

# به نام خدا

پروژه دوم درس شبکه های کامپیوتری

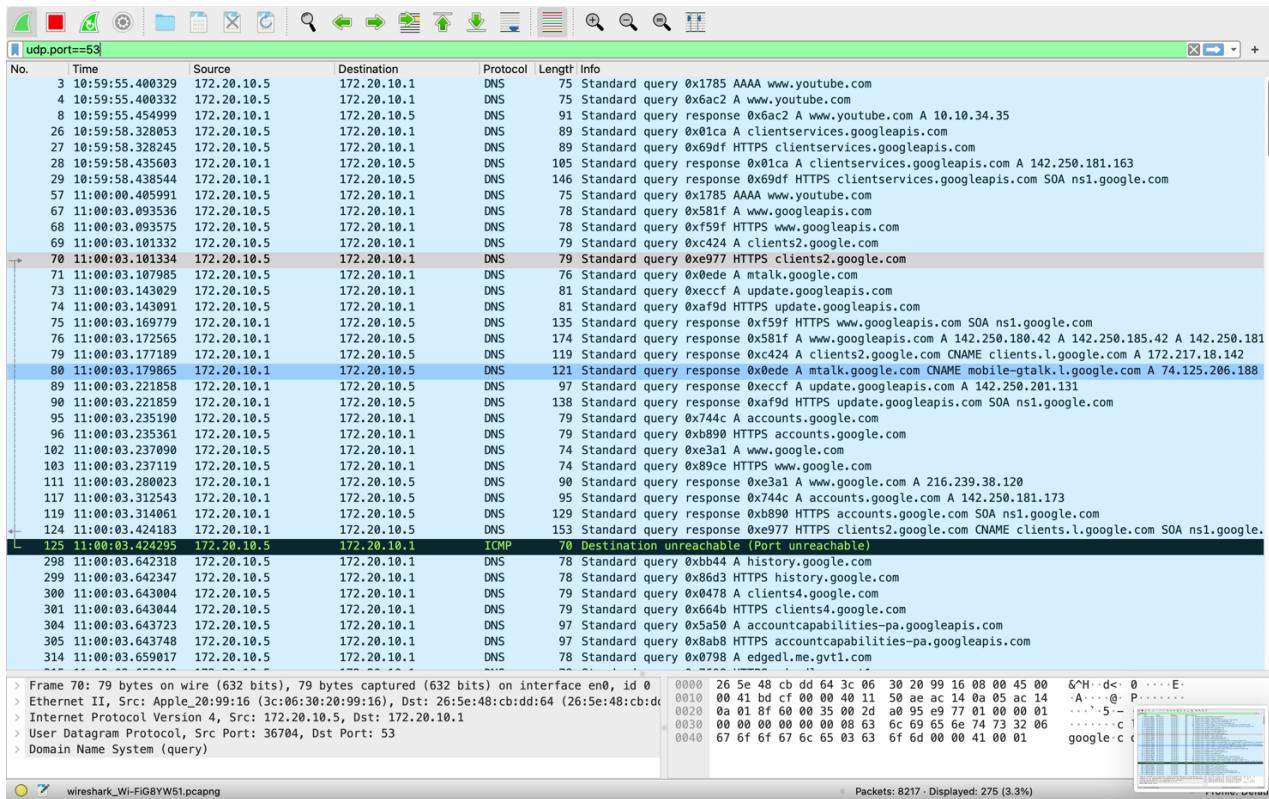
استاد حیدرپور

دستیار آموزشی : آرمان سعیدی

تهیه کننده : پرنیان ملک زاده



## سؤال (۱)



هنگام ترجمه آدرس ip، آبجکت ها و فایل های html را نیز ترجمه میکند. تا آن صفحه به صورت مناسب نمایش داده شود.

## سؤال (۲)

```
> Frame 28: 105 bytes on wire (840 bits), 105 bytes captured (632 bits) on interface en0, id 0
> Ethernet II, Src: Apple_20:99:16 (3c:06:30:99:16), Dst: 26:5e:48:cb:dd:64 (26:5e:48:cb:dd:64)
> Internet Protocol Version 4, Src: 172.20.10.1, Dst: 172.20.10.5
> User Datagram Protocol, Src Port: 53, Dst Port: 49348
    Source Port: 53
    Destination Port: 49348
    Length: 71
    Checksum: 0x00fc [unverified]
        [Checksum Status: Unverified]
        [Stream index: 6]
        [Timestamps]
        UDP payload (63 bytes)
< Domain Name System (response)
    Transaction ID: 0x01ca
    > Flags: 0x8180 Standard query response, No error
    Questions: 1
```

با توجه به تصویر مشاهده میشود که dns از پروتکل udp استفاده شده است. های استفاده شده ۱۰.۱ و ۱۷۲.۲۰.۱۰.۵ هستند که به ip router تعلق دارد. و علت رد و بدل شدن پیامها بین این دو ip این است که دیوایس ما از local dns server , query router به عنوان dns server عمل میکند. چون آدرس ۱۷۲.۱۶.۲۳۸.۲ ، Gateway router میباشد:

```
parntan@parntan:~$ ip route
default via 172.16.238.2 dev ens160 proto dhcp src 172.16.238.129 metric 100
```

## سوال (۳)

نتیجه برای h۱ ping h۲ و اسکرین واير شارک برای interface s۱ – eth۱ میباشد:

No.	Time	Source	Destination	Protocol	Length	Info
4	1.027545783	10.0.0.2	10.0.0.1	ICMP	98	Echo (ping) reply id=0x59ec,
5	2.029694704	10.0.0.1	10.0.0.2	ICMP	98	Echo (ping) request id=0x59ec,
6	2.029807028	10.0.0.2	10.0.0.1	ICMP	98	Echo (ping) reply id=0x59ec,
7	3.043906423	10.0.0.1	10.0.0.2	ICMP	98	Echo (ping) request id=0x59ec,
8	3.044009706	10.0.0.2	10.0.0.1	ICMP	98	Echo (ping) reply id=0x59ec,
9	3.203452532	d6:c9:84:6e:63:1b	36:3f:a4:dc:17:ba	ARP	42	Who has 10.0.0.1? Tell 10.0.0.2
10	3.203487529	36:3f:a4:dc:17:ba	d6:c9:84:6e:63:1b	ARP	42	10.0.0.1 is at 36:3f:a4:dc:17:ba
11	4.067630142	10.0.0.1	10.0.0.2	ICMP	98	Echo (ping) request id=0x59ec,
12	4.067742799	10.0.0.2	10.0.0.1	ICMP	98	Echo (ping) reply id=0x59ec,
13	5.091834159	10.0.0.1	10.0.0.2	ICMP	98	Echo (ping) request id=0x59ec,
14	5.091935526	10.0.0.2	10.0.0.1	ICMP	98	Echo (ping) reply id=0x59ec,

```
Frame 1: 98 bytes on wire (784 bits), 98 bytes captured
Ethernet II, Src: 36:3f:a4:dc:17:ba (36:3f:a4:dc:17:b
Internet Protocol Version 4, Src: 10.0.0.1, Dst: 10.0.0.1
Internet Control Message Protocol
0000  d6 c9 84 6e 63 1b 36 3f a4 dc 17 ba 08 00 45 00
0010  00 54 bd 7b 40 00 40 01 69 2b 0a 00 00 01 0a 00
0020  00 02 08 00 4e 6f 59 ec 00 24 1c d6 74 64 00 00
0030  00 00 ff 72 00 00 00 00 00 00 10 11 12 13 14 15
0040  16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25
0050  26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35
0060  36 37
```

## سوال (۴)

```
parntan@parntan:~$ cat /etc/resolv.conf
# This file is managed by man:systemd-resolved(8). Do not edit.
#
# This is a dynamic resolv.conf file for connecting local clients to the
# internal DNS stub resolver of systemd-resolved. This file lists all
# configured search domains.
#
# Run "resolvectl status" to see details about the uplink DNS servers
# currently in use.
#
# Third party programs must not access this file directly, but only through the
# symlink at /etc/resolv.conf. To manage man:resolv.conf(5) in a different way,
# replace this symlink by a static file or a different symlink.
#
# See man:systemd-resolved.service(8) for details about the supported modes of
# operation for /etc/resolv.conf.

nameserver 127.0.0.53
options edns0 trust-ad
search localdomain
```

این آدرس، آدرس IP یک سرور DNS است که توسط سیستم عامل برای نام‌گذاری دامنه‌ها استفاده می‌شود. در واقع، این آدرس به عنوان یکی از سرورهای DNS پیش‌فرض سیستم تنظیم شده است. هنگامی که کاربر یک نام دامنه را در مرورگر وارد می‌کند، سیستم عامل ابتدا به سرور DNS مربوطه درخواست می‌دهد آدرس IP مربوط به آن نام دامنه را بازگرداند و سپس به آن آدرس IP درخواست ارسال می‌شود.

