Comunicações Móveis

Projeto SDN 1

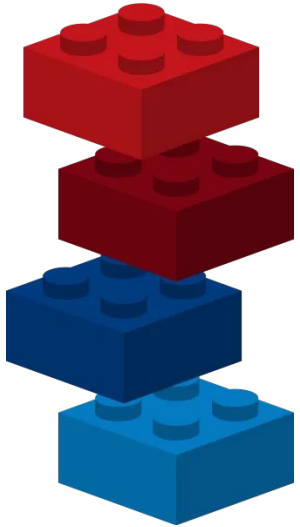
Bruno Lemos	98221
Tiago Marques	98459
João Viegas	98372

universidade de aveiro
theoria poiesis praxis

Tutoriais

Tutoriais do faucet :

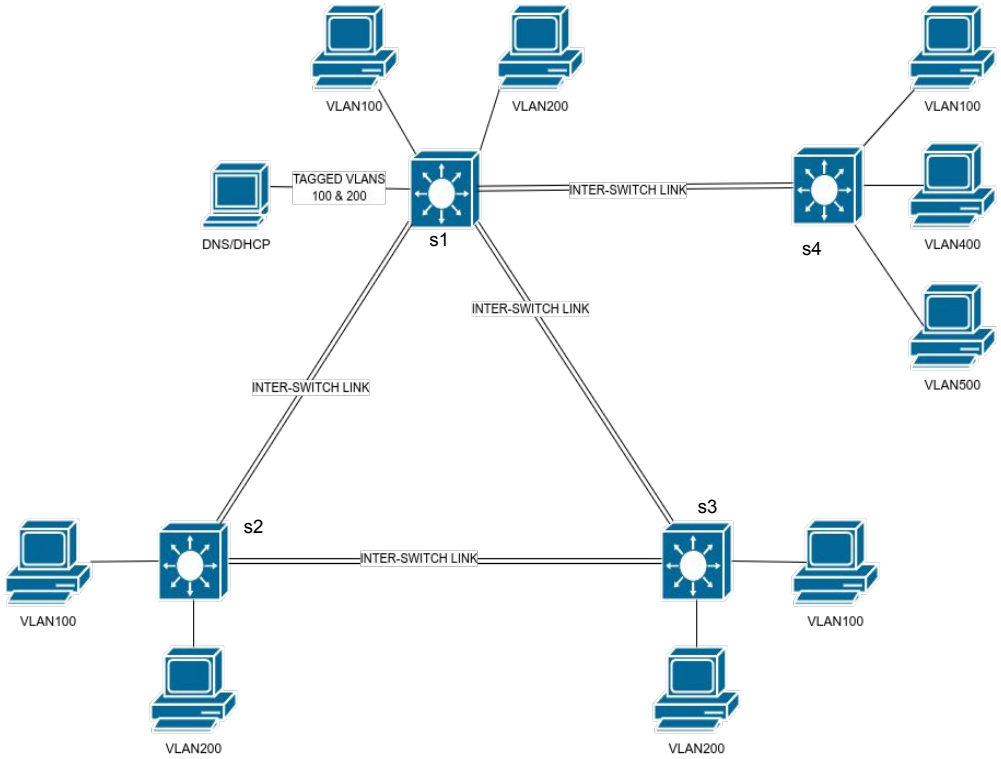
- ACL
- Vlan
- Routing
- Connection Tracking
- Stacking
- NFV Services



Tutoriais do OVS :

- Faucet
- IPSEC

Network Topology



- Temos 4 vlans
- Loop entre os switches s1, s2 e s3
- 1 Host que serve como DNS/DHCP server
- Uso de ACLs
- Stacking (inter-switch link)
- Todos os links switch<->host são nativos menos o link s1<->DNS/DHCP que é um link ‘tagged’ para permitir as conexões do DNS/DHCP com as vlans

Connection Tracking

- Manter o controlo sobre as ligações de rede ativas
- Criação de uma tabela de conexões com as informações
  - Endereço IP origem e destino
  - Número do porto de origem e destino
  - protocolo de rede utilizado

Fluxo

1. Pacote de rede enviado
2. Verificar tabela de conexões
3. Se a conexão não existir na tabela adicionar conexão
4. Pacote enviado pelo destino correto

