NI Q&A

Describe the two type of cross-talk noise in ADSL and how to solve them

In the ADSL architecture, the twisted pairs of the users are merged together in a Binder Group starting from the Cabinet to the C.O. In the Binder Group, different cables transmit in the same frequencies at the same time, this results in an interference called cross-talk. Cross-talk is divided in two categories: FEXT (Far End Cross Talk) and NEXT (Near End Cross Talk). FEXT arises when we have two receivers near to each other. Both of them receive the transmission of their respective sender plus the transmission of the other sender, because the latter crosses the cable and interferes with the other cables. The transmission that generates the interference is already altered when crossing the cable, and, in addition, there is the distance factor that continuously attenuates the signal. For these reasons, FEXT it is not considered an issue. On the other hand, NEXT happens when we have a transmitter and a receiver near to each other. The receiver receives both the signal directed to it (transmitted by its sender) and the signal generated by the transmitter near to it. This type of interference is worse than FEXT because the good signal is attenuated due to the distance, while the signal that interferes is strong because of the short distance. In the ADSL the NEXT happens when a user is in Downstream and another user is in Upstream. Meaning that from the point of view of the C.O. we have a transmitter and a receiver close to each other. So in this case the NEXT affects the Upstream receiver, that is a modem (ATU-C) in the DSLAM (the transmitter is an ATU-C modem too). There are a few ways to solve the NEXT problem. The first way is the so called *Echo Cancellation*. In order to perform it, we need the Upstream receiver to know what the Downstream transmitter actually transmits, and in addition both of them must lay in the same device. In this way the receiver can cancel the transmission that interferes. The second way is to reduce the number of cables in the Binder Group, reducing the probability of having Upstream and Downstream of two different users at the same time. In the end, the last method is to avoid frequency overlapping using two distinct range of frequencies for the Upstream and the Downstream.

Talk about solutions to use Optical Fiber in the Access Network

A variety of different solutions can be used to implement the optical fiber in the access network. These solutions are the result of trade-offs between cost of installation of the fiber (and consequent removal of copper wires) and economic profit thanks to the augmenting of the throughput. The name of the various solutions is determined by the position of the EOI, which stands for Electro Optical Interface. The EOI has the same job of the DSLAM for the ADSL architecture. If the EOI is placed directly in the C.O. we talk about Fiber to the Exchange, because the fiber coming from the core network stops at the Exchange (the C.O. itself) and from it to the user only copper wire is used. If the EOI is placed in the cabinet or in the curb, the name of the architecture is respectively Fiber to the Cabinet and Fiber to the curb. Meaning that the fiber ends at these components. In the end, the last solutions are the Fiber to the Building and the Fiber to the Home, which belong to the Fiber to the Premises group. In the first one we have the EOI placed inside the building and directly reached by the fiber, while copper wire is used only to connect every host of the building to the EOI. In the second case instead we do not have the needing of an EOI becuase every host is directly connected by means of the optical fiber. Now that we have explained the different solutions to use the optical fiber, let's focus of the FTTH one. A first way to implement FTTH is to assign a single fiber to every building.

However, this results is a too costly solution because every time we need to add an user, we have to pay for the cost of the installation of another fiber cable to the C.O. (the same if i have to remove an user). The second way is to start from the C.O. and arriving at the curb with a single fiber, and then dividing it between the different users. This is a useful solution because the curb can regenerate the signal, but i need to provide power to it in order to let it operate on the optical signals. The last solution is the one of using a passive optical splitter. It is the cheapest solution and it is basically a bunch of fiber grouped together. Being a passive device means that i do not have to provide power to it. It does not regenerate the signal but only divides it.

