Ecole Supérieure d'Informatique et du Numérique
COLLEGE OF ENGINEERING & ARCHITECTURE

TP n° 5

Pr. Mohammed Bouhorma

TP : Analyse des Échanges OpenFlow entre Mininet et ONOS

Objectif :

Analyser et comprendre les échanges OpenFlow entre le contrôleur ONOS et les switches dans une topologie Mininet, en utilisant Wireshark pour capturer les messages comme Hellow, **Packet-In**, **Packet-Out, Flow-Mod**.

Prérequis :

- **Mininet** et **ONOS** installés sur la même machine.
- **Wireshark** pour capturer et analyser les paquets.

Matériel requis :

- Accès à une machine avec Mininet et ONOS.
- Connexion internet pour télécharger les outils nécessaires si ce n'est pas déjà fait.

Partie 1 : Préparation de la topologie Mininet et lancement d'ONOS

1. **Lancer ONOS** : Démarrez ONOS et vérifiez que le contrôleur est opérationnel.
2. **Obtenez l'adresse IP d'Ubuntu :** Une fois la machine virtuelle démarrée, ouvrez un terminal dans Ubuntu et exécutez la commande suivante pour récupérer l'adresse IP du « Docker » :

```
ifconfig
```

```
zineb@zineb-virtual-machine:~$ ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        inet 172.17.0.1  netmask 255.255.0.0  broadcast 172.17.255.255
        ether 22:6b:36:71:9d:1c  txqueuelen 0  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.180.129  netmask 255.255.255.0  broadcast 192.168.180.255
        inet6 fe80::c7fe:209f:250d:d7ba  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:5d:a4:ef  txqueuelen 1000  (Ethernet)
        RX packets 477  bytes 400695 (400.6 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 270  bytes 35982 (35.9 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 138  bytes 12664 (12.6 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 138  bytes 12664 (12.6 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

3. **Lancer ONOS** : Démarrez ONOS et vérifiez que le contrôleur est opérationnel.

```
sudo docker start onos
```

```
zineb@zineb-virtual-machine:~$ sudo docker start onos1
onos1
zineb@zineb-virtual-machine:~$
```

Si ONOS est en cours d'exécution, vous pouvez accéder à l'interface web d'ONOS depuis votre navigateur en utilisant l'adresse IP obtenue précédemment et le port 8181, par exemple : http://<adresse_IP>:8181/onos/ui.

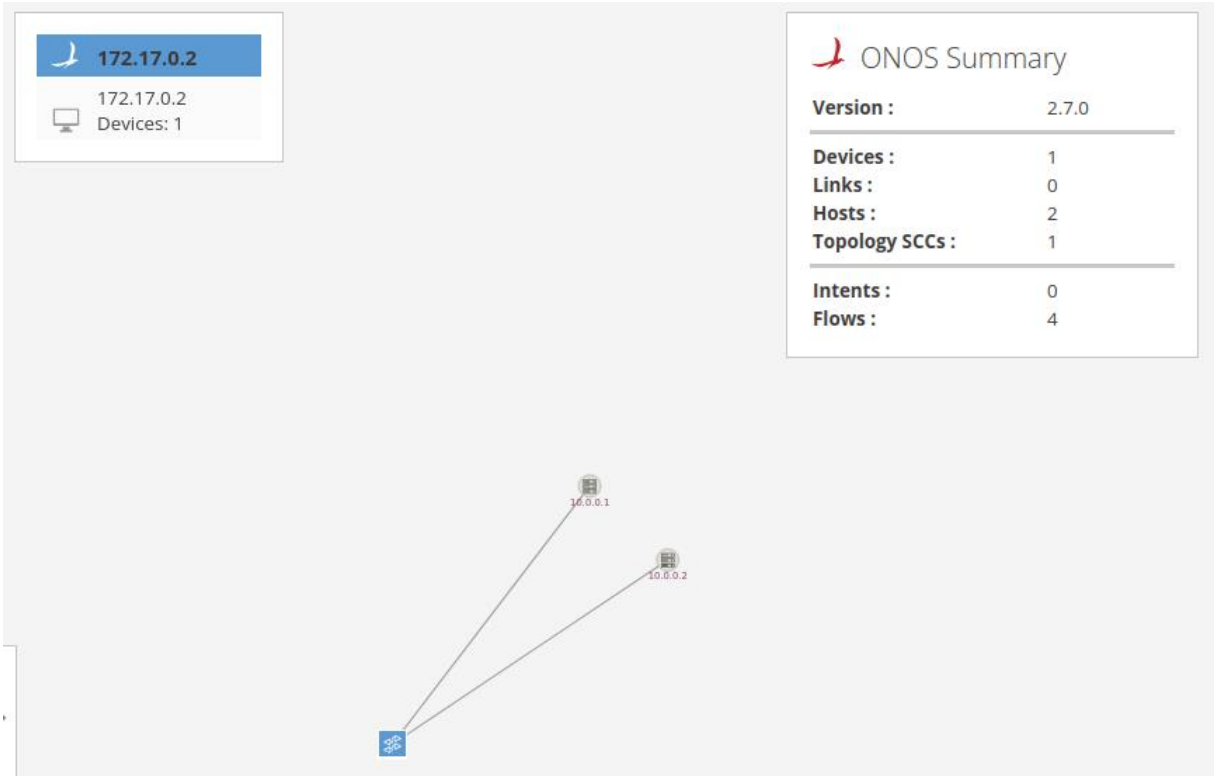
Cette commande vous permet de connaitre les ports utilisés par ONOS

