Consortium GARR

8° BORSISTI DAY
"ORIO CARLINI"
MERCOLEDÌ 14 DICEMBRE 2016 - ROMA

Sicurezza per il Data Center basato sul Software Defined Networking (SDN)

Alessandro Mauceri

Borsista GARR Bando 01/15 presso l'INFN di Catania

Tutor Dr. Giuseppe Andronico

INFN
Istituto Nazionale di Fisica Nucleare
Sezione di Catania

Indice

- Obiettivi della ricerca
- Attività di ricerca svolta
- Obiettivi raggiunti
- Attività proposta per il II anno di borsa

Obiettivi della ricerca

- Utilizzare SDN per affrontare problemi di sicurezza all'interno di un infrastruttura di rete come quella della Sezione INFN di Catania
- Implementare una topologia di rete all'interno della quale attuare un sistema di sicurezza minimale sfruttando i paradigmi SDN, NFV e Cloud computing
- Rilevare e mitigare attacchi informatici all'interno del Data Center tramite controller SDN

L'attività di ricerca

- **Trimestre I**
  - **Sicurezza nei Data Center**
  - **Approfondimento Software Defined Networking (SDN)**
- **Trimestre II**
  - Comprensione e integrazione OpenStack - OpenDaylight
  - Approfondimento specifiche ETSI NFV e ricerca tools/frameworks
  - Realizzazione infrastruttura di test
- **Trimestre III/IV**
  - Scenario SYN flood attack
  - Scenario Brute-force attack
  - Consolidamento NFV per realizzare scenari più complessi

I trimestre

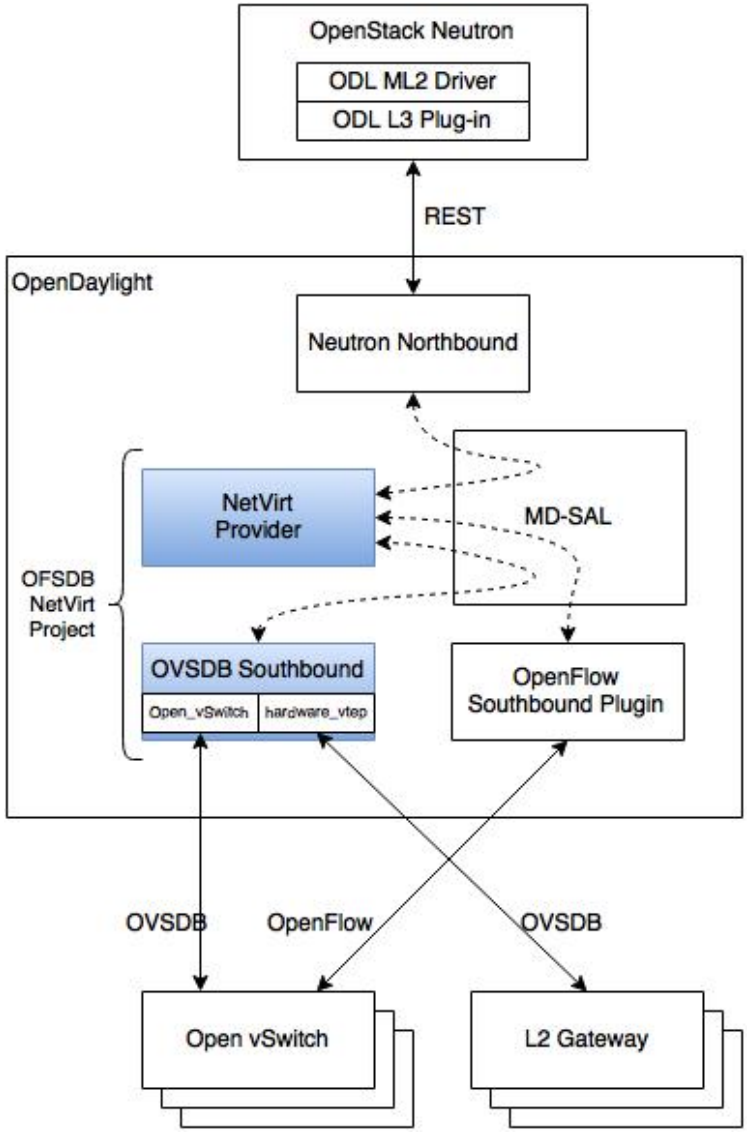
- **Sicurezza nei Data Center**
  - Principali tecniche d'attacco (i.e. attacchi DoS, spoofing, sniffing, brute-force)
  - Principali tecniche di difesa (i.e. firewall, DPI, IPS/IDS, crittografia)
- **Approfondimento SDN**
  - Valutazione frameworks di sviluppo controllers SDN (i.e. OpenDaylight, ONOS)
  - Virtual Tenant Network (VTN)
  - TSDR Data Store

L'attività di ricerca

- **Trimestre I**
  - Sicurezza nei Data Center
  - Approfondimento Software Defined Networking (SDN)
- **Trimestre II**
  - **Comprensione e integrazione OpenStack - OpenDaylight**
  - **Approfondimento specifiche ETSI NFV e ricerca tools/frameworks**
  - **Realizzazione infrastruttura di test**
- **Trimestre III/IV**
  - Scenario SYN flood attack
  - Scenario Brute-force attack
  - Consolidamento NFV per realizzare scenari più complessi

