Name: Md.Sohag Hossain

ID : IT-17060

Assignment no: 02

Name of the assignment: OpenFlow Control

Objectives:

- ❖ Understand the working principles of OpenFlow protocol.
- ❖ Configure a basic Software Defined Network for end-to-end communications.
- ❖ Understand the difference between interacting with real and virtual networks.

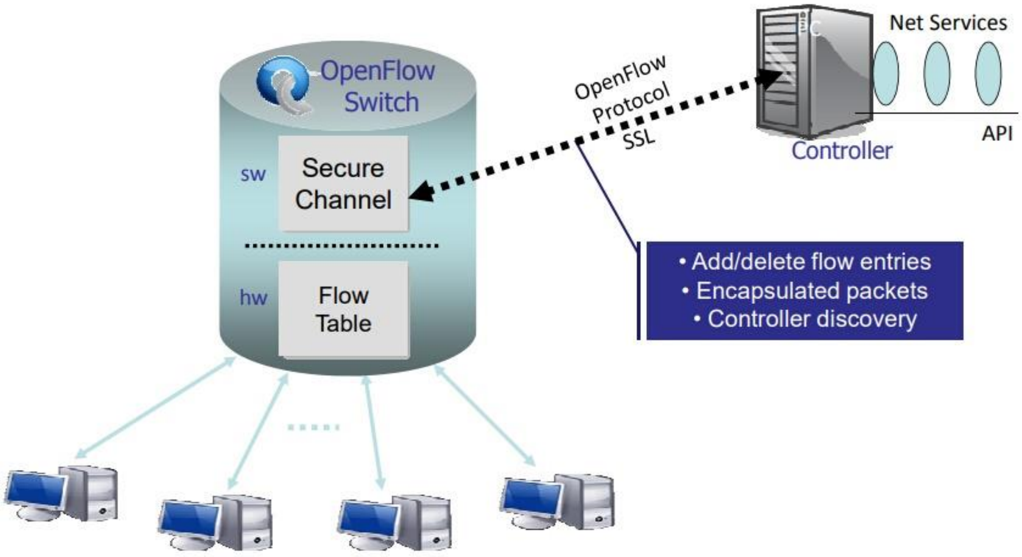
Theory:

OpenFlow Protocol:

OpenFlow protocol defines the communication between an OpenFlow controller and an OpenFlow switch. This protocol is what most uniquely identifies OpenFlow technology. At its essence, the protocol consists of a set of messages that are sent from the controller to the switch and a corresponding set of messages that are sent in the opposite direction. The messages, collectively, allow the controller to program the switch so as to allow fine-grained control over the switching of user traffic. The most basic programming defines, modifies and deletes flows

OpenFlow Switch:

OpenFlow is a programmable network protocol for SDN environment, which is used for communication between OpenFlow switches and controllers. OpenFlow separates the programming of network device from underlying hardware, and offers a standardized way of delivering a centralized, programmable network that can quickly adapt to changing network requirements.



An OpenFlow switch is an OpenFlow-enabled data switch that communicates over OpenFlow channel to an external controller. It performs packet lookup and forwarding according to one or more flow tables and a group table. The OpenFlow switch communicates with the controller and the controller manages the switch via the OpenFlow switch protocol. They are either based on the OpenFlow protocol or compatible with it.

Switch Components:

An OpenFlow Switch consists of one or more flow tables and a group table, which perform

packet lookups and forwarding, and an OpenFlow channel to an external controller (Figure 2-1).

The switch communicates with the controller and the controller manages the switch via the

OpenFlow protocol.

Using the OpenFlow protocol, the controller can add, update, and delete flow entries in flow tables, both reactively (in response to packets) and proactively. Each flow table in the switch

contains a set of flow entries; each flow entry consists of match fields, counters, and a set of

instructions to apply to matching packets.

