

باسمه تعالی



دانشگاه صنعتی شریف

دانشکده مهندسی برق

## درس شبکه موبایل نرم افزار محور

تمرین سری پنجم

علی محرابیان 96102331

استاد: دکتر خلیج

بهار 1399



.1

در سمت سرور از دستورات زیر استفاده می کنیم.

```
Command Prompt - docker run -it --rm --network=mehrabian --name=iperfserver_96102331 -p 5201:5201 networkstatic/iperf3 -s

-O, --omit N          omit the first n seconds
-T, --title str        prefix every output line with this string
--get-server-output    get results from server

[KMKG] indicates options that support a K/M/G suffix for kilo-, mega-, or giga-

iperf3 homepage at: http://software.es.net/iperf/
Report bugs to: https://github.com/esnet/iperf

C:\Users\Ali Mehrabian>docker network create --driver bridge mehrabian
5b610b280dbdc7b816d67e73c374f4bd0aab26374b1b8922e271d5029f3e8e16

C:\Users\Ali Mehrabian>docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
a4d9df9d9d663       bridge             bridge              local
2ada4e5ed6a4        host               host                local
5b610b280dbd         mehrabian          bridge              local
6ed10e92c7d8        none              null                local

C:\Users\Ali Mehrabian>docker run -it --rm --network=mehrabian --name=iperfserver_96102331 -p 5201:5201 networkstatic/iperf3 -s

Server listening on 5201

Accepted connection from 172.18.0.3, port 45474
[ 5] local 172.18.0.2 port 5201 connected to 172.18.0.3 port 45476
[ ID] Interval      Transfer    Bandwidth
[ 5] 0.00-1.00 sec  1.67 GBytes 14.3 Gbits/sec
[ 5] 1.00-2.00 sec  1.69 GBytes 14.5 Gbits/sec
[ 5] 2.00-3.00 sec  2.07 GBytes 17.8 Gbits/sec
[ 5] 3.00-4.00 sec  1.71 GBytes 14.7 Gbits/sec
[ 5] 4.00-5.00 sec  1.78 GBytes 15.3 Gbits/sec
[ 5] 5.00-6.00 sec  1.66 GBytes 14.3 Gbits/sec
[ 5] 6.00-7.00 sec  1.91 GBytes 16.4 Gbits/sec
[ 5] 7.00-8.00 sec  1.72 GBytes 14.8 Gbits/sec
[ 5] 8.00-9.00 sec  1.82 GBytes 15.6 Gbits/sec
[ 5] 9.00-10.00 sec 1.64 GBytes 14.1 Gbits/sec
[ 5] 10.00-10.00 sec 3.75 MBytes  9.30 Gbits/sec

[ ID] Interval      Transfer    Bandwidth  Retr
[ 5] 0.00-10.00 sec 17.7 GBytes 15.2 Gbits/sec  0
[ 5] 0.00-10.00 sec 17.7 GBytes 15.2 Gbits/sec

sender
receiver

Server listening on 5201
```



در سمت client، به صورت زیر است.

```
Command Prompt
Microsoft Windows [Version 10.0.18363.900]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Ali Mehrabian>docker run -it --rm --name=iperfclient_96102331 --network=mehrabian networkstatic/iperf3 -c iperfserver_96102331
Connecting to host iperfserver_96102331, port 5201
[ 4] local 172.18.0.3 port 45476 connected to 172.18.0.2 port 5201
[ ID] Interval      Transfer    Bandwidth  Retr  Cwnd
[ 4] 0.00-1.00 sec  1.68 GBytes 14.4 Gbits/sec  0    714 KBytes
[ 4] 1.00-2.00 sec  1.69 GBytes 14.5 Gbits/sec  0    1.05 MBytes
[ 4] 2.00-3.00 sec  2.07 GBytes 17.8 Gbits/sec  0    1.15 MBytes
[ 4] 3.00-4.00 sec  1.71 GBytes 14.7 Gbits/sec  0    1.61 MBytes
[ 4] 4.00-5.00 sec  1.79 GBytes 15.4 Gbits/sec  0    1.78 MBytes
[ 4] 5.00-6.00 sec  1.66 GBytes 14.2 Gbits/sec  0    1.87 MBytes
[ 4] 6.00-7.00 sec  1.91 GBytes 16.5 Gbits/sec  0    1.96 MBytes
[ 4] 7.00-8.00 sec  1.72 GBytes 14.8 Gbits/sec  0    2.06 MBytes
[ 4] 8.00-9.00 sec  1.82 GBytes 15.6 Gbits/sec  0    2.06 MBytes
[ 4] 9.00-10.00 sec 1.64 GBytes 14.1 Gbits/sec  0    2.16 MBytes
-----
[ ID] Interval      Transfer    Bandwidth  Retr
[ 4] 0.00-10.00 sec 17.7 GBytes 15.2 Gbits/sec  0      sender
[ 4] 0.00-10.00 sec 17.7 GBytes 15.2 Gbits/sec  0      receiver

iperf Done.

C:\Users\Ali Mehrabian>
```