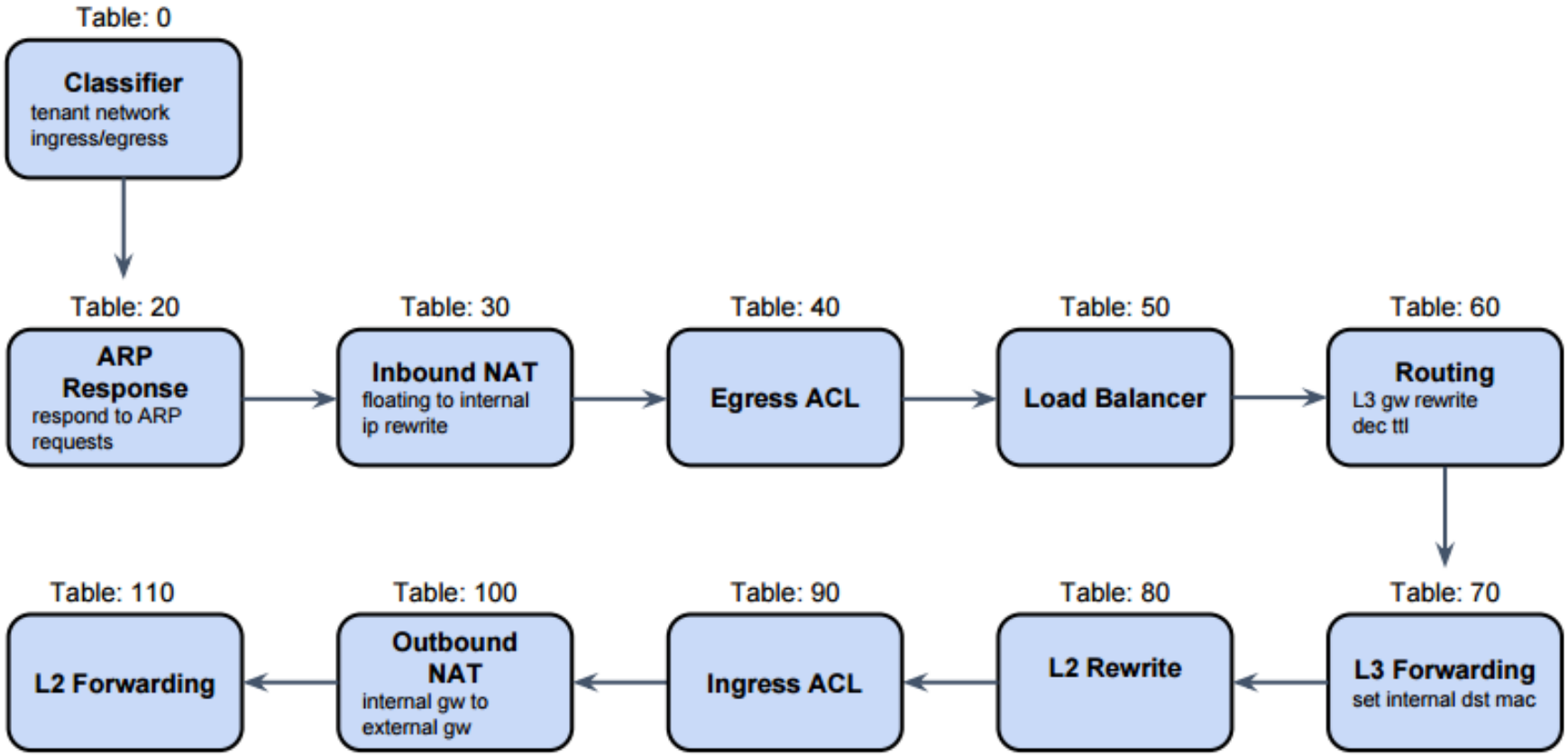
Integrazione OpenStack - OpenDaylight

- **OpenStack Mitaka**
  - Networking-odl
    - ✓ ODL ML2 Driver
    - ✓ ODL L3 plugin
    - ✓ Drivers for LBaaS,FWaaS, VPNaaS…
    - ✓ Networking-sfc Driver
- **OpenDaylight Beryllium**
  - Neutron northbound
  - OpenStack service providers
    - ✓ ovsdb/netvirt
    - ✓ VTN manager
  - Southbound protocol
    - ✓ OpenFlow
    - ✓ OVSDB



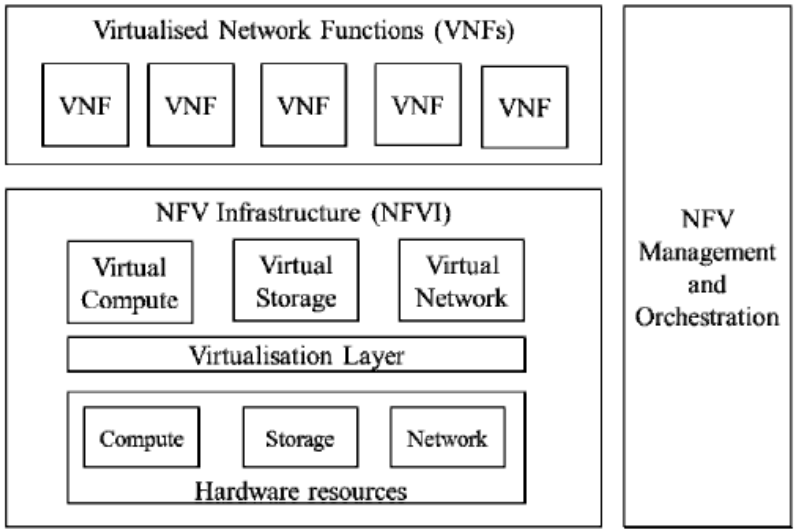
OVSDB NetVirt Architecture

Comprensione della NetVirt Flow Pipeline



Approfondimento NFV

- Visione specifiche ETSI NFV
  - ETSI NFV Architecture Framework
  - ETSI NFV Management and Orchestration (MANO)
  - ETSI NFV Software Architecture

