

DevOps Interview Preparation Training

1) What is configuration management and how does it helps an organization?

Configuration Management:

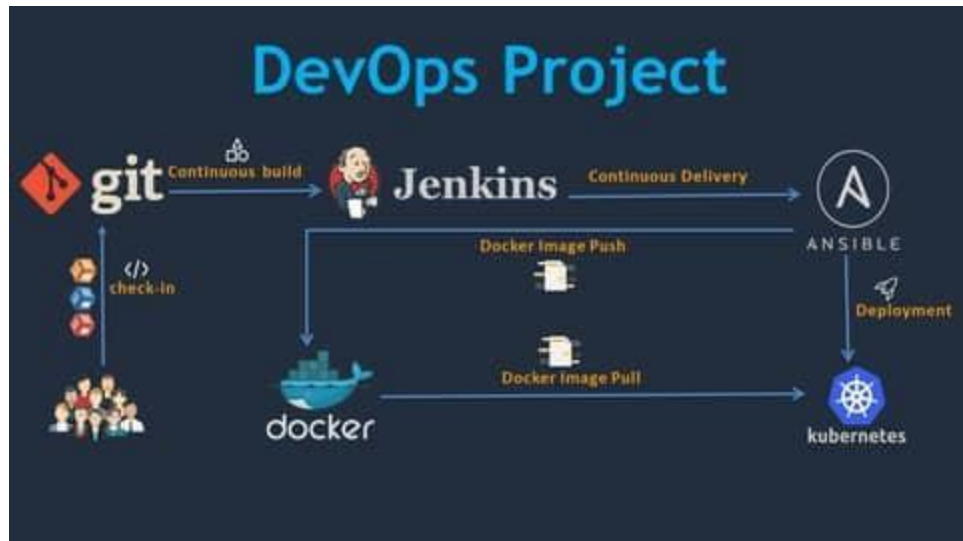
- Configuration management tools manages configuration IT infrastructures like OS, network devices, Application, software's and cloud computing services are called as configuration management tools.
- Configuration of IT Infrastructure like software and OS changes like add, remove and update them for good.
- Doing such changes to hundreds of servers and devices is very time consuming and error prone task if you do it manually.
- These tools help you manage and automate all that changes with ease and from a centralized place.
- Configuration of various servers and devices are managed from a central server.
- We can manage web servers, db servers or switch all of their configuration like software, patches, config files etc.

2) Why do we need Ansible, what activities are you doing in the project using ansible?

- Ansible is an open source IT Configuration Management, Deployment & Orchestration tool.
- It aims to provide large productivity gains to a wide variety of automation challenges.
- This tool is very simple to use yet powerful enough to automate complex multi-tier IT application environments.

Activities on Project:

- In DevOps, as we know development and operations work is integrated. This integration is very important for modern test-driven application design. Hence, Ansible integrates this by providing a stable environment to both development and operations resulting in smooth orchestration.



- When developers begin to think of infrastructure as part of their application i.e as Infrastructure as code (IaC), stability and performance become normative. Infrastructure as Code is the process of managing and provisioning computing infrastructure (processes, bare-metal servers, virtual servers, etc.) and their configuration through machine-processable definition files, rather than physical hardware configuration or the use of interactive configuration tools. This is where Ansible automation plays a major role and stands out among its peers.

3) Difference between Ansible and Terraform

Terraform	Ansible
Terraform is a provisioning tool.	Ansible is a configuration management tool.
It follows a declarative Infrastructure as a Code approach.	It follows a procedural approach.
It is the best fit for orchestrating cloud services and setup cloud infrastructure from scratch.	It is mainly used for configuring servers with the right software and updates already configured resources.
Terraform does not support bare metal provisioning by default.	Ansible supports the provisioning of bare metal servers.
It does not provide better support in terms of packaging and templating.	It provides full support for packaging and templating.

It highly depends on lifecycle or state management.	It does not have lifecycle management at all
---	--

4) What is a YAML file and how do we use it in Ansible?

Ansible uses YAML syntax for expressing Ansible playbook.

To understand better please follow given link:

<https://ansible-docs.readthedocs.io/zh/stable-2.0/rst/YAMLSyntax.html>

https://docs.ansible.com/ansible/latest/reference_appendices/YAMLSyntax.html

5) What is Ansible Inventory and its types?

