Routing

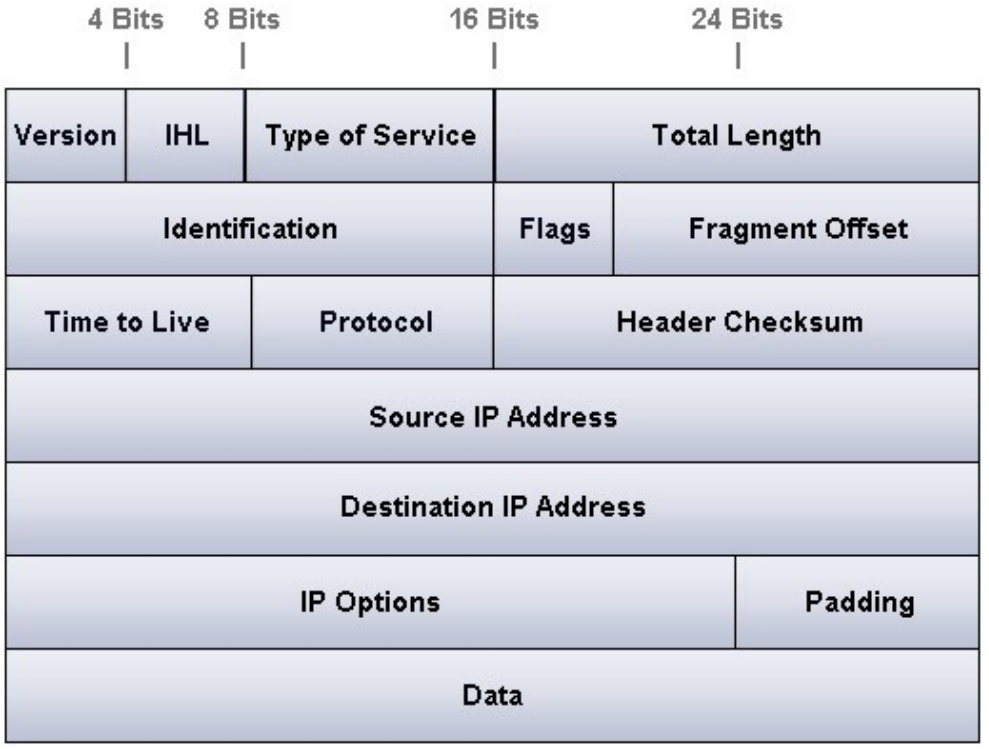
Linux System Administration
Spring 2026

Debugging Routing

- `ping`
  - Tests end-to-end connectivity
- `traceroute` or (or `tracert` on Windoze)
  - Shows the route the packet takes
  - Linux traceroute use UDP packets
- `ip route` or `route` or `netstat -nr`
  - Configure or show routing

Anatomy of an IP packet

- Always big-endian
- Version=4
- IHL=IP Hdr Len
- Type of Service
  - Min delay
  - Max throughput
- Flags & Frag Off
  - Large packets
- Protocols add additional header in data section



Configuring an IPv4 interface

- Must have
  - IP address
  - Netmask
  - Default gateway
- Nice to have
  - DNS server address
  - NTP server address

- Example:
  - `IP 100.64.0.42/24`
  - `GW 100.64.0.1`
  - `DNS1 128.138.240.1`
    `DNS2 128.138.130.30`
  - `NTP 132.163.97.1`
    - `time-a.wwv.nist.gov`

Routing

- Router connects multiple subnets
- The next hop must be reachable at Layer 2
- Each subnet has an associated target
  - via (IP address of gateway)
    - Packet sent to the MAC address of the gateway
  - interface
    - Packet sent to the MAC address of destination
- Selects **best** match out of all possibilities

Routing Example

- Destination IP in packet 100.64.42.6
- Routing Table (*gateway and MAC not shown*)

ifce	subnet	remarks
eno1	**0.0.0.0/0**	match 0 bits (default gateway)
eno2	**128.0.0.0/8**	does not match 1st octet
eno3	**100.0.0.0/8**	match 8 bits
eno4	**100.64.0.0/16**	match 16 bits
eno5	**100.64.42.0/24**	**match 24 bits (best match)**
eno6	**100.64.10.0/24**	does not match 3rd octet
eno7	**100.64.42.8/29**	does not match 5 bits in 4th octet