Differences in topology and technology of Access, Core and End network

Advantages and disadvantages of Direct and Indirect routing

In the mobility implementation, the Mobile IP standard communication protocol is used. Each user is associated to two different IP addresses. The first is called Permanent IP Address and it is the static identity of the user. The second one is called Care-of IP Address (also know as Temporary IP Address) and it is the one that is given to the user by the network where he is currently in. There are also two kind of network elements: the Home Agent and the Foreign Agent. The Home Agent it's the MSC (Mobile Switching Center) where the user tipically stays, which holds the user's Permanent IP in the HLR (Home Location Register). The Foreign Agent is the MSC of another network where the user currently stays. When an user enters another network that is different from the one he belongs, the Foreign Agent of that network assigns a Care-of IP Address to that user, storing it into the VLR (Visitor Location Register). Now that we have defined how the Mobile IP works, let's explain the way in which Indirect and Direct routing are implemented. Suppose there is a Correspondent (which is the entity wishing to communicate with the mobile node) that wants to communicate with a user that it is not in his Home Network. The packet in anyway routed to the Home Network of the user using the permanent address. The Home Agent knows that the user is in another network and then forwards the packet to the Visited Network where he currently is. Now, if the user wants to send a packet to the Correspondent, he can do it directly, forming the so called *Triangle Routing*. The disadvantage of this technique is that even if the correspondent is in the same Visited Network of the user, still the packed will be forwarded to the Home Agent of the user, creating the triangle anyway and resulting in a loss of efficiency. The advantage is that the Correspondent just need to know the Permanent IP Address of the user in order to communicate with him, without the needing of further updates, this approach is called *Indirect Routing*. The second approach, called *Direct Routing*, works as follows: the Correspondent wants to communicate to a user that, again, is not in his Home Network. The correspondent sends a packet to the Home Network asking where the user is, and the Home Agent answers with the user's Care-of Address (obtained by the Foreign Agent), so this time is the Correspondent itself that communicates directly with the user in the Visited Network. This technique removes the Triangle Routing but introduces another problem that arises when the user moves again. Indeed, in this situation, the Correspondent doesn't know how to reach the user anymore, so goes again to the Home Agent to ask where the user is. While in the Indirect Routing all these things were transparent by the point of view of the Correspondent. What happens now is that the old Foreign Agent becomes the so called Anchor Agent, working as a "hook" to reach the new Visited Network where the user currently is. This builds up a chain, meaning that the packets are always router to the first Anchor Agent, and then the latter follows the chain in order to reach the user.

Noise in digital transmission and resolution

Digital signal has a certain number of levels, and when we transmit it, if the channel has a low bandwidth it will cut many frequencies, resulting in a distortion of the signal (making it difficult for the receiver to reconstruct it). It is not only the bandwidth of the channel that affects the shape of the signal, but also the noise of the channel. But the datarate is not directly limited by the channel

bandwidth and the high noise of the channel, meaning that we can decide to send my signal at an high datarate whatever is the bandwidth or the noise, but what happens is that then my signal gets distorced depending on the two factors just described. Therefore, before transmitting, we need to decide the shape of the signal in a suitable way for the channel. Using Nyquist Formula, we can understand which is the maximum capacity of the channel:

$$C = 2Blog_2M$$

Where C is the capacity of the channel, B is the bandwidth and M are the levels of the digital signal to be transmitted. Now, looking at the formula, we could say that even if B is low, we can set M very high in order to achieve an high datarate, but this is not true because the higher the levels, the harder is that the receiver is able to distinguish them. So, in order to find M, we rely on the Shannon's formula:

$$C = 2Blog_2(1 + SNR)$$

Where SNR is the *Signal to Noise Ratio*, which is a characteristic of the channel expressed in dB. Knowing the SNR, we can calculate C and then, substituting it in the Nyquist formula, calculate M in order to find out the maximum number of levels that we can use for the signal. Notice that the higher the SNR (so the lower is the noise), the higher are the levels.

