UNIVERSITY OF DODOMA

College of Informatics and Virtual Education

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CG 311: Software Defined Networking

Presentation Assignment

Objective

This assignment is designed to deepen your understanding of key concepts and technologies in Software-Defined Networking (SDN). Each group will research and deliver a structured presentation on one of the listed topics, focusing on its technical principles, architecture, real-world applications, and emerging trends. The goal is to develop a comprehensive understanding of SDN components and their role in modern networking.

A. Topic Selection and Guidelines

Each group must select one topic from the list and focus entirely on that subject for their presentation. Your research should provide a detailed exploration of the topic, covering fundamental concepts, implementation details, industry relevance, and future developments. To encourage diverse discussions, no two groups should work on the same topic. Topics will be assigned on a first-come, first-served basis. Once a group has chosen a topic, they must confirm their selection with the instructor before proceeding with their research and presentation development. **You are encouraged to use academic papers, official SDN documentation, whitepapers, and vendor resources to support your research.**

B. Presentation Topics

1. **Software OpenFlow Switches** (e.g., Open vSwitch, Lagopus, Pantou, CPqD, etc.)

   Software OpenFlow switches are critical components of SDN, enabling dynamic control over network traffic without relying on traditional networking hardware. ***This topic should explore:***

   - The architecture and functionality of software-based OpenFlow switches.

- A comparison of different software OpenFlow switch implementations, including Open vSwitch (OVS), Lagopus, Pantou, and CPqD.
- The role of software switches in cloud networking, virtualization, and network testing.
- Performance trade-offs between software and hardware OpenFlow switches.
- Use cases in data centers, research, and enterprise environments.

2. **Hardware OpenFlow Switches** (e.g., Arista 7050, Pica8, NoviFlow, HP 5400R, Barefoot Tofino, etc.)

Hardware OpenFlow switches integrate OpenFlow support into dedicated network hardware for high-performance forwarding and scalability. ***This topic should discuss:***

- The role of OpenFlow in hardware-based switching.
- Key hardware OpenFlow switch manufacturers, including Arista, Pica8, NoviFlow, HP, Dell, and Barefoot (Intel Tofino).
- A comparison of ASIC-based OpenFlow switches versus software switches in terms of performance, scalability, and energy efficiency.
- How hardware OpenFlow switches are used in carrier networks, enterprise data centers, and software-defined WANs (SD-WAN).
- Challenges in OpenFlow hardware adoption and integration.

3. Open vSwitch Database Management Protocol (OVSDB)

OVSDB is a management protocol used for configuring and monitoring Open vSwitch (OVS) instances. ***This topic should explore:***

- The role of OVSDB in SDN environments and its relationship with OpenFlow.
- How OVSDB manages switch configurations, port settings, and network policies.
- A comparison between OVSDB and alternative network configuration protocols (e.g., NETCONF, REST APIs).
- Practical use cases in cloud computing, network virtualization, and data center automation.
- Integration of OVSDB with SDN controllers and orchestration frameworks (e.g., OpenDaylight, ONOS).

4. **OpenFlow Controllers** (e.g., ONOS, OpenDaylight, NOX, POX, Ryu, Floodlight, etc.)

OpenFlow controllers serve as the central intelligence of SDN networks, managing traffic flows and network policies dynamically. ***This topic should include:***

- The architecture and functionality of OpenFlow-based SDN controllers.
- An in-depth comparison of popular controllers, including ONOS (Open Network Operating System), OpenDaylight, NOX, POX, Ryu, Floodlight, and Trema.
- The differences between proactive and reactive flow control strategies.
- The role of SDN controllers in managing large-scale networks, including carrier networks, ISPs, and enterprise IT environments.
- Challenges in OpenFlow controller scalability, redundancy, and fault tolerance.