- Packet sent over interface **eno5**

Example: Machine A

- Interfaces

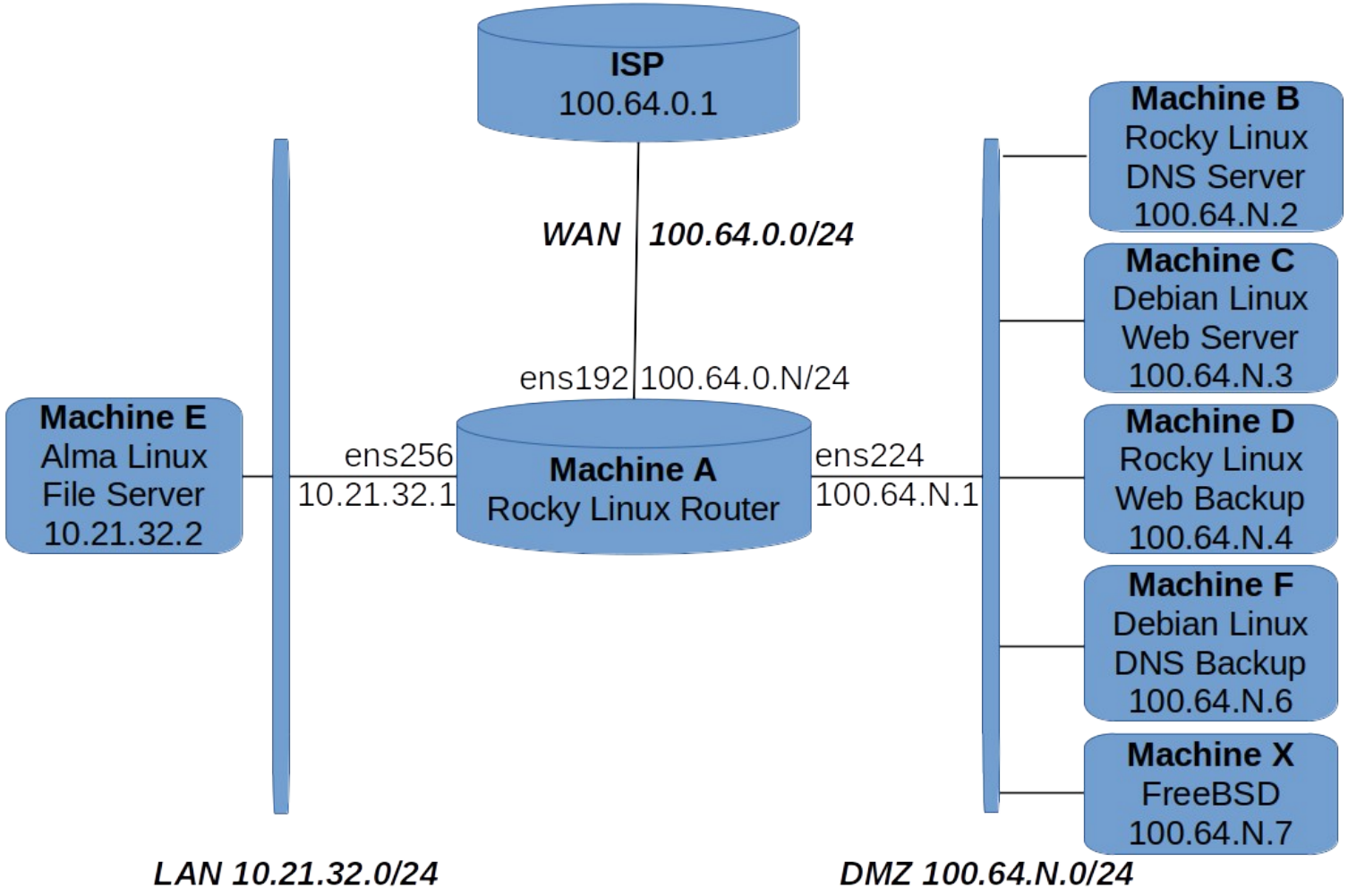
```
ens192 WAN IP 100.64.0.42/24 NET 100.64.0.0/24   gateway 100.64.0.1
ens224 DMZ IP 100.64.42.1/24 NET 100.64.42.0/24
ens256 LAN IP 10.21.32.1/24  NET 10.21.32.0/24
```

- No gateway for DMZ and LAN because Machine A is actually the gateway
- All routes except default added by kernel

```
ip route
default via 100.64.0.1 dev ens192 proto static metric 100
10.21.32.0/24  dev ens256 proto kernel scope link src 10.21.32.1  metric 102
100.64.0.0/24  dev ens192 proto kernel scope link src 100.64.0.42 metric 100
100.64.42.0/24 dev ens224 proto kernel scope link src 100.64.42.1 metric 101
```

- NAT LAN to WAN (`/etc/sysconfig/nftables.conf`)

```
table ip nat {
  chain POSTROUTING {
  type nat hook postrouting priority srcnat; policy accept;
  oifname "ens192" ip saddr 10.21.32.0/24  masquerade
  }
}
```


ISP
100.64.0.1
WAN
100.64.0.0/24
ens192
100.64.0.N/24
Machine A
Rocky Linux Router
Machine E
Alma Linux
File Server
10.21.32.2
ens256
10.21.32.1
ens224
100.64.N.1
Machine B
Rocky Linux
DNS Server
100.64.N.2
Machine C
Debian Linux
Web Server
100.64.N.3
Machine D
Rocky Linux
Web Backup
100.64.N.4
Machine F
Debian Linux
DNS Backup
100.64.N.6
Machine X
FreeBSD
100.64.N.7
LAN 10.21.32.0/24
DMZ 100.64.N.0/24

traceroute

- Run on N=41 Machine C `100.64.41.3`
- traceroute 100.64.42.3

1	`100.64.41.1`	N=41 Machine A DMZ
2	`100.64.0.1`	ISP Gateway
3	`100.64.0.42`	N=42 Machine A WAN
4	`100.64.42.3`	N=42 Machine C