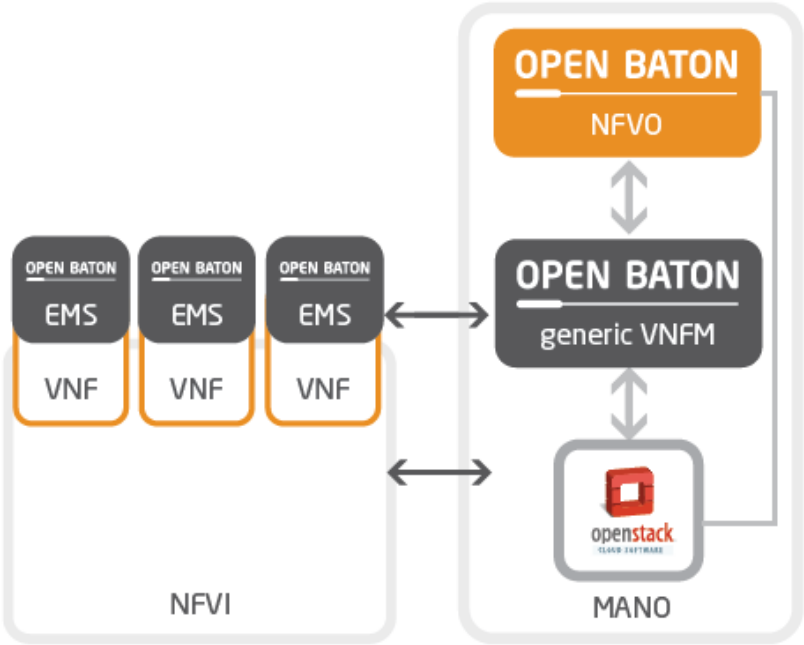


- Valutazione frameworks implementazione MANO
  - **Tacker:** è un progetto OpenStack che fornisce un generico VNF Manager (VNFM) e un NFV Orchestrator (NFVO) per distribuire e gestire funzioni di rete virtuali (VNFs)
  - **Open Source MANO (OSM):** set di tools che coprono i vari componenti dell'architettura MANO
  - **Open Baton:** sviluppato da Fraunhofer FOKUS è una implementazione highly compliant con le specifiche ETSI MANO. Permette il monitoring delle VNFs attraverso Zabbix

Overview Open Baton

Open Baton: implementazione Open Source delle specifiche ETSI MANO, propone di promuovere, all'interno di un ambiente NFV, l'integrazione tra VNF e Cloud Infrastructure providers

