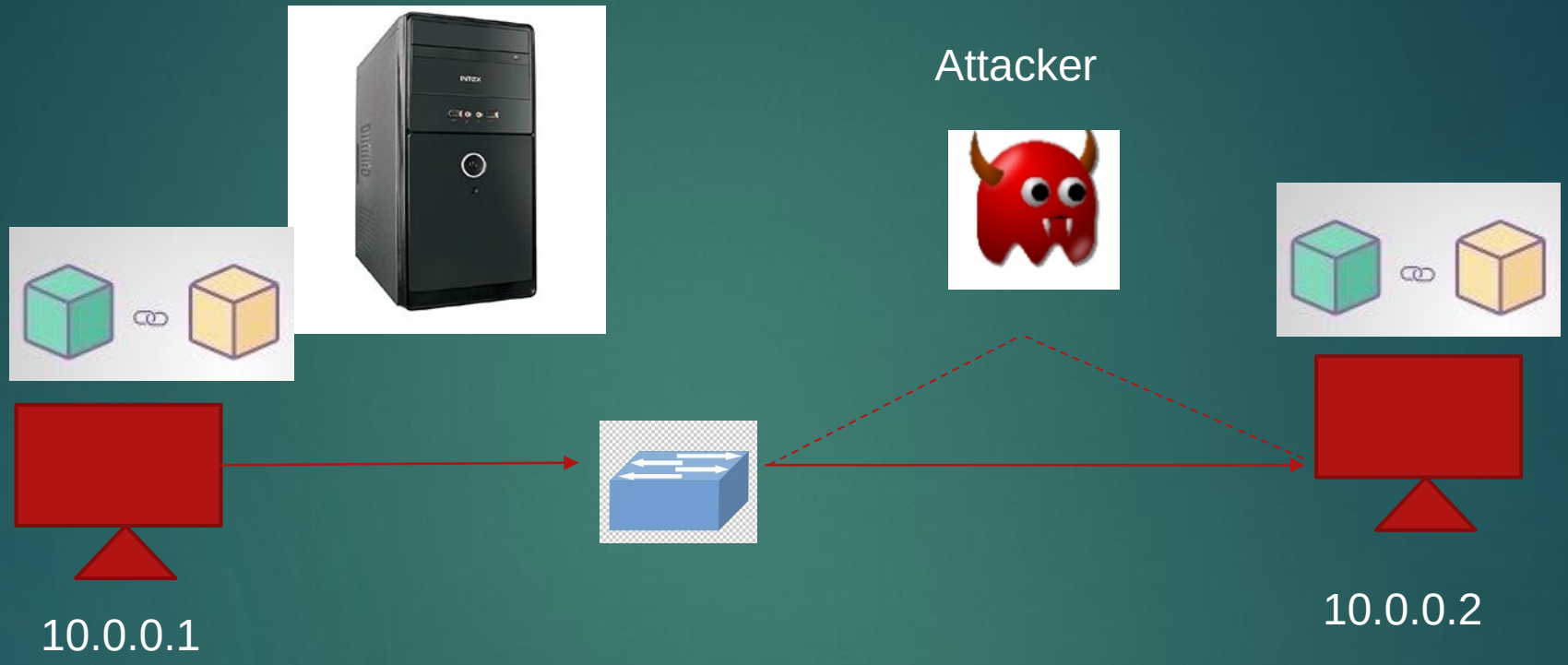


Fig: Blockchain for ARP Poisoning attack.

1. Before updating the ARP tables this arp request block is sent to every host in the network.
2. Host 1 will reject the block as its IP/MAC pairing is wrongly reflected in the ARP request, so the block will not be validated and hence the transaction got rejected and thus ARP poisoning will not take place.

