30 Expected DevOps Interview Questions and Answers

August 18, 2016 | By Linoxide in DEVOPS

If you are searching for devOps job, you are making good career decisions since it is well paid and heavily demanded job nowadays. In this article, we will go through the DevOps interview questions that you might expect from your interviewer. The questions are many and we couldn't possibly list all of them, but after reading this article you should have a better idea what awaits you and what you should be prepared for.

Basic Questions

1) DevOps ! How can you define it in your words ?

Its highly effective daily collaboration between software developers and IT operations / web operation engineers to produce a working system or release software.

A devOps implementation is generally aligned with Agile methodologies where deploying working software to Production is generally the highest priority. On Agile implementations, emphasis is placed on people over processes, so a DevOps engineer must be willing to work very closely with Agile development teams to ensure they have an environment necessary to support functions such as automated testing, continuous Integration and continuous Delivery. On a traditional implementation, without DevOps, the operations team is often isolated from developers, often working under a help desk model under general service level agreements where the system operations team treats developers as a customer. This is a proven model which obviously can work very well, but in a DevOps environment, development and operations are streamlined and barriers between the two groups should not exist.

2) Why we need DevOps ?

Companies are now facing the need to delivering more and faster and better applications to meet the ever more pressing demands of conscious users to reduce the " Time To Market ". Devops often helps deployment to happen very fast.

3) What is agile development and Scrum ?

Agile development used as an alternative to Waterfall development practice. In Agile, the development process is more iterative and incremental, there is more testing and feedback at every stage of development as opposed to only the last stage in Waterfall.

Scrum is used to manage complex software and product development, using iterative and incremental practices. Scrum has three roles ie product owner, scrum master, and team.

4) Can we consider DevOps as an agile methodology ?

Of course! DevOps is a movement to reconcile and synchronize development and production start through a set of good practices . Its emergence is motivated by a deep changing demands of business, who want to speed up the changes to stick closer to the requirements of business and the customer.

5) What is DevOps engineer's duty with regards to Agile development ?

DevOps engineer work very closely with Agile development teams to ensure they have an environment necessary to support functions such as automated testing, continuous Integration and continuous Delivery. DevOps engineer must be in constant contact with the developers and make all required parts of environment work seamlessly.

Technical Questions

6) Have you worked on containers ?

Containers are form of lightweight virtualization, more heavy than chroot but lighter than hypervisors. They provide isolation among processes while using same kernel as the host machine, and cgroups functionality within kernel. But container formats differ among themselves in a way that some provide more VM-like experience while other containerize only application.

LXC containers are most VM-like and most heavy weight, while Docker used to be more light weight and was initially designed for single application container. But in more recent releases Docker introduced whole machine containerization features so now Docker can be used both ways. There is also rkt from CoreOS and LXD from Canonical, which builds upon LXC.

7) What is Kubernetes? Explain

It is massively scalable tool for managing containers, made by Google. It is used internally on huge deployments and because of that it is maybe the best option for production use of containers. It supports self healing by restating non responsive containers, it pack containers in a way that they take less resources and has many other great features.

8) What is the function of CI (Continuous Integration) server ?

CI server function is to continuously integrate all changes being made and committed to repository by different developers and check for compile errors. It needs to build code several times a day, preferably after every commit so it can detect which commit made the breakage if the breakage happens.

Note: Other available and popular CI tools are Jenkins, TeamCity, CircleCI , Hudson, Buildbot etc

9) What is Continuous Delivery ?

Is it practice of delivering the software for testing as soon as it is build by CI (Continuous Integration) server's. It requires heavy use of Versioning Control System for so always available to developers and testers alike.

10) What is Vagrant and what is it used for ?

Vagrant is a tool that can create and manage virtualized (or containerized) environments for testing and developing software. At first, Vagrant used virtualbox as the hypervisor for virtual environments, but now it supports also KVM.

11) Do you ever used any scripting language ?

As far as scripting languages go, the simpler the better. In fact, the language itself isn’t as important as understanding design patterns and development paradigms such as procedural, object-oriented, or functional programming.

Currently, several scripting languages are available so the question arises : what is the most appropriate language for DevOps approach? Simply everything , it depends on the context of the project and tools used for example if Ansible used its good have knowledge in Python and if its for Chef its on Ruby.

12) What is the role of a configuration management tool in devops ?

Automation plays an essential role in server configuration management. For that purpose we use CM tools , they store information about versions and builds of the software and testware and provide the traceability between software and testware.

13) What is the purpose of CM tools and which one you have used ?

Configuration Management tools' purpose is to automatize deployment and configuration of software on big number of servers. Most CM tools usually use agent architecture which means that every machine being manged needs to have agent installed. My favorite tool is one that uses agentless architecture - Ansible. It only requires SSH and Python. And if raw module is being used, not even Python is required because it can run raw bash commands. Other available and popular CM tools are Puppet, Chef, SaltStack.

14) What is OpenