Software Defined Systems

Week 3 Software Defined Networking (SDN) Ecosystem

Lecturer: Biniam Behailu

Addis Ababa Science and Technology

Addis Ababa, Ethiopia

Contents

Software Defined Networking (SDN) Ecosystem

Learning objectives

- Define Software Defined Networking and its core components.
- Explore the roles of controllers, data planes, and applications.
- Assess the Benefits and Challenges of SDN.
- Understand real-world applications and scenarios where SDN is implemented.

What is SDN?

- Software-defined networking (SDN) is an architecture that abstracts different, distinguishable layers of a network to make networks agile and flexible.
- It is an approach to networking that uses software controllers that can be driven by application programming interfaces (APIs) to communicate with hardware infrastructure to direct network traffic.

What is SDN?

- The goal of SDN is to improve network control by enabling enterprises and service providers to respond quickly to changing business requirements.
- SDN technology enables IT administrators to configure their networks using a software application
- SDN software is interoperable, meaning it should be able to work with any router or switch, no matter which vendor made it.

The Need for a New Network Architecture

- The explosion of mobile devices and content, server virtualization, and arrival of cloud services are among the trends.
- These are some of the factors driving the networking industry to re-examine traditional network architectures.

The Need for a New Network Architecture

Changing traffic patterns

- Within the enterprise data center, traffic patterns have changed significantly.
- In contrast to client-server applications where the bulk of the communication occurs between one **client** and **one server**, today's applications access different **databases** and **servers**.

The Need for a New Network Architecture

The consumerization of IT

- Users are increasingly employing mobile personal devices such as **smartphones**, **tablets**, and **notebooks** to access the corporate network.
- IT is under pressure to accommodate these personal devices in a fine-grained manner while protecting corporate data and intellectual property and meeting compliance mandates

The Need for a New Network Architecture

The rise of cloud services

- Enterprises have enthusiastically embraced both **public** and **private cloud** services, resulting in unprecedented growth of these services.
- Enterprise business units now want the agility to access applications, infrastructure, and other IT resources on demand.
- To add to the complexity, IT's planning for cloud services must be done in an environment of increased security, compliance, and auditing requirements.