DMT Modulation

DMT Modulation (Discrete Multi-Tone) is the most used kind of ADSL modulation. We consider the bandwidth divided in different subchannels called tones. These tones are separated each one by a gap of 4.3 KHz. DMT is able to allocate data so that the throughput of every single subchannel is maximized. If some subchannel can not carry any data, it can be turned off and the use of available bandwidth is optimized. The idea is to simulate an ideal channel, which is a set of frequencies that are attenuated in the same manner all at the same time. However, in reality the channels do not behave like that, but if we consider a bunch of subchannels, they are approximatively attenuated in the same manner. So the goal is to have a lot of small ideal channels. The standard states that the bandwidth is divided into 256 subchannels of 4,3125 KHz each, resulting in a whole bandwidth of 1104 KHz. The behavior of each channel is different, because on each subtone we can have more or less datarate depending of the SNR in that specific subtone. Within each channel, modulation uses Quadrature Amplitude Modulation (QAM).

How the frequency band is used in the ADSL and how this is reflected in the ADSL architecture

The idea of ADSL, as well as of all the other DSL technologies, basically arises from the awareness that the available bandwidth of the twisted pair is much greater than the one used for the usual voice traffic and therefore potentially exploitable for a transmission efficient broadband data. What was done was to have, thanks to the Frequency Multiplexing, three separate channels in the same twisted pair: voice, upstream and downstream. This new standard with three kinds of channels was delivering to the users 8 Mbit/s over 2 kilometers of Unshielded Twisted Pair (UTP). The ADSL did not change the ISDN architecture, but only the way in which the twisted pair was used. The bandwidth dedicated to the telephone was from 0 to 4KHz (actually it reached 20KHz due to the roll-off). The bandwidth of the upstream started from 25 KHz and ended to 200 KHz. In the end, the one given to the downstream was from 250 KHz to 1000 KHz. There is the possibility of augmenting the downstream bandwidth, overlapping it on the upstream's one. The resulting architecture was the following. A modem called ATU-R (ADSL Termination Unit - Remote) modulates the signal coming from the host in order to let it match the upstream or downstream bandwidth. A device called splitter joins the two cables coming from the POTS-R (Plain Old Telephone Service - Remote) and from the ATU-R into a single cable, and then works as a low-pass filter for the voice frequencies and as a high-pass filter for the upstream and downstream frequencies. In the C.O., a splitter too was added in order to divide the

different signals and redirect them into the PSTN (Public Switched Telephone Network) or in the WAN. The frequencies that come out from the high-pass filter go to the DSLAM (Digital Subscriber Line Access Multiplexer) before going outside the C.O., which is a cluster of ATU-C (Central), one for each ATU-R of the connected users.

Differences between Pre-ADSL and ADSL + Difference Between voice and digital modems + Which are the key differences of the old use of the copper wire to provide data (analog voice and modem) and the digital one

The ISDN was the first attempt to use the telephone line to transmit also internet data, this was called Integrated Service (IS), for which the internet data was modulated onto an analog signal that could be transmitted on the telephone line in the 0-4KHz bandwidth. Since the data was transmitted in the same bandwidth of the voice, only voice or data could be transmitted at a given time. In the Pre-ISDN architecture, every user had its own twisted pair, starting from him and going towards the C.O. Then the twisted pairs of different users were physically put together by the Cabinet in a bigger cable, called Binder Group. In the ISDN architecture, there is a modem on the user side that has the job to transmit the digital signal of the host by means of an analog signal. It modulates an analog signal with a specific shape (the digital one, with levels etc) able to transport in the correct way the digits. So the modem actually generates an analog signal that represents the bits the user wants to transmit. With the advent of the ADSL, three different channels in the same twisted pair were created: one for the telephone, one for the upstream and one for the downstream. The modem of the ADSL, called ATU-R, has to performs a modulation of the signal coming from the host in order to make the signal matching with the ADSL standard according to the range of frequencies (upstream, downstream and voice). There is also another device called splitter, that is a low-pass filter for the telephone bandwidth and a high-pass filter for the data bandwidth. We have a splitter on the C.O. side too. In the end, once the data signal passed through the splitter, it encounters a device called DSLAM, that has the job of multiplexing the signals coming from the users. In the DSLAM we have a number of modems, called ATU-C, equal to the number of users connected to that C.O.

