ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Εγκαθίδρυση συνδεσιμότητας επιπέδου 2 μεταξύ πολλαπλών απομονωμένων περιβαλλόντων cloud τύπου Openstack

ΓΡΙΒΑΣ ΕΥΘΥΜΙΟΣ

ΑΡΙΘΜΟΣ ΜΗΤΡΩΟΥ: 1047014

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΔΕΝΑΖΗΣ ΣΠΥΡΙΔΩΝ

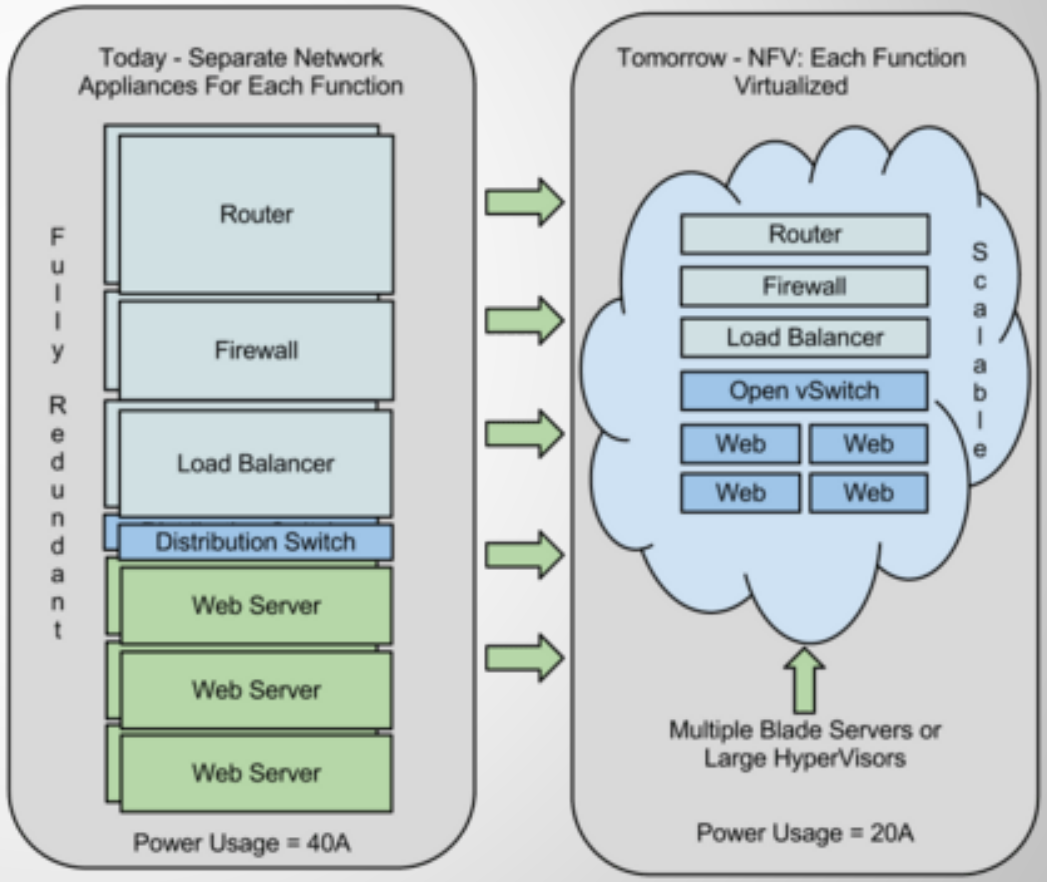
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΠΟΛΟΓΙΣΤΩΝ – 01/07/20

Περίγραμμα παρουσίασης

- Περιγραφή τεχνολογιών για διασύνδεση clouds σε NFV περιβάλλοντα
- Ανάγκες εκπόνησης, στόχος της εργασίας και υλοποίηση
- Μελλοντικές επεκτάσεις

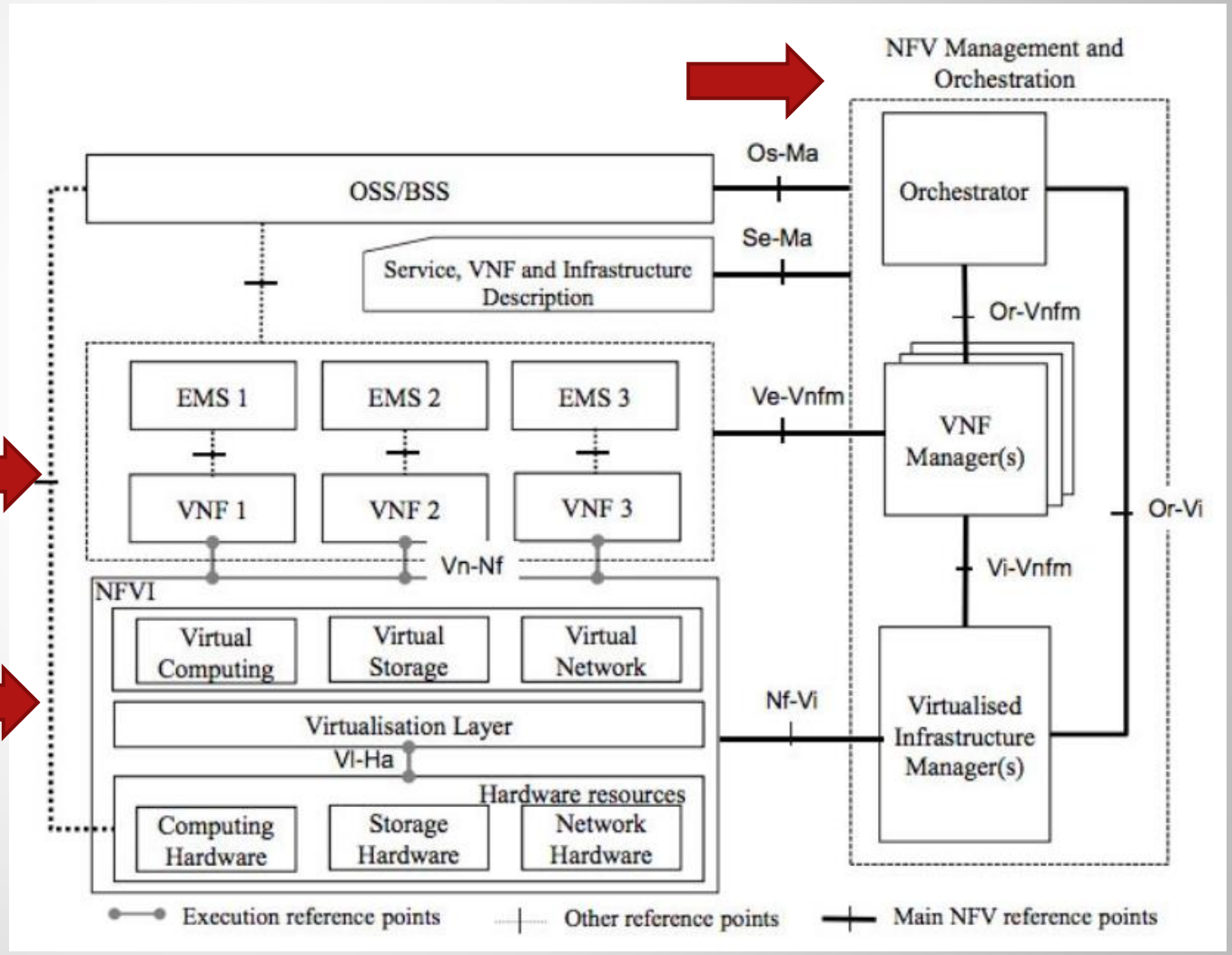
Network Functions Virtualization (NFV)

- Εικονοποίηση δικτυακών λειτουργιών που έτρεχαν σε εξειδικευμένο hardware
- Γενικού σκοπού εξυπηρετητές φιλοξενούν πολλές δικτυακές λειτουργίες
- Χρήση κυρίως για περιβάλλοντα τηλεπικοινωνιακών παροχών