Proposed Architecture

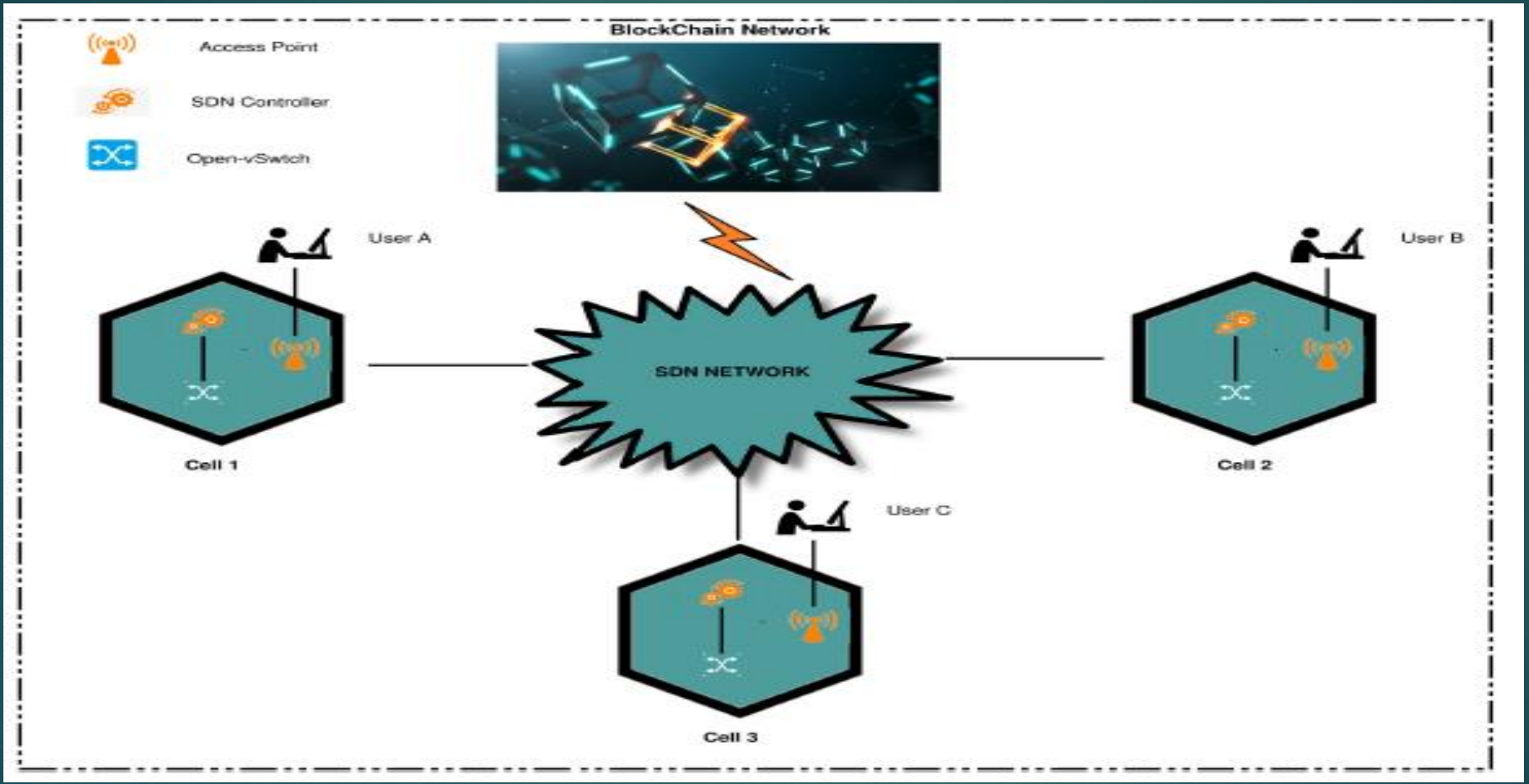


Fig: Blockchian Architecture for SDN.

Conclusion

- SDN network has redefined the networking domain by making it programmable and the various network devices can be programmed remotely supporting the dynamic configuration of the network.
- The programming capability of the network can be employed to solve the various issues which arise in traditional network, IOT network.
- In the past, different Statistical and Cryptographic techniques have been used to detect the attacks in network.
- Different statistical techniques which are applied are found to be less competitive in comparison with Machine Learning approach.
- Blockchain can be integrated with SDN to make SDN more secure

References

1. Anand, N., Sarath Babu, and B. S. Manoj. "On detecting compromised controller in software defined networks." *Computer Networks* 137 (2018): 107-118.

