


I. Import i konfiguracja maszyn wirtualnych

1. Uruchom program **VMware Workstation Pro** i zaimportuj maszynę wirtualną `OpenDaylight_Debian_12`.
2. Pobierz i zaimportuj maszynę wirtualną z Mininetem w sposób analogiczny do tego, który stosowałeś na poprzednich zajęciach.
3. Uruchom obie maszyny. Dane logowania do maszyn są następujące:

```
# Maszyna Mininet #  
# Nazwa użytkownika: mininet  
# Hasło: mininet  
  
# Maszyna OpenDaylight_Debian_12 #  
# Nazwa użytkownika: piesk  
# Hasło: piesk
```

4. Po zalogowaniu się w terminalu sprawdź adresy dodanych kart sieciowych na obu maszynach poleceniem:

```
ip addr
```

-  5. Po upewnieniu się, że karty poprawnie pobrały adresy IP sprawdź połączenie pomiędzy maszynami poleceniem `ping`.
6. Jeżeli wszystko działa poprawnie przejdź do części II.

II. Uruchomienie kontrolera OpenDaylight

1. Przejdź do maszyny `OpenDaylight_Debian_12`.
2. Uruchom konsolę Karaf za pomocą poniższego polecenia – proces uruchamiania może chwilę potrwać.

```
/opt/opendaylight/bin/karaf
```

3. W konsoli Karaf sprawdź listę wszystkich dostępnych pakietów poleceniem:

```
feature:list
```

4. ...oraz tych już zainstalowanych pakietów poleceniem:

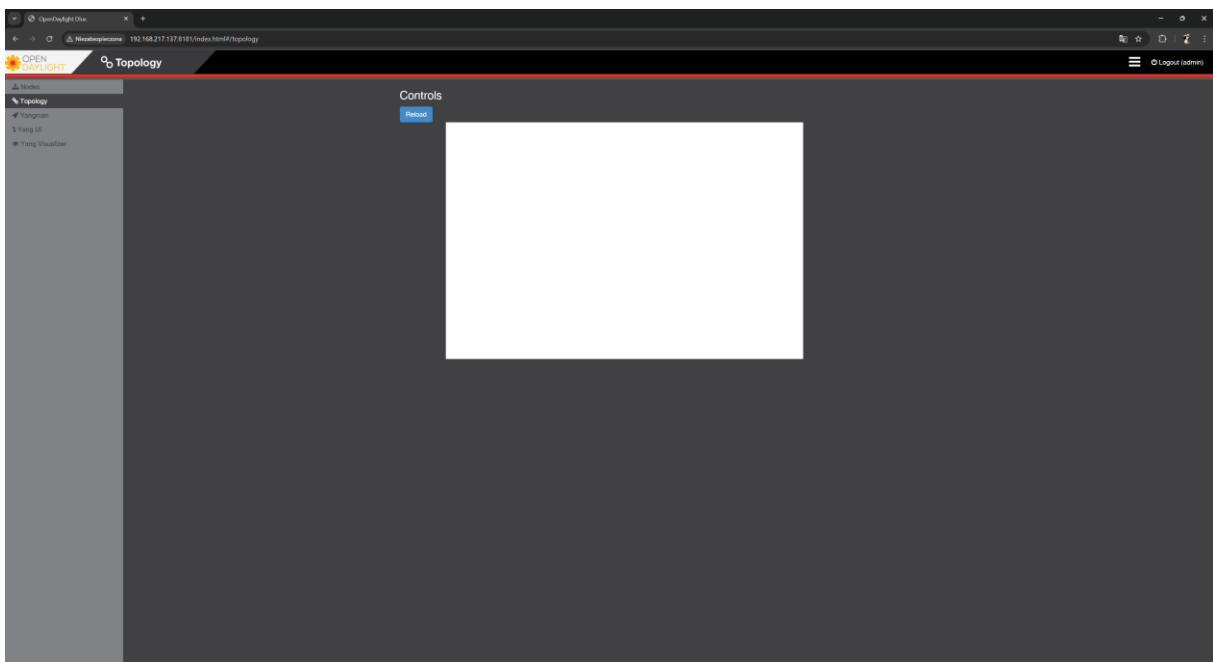
```
feature:list -i
```

5. Uruchom przeglądarkę internetową
6. Przejdź pod adres: `adres_ip_maszyny:8181/index.html`.
7. Zaloguj się do nakładki graficznej DLUX (OpenDaylight User Experience) wpisując:

```
Username: admin
```

```
Password: admin
```

