LPI 109.1 - Fundamentals of internet protocols

Curs 2021 - 2022

ASIX M01-ISO 109 Networking Fundamentals

Fundamentals of internet protocols

Description

Key concepts:

- ❏ Demonstrate an understanding of network masks and CIDR notation.
- ❏ Knowledge of the differences between private and public "dotted quad" IP addresses.
- ❏ Knowledge about common TCP and UDP ports and services (20, 21, 22, 23, 25, 53, 80, 110, 123, 139, 143, 161, 162, 389, 443, 465, 514, 636, 993, 995).
- ❏ Knowledge about the differences and major features of UDP, TCP and ICMP.
- ❏ Knowledge of the major differences between IPv4 and IPv6.
- ❏ Knowledge of the basic features of IPv6.

Commands and files:

- ❏ /etc/services
- ❏ IPv4, IPv6
- ❏ Subnetting
- ❏ TCP, UDP, ICMP

Fundamentals of internet protocols

In almost all cases, a computer will be connected to a network in order to provide the access that users need. Providing access to the network is the responsibility of the root user on the system.

- TCP/IP Protocol
- IP address (static, dynamic fixed/range)
- Routing table
- LAN / WAN
- IPv4 / IPv6
- OSI Model (layers and PDUs)

Most networks today, including all computers on the internet, use the TCP/IP protocol developed by the Internet Engineering Task Force (IEFT) as the standard for how to communicate on the network.

In the TCP/IP model, the unique identifier for a computer is its IP address. An IP address can be either static, which is assigned manually, or dynamic, which is assigned by the Dynamic Host Configuration Protocol (DHCP) running as a service on the network.

A routing table is a small in-memory database used to calculate the optimal journey through other routers in the same or other networks for messages it is responsible for forwarding to a destination address.

TCP/IP is a combination of the two most popular protocols. The TCP (Transmission Control Protocol) protocol is used for reliable delivery of data between computers connected through a LAN (Local Area Network) or the internet, and the IP (Internet Protocol) protocol is mainly responsible for the routing of data packets to the destination address.

There are two versions of the IP protocol used across the internet, IP version 4 (IPv4) and the more current IP version 6 (IPv6), with IPv4 being in use on the majority of the systems today.

The OSI Model is comprised of seven different conceptual layers, each responsible for a different function within the network and each with its own protocol data unit (PDU), which represents the unit of data handled at that layer. The following is a summary of the layers in the OSI model:

7. Application
   User interface, Application Programming Interface (API)
6. Presentation
   Data representation, encryption
5. Session
   Controls connections between hosts, maintains ports and sessions
4. Transport
   Uses transmission protocols to transmit data (TCP, UDP)
3. Network
   Determines path of data, IP
2. Data Link
   Physical addressing (MAC), delivery of frames (Protocol Data Units (PDU))
1. Physical
   Transmits raw data between physical media

IPv4 Addresses

The IP is the protocol responsible for the logical addressing of a host, enabling the packet to be sent from one host to another. For this each device on the network is assigned a unique IP address, and it is possible to assign more than one address to the same device.

The IP address is used to identify the network interface of a device (i.e., phone, computer, etc.) on the network. The IP address is a numerical identifier in decimal dotted quad format since there are four values in an IPv4 address (for example, 192.224.10.8).

he IP address is made up of 4 octets, which are sets of 8-bit values. The value of each of these octets can range from decimal values 0 – 255 (or in binary, 00000000-11111111).

```
IP: 10.10.8.1
00001010. 00001001. 00001000. 00000001
|______|  |______|  |______|  |______|
    |         |         |         |
Octet #1   Octet #2   Octet #3  Octet #4
```

Network IP Classes

An IPv4 network addressing scheme has been designed on the basis of the octets. It classifies networks into 5 classes: A, B, C, D, and E.