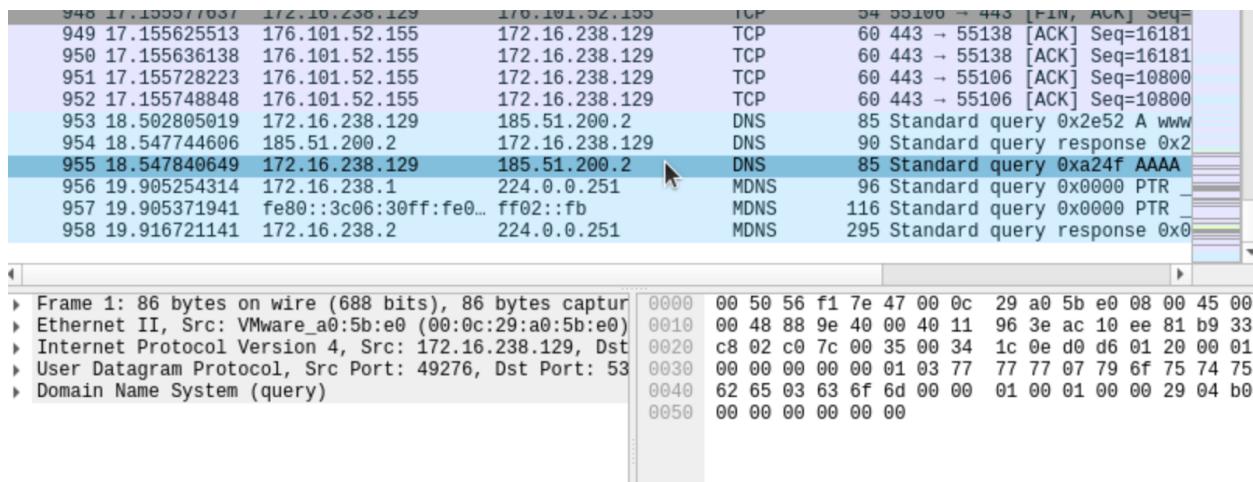
فایل `/usr/lib/systemd/resolv.conf` یک فایل پیکربندی است که توسط `systemd` برای تنظیم تنظیمات DNS سیستم استفاده می‌شود. این فایل شامل آدرس‌های IP سرورهای DNS است که سیستم برای نام‌گذاری دامنه‌ها استفاده می‌کند. وظیفه این فایل اطمینان حاصل کردن از اینکه سیستم می‌تواند به درستی با سرورهای DNS ارتباط برقرار کند و بتواند نام‌های دامنه را به آدرس‌های IP مربوطه تبدیل کند.

```
parnian@parnian:~$ cat /usr/lib/systemd/resolv.conf
# This file belongs to man:systemd-resolved(8). Do not edit.
#
# This is a static resolv.conf file for connecting local clients to the
# internal DNS stub resolver of systemd-resolved. This file lists no search
# domains.
#
# Run "resolvectl status" to see details about the uplink DNS servers
# currently in use.
#
# Third party programs must not access this file directly, but only through the
# symlink at /etc/resolv.conf. To manage man:resolv.conf(5) in a different way,
# replace this symlink by a static file or a different symlink.
#
# See man:systemd-resolved.service(8) for details about the supported modes of
# operation for /etc/resolv.conf.

nameserver 127.0.0.53
options edns0 trust-ad
```

```
parnian@parnian:~$ locate resolv.conf
/etc/resolv.conf
/etc/cloud/templates/resolv.conf.tmpl
/snap/core20/1614/etc/resolv.conf
/snap/core20/1614/etc/cloud/templates/resolv.conf.tmpl
/snap/core20/1614/run/systemd/resolve/stub-resolv.conf
/snap/core20/1614/usr/lib/systemd/resolv.conf
/snap/core20/1895/etc/resolv.conf
/snap/core20/1895/etc/cloud/templates/resolv.conf.tmpl
/snap/core20/1895/run/systemd/resolve/stub-resolv.conf
/snap/core20/1895/usr/lib/systemd/resolv.conf
/usr/lib/systemd/resolv.conf
/usr/share/man/man5/resolv.conf.5.gz
```

## سوال (۵)



با توجه به شکل بالا چون ip شکن را در فایل resolv.conf قرار دادیم اکنون این ip به عنوان local dns server میشود و کویری ها به این ip فرستاده میشود.

## سوال (۶)

### ۱. فایل /etc/resolv.conf:

این فایل برای تنظیمات DNS در لینوکس استفاده می شود. فایل resolv.conf شامل اطلاعات مربوط به سرور DNS است که توسط سیستم استفاده می شود.

### ۲. فایل /etc/hosts:

این فایل شامل نام های دامنه و آدرس های IP مربوط به آن ها است. این فایل معمولا برای تعریف نام های دامنه محلی استفاده می شود. برای مثال، اگر شما یک سرور وب درون شبکه خود دارید، می توانید نام آن را در فایل hosts تعریف کنید تا بتوانید به آن دسترسی پیدا کنید.

### ۳. فایل /etc/nsswitch.conf:

این فایل برای تعیین ترتیب استفاده از منابع مختلف برای جستجوی اطلاعات شبکه استفاده می شود. برای مثال، اگر شما یک نام کاربری را در سیستم جستجو کنید، nsswitch.conf مشخص می کند که اطلاعات باید از فایل /etc/passwd یا LDAP یا NIS بخواند. این فایل برای تنظیمات پیشرفته شبکه مفید است.

## سوال (۷)

```
root@parnian:/home/parnian# ip netns exec Host ping localhost
PING localhost (127.0.0.1) 56(84) bytes of data.
64 bytes from localhost (127.0.0.1): icmp_seq=1 ttl=64 time=0.182 ms
64 bytes from localhost (127.0.0.1): icmp_seq=2 ttl=64 time=0.152 ms
64 bytes from localhost (127.0.0.1): icmp_seq=3 ttl=64 time=0.134 ms
64 bytes from localhost (127.0.0.1): icmp_seq=4 ttl=64 time=0.109 ms
64 bytes from localhost (127.0.0.1): icmp_seq=5 ttl=64 time=0.131 ms
64 bytes from localhost (127.0.0.1): icmp_seq=6 ttl=64 time=0.133 ms
64 bytes from localhost (127.0.0.1): icmp_seq=7 ttl=64 time=0.128 ms
64 bytes from localhost (127.0.0.1): icmp_seq=8 ttl=64 time=0.116 ms
^C
--- localhost ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7169ms
rtt min/avg/max/mdev = 0.109/0.135/0.182/0.021 ms
```

همانطور که مشاهده میشود به آدرس IP ۱۲۷.۰.۰.۱ ترجمه می شود.

با دستور `ip link set lo up` در این namespace فعال شده و آدرس IP ۱۲۷.۰.۰.۱ به آن تخصیص داده می شود. بنابراین، آدرس ۱۲۷.۰.۰.۱ از دستور `ip link set lo up` IP ۱۲۷.۰.۰.۱ بدهست آمده است.