Matching starts at the first flow table and may continue to additional flow tables. Flow entries

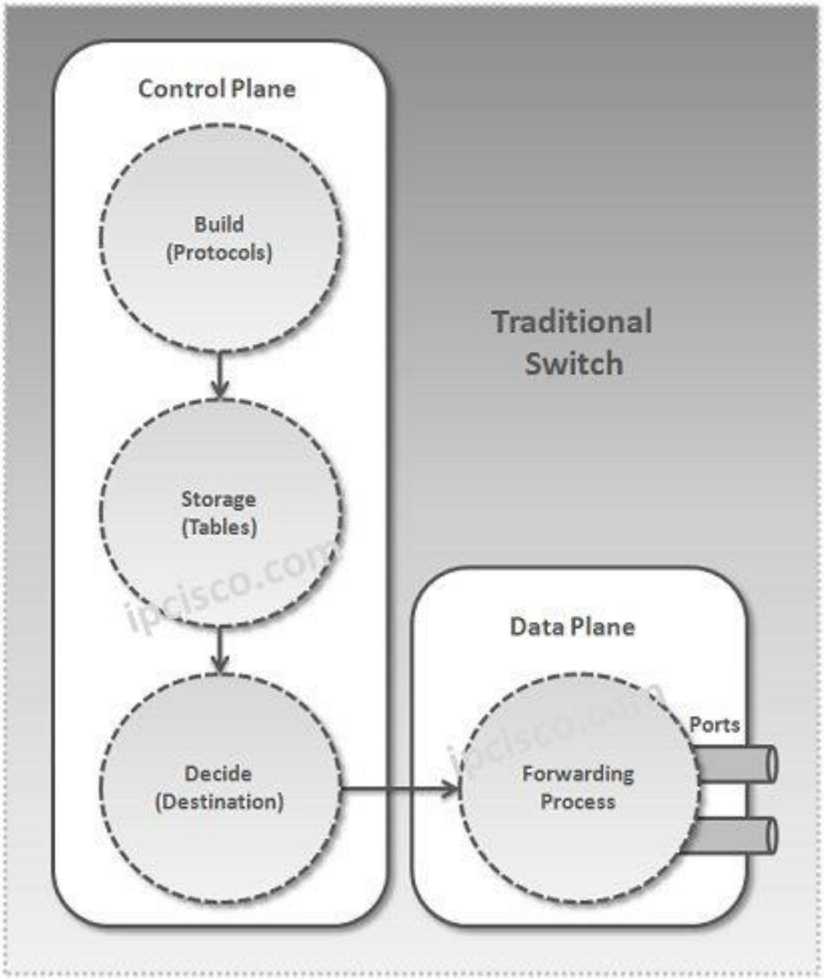
match packets in priority order, with the first matching entry in each table being used. If a

matching entry is found, the instructions associated with the specific flow entry are executed. If

no match is found in a flow table, the outcome depends on configuration of the table-miss flow

entry: for example, the packet may be forwarded to the controller over the OpenFlow channel,

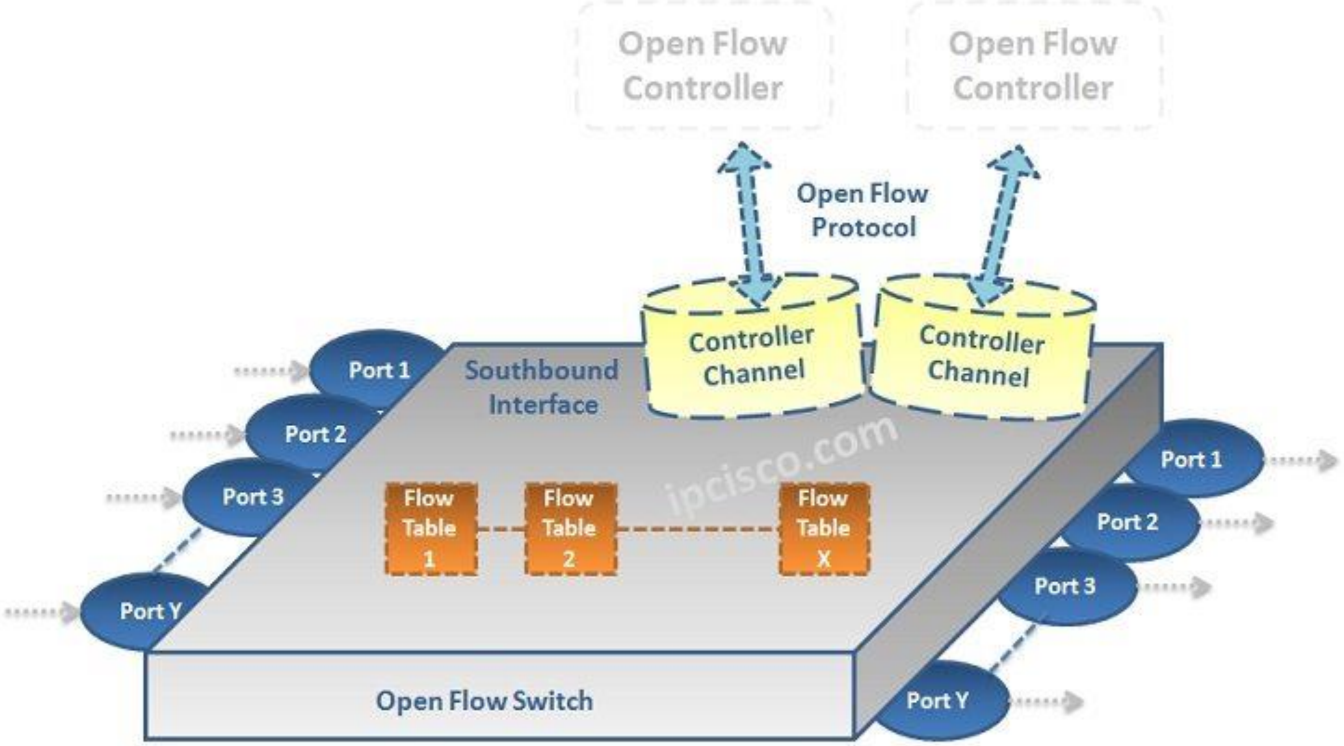
dropped, or may continue to the next flow table.