Connection Tracking - Nat

```
conntrack_fw:
    # Permit all ARP traffic such that hosts can resolve one another's MACs
    - rule:
        eth_type: 0x0806 #arp
        actions:
            allow: True
    # Begin tracking ALL untrackked IPv4 connections
    - rule:
        eth_type: 0x0800  # ipv4
        ct_state: 0/0x20 # match -trk (untracked)
        actions:
            # Re-inject tracked packet into the OP pipeline, containing additional connection
            # metadata, to default table 0. The tracked packet is again evaluated by Faucet ACLs
            # in table 0. The original, untracked packet is effectively dropped
            ct:
                zone: 10 # arbitrary conntrack zone ID to match agains later
                table: 0
    # Commit new ipv4 connection from host44 to host45
    - rule:
        eth_type: 0x0800 # ipv4
        ipv4_src: 10.0.4.44
        ipv4_dst: 10.0.5.45
        ct_state: 0x21/0x21 # match +new - packet to establish a new connection
        actions:
            # Commit the connection to the connection tracking module whicj will be stored
            # beyond the lifetime of packet in the pipeline
            ct:
                zone: 10 # the same conntrack zone ID as above
                flags: 1 # commit the new connection
                table: 1 # implicit allow the new connection packet via faucet table 1
                nat: #sNAT the connection to the faucet VIP
                    flags: 1
                    range_ipv4_min: 10.0.4.254
                    range_ipv4_max: 10.0.4.254
    # Allow packets in either direction from existing connections initiated by host44
    - rule:
        eth_type: 0x0800 # ipv4
        ct_zone: 10 # match packets associated with our conntrack zone ID
        ct_state: 0x22/0x22 # match +est - packets in an established connection
        actions:
            ct:
                zone: 10
                flags: 1
                table: 1
                nat:
                    flags: 1
    - rule:
        eth_type: 0x0800 # ipv4
        ipv4_src: 10.0.5.45
        ipv4_dst: 10.0.4.44
        actions:
            allow: False
```

"Node: h44"

```
root@tiago-VirtualBox:/home/tiago/Desktop/CM_SDN-master# ping -c 5 10.0.5.45
PING 10.0.5.45 (10.0.5.45) 56(84) bytes of data.
64 bytes from 10.0.5.45: icmp_seq=1 ttl=63 time=0.270 ms
64 bytes from 10.0.5.45: icmp_seq=2 ttl=63 time=0.360 ms
64 bytes from 10.0.5.45: icmp_seq=3 ttl=63 time=0.109 ms
64 bytes from 10.0.5.45: icmp_seq=4 ttl=63 time=0.101 ms
64 bytes from 10.0.5.45: icmp_seq=5 ttl=63 time=0.118 ms

--- 10.0.5.45 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4072ms
rtt min/avg/max/mdev = 0.101/0.191/0.360/0.104 ms
```

Interface h44<->s4

```
3 6.228039390   10.0.4.44    10.0.5.45    ICMP   98 Echo (ping) request  id=0x15b6, seq=1/256, ttl=64 (reply in 4)
4 6.228594841   10.0.5.45    10.0.4.44    ICMP   98 Echo (ping) reply    id=0x15b6, seq=1/256, ttl=63 (request in…
```

Interface s4<->h45

```
1 0.000000000   10.0.4.254   10.0.5.45    ICMP   98 Echo (ping) request  id=0x15b6, seq=1/256, ttl=63 (reply in 2)
2 0.000060062   10.0.5.45    10.0.4.254   ICMP   98 Echo (ping) reply    id=0x15b6, seq=1/256, ttl=64 (request in…
```

Stacking

- Combinar múltiplos modelos com fim a obter um único modelo
  - Gerir vários dispositivos a partir de um único dispositivo
- Os vários dispositivos comunicam entre si e compartilham informações do tipo:
  - Configuração
  - Estado
  - Gerenciamento
- Usa LLDP (Link Layer Discovery Protocol) para conhecer a topologia entre os switches

Benefícios

➔ Administradores de rede fazem a gestão a partir de um único dispositivo
➔ Melhor utilização dos recursos e desempenho da rede
➔ Aumentar a robustez da rede