The Need for a New Network Architecture

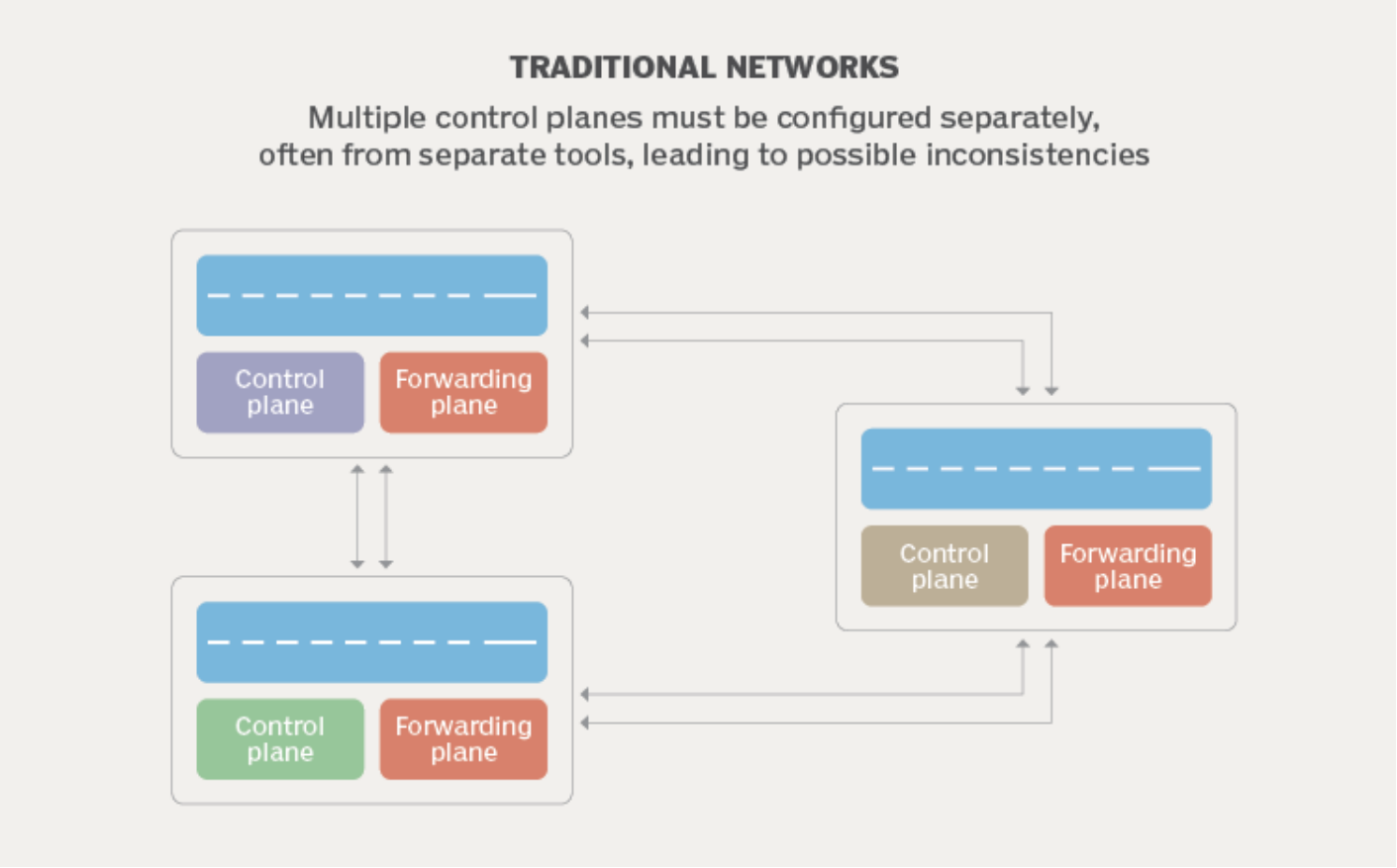
“Big data” means more bandwidth

- Handling today’s “big data” or mega datasets requires massive parallel processing on thousands of servers, all of which need direct connections to each other.
- The rise of mega datasets is fueling a constant demand for additional network capacity in the data center.

The Need for a New Network Architecture

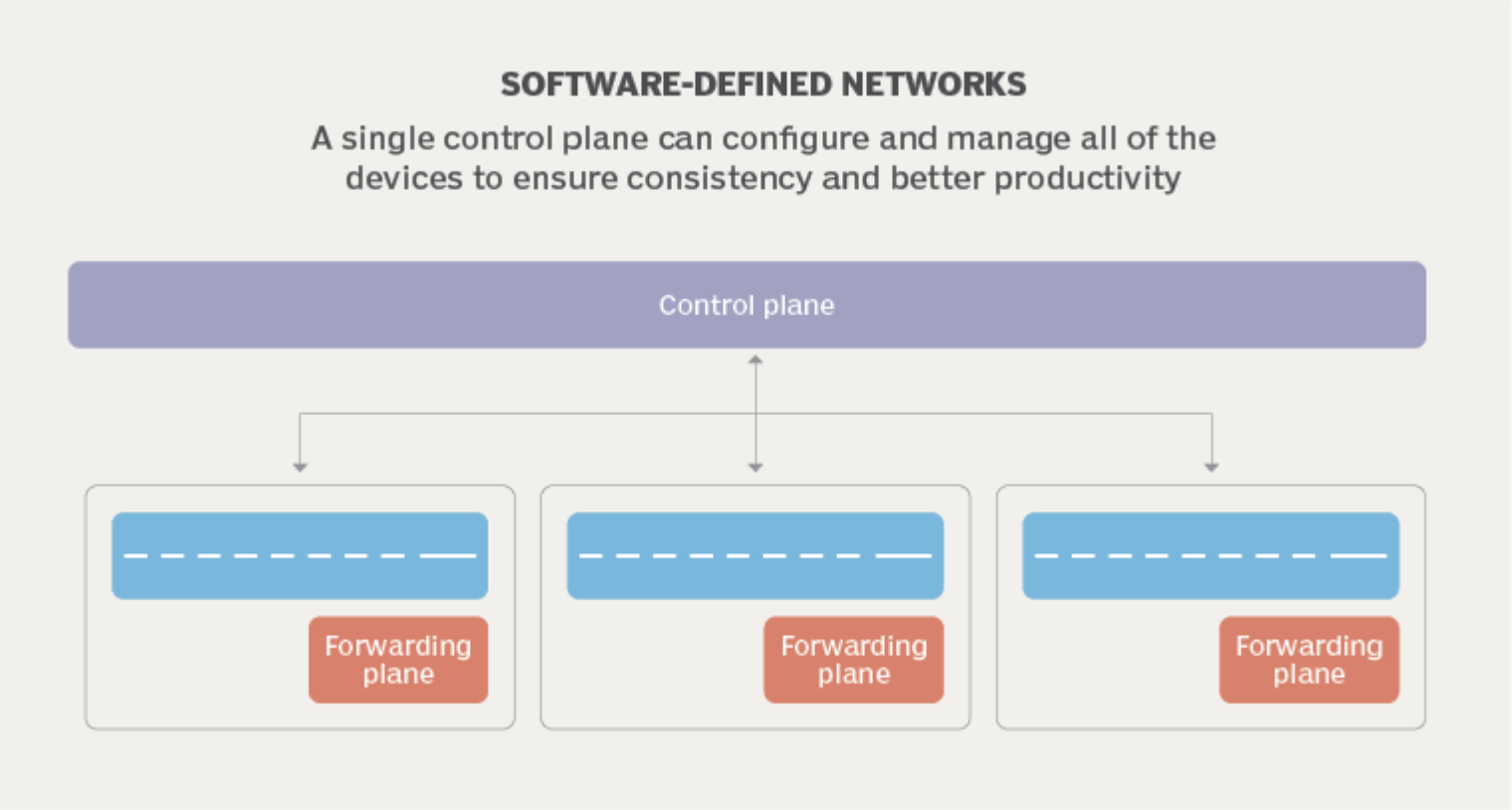
- Inability to scale
- Bandwidth limitations
- Vendor dependence
- Lack of flexibility
- Reliability
- Latency
- Complex configuration and management
- Security vulnerabilities