Class A

The network is denoted by the first octet, and the remaining three octets are used to create subnets (to be discussed) or identify hosts on the network. The first bit of the first octet is always 0, so the range of values permissible is 00000001 – 01111111, i.e., 1 – 127 in decimal value (the first number of an IP address cannot be 0 by the definition of IP addresses). The network ID cannot have all bits set to either 1s or 0s, which means the total number of class A networks available is only 127. However, the 127 network is a special network referred to as a loopback (127.0.0.1) network, not a real class A network that is used on the internet. An example of a class A address would be 65.16.45.126. [1-127.h.h.h]

Class B

The network is denoted by the 1st and 2nd octets, and the remaining 2 octets are used to create subnets or identify hosts. The 1st and 2nd bits of the 1st octet are set to 1 and 0 respectively, so the range of values permissible is 10000000 – 10111111, i.e., decimals 128 - 191. An example of a Class B address would be 165.16.45.126. [128-192.n.h.h]

Class C

The network is denoted by the 1st, 2nd, and 3rd octets, and the last octet is used to create subnets or identify hosts. The 1st, 2nd, and 3rd bits of the 1st octet are set to 1, 1, and 0 respectively, so the range of values permissible is 11000000 – 11011111, i.e., decimals 192 – 223. An example of a Class C address would be 205.16.45.126. [192-223.n.n.h]

Class D

These addresses are not assigned to network interfaces and are used for multicast operations such as audio-video streaming. The 1st, 2nd, 3rd, and 4th bits of the first octet are set to 1, 1, 1, and 0 respectively, so the range of values permissible is 11100000 - 11101111, i.e., decimals 224 - 239. An example of a Class D address would be 224.0.0.6. [224-239.m.m.m]

Class E

These addresses are reserved for future use.

Class	First Octect	Range	Example
A	1-126	`1.0.0.0` – `126.255.255.255`	`10.25.13.10`
B	128-191	`128.0.0.0` – `191.255.255.255`	`141.150.200.1`
C	192-223	`192.0.0.0` – `223.255.255.255`	`200.178.12.242`

Network Masks & Subnetting

In order for computers to communicate directly on the same network, all of the computers must be on the same subnet. A subnet is either an entire class A, B, or C network or a portion of one of these networks. To take a large class network and create a smaller portion, use a subnet mask.

The subnet mask is used to differentiate the network and subnet components of the IP address. The subnet mask is not an IP address in itself; it is a numeric pattern used to indicate the portion of the IP address that contains the network identifier and the host address part.

In the subnet mask, the octets where the mask bits are 1 represent the network ID, whereas the octets where the mask bits are 0 represent the host ID.

The addresses for Class A, B, and C have default masks as follows:

Class A	255.0.0.0	/8
Class B	255.255.0.0	/16
Class C	255.255.255.0	/24

Network IP/mask example

```
IP address 10.9.8.1
Net Mask   255.0.0.0

IP address 00001010. 00001001. 00001000. 00000001
Mask       11111111. 00000000. 00000000. 00000000

           11111111. 00000000. 00000000. 00000000
           |______|  |__________________________|
               |                   |
           Network ID           Host ID
```

```
IP Address 10.9.8.1, the network ID is 10 and the host ID is 9.8.1.
Network IP address 10.0.0.0
Host IP address    10.9.8.1
```

Decimal	Binary	CIDR
`255.0.0.0`	`11111111.00000000.00000000.00000000`	`8`
`255.255.0.0`	`11111111.11111111.00000000.00000000`	`16`
`255.255.255.0`	`11111111.11111111.11111111.00000000`	`24`

Class	First Octet	Range	Default Mask
A	1-126	`1.0.0.0` – `126.255.255.255`	`255.0.0.0 / 8`
B	128-191	`128.0.0.0` – `191.255.255.255`	`255.255.0.0 / 16`
C	192-223	`192.0.0.0` – `223.255.255.255`	`255.255.255.0 / 24`

Through an IP Address and a Mask, we can identify the network address and the broadcast address, and thus define the range of IPs for the network/subnet.