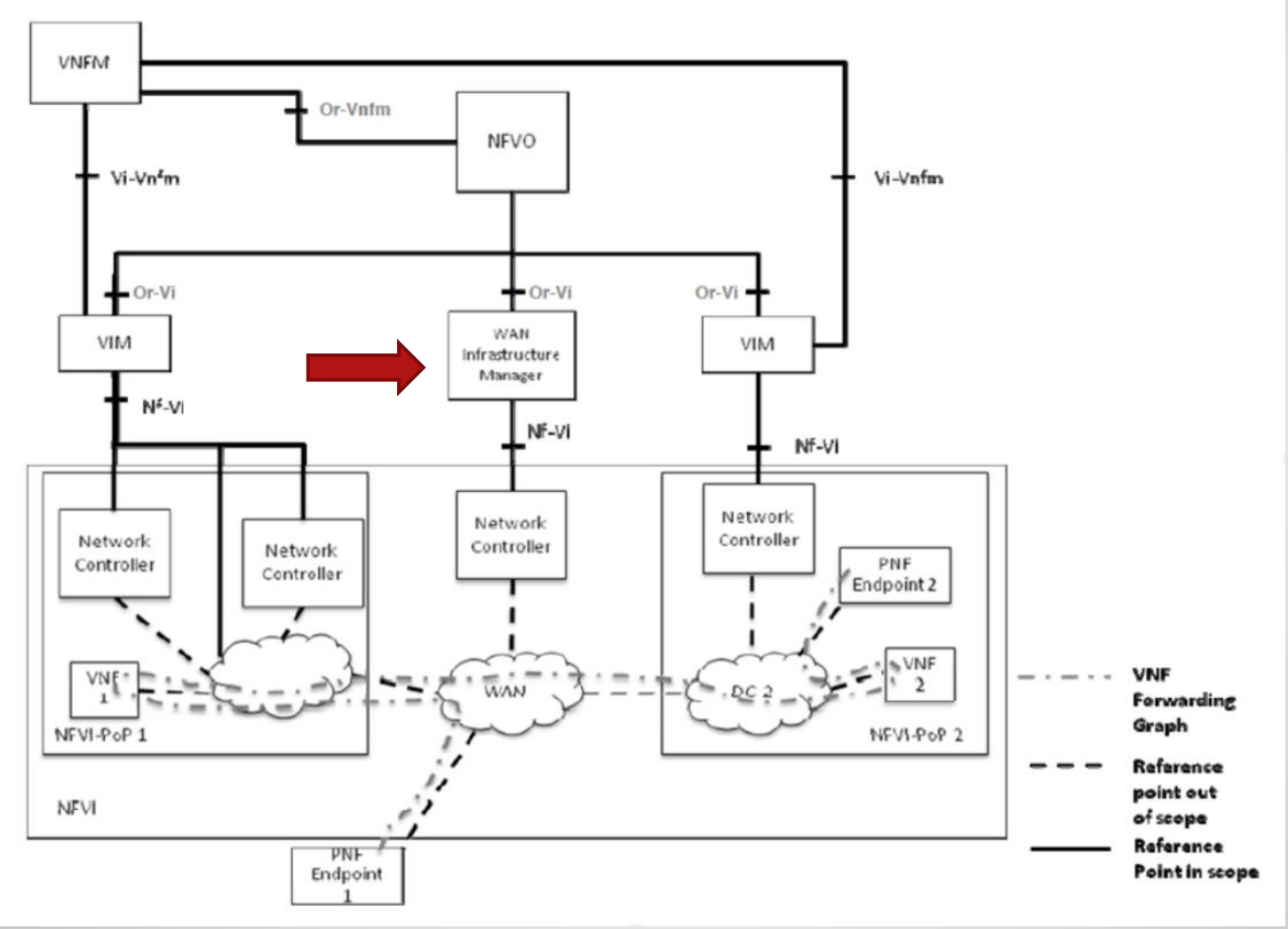


Αρχιτεκτονική NFV

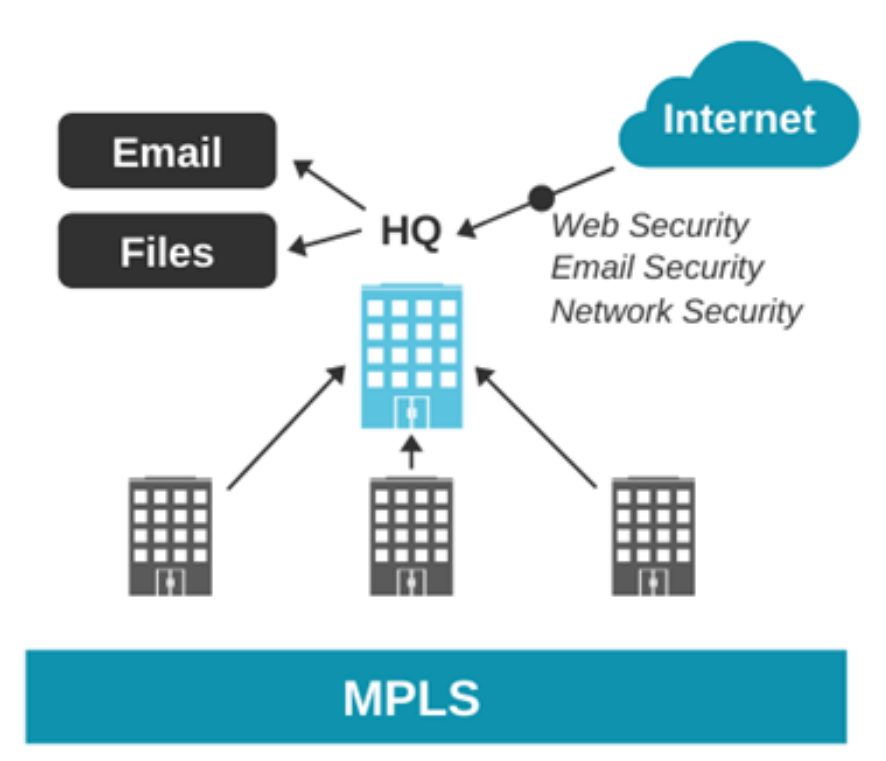
- NFV Infrastructure (NFVI) – NFVI PoP's
- Virtualized Network Function (VNF)
- NFV Management and Orchestration (NFV MANO)

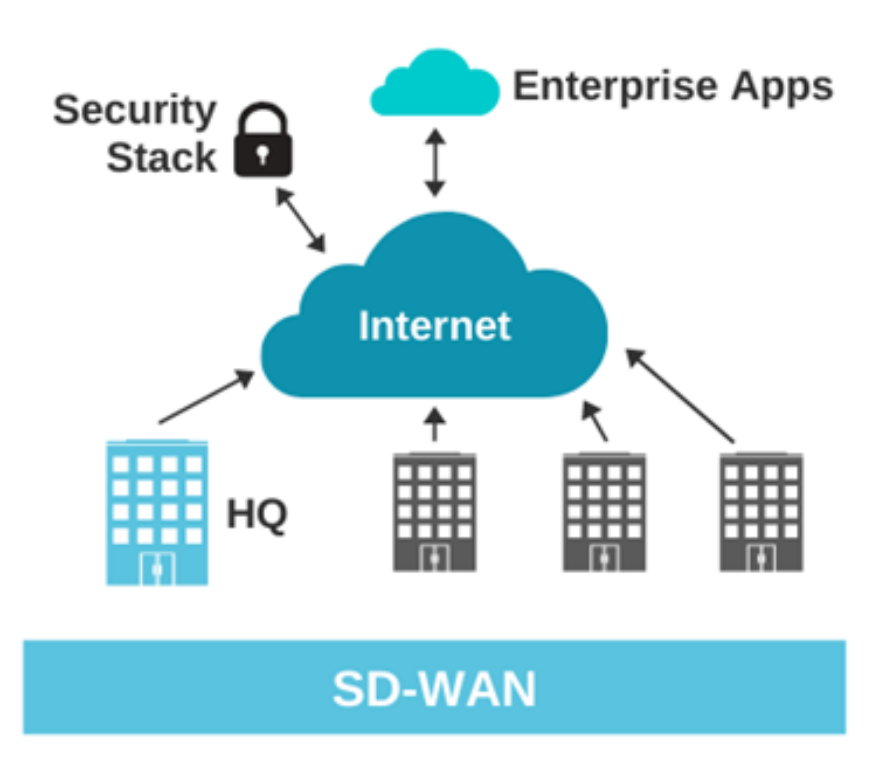


WAN Infrastructure Manager (WIM)