The Need for a New Network Architecture



Source: https://www.techtarget.com/searchnetworking/opinion/The-evolution-of-SDN-and-its-role-in-networking

The Need for a New Network Architecture



Source: https://www.techtarget.com/searchnetworking/opinion/The-evolution-of-SDN-and-its-role-in-networking

SDN Architecture and Components

- To understand software-defined networks, we need to understand the various planes involved in networking.
    - Management/ Application Plane
    - Control Plane
    - Data Plane (Forwarding Plane)

SDN Architecture and Components

Data Plane (Forwarding Plane)

- Data plane operates at the **network devices** (routers, switches, firewalls) and performs tasks such as **packet forwarding**, **switching**, and **filtering based** on predefined rules.
- The data plane processes packets based on the information contained in the packet headers (e.g., source and destination addresses) and directs them to the appropriate output ports.

SDN Architecture and Components

Control Plane

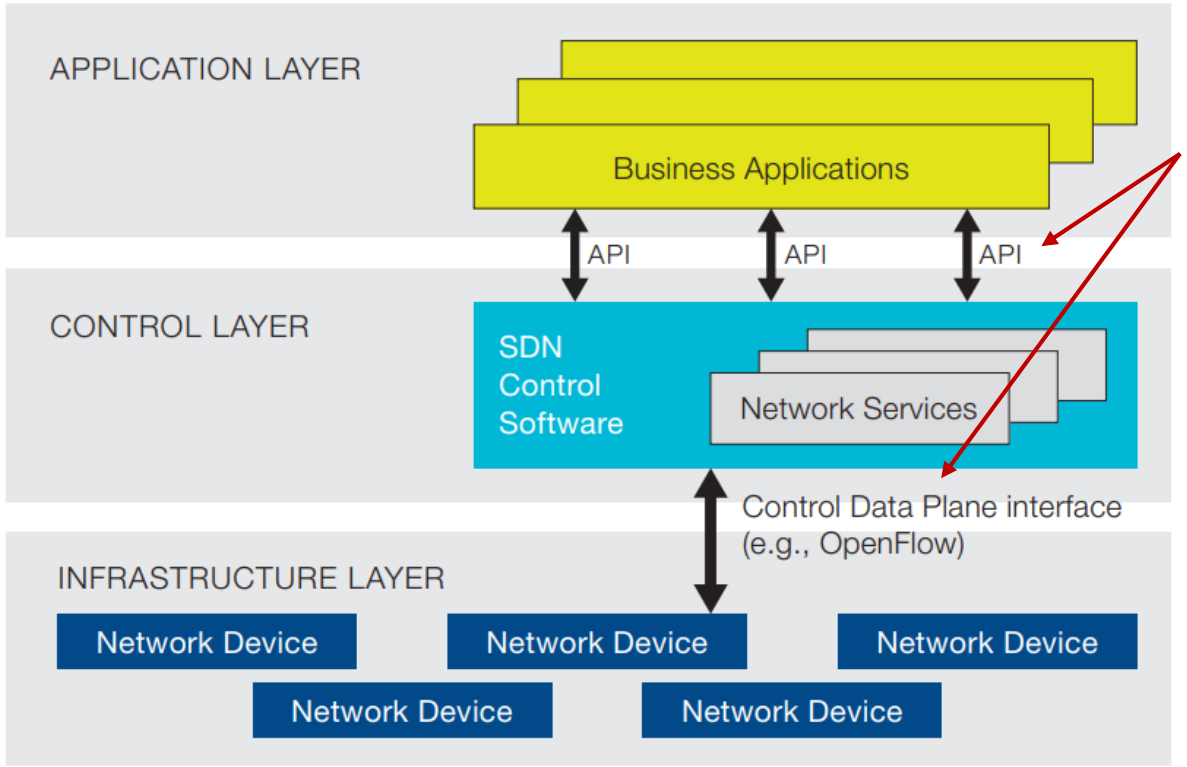
- Control plane handles tasks such as **routing**, **addressing**, and **network topology discovery**.
- The control plane communicates with network devices to exchange routing information, update forwarding tables, and establish paths for network traffic.
- In traditional networking, the control plane is distributed across network devices, with each device independently making routing decisions.

