

Trabalho – Topologia Linear com Mininet

Introdução

Este relatório descreve a criação e análise de uma topologia linear com 6 switches no Mininet.

Foram utilizadas configurações de largura de banda de **25 Mbps**, endereços MAC padronizados e o **controlador padrão do Mininet**.

O objetivo é verificar a conectividade entre os nós e realizar testes de desempenho **TCP via iperf**.

Etapas e Comandos

1. Criação da Topologia

Comando: `sudo mn --topo linear,6 --mac --link tc,bw=25`

Descrição: Cria a topologia linear com 6 switches, MACs automáticos e links limitados a 25 Mbps.

2. Inspeção de Interfaces

Comando: `dump`

Descrição: Exibe IPs, MACs e portas de cada nó (hosts e switches).

3. Teste de Conectividade

Comando: `pingall`

Descrição: Verifica comunicação entre todos os nós.

Resultado esperado: 0% dropped (30/30 received).

4. Servidor TCP (Host 1)

Comando: `h1 iperf -s -p 5555`

Descrição: Define o host 1 como servidor TCP escutando na porta 5555.

5. Cliente TCP (Host 2)

Comando: `h2 iperf -c 10.0.0.1 -p 5555 -t 15 -i 1`

Descrição: Executa teste TCP por 15 segundos, relatando throughput a cada 1 segundo (~25 Mbps esperados).

Resultados

Os testes de conectividade com o comando `pingall` apresentaram **0% de perda de pacotes**, indicando comunicação plena entre os nós.

O teste de desempenho TCP (`iperf`) demonstrou throughput médio de aproximadamente **25 Mbps**, conforme a limitação configurada na topologia.

Conclusão

Conclui-se que a topologia linear com 6 switches foi configurada com sucesso no Mininet, atendendo aos requisitos propostos.

A rede apresentou conectividade total e desempenho condizente com a largura de banda definida.

O experimento comprova a eficiência do Mininet na simulação de redes e no controle de parâmetros como banda, atraso e perda de pacotes.

PRINTS

```
mininet@mininet-vm:~$ sudo mn --topo linear,6 --mac --link tc,bw=25
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2 h3 h4 h5 h6
*** Adding switches:
s1 s2 s3 s4 s5 s6
*** Adding links:
(h1, s1) (25.00Mbit) (h2, s2) (25.00Mbit) (h3, s3) (25.00Mbit) (25.00Mbit)
(h4, s4) (25.00Mbit) (h5, s5) (25.00Mbit) (25.00Mbit) (h6, s6) (25.00Mbit) (25.00Mbit)
(s2, s1) (25.00Mbit) (s3, s2) (25.00Mbit) (s4, s3) (25.00Mbit) (25.00Mbit) (s5, s4) (25.00Mbit) (s6, s5)
*** Configuring hosts
h1 h2 h3 h4 h5 h6
*** Starting controller
c0
*** Starting 6 switches
s1 s2 s3 s4 s5 s6 ... (25.00Mbit) (25.00Mbit) (25.00Mbit) (25.00Mbit) (25.00Mbit) (25.00Mbit)
(25.00Mbit) (25.00Mbit) (25.00Mbit) (25.00Mbit) (25.00Mbit) (25.00Mbit)
*** Starting CLI:
```

```
mininet> dump
<Host h1: h1-eth0:10.0.0.1 pid=1442>
<Host h2: h2-eth0:10.0.0.2 pid=1444>
<Host h3: h3-eth0:10.0.0.3 pid=1446>
<Host h4: h4-eth0:10.0.0.4 pid=1448>
<Host h5: h5-eth0:10.0.0.5 pid=1450>
<Host h6: h6-eth0:10.0.0.6 pid=1452>
<OVSSwitch s1: lo:127.0.0.1,s1-eth1:None,s1-eth2:None pid=1457>
<OVSSwitch s2: lo:127.0.0.1,s2-eth1:None,s2-eth2:None,s2-eth3:None pid=1460>
<OVSSwitch s3: lo:127.0.0.1,s3-eth1:None,s3-eth2:None,s3-eth3:None pid=1463>
<OVSSwitch s4: lo:127.0.0.1,s4-eth1:None,s4-eth2:None,s4-eth3:None pid=1466>
<OVSSwitch s5: lo:127.0.0.1,s5-eth1:None,s5-eth2:None,s5-eth3:None pid=1469>
<OVSSwitch s6: lo:127.0.0.1,s6-eth1:None,s6-eth2:None pid=1472>
<Controller c0: 127.0.0.1:6653 pid=1435>
```

```
mininet> nodes
available nodes are:
c0 h1 h2 h3 h4 h5 h6 s1 s2 s3 s4 s5 s6
```

```
mininet> pingall
*** Ping: testing ping reachability
h1 -> h2 h3 h4 h5 h6
h2 -> h1 h3 h4 h5 h6
h3 -> h1 h2 h4 h5 h6
h4 -> h1 h2 h3 h5 h6
h5 -> h1 h2 h3 h4 h6
h6 -> h1 h2 h3 h4 h5
*** Results: 0% dropped (30/30 received)
```

```
mininet> h1 ifconfig
h1-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.1 netmask 255.0.0.0 broadcast 10.255.255.255
        ether 00:00:00:00:00:01 txqueuelen 1000 (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

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        loop txqueuelen 1000 (Local Loopback)
            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
```

```
mininet> h1 iperf -s -p 5555
```

```
-----  
Server listening on TCP port 5555  
TCP window size: 85.3 KByte (default)
```

```
mininet> h2 iperf -c 10.0.0.1 -p 5555 -t 15 -i 1
```

```
-----  
Client connecting to 10.0.0.1, TCP port 5555  
TCP window size: 85.3 KByte (default)
```

```
[ 3] local 10.0.0.2 port 52332 connected with 10.0.0.1 port 5555  
[ ID] Interval Transfer Bandwidth  
[ 3] 0.0- 1.0 sec 3.00 MBytes 25.2 Mbits/sec  
[ 3] 1.0- 2.0 sec 2.88 MBytes 24.1 Mbits/sec  
[ 3] 2.0- 3.0 sec 2.88 MBytes 24.1 Mbits/sec  
[ 3] 3.0- 4.0 sec 2.88 MBytes 24.1 Mbits/sec  
[ 3] 4.0- 5.0 sec 2.88 MBytes 24.1 Mbits/sec  
[ 3] 5.0- 6.0 sec 2.88 MBytes 24.1 Mbits/sec  
[ 3] 6.0- 7.0 sec 2.88 MBytes 24.1 Mbits/sec  
[ 3] 7.0- 8.0 sec 2.75 MBytes 23.1 Mbits/sec  
[ 3] 8.0- 9.0 sec 2.88 MBytes 24.1 Mbits/sec  
[ 3] 9.0-10.0 sec 2.88 MBytes 24.1 Mbits/sec  
[ 3] 10.0-11.0 sec 2.88 MBytes 24.1 Mbits/sec  
[ 3] 11.0-12.0 sec 2.75 MBytes 23.1 Mbits/sec  
[ 3] 12.0-13.0 sec 2.88 MBytes 24.1 Mbits/sec  
[ 3] 13.0-14.0 sec 2.88 MBytes 24.1 Mbits/sec  
[ 3] 14.0-15.0 sec 2.88 MBytes 24.1 Mbits/sec  
[ 3] 0.0-15.0 sec 43.0 MBytes 24.0 Mbits/sec
```