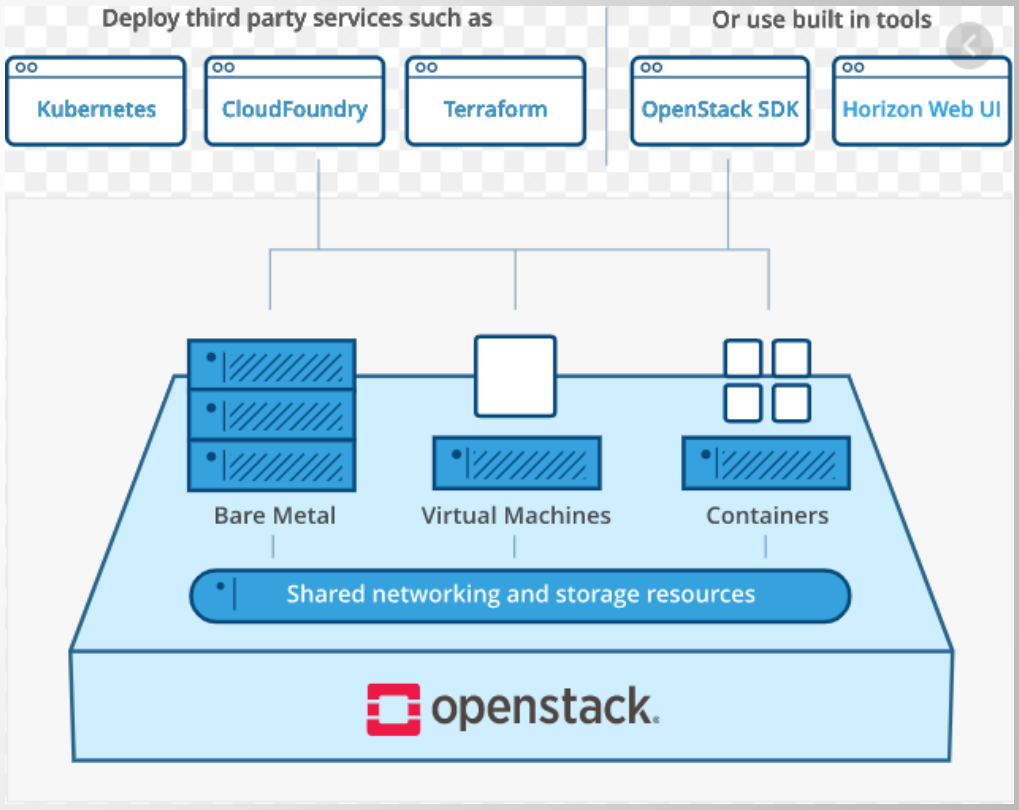
MPLS vs SD-WAN





Openstack

- Ξεκίνησε το 2010 σαν από κοινού project της Rackspace και της NASA
- Λειτουργικό σύστημα για cloud που ελέγχει μεγάλες ομάδες πόρων υπολογιστικής ισχύος, αποθήκευσης και δικτύωσης σε ένα κέντρο δεδομένων
- Ανοιχτού κώδικα



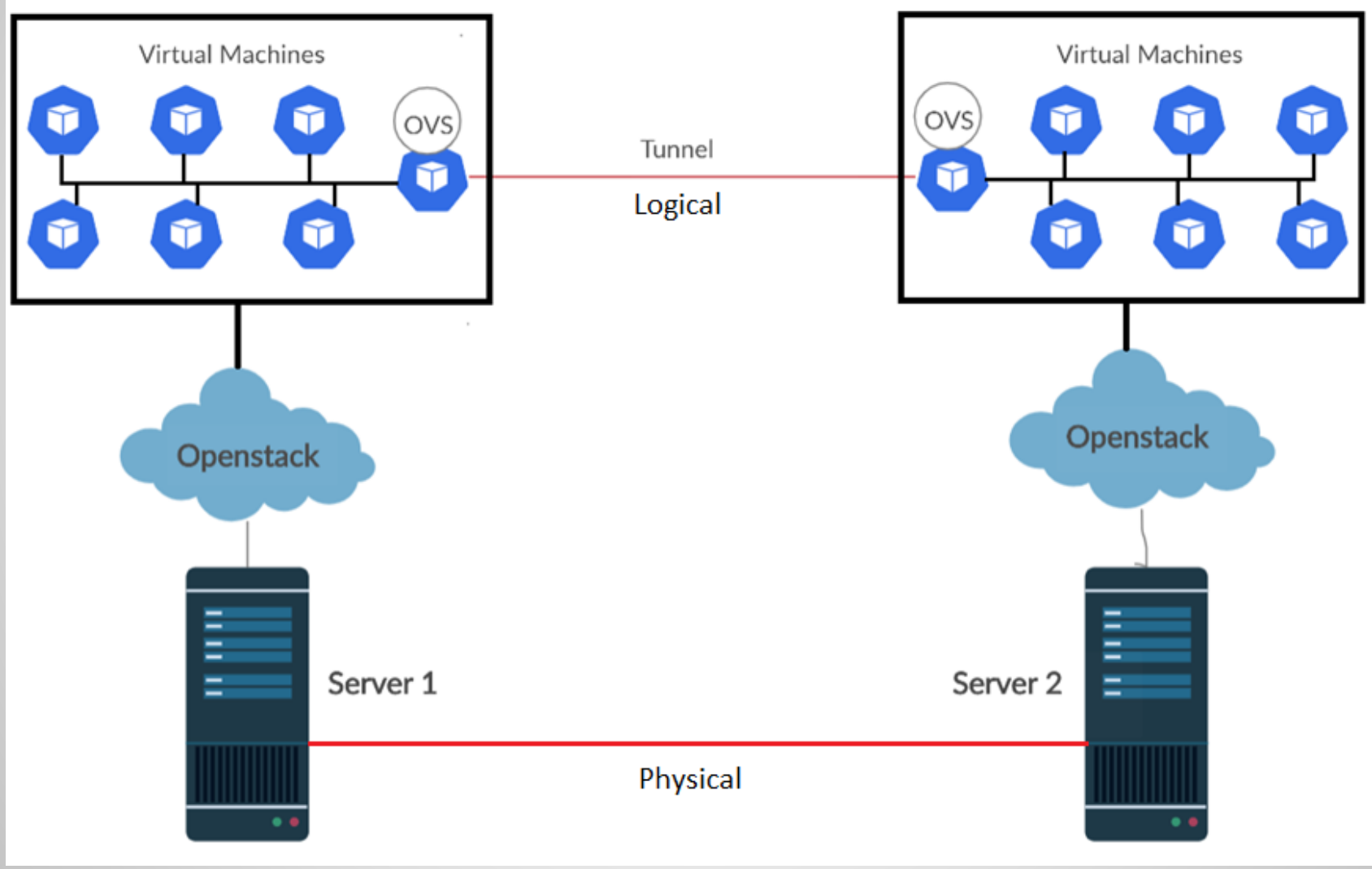
Πρόβλημα διπλωματικής

- Κατανεμημένα περιβάλλοντα cloud σε διαφορετικές τοποθεσίες
- Σύνδεση ανομοιογενών περιβαλλόντων (edge, core)
- Κατανεμημένες υπηρεσίες όπως VNF's
- Ανάγκη επικοινωνίας επιπέδου 2