Open Flow Switches consist of some key elements. These key elements has special duties in Open Flow Switches. What are these Open Flow Switch elements? These are :

- Flow Tables
- Secure Channel

- Open Flow Protocol



OpenFlow specification terms:

This section describes key OpenFlow specification terms:

- **Byte**: an 8-bit octet.

- **Packet:** an Ethernet frame, including header and payload.

- **Port:** where packets enter and exit the OpenFlow pipeline. May be a physical port, a logical port defined by the switch, or a reserved port defined by the OpenFlow protocol.

 Pipeline: the set of linked flow tables that provide matching, forwarding, and packet

modifications in an OpenFlow switch.

 Flow Table: A stage of the pipeline, contains flow entries.

 Flow Entry: an element in a flow table used to match and process packets. It contains a

set of match fields for matching packets, a priority for matching precedence, a set of

counters to track packets, and a set of instructions to apply.

 Match Field: a field against which a packet is matched, including packet headers, the

ingress port, and the metadata value. A match field may be wildcarded (match any value)

and in some cases bitmasked.

 Metadata: a maskable register value that is used to carry information from one table to

the next.

 Instruction: Instructions are attached to a flow entry and describe the OpenFlow

processing that happen when a packet matches the flow entry. An instruction either

modifies pipeline processing, such as direct the packet to another flow table, or contains a

set of actions to add to the action set, or contains a list of actions to apply immediately to

the packet.

 Action: an operation that forwards the packet to a port or modifies the packet, such as

decrementing the TTL field. Actions may be specified as part of the instruction set

associated with a flow entry or in an action bucket associated with a group entry. Actions

may be accumulated in the Action Set of the packet or applied immediately to the packet.

 Action Set: a set of actions associated with the packet that are accumulated while the packet is processed by each table and that are executed when the instruction set instructs

the packet to exit the processing pipeline.

- **Group:** a list of action buckets and some means of choosing one or more of those

buckets to apply on a per-packet basis.

- **Action Bucket:** a set of actions and associated parameters, defined for groups.

- **Tag:** a header that can be inserted or removed from a packet via push and pop actions.

- **Outermost Tag:** the tag that appears closest to the beginning of a packet.

- **Controller:** An entity interacting with the OpenFlow switches using the OpenFlow

protocol.

- **Meter:** a switch element that can measure and control the rate of packets. The meter

trigger a meter band if the packet rate or byte rate passing through the meter exceed a

predefined threshold. If the meter band drops the packet, it is called a Rate Limiter.

Traditional Switching Hub:

Short for *port-switching hub,* a special type of hub that forwards packets to the appropriate port based on the packet's address. Conventional hubs simply rebroadcast every packet to every port. Since switching hubs forward each packet only to the required port, they provide much better performance. Most switching hubs also support load balancing, so that ports are dynamically reassigned to different LAN segmentsbased on traffic patterns.