VDSL and Vectoring

VDSL is an xDSL technology, where the V stands for Very High data rate DSL. VDSL's goal was to provide data transmission faster than the ADSL. A VDSL connection uses up to seven frequency bands, so one can allocate the data rate between upstream and downstream differently depending on the service offering and spectrum regulations. VDSL adopts two types of bandwidth division, the frequency division, used for symmetric systems and the Time Division. In the Time Division, downstream and upstream channels occupy the same bandwidth. VDSL Vectoring is a technique used to reduce the crosstalk between the receiver and the transmitter to improve the performance. The crosstalk is canceled by the injection of an anti-signal on the line. This solution requires the full synchronization over the full vectored system, all data samples are shared between all the lines, requires the computation of the anti-signal and a crosstalk estimation mechanism to derive the crosstalk coefficient. The vectoring mechanism is specified in ITU-T G.993.5 (G.vector) standard for DSLAM/CPE interoperability. This technique allows reaching a high data rate even to the users that are distant from the cabinet.

Mobility

We can distinguish wireless technologies in two different aspects: bitrate and mobility. Mobility determines how the network is build in terms of network components of a Cellular Network. The challenge we have to face when talking about Wireless communication is that we want to be able to move during the communication. Wireless communication can be divided into WiFi, that implements the Wireless LAN, and the Cellular Communication. The WiFi allows very limited mobility for the user, indeed, once the user goes out the range of the access point, it drops the signal. So Wireless does

not always mean Mobility. The Cellular Communication instead uses a base station as an access point and guarantees that, if we step out the range of the base station, we keep the connection. The general Wireless infrastructure is composed of hosts that want to connect to the different access points (base stations or just modems in the case of WiFi), where each access point is able to cover a certain area called cell. The base stations can be connected with each other using a wired or wireless connection. All these elements belong to the Access Network part, the Core Network (backbone) remains the same of the Telephone infrastructure. The problem that the base station need to manage is the so called Hand-Off, which consists into letting the connection of an user remain alive even if he goes out of his current antenna's range. There are also the Ad Hoc Networks, where there are no access points but only users that establish a connection between them, so the only element of the networks are the hosts themselves. Talking again about the Mobility, the latter is handled by a device called MSC (Mobile Switching Center), which is a network element that finds out where the user is, follows his behavior and sees which base station is actually serving him, in order to understand which is the next base station to be assigned to him. This works at a physical level using the signal strength. Moreover, the system implements the dynamic streaming quality, so if the datarate of the user is low, the http requests are done for videos with lower quality. (Mobile IP, Indirect and Direct routing explained in another answer).

MIMO

MIMO stands for Multiple Input Multiple Outputs and refers to the multiple antennas on both the receiver and the sender side. Since LTE requires a 100Mbps peak downlink MIMO is needed to achieve this result. A classical configuration in LTE is a solution that sees 2 antennas on both sides but supports even a 4x4 antennas configuration. The transmission operative modes of multiple antennas are: Spatial multiplexing, beamforming, single-stream transmit diversity. The Spatial multiplexing operating mode sends different data streams on different antennas simultaneously, achieve a higher data rate but has some limitation due to path correlation. Transmit Diversity Mode is used when the goal is to combat fading since a replica of the same signal is sent on several antennas, this model gets a better SNR at the receiver. Beamforming uses multiple antennas to control the direction of a wave-front. These antennas support both a Single User MIMO and Multiple User MIMO. In SU-MIMO the purpose is to increase the user Data rate, the functioning is the simultaneous transmission of different data streams to one user, it's efficient when the user has good channel conditions. For MU-MIMO the intent is to increase the sector capacity by selecting those users that are experiencing god channel conditions, it's efficient when a large number of users have an active data transmission at the same time. 5G maintains the MIMO transmission but improves it. 5G supports up to 8x8 antennas (but at the moment few devices have 8 antennas for the massive MIMO), called Massive MIMO. Massive MIMO also improves the Beamforming, improves the network capacity, the coverage and the user experience thanks to a better beamforming.