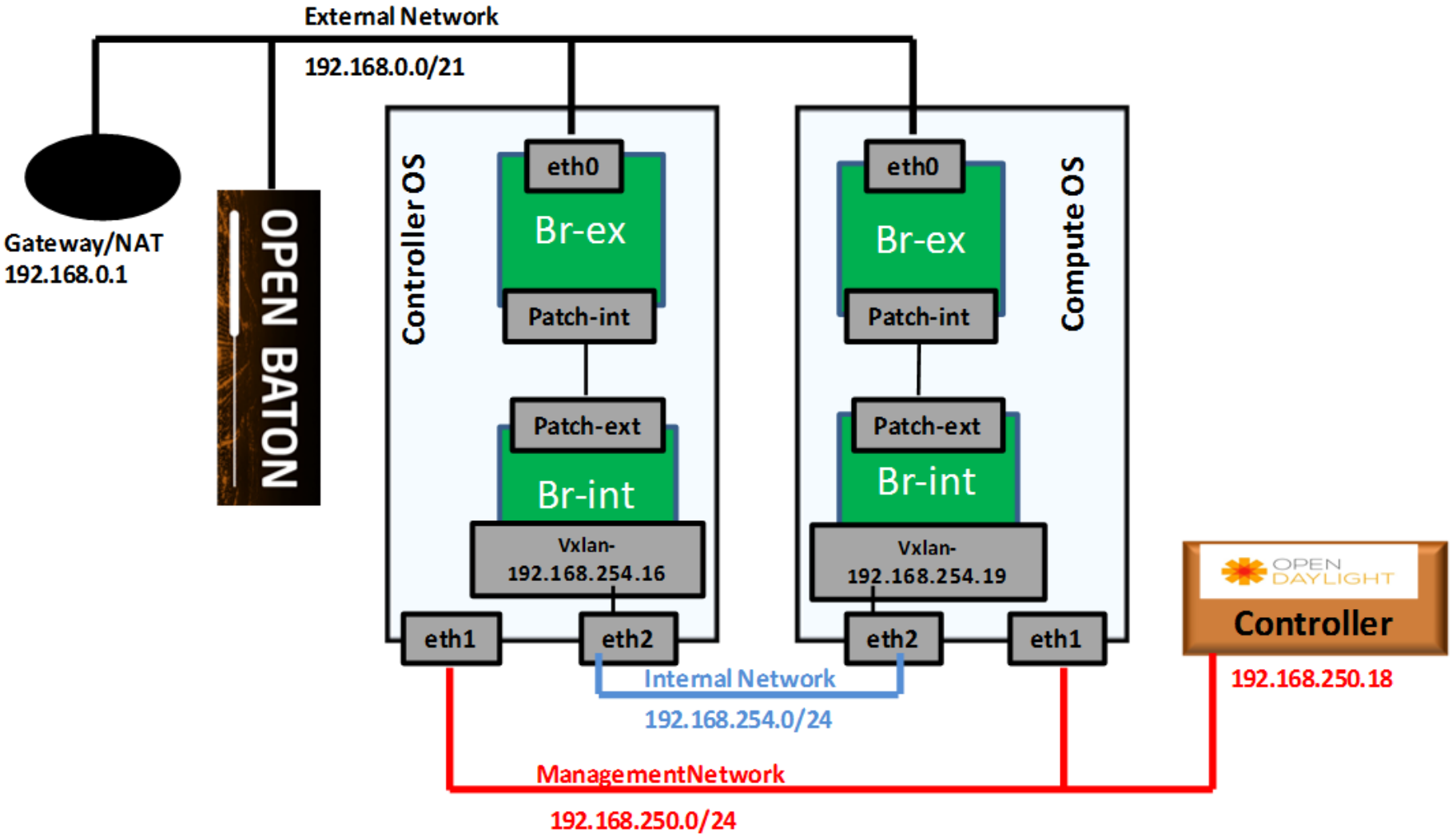
- **Un NFV Orchestrator**: coordinazione e allocazione delle risorse richieste per istanziare le VNFs
- **Un generico VNF Manager**: gestisce il ciclo di vita delle VNFs (create/update/delete)
- **Un set di librerie**: per sviluppare un proprio VNFM
- **Runtime operations**: fault management, autoscaling
- **Un plugin OpenStack**: per usare OpenStack come Virtualized Infrastructure Manager (VIM)




```
- Metadata.yaml
- vnfd.json
- scripts/
    - 1_script.sh
    - 2_script.sh
```

VNF Package structure

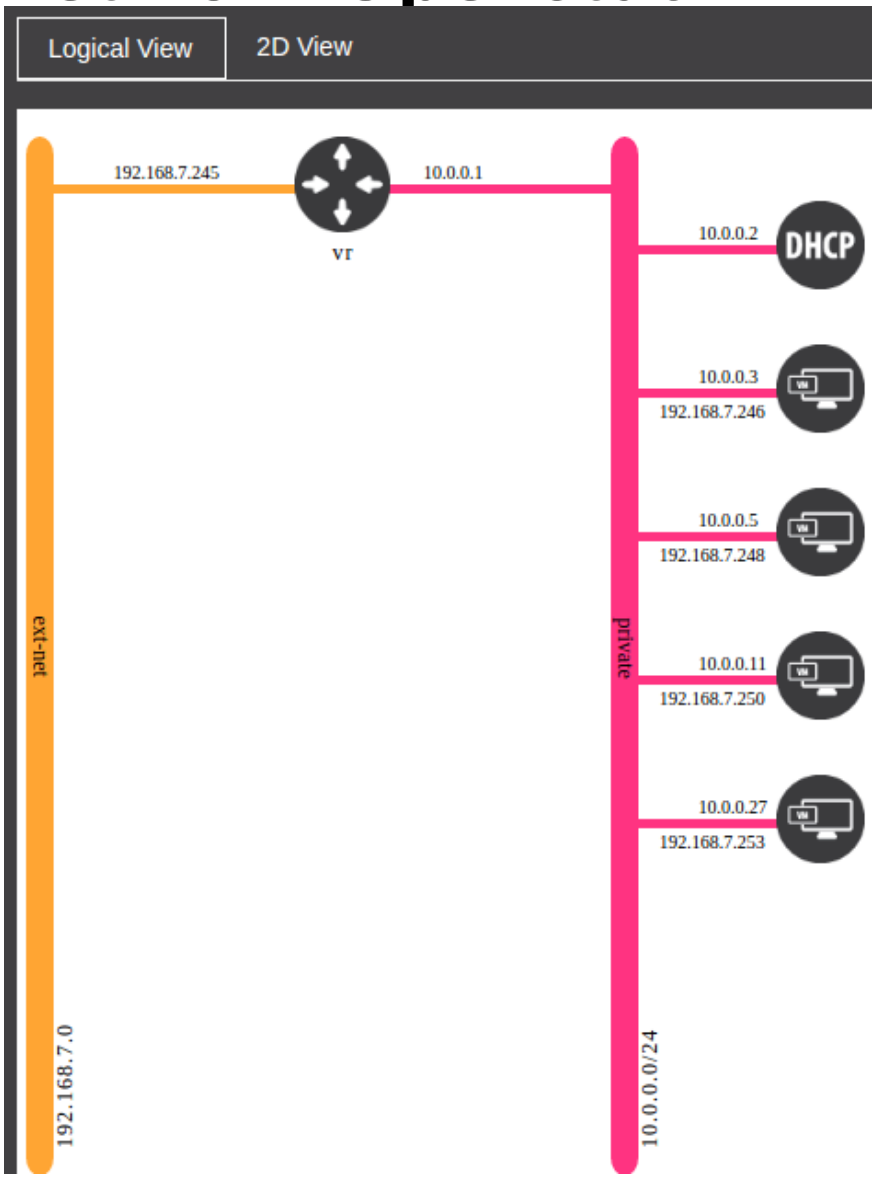
Infrastruttura di test



Infrastruttura di test: Network Openstack

Private-network (10.0.0.0/24)