- Inventory is a text file where you define the host information that you want to manage with ansible.
- The default inventory file location is `/etc/ansible/hosts`.

Ansible there are two types of inventory: Static and Dynamic

Static:

- In Ansible, a static inventory file is a plain text file that contains a list of managed hosts declared under a host group using either hostnames or IP addresses.
- Ansible also allows groups of hosts to be placed under a group name as like **webserver**s and **database_servers** groups have been placed under the **datacenter** group.

Dynamic:

- In a configuration – especially a cloud setup such as **AWS** where the inventory file keeps constantly changing as you add or decommission servers, keeping tabs on the hosts defined in the inventory file becomes a real challenge. It becomes inconvenient going back to the host file and updating the list of hosts with their IP addresses.
- A dynamic inventory is a script written in Python, PHP or any other programming language. It comes in handy in cloud environments such as AWS where IP addresses change once a virtual server is stopped and started again.

6) What is the ad-hoc command in Ansible?

- An Ansible ad hoc command uses the `/usr/bin/ansible` command-line tool to automate a single task on one or more managed nodes. ad hoc commands are quick and easy, but they are not reusable.

- Ansible gives a quick method to communicate and execute commands on remote/local machines through Adhoc commands
- **Refer:**
https://docs.ansible.com/ansible/latest/user_guide/intro_adhoc.html#:~:text=An%20Ansible%20ad%20hoc%20command,but%20they%20are%20not%20reusable.

7) Install Nginx using Ansible playbook?

Follow the refer link for Nginx Installation:

<https://www.digitalocean.com/community/tutorials/how-to-deploy-a-static-html-website-with-ansible-on-ubuntu-20-04-nginx>

8) What are callback plugins in Ansible?

Callback plugins enable adding new behaviors to Ansible when responding to events. By default, callback plugins control most of the output you see when running the command line programs, but can also be used to add additional output, integrate with other tools and marshal the events to a storage backend.

Refer link:

<https://docs.ansible.com/ansible/latest/plugins/callback.html#:~:text=Callback%20plugins%20enable%20adding%20new,events%20to%20a%20storage%20backend.>

9) What is notify handlers?

- Handlers are special tasks that only get executed when triggered via the notify directive.
- Handlers are executed at the end of the play once all tasks are finished.
- In Ansible handlers are typically used to start, reload, restart and stop services.
- To understand Refer Link:
<https://www.digitalocean.com/community/tutorials/how-to-define-and-use-handlers-in-ansible-playbooks>

10) list out some of the modules are you using in the project

The frequently used modules in real time as below example:

- ping
- copy
- fetch
- shell
- file
- package
- yum
- service
- user and group

- lineinfile

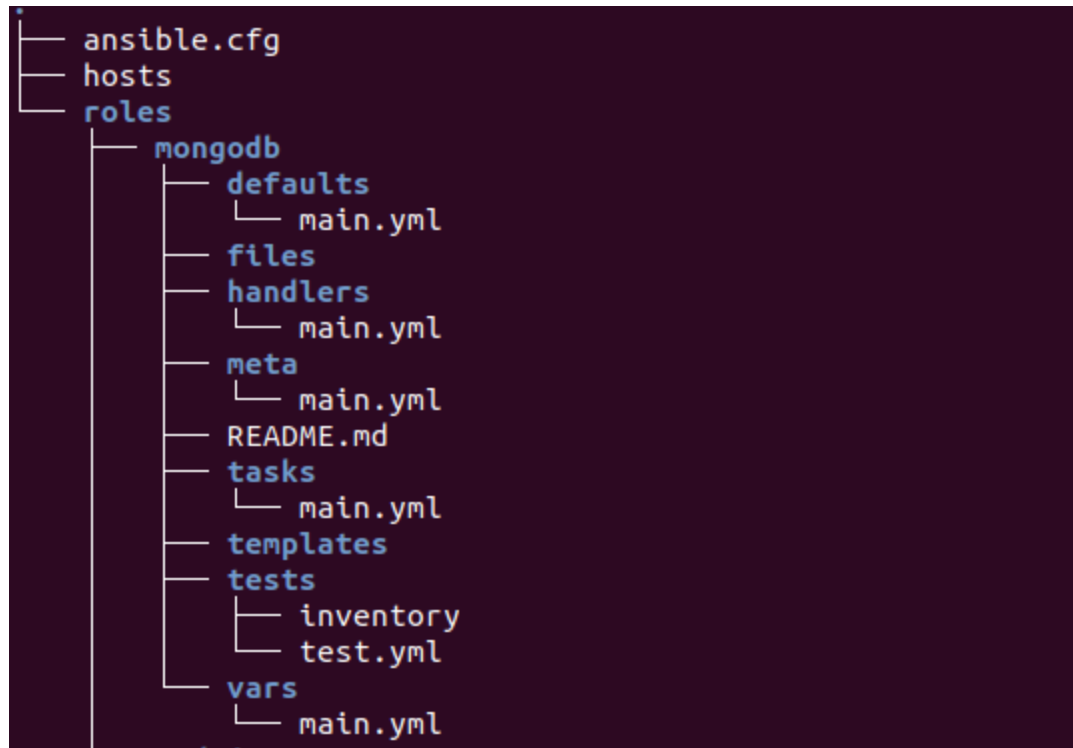
Note: Above are module example, however you will explain yours example which you are confident to explain the syntax and usecase.