- IP/Mask
- Network
- IP addresses range
- Broadcast

Subnetting

In an organization the ISP provider assigns an IP, the organization can create subnets using bits of the host part, this is called subnetting.

Example organization using the Class C address assigned by the ISP: 202.16.8.0

```
IP Address provided by the ISP: 202.16.8.0
Subnets needed: 2 bits to create 4 subnets
```

```
Default Mask      11111111. 11111111. 11111111. 00000000   /24 255.255.255.0
New Subnet Mask:  11111111. 11111111. 11111111. 11000000   /26 255.255.255.192
```

```
Using the two bits will give 4 (22) subnets, the remaining 6 bits will give 64 (26) host
addresses for each subnet. The address range will be as follows:

Subnet1 202.16.8.0 - 202.16.8.64
Subnet2 202.16.8.65 - 202.16.8.128
Subnet3 202.16.8.129 - 202.16.8.192
Subnet4 202.16.8.193 - 202.16.8.224
```

- IP/Mask
- Network
- IP addresses range

- Broadcast

Public and private IPv4 addresses

There are two types of IP addresses used on a network:
- public
- private

The InterNIC (Network Information Center) is the global body responsible for assigning public addresses. They assign class-based network IPs, which are always unique. The public addresses are available with internet routers so that data can be delivered correctly. Public addresses are now exhausted.

According to RFC 1918, a portion of the IP address space has been designated as "private addresses". This range of addresses does not overlap with the public addresses. The private addresses can be reused. The private address space is not directly reachable through the internet. To access devices utilizing the private address space, a router using NAT (Network Address Translation) will need to be configured.

If each device connected to the Internet in the world had a unique IP address, there would not be enough IPs (v4) for everyone. For this, private IP addresses were defined.

There are three blocks of private addresses:

Class A

10.0.0.0/8

Allows the range of addresses from 10.0.0.1 to 10.255.255.254. The 24 bits from the host ID are available for subnetting.

Class B

172.16.0.0/12

Allows the range of addresses from 172.16.0.1 to 172.31.255.254. The 20 bits from the host ID are available for subnetting.

Class C

192.168.0.0/16

Allows the range of addresses from 192.168.0.1 to 192.168.255.254. The 16 bits from the host ID are available for subnetting.

Class	First Octet	Range	Private IPs
A	1-126	1.0.0.0 – 126.255.255.255	10.0.0.0 – 10.255.255.255
B	128-191	128.0.0.0 – 191.255.255.255	172.16.0.0 – 172.31.255.255
C	192-223	192.0.0.0 – 223.255.255.255	192.168.0.0 – 192.168.255.255

<PENDENT> Loopback
<PENDENT> IP LINK LOCAL

IPv4 and IPv6 addresses

The IPv4 addresses are made up of four 8-bit octets for a total of 32-bits. This means the maximum number of possible addresses is 2^{32}, which is less than 4,294,967,296. Uses broadcasting to send data to all hosts on a subnet.

The IPv6 addresses are based on 128-bits. Using similar calculations, as shown above, the maximum number of possible addresses is 2^{128}. The IPv6 addresses consist of eight 16-bit segments. IPv6 addresses are usually expressed in hexadecimal format. No broadcast addresses, uses multicast scoped addresses as a way to selectively broadcast

- IPv4 32 bits, 2^{16} addresses, representation in 4 octets, example: 192.168.20.8.
- IPV6 128 bits, 2^{128} addresses, representation in 8 hex blocks, example: 4AAE:F200:0342:AA00:0135:4680:7901:ABCD

The differences between the two include many other aspects, including security, package contents, and speed of transport.