Stacking - How it works

O Faucet manda pacotes LLDP aos diversos OVS switches, de forma a descobrir os diversos ‘stack’ ports e conectá-los.

```
faucet.valve INFO     DPID 3 (0x3) s3 LLDP on 0e:00:00:00:00:01, Port 3 from 0e:00:00:00:00:01 (remote DPID 1 (0x1), port 4) state INITIALIZING
faucet.valve INFO     DPID 2 (0x2) s2 LLDP on 0e:00:00:00:00:01, Port 3 from 0e:00:00:00:00:01 (remote DPID 1 (0x1), port 3) state INITIALIZING
faucet.valve INFO     DPID 4 (0x4) s4 LLDP on 0e:00:00:00:00:01, Port 4 from 0e:00:00:00:00:01 (remote DPID 1 (0x1), port 5) state INITIALIZING
faucet.valve INFO     DPID 1 (0x1) s1 LLDP on 0e:00:00:00:00:01, Port 3 from 0e:00:00:00:00:01 (remote DPID 2 (0x2), port 3) state INITIALIZING
faucet.valve INFO     DPID 1 (0x1) s1 LLDP on 0e:00:00:00:00:01, Port 5 from 0e:00:00:00:00:01 (remote DPID 4 (0x4), port 4) state INITIALIZING
faucet.valve INFO     DPID 1 (0x1) s1 LLDP on 0e:00:00:00:00:01, Port 4 from 0e:00:00:00:00:01 (remote DPID 3 (0x3), port 3) state INITIALIZING
faucet.valve INFO     DPID 2 (0x2) s2 LLDP on 0e:00:00:00:00:01, Port 4 from 0e:00:00:00:00:01 (remote DPID 3 (0x3), port 4) state INITIALIZING
faucet.valve INFO     DPID 3 (0x3) s3 LLDP on 0e:00:00:00:00:01, Port 4 from 0e:00:00:00:00:01 (remote DPID 2 (0x2), port 4) state INITIALIZING
faucet.valve INFO     DPID 1 (0x1) s1 LLDP on 0e:00:00:00:00:01, Port 3 from 0e:00:00:00:00:01 (remote DPID 2 (0x2), port 3) state UP
faucet.valve INFO     DPID 1 (0x1) s1 LLDP on 0e:00:00:00:00:01, Port 4 from 0e:00:00:00:00:01 (remote DPID 3 (0x3), port 3) state UP
faucet.valve INFO     DPID 1 (0x1) s1 LLDP on 0e:00:00:00:00:01, Port 5 from 0e:00:00:00:00:01 (remote DPID 4 (0x4), port 4) state UP
faucet.valve INFO     DPID 2 (0x2) s2 LLDP on 0e:00:00:00:00:01, Port 3 from 0e:00:00:00:00:01 (remote DPID 1 (0x1), port 3) state UP
faucet.valve INFO     DPID 2 (0x2) s2 LLDP on 0e:00:00:00:00:01, Port 4 from 0e:00:00:00:00:01 (remote DPID 3 (0x3), port 4) state UP
faucet.valve INFO     DPID 4 (0x4) s4 LLDP on 0e:00:00:00:00:01, Port 4 from 0e:00:00:00:00:01 (remote DPID 1 (0x1), port 5) state UP
faucet.valve INFO     DPID 3 (0x3) s3 LLDP on 0e:00:00:00:00:01, Port 3 from 0e:00:00:00:00:01 (remote DPID 1 (0x1), port 4) state UP
faucet.valve INFO     DPID 3 (0x3) s3 LLDP on 0e:00:00:00:00:01, Port 4 from 0e:00:00:00:00:01 (remote DPID 2 (0x2), port 4) state UP
```

```
64 104.597321636 0e:00:00:00:00:01    LLDP_Multicast      LLDP       61 MA/0e:00:00:00:00:01 IN/3 15 SysN=s1
65 104.598948772 0e:00:00:00:00:01    LLDP_Multicast      LLDP       61 MA/0e:00:00:00:00:01 IN/3 15 SysN=s2
▸ Frame 65: 61 bytes on wire (488 bits), 61 bytes captured (488 bits) on interface s1-eth3, id 0
▸ Ethernet II, Src: 0e:00:00:00:00:01 (0e:00:00:00:00:01), Dst: LLDP_Multicast (01:80:c2:00:00:0e)
▾ Link Layer Discovery Protocol
    ▸ Chassis Subtype = MAC address, Id: 0e:00:00:00:00:01
    ▸ Port Subtype = Interface name, Id: 3
    ▸ Time To Live = 15 sec
    ▸ System Name = s2
    ▸ Port Description = s2 to s1
    ▸ Unknown - Unknown (1)
    ▸ Unknown - Unknown (2)
    ▸ End of LLDPDU
▸ VSS Monitoring Ethernet trailer, Source Port: 0
```

Graças ao ‘stacking’, o faucet é capaz de detectar topologias e mudanças na mesma, consequentemente, o controlador possui conhecimento suficiente para calcular a spanning tree sem precisar de correr o protocolo STP (Spanning Tree Protocol).