External-network (192.168.0.0/21)

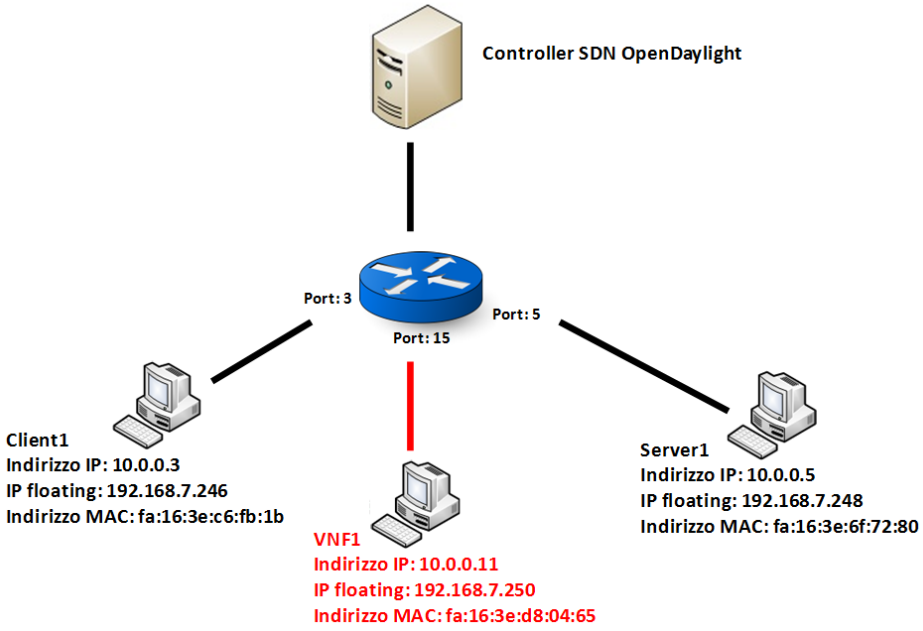


L'attività di ricerca

- **Trimestre I**
  - Sicurezza nei Data Center
  - Approfondimento Software Defined Networking (SDN)
- **Trimestre II**
  - Comprensione e integrazione OpenStack - OpenDaylight
  - Approfondimento specifiche ETSI NFV e ricerca tools/frameworks
  - Realizzazione infrastruttura di test
- **Trimestre III/IV**
  - **Scenario SYN flood attack**
  - **Scenario Brute-force attack**
  - **Consolidamento NFV per realizzare scenari più complessi**

Scenario SYN flood attack (1)

- **Br-int:** OVS bridge al quale vengono iniettate le flow-entries tramite controller SDN
- **Client1:** macchina in cui viene eseguito uno script che permette di generare un SYN flood attack
- **Server1:** macchina vittima
- **1 analizzatore di traffico:** VNF in cui è in esecuzione uno script che permette di:
  - analizzare il traffico
  - riconoscere un SYN flood attack, l'indirizzo IP della macchina attaccante e vittima
  - Iniettare, in real-time, la regola che permette, sopra una certa soglia di connessioni, di bloccare il flusso (tramite API RESTConf)



Flag SYN!=0 Flag ACK=0

tcpdump -i eth0 -n -c 100 **'tcp[tcpflags] & (tcp-syn) != 0'** and **'tcp[tcpflags] & (tcp-ack) == 0'**

Scenario SYN flood attack (2)

```
stack@localhost:~/devstack$ sudo ovs-ofctl -O OpenFlow13 dump-flows br-int | grep fa:16:3e:6f:72:80
 cookie=0x0, duration=348256.398s, table=0, n_packets=151300106, n_bytes=8170215884, in_port=5,dl_src=fa:16:3e:6f:72:80 actions=set_field:0x447->tun_id,load:0x1->NX
 cookie=0x0, duration=2882.347s, table=20, n_packets=1821, n_bytes=76482, priority=1024,arp,tun_id=0x447,arp_tpa=10.0.0.5,arp_op=1 actions=move:NXM_OF_ETH_SRC[]->NX
0->eth_src,load:0x2->NXM_OF_ARP_OP[],move:NXM_NX_ARP_SHA[]->NXM_NX_ARP_THA[],move:NXM_OF_ARP_SPA[]->NXM_OF_ARP_TPA[],load:0xfa163e6f7280->NXM_NX_ARP_SHA[],load:0xa0
 cookie=0x0, duration=348256.535s, table=40, n_packets=0, n_bytes=0, priority=61010,arp,arp_sha=fa:16:3e:6f:72:80 actions=goto_table:50
 cookie=0x0, duration=348256.235s, table=40, n_packets=151298227, n_bytes=8170133954, priority=61007,ip,dl_src=fa:16:3e:6f:72:80 actions=goto_table:50
 cookie=0x0, duration=348256.518s, table=40, n_packets=0, n_bytes=0, priority=36001,ip,in_port=5,dl_src=fa:16:3e:6f:72:80,nw_src=10.0.0.5 actions=goto_table:50
 cookie=0x0, duration=2882.062s, table=70, n_packets=152399908, n_bytes=8240673409, priority=1024,ip,tun_id=0x447,nw_dst=10.0.0.5 actions=set_field:fa:16:3e:6f:72:8
 cookie=0x0, duration=348256.549s, table=90, n_packets=0, n_bytes=0, priority=61010,arp,arp_tha=fa:16:3e:6f:72:80 actions=goto_table:100
 cookie=0x0, duration=1129.813s, table=110, n_packets=151319557, n_bytes=8182331586, priority=16384,tun_id=0x447,dl_dst=fa:16:3e:6f:72:80 actions=output:5,output:15
stack@localhost:~/devstack$
```