- ipv4 200.216.10.15:443
- ipv6 [2001:0db8:85a3:08d3:1319:8a2e:0370:7344]:443
- IPv6 does not implement the broadcast feature exactly as it exists in IPv4. However the same result can be achieved by sending the packet to the address ff02::1, reaching all hosts on the local network. Something similar to using 224.0.0.1 on IPv4 for multicasting as a destination.
- Through the SLAAC (Stateless Address Autoconfiguration) feature, IPv6 hosts are able to self-configure.
- The TTL (Time to Live) field of IPv4 has been replaced by the “Hop Limit” in the IPv6 header.
- All IPv6 interfaces have a local address, called link-local address, prefixed with fe80::/10.
- IPv6 implements the Neighbor Discovery Protocol (NDP), which is similar to the ARP used by IPv4, but with much more functionality.

IPv6 Abbreviations

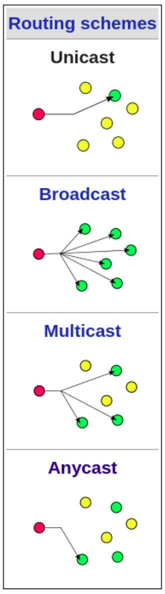
IPv6 defines ways to shorten addresses in some situations.

- The first possibility is to reduce strings from 0000 to just 0
- In addition, in case of group strings with a value of 0, they can be omitted (using ::)
- However, this last abbreviation can only be done once in the address

```
IPv6              2001:0db8:85a3:0000:0000:0000:0000:7344
abbreviation-1    2001:0db8:85a3:0:0:0:0:7344
abbreviation-2    2001:0db8:85a3::7344
```

```
IPv6           2001:0db8:85a3:0000:0000:1319:0000:7344
abbreviation-1 2001:0db8:85a3:0:0:1319:0:7344
abbreviation-2 2001:0db8:85a3::1319:0:7344
```

IPv6 Addresses type



Unicast

Identifies a single network interface. By default, the 64 bits on the left identify the network, and the 64 bits on the right identify the interface.

Multicast

Identifies a set of network interfaces. A packet sent to a multicast address will be sent to all interfaces that belong to that group. Although similar, it should not be confused with broadcast, which does not exist in the IPv6 protocol.

Anycast

This also identifies a set of interfaces on the network, but the packet forwarded to an anycast address will be delivered to only one address in that set, not everyone.

Default route

The function of routing is to send an IP packet, consisting of a header (source and destination address) and encapsulated data, from one point to another.

- routing
- IP packet
- routing table
- default gateway (default route)

All devices have routing tables, which contain routes used to calculate the optimal journey of the messages that it is responsible for forwarding through other routers in the same or other networks.

- When a computer sends packets to another computer, it consults its routing table.
- If a packet is being sent to a destination on the same subnet, no routing is needed, and the packet is sent directly to the computer
- If a packet is being sent to another network then the routing table is consulted and the packed is send to the next hop.
- If a packet is being sent to the internet (or no destination network is found in the routing table) then is send to the default gateway.

The routing table is a list of other routers that are connected to the current router. If the router receives a packet for a network destination that it has in its routing table (typically, this will be another local network), it simply forwards it. Otherwise, the router will send the packet to its default route (typically, this will be the way to get to the internet).

```
[root@localhost cups]# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         _gateway        0.0.0.0         UG    100    0        0 ens3
172.16.5.0      0.0.0.0         255.255.255.0   U     100    0        0 ens3
192.168.124.0   0.0.0.0         255.255.255.0   U     0      0        0 virbr0

[root@localhost cups]# ip route
default via 172.16.5.254 dev ens3 proto dhcp metric 100
172.16.5.0/24 dev ens3 proto kernel scope link src 172.16.5.1 metric 100
192.168.124.0/24 dev virbr0 proto kernel scope link src 192.168.124.1 linkdown
```

Field description:

- The first column contains the Destination network address. The word default signifies the default route.
- The second column contains the defined Gateway for the specified destination. In the event that an asterisk * is shown, it means that a gateway is not needed to access the destination network.
- The Genmask column shows the netmask for the destination network.
- In the Flags column, a U means the route is up and available, whereas the G means that the specified gateway should be used for this route.
- The Metric column defines the distance to the destination. This is typically listed in the number of hops (the number of routers between source and destination).
- The Ref column is not used by the Linux kernel.
- The Use column is used to define the number of lookups for the route.
- The Iface column is used to define the exit interface for this route.