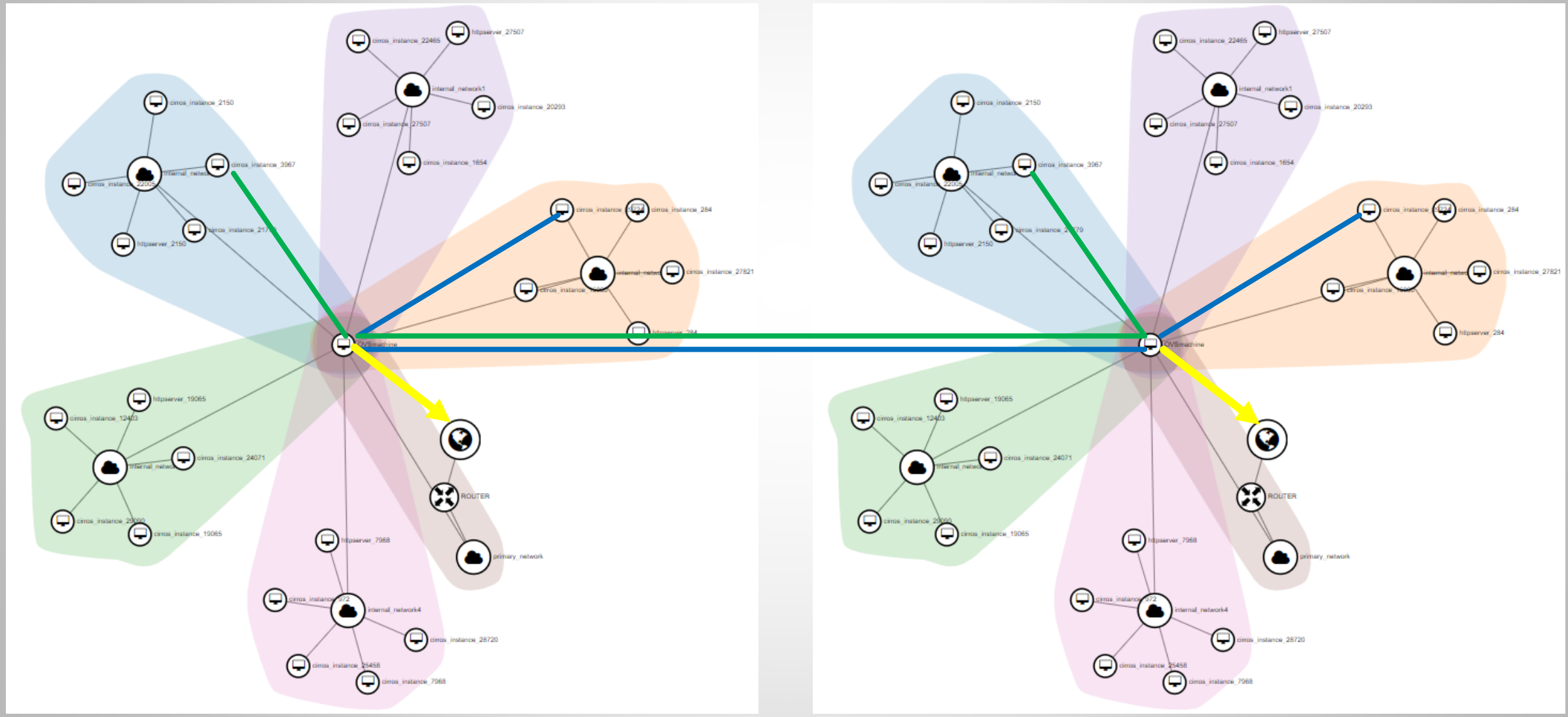
Στόχος διπλωματικής

- Υλοποίηση ενός VNF
- Παροχή συνδεσιμότητας επιπέδου 2 μεταξύ περιβαλλόντων cloud τύπου Openstack που βρίσκονται σε διαφορετικές τοποθεσίες
- Χρήση μεταγωγέων καθαρού λογισμικού βασισμένοι στο Open vSwitch (OVS)
- Tunnels από άκρο σε άκρο
- Κάθε cloud μπορεί να βρίσκεται πίσω από τοπολογία NAT

Σχηματικό διάγραμμα (1)



Σχηματικό διάγραμμα (2)



Λεπτομέρειες υλοποίησης

1. Virtual Private Network (VPN)
2. Custom image για πακετάρισμα του Open vSwitch με τη χρήση του Packer
3. Virtual Extensible LAN (VXLAN) για τη δημιουργία των tunnels
4. MSS = 1422 bytes

Επέκταση διπλωματικής

- Αφαίρεση VPN server
- Απευθείας σύνδεση σε δημόσιο δίκτυο
- Controller για αυτόματη σχεδίαση και διαχείριση των tunnels στους μεταγωγείς
- Ενορχήστρωση με Kubernetes

Ερωτήσεις;

Ευχαριστώ για την παρακολούθηση

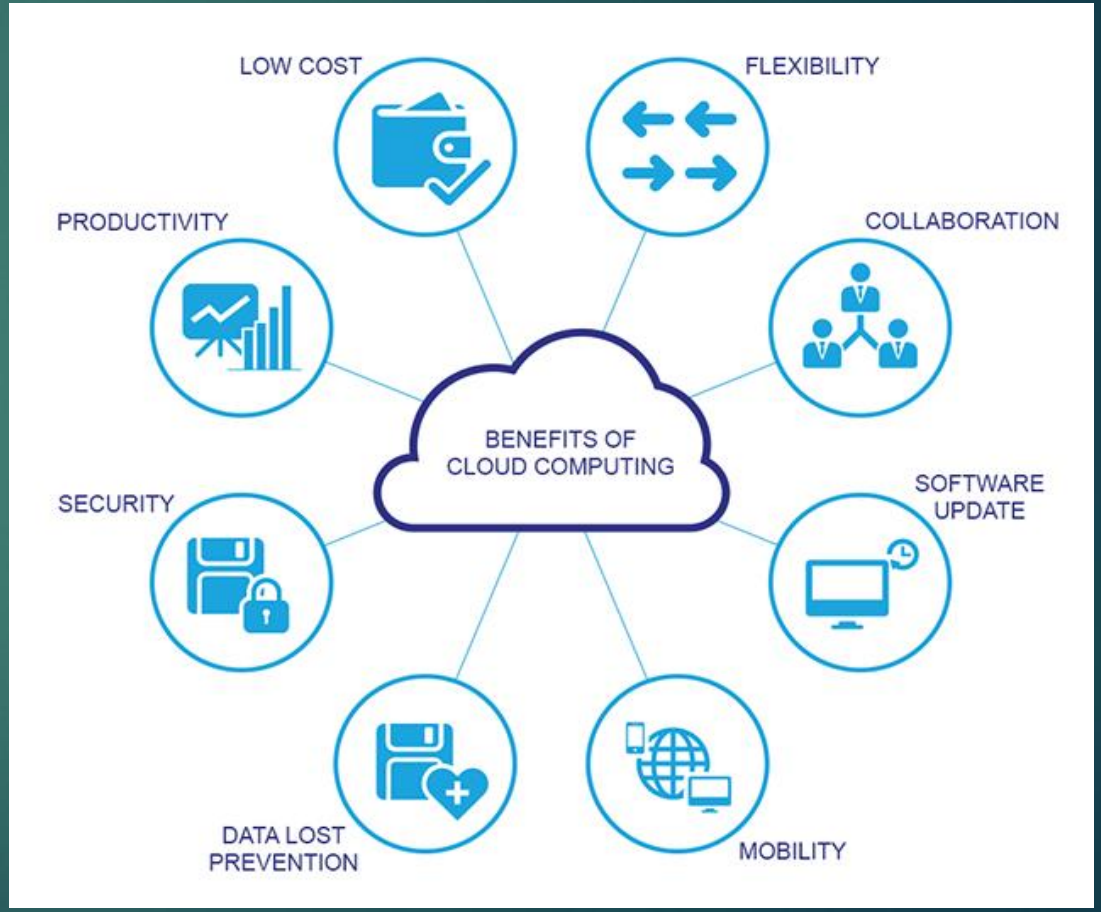
Back-up slides

Ο κώδικας

- https://github.com/thimiosgr/CloudConnectivity

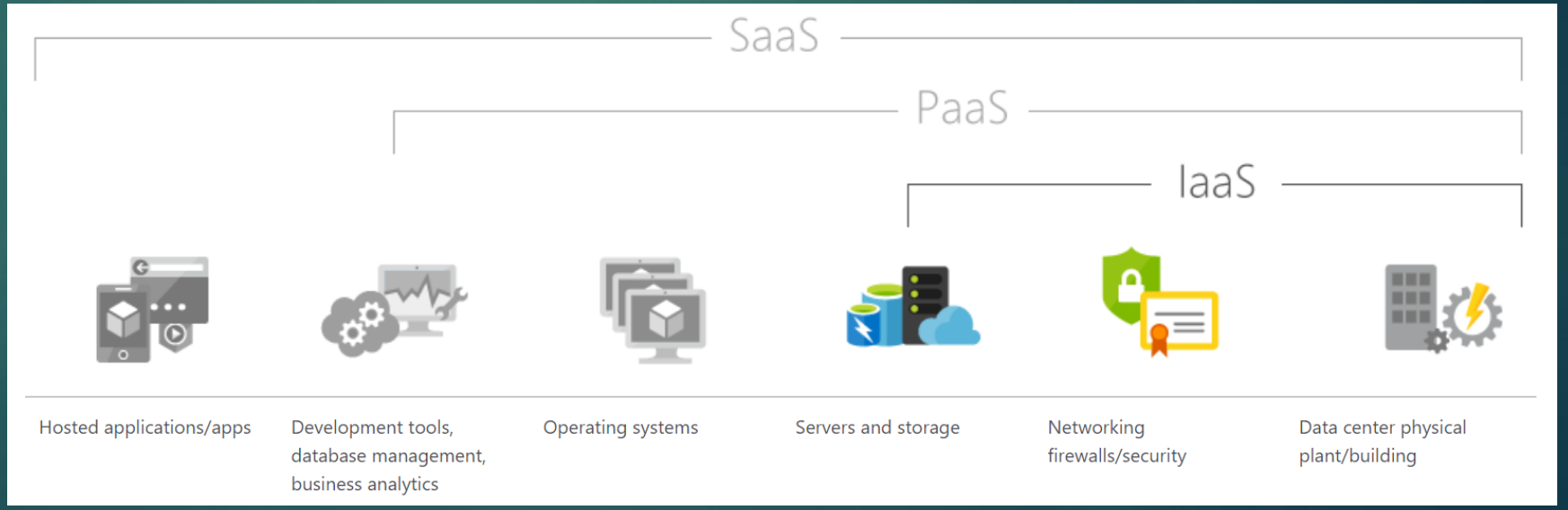
Cloud Computing

- Κατ' απαίτηση παράδοση υπηρεσιών πληροφορικής – συμπεριλαμβανομένων διακομιστών, αποθηκευτικό χώρο, βάσεις δεδομένων κλπ. – μέσω του Διαδικτύου
- 69% των επιχειρήσεων ήδη χρησιμοποιούν τεχνολογία cloud και 18% σκοπεύουν να χρησιμοποιήσουν κάποια στιγμή στο μέλλον σύμφωνα με μία μελέτη του International Data Group
- Εταιρείες που επενδύουν στο cloud έχουν μέχρι και 53% ταχύτερη αύξηση εσόδων από τους ανταγωνιστές τους σύμφωνα με τη Dell



