- What did you learn yesterday/this week?
    - I was reading about differences types of encryption, and their usage, when use an asymmetric key or a symmetric key and what are the limits and applications of GCP when need to encrypt backups for example.
- Talk about your preferred development/administration environment. (OS, Editor, Browsers, Tools, etc.)
    - OS - Linux editor - Vscode Browser- chromium/firefox tools - terraform, vim, git, ZSH, ohmyzsh, Python, ansible, terminator, packer, meld, helm, kubectl, aws tools, kubetail, eksctl distro - Arch Linux environment - i3wm + lxde
- Tell me about the last major Linux project you finished.
    - My major project in life was an e-commerce creation/migration project using opensource tools like centos, java, springcloud, mongodb, centos, rabbitmq, cassandra, kafka+zookeeper, Azure, percona-mysql, NetflixOSS (as API gateway), nodejs, took 2 years to complete, I joined since the very beginning.
- Tell me about the biggest mistake you've made in [some recent time period] and how you would do it differently today. What did you learn from this experience?
    - I hired someone that it's a very good engineer but has very bad social skills. This new person broke completely the team synergy and at the end the team itself. I learned that friendly > technical skills
- Why we must choose you?
    - I have a lot of experience, with different types of environments, I worked with all three main cloud providers (GCP, AWS and Azure), in different size of companies. I have a huge experience scaling services, learning new technologies, automating simple tasks, a cool mind to troubleshoot and solve problems in critical environments. I have been working as tech lead, and product owner for SRE team.
- What function does DNS play on a network?
    - It's on the core for any environment, responsible to translate IP addresses into names, DNS can also provides a load balancer layer using geolocation, service discovery using SRV entry and a lot of others features, like domain ownership confirmation using TXT entries (useful to generate SSL certs, for example)
- What is HTTP?
    - HTTP (hypertext transport protocol) it's a protocol that defines how messages are formated and transmitted via web, and what actions webservers and browsers should take in response of various commands.
- What are HTTP status codes?
    - HTTP status codes are predefined status of the task at the server
        - 1xx - represents informational responses
        - 2xx - represents succesful responses

        - 3xx - represents redirect responses
        - 4xx - represents client errors
        - 5xx - represents server errors
    - The most commmons status codes are:
        - 200 Success/OK
        - 201 - CREATED - used by POST or PUT methods
        - 304 - NOT MODIFIED - used in conditional GET Request to reduce bandwitdth use
        - 400 - BAD REQUEST - This can be due to validation errors or missing input data
        - 404 - NOT FOUND - Resource method is not available
        - 500 - INTERNAL SERVER ERROR - server threw some exceptions while running the method
        - 502 - BAD GATEWAY - Server was not able to get the response from another upstream server
- Describe the most common HTTP methods/verbs, and give examples:
    - GET - Read only operation, used to fetch detail from the server, downloads
    - POST - This method is used for the creationg of new resources on the server
    - PUT - This method is used to update existing resource on the server or to replace the resource, PUT it's indepotent, and POST isn't, with PUT you can update a resource N times, but if you try with post you will create N resources. PUT can create resources.
    - PATCH - Applies a partial update to a resource and doesn't create a new resource
    - DELETE - This method is used to delete the resource on the server
    - TRACE - Provides a loop back test along the path to the target resource providing a useful debugging mechanism.
    - OPTIONS - Fetches the list of supported options of resources present on the server.
- What is an HTTP proxy and how does it work?
    - An HTTP proxy it's a service that forwards HTTP connections, for example a user A want's to access a server www.b.com, but for security reasons the user cannot have direct access on internet, so the user's browser will ask for the HTTP proxy to access the website, the webserver from www.b.com will receive a HTTP message from the proxy and will answer as usual, but in the http HEADER from the message will be changed by the HTTP proxy server who will add or change some headers like User-Agent, X-Forwarded-For, used a lot by companies that need to control their users Internet access.
- Describe briefly how HTTPS works.

    - HTTPS use the same HTTP protocol but creates a security layer(tunnel) using SSL/TLS, on top of it, this prevents anyone modify or inspect what's happening inside this tunnel and ensure the client it's communicating with the right server. The SSL handshake is established and after that all HTTP responses are send by. The SSL uses both types of encryption, symmetrical and assymetrical, first when the key exchange happens, a assymetrical encriptions is put in place to the result of the key exchange be a symmetrical encryption.

- What is SMTP? Give the basic scenario of how a mail message is delivered via SMTP.
    - SMTP (Simple Mail transport protocol) works in the application layer, and uses a process called "store and forward", working close to a MTA (Mail Tranfer Agent), this MTA service sends via SMTP a package with the messages, when this message arrives at the destination, the client will use POP3/IMAP to download it.
- What is RAID? What is RAID0, RAID1, RAID5, RAID10? RAID (Redundant Array of Independent Disks) it' s a data storage virtualization technology that combines multiple physical disks in one logical volume.
    - RAID0 - Striping, the data is striped between 2 or more disks, improving speed but compromising availability.
    - RAID1 - Mirroring, the data is mirrored between 2 or more disks, improving availability (the server could lose N - 1 disks) but compromises the speed (replication time).
    - RAID5 - Blocks striped but uses distributed parity, RAID5 uses minimum 3 disks, that stripe data between them, but replicates blocks too, it's a good RAID for databases, in this scenario read operations are good, but write can be slow.
    - RAID10 - It's a implementation using RAID0 + RAID1, uses 4 disks, striping data between 2 of then and mirroring this data in another 2 disks.
- What is a level 0 backup? What is an incremental backup?
    - Level0 backup it's a full backup (all blocks), an after a level 0 backup, we can initializes an incremental backup rotine (only the difference between the blocks)
- Describe the general file system hierarchy of a Linux system.
    - / - root folder
    - /etc - configuration files provided by the package manager
    - /bin - binaries files
    - /sbin - system binaries files (important binaries for the OS)
    - /boot - Static files for boot processes (boot loader)
    - /dev - Device files
    - /lib - Essential shared libraries and kernel modules
    - /usr - Secundary hierarchy

- /mnt - Mounting point for temporary filesystem
- /media - Mounting point for removable media
- /opt - Add-on application software packages
- /srv - Data service provided by this system
- /tmp - Temporary files
- /var - Variable data
- /root - Root user folder
- /home - Home users folders