Example description:

- The default route or gateway is set to interface ens3 (via IP address 172.16.5.254 the IP address of the router). This is the way to connect to internet an external routes.

```
default         _gateway        0.0.0.0         UG    100    0        0 ens3
default via 172.16.5.254 dev ens3 proto dhcp metric 100
```

- The network 172.16.5.0/24 is attached to the interface ens2 with IP address 172.16.5.1.

```
172.16.5.0      0.0.0.0         255.255.255.0   U     100    0        0 ens3
172.16.5.0/24 dev ens3 proto kernel scope link src 172.16.5.1 metric 100
```

- The network 192.168.124.0/24 is attached to the interface virbr0 via IP address 192.168.124.1.

```
192.168.124.0   0.0.0.0         255.255.255.0   U     0      0        0 virbr0
192.168.124.0/24 dev virbr0 proto kernel scope link src 192.168.124.1 linkdown
```

The NAT (Network Address Translation) feature allows hosts on an internal network, which uses private IPs, to have access to the Internet as if they were directly connected to it, with the Public IP used on the gateway.

Understanding TCP / UDP / ICMP

The Transmission Control Protocol (TCP) provides connection-oriented service between two applications exchanging data. The protocol guarantees delivery of data.

- connection (3 ways exchange) [establiment de connexió / connexió de 3 vies]
- reliable [confiable]
- sequence number & acknowledgements [números de seqüència i acús de rebut]
- PDU: segment

```
 For example, consider accessing a server via a web browser. The user’s computer will
resolve the IP address for the web server and connect to the web server via the standard
HTTP port 80. After establishing the connection, the client and server processes
exchange information about the socket size used to buffer data and the initial sequence
number of packets.

The sequence number mechanism in the header ensures ordered delivery of data. The web
server will then service GET requests sent on the HTTP port for web pages. For error
control, TCP uses the acknowledgment number in the header. The client sends the
acknowledgment number to the server. If the server sends 2000 bytes of data to the
client and the client acknowledges only 1000 bytes, then it indicates loss of data. The
web server will then retransmit the data.
```

User Datagram Protocol (UDP) provides connectionless service between two applications exchanging data. Unlike TCP, UDP has no error control and does not guarantee the transfer of data. UDP sends data without notifying the receiver prior to sending. As a result, it does not offer either ordered or reliable delivery. UDP is like the traditional postal system; you are not notified that a letter will be delivered to your mailbox.

- no connection established
- no prior notifying
- no reliable
- no sequence number and not Acknowledgement.
- PDU: datagram

The header of UDP packets is lightweight as compared to TCP packets since it does not contain sequence or acknowledgment numbers. It uses a simple, optional checksum mechanism for error-checking. UDP is faster than TCP and is used in services such as VoIP, streaming video (Netflix), and DNS (Domain Name Service).

The Internet Control Message Protocol (ICMP) is a diagnostic protocol used to notify about network problems that are causing delivery failures. This protocol is considered as a part of the IP protocol, though it is processed differently than normal IP packets. Some of the common types of ICMP messages are:

- Destination Unreachable
- Redirect (i.e., use an alternative router instead of this one)
- Time exceeded (i.e., IP TTL exceeded)
- Source Quench (i.e., host or router is congested)
- Echo Reply/Request (i.e., the ping command)

Services: /etc/services

In order to make it easy to distinguish between packets destined for different services, each service is assigned one or more port numbers.