## سوال (۸)

باید به داخل etc یک فolder با نام hosts بسازیم و در آنجا یک فolder Host بسازیم و یک فایل جدید با نام host ایجاد کنیم و در نهایت آدرس localhost را در انجا قرار دهیم.

```
root@parnian:/home/parnian# cat /etc/hosts
127.0.0.3 localhost
127.0.1.1 parnian

# The following lines are desirable for IPv6 capable hosts
::1      ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

root@parnian:/home/parnian# ip netns exec Host ping localhost
PING localhost (127.0.0.3) 56(84) bytes of data.
64 bytes from localhost (127.0.0.3): icmp_seq=1 ttl=64 time=0.196 ms
64 bytes from localhost (127.0.0.3): icmp_seq=2 ttl=64 time=0.138 ms
64 bytes from localhost (127.0.0.3): icmp_seq=3 ttl=64 time=0.136 ms
64 bytes from localhost (127.0.0.3): icmp_seq=4 ttl=64 time=0.144 ms
64 bytes from localhost (127.0.0.3): icmp_seq=5 ttl=64 time=0.132 ms
64 bytes from localhost (127.0.0.3): icmp_seq=6 ttl=64 time=0.134 ms
64 bytes from localhost (127.0.0.3): icmp_seq=7 ttl=64 time=0.128 ms
^C
--- localhost ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6139ms
rtt min/avg/max/mdev = 0.128/0.144/0.196/0.021 ms
```

## سوال (۹)

```
parnian@parnian:~$ cat /etc/hosts
127.0.0.3 localhost
127.0.1.1 parnian
8.8.8.8 google.com
# The following lines are desirable for IPv6 capable hosts
::1      ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

چون آدرس ip google.com را نمیداند اینجا به مشکل بر میخوریم پس باید آدرس ip را در فایل /etc/hosts قرار دهیم.

## سوال (۱۱)

The **veth** devices are virtual Ethernet devices. They can act as tunnels between network namespaces to create a bridge to a physical network device in another namespace, but can also be used as standalone network devices.

**veth** devices are always created in interconnected pairs. A pair can be created using the command:

```
# ip link add <p1-name> type veth peer name <p2-name>
```

In the above, *p1-name* and *p2-name* are the names assigned to the two connected end points.

Packets transmitted on one device in the pair are immediately received on the other device. When either devices is down the link state of the pair is down.

## سوال (۱۲)

```
parnian@parnian:~$ sudo ip link set veth-Host netns Host
parnian@parnian:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens160: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq_codel state DOWN group default qlen 1000
0
    link/ether 00:0c:29:a0:5b:e0 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::20c:29ff:fea0:5be0/64 scope link
        valid_lft forever preferred_lft forever
4: veth-default@if3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 42:73:04:69:21:74 brd ff:ff:ff:ff:ff:ff link-netns Host
parnian@parnian:~$ ip netns exec Host ip a
setting the network namespace "Host" failed: Operation not permitted
parnian@parnian:~$ sudo ip netns exec Host ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
3: veth-Host@if4: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 3e:20:b2:bf:56:5b brd ff:ff:ff:ff:ff:ff link-netnsid 0
parnian@parnian:~$
```

همانطور که مشاهده میشود ، اینترفیس Host اضافه شده و به veth-dafault . default netns

## سوال (۱۳)

```
parnian@parnian:~$ sudo iptables -L -v
Chain INPUT (policy ACCEPT 4 packets, 746 bytes)
pkts bytes target     prot opt in     out     source               destination
                                                               destination

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target     prot opt in     out     source               destination
0      0 ACCEPT      all   --  wlp2s0  veth-default anywhere  anywhere
                                                               anywh
ere

Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target     prot opt in     out     source               destination
```

دستورات بالا به منظور ایجاد قوانین جدول FORWARD در iptables استفاده می شوند. این قوانین به این صورت عمل می کنند که ترافیکی که از دستگاه مجازی veth-default به سمت وایرلنس wlp2s0 رفته یا از وایرلنس wlp2s0 به سمت دستگاه مجازی veth-default برگشته، مجاز است و باید اجازه عبور داده شود.

دستور -L -v نیز لیستی از تمامی قوانین iptables را به همراه تعداد بسته هایی که توسط هر قانون تاثیر گذار بوده است، نمایش می دهد. این خروجی برای بررسی تاثیر قوانین iptables بر روی ترافیک شبکه بسیار مفید است.

## سوال (۱۴)

```
root@parnian:/home/parnian# ip netns exec R ip route
10.0.1.0/24 dev R-eth1 proto kernel scope link src 10.0.1.1
10.0.2.0/24 dev R-eth2 proto kernel scope link src 10.0.2.1
```

کامند بالا نشان میدهد که روتر ساخته شده ip ای که هر اینترفیس به آن متصل است میداند.

ابتدا با این کامند

```
# ip link add H1-eth0 type veth peer name R-eth1
# ip link add H2-eth0 type veth peer name R-eth2
```

ما دو لینک مجازی با نام های H1-eth0, H2-eth0 ساختیم و آنها را به ترتیب به R-eth1, R-eth2 متصل کردیم. و سپس توسط :