.2

در این سوال، از دستورات زیر استفاده کردیم.

```
Users\Ali Mehrabian>docker pull openbaton/standalone
Using default tag: latest
latest: Pulling from openbaton/standalone
Digest: sha256:a16735e38fa56b32785682728ccbd2bcffdf1360d2ef69cf2f0fe6444e9ccd6d
Status: Image is up to date for openbaton/standalone:latest
Users\Ali Mehrabian>docker run --name openbaton -d -h openbaton-rabbitmq -p 8080:8080 -p 5672:5672 -p 15672:15672 -p 8443:8443 -e RABBITMQ_BROKERIP=<RabbitMQ IP> openbaton/standalone
system cannot find the file specified.

Users\Ali Mehrabian>docker run --name openbaton -d -h openbaton-rabbitmq -p 8080:8080 -p 5672:5672 -p 15672:15672 -p 8443:8443 -e RABBITMQ_BROKERIP=192.168.1.1 openbaton/standalone
6fc0bfff4f310a123862188ef32c7e722d7cebe8a6cad9c114fc62873ae

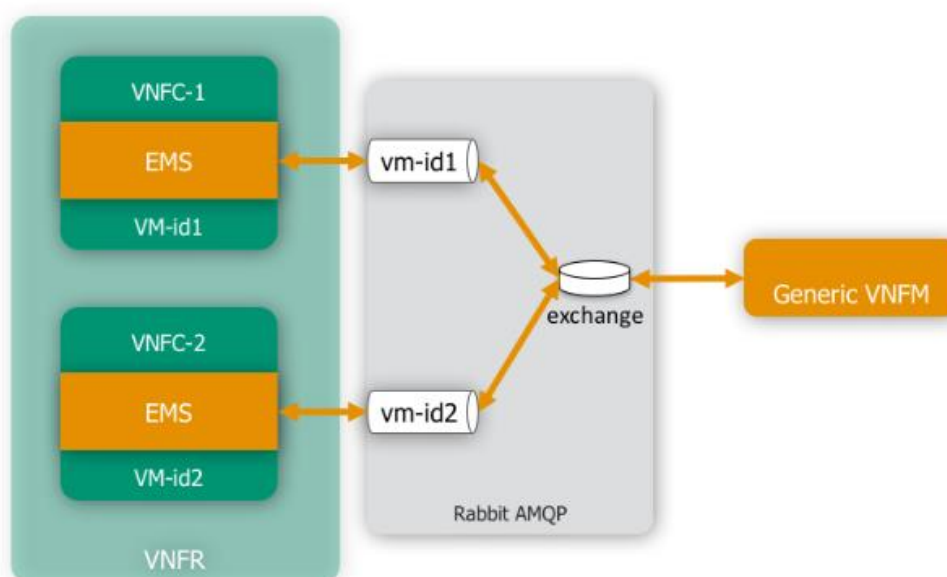
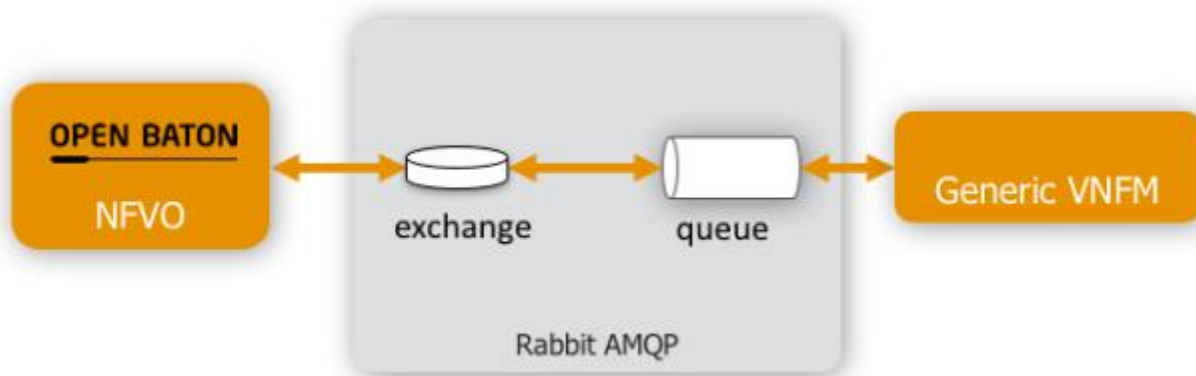
Users\Ali Mehrabian>docker ps -a
```