- service
- port number
- well-known ports (0-1023)
- dynamic ports
- /etc/services

The /etc/services file is used for mapping application service names to port numbers. Most services in modern Linux use separate configuration files to specify the ports that they communicate through. However, the /etc/services file is useful as most default service configuration files will initially have the same port numbers as found by the /etc/services file.

```
echo            7/tcp
echo            7/udp
discard         9/tcp           sink null
discard         9/udp           sink null
systat          11/tcp          users
systat          11/udp          users
daytime         13/tcp
daytime         13/udp
qotd            17/tcp          quote
qotd            17/udp          quote
chargen         19/tcp          ttytst source
chargen         19/udp          ttytst source
ftp-data        20/tcp
ftp-data        20/udp
# 21 is registered to ftp, but also used by fsp
ftp             21/tcp
ftp             21/udp          fsp fspd
ssh             22/tcp                          # The Secure Shell (SSH) Protocol
ssh             22/udp                          # The Secure Shell (SSH) Protocol
```

```
telnet          23/tcp
telnet          23/udp
# 24 - private mail system
lmtp            24/tcp                          # LMTP Mail Delivery
lmtp            24/udp                          # LMTP Mail Delivery
smtp            25/tcp          mail
smtp            25/udp          mail
```

LPI common ports to know:

```
20/tcp   ftp-data
21/tcp   ftp
21/udp   ftp
22/tcp   ssh
22/udp   ssh
23/tcp   telnet
25/tcp   smtp
53/tcp   domain (dns)
53/udp   domain (dns)
80/tcp   http
80/udp   http
110/tcp pop3
110/udp pop3
119/tcp nntp
139/tcp netbios-ssn
139/udp netbios-ssn
143/tcp imap2
143/udp imap2
161/tcp snmp
161/udp snmp
443/tcp https
443/udp https
465/tcp ssmtp
993/tcp imaps
993/udp imaps
995/tcp pop3s
995/udp pop3s
```

Port	Service
20	FTP (data)
21	FTP (control)
22	SSH (Secure Socket Shell)
23	Telnet (Remote connection without encryption)
25	SMTP (Simple Mail Transfer Protocol), Sending Mails
53	DNS (Domain Name System)
80	HTTP (Hypertext Transfer Protocol)
110	POP3 (Post Office Protocol), Receiving Mails
123	NTP (Network Time Protocol)
139	Netbios
143	IMAP (Internet Message Access Protocol), Accessing Mails
161	SNMP (Simple Network Management Protocol)
162	SNMPTRAP, SNMP Notifications
389	LDAP (Lightweight Directory Access Protocol)
443	HTTPS (Secure HTTP)
465	SMTPS (Secure SMTP)
514	RSH (Remote Shell)
636	LDAPS (Secure LDAP)
993	IMAPS (Secure IMAP)
995	POP3S (Secure POP3)

Commands & services

Service ftp

FTP is a protocol that uses TCP for transport and reliable delivery. The ftp command provides the user interface to the standard File Transfer Protocol (FTP). Using the ftp utility, a user can transfer files to and from remote machines.

- $ ftp ftp_server_host_name [or IP address]
- ftp> ls
- ftp> get file
- ftp> mget wildcard-file
- ftp> put file
- ftp> pwd
- ftp> lcd
- ftp> ascii / bin mode transfer
- ftp> quit
- Ports: 20, 21

The default file transfer mode for the ftp utility is ASCII, which is used for ordinary text files. To transfer other types of files (i.e. program files, zip files, or tar files, etc.), it is recommended that the server is in binary transfer mode.

Service Telnet

The Telnet Protocol is used for interactive communication with host machines using TCP/IP. The client's machine becomes a virtual terminal for the remote host. The telnet command provides the user interface to the standard Telnet protocol.

- telnet host_name [or IP address]
- exit
- logout ctrl+]
- Port: 23

The telnet protocol suffers the same defect as ftp; the packets that are sent are not encrypted. This means that your user name, your password, and any data sent during a telnet session can easily be captured, so your credentials may become compromised. ssh is a safer alternative for telnet.