2. Maimó, L. F., Gómez, A. L. P., Clemente, F. J. G., & Pérez, M. G. A Self-Adaptive Deep Learning-Based System for Anomaly Detection in 5G Networks.

3. Latah, Majd, and Levent Toker. "Towards an Efficient Anomaly-Based Intrusion Detection for Software-Defined Networks." *arXiv preprint arXiv:1803.06762* (2018).

4. Aldwairi, Tamer, Dilina Perera, and Mark A. Novotny. "An evaluation of the performance of Restricted Boltzmann Machines as a model for anomaly network intrusion detection." *Computer Networks* 144 (2018): 111-119.

5. AlEroud, Ahmed, and Izzat Alsmadi. "Identifying cyber-attacks on software defined networks: An inference-based intrusion detection approach." *Journal of Network and Computer Applications* 80 (2017): 152-164.

References

6. Liu, Hongyu, et al. "CNN and RNN based payload classification methods for attack detection." Knowledge-Based Systems 163 (2019): 332-341.

7. Dabbagh, Mehiar, et al. "Software-defined networking security: pros and cons." IEEE Communications Magazine 53.6 (2015): 73-79.

8. Tang, Tuan A., et al. "Deep learning approach for network intrusion detection in software defined networking." Wireless Networks and Mobile Communications (WINCOM), 2016 International Conference on. IEEE, 2016.

9. Mohammadi, Reza, Reza Javidan, and Mauro Conti. "Slicots: An sdn-based lightweight countermeasure for tcp syn flooding attacks." IEEE Transactions on Network and Service Management 14.2 (2017): 487-497.

10. Yoon, Changhoon, et al. "Flow wars: Systemizing the attack surface and defenses in software-defined networks." IEEE/ACM Transactions on Networking 6 (2017): 3514-3530.

References

11. Maninderpal Singh, Gagangeet Singh Singh Aujla, Amritpal Singh, Neeraj Kumar, and Sahil Garg. Deep learning based blockchain framework for secure software defined industrial networks. IEEE Transactions on Industrial Informatics, 2020a.

12. Nehra, A., Tripathi, M., & Gaur, M. S. FICUR: Employing SDN programmability to secure ARP. In 2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC) (pp. 1-8). IEEE.

13. Sebbar, A., Zkik, K., Boulmalf, M., & El Kettani, M. D. E. C. (2019). New context-based node acceptance CBNA framework for MitM detection in SDN Architecture. Procedia Computer Science, 160, 825-830.137.

14. Alharbi, T. (2020). Deployment of blockchain technology in software defined networks: A survey. IEEE Access, 8, 9146-9156.

15. Wenjuan Li, Weizhi Meng, Zhiqiang Liu, and Man-Ho Au.Towards blockchain-based software-defined networking: security challenges and solutions. IEICE Transactions on Information and Systems, 103(2):196–203, 2020.

References

16. M. Brooks, B. Yang, A man-in-the-middle attack against opendaylight sdn controller, in: Proceedings of the 4th Annual ACM Conference on Research in Information Technology, 2015, pp. 45–49.

17. S. Hong, L. Xu, H. Wang, G. Gu, Poisoning network visibility in software-defined networks: New attacks and countermeasures., in: NDSS, Vol. 15, 2015, pp. 8–11.

18. S. Y. Nam, D. Kim, J. Kim, Enhanced arp: preventing arp poisoning-based man-in-the-middle attacks, IEEE communications letters 14 (2) (2010) 187–189.

19. Zakaria Abou El Houda, Abdelhakim Senhaji Hafid, and Lyes Khoukhi. Cochain-sc: An intra-and inter-domain ddos mitigation scheme based on blockchain using sdn and smart contract. IEEE Access, 7:98893–98907, 2019.

20. Durbadal Chattaraj, Sourav Saha, Basudeb Bera, and Ashok Kumar Das. On the design of blockchain-based access control scheme for software defined networks. In IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pages 237–242. IEEE, 2020.

THANKS

Any Queries??