```
# ip link set H1-eth0 netns H1
# ip link set H2-eth0 netns H2
# ip link set R-eth1 netns
```

```
# ip link set R-eth2 netns R
```

H1 را توسط R-eth1 به R متصل و H2 را توسط R-eth2 به R متصل کردیم.  
و سپس به هر لینک یک آی پی اختصاص دادیم:

```
# ip netns exec H1 ifconfig lo up
# ip netns exec H1 ifconfig H1-eth0 10.0.1.2 netmask 255.255.255.0
# ip netns exec H1 ifconfig H1-eth0 up
# ip netns exec H2 ifconfig lo up
# ip netns exec H2 ifconfig H2-eth0 10.0.2.2 netmask 255.255.255.0
# ip netns exec H2 ifconfig H2-eth0 up
# ip netns exec R ifconfig lo up
# ip netns exec R ifconfig R-eth1 10.0.1.1 netmask 255.255.255.0 # ip netns exec R ifconfig R-eth1 up
# ip netns exec R ifconfig R-eth2 10.0.2.1 netmask 255.255.255.0 # ip netns exec R ifconfig R-eth2 up
```

H1-eth0 IP address = 10.0.1.2      —> peer with —>    R-eth1 IP address = 10.0.1.1  
H1-eth0 IP address = 10.0.2.2      —> peer with —>    R-eth2 IP address = 10.0.2.1

و در آخر:

```
# ip netns exec H1 ip route add default via 10.0.1.1 dev H1-eth0
# ip netns exec H2 ip route add default via 10.0.2.1 dev H2-eth0
```

که R را به عنوان دیفالٹ رپتر برای H1, H2 قرار میدهد.  
با اینکار اگر پکتی داشته باشیم که بخواهد از H1 به H2 ارسال شود، H1 آن را به روتر R میفرستد و چون رپتر ما میداند  
که توسط R-eth2 به H2 متصل است، پکت را به این لینک ارسال میکند.

```
root@parnian:/home/parnian# ip netns exec H1 ping -c 3 10.0.2.2
PING 10.0.2.2 (10.0.2.2) 56(84) bytes of data.
64 bytes from 10.0.2.2: icmp_seq=1 ttl=63 time=0.283 ms
64 bytes from 10.0.2.2: icmp_seq=2 ttl=63 time=0.211 ms
64 bytes from 10.0.2.2: icmp_seq=3 ttl=63 time=0.188 ms

--- 10.0.2.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2054ms
rtt min/avg/max/mdev = 0.188/0.227/0.283/0.040 ms
```

## سؤال (١٥)

```
parnian@parnian:~$ sudo ovs-ofctl show s
OFPT_FEATURES_REPLY (xid=0x2): dpid:0000a690469b7a44
n_tables:254, n_buffers:0
capabilities: FLOW_STATS TABLE_STATS PORT_STATS QUEUE_STATS ARP_MATCH_IP
actions: output enqueue set_vlan_vid set_vlan_pcp strip_vlan mod_dl_src mod_dl_dst
mod_nw_src mod_nw_dst mod_nw_tos mod_tp_src mod_tp_dst
1(s-eth1): addr:d6:a9:bc:e1:39:58
    config:      0
    state:       0
    current:    10GB-FD COPPER
    speed: 10000 Mbps now, 0 Mbps max
2(s-eth2): addr:c6:af:38:50:9d:a0
    config:      0
    state:       0
    current:    10GB-FD COPPER
    speed: 10000 Mbps now, 0 Mbps max
LOCAL(s): addr:a6:90:46:9b:7a:44
    config:    PORT_DOWN
    state:     LINK_DOWN
    speed: 0 Mbps now, 0 Mbps max
OFPT_GET_CONFIG_REPLY (xid=0x4): frags=normal miss_send_len=0
parnian@parnian:~$
```

## Actions:

1. **strip\_vlan**: This command removes the VLAN tag from an Ethernet frame, allowing it to be forwarded as an untagged frame. It is commonly used when a device does not support VLAN tagging or when the VLAN tag is no longer needed in the network.
2. **add\_vlan**: This command adds a VLAN tag to an Ethernet frame, allowing it to be identified and processed by devices that support VLANs. It is commonly used when configuring VLANs on a switch or router.
3. **set\_vlan**: This command sets the VLAN tag for an Ethernet frame to a specific value. It is commonly used when configuring VLANs on a switch or router.
4. **show\_vlan**: This command displays information about the VLANs configured on a switch or router, including the VLAN ID, name, and status.
5. **clear\_vlan**: This command removes all VLAN configuration from a switch or router, resetting it to its default state.

Overall, these commands are used to configure and manage VLANs in a network, allowing for better organization and segmentation of traffic.

## Capabilities:

1. FLOW\_STATE: This refers to the current state of a flow in a network switch or router. A flow is a set of packets that share common attributes, such as source and destination IP addresses, protocol type, and port numbers. The flow state includes information such as the number of packets and bytes in the flow, the time since the last packet was seen, and any actions that have been taken on the flow (such as forwarding or dropping).
2. TABLE\_STATE: This refers to the current state of a routing or forwarding table in a network switch or router. These tables are used to determine how to forward packets based on their destination addresses. The table state includes information such as the number of entries in the table, the time since the last update, and any changes that have been made to the table (such as adding or removing routes).
3. PORT\_STATE: This refers to the current state of a physical or logical port on a network switch or router. Ports are used to connect devices to the network and transfer data between them. The port state includes information such as whether the port is up or down, the speed and duplex settings of the port, and any errors or congestion that have been detected on the port.

## سؤال (۱۶)