Some newer switching hubs support both traditional Ethernet (10 Mbps) and Fast Ethernet (100 Mbps) ports. This enables the administrator to establish a dedicated, Fast Ethernet channel for high-traffic devices such as servers.

Switching Hub by OpenFlow:

OpenFlow switches can perform the following by receiving instructions from OpenFlow controllers such as Ryu.

- Rewrites the address of received packets or transfers the packets from the specified port.
- Transfers the received packets to the controller (Packet-In).
- Transfers the packets forwarded by the controller from the specified port (Packet-Out).

It is possible to achieve a switching hub having those functions combined.

How the RYU GUI interface can be improved? Provide some ideas.

Ans:

- ❖ Provide an interface to add or remove flow.
- ❖ Provide the configuration of the switches, ports, IP address Vlans.

Explain at least two the advantage and disadvantage of using mininet or Zodiac FX?

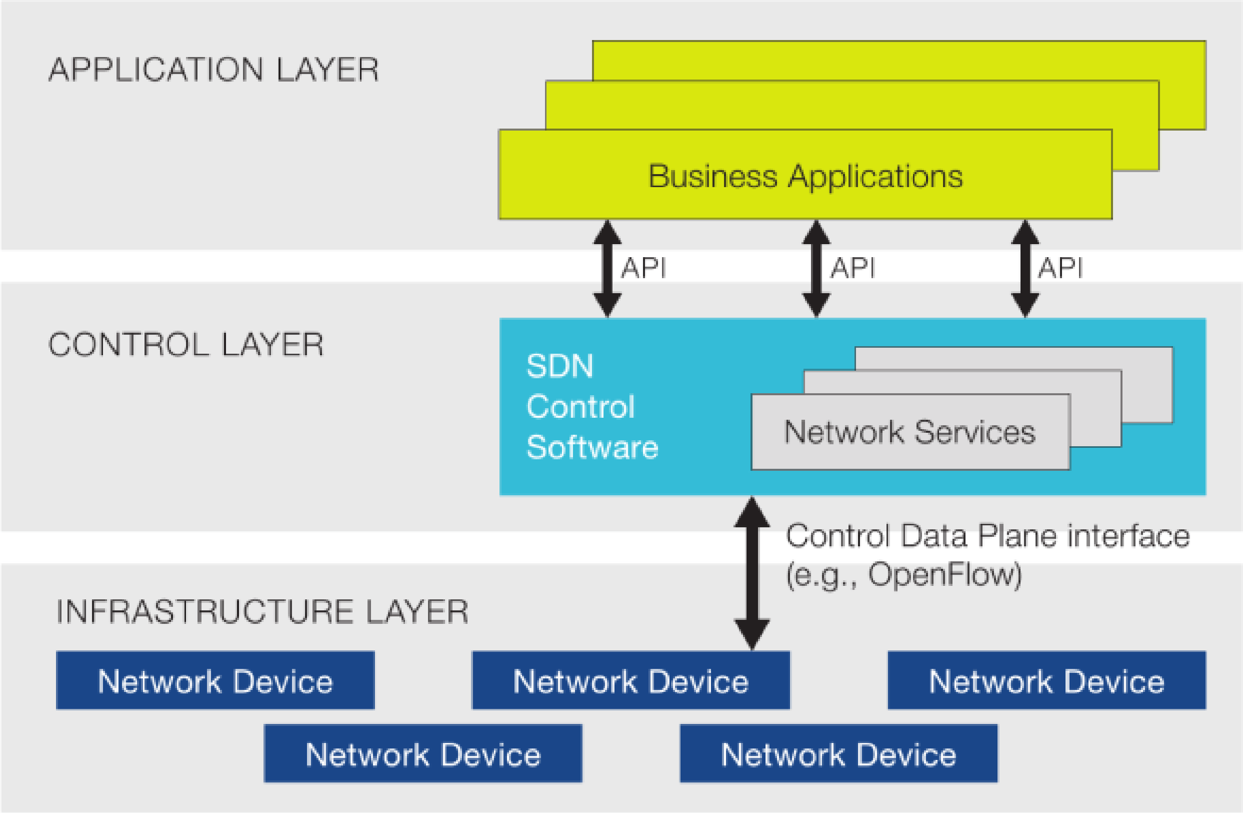
Ans:

Advantages:

- ✓ Scalability and flexibility in terms of topology
- ✓ Real hardware provide a better learning experience Disadvantages:
- ➢ Difficult to visualize conflicts
- ➢ Limited number or physical ports