NFV Services

- Substitui routers, firewalls e switches por um software executado em máquinas virtuais
- Os serviços de rede utilizam hardware genérico para executar funções de rede específicas
- Pode ser usada para implementar firewalls, IDS/IPS, load balancers, entre outros

Benefícios

➔ Flexibilidade no hardware
➔ Simplificar gestão e manutenção do sistema da rede
➔ Controlo centralizado

NFV Services - DNS/DHCP

Set up DHCP

1. Criamos instâncias dentro do namespace do host DNS/DHCP, uma para cada vlan
2. Usamos o comando dhcp para atribuir o ip e gateway aos nossos hosts

```
Listening on LPF/h12-eth1/00:00:00:00:00:03
Sending on   LPF/h12-eth1/00:00:00:00:00:03
Sending on   Socket/fallback
DHCPDISCOVER on h12-eth1 to 255.255.255.255 port 67 interval 3 (xid=0xfe3d9e2d)
DHCPDISCOVER on h12-eth1 to 255.255.255.255 port 67 interval 5 (xid=0xfe3d9e2d)
DHCPOFFER of 10.0.2.10 from 10.0.2.1
DHCPREQUEST for 10.0.2.10 on h12-eth1 to 255.255.255.255 port 67 (xid=0x2d9e3dfe)
DHCPACK of 10.0.2.10 from 10.0.2.1 (xid=0xfe3d9e2d)
bound to 10.0.2.10 -- renewal in 1502 seconds.
```

Set up DNS

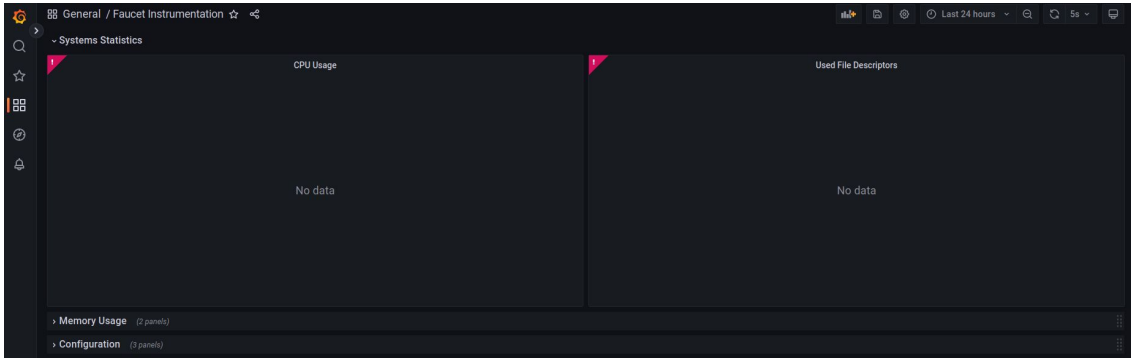
1. Atribuímos um nome DNS ao namespace do host DNS/DHCP
2. Fazendo uso das ACLs do faucet para reencaminhar todos os pacotes DNS para o nosso DNS/DHCP

Grafana

Grafana é uma ferramenta de visualização de dados e monitorização.

- Permite criar gráficos e dashboards para serem analisados em tempo real.
- É muito utilizado em aplicações de monitorização de rede e análise de dados.

É uma excelente ferramenta para usar em paralelo com o FAUCET apesar de não termos conseguido obter resultados relevantes.



Poseidon

- O que é?
  - Projeto open source para permitir monitorização de redes em SDN
  - Faz uso da monitorização da rede e, juntamente com um algoritmo de inteligência artificial, tenta optimizar a utilização de recursos da rede.

- O que conseguimos? Problemas
  - Comandos do script usado para fazer a instalação do software não têm o efeito deseja
  - Alguns comandos do script não estavam sequer implementados
  - Não conseguimos colocar o Poseidon a funcionar por falta de suporte e informação.

SDN APP

- Estabelecimento de conexão com o controlador (HTTP)
- Envio de informação de uma aplicação (Python)

```python
import requests

# Endereço do controlador Faucet
controller_address = "http://0.0.0.0:6653"

# Dados da regra de encaminhamento a ser adicionada
data = {
    "table_id": 0,
    "priority": 1,
    "match": {},
    "actions": ["CONTROLLER"]
}

# Envia a solicitação POST para o controlador Faucet
response = requests.post(controller_address + "/rules", json=data)

# Exibe a resposta do controlador Faucet
print(response.text)
```