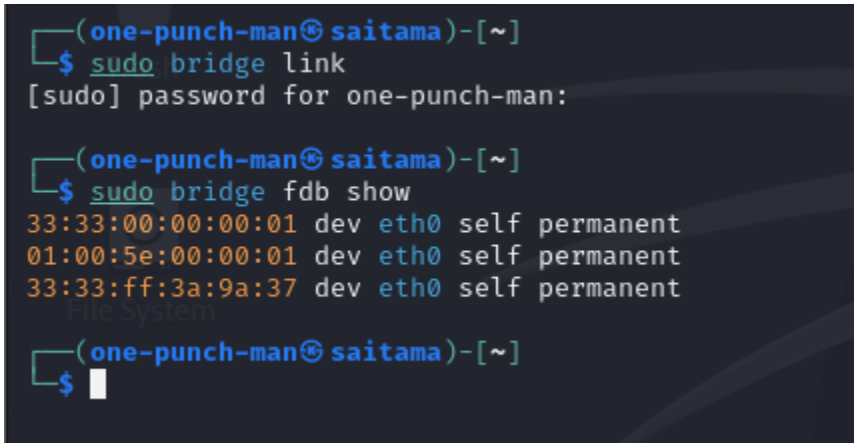
Method 1: Using the bridge Command
The bridge command is straightforward for inspecting the MAC address table of a bridge device.



Method 2 using ip command :

A bridge is created to check the config (as previously there was no bridge)

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip link add name br0 type bridge

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip link set br0 up

┌──(one-punch-man㉿saitama)-[~]
└─$ ip link show type bridge
3: br0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UNKNOWN mode DEFAULT group default qlen 1000
link/ether 32:53:a0:77:f2:5f brd ff:ff:ff:ff:ff:ff

Adding interfaces to the bridge

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip link add name dummy0 type dummy

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip link set dummy0 up

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip link set eth0 master br0

┌──(one-punch-man㉿saitama)-[~]

└─$ sudo ip link set dummy0 master br0

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip link set eth0 up

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo bridge link

2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 master br0 state forwarding priority 32 cost 5
4: dummy0: <BROADCAST,NOARP,UP,LOWER_UP> mtu 1500 master br0 state forwarding priority 32 cost 100

Populating the MAC table
FDB(forwarding database) builds as the bridge learns MAC addresses from traffic.

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip netns add ns1

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip netns add ns2

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip link add veth1 type veth peer name veth1-br

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip link add veth2 type veth peer name veth2-br

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip link set veth1 netns ns1

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip link set veth2 netns ns2

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip link set veth1-br master br0

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip link set veth2-br master br0

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip netns exec ns1 ip link set veth1 up

```
┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip netns exec ns2 ip link set veth2 up

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip link set veth1-br up

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip link set veth2-br up
```

Assigning IP's and Trying ping command to verify connection

```
┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip netns exec ns1 ip addr add 192.168.1.1/24 dev veth1

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip netns exec ns2 ip addr add 192.168.1.2/24 dev veth2

┌──(one-punch-man㉿saitama)-[~]
└─$ sudo ip netns exec ns1 ping 192.168.1.2 -c 10
PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.
64 bytes from 192.168.1.2: icmp_seq=1 ttl=64 time=0.131 ms
64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=0.136 ms
64 bytes from 192.168.1.2: icmp_seq=3 ttl=64 time=0.086 ms
^C
--- 192.168.1.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2042ms
rtt min/avg/max/mdev = 0.086/0.117/0.136/0.022 ms
```

Result shows the entries in the MAC table

```
┌──(one-punch-man㉿saitama)-[~]
└─$ sudo bridge fdb show
00:50:56:f3:4d:5b dev eth0 master br0
00:50:56:c0:00:08 dev eth0 master br0
00:0c:29:3a:9a:37 dev eth0 vlan 1 master br0 permanent
00:0c:29:3a:9a:37 dev eth0 master br0 permanent
33:33:00:00:00:01 dev eth0 self permanent
01:00:5e:00:00:01 dev eth0 self permanent
33:33:ff:3a:9a:37 dev eth0 self permanent
33:33:00:00:00:01 dev br0 self permanent
01:00:5e:00:00:6a dev br0 self permanent
33:33:00:00:00:6a dev br0 self permanent
01:00:5e:00:00:01 dev br0 self permanent
33:33:ff:77:f2:5f dev br0 self permanent
```

76:39:a2:64:8f:3e dev dummy0 vlan 1 master br0 permanent
76:39:a2:64:8f:3e dev dummy0 master br0 permanent
33:33:00:00:00:01 dev dummy0 self permanent
01:00:5e:00:00:01 dev dummy0 self permanent
42:a1:56:68:5a:ba dev veth1-br master br0
2e:43:4e:4e:f3:08 dev veth1-br vlan 1 master br0 permanent
2e:43:4e:4e:f3:08 dev veth1-br master br0 permanent
33:33:00:00:00:01 dev veth1-br self permanent
01:00:5e:00:00:01 dev veth1-br self permanent
33:33:ff:4e:f3:08 dev veth1-br self permanent
2a:b4:2b:38:7f:3d dev veth2-br master br0
7e:42:c8:15:dc:86 dev veth2-br vlan 1 master br0 permanent
7e:42:c8:15:dc:86 dev veth2-br master br0 permanent
33:33:00:00:00:01 dev veth2-br self permanent
01:00:5e:00:00:01 dev veth2-br self permanent
33:33:ff:15:dc:86 dev veth2-br self permanent