Lista flow-entries nel br-int. Il flusso destinato alla macchina vittima viene dirottato alla VNF1

```
root@client1:/opt# ls
synflood.pl  udpflood.pl
root@client1:/opt# perl synflood.pl
Usage: synflood.pl <source host> <source port> <dest host> <dest port>
root@client1:/opt# perl synflood.pl 10.0.0.3 80 10.0.0.5 80
```

Esecuzione SYN flood attack da parte dell'utente malevolo

```
IP macchina attaccante 10.0.0.3.80
IP macchina attaccata 10.0.0.5.80:
SYN flood attack riconosciuto
iniettata flow-enty drop1
```

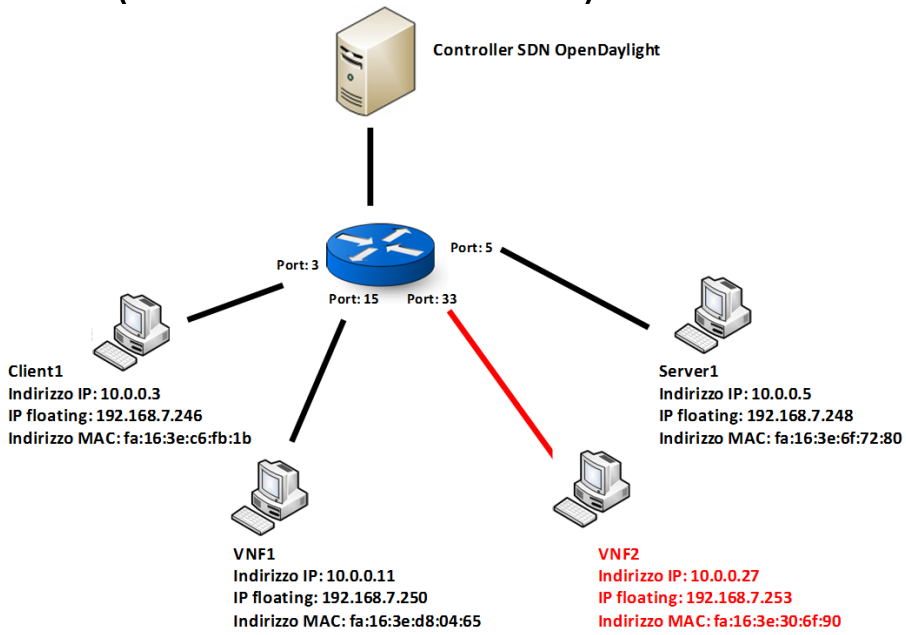
Output script in esecuzione nella VNF1

```
stack@localhost:~/devstack$ sudo ovs-ofctl -O OpenFlow13 dump-flows br-int | grep 10.0.0.5
 cookie=0x0, duration=5210.284s, table=20, n_packets=1836, n_bytes=77112, priority=1024,arp,tun_id=0x447,arp_tpa=10.0.0.5,arp_op=1 actions=move
0->eth_src,load:0x2->NXM_OF_ARP_OP[],move:NXM_NX_ARP_SHA[]->NXM_NX_ARP_THA[],move:NXM_OF_ARP_SPA[]->NXM_OF_ARP_TPA[],load:0xfa163e6f7280->NXM_N
 cookie=0x0, duration=350560.597s, table=30, n_packets=155, n_bytes=19799, priority=1024,ip,in_port=2,nw_dst=192.168.7.248 actions=set_field:10
 cookie=0x0, duration=350584.456s, table=40, n_packets=0, n_bytes=0, priority=36001,ip,in_port=5,dl_src=fa:16:3e:6f:72:80,nw_src=10.0.0.5 actio
 cookie=0x0, duration=5210s, table=70, n_packets=154312591, n_bytes=8343960491, priority=1024,ip,tun_id=0x447,nw_dst=10.0.0.5 actions=set_field
 cookie=0x0, duration=350560.591s, table=100, n_packets=117, n_bytes=15682, priority=512,ip,tun_id=0x447,dl_dst=fa:16:3e:66:af:e5,nw_src=10.0.0
eld:82:02:90:8e:e2:2e->eth_dst,set_field:192.168.7.248->ip_src,output:2
 cookie=0x0, duration=171.398s, table=110, n_packets=251185, n_bytes=13563990, priority=16389,ip,nw_src=10.0.0.3,nw_dst=10.0.0.5 actions=drop
stack@localhost:~/devstack$
```