CONTAINER ID	IMAGE	COMMAND NAMES	CREATED	STATUS	PORTS
6fc0bfff4	openbaton/standalone	"/usr/bin/supervisord"	36 seconds ago	Up 31 seconds	0.0.0.0:5672->5672/tcp, 0.0.0.0:8080->8080/tcp, 0.0.0.0:8443->8443/tcp
443->8443/tcp, 0.0.0.0:15672->15672/tcp	hello-world	"/hello"	6 days ago	Exited (0) 6 days ago	
66	ubuntu	"bash --name iperfse..."	6 days ago	Exited (2) 6 days ago	
nostalgic_mccarthy	hello-world	"/hello"	6 days ago	Exited (0) 6 days ago	
69	hello-world	"--name sss"	6 days ago	Created	
cranky_bell	hello-world	"/hello"	6 days ago	Exited (0) 6 days ago	
nginx	hello-world	"/hello"	6 days ago	Exited (0) 6 days ago	
goofy_sinoussi	nginx	"/docker-entrypoint..."	6 days ago	Exited (255) 6 days ago	0.0.0.0:8080->80/tcp
mynginx1	ubuntu	"bash"	6 days ago	Exited (0) 6 days ago	
stupefied_hamilton	hello-world	"/hello"	6 days ago	Exited (0) 6 days ago	
wonderful_mestorf	ubuntu	"/bin/bash"	6 days ago	Exited (0) 6 days ago	
condescending_lewin	ubuntu	"/bin/bash"	6 days ago	Exited (0) 6 days ago	
angry_bohr					



**Rabbitmq** is a message queueing software. It is software where queues are defined, to which applications connect in order to transfer a message or messages.

In openbaton, communication NFVO ↔ VNFM ↔ EMS is done using the AMQP protocol over RabbitMQ.





**NFVO:** A NFV Orchestrator managing the lifecycle of Network Service Descriptors (NSD) and interfacing with one or more VNF Manager(s) (VNFM)