- traceroute 100.64.0.42

1	`100.64.41.1`	N=41 Machine A DMZ
2	`100.64.0.42`	N=42 Machine A WAN

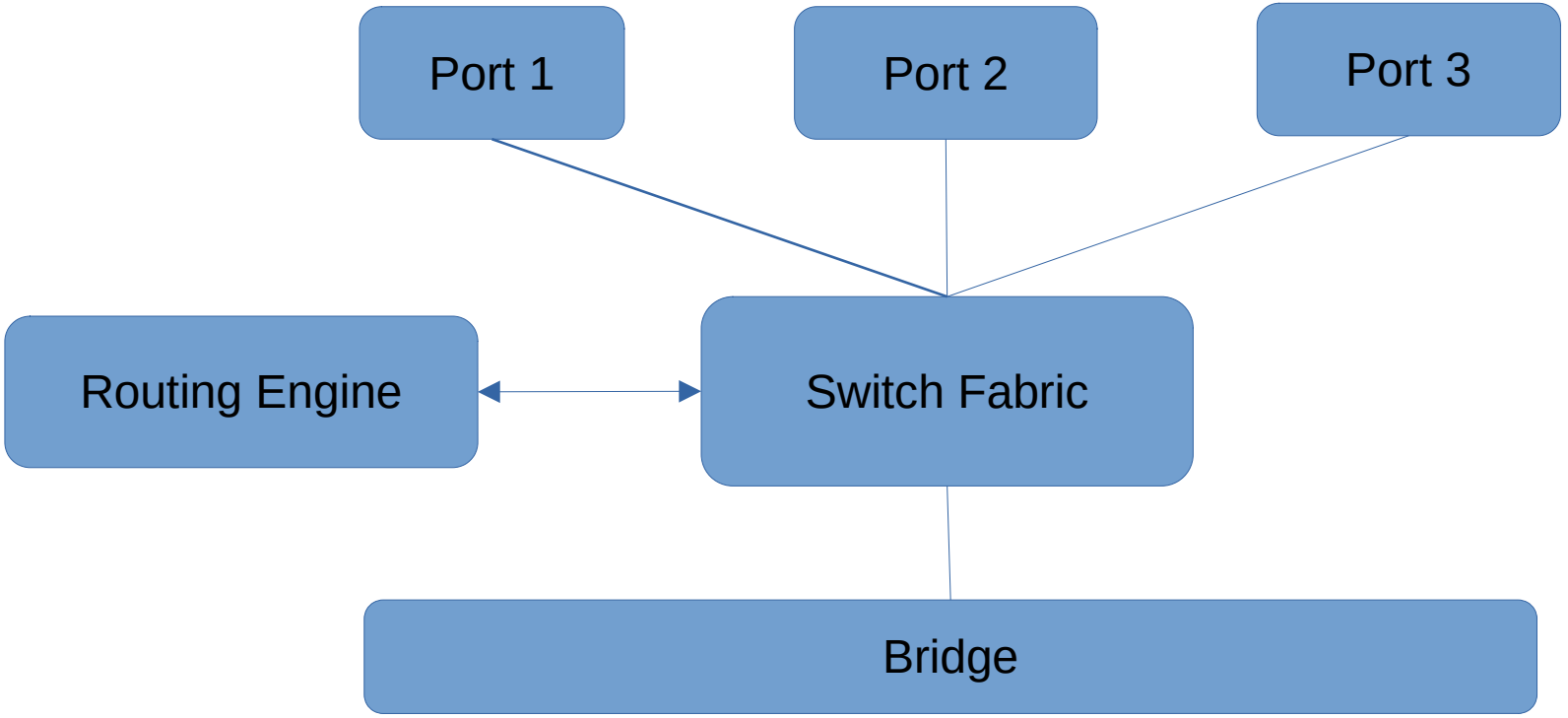
Dynamic Routing

- `OSPF` **O**pen **S**hortest **P**ath **F**irst
  - Interior gateway protocol
    - Weighted link state Routing
  - Mostly used in enterprise level interior networks
- `BGP` **B**order **G**ateway **P**rotocol
  - Exterior gateway protocol
    - Path-vector routing
  - Runs most of the internet

Linux routing

- Static routing using the kernel
  - `/etc/sysctl.conf`
    `net.ipv4.ip_forward = 1`
- Dynamic routing with Quagga (fork of GNU Zebra)
  - `ospfd`, `ospf6d` - OSPF 2 & 3
  - `bgpd` - BGP
- Without a forwarding plane, throughput is limited by CPU
  - Use a hardware router instead

Anatomy of a router



Network Address Translation (NAT)

- Port Address Translation (masquerade)
  - Remaps address and port
    `10.21.32.2:9876` becomes `100.64.42.1:7654`
  - Outbound:
    - Replace source IP and port with public IP and port
  - Inbound to NAT port:
    - Replace destination IP and port with private IP and port
- ICMP or port-less protocols use other packet data