SDN Architecture and Components

Management/ Application Plane

- The management plane provides network administrators with the tools and interfaces to interact with network devices, monitor network performance, and ensure the network is functioning optimally.
- It involves tasks such as device configuration, software updates, performance monitoring, and network troubleshooting.

SDN Architecture and Components



APIs

- These three layers communicate using respective **northbound** and **southbound** APIs.

Source: https://blogs.cisco.com/networking/lets-clear-up-some-misconceptions-is-sdn-relevant-to-the-enterprise-wan

SDN Architecture and Components

Northbound Interface

- A northbound interface is an application programming interface (**API**) or **protocol** that allows a **lower-level** network component to communicate with a **higher-level** or more central component.
- The northbound interface allows applications to request network services, provide policy information, and receive network state information from the SDN controller.

SDN Architecture and Components

Southbound Interface

- The southbound interface connects the control layer with the infrastructure layer.
- It consists of communication protocols and APIs that enable the SDN controller to communicate with the network devices.
- Popular southbound interface standards are Simple Network Management Protocol (SNMP), OpenFlow, and Open Shortest Path First (OSPF).

OpenFlow Protocol and its Role in SDN

- **OpenFlow** (**OF**) is considered one of the **first** software-defined networking (SDN) standards.
- It’s an **open source** standard supported by many vendors, is the first software defined networking (SDN) control protocol.
- It’s an interface for remotely controlling the forwarding tables in network switches, routers and access points.

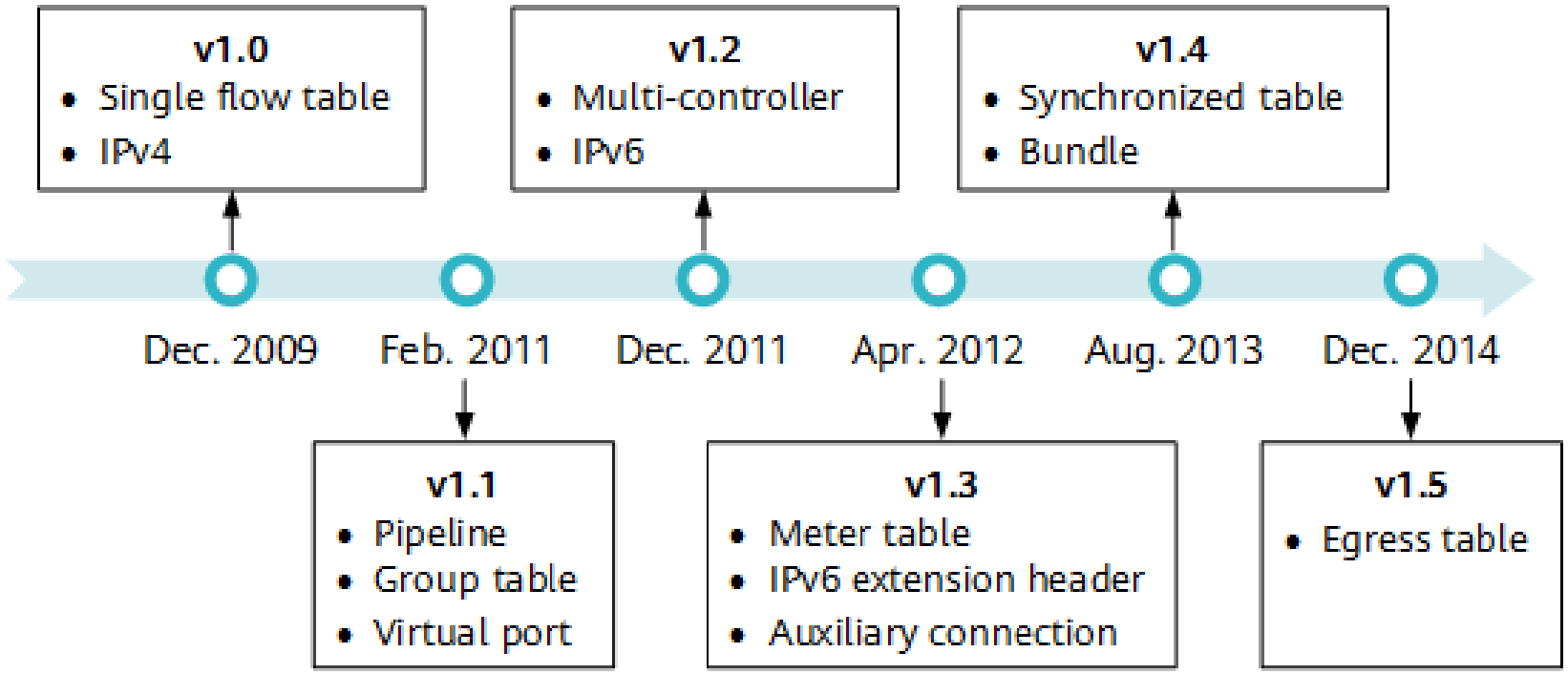
OpenFlow Protocol and its Role in SDN

- It is maintained by the Open Network Foundation(ONF).
- OpenFlow is a network control protocol.
- Network traffic does not go through the OpenFlow protocol.
- Instead, OpenFlow sends the control signals that tell the network switches how to route the network traffic.