```
actions=NORMAL
parnian@parnian:~$ sudo ovs-ofctl dump-flows s
cookie=0x0, duration=1840.250s, table=0, n_packets=38, n_bytes=2908, priority=0
actions=NORMAL
```

این جدول شامل رکوردهایی است که توصیف کننده جریان‌های (flows) مختلف در سوئیچ هستند. این رکورد خاص شامل اطلاعات زیر است:

cookie:

یک شناسه ۶۴ بیتی که توسط کنترل کننده شبکه برای شناسایی جریان‌ها استفاده می‌شود.

duration:

مدت زمانی که این جریان وجود دارد.

table:

شماره جدولی که این جریان در آن قرار دارد.

n\_packets:

تعداد بسته‌هایی که در این جریان ارسال شده است.

n\_bytes:

حجم داده‌ای که در این جریان ارسال شده است.

priority:

اولویت این جریان در مقایسه با سایر جریان‌ها.

actions:

عملیاتی که در این جریان انجام می‌شود، مانند ارسال به یک پورت خاص یا رها کردن بسته. در این حالت، عملیات **NORMAL** به معنای ارسال بسته به جدول بعدی است.

## سوال (۱۷)

```
root@parnian:/home/parnian# ip netns exec H2 ifconfig
H2-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 10.0.0.2 netmask 255.0.0.0 broadcast 10.255.255.255
        inet6 fe80::b467:7ff:feb2:a184 prefixlen 64 scopeid 0x20<link>
              ether b6:67:07:b2:a1:84 txqueuelen 1000 (Ethernet)
                    RX packets 41 bytes 4293 (4.2 KB)
                    RX errors 0 dropped 0 overruns 0 frame 0
                    TX packets 21 bytes 1678 (1.6 KB)
                    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
              I
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
              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


---


root@parnian:/home/parnian# ip netns exec 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
        inet6 fe80::2099:3fff:fecc:b0e7 prefixlen 64 scopeid 0x20<link>
              ether 22:99:3f:cc:b0:e7 txqueuelen 1000 (Ethernet)
                    RX packets 48 bytes 4879 (4.8 KB)
                    RX errors 0 dropped 0 overruns 0 frame 0
                    TX packets 20 bytes 1608 (1.6 KB)
                    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
              I
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
              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


---


root@parnian:/home/parnian# ovs-ofctl dump-flows s
  cookie=0x0, duration=198.789s, table=0, n_packets=6, n_bytes=252, dl_dst=22:99:3f:cc:b0:e7 actions=output:"s-eth2"
  cookie=0x0, duration=112.601s, table=0, n_packets=11, n_bytes=910, dl_dst=b6:67:07:b2:a1:84 actions=output:"s-eth1"
  cookie=0x0, duration=82.508s, table=0, n_packets=6, n_bytes=252, arp actions=NORMAL
```

```
root@parnian:/home/parnian# ip netns exec H1 ping 10.0.0.2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=0.664 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.177 ms
64 bytes from 10.0.0.2: icmp_seq=3 ttl=64 time=0.106 ms
64 bytes from 10.0.0.2: icmp_seq=4 ttl=64 time=0.189 ms
64 bytes from 10.0.0.2: icmp_seq=5 ttl=64 time=0.191 ms
64 bytes from 10.0.0.2: icmp_seq=6 ttl=64 time=0.159 ms
```

## بخش دوم

## سوال (۱)

```
root@parnian:/home/parnian# ip netns add Client
root@parnian:/home/parnian# ip netns add server
root@parnian:/home/parnian# ip link add H1-eth0 type veth peer name s-eth1
root@parnian:/home/parnian# ip link add H2-eth0 type veth peer name s-eth2
root@parnian:/home/parnian# ip link set H1-eth0 netns Client
root@parnian:/home/parnian# ip link set H2-eth0 netns Server
root@parnian:/home/parnian# ovs-vsctl add-port s s-eth1
root@parnian:/home/parnian# ovs-vsctl add-port s s-eth2
```

## سوال (۲)

```
root@parnian:/home/parnian# ip netns exec Client ifconfig lo up
root@parnian:/home/parnian# ip netns exec Client ifconfig H1-eth0 100.1.0.1
root@parnian:/home/parnian# ip netns exec Server ifconfig lo up
root@parnian:/home/parnian# ip netns exec Server ifconfig H2-eth0 100.1.0.2
root@parnian:/home/parnian#
```

## سوال (۳)

```
[Terminal] ian:/home/parnian# ifconfig s-eth1 100.1.0.3
[Terminal] ian:/home/parnian# ifconfig s-eth2 100.1.0.4
[Terminal] ian:/home/parnian# ip netns exec Client ip route add default via 100.1.0.3
root@parnian:/home/parnian# ip netns exec Server ip route add default via 100.1.0.4
```

```
root@parnian:/home/parnian# ip netns exec Server route
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref  Use Iface
default         100.1.0.4      0.0.0.0        UG    0      0      0 H2-eth0
100.0.0.0       0.0.0.0        255.0.0.0      U      0      0      0 H2-eth0
```