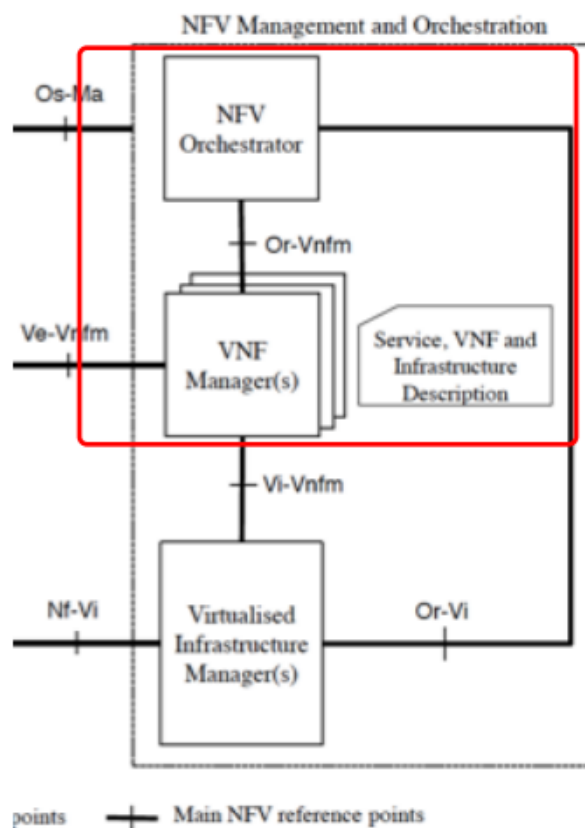
Nfvo is a modular software composed by many parts.

API: his module contains the necessary classes exposing APIs as ReST server.

Main: This module contains the classes in charge of the startup of the whole system and gathering configuratons for instance.

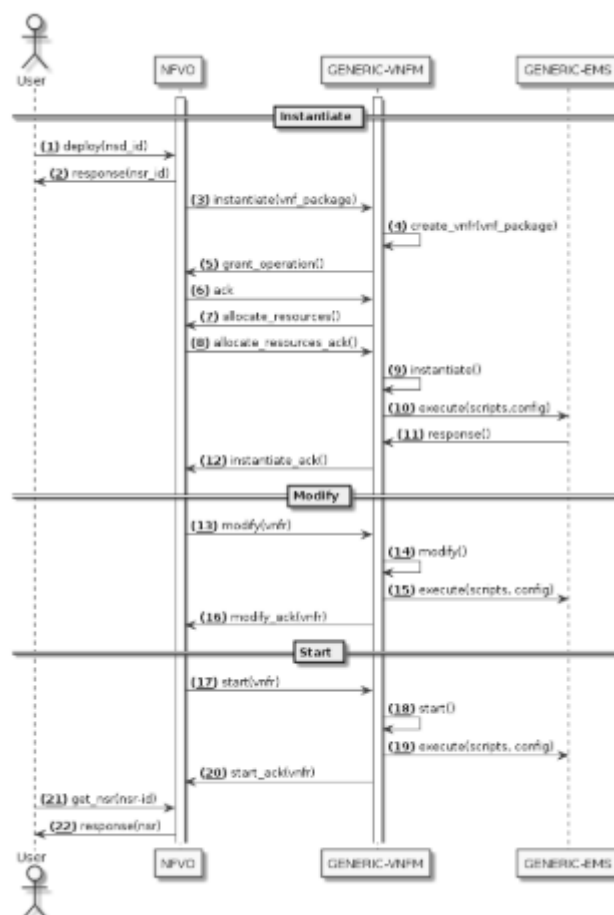
Core-int: This module contains the interfaces of the core functionalities regarding only the internal NFVO interfaces. Most of them are definend in ETSI MANO specification in NFV-MANO interfaces section.

...





**Generic VNF:** The Generic VNF Manager is an implementation following the ETSI MANO specifications. It works as intermediate component between the NFVO and the VNFs, particularly the Virtual Machines on top of which the VNF software is installed. In order to complete the lifecycle of a VNF, it interoperates with the Open Baton Element Management System (EMS) which acts as an agent inside the VMs and executing scripts contained in a VNF package or defined via the `scripts-link` inside the VNFD. The Generic VNFManager is capable of handling errors caused while executing these scripts. Together with the NFVO, it allows update of failed scripts, and resume NSR from the failed lifecycle state. Currently, this is applicable only for scripts contained inside VNF Packages and not for scripts referred using `scripts-link`. This VNFM may be assigned to the management of a single VNF instance, or the management of multiple VNF instances of the same type or of different types.





**Openstack4j**: OpenStack4j is an open source library that helps you manage an OpenStack deployment. It is a fluent based API giving you full control over the various OpenStack services. OpenStack is an opensource cloud operating system that provides a Versatile platform for computing storage and networking resources across bare metal, Virtual machines and containers.