OpenFlow Protocol and its Role in SDN

- OpenFlow originated from the **Clean Slate Program** of Stanford University.
- This program considered how the Internet could be redesigned with a "**clean slate**", and aimed to change the network infrastructure that was slightly out of date and difficult to evolve.

OpenFlow Protocol and its Role in SDN

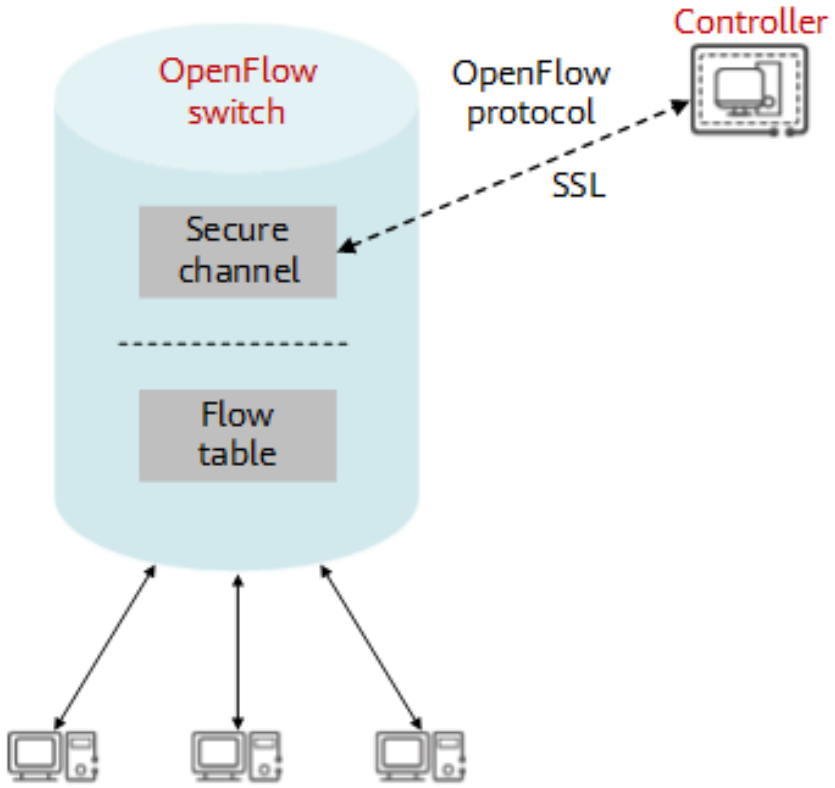


Source: https://info.support.huawei.com/info-finder/encyclopedia/en/OpenFlow.html

How does OpenFlow Works?

- The OpenFlow architecture consists of a **controller**, **OpenFlow switch**, and **secure channel**.
- The **controller** controls the network in a centralized manner to implement the functions of the control layer.
- The **OpenFlow switch** is responsible for forwarding at the data layer.
- It exchanges messages with the controller through a secure channel to receive forwarding entries and report its status.

How does OpenFlow Works?