Μοντέλα υπηρεσιών

- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Service as a Service (SaaS)
- Serverless

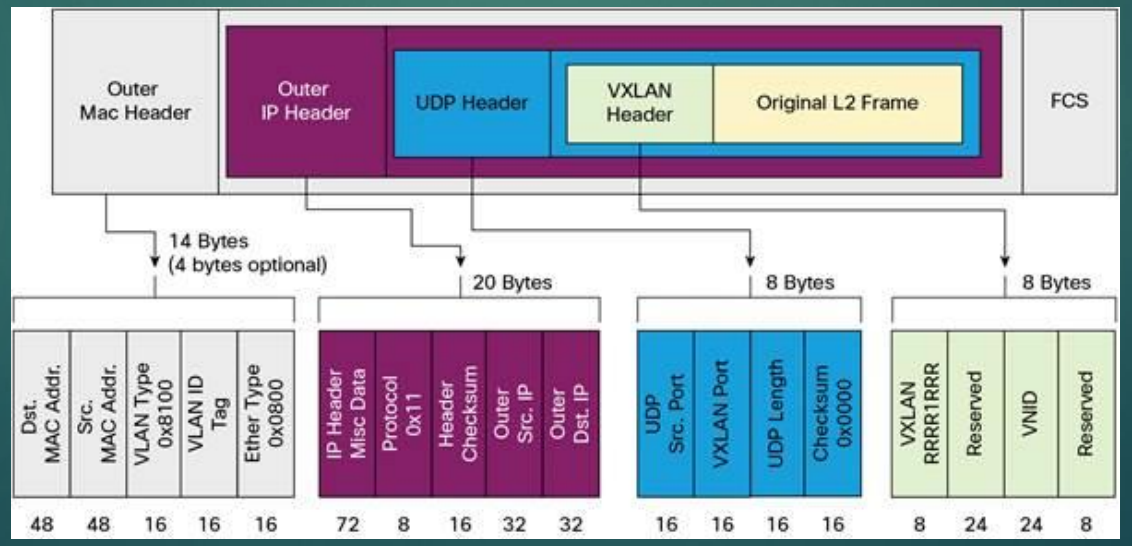


Πρωτόκολλα σηράγγωσης

- Επιτρέπουν επικοινωνίες μεταξύ ιδιωτικών δικτύων πάνω από ένα δημόσιο δίκτυο (όπως το Διαδίκτυο)
- Εσωκλείουν στο δεδομενόγραμμα ένα ολόκληρο πακέτο δεδομένων που χρησιμοποιεί ένα διαφορετικό πρωτόκολλο
- Generic Routing Encapsulation (GRE)
- Virtual Extensible LAN (VXLAN)
- Geneve
- Internet Protocol Security (IPsec)

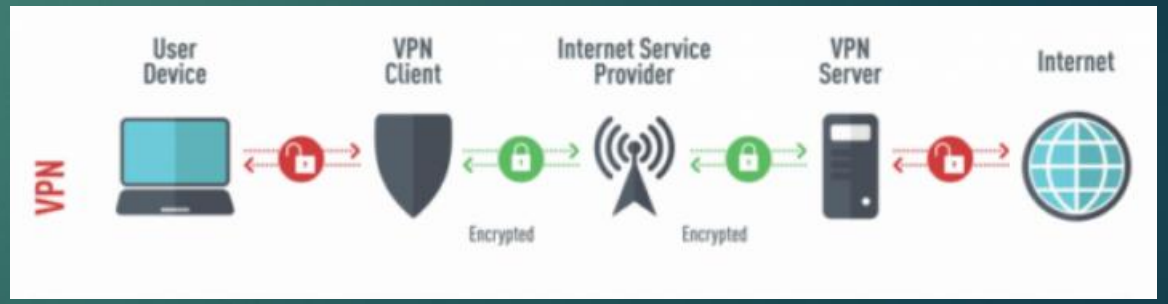
VXLAN

- Σχεδιάστηκε για να επιλύσει τα προβλήματα επεκτασιμότητας που σχετίζονται με μεγάλες εφαρμογές cloud
- Δημιουργήθηκε από τις VMware, Arista Networks και Cisco
- Ενθυλακώνει πλαίσια Ethernet επιπέδου 2 μέσα σε δεδομενογράμματα UDP επιπέδου 4
- Προσθέτει 50 έως 54 bytes κεφαλίδας στο αρχικό πλαίσιο Ethernet



Virtual Private Network (VPN)

- Επεκτείνει ένα ιδιωτικό δίκτυο πάνω από ένα δημόσιο δίκτυο
- Η αρχική IP του χρήστη αντικαθίσταται από μία IP του παρόχου VPN
- Συχνά χρησιμοποιείται από εταιρείες για την προστασία ευαίσθητων δεδομένων
- OpenVPN
- L2TP/IPsec
- Point-to-Point Tunneling Protocol (PPTP)
- Secure Socket Tunneling Protocol (SSTP)



OpenVPN

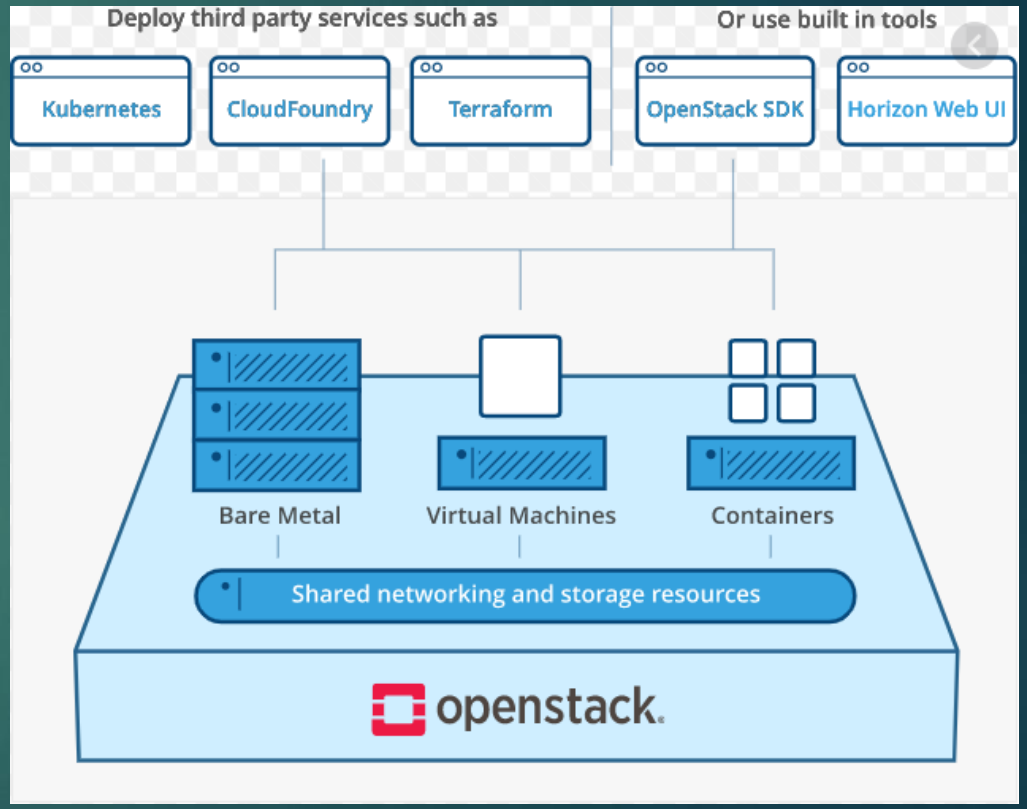
- Ανοιχτού κώδικα
- Από τα πιο ασφαλή πρωτόκολλα
- Μπορεί να χρησιμοποιηθεί και με TCP και με UDP
- Εξαιρετικά διαμορφώσιμο
- Συμβατό με πολλές πλατφόρμες