```
zineb@zineb-virtual-machine:~/mininet$ sudo ss -tuln | grep 66
tcp   LISTEN 0      4096         0.0.0.0:6633       0.0.0.0:*
tcp   LISTEN 0      10           0.0.0.0:6654       0.0.0.0:*
tcp   LISTEN 0      4096         0.0.0.0:6653       0.0.0.0:*
tcp   LISTEN 0      4096            [::]:6633          [::]:*
tcp   LISTEN 0      4096            [::]:6653          [::]:*
```

Lancer une topologie simple

```
sudo mn --topo single,2   --controller=remote,ip=172.17.0.2,port=6653   --switch
ovs,protocols=OpenFlow13
```

Voilà ce qu'il faut avoir sur ONOS :



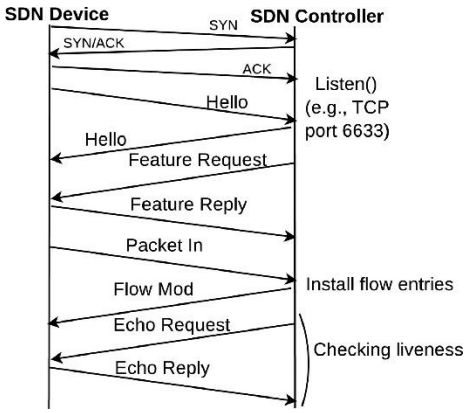
Nettoyer Mininet et passer à la deuxième partie du TP

Mininet>exit
Sudo mn -c

Partie 2 : Filtrage des paquets

Pour filtrer des paquets spécifiques dans Wireshark on peut utiliser les filtres suivants pour chaque type de message dans le protocole OpenFlow, par exemple :

1. **Messages "Hello"** : Ces messages sont échangés au début de la communication entre le contrôleur et le commutateur pour établir une connexion.
   - Filtre Wireshark : openflow_v4.type == 0
2. **Messages "Flow Mod"** : Utilisés pour ajouter, modifier ou supprimer des flux dans la table de flux d'un commutateur.
   - Filtre Wireshark : openflow_v4.type == 14
3. **Messages "Packet In"** : Envoyés du commutateur au contrôleur pour informer de la réception d'un paquet pour lequel il n'a pas de règle de flux correspondante.
   - Filtre Wireshark : openflow_v4.type == 10
4. **Messages "Packet Out"** : Envoyés par le contrôleur au commutateur pour lui indiquer comment traiter un paquet spécifique.
   - Filtre Wireshark : openflow_v4.type == 13



Exercice 1 :

1. Lancer Wireshark dans un terminal et capturer le trafic interne entre Mininet et ONOS. : sudo wireshark
2. **Lancer ONOS** : Démarrez ONOS et vérifiez que le contrôleur est opérationnel (visualiser la topologie dans ONOS.

