Course Title: Telecommunication Engineering
Assignment Name: Controller REST API
Assignment No: 03
Name: Shuvo Biswas
ID: IT-16014

01.Objectives:

The objective of the assignment 3 is to:

- Understand the working principles of Controller Rest API.
- Understand the difference between proactive and reactiveinstallationflows.

02.Theory:-

OpenFlow: **Reactive versus Proactive**

OpenFlow is still the only one wire protocol that has a reasonably good chance at becoming the de-facto open SDN southbound messaging standard. When using OpenFlow to populate tables in switches there are essentially three modes of operation:

 Reactive Flow Instantiation:

When a new flow comes into the switch, the OpenFlow agent software on the switch does a lookup in the flow tables. If no match for the flow is found, the switch creates an OFP packet-in packet and sends it off to the controller for instructions.

Reactive mode reacts to traffic, consults the OpenFlow controller and creates a rule in the flow table based on the instruction.

 Proactive Flow Instantiation:

Rather than reacting to a packet, an OpenFlow controller could populate the flow tables ahead of time for all traffic matches that could come into the switch. By pre-defining all of the flows and actions ahead of time in the switches flow tables, the packet-in event never occurs. The result is all packets are forwarded at line rate. Proactive OpenFlow flow tables eliminate any latency induced by consulting a controller on every flow.

 Hybrid flow instantiation: A combination of both would allow for flexibility of reactive for particular sets a granular traffic control that while still preserving low-latency forwarding for the rest of the traffic.

Controller: REST API

 Application program interface (API) is an interface presented by software (such as a network operating system) that provides the capability to collect information from or
make a change to an underlying set of resources.

 APIs in the context of SDN: In an open SDN model, a common interface discussed is the northbound interface (NBI). The NBI is the interface between software applications, such as operational support systems, and a centralized SDN controller. One of the
common API technologies used at the northbound interface is the Representational State Transfer (REST) API. REST APIs use the HTTP/HTTPS protocol to execute common operations on resources represented by Uniform Resource Identifier (URI) strings. An application may use REST APIs to send an HTTP/HTTPS GET message via an SDN controller's IP address. That message would contain a URI string referencing the relevant network device and comprising an HTTP payload with a JSON header that has the proper parameters for a particular interface and statistic.

 Datapath Identifier of Openflow Switch: Each OpenFlow instance on a switch is identified by a Datapath Identifier. This is a 64 bit number determined as follows according to the OpenFlow specification: “The datapath_id fleld uniquely identifies a datapath. The lower 48 bits are intended for the switch MAC address, while the top 16 bits are up to the implementer. An example use of the top 16 bits would be a VLAN ID to distinguish multiple virtual switch instances on a single physical switch.”

Some Basic Questions:

Question 5.1: Explain the advantages of REST API of the Controller.

Answer to The Question No-5.1:

There are various types of advantages of REST API of the Controller. These are given below:

- ✓ As stated above it is very easy to learn, and understand. Its simplicity certainly becomes one of its primary advantages.
- ✓ With the help of REST API, one also will be able to organize complicated applications into a very easy to use resource.
- ✓ Another big advantage that comes in with REST API is the High load can be managed with the help of HTTP proxy server and cache.
- ✓ REST API is cleaner and very easy to explore and discover.
- ✓ It also makes it very simple for new clients to work upon other's applications whether it has been designed specifically for the respective purpose or not.
- ✓ Offers ubiquitous allowance with the use of standard HTTP procedure call-outs to requests and retrieve data.
- ✓ As REST API significantly depends upon codes, one can use it to synchronize data with the websites without any kind of complications.
- ✓ It also brings in a flexible approach and formats by serializing data in JSON or XML format.
- ✓ With the help of this, User can avail access to the same standard objects and data model as compared to SOAP-based Web services.
- ✓ It also brings in standards-based protection with the use of OAuth protocol for verifying your REST requests.

Question 5.2: What is the difference between SDN and openflow?

Answer to The Question No-5.2:

SDN:

- ✓ Software Defined Networking(SDN) is a networking paradigm which proposes separation of control plane from the data plane.