5. **Mininet: Open-Source Network Emulation Environment**

Mininet is a widely used open-source network emulator that allows users to create and test SDN environments. ***This topic should discuss:***

- The architecture and design of Mininet.
- How Mininet enables testing of OpenFlow controllers, switches, and SDN applications.
- Key features such as topology emulation, link management, and API extensibility.
- A comparison between Mininet and alternative network emulation platforms (e.g., GNS3, CORE, Cloonix).
- Practical examples of using Mininet for research, development, and SDN education.

6. **OpenFlow Configuration Protocol and Bootstrapping**

OpenFlow configuration and bootstrapping mechanisms define how OpenFlow-enabled devices connect, authenticate, and initialize within SDN environments. ***This topic should cover:***

- The OpenFlow handshake process and secure channel establishment.

- Methods for configuring OpenFlow switches, including OpenFlow Table Pipeline configuration.
- The role of Transport Layer Security (TLS) in OpenFlow communication.
- Differences between static and dynamic bootstrapping methods.
- Challenges in OpenFlow switch onboarding, interoperability, and performance tuning.

7. **Network Function Virtualization (NFV) and its Relationship with SDN**

NFV is a paradigm that virtualizes traditional network functions (e.g., firewalls, load balancers, DPI) and runs them on general-purpose hardware. ***This topic should explore:***

- The relationship between NFV and SDN in modern networking.
- How NFV enables cost reduction, scalability, and automation in telecom and enterprise networks.
- The architecture of NFV, including the role of Virtual Network Functions (VNFs), NFV Orchestrators (NFVO), and Management & Orchestration (MANO) frameworks.
- Real-world use cases, including NFV in 5G networks, cloud service providers, and software-defined WANs (SD-WAN).
- Challenges and future trends in NFV, including performance bottlenecks, standardization efforts, and integration with AI-driven networking.

8. P4 Programming Language

P4 (Programming Protocol-independent Packet Processors) is an open-source programming language designed specifically for programmable network devices like switches, routers, and network interface cards (NICs). Traditional networking devices have hardcoded packet processing logic, making them inflexible. To address this limitation, P4 was developed to allow developers to program the **data plane** to dictate how packets are processed then forwarded using software. This makes networking more flexible, customizable, and aligned with the principles of Software-Defined Networking (SDN). ***This topic covers:***

- Introduction to P4 – Understanding what P4 is, its purpose, and its role in SDN.
- Challenges in Traditional Networking – Exploring why fixed-function network devices limit flexibility and innovation.

- How P4 Works – Examining its key components, including packet parsers, match-action tables, and control logic.
- P4 Architecture – Understanding how P4 programs are deployed on hardware and software-based network devices.
- Use Cases and Applications – Covering areas like data center networking, security, 5G, IoT, and network telemetry.
- Comparison with Other SDN Approaches – Highlighting how P4 differs from traditional SDN models like OpenFlow.
- Real-World Implementations – Looking at how companies and researchers are using P4 in production environments.
- The Future of P4 – Discussing emerging trends, challenges, and advancements in programmable networking.

C. Presentation Requirements

i. Duration: 10-15 minutes per presentation.

ii. Format: PowerPoint, Google Slides, or equivalent, with clear visuals and structured content.

iii. Content Structure:

1. Introduction to the topic and its relevance.
2. Explanation of fundamental concepts and technical principles.
3. Overview of key components and functionality.
4. Real-world applications and industry significance.
5. Emerging trends and technological advancements.
6. Conclusion and key takeaways.
7. References (at least three credible academic or industry sources).

D. **Submission:** Groups must submit their presentation slides and a brief summary document (300-500 words) outlining key points before the presentation date.

E. Assessment Criteria

i. Depth of Research, and Technical Accuracy – 30%

ii. Clarity, Organization, and Structure – 20%

iii. Presentation Skills, and Engagement – 20%

iv. Use of Visuals, Examples, and Demonstrations – 15%

v. Proper Referencing, and Citation of Sources – 15%

F. Key Dates

i. Submission Deadline: 09/02/2025

ii. Presentation Date: TBD