LTE vs LTE-A

LTE System is the fourth generation of cellular technology, developed to support the high demand for data and resolve some issues linked to mobility. Key technologies in LTE are Adaptive modulation and coding, hybrid ARQ, Spectrum flexibility and MIMO transmission. The Adaptive modulation is a system that uses the channel quality indication so the transmitter can adaptively determine the modulation and coding schemes, in LTE different users would be getting a different rate. The HARQ takes care of error packets, using a combination of Forward error correction and Automatic Repeat request, if the received data has an error then the receiver buffers the data and request a retransmission from the sender. This system works at the physical layer. The spectrum flexibility is used to support a subset of 6 different system bandwidths, in LTE the uplink uses the SC-FDMA, the downlink uses the OFDMA. The OFDMA allocates users in the time and frequency domain. In SC-FDMA modulated data symbols are pre-coded to be transformed into the frequency domain, then sub-carrier mapping allocates the signal to available subcarrier and at the end each subcarrier carries a portion of DFT

spread data symbols. MIMO transmission is a multiple input multiple output scheme of antennas that increase the downlink data rate in LTE. LTE configuration of MIMO has up to 4x4 antennas. LTE-A is an advanced evolution of the LTE, the data rate is higher and supports more technologies than LTE, it's a sort of prelude of the 5G. LTE-A devices can aggregate up to 5 Component carriers, each up to 20MHz. It can be divided into 3 schemas: single spectrum band, Intraband non-contiguous, interband non-contiguous. The benefit of the carrier aggregation is the load balancing and energy savings. MIMO in LTE-A supports a configuration of 8x8 antennas, with advanced beamforming and scheduling techniques. The base station has now a large antenna array that allows the system to serve a large number of users and with a large number of antennas, there is no saturation. In LTE-A there is the introduction of intermediate relay nodes to forward the traffic to areas with no coverage. Machine to machine communication at first and then Device to device communication was also developed in LTE-A, exploiting ad hoc peer to peer communication between devices.

5G Requirements

Main requirements for 5G are:

- A bit rate up to 10Gbps
- 100 % coverage
- Up to 99% perceived availability
- Enormous number of connected devices for the massive IoT
- Higher battery Life
- Low latency
- Reduction in battery usage
- High bandwidth

The usage of massive MIMO allows to send and receive more data for the same user, this increase the bit rate even when a large number of users connect to the same network. Design of smart antenna is important for effective mm-wave communications, beams must be directional, Base stations must have a large number of antennas and the grid of antennas is capable of directing horizontal and vertical beams.

NETKIT

Describe public and private key in SSH authentication with an example

We're talking about the asymmetric cryptography. In this kind of cryptography, we have two different keys for each entity: the public one is used to encrypt, instead the private one to decrypt. In the SSH authentication, the public key represents the identity of a user and the private key the proof that the user is claiming the truth. Every time a client must set an authentication through asymmetric cryptography, it must generate a pair of keys, the public key and the associated private one. It can do this using a built-in tool in the SSH framework called ssh-keygen. The next step is to send the client public key to the server in order to save it in the folder of the authorized keys. In this way the server will keep track of all the trusted clients. Now every time the client tries to connect to the server via ssh, it will be asked to solve a challenge. The server will encrypt a challenge (secret) with the client public key. Then the secret message will be sent to the client through the secure ssh pipe. The client will decrypt the message with its private key obtaining so the initial secrete challenge. Then the client will encrypt the challenge with the server public key and will send the result to the server. Now the server can check the client identity decrypting the received message with its own private key. If and only if the initial challenge and the decrypted one matched, the client would be authorized to access the server.