Open vSwitch

- Υλοποίηση ανοιχτού κώδικα ενός διανεμημένου εικονικού διακόπτη πολλαπλών επιπέδων
- Παρέχει μια στοίβα εναλλαγής για περιβάλλοντα εικονοποίησης υλικού
- Μπορεί να λειτουργήσει τόσο ως διακόπτης δικτύου που βασίζεται σε λογισμικό που εκτελείται σε μία εικονική μηχανή (VM), όσο και ως στοίβα ελέγχου για ειδικό εξοπλισμό εναλλαγής
- Υποστηρίζει πολλές δυνατότητες
- Παίζει το ρόλο της πύλης δικτύου για τα εντελώς απομονωμένα δίκτυα του κάθε cloud προς τα υπόλοιπα cloud

Openstack

- Λειτουργικό σύστημα για cloud που ελέγχει μεγάλες ομάδες πόρων υπολογιστικής ισχύος, αποθήκευσης και δικτύωσης σε ένα κέντρο δεδομένων
- Μία διεπαφή χρήστη είναι διαθέσιμη
- Ξεκίνησε το 2010 σαν από κοινού project της Rackspace και της NASA
- Ανοιχτού κώδικα



Main components

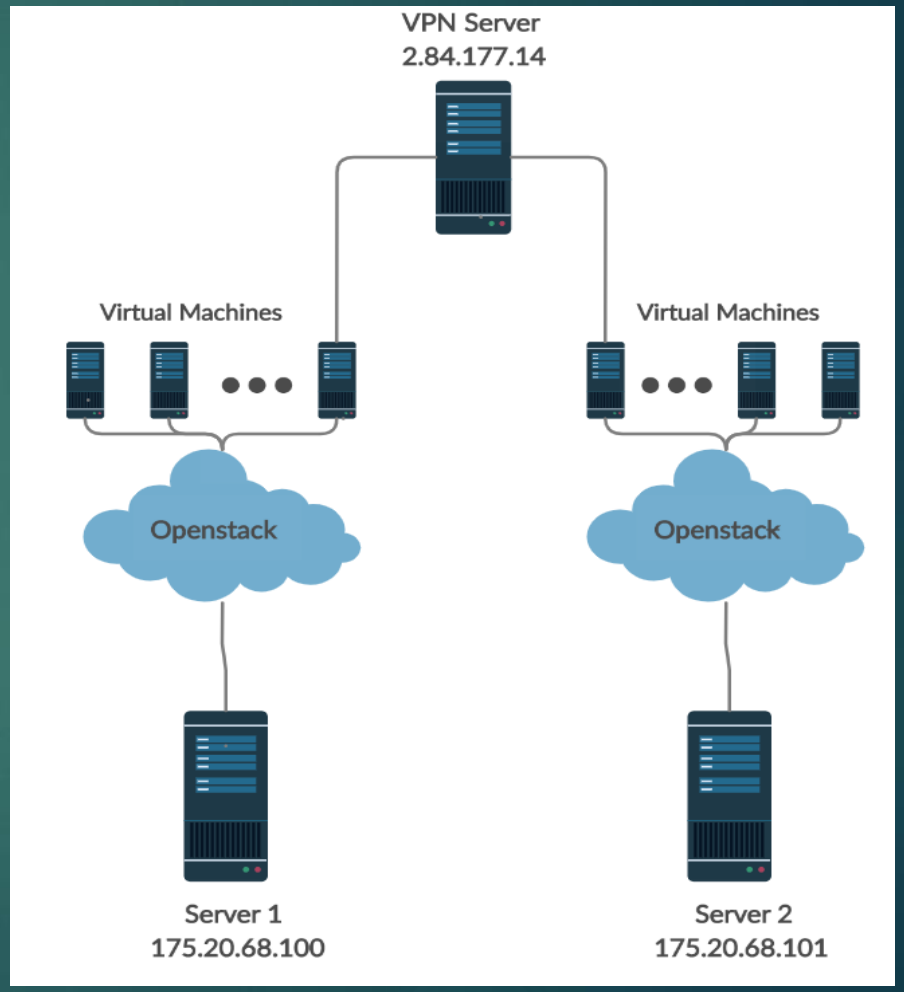
1. Nova
2. Swift
3. Cinder
4. Neutron
5. Horizon
6. Keystone
7. Glance
8. Ceilometer
9. Heat

Packer

- Εργαλείο για τη δημιουργία εικόνων μηχανής (images) για πολλές πλατφόρμες
- Οι εικόνες μηχανής του Packer επιτρέπουν στους χρήστες να εκκινήσουν πλήρως εξοπλισμένα και διαμορφωμένα μηχανήματα σε δευτερόλεπτα
- Ανοιχτού κώδικα
- Έχει μικρές απαιτήσεις συστήματος και είναι συμβατό με τα περισσότερα λειτουργικά συστήματα
- Δημιουργήθηκε από τη HashiCorp
- Δημιουργία 2 images

Η εργασία

- Δύο μηχανήματα που τρέχουν το Openstack μέσω του Devstack
- Ελάχιστες απαιτήσεις συστήματος: 8 GB RAM, 50 GB αποθηκευτικός χώρος
- VPN server σε εξωτερικό μηχάνημα
- Εικονικές μηχανές σε κάθε cloud
- 1 εικονική μηχανή χρησιμοποιεί το Open vSwitch
- https://github.com/thimiosgr/CloudConnectivity



VPN server

- Χρήση του πρωτοκόλλου OpenVPN
- TCP, port 443
- Virtualbox, Ubuntu Server 16.04
- Απαιτήσεις συστήματος: 512 MB RAM, 2 GB αποθηκευτικός χώρος
- Διαμορφωμένος να δίνει συγκεκριμένες IP σε κάθε εικονική μηχανή που συνδέεται

Διαμορφωμένες εικόνες μηχανής


```
"builders": [
  {
    "type": "openstack",
    "ssh_username": "ubuntu",
    "identity_endpoint": "http://150.140.186.115/identity",
    "image_name": "webserver",
    "source_image": "e9ad70c4-438e-483b-bff7-931b2181bad6",
    "flavor": "ds512M",
    "networks": [
      "3e42296e-cdbf-4099-a4c4-3ea4df87535c"
    ],
    "use_floating_ip": true,
    "floating_ip_pool": "public"
  }
```

```
#!/bin/bash

python -m SimpleHTTPServer
```

```
[Unit]
After=network.target

[Service]
ExecStart=/usr/local/bin/httpserver.sh

[Install]
WantedBy=default.target
```

```
"provisioners": [
  {
    "type": "file",
    "source": "/home/comlab/CloudConnectivity/scripts/httpserver.sh",
    "destination": "/home/ubuntu/httpserver.sh"
  },
  {
    "type": "file",
    "source": "/home/comlab/CloudConnectivity/scripts/httpserver.service",
    "destination": "/home/ubuntu/httpserver.service"
  },
  {
    "type": "shell",
    "inline_shebang": "/bin/bash -e",
    "inline": [
      "sleep 1",
      "sudo bash -c 'echo \"net.ipv4.ip_forward=1\" >> /etc/sysctl.conf' 2>&1",
      "sudo mv httpserver.sh /usr/local/bin > /dev/null 2>&1",
      "sudo chmod 755 /usr/local/bin/httpserver.sh > /dev/null 2>&1",
      "sudo mv httpserver.service /etc/systemd/system > /dev/null 2>&1",
      "sudo chmod 664 /etc/systemd/system/httpserver.service > /dev/null 2>&1",
      "sudo systemctl daemon-reload > /dev/null 2>&1",
      "sudo systemctl enable httpserver.service > /dev/null 2>&1",
      "sudo apt update > /dev/null 2>&1",
      "sudo apt install python -y > /dev/null 2>&1"
    ]
  }
]
```