```
root@parnian:/home/parnian# ip netns exec Client route
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref  Use Iface
default         100.1.0.3      0.0.0.0        UG    0      0      0 H1-eth0
100.0.0.0       0.0.0.0        255.0.0.0      U      0      0      0 H1-eth0
```

## سوال (۴)

```
root@parnian:/home/parnian# ip link add World type veth
root@parnian:/home/parnian# ovs-vsctl add-port s World
root@parnian:/home/parnian# ip link set World up
root@parnian:/home/parnian# ifconfig World 100.1.0.5/24
root@parnian:/home/parnian# sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
root@parnian:/home/parnian# iptables -A FORWARD -o ens160 -i Sw-To-Wl -j ACCEPT
root@parnian:/home/parnian# iptables -A FORWARD -i ens160 -i Sw-To-Wl -o ACCEPT
```

ابتدا یک لینک برای ارتباط با دنیای خارج میسازیم، سپس آن را به سوییچ متصل میکنیم و در نهایت آن را به wifi متصل میکنیم.

## سوال (۵)

```
root@parnian:/home/parnian# ovs-vsctl add-flow switch "priority=101,ip_dst=100.1.0.1,actions=output:1"
ovs-vsctl: unknown command 'add-flow'; use --help for help
root@parnian:/home/parnian# ovs-ofctl add-flow switch "priority=101,ip_dst=100.1.0.1,actions=output:1"
2023-06-03T10:32:03Z|00001|ofp_match|INFO|normalization changed ofp_match, details:
2023-06-03T10:32:03Z|00002|ofp_match|INFO| pre: nw_dst=100.1.0.1
2023-06-03T10:32:03Z|00003|ofp_match|INFO|post:
root@parnian:/home/parnian# ovs-ofctl add-flow switch "priority=100,ip_dst=100.1.0.2,actions=output:2"
2023-06-03T10:32:28Z|00001|ofp_match|INFO|normalization changed ofp_match, details:
2023-06-03T10:32:28Z|00002|ofp_match|INFO| pre: nw_dst=100.1.0.2
2023-06-03T10:32:28Z|00003|ofp_match|INFO|post:
root@parnian:/home/parnian# ovs-ofctl add-flow switch "priority=50,actions=output:3"
```

## سوال (۶)

```
root@parnian:/home/parnian# ip netns exec clnt ping server.com
PING server.com (100.1.0.2) 56(84) bytes of data.
64 bytes from server.com (100.1.0.2): icmp_seq=1 ttl=64 time=1008 ms
64 bytes from server.com (100.1.0.2): icmp_seq=2 ttl=64 time=0.311 ms
64 bytes from server.com (100.1.0.2): icmp_seq=3 ttl=64 time=0.079 ms
64 bytes from server.com (100.1.0.2): icmp_seq=4 ttl=64 time=0.036 ms
64 bytes from server.com (100.1.0.2): icmp_seq=5 ttl=64 time=0.054 ms
64 bytes from server.com (100.1.0.2): icmp_seq=6 ttl=64 time=0.062 ms
64 bytes from server.com (100.1.0.2): icmp_seq=7 ttl=64 time=0.022 ms
64 bytes from server.com (100.1.0.2): icmp_seq=8 ttl=64 time=0.051 ms
64 bytes from server.com (100.1.0.2): icmp_seq=9 ttl=64 time=0.062 ms
64 bytes from server.com (100.1.0.2): icmp_seq=10 ttl=64 time=0.048 ms
^C
--- server.com ping statistics ---
I
```

```
root@parnian:/home/parnian# cat /etc/hosts
127.0.0.1 localhost
127.0.1.1 parnian
8.8.8.8 google.com
100.1.0.2 server.com
```

---

## سؤال ٧

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>My Website</title>
7 </head>
8 <body>
9   <h1>parnian malekzadeh</h1>
10  <h2>40017053</h2>
11 </body>
12 </html>
```

---

## سؤال ٨

```
root@parnian:/home/parnian# ip netns exec srvr python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
100.1.0.1 - - [04/Jun/2023 08:07:50] "GET / HTTP/1.1" 200 -
```

---

```
parnian@parnian:~$ sudo ip netns exec clnt wget server.com
[sudo] password for parnian:
--2023-06-04 08:07:50-- http://server.com/
Resolving server.com (server.com)... 100.1.0.2
Connecting to server.com (server.com)|100.1.0.2|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 257 [text/html]
Saving to: 'index.html.1'

index.html.1          100%[=====] 257  --.-KB/s

2023-06-04 08:07:50 (6.31 MB/s) - 'index.html.1' saved [257/257]
```

---

## سوال (۹)

```
root@parnian:/home/parnian# ip netns exec client wget github.com
--2023-06-04 19:05:29-- http://github.com/
Resolving github.com (github.com)... 192.30.255.113
Connecting to github.com (github.com)|192.30.255.113|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://github.com/ [following]
--2023-06-04 19:05:29-- https://github.com/
Connecting to github.com (github.com)|192.30.255.113|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'index.html.2'

index.html.2                                [          =>          ] 222.26K   217KB/s   in 1.0s

2023-06-04 19:05:32 (217 KB/s) - 'index.html.2' saved [227595]
```

همانطور که واضح است مانند بخش اول باید به `/etc/hosts` رفته و آدرس IP سایتی که میخواهیم به آن پینگ کیم در آن فایل قرار دهیم که چون در بخش اول `google.com` را قرار داده بودیم دیگر نیاز به این کار نبود.

