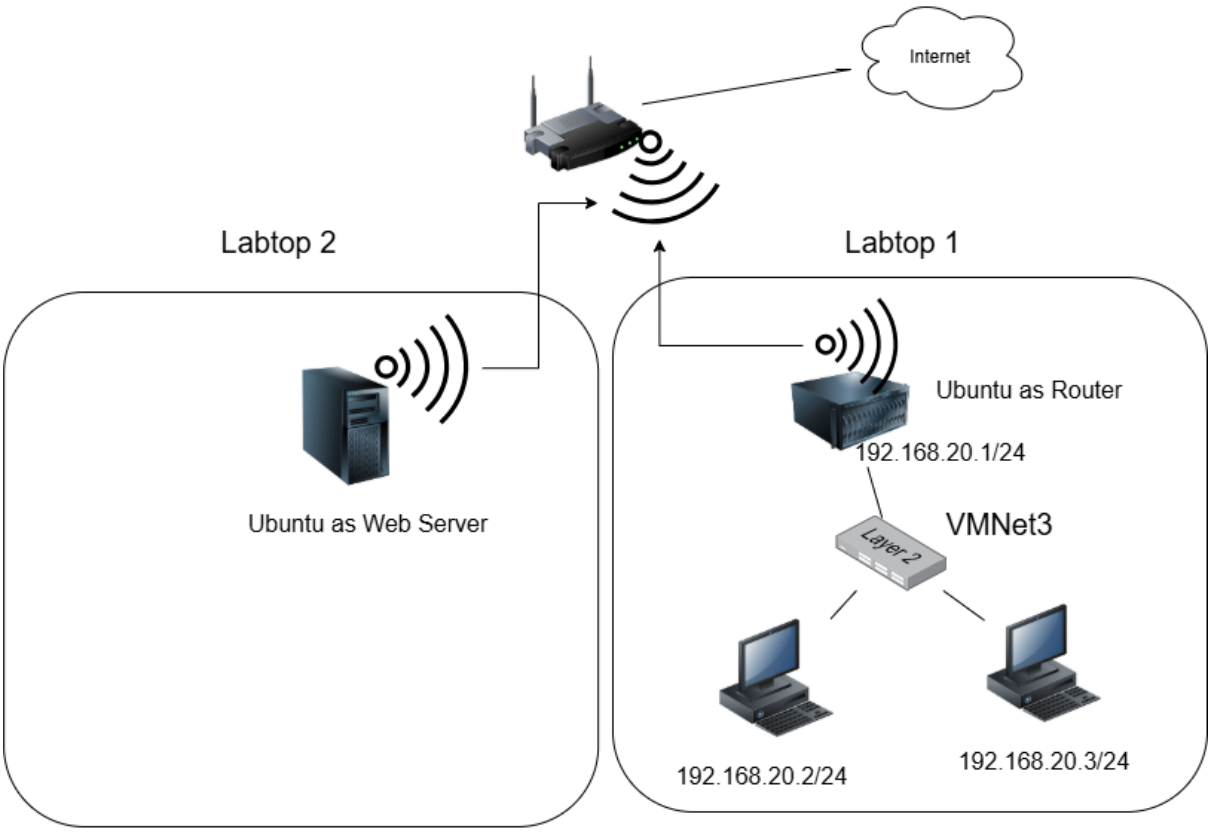
LAB – MIDDLE TERM ASSESSMENT



Requirements:

1/ One group has at least two students. The labtop of one student must been installed and set up an Ubuntu Server Virtual Machine that contains Docker and deploy a web site in the Docker. The other labtops must have an Ubuntu Server (or Windows Server) Virtual Machine configured as Router, one or two Workstations Virtual Machines (Win10, Win11 or Ubuntu Desktop) connected to each others via Lan Segment) like Labtop1 on the image. (3 marks)

2/ All of Workstations ping the the web server successfully. (3 marks)

3/ All of Workstations access the web site via IP address (2 marks)

4/ All of Workstations access the web site via domain name (2 marks)

2.1 Routing with Ubuntu Server:

2.2 Network Address Translation in Ubuntu Server:

2.2.1 Give the NICs IPs (Netplan)

Edit /etc/netplan/01-router.yaml:

```
network:
  version: 2
  ethernets:
    ens37:              # WAN
      dhcp4: true       # or set a static address+gateway from your provider
      dhcp6: false
    ens33:              # LAN
      addresses: [192.168.20.1/24]
      dhcp4: false
      dhcp6: false
      # IMPORTANT: no gateway on the LAN interface
```

Apply: ***sudo netplan apply***

2.2.2 Enable IP forwarding (kernel)

```
sudo tee /etc/sysctl.d/99-router.conf >/dev/null <<'EOF'
net.ipv4.ip_forward = 1
# Optional but recommended when routing between subnets:
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.default.accept_redirects = 0
net.ipv4.conf.all.send_redirects = 0
net.ipv4.conf.default.send_redirects = 0
EOF

sudo sysctl --system
```

Verify:

```
sysctl net.ipv4.ip_forward   # should be 1
```

2.2.3 NAT + firewall with nftables

Install & enable:

```
sudo apt update
sudo apt install -y nftables
sudo systemctl enable --now nftables
```

Write /etc/nftables.conf:

```
flush ruleset

table inet filter {
```

```
  chain input {
    type filter hook input priority 0;
    policy drop;
    ct state established,related accept
    iifname "lo" accept
    # Allow SSH to the router from the LAN (optional)
    iifname "ens33" tcp dport 22 accept
    # If you run dnsmasq on this box for the LAN
    iifname "ens33" udp dport { 53, 67 } accept
    iifname "ens33" tcp dport 53 accept
    # Allow ping to the router itself (v4 & v6)
    ip protocol icmp accept
    ip6 nexthdr icmpv6 accept
  }
  chain forward {
    type filter hook forward priority 0;
    policy drop;
    ct state established,related accept
    # Allow LAN -> WAN
    iifname "ens33" oifname "ens37" accept
    # (WAN -> LAN stays dropped by policy)
  }
  chain output {
```

```
    type filter hook output priority 0;
    policy accept;
  }
}
table ip nat {
  chain prerouting {
    type nat hook prerouting priority -100;
  }
  chain postrouting {
    type nat hook postrouting priority 100;
    # MASQUERADE LAN addresses when leaving via WAN
    oifname "ens37" ip saddr 192.168.20.0/24 masquerade
  }
}
```

Reload rules:

```
sudo nft -f /etc/nftables.conf
nft list ruleset
```

2.2.4 (Optional) Hand out IPs to LAN clients (dnsmasq)

If you want the router to provide DHCP/DNS on the LAN:

```
sudo apt install -y dnsmasq
sudo tee /etc/dnsmasq.d/lan.conf >/dev/null <<'EOF'
interface=ens33
bind-interfaces
# DHCP pool
dhcp-range=192.168.20.100,192.168.20.200,12h
# Default gateway (this router)
dhcp-option=3,192.168.20.1
# DNS servers (this router, then public)
```

```
dhcp-option=6,192.168.20.1,8.8.8.8
# Optional: provide domain
# domain=lan
EOF

sudo systemctl restart dnsmasq
```

(If you already run another DHCP on that segment, **don't** enable this.)