Εκτέλεση κώδικα

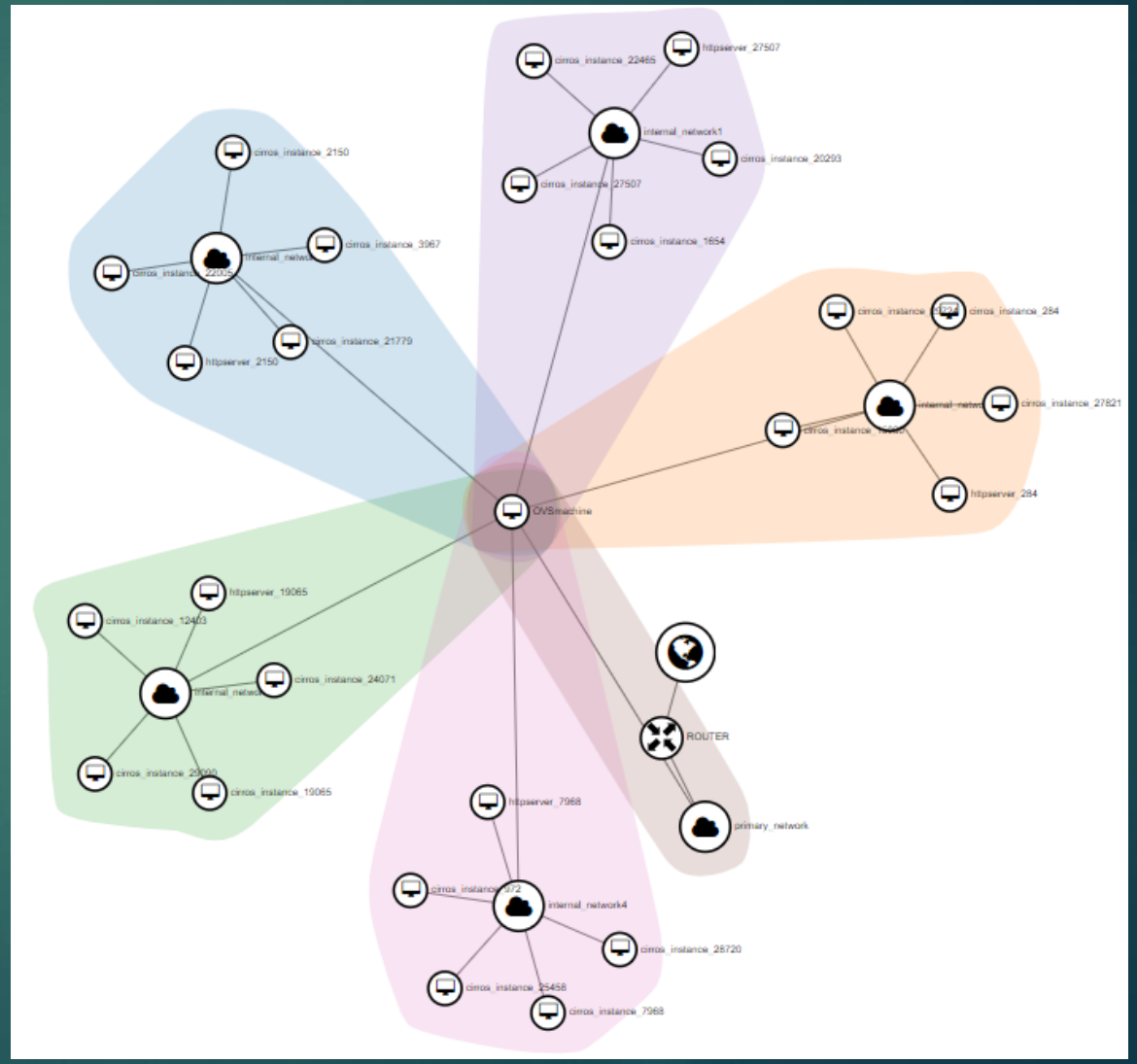
- Εγκατάσταση Packer
- Εγκατάσταση jq (JSON processor)
- Διαμόρφωση των JSON files ανάλογα με τις προτιμήσεις του χρήστη
- Δημιουργία εικόνων μηχανής
- Δημιουργία τοπολογίας δικτύου
- Εκκίνηση εικονικών μηχανών σε κάθε εσωτερικό δίκτυο

internal_network2	**internal_network2_subnet** 192.168.2.0/24
internal_network3	**internal_network3_subnet** 192.168.3.0/24
public	**public-subnet** 172.24.4.0/24 **ipv6-public-subnet** 2001:db8::/64
internal_network5	**internal_network5_subnet** 192.168.5.0/24
internal_network1	**internal_network1_subnet** 192.168.1.0/24
primary_network	**primary_network_subnet** 192.168.0.0/24
internal_network4	**internal_network4_subnet** 192.168.4.0/24

Τοπολογία δικτύου

- Η τοπολογία είναι ίδια σε κάθε cloud
- 1 δίκτυο με πρόσβαση στο Διαδίκτυο στο οποίο συνδέεται η μηχανή που θα λειτουργήσει ως πύλη δικτύου
- 5 εσωτερικά δίκτυα με 5 εικονικές μηχανές στο κάθε ένα
- 4 απλές εικονικές μηχανές (CirrosOS) και ένας εξυπηρετητής ιστού

OVSmachine	OVSimage	**internal_network1** 192.168.1.212 **internal_network4** 192.168.4.30 **internal_network2** 192.168.2.188 **primary_network** 192.168.0.25 **internal_network3** 192.168.3.20 **internal_network5** 192.168.5.71



Κώδικας Τοπολογίας


```
PRIMARY_NETWORK_ID=$(openstack network create primary_network --provider-network-type vxlan | grep " id " | awk '{print $4}' -)
INTERNAL_NETWORK1_ID=$(openstack network create internal_network1 --provider-network-type vxlan --disable-port-security | grep " id " | awk '{print $4}' -)
OPENSTACK_ARR[0]=$INTERNAL_NETWORK1_ID
INTERNAL_NETWORK2_ID=$(openstack network create internal_network2 --provider-network-type vxlan --disable-port-security | grep " id " | awk '{print $4}' -)
OPENSTACK_ARR[1]=$INTERNAL_NETWORK2_ID
INTERNAL_NETWORK3_ID=$(openstack network create internal_network3 --provider-network-type vxlan --disable-port-security | grep " id " | awk '{print $4}' -)
OPENSTACK_ARR[2]=$INTERNAL_NETWORK3_ID
INTERNAL_NETWORK4_ID=$(openstack network create internal_network4 --provider-network-type vxlan --disable-port-security | grep " id " | awk '{print $4}' -)
OPENSTACK_ARR[3]=$INTERNAL_NETWORK4_ID
INTERNAL_NETWORK5_ID=$(openstack network create internal_network5 --provider-network-type vxlan --disable-port-security | grep " id " | awk '{print $4}' -)
OPENSTACK_ARR[4]=$INTERNAL_NETWORK5_ID
ROUTER_ID=$(openstack router create ROUTER | grep " id " | awk '{print $4}' -)
PRIMARY_NETWORK_SUBNET_ID=$(openstack subnet create primary_network_subnet --network $PRIMARY_NETWORK_ID --subnet-range 192.168.0.0/24 --dhcp --dns-nameserver 8.8.8.8

openstack subnet create internal_network1_subnet --network $INTERNAL_NETWORK1_ID --subnet-range 192.168.1.0/24 --dhcp --gateway none > /dev/null 2>&1
openstack subnet create internal_network2_subnet --network $INTERNAL_NETWORK2_ID --subnet-range 192.168.2.0/24 --dhcp --gateway none > /dev/null 2>&1
openstack subnet create internal_network3_subnet --network $INTERNAL_NETWORK3_ID --subnet-range 192.168.3.0/24 --dhcp --gateway none > /dev/null 2>&1
openstack subnet create internal_network4_subnet --network $INTERNAL_NETWORK4_ID --subnet-range 192.168.4.0/24 --dhcp --gateway none > /dev/null 2>&1
openstack subnet create internal_network5_subnet --network $INTERNAL_NETWORK5_ID --subnet-range 192.168.5.0/24 --dhcp --gateway none > /dev/null 2>&1
openstack router set $ROUTER_ID --external-gateway $PUBLIC_NETWORK > /dev/null 2>&1
openstack router add subnet $ROUTER_ID $PRIMARY_NETWORK_SUBNET_ID  > /dev/null 2>&1

COUNTER=0
while [ "$COUNTER" -lt "${#OPENSTACK_ARR[@]}" ];
do
  for i in $(seq 1 4)
  do
    RANDOM_INTEGER=$(echo $((1 + RANDOM)))
    openstack server create --image cirros-0.4.0-x86_64-disk --flavor m1.nano --network ${OPENSTACK_ARR[COUNTER]} "cirros_instance_${RANDOM_INTEGER}" > /dev/null 2>&1
  done
  openstack server create --image SimpleHTTPserver --flavor m1.heat_int --network ${OPENSTACK_ARR[COUNTER]} "httpserver_${RANDOM_INTEGER}" > /dev/null 2>&1
COUNTER=$((COUNTER+1))
done
```