8. Przejdź do zakładki Topology (powinna być pusta, nie ma potrzeby wciskać Reload!). Pozostaw przeglądarkę włączoną.



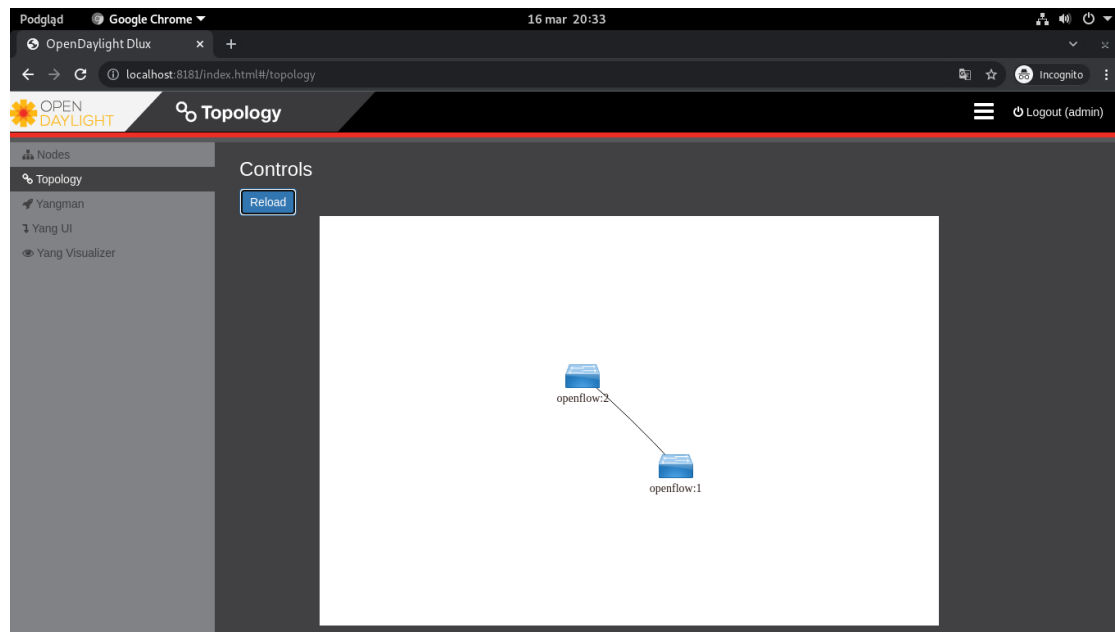
III. Konfiguracja prostej topologii sieci wykorzystującej kontroler OpenDaylight

1. Przejdź do maszyny z Mininet.
2. Uruchom liniową topologię złożoną z 2 przełączników OVS (Open vSwitch) i 8 hostów, korzystając z protokołu OpenFlow w wersji 1.3 i zdalnego kontrolera OpenDaylight (**UWAGA:** całe poniższe polecenie wprowadź jako jedną komendę):

```
sudo mn --topo=linear,2,4 --controller=remote,ip=IP_Kontrolera,port=6633  
--switch=ovs,protocols=OpenFlow13
```

UWAGA: po przecinkach w komendzie nie wstawiaj znaku spacji!
Jeśli otrzymałeś błąd „Unable to contact remote controller” sprawdź adres IP kontrolera podany w poleceniu, połączenie pomiędzy maszynami oraz czy konsola Karaf działa poprawnie!