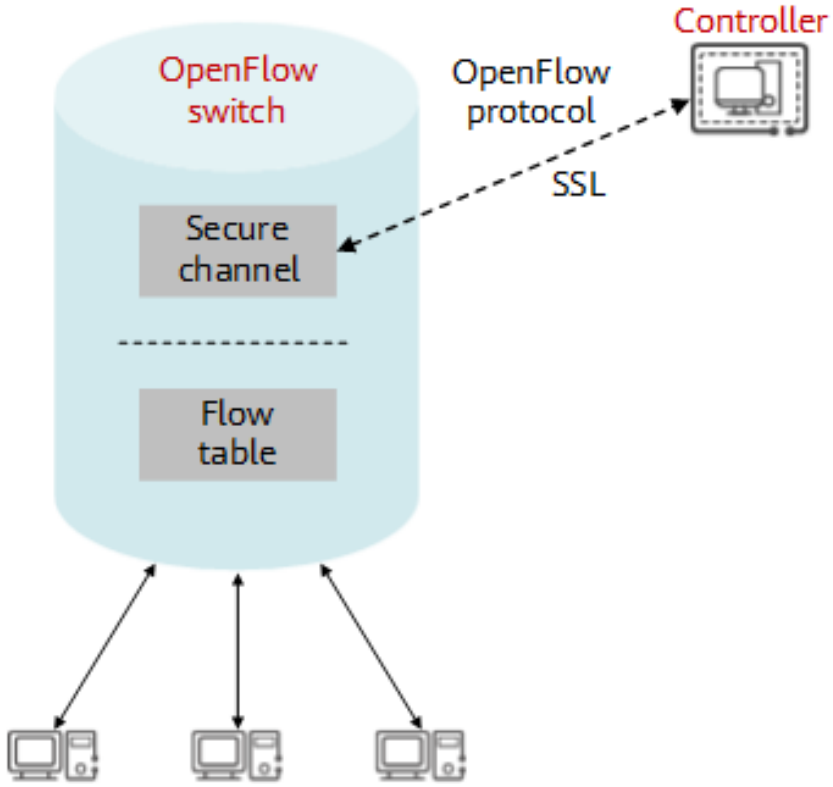


OpenFlow Controller

- An OpenFlow controller is the brain of the SDN architecture and is located at the control layer to instruct data forwarding through the OpenFlow protocol.

Source: https://info.support.huawei.com/info-finder/encyclopedia/en/OpenFlow.html

How does OpenFlow Works?

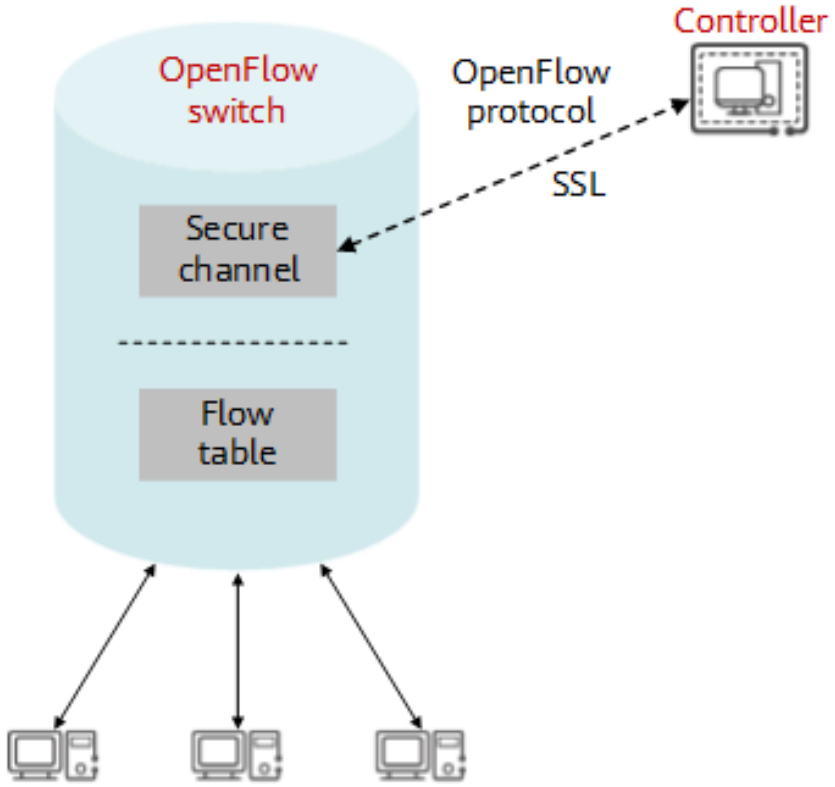


OpenFlow Controller

- Currently, mainstream OpenFlow controllers are classified into two types: open-source controllers and vendor-developed commercial controllers.

Source: https://info.support.huawei.com/info-finder/encyclopedia/en/OpenFlow.html

How does OpenFlow Works?