Open vSwitch machine

- Στην πρώτη εκκίνηση συνδέεται αυτόματα στον εξυπηρετητή VPN
- Δημιουργεί εικονικούς μεταγωγείς (Open vSwitch)
- Στήνει σήραγγες με τερματικό σημείο το αντίστοιχο μηχάνημα που τρέχει στο άλλο cloud

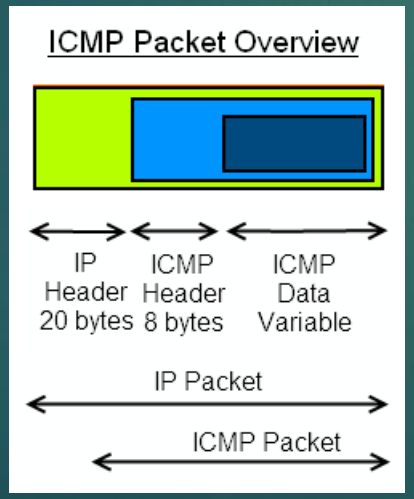
```
ubuntu@ovsmachine:~$ ifconfig tun0
tun0      Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          inet addr:10.8.0.20  P-t-P:255.255.255.255  Mask:255.255.255.255
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:6453 errors:0 dropped:0 overruns:0 frame:0
          TX packets:5997 errors:0 dropped:457 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:1286304 (1.2 MB)  TX bytes:1225444 (1.2 MB)
```

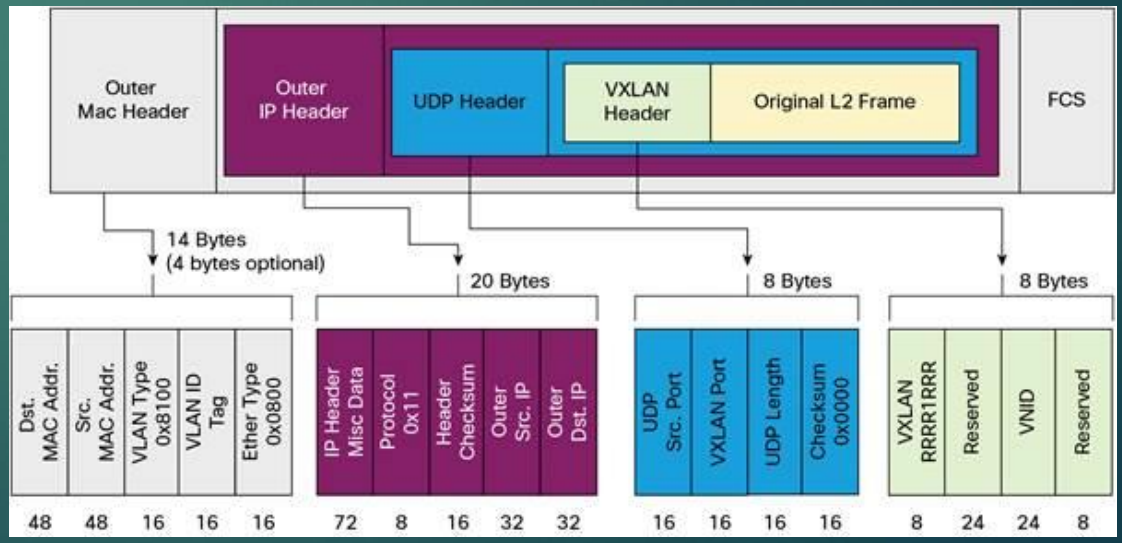
```
ubuntu@ovsmachine:~$ sudo ovs-vsctl show
    Bridge "brbca8c1369d"
        Port "vxlanbca8c1369d"
            Interface "vxlanbca8c1369d"
                type: vxlan
                options: {key="2002", remote_ip="10.8.0.40"}
        Port "ens6"
            Interface "ens6"
        Port "brbca8c1369d"
            Interface "brbca8c1369d"
                type: internal
    Bridge "brdc603aaf57"
        Port "ens7"
            Interface "ens7"
        Port "brdc603aaf57"
            Interface "brdc603aaf57"
                type: internal
        Port "vxlandc603aaf57"
            Interface "vxlandc603aaf57"
                type: vxlan
                options: {key="2003", remote_ip="10.8.0.40"}
```

MSS calculation

- Το Maximum Segment Size (MSS) υπολογίστηκε με την εντολή ping
- Το αρχικό MTU είναι 1500 bytes

Packet part	Number of bytes
ICMP HEADER	8
ORIGINAL IP HEADER	20
VXLAN HEADER	8
OUTER UDP HEADER	8
OUTER IP HEADER	20
OUTER MAC HEADER	14
PAYLOAD	1500 – 8 – 20 – 8 - 8 – 20 – 14 = 1422

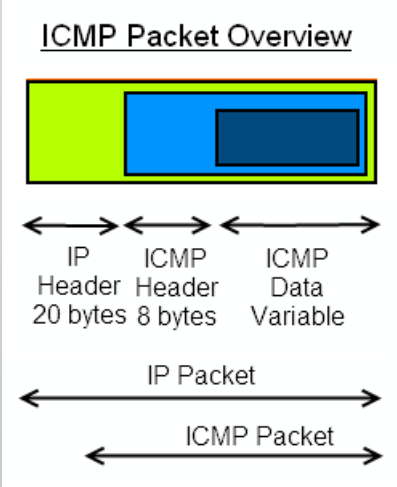


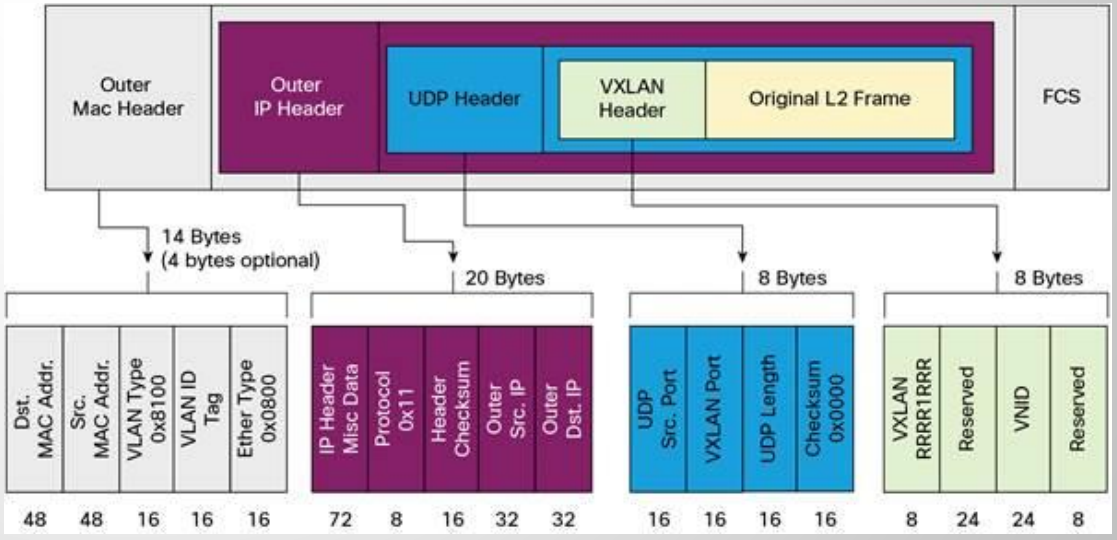


MSS calculation

- Το Maximum Segment Size (MSS) υπολογίστηκε με την εντολή ping
- Το αρχικό MTU είναι 1500 bytes

Packet part	Number of bytes
ICMP HEADER	8
ORIGINAL IP HEADER	20
VXLAN HEADER	8
OUTER UDP HEADER	8
OUTER IP HEADER	20
OUTER MAC HEADER	14
PAYLOAD	1500 – 8 – 20 – 8 - 8 – 20 – 14 = 1422





Custom image

- Packer
- Δημιουργία image από base image μέσω αρχείων μορφής JSON
- Base image: Ubuntu Server 16.04 Cloud Version
- 1ο image: Αυτόματο στήσιμο εικονικών μεταγωγέων, αυτόματο στήσιμο tunnels με το αντίστοιχο μηχάνημα στο απέναντι cloud
- 2ο image: απλός εξυπηρετητή ιστού που ακούει στο port 8000