11) How to use Ansible Vault in Playbooks to Protect Sensitive data?

- The “Vault” is a feature of Ansible that allows you to keep sensitive data such as passwords or keys in encrypted files, rather than as plaintext in playbooks or roles.
- These vault files can then be distributed or placed in source control.
- To enable this feature, a command line tool, ansible-vault is used to edit files, and a command line flag --ask-vault-pass or --vault-password-file is used.
- You can also modify your ansible.cfg file to specify the location of a password file or configure Ansible to always prompt for the password.
- Refer link:
https://docs.ansible.com/ansible/2.5/user_guide/playbooks_vault.html#:~:text=To%20enable%20this%20feature%2C%20a,always%20prompt%20for%20the%20password.

12) What kind of roles have you used in your project?

- While it's possible to write a playbook in one very large file eventually you will want to reuse files and start to organize things.
- In our playbook we have variables, tasks, handlers and templates. This can slowly grow and will become eventually difficult to read and manage if it is single page playbook.
- **Roles** is a directory structure where we distributed the content of our main playbook into proper directory structure.



- **Files:** This directory contains regular files that need to be transferred to the hosts you are configuring for this role. This may also include script files to run.
- **Handlers:** All handlers that are in your playbook which can be added into this directory.
- **Meta:** This directory can contain files that establish role dependencies. You can list roles that must be applied before the current role can work correctly.
- **Templates:** You can place all files that use variables to substitute information during creation in this directory.
- **Tasks:** This directory contains all of the tasks that would normally be in a playbook. These can be reference files and templates contained in their respective directories without using a path.
- **Vars:** Variables for the roles can be specified in this directory and used in your configuration files.

Note: Instead of having all our code in one playbook, we can distribute it into different directory structure.

Linux Interview Questions and Answer:

13) What is swap space?

- Swap space, as the name suggests, is basically a space on a hard disk that is used when the amount of physical memory or RAM is full.
- It is considered a substitute for physical memory.

- Its main function is to substitute disk space for RAM memory when real RAM does not have enough space to hold all programs that are executing, and more space is required.
- In simple words, it can be used as an extension of RAM by Linux.

14) What do you mean by a Process States in Linux?

Linux Process is a type of Process that can be in a number of different states. The process enters these states from start till end as follows:

- **New/Ready:** In this state, a new process is created and ready to run.
- **Running:** In this state, the process is being executed.
- **Blocked/wait:** In this state, the process is waiting for input from the user and if doesn't have resources to run such as memory, file locks, input, then it can remain in a waiting or blocked state.
- **Terminated/ Completed:** In this state, the process has completed the execution or terminated by the OS.
- **Zombie:** In this state, the process is terminated but information regarding the process still exists and is available in the process table.

15) Name different types of modes used in VI editor.

VI/VIM editor (visual editor) is basically a default text editor that usually comes with most of the Linux OS.

There basically three types of mode.