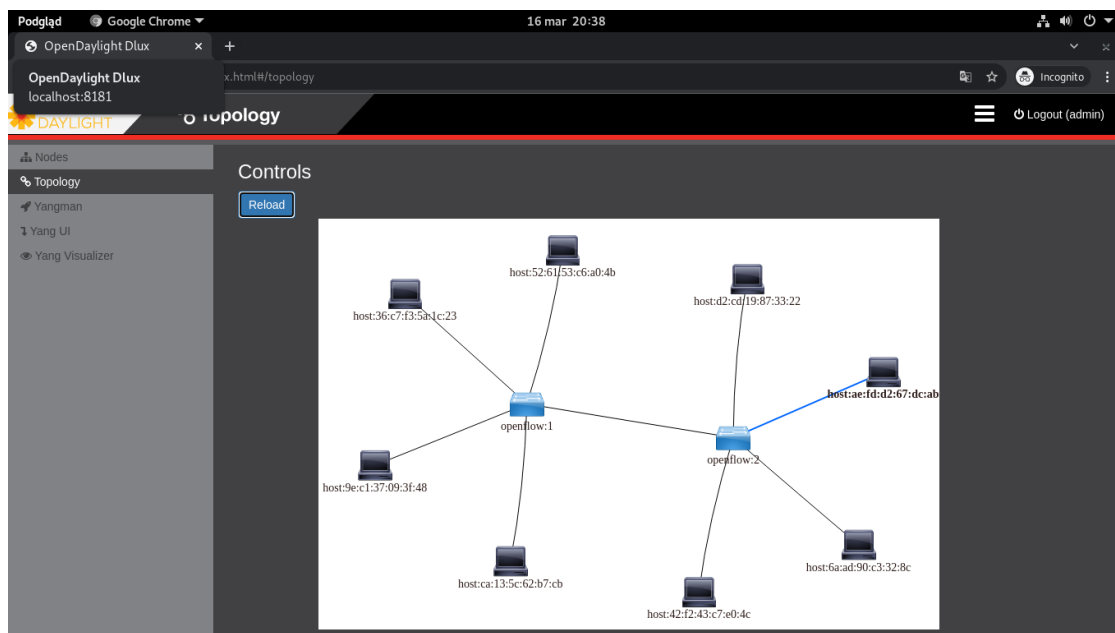
3. Wróć do maszyny OpenDaylight_Debian_12. Naciśnij przycisk Reload w zakładce Topology. Na ekranie powinny pojawić się 2 połączone przełączniki OVS:



4. Wróć do maszyny Mininet. Wykonaj polecenie

```
pingall
```

5. Wróć do przeglądarki, odśwież wizualizację topologii w DLUX i sprawdź, czy hosty są widoczne na topologii.



6. W maszynie Mininet wydaj polecenie exit, a następnie sudo mn -c.

Pamiętaj, aby przed utworzeniem nowej topologii w Mininiecie zawsze wyczyścić starą konfigurację poleceniem `sudo mn -c`.

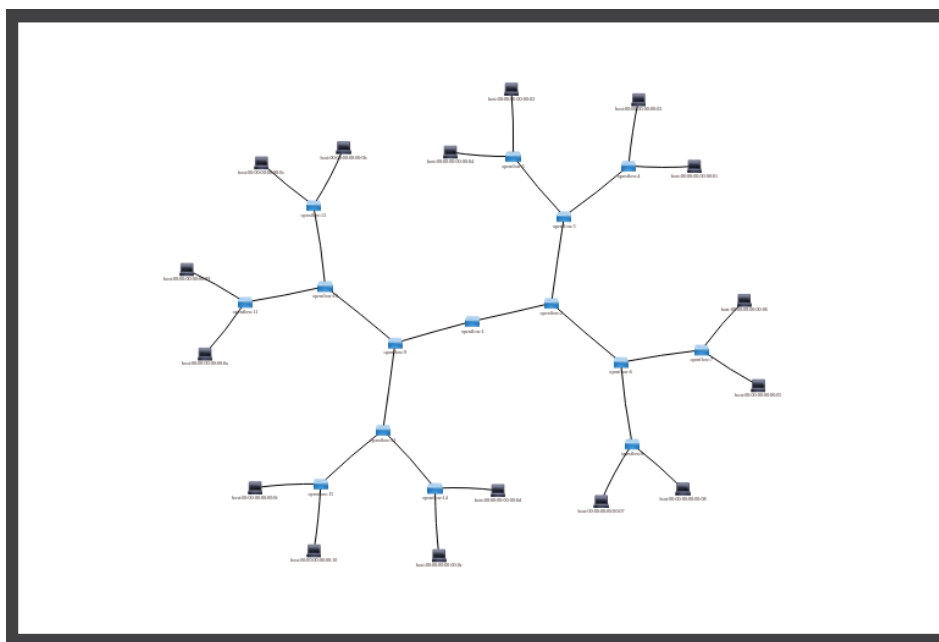
7. Modyfikując polecenie z ppkt 3.2:


- a) Stwórz hierarchiczną topologię drzewa (tree) o dowolnej liczbie gałęzi i liści. (**UWAGA:** zalecane są wartości <5 ponieważ liczba hostów i połączeń gwałtownie rośnie, co może spowodować brak pamięci RAM na maszynie).
- b) Wykorzystaj przełącznik `--mac`, aby wykorzystać proste adresy MAC dla hostów w sieci.
- c) W zależności od Twojego numeru indeksu skonfiguruj adresację przełącznikiem `--ipbase` w następujący sposób:

Numer nieparzysty: 172.16.0.0/16

Numer parzysty: 172.17.0.0/16


Pamiętaj, że hosty pojawią się na schemacie dopiero po wydaniu polecenia `pingall`.



-  8. Przybliż (myszką) wizualizację topologii i sprawdź jakie są teraz adresy MAC hostów oraz adresację w sieci. Umieść zrzut ekranu w sprawozdaniu oraz wynik polecenia `dump` na maszynie mininet.

9. Uruchom serwer `http` na jednym z hostów poleceniem:

```
ID_hosta1 python3 -m http.server 80 &
```

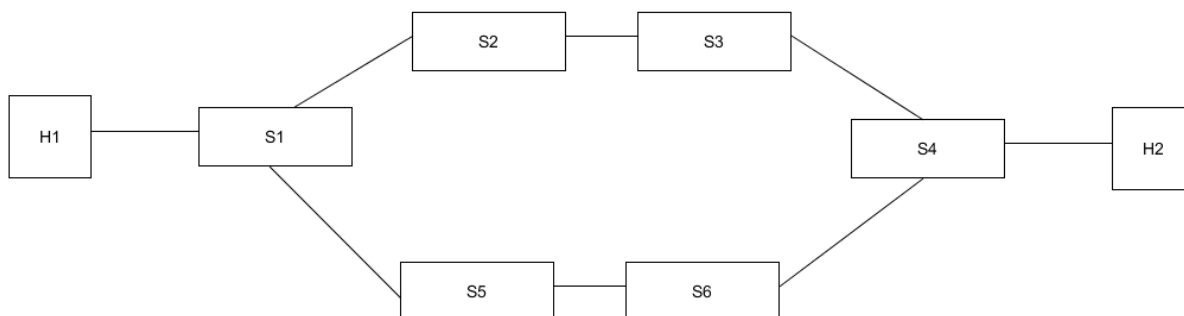
-  10. Wykonaj request `http` z innego hosta na serwer:

```
ID_hosta2 wget -O - - ID_hosta1
```

11. Usuń utworzoną topologię.

IV. Tworzenie własnej topologii i kontrola przepływu (flow control)

1. Utwórz nowy plik z rozszerzeniem `.py` (np. `moja_topologia.py`) na maszynie Mininet.
2. W pliku umieść odpowiednio zmodyfikowany skrypt do tworzenia prostej topologii, uwzględniając połączenia zgodnie ze schematem przedstawionym na rysunku.



```
from mininet.topo import Topo
class Jakas_nazwa( Topo ):
    def build( self ):
# Dodawanie przelacznikow
        switches = ['s1','s2','s3','s4','s5','s6']
        for switch in switches:
            switch=self.addSwitch(switch)
# Dodawanie hostow
        hosts = ['h1','h2']
        for host in hosts:
            host=self.addHost(host)
# Dodawanie polaczen (analogicznie dodaj brakujace polaczenia)
        self.addLink( 's1', 's2')
        ...
        ...
topos = { 'nazwaMojejTopologii': ( lambda: Jakas_nazwa() ) }
```

3. Zapisz utworzony plik.
4. Wydadź polecenie kompilujące i tworzące topologię (**UWAGA:** całe poniższe polecenie wprowadź jako jedną komendę):

```
sudo mn --custom nazwa_pliku.py --topo nazwaMojejTopologii --
controller=remote,ip=ADRES_KONTROLERA,port=6633 --mac --switch=ovs,protocols=OpenFlow13
```



5. Wydadź polecenie `pingall`, aby sprawdzić łączność w sieci, a następnie zwizualizuj topologię w `DLUX`. Umieść zrzut ekranu w sprawozdaniu.