```
sudo mn --topo single,2   --controller=remote,ip=172.17.0.2,port=6653   --switch ovsk,protocols=OpenFlow13
```

3. **Ouvrir Wireshark** et appliquez le filtre suivant :

```
openflow_v4.type==0
```

Question : Quel est le rôle de ce filtre ?

Le filtre openflow_v4.type==0 permet de ne voir que les messages HELLO OpenFlow 1.3, c'est-à-dire les échanges initiaux entre Mininet (switches) et ONOS (contrôleur).

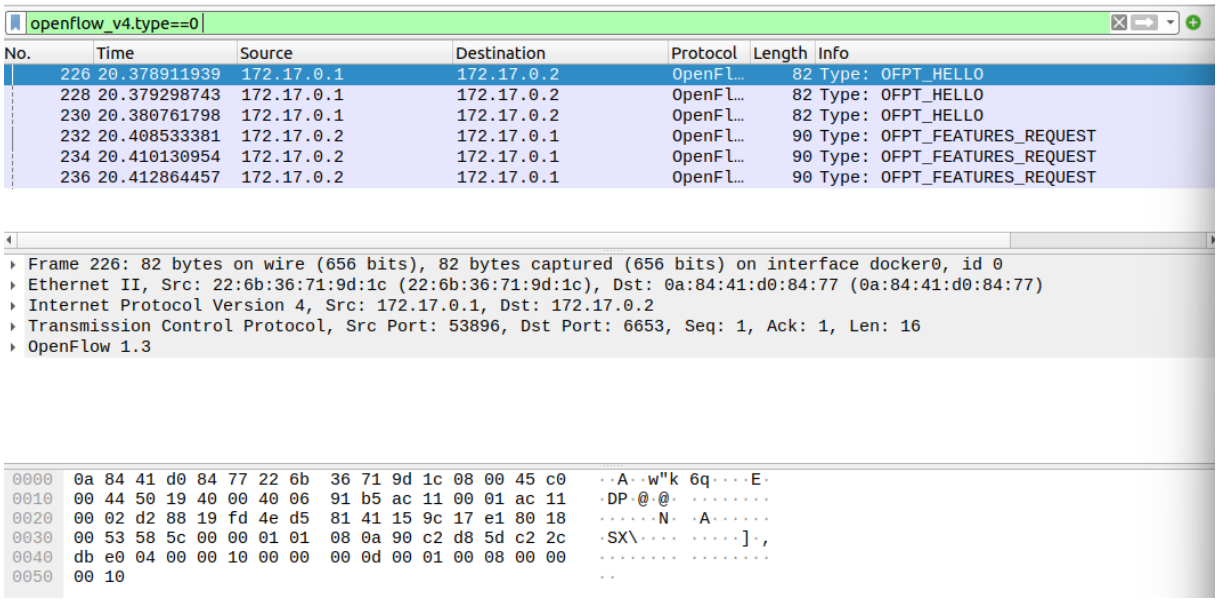
Nettoyer Mininet :

Mininet>exit
Sudo mn -c

4. Lancer à nouveau Wireshark
5. **Démarrer Mininet avec une autre topologie** : Utilisez la commande suivante pour créer une topologie linéaire avec trois switches et un contrôleur distant.

```
sudo mn --topo linear,3 --controller=remote,ip=172.17.0.2 --switch ovs,protocols=OpenFlow13 --mac
```

```
openflow_v4.type==0
```



Question : Combien de paquets “Hellow” vous avez pu capturer ? pourquoi ?

```
mininet> pingall
```

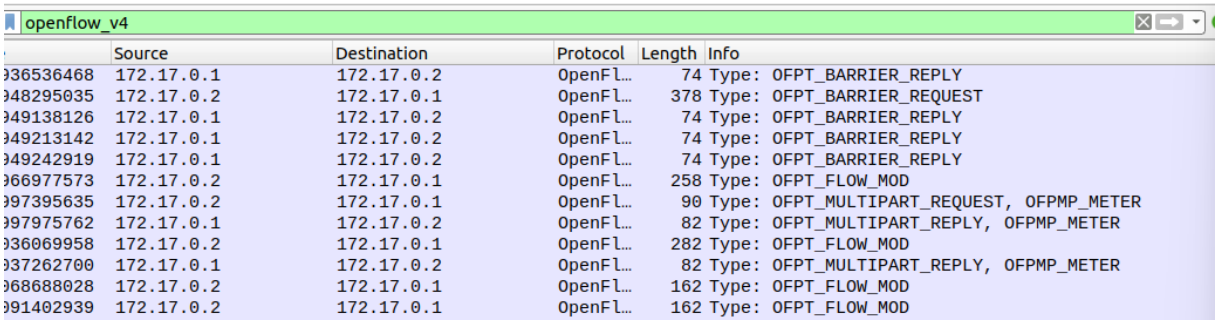
On a capturé 3 paquets HELLO, un par switch, car chaque switch envoie un message HELLO au contrôleur pour établir la session OpenFlow.

Exercice 2 : Capture des Messages OpenFlow avec Wireshark

1. **Ouvrir Wireshark** et sélectionner l'interface pour capturer le trafic interne entre Mininet et ONOS.
2. **Appliquer un filtre de capture** pour se concentrer sur les messages OpenFlow :