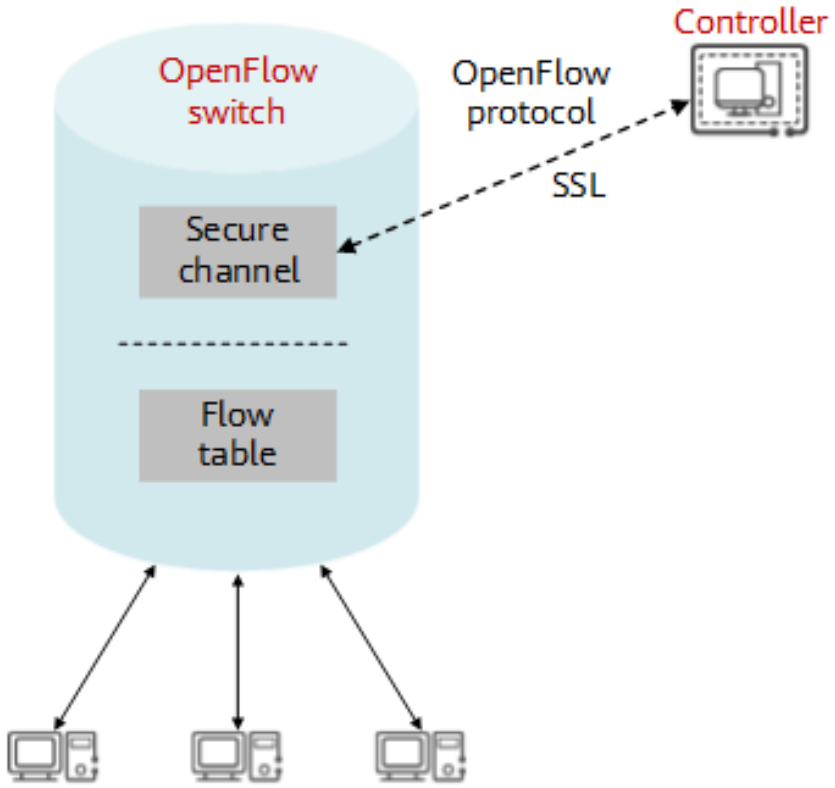


OpenFlow Controller

- The widely used open-source controllers include NOX, POX, and OpenDaylight.
- Huawei's iMaster NCE, Cisco Application Centric Infrastructure (ACI), HP VAN SDN Controller, Juniper Contrail, Big Switch Networks Controller are commercial ones.

Source: https://info.support.huawei.com/info-finder/encyclopedia/en/OpenFlow.html

How does OpenFlow Works?

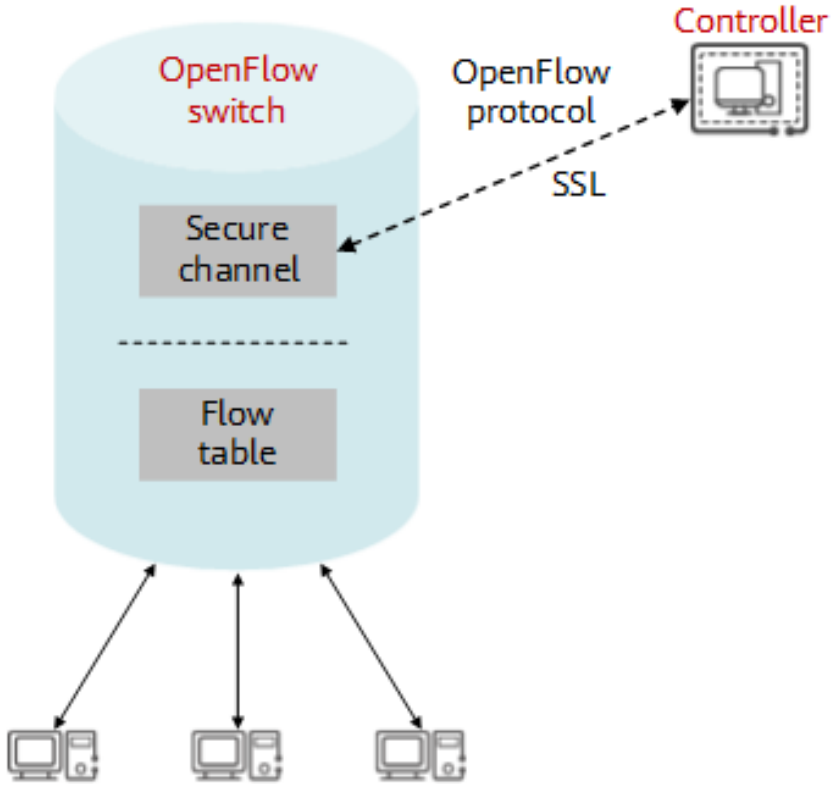


OpenFlow Secure Channel

- A secure channel is established between a **controller** and an **OpenFlow switch**.
- Through this channel, the controller controls and manages the switch, and receives feedback from the switch.

Source: https://info.support.huawei.com/info-finder/encyclopedia/en/OpenFlow.html

How does OpenFlow Works?



OpenFlow Switch

- As a core component of the OpenFlow network, an OpenFlow switch is mainly responsible for forwarding at the data layer.
- It can be a physical or virtualized switch/router.

Source: https://info.support.huawei.com/info-finder/encyclopedia/en/OpenFlow.html

Advantages of OpenFlow

- The SDN nature of OpenFlow allows for quick response to changes and failures.
- It is also highly flexible and can manage highly complex rules.
- Centralized Network Control
- Network Programmability
- Scalability and Flexibility
- Rapid Network Innovation
- Traffic Engineering and Optimization
- Interoperability and Vendor Neutrality
- Network Monitoring and Troubleshooting

OpenFlow: Case Study

- Imagine a campus area network (CAN) with **many buildings**, **switches** and **two internet connections**.