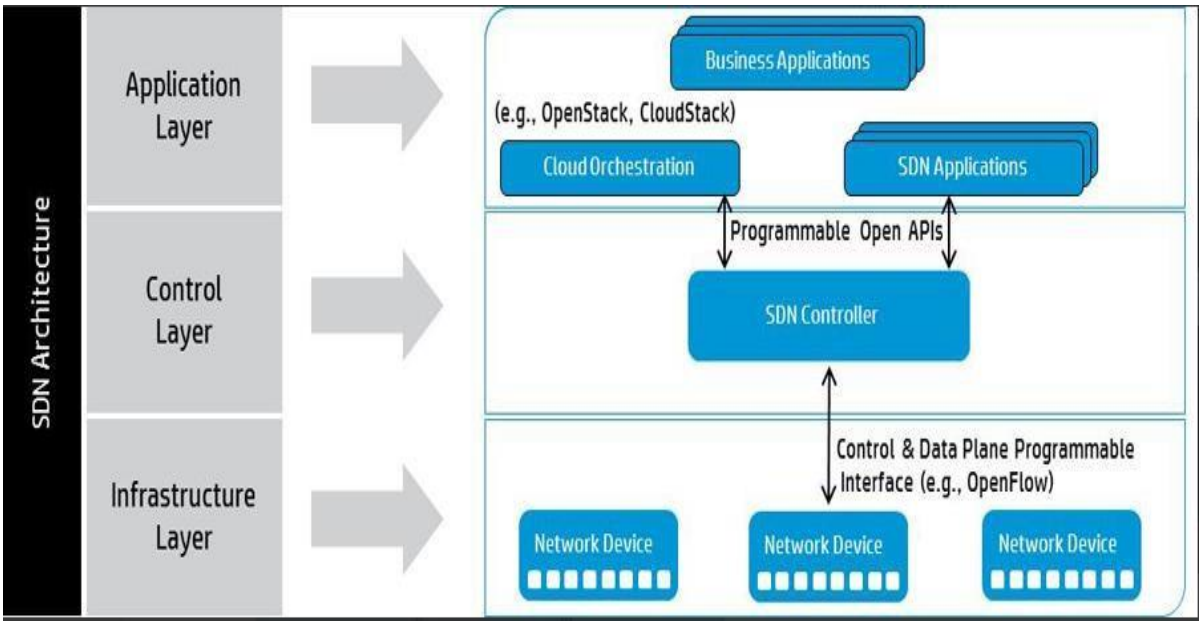
- ✓ SDN architecture is a networking concept where the control plane and the data plane are physically separated and the data plane is controlled by logically centralised control plane.

- ✓ SDN is software that gives network administrators a console interface where they can provision, manage, and break down networks without having to physically set up network switches and devices.

- ✓ The most fundamental rule of SDN architecture is:- Data plane is directly programmable by controller and centrally managed.

- ✓ Controller provide central view of the network and also allows switches to be directly controllable via network application written on top of Controller.

- ✓ This kind of abstraction is done via Northbound API. The Southbound API provides interaction between controller and switches.



Openflow:

- ✓ Openflow is a forwarding protocol, which is being used for the interaction between data plane and control plane.
- ✓ Any network is said to be openflow network only if the switches and controller in the network supports openflow.
- ✓ OpenFlow provides the functionalities proposed by SDN.
- ✓ OpenFlow is the most popular protocol to provide this interface.
- ✓ Some other protocols are NETCONF, SNMP, XMPP.
- ✓ So SDN is a paradigm and OpenFlow is a way of implementing it.
- ✓ As an open protocol, OpenFlow underpins the various SDN controller solutions.
- ✓ The complete SDN solution is taking SDN controller as the core, backed by OpenFlow switches and NFV to offer bountiful SDN app for a new smart, dynamic, open, custom network

Conclusion:

In this assignment, we tackled the tough subject of Controller REST API , using REST principles to guide the design of our Rails applications, mainly as they apply to the routing system and controller actions. We learned how the foundation of RESTful Rails is the resources method in your routes file and how to use the numerous options available to make sure that you can structure your application exactly how it needs to be structured.