What is Netfilter and how it is used to implement firewalls

Netfilter is a framework that let us to manage the Linux kernel hooks in order to intercept and manipulate the network packets. A hook is a well-defined point through which a network packet flows. There are 5 built-in hooks in the Netfilter framework: PRE-ROUTING, POST-ROUTING, FORWARD, LOCAL-INPUT, LOCAL-OUTPUT. Each network packet will flow through a route composed by a subset of those hooks. The first two are triggered before and after a routing decision. The FORWARD hook is triggered if the network packet has just to be forwarded to a next hop. The last two are triggered if a packet is processed locally. Every time a hook is triggered by a packet, this one is processed looking the rules contained in the associated tables. There is a built-in table for each activity of the packet: FILTER, NAT, MANGLE, RAW. It is associated a matching field to each rule, meaning that we want to select only a specific type of network packet. A matching can happen checking the protocol (TCP/UDP), a source/destination address, the source/destination port, the in/out interface of NIC and so on so forth. We can also use another main parameter of network packet, in order to distinguish among them: every packet has got a STATE field kept in the header. A STATE can be equal to NEW, ESTABLISHED, RELATED, INVALID, it depends to the state of the connection. To each rule it is also associated an action. The most common actions are ACCEPT, DROP, MASQUERADE, DNAT, SNAT, LOG. For example, if we want to avoid connections started from other devices, we can add a rule for the FORWARD hook in the FILTER table matching all the NEW connections without our IP address set as the source address.

Usage of Link State Packets in OSPF and cost estimation

The OSPF protocol is a Dynamic Routing one. It is one of the fastest and the most spread one. It belongs to the Link State Protocols family, meaning that each node of the network must know the entire network topology. The protocol uses several control packets in order to build the routing table for each node. All fundamental information about the topology is carried by a packet called LSA (Link State Advertisement) or "Hello packet", it is the primary kind of communication between network devices. In fact, the first operation of each node is to send an LSA packet to all its neighbours. Then, again, each node will forward all the received LSAs to all its neighbours. At the end of this first phase every device can create its own database. When a node wants to check the correctness of its database, it sends an LSDB (Link State Database) packet, that is a control one. If there is something missing, a device can send an LSR packet (Link State Request) in order to ask the missing tuple in the database. The neighbours will ask sending LSUs (Link State Update) packets containing the most recent information about the request. To close the connection, it will be sent an LSAck (Link State Acknowledgment) packet. The "Hello" packets are sending periodically, in order to enforce reliability in case a link fails or if a device is added to the topology. A cost is associated to each link. It represents the physical cost for a packet to route through that link. Since the protocol, after the data have been exchanged, compute the Shortest Path for each destination, the Dijkstra algorithm will use those costs in order to achieve the minimum spanning tree of the network.

Remote and Local Port Forwarding

The SSH Port Forwarding is a mechanism in SSH for tunnelling application ports from the client machine to the server machine or vice versa. There are two ways: the local and the remote port forwarding. One is the contrary of the other. The local port forwarding let the client machine redirect all the incoming traffic on a specific port to a configurated destination port through an SSH server. For example, this method is used when a client wants to connect to a service on an internal network from the outside. In this case the client couldn't be able to access the service because it would be blocked by the firewall of the network. So, it can establish an SSH Tunnel with the internal server and redirect all the service traffic to it. The SSH internal server will redirect the received packets to the internal service. In this case the SSH Server is called "jump" server. The second case, as I said before, is a contrary situation. The remote word means that the forwarded traffic is the server one. Let's think about this situation: a web application runs on our client machine, but our firewall

doesn't accept traffic from the outside. Now, we can set an SSH Tunnel with a remote server in order to forward all its incoming traffic on a specific port to a local chosen port (maybe the one which the web application is hosted on). In this way the clients can connect to the accessible outside server and this one will forward the requests to us.

Describe with an example how Tracert (Trace Route) can discover the path taken from a packet toward a specific destination

The Traceroute command can discover not only the path that a packet travels from a source to a custom destination, but also the names, identities and latencies of the touched devices. The command can know all these kinds of information thanks to the TTL field kept in the header of an IP packet. TTL means Time To Live. The Living time of a packet isn't measured in seconds, but in hops. A hop is just a step in the path of a packet. When we set the TTL field we are deciding how many devices at most the packet can touch. Every time a device receives a packet, it reduces the TTL number by one. When the TTL field reaches 0, the last device sends a response error packet (ICMP TTL exceeded) to the original sender. The packet that the original sender received contains the required information about the last reached router. So, the trick is to send a lot of packets with the TTL field increased by one from the previous. In this way the sender can collect the names of the touched devices one by one. The process stops when the destination is really reached. In this case the original sender will receive a different response error packet. In fact, since the Traceroute command uses a common unused UDP Port, when the packet arrives to its destination, the final router will response with an ICMP Destination/PORT Unreachable packet.