Querying DNS servers

The host and dig commands are used for DNS (Domain Name System) lookups, as is the nslookup command (deprecated but…).

The host command is used to resolve hostnames to IP addresses and IP addresses to hostnames. The utility uses UDP for transport of queries to the servers listed in the /etc/resolv.conf file.

```
$ host www.pue.es
www.pue.es is an alias for pue-app-srv.pue.es.
pue-app-srv.pue.es has address 176.34.150.171

$ host pue.es
pue.es has address 176.34.150.171
pue.es mail is handled by 10 aspmx2.googlemail.com.
pue.es mail is handled by 10 aspmx3.googlemail.com.
pue.es mail is handled by 5 alt1.aspmx.l.google.com.
pue.es mail is handled by 5 alt2.aspmx.l.google.com.
pue.es mail is handled by 1 aspmx.l.google.com.

$ host www.escoladeltreball.org
www.escoladeltreball.org is an alias for fo1.escoladeltreball.org.
fo1.escoladeltreball.org is an alias for fibra1-tel.dynalias.org.
fibra1-tel.dynalias.org has address 81.40.3.148

$ host 176.34.150.171
171.150.34.176.in-addr.arpa domain name pointer
ec2-176-34-150-171.eu-west-1.compute.amazonaws.com.
```

```
$ host -t MX gencat.cat
gencat.cat mail is handled by 5 mx1.hc489-80.eu.iphmx.com.
gencat.cat mail is handled by 20 smtp2.gencat.cat.
gencat.cat mail is handled by 10 smtp1.gencat.cat.
gencat.cat mail is handled by 5 mx2.hc489-80.eu.iphmx.com.
gencat.cat mail is handled by 30 smtp3.gencat.cat.

$ host -t NS .
. name server f.root-servers.net.
. name server b.root-servers.net.
. name server e.root-servers.net.
. name server c.root-servers.net.
. name server d.root-servers.net.
. name server i.root-servers.net.
. name server j.root-servers.net.
. name server g.root-servers.net.
. name server a.root-servers.net.
. name server h.root-servers.net.
. name server m.root-servers.net.
. name server k.root-servers.net.
. name server l.root-servers.net.
```