## سوال (۱۰)

Client:

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
8: H1-eth0@if7: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether fa:e3:ee:c1:d0:6c brd ff:ff:ff:ff:ff:ff link-netnsid 0
        inet 100.1.0.1/24 brd 100.1.0.255 scope global H1-eth0
            valid_lft forever preferred_lft forever
        inet6 fe80::f8e3:eff:fec1:d06c/64 scope link
            valid_lft forever preferred_lft forever
```

Sever:

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
10: H2-eth0@if9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether e2:b4:b6:c7:25:b6 brd ff:ff:ff:ff:ff:ff link-netnsid 0
        inet 100.1.0.2/24 brd 100.1.0.255 scope global H2-eth0
            valid_lft forever preferred_lft forever
        inet6 fe80::e0b4:b6ff:fec7:25b6/64 scope link
            valid_lft forever preferred_lft forever
```

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	100.1.0.1				
2	0.000412987	100.1.0.2				
3	0.000452489	100.1.0.1				
4	0.000564540	100.1.0.1				
5	0.000586798	100.1.0.2				
6	0.001752593	100.1.0.2				
7	0.001771419	100.1.0.1				
8	0.001831795	100.1.0.2				
9	0.001840978	100.1.0.1				
10	0.001916792	100.1.0.2				
11	0.003129099	100.1.0.1				
12	0.003165085	100.1.0.2				
13	5.081901127	fa:e3:ee:c1:d0:6c	e2:b4:b6:c7:25:b6	ARP	42 Who has 100.1.0.2? Tell 100.1.0.1	
14	5.082189266	e2:b4:b6:c7:25:b6	fa:e3:ee:c1:d0:6c	ARP	42 Who has 100.1.0.1? Tell 100.1.0.2	
15	5.082203922	fa:e3:ee:c1:d0:6c	e2:b4:b6:c7:25:b6	ARP	42 100.1.0.1 is at fa:e3:ee:c1:d0:6c	
16	5.082208175	e2:b4:b6:c7:25:b6	fa:e3:ee:c1:d0:6c	ARP	42 100.1.0.2 is at e2:b4:b6:c7:25:b6	

مک آدرس مربوط به لینک است و از یک hop به hop دیگر متفاوت میباشد. پس پکت ارسالی مک آدرس سرور که اولین hop بعد از سوییج است را دارد تا سوییج بداند آن را به چه پورتی ارسال کند پس پکت ارسالی ابتدا به سوییج میرود تا بعد از آن به سرور برود.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	100.1.0.1	100.1.0.2	TCP	74	39346 - 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK
2	0.000412987	100.1.0.2	100.1.0.1	TCP	74	80 - 39346 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SACK
3	0.000452489	100.1.0.1	100.1.0.2	TCP	66	39346 - 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 Tsv=30444
4	0.000564540	100.1.0.1	100.1.0.2	HTTP	191	GET / HTTP/1.1
5	0.000586798	100.1.0.2	100.1.0.1	TCP	66	80 - 39346 [ACK] Seq=1 Ack=126 Win=65152 Len=0 Tsv=30444
6	0.001752593	100.1.0.2	100.1.0.1	TCP	252	80 - 39346 [PSH, ACK] Seq=1 Ack=126 Win=65152 Len=186 Tsv=30444
7	0.001771419	100.1.0.1	100.1.0.2	TCP	66	39346 - 80 [ACK] Seq=126 Ack=187 Win=64128 Len=0 Tsv=30444
8	0.001831795	100.1.0.2	100.1.0.1	HTTP	278	HTTP/1.0 200 OK (text/html)
9	0.001840978	100.1.0.1	100.1.0.2	TCP	66	39346 - 80 [ACK] Seq=126 Ack=391 Win=64128 Len=0 Tsv=30444
10	0.001916792	100.1.0.2	100.1.0.1	TCP	66	80 - 39346 [FIN, ACK] Seq=391 Ack=126 Win=65152 Len=0 Tsv=30444
11	0.003129099	100.1.0.1	100.1.0.2	TCP	66	39346 - 80 [FIN, ACK] Seq=126 Ack=392 Win=64128 Len=0 Tsv=30444
12	0.003165085	100.1.0.2	100.1.0.1	TCP	66	80 - 39346 [ACK] Seq=392 Ack=127 Win=65152 Len=0 Tsv=30444
13	5.081901127	fa:e3:ee:c1:d0:6c	e2:b4:b6:c7:25:b6	ARP	42 Who has 100.1.0.2? Tell 100.1.0.1	
14	5.082189266	e2:b4:b6:c7:25:b6	fa:e3:ee:c1:d0:6c	ARP	42 Who has 100.1.0.1? Tell 100.1.0.2	
15	5.082203922	fa:e3:ee:c1:d0:6c	e2:b4:b6:c7:25:b6	ARP	42 100.1.0.1 is at fa:e3:ee:c1:d0:6c	
16	5.082208175	e2:b4:b6:c7:25:b6	fa:e3:ee:c1:d0:6c	ARP	42 100.1.0.2 is at e2:b4:b6:c7:25:b6	