Advantages and disadvantages of IP Static Routing vs Dynamic Routing

Static Routing needs a network administrator in order to be set. The main advantage of this kind of configuration is that it is very easy to implement in a small network. Moreover, it is very secure because, since the routes are set a priori, no packets or routing algorithm are needed to build the routing tables of each device. It is very predictable because, given a source and a destination device and since the routing tables are built statically by the administrator, a packet will follow always the same path. But this solution is recommended only for small network topologies. A single person can't manage a huge number of devices/links. The static routing isn't reliable too. In case a link failed, it would be necessary the intervention of a network administrator in order to re-route traffic. The Dynamic Routing solves these problems using an algorithm which can fit all the network topologies used around the world. It is independent of the network size and it keeps in count the possibility that a link can fails. The algorithm spread from each device to all the network control packets in order to be reliable with respect to the changes or the fails of the network itself. But the settings of a dynamic algorithm require a more complex effort initially. It is less secure than the static one because packets flows through network periodically to support updates of the routing table. Moreover, the devices must have a computational power and a private physical memory in order to run the dynamic algorithm, this makes routers more sophisticated and expensive.

Which are the fields of an IP Routing Table and how these are used to route a packet toward a destination (make an example)

When a packet must be delivered, a delivery device can be in one of these two situations: in the best case it is linked directly to the destination device, in the other case it must decide to which device it has to forward the packet in order to let it reach its final destination. These situations are solved looking the IP Routing Table in which there are stored all the possible routes that a packet can follow starting from a specific node. Each row of a device's table contains information about a network that is reachable from itself. Specifically, are stored the network ID, its netmask, the next-hop or gateway, the interface to which send the packet and the cost of this passage. It can be noticed that a network is described as the union of the network ID and the netmask, because to store all the possible IPs directly

would be too expensive. In order to choose the next hop for a packet, it is needed to recognize its associated network. The rule to follow is the "Longest Prefix Matching": for each entry of the table, we have to convert the IP address and the netmask in the bit form, make the logical AND between these two and choose the network that best matches the previous result. Once we found the network, we can forward the packet through the suggested next hop. Example: our IP is 192.168.10.5, there are two entries in the table 192.168.10.0/24 and 192.168.10.5/30. In this case both of these entries match with our IP AND NETMASK, but the second entry wins, because it has a match of 30 digits!

VPN

A Virtual Private Network allow you to connect to a private network remotely using the world IP public network. It was created to let people access private services of a private network (such as a company's one) comfortably and securely from its own device connected to the public internet. This is reachable through encryption, authentication, packet tunnelling and firewalls. Every intranet has got its own firewall that allows only the internal traffic. This is the first security measure applied to make a network private. In order to use the internal services of an intranet we have to make requests from the inside. If we are outside, we can do this tunnelling the packets. It means that the private packet is incapsulated in a public packet and sent to the firewall router of the intranet through the public network. When the public packet is received, the router will forward the private packet to its internal destination. But there is still a problem: if we simply tunnel the private packet, all the devices the public packet flows through could read the content that we are hiding. In order to reach confidentiality, we have to encrypt it. In this way only the intranet devices will able to decrypt it. The most common authentication scheme used is the certificate-based public key one. It brings a new security concept: the public key certificate. It is the binding between a public key and the identity of the legitimate owner. Moreover, this new concept is based on a Public Key Infrastructure (PKI), a hierarchical structure of Certificate Authorities. Each certificate has to be signed by a trusted CA, it means that the CA's certificate has already been signed by another trusted CA. This iterative mechanism of signs brings us to trust the highest CAs a priori. Authentication is reached building a PKI for the VPN. We have to generate a certificate for each device of the VPN. Then we ask a CA to sign them in order to state the devices veracity. But how can we verify a certificate validity? We know that the internal device will accept only clients which its certificate is signed by a specific CA. When a client tries to connect to the VPN server, it sends the certificate and its associated sign. Notice that a sign is the output of a hash function applied on the certificate encrypted with the private key of the CA. Then the server performs three actions: first it applies the hash function to the certificate received, then it decrypts the sign with the CA public key getting the client certificate hash, finally if the hashes coincides the client is authenticated. If a private key is stolen, the CA can just revoke the associated certificate, without the need of rebuilding all the PKI.