The dig (Domain Information Groper) command is used for troubleshooting the configuration of DNS servers. DNS server administrators like the output of the dig command because it is in the same format that the information is entered into a DNS server configuration file.

```
$ dig pue.es

; <<>> DiG 9.11.28-RedHat-9.11.28-1.fc32 <<>> pue.es
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 7626
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 4, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1460
;; QUESTION SECTION:
;pue.es.                        IN      A

;; ANSWER SECTION:
pue.es.             3600    IN      A       176.34.150.171

;; AUTHORITY SECTION:
pue.es.             172800  IN      NS      ns-224.awsdns-28.com.
pue.es.             172800  IN      NS      ns-769.awsdns-32.net.
pue.es.             172800  IN      NS      ns-1113.awsdns-11.org.
pue.es.             172800  IN      NS      ns-1988.awsdns-56.co.uk.

;; Query time: 274 msec
;; SERVER: 80.58.61.250#53(80.58.61.250)
;; WHEN: Wed Nov 10 21:05:27 CET 2021
;; MSG SIZE  rcvd: 191
```

```
$ dig -t NS .

; <<>> DiG 9.11.28-RedHat-9.11.28-1.fc32 <<>> -t NS .
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60595
;; flags: qr rd ra; QUERY: 1, ANSWER: 13, AUTHORITY: 0, ADDITIONAL: 27

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1460
;; QUESTION SECTION:
;.                             IN      NS
```

```
;; ANSWER SECTION:
.                     479194  IN      NS      j.root-servers.net.
.                     479194  IN      NS      e.root-servers.net.
.                     479194  IN      NS      d.root-servers.net.
.                     479194  IN      NS      c.root-servers.net.
.                     479194  IN      NS      m.root-servers.net.
.                     479194  IN      NS      f.root-servers.net.
.                     479194  IN      NS      k.root-servers.net.
.                     479194  IN      NS      a.root-servers.net.
.                     479194  IN      NS      l.root-servers.net.
.                     479194  IN      NS      h.root-servers.net.
.                     479194  IN      NS      g.root-servers.net.
.                     479194  IN      NS      b.root-servers.net.
.                     479194  IN      NS      i.root-servers.net.

;; ADDITIONAL SECTION:
a.root-servers.net.   3560794 IN      A       198.41.0.4
b.root-servers.net.   3560794 IN      A       199.9.14.201
c.root-servers.net.   3560794 IN      A       192.33.4.12
d.root-servers.net.   3560794 IN      A       199.7.91.13
e.root-servers.net.   3560794 IN      A       192.203.230.10
f.root-servers.net.   3560794 IN      A       192.5.5.241
g.root-servers.net.   3560794 IN      A       192.112.36.4
h.root-servers.net.   3560794 IN      A       198.97.190.53
i.root-servers.net.   3560794 IN      A       192.36.148.17
j.root-servers.net.   3560794 IN      A       192.58.128.30
k.root-servers.net.   3560794 IN      A       193.0.14.129
l.root-servers.net.   3560794 IN      A       199.7.83.42
m.root-servers.net.   3560794 IN      A       202.12.27.33
a.root-servers.net.   3560794 IN      AAAA    2001:503:ba3e::2:30
b.root-servers.net.   3560794 IN      AAAA    2001:500:200::b
c.root-servers.net.   3560794 IN      AAAA    2001:500:2::c
d.root-servers.net.   3560794 IN      AAAA    2001:500:2d::d
e.root-servers.net.   3560794 IN      AAAA    2001:500:a8::e
f.root-servers.net.   3560794 IN      AAAA    2001:500:2f::f
g.root-servers.net.   3560794 IN      AAAA    2001:500:12::d0d
h.root-servers.net.   3560794 IN      AAAA    2001:500:1::53
i.root-servers.net.   3560794 IN      AAAA    2001:7fe::53
j.root-servers.net.   3560794 IN      AAAA    2001:503:c27::2:30
k.root-servers.net.   3560794 IN      AAAA    2001:7fd::1
l.root-servers.net.   3560794 IN      AAAA    2001:500:9f::42
m.root-servers.net.   3560794 IN      AAAA    2001:dc3::35

;; Query time: 7 msec
;; SERVER: 80.58.61.250#53(80.58.61.250)
;; WHEN: Wed Nov 10 21:04:35 CET 2021
;; MSG SIZE  rcvd: 811
```

```
$ dig +trace lms.pue.es
```

Example Exercises

1. Using the IP 172.16.30.230 and netmask 255.255.255.224, identify:
    a. The CIDR notation for the netmask
    b. Network address
    c. Broadcast address
    d. Number of IPs that can be used for hosts in this subnet
    e. Is it a network, host or broadcast address?
2. Using the IP 192.168.30.1/20, identify:

    a. The netmask
    b. Network address
    c. Broadcast address
    d. Number of IPs that can be used for hosts in this subnet
    e. Is it a network, host or broadcast address?
3. The CIDR notation for the netmask 255.255.255.252
4. Which class are the addresses? Public or private?
    a. 172.32.16.2
    b. 10.1.1.1
    c. 192.167.12.34
5. Which setting is required on a host to allow an IP communication with a host in a different logical network?
6. Which port is the default for the SMTP protocol?
7. How many different ports are available in a system?
8. Which transport protocol ensures that all packets are delivered properly, verifying the integrity and the order of the packets?
9. Which type of IPv6 address is used to sent a packet to all interfaces that belong to group of hosts?

10. Realitza els exercicis indicats a: 109.1 Fundamentals of internet protocols
11. Realitza els exercicis del Question-Topics 109.1