1. **Command Mode/ Regular Mode:** When vi starts up, it is in Command Mode. This mode is where vi interprets any characters we type as commands and thus does not display them in the window. This mode allows us to move through a file, and to delete, copy, or paste a piece of text. To enter into Command Mode from any other mode, it requires pressing the **[Esc]** key. If we press [Esc] when we are already in Command Mode, then vi will beep or flash the screen.
2. **Insert Mode:** This mode enables you to insert text into the file. Everything that's typed in this mode is interpreted as input and finally, it is put in the file. The vi always starts in command mode. To enter text, you must be in insert mode. To come in insert mode you simply type i. To get out of insert mode, press the Esc key, which will put you back into command mode.
3. **Escape Mode:** Line Mode is invoked by typing a colon [:], while vi is in Command Mode. The cursor will jump to the last line of the screen and vi will wait for a command. This mode enables you to perform tasks such as saving files, executing commands.

16) What do you mean by the daemons?

- Daemons also referred to as the background process, is a long-running Linux program that runs in the background.
- They do not have any controlling terminal, therefore, they run in the background.
- These are the processes that are generally started when the system is bootstrapped and terminate or end only when the system is shut down.
- It is simply the way of extending the functionality of the base OS.
- It provides and offers several functions that are not available in OS.
- Its main purpose is to handle periodic requests and then forward the requests to the appropriate programs for execution.

17) What is load average in Linux?

- Load average, as the name suggests, is the average system load on Linux servers being calculated over a given period of time.
- The load average of Linux servers can be found using “**top**” and “**uptime**” commands.
- It is simply used to keep track of system resources.
- It is represented by a decimal number starting at 0.00.
- It tells you the load that the system has been under.

18) What are INODE and Process Id?

- **INODE:** It is a unique name given to each file by OS. Each inode has a unique inode number within a file system. It stores various information about files in Linux such as ownership, file size, file type, access mode, number of links, etc.
- **Process Id (Identifier):** It is a unique Id given to each process. It is simply used to uniquely identify an active process throughout the system until the process terminates.

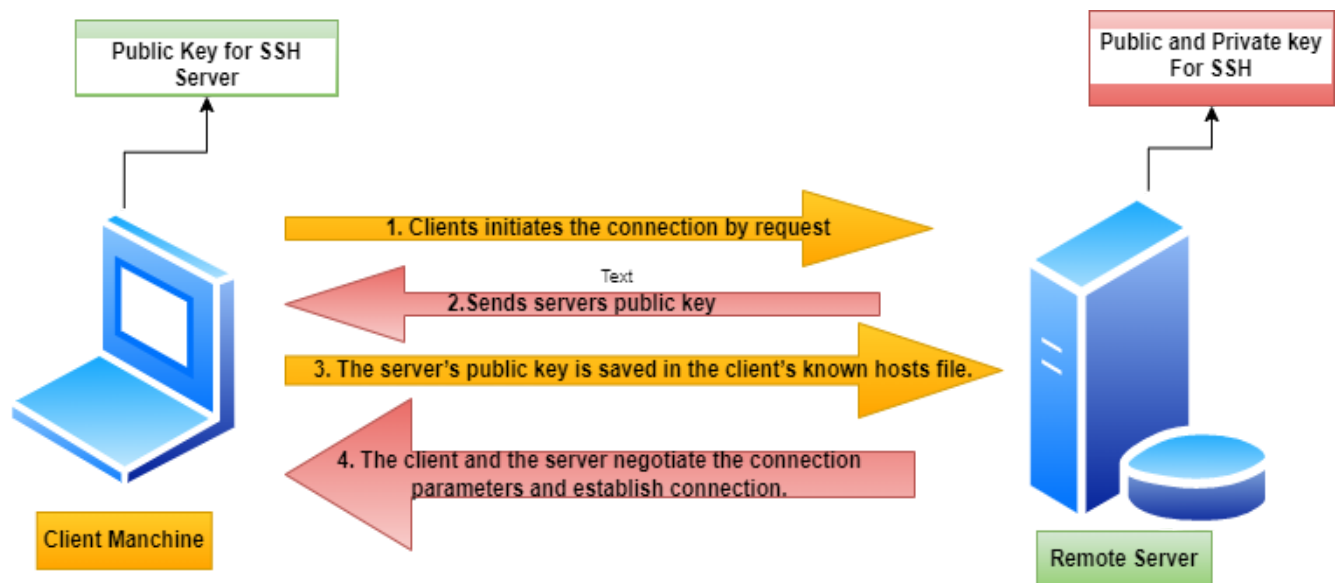
19) Name default ports used for DNS, SMTP, FTP, SSH, DHCP and squid.