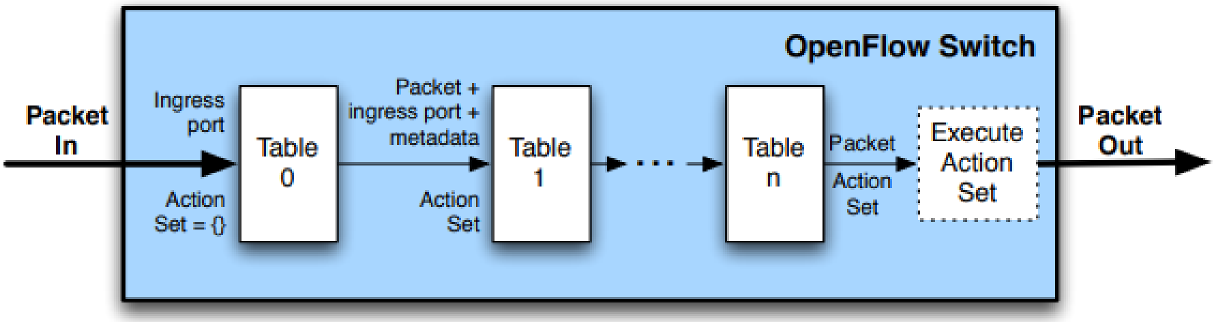
Explain how the open flow tables are created?

OpenFlow is an open standard for a communications protocol that enables the control plane to break off and interact with the forwarding plane of multiple devices from some central point, decoupling roles for higher functionality and programmability.



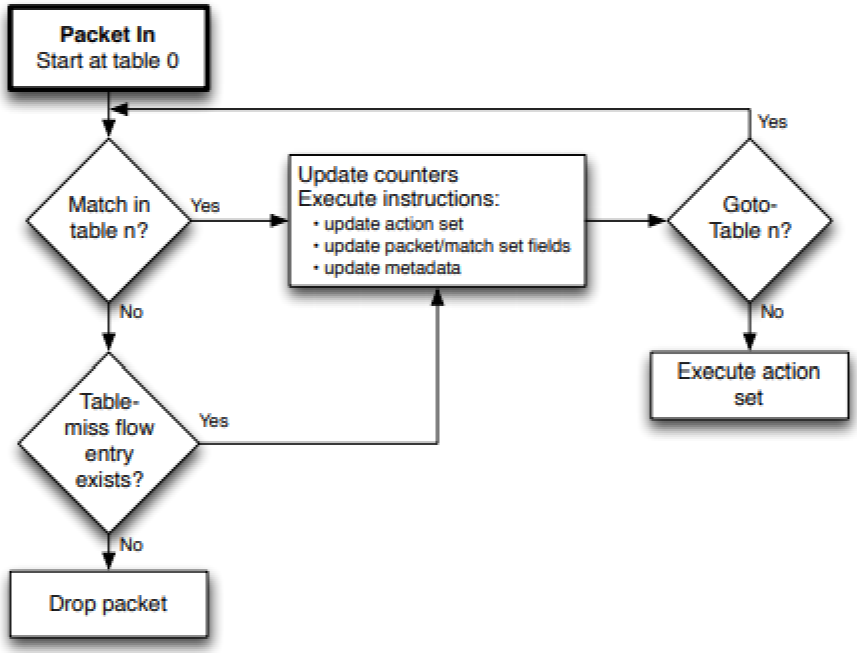
OpenFlow Tables:

This section describes the components of flow tables and group tables, along with the mechanics of matching and action handling. 12 © 2013; The Open Networking Foundation OpenFlow Switch Specification Version 1.3.2 Table 0 Table 1 Table n Packet Execute Action Set Packet In Action Set Action Set = {} OpenFlow Switch Packet Out ... Ingress port Packet + ingress port + metadata Action Set (a) Packets are matched against multiple tables in the pipeline



(a) Packets are matched against multiple tables in the pipeline

Matching:



Figure˜3: Flowchart detailing packet flow through an OpenFlow switch.

Conclusion:

It takes a lot time to complete this assignment. I have given lot of efforts to complete this. But can not do the complete assignment. I am not able to install ryu graphical interface controller. But I am able to learn about OpenFlow, openflow Protocol, openflow component ,mininet and many others.