```
tcp.port == 6653 or openflow_v4
```

Le port 6653 est le port par défaut utilisé par ONOS pour OpenFlow.



3. Utiliser ce filtre pour voir les premiers messages échangés entre le contrôleur ONOS et un OVS. Pour chaque résultat expliquez et analyser vos résultats.

openflow_v4.type== 14

No.	Time	Source	Destination	Protocol	Length	Info
437	21.966977573	172.17.0.2	172.17.0.1	OpenFl…	258	Type: OFPT_FLOW_MOD
442	22.036069958	172.17.0.2	172.17.0.1	OpenFl…	282	Type: OFPT_FLOW_MOD
447	22.068688028	172.17.0.2	172.17.0.1	OpenFl…	162	Type: OFPT_FLOW_MOD
449	22.091402939	172.17.0.2	172.17.0.1	OpenFl…	162	Type: OFPT_FLOW_MOD

4. **Générer du trafic entre les hôtes** :
   Dans Mininet, initiez un ping entre deux hôtes, par exemple entre h1 et h3 :

```
mininet> h1 ping h3
```

5. **Observer les messages OpenFlow dans Wireshark** :

   Pendant que le ping est en cours, examinez les paquets **Packet-In** envoyés par le switch au contrôleur, et les réponses **Flow-Mod** envoyées par ONOS au switch.

172.17.0.2	172.17.0.1	OpenFl…	148	Type: OFPT_PACKET_OUT
172.17.0.1	172.17.0.2	OpenFl…	150	Type: OFPT_PACKET_IN
172.17.0.2	172.17.0.1	OpenFl…	148	Type: OFPT_PACKET_OUT
172.17.0.1	172.17.0.2	OpenFl…	150	Type: OFPT_PACKET_IN
172.17.0.2	172.17.0.1	OpenFl…	148	Type: OFPT_PACKET_OUT
172.17.0.1	172.17.0.2	OpenFl…	150	Type: OFPT_PACKET_IN
172.17.0.2	172.17.0.1	OpenFl…	148	Type: OFPT_PACKET_OUT
172.17.0.1	172.17.0.2	OpenFl…	150	Type: OFPT_PACKET_IN
172.17.0.2	172.17.0.1	OpenFl…	148	Type: OFPT_PACKET_OUT
172.17.0.1	172.17.0.2	OpenFl…	150	Type: OFPT_PACKET_IN

Partie 3 : Analyse des Messages OpenFlow

Questions :

1. Quelles sont les étapes de l’échange OpenFlow observé entre ONOS et les switches ?

Le switch se connecte au contrôleur ONOS via le port OpenFlow (ex. 6653).

Le switch envoie un Hello au contrôleur.

ONOS répond par un Hello et échange des messages de configuration (Features, Capabilities).

Lorsqu’un paquet inconnu arrive sur le switch, il envoie un Packet-In à ONOS.

ONOS analyse le paquet et répond avec un Flow-Mod pour installer une règle de flux.

2. Que signifie le message Packet-In capturé dans Wireshark ?

C’est un message envoyé par le switch au contrôleur pour signaler un paquet qu’il ne sait pas traiter.

Il contient le paquet complet ou une partie, ainsi que les informations de port d’entrée.

Cela permet au contrôleur de décider quoi faire avec ce paquet (forward, drop, modifier…).

3. Quelles informations trouve-t-on dans le message Flow-Mod envoyé par ONOS au switch ?

Critères du flux (match) : par exemple IP source/destination, ports, VLAN…

Actions à effectuer : forward vers un port, drop, modifier le paquet…

Priorité de la règle et durée de vie (idle/hard timeout).

4. Pourquoi le contrôleur ONOS installe-t-il une règle de flux suite au message Packet-In ?

Pour que les futurs paquets similaires soient traités directement par le switch sans revenir au contrôleur.

Cela réduit la charge sur le contrôleur et accélère le traitement des paquets.

5. Quel impact a cette règle de flux sur les paquets suivants entre h1 et h3 ?

Les paquets suivants correspondant à cette règle sont directement envoyés par le switch vers le port correct. Ils ne déclenchent plus de Packet-In, donc la communication est plus rapide et efficace.