Default ports used for various services are as follows:

- DNS = 53
- SMTP= 25
- FTP = 20 (Data transfer), 21 (Connection established)
- SSH = 22
- DHCP = 67/ UDP (dhcp server), 68/ UDP (dhcp client)
- Squid = 3128

20) What is SSH? How we can connect to a remote server via SSH.

- Secure Shell, sometimes referred to as **Secure Socket Shell**, is a protocol which allows you to connect securely to a remote computer or a server by using a text-based interface.
- When a secure SSH connection is established, a shell session will be started, and you will be able to manipulate the server by typing commands within the client on your local computer.
- System and network administrators use this protocol the most, as well as anyone who needs to manage a computer remotely in a highly secure manner.



21) Write the difference between Soft and Hard links?

Hard Links:

- It is a special kind of file that points to the same underlying inode as another file.
- It can be referred to as an additional name for an existing file on Linux OS.
- Total number of hard links for a file can be displayed using the "ls -l" command.
- Such links cannot be used across file systems.
- Hard links can be created using the following command: `$ ln [original filename] [link name]`

Soft Links:

- It is also termed a symbolic Link.
- Soft links are kinds of files that usually point to another file.
- It does not include any amount of data in the target file and simply points to another entry anywhere in the file system.
- Such links can be used across file systems.
- Soft links can be created using the following command: `$ ln -s [original filename] [link name]`

22) What is netstat command?

- netstat (Network statistics) command is generally a networking tool being used for troubleshooting and configuration and used to display all network connections on a system.
- It simply provides a way to check whether various aspects of TCP/IP are working and what connections are present.

23) Which command is used to check the size of file or directory?

- The command used to check the size of the file or directory is “du”.
- Here “du” stands for disk usage that is used to check information of disk usage of files and directories on a machine.
- It is also used to display files and directory sizes in a recursive manner.
- Example : `$ du -sh *` -- In this command you can check the total file size on the current directory

24) Explain working of env command.

- “env” command is basically a shell command that is used to print a list of current environmental variables.
- Here, “env” stands for the environment.
- It can also run another process in another environment without any modification of the current environment.
- It allows you to run programs in a modified environment.
- It is generally used by shell scripts to launch the correct interpreter.
- It can also be useful for checking if wrong environment variables prevent the application from starting during troubleshooting.

25) What is pipe?

- A pipe is basically a form of redirection that is used to send the output of one command to another command for further processing.
- It simply takes the output from one command and uses it as an input for another.

- It provides asynchronous execution of commands with help of buffered I/O routines.

26) What do you mean by umask?

- **Umask**, also known as user file creation mask, is a linux command that allows you to set up default permissions for new files and folders that you create.
- In Linux OS, umask command is used to set default file and folder permission.
- When root user created , directory =755 and file = 644.
- Root user default umask value is 022
- Normal user default umask value is 002

27) What command would you use to check how much memory is being used by Linux?

\$ Free -m

28) How to terminate a running process in Linux?

- Every process has a unique **process id**.
- To terminate the process, we first need to find the process id.
- The **ps** command will list all the running processes along with the process id.
- And then we use the **kill** command to terminate the process.
- E.g. \$ kill -9 <PID>

29) What is a Zombie Process?

- Zombie Process, also referred to as a defunct or dead process in Linux, is a process that has finished the execution, but its entry remains in the process table. It usually happens due to a lack of correspondence between parent and child processes.
- This process occurs for the child process because the parent process needs to read the status of the child process.
- Once it is completed using the wait system call, this process is removed from the process table.

30) Name the command that is used to check all the listening ports and services

To list all TCP or UDP ports that are being listened on, including the services using the ports and the sockets status use the following command:

\$ netstat -lntp

The options used in this command have following meaning:

- -t – show TCP ports
- -u – Show UDP ports

- -n – Show numerical addresses instead of resolving hosts
- -l – Show only listening ports
- -p – Show the PID and name of the listener's process.

Note:

- This information is shown only if you run the command as root or sudo user.

Special Note:

- *Please prefer our DIP video to understand better about these all answer with better explanation*