Describe IPTABLES fields and provide an example

Iptables is the userspace application in order to manage the Netfilter tables. Its structure is composed by chains and tables. Chains are just mapping of Linux Kernel hooks. Every time we use the iptables command we have to specify some arguments. First is the nature of the command itself, it means if we want to add, remove, rename a rule or if we want to set a policy to a chain. The second parameter is the chain, we must specify which hook we want to modify. It is possible also to specify the table within the chain. Notice that just a subset of tables is associated to each chain. For example, to the INPUT chain (mapping of LOCAL_INPUT hook) are associated the FILTER and the MANGLE table. Then we have to specify the matching rule, it other words all the parameters that identify the interested packets. The matching rules can be composed by a lot of parameters: we can distinguish among all the packets by the source/destination address, the protocol and/or the ports used, the state, the incoming or outgoing interface and so on so forth. Finally, we have to decide an action, called also target, that has to follow if a packet matches that rule. The most common targets are DROP or ACCEPT for the filter table, or SNAT, DNAT, MASQUERADE for the nat table. An example of

usage, in the case we want to set a policy to the FORWARD chain, is iptables -P FORWARD DROP, or another case we want to DROP all the packets that use the UDP protocol in the OUTPUT chain is iptables -A OUTPUT -p UDP DROP. It can be noticed that if the table parameter is not set, it assumes the default value FILTER.

DNS and recursive queries

During the grow up of the public network, a new problem arose: how can humans access the right device without remembering its IP? Around the world too much IPs exist to think to remember all of them. Other problems are linked to this one like the load balancing, the change of an IP and others. These problems were solved with the introduction of DNS (Domain Name System). A DNS is a kind of database that stores the IPs and their associated names. In order to reach efficiency, the DNS is a distributed database. How can we take advantages form the distributed nature of DNS? We can see the distributed structure of the DNS like an inverted tree. Each node refers to a group of DNS Servers that manage a Zone. They are identified by a domain name. All the hosts that we can reach are the leaves of the tree. So, every time a client makes a query to DNS, he asks for the IP of a host (leaf). It must specify the Fully Qualified Domain Name of that host. It is a path through the inverted tree composed of specific domain names linked by dots (for example www.example.com). There are two ways to make the queries: the iterative one and the recursive one. In the first approach, a client read all the domain names of the FQDN from the right to the left, then it asks in that order to each associated DNS Server. Each DNS Server will response with the IP address of the requested resource. Let's make an example: if we want to solve the www.example.com FQDN, first of all we have to ask to the "com." DNS Server where is the "example.com." Server. Next we can ask to the "example.com." server where is the "www.example.com." resource. At the last iteration we will have the IP of the FQDN. In the second approach, instead, the client doesn't make all those queries. The client asks to a single DNS Server and waits for a response. This server will resolve the query asking recursively in the same way as I describe before. At the end the DNS Server will give the resolved IP to the client. The leaves of the tree can be a name server NS, a mail exchanger MX, an IPv4 A, an IPv6 AAAA, a canonical name CNAME and so on so forth. Thanks to the DNS, the network structure is hidden to the clients. If a host changes its IP, it is enough to change one entry in the DNS Server associated. The client doesn't know that it's happened. Moreover, this system is reliable thanks to the redundancy of information. There are a lot of copy servers that are activated in case one fails.