Normal Operation

- Network traffic flows through the closest internet connections to reach its destination efficiently.
- Each switch forwards packets based on predefined flow rules set by the SDN controller.

OpenFlow: Case Study

Link Failure

- If a link between **two** buildings **fails**, the switches **detect** this change and **report** the status to the **SDN controller**.
- The controller responds by **recalculating** the **optimal forwarding paths** and updates the **flow rules** on the affected switches, ensuring that traffic is rerouted through alternative paths.

OpenFlow: Case Study

Internet Connection Failure

- In the event of an internet connection failure, the SDN controller dynamically adjusts the flow rules to reroute all internet-bound traffic through the remaining functional link, maintaining connectivity for users.

OpenFlow: Case Study

Traffic Management

- A large CAN can generate significant unwanted traffic, such as broadcast requests or specific protocols like **Apple Bonjour**, which can overwhelm the network.
- Instead of deploying expensive firewalls between each building or floor, the **SDN controller** can implement **flow rules** to drop **unwanted traffic** at the switches, preventing it from propagating across the network.

OpenFlow: Case Study

Dynamic Policy Enforcement

- By using OpenFlow, the network can adapt to changing conditions in real-time, allowing for efficient traffic management and enhanced security without adding significant hardware costs.

SDN Use Cases

Data Center Networks

- SDN is widely used in data centers to improve network agility, simplify management, and support virtualization and cloud computing environments.
- It enables automated provisioning, network slicing, and seamless migration of virtual machines.

SDN Use Cases

Wide Area Networks (WANs)

- SDN can optimize WAN performance and manage traffic across multiple sites by enabling centralized control and dynamic path selection.
- It simplifies WAN management, reduces costs, and improves application performance.

SDN Use Cases

Campus and Enterprise Networks

- SDN brings flexibility and automation to campus and enterprise networks, enabling policy-based network management, simplified network configuration, and secure access control.
- It supports dynamic network partitioning and prioritization of traffic.

SDN Use Cases

Internet of Things (IoT) Networks

- SDN provides a scalable and flexible infrastructure for managing large-scale IoT deployments.
- It enables efficient device connectivity, dynamic routing, and security enforcement, facilitating IoT deployments in various industries.

SD-WAN

- SD-WAN (Software-Defined Wide Area Networking) is a software-defined approach to managing the WAN.
- It uses software-defined networking (SDN) principles to abstract the underlying network infrastructure and provide centralized control and management of the WAN.
- SD-WAN evolved from MPLS technology, which has powered private connectivity for more than two decades.

SD-WAN

- Multiprotocol Label Switching (MPLS) enables Enterprises and Service Providers to build next-generation intelligent networks that deliver a wide variety of advanced, value-added services over a single infrastructure.
- While MPLS handled failure scenarios with **backup links**, SD-WAN handles them with **real-time traffic steering** based on centralized policy.
- Also, since SD-WAN unifies the entire WAN backbone, it delivers comprehensive analytics across the network globally.

Advantages of SD-WAN

- Reducing costs with transport independence across MPLS, 4G/5G LTE, and other connection types.
- Improving application performance and increasing agility.
- Optimizing user experience and efficiency for software-as-a-service (SaaS) and public-cloud applications.
- Simplifying operations with automation and cloud-based management.

Intent-based networking (IBN)

Intent-based networking (IBN)

- IBN is an approach to network **management** and **automation** that focuses on aligning **network behavior** with the desired **intent** of the **organization** or **network administrator**.
- IBN aims to simplify network operations, improve agility, and enhance security by translating high-level business objectives into specific network configurations and policies.

Test Your Knowledge

1. What is the primary function of an SDN controller?

    A) To forward packets between devices

    B) To manage the data plane of network devices

    C) To provide a user interface for network management

    D) To separate the control and data planes

Test Your Knowledge

1. What is the primary function of an SDN controller?

   A) To forward packets between devices

   B) To manage the data plane of network devices

   C) To provide a user interface for network management

   D) To separate the control and data planes

Reason: The SDN controller centralizes control by separating the control plane from the data plane, allowing for more flexible and programmable network management.

Test Your Knowledge

2. What is a key benefit of using OpenFlow in an SDN environment?

A) It simplifies hardware requirements.

B) It allows for vendor-specific solutions only.

C) It provides a standardized way to program network devices.

D) It eliminates the need for network security.

Test Your Knowledge

2. What is a key benefit of using OpenFlow in an SDN environment?

A) It simplifies hardware requirements.

B) It allows for vendor-specific solutions only.

C) It provides a standardized way to program network devices.

D) It eliminates the need for network security.

Reason: OpenFlow standardizes how network devices can be controlled and programmed, enabling interoperability among devices from different vendors.

Thank you!

Lecturer: Biniam Behailu

Addis Ababa Science and Technology

Addis Ababa, Ethiopia