Lista flow-entries nel br-int. Il flusso che ha come sorgente 10.0.0.3 viene droppato

Scenario Brute-force attack (1)

- **1 secondo analizzatore di traffico:** VNF2 in cui è in esecuzione uno script che permette di:
  - analizzare il traffico
  - riconoscere un SSH Brute-force attack, l'indirizzo IP della macchina attaccante e vittima
  - Iniettare, in real-time, la regola che permette, sopra una certa soglia di connessioni ed un certo intervallo temporale di bloccare il flusso (tramite API RESTConf)



timeout 20 tcpdump -i eth0 **'dst port 22** 'and **'tcp[tcpflags] & (tcp-syn) != 0'**

Scenario Brute-force attack (2)

```
stack@localhost:~/devstack$ sudo ovs-ofctl -O OpenFlow13 dump-flows br-int | grep fa:16:3e:6f:72:80
 cookie=0x0, duration=4227732.644s, table=0, n_packets=162891143, n_bytes=8796275491, in_port=5,dl_src=fa:16:3e:6f:72:80 actions=set_field:0x447->tun_id,load:0x1->NXM_NX_REG
 cookie=0x0, duration=67365.332s, table=20, n_packets=2246, n_bytes=94332, priority=1024,arp,tun_id=0x447,arp_tpa=10.0.0.5,arp_op=1 actions=move:NXM_OF_ETH_SRC[]->NXM_OF_ETH_
80->eth_src,load:0x2->NXM_OF_ARP_OP[],move:NXM_NX_ARP_SHA[]->NXM_NX_ARP_THA[],move:NXM_OF_ARP_SPA[]->NXM_OF_ARP_TPA[],load:0xfa163e6f7280->NXM_NX_ARP_SHA[],load:0xa000005->NX
 cookie=0x0, duration=4227732.781s, table=40, n_packets=0, n_bytes=0, priority=61010,arp,arp_sha=fa:16:3e:6f:72:80 actions=goto_table:50
 cookie=0x0, duration=4227732.481s, table=40, n_packets=162888839, n_bytes=8796175711, priority=61007,ip,dl_src=fa:16:3e:6f:72:80 actions=goto_table:50
 cookie=0x0, duration=4227732.764s, table=40, n_packets=0, n_bytes=0, priority=36001,ip,in_port=5,dl_src=fa:16:3e:6f:72:80,nw_src=10.0.0.5 actions=goto_table:50
 cookie=0x0, duration=67364.965s, table=70, n_packets=170503259, n_bytes=9243824995, priority=1024,ip,tun_id=0x447,nw_dst=10.0.0.5 actions=set_field:fa:16:3e:6f:72:80->eth_ds
 cookie=0x0, duration=4227732.795s, table=90, n_packets=0, n_bytes=0, priority=61010,arp,arp_tha=fa:16:3e:6f:72:80 actions=goto_table:100
 cookie=0x0, duration=598.094s, table=110, n_packets=162952069, n_bytes=8836057866, priority=16384,tun_id=0x447,dl_dst=fa:16:3e:6f:72:80 actions=output:5,output:15,output:33
stack@localhost:~/devstack$
```

Lista flow-entries nel br-int. Il flusso destinato alla macchina vittima viene dirottato alla VNF1 e VNF2

```
Fri Nov 18 15:50:21 CET 2016
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
7 packets captured
7 packets received by filter
0 packets dropped by kernel
IP macchina attaccante 10.0.0.3
IP macchina attaccata 10.0.0.5
Brute-force attack riconosciuto
iniettata flow-entry drop-10.0.0.3-10.0.0.5.ssh:
```

Output script in esecuzione nella VNF2

Flow-entry "drop-10.0.0.3-10.0.0.0-5-ssh"

GET http://192.168.7.18:8080/restconf/config/opendaylight-inventory:nodes/node/openflow:216216030865532/table/110/flow/drop-10.0.0.3-10.0.0.5.ssh:

Body Cookies Headers (5) Tests

Pretty Raw Preview JSON


```
{
  "flow-node-inventory:flow": [
    {
      "id": "drop-10.0.0.3-10.0.0.5.ssh:",
      "instructions": {
        "instruction": [
          {
            "order": 0,
            "apply-actions": {
              "action": [
                {
                  "order": 0,
                  "drop-action": {}
                }
              ]
            }
          }
        ]
      },
      "hard-timeout": 0
      "match": {
        "ethernet-match": {
          "ethernet-type": {
            "type": 2048
          }
        },
        "ipv4-source": "10.0.0.3/32",
        "ipv4-destination": "10.0.0.5/32"
      },
      "strict": false,
      "table_id": 110,
      "priority": 16389,
      "idle-timeout": 0
    }
  ]
}
```

Esempio script in esecuzione nella VNF2 "Brute-force protection"

```
#!/bin/bash
URL_ODL=192.168.7.18:8080
Br_int=openflow:216216030865532
function drop_flow()
{
curl  --user admin:admin  -H 'content-type: application/json' -X PUT -d \
'{"flow-node-inventory:flow":[{"id":"'$3\
'","instructions":{"instruction":[{"order":0,"apply-actions":{"action":[{"order":0,"drop-action":{}}]}}]},"hard-timeout":0,"match":{"ethernet-match":{"ethernet-type":
{"type":2048}},"ipv4-source":"'\
"$1"'","ipv4-destination":"'\
"$2"'"},"strict":false,"table_id":110,"priority":16389,"idle-timeout":0}]}' \
http://${URL_ODL}/restconf/config/opendaylight-inventory:nodes/node/${Br_int}/table/110/flow/$3
}
while :; do
 date;
 string=$(timeout 20 tcpdump -i eth0  'dst port 22 'and 'tcp[tcpflags] & (tcp-syn) != 0' | awk ' s=$3, d=$5 { print gensub(/(\.[^.]+$)/,""","g",s)" "d }'  | uniq -c )
 cnt=$(echo $string | awk '{print $1}')
 src=$(echo $string | awk '{print $2}')
 dst=$(echo $string | awk '{print $3}')
 if [ "$cnt" == "" ] ; then
      cnt=0
 fi
 if (( $cnt > 5 )) ; then
      ruleid="drop-${src}-${dst}"
     echo "IP macchina attaccante ${src}"
     echo "IP macchina attaccata ${dst%.*}"
drop_flow ${src}/32 ${dst%.*}/32 ${ruleid}
     echo "Brute-force attack riconosciuto"
     echo "iniettata flow-entry $ruleid"
 fi
 echo;
 sleep 1
done
```

Obiettivi raggiunti

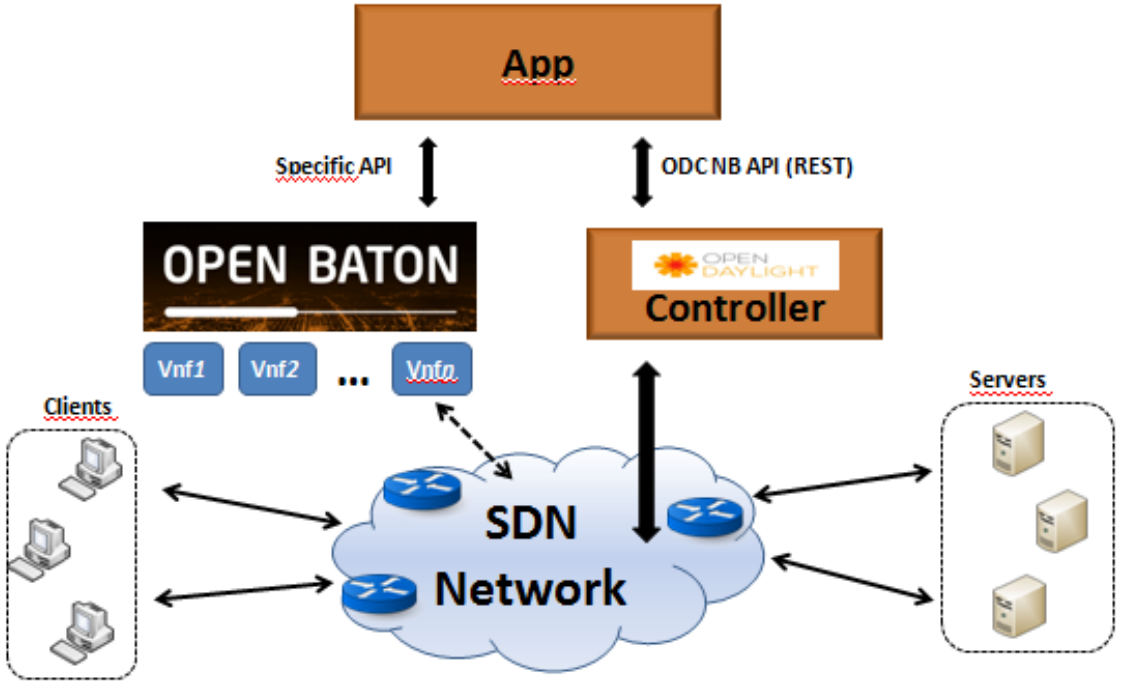
- Approfondimento dei paradigmi di rete SDN, NFV e Cloud Computing
- Comprensione e integrazione OpenStack Mitaka - OpenDaylight Beryllium
- Realizzazione di un'infrastruttura di test in un ambiente virtuale NFV/SDN e deployment di VNFs attraverso Open Baton
- Riconoscere un attacco SYN-flood e/o Brute-force e reagire creando, in real-time, delle regole ad hoc da iniettare agli OVS bridges usando OpenDaylight

Proposta attività II anno (1)

- Comprensione delle Service Functions Chains (SFCs) in OpenDaylight e loro integrazione nell'infrastruttura di test utilizzata
- Miglioramento e ottimizzazione dei risultati ottenuti spostando quanto fatto dall'infrastruttura di test ad una parte dell'infrastruttura di produzione, composta da switches hardware OpenFlow compliant e switch virtuali (OVS) relativi ad installazioni di OpenStack/infrastrutture di virtualizzazione esistenti nella sezione INFN di Catania

Proposta attività II anno (2)

- Realizzazione di una applicazione d'intelligenza gestionale con GUI al fine di armonizzare e re-ingegnerizzare i tools utilizzati nell'infrastruttura di test per l'utilizzo effettivo in un ambiente di produzione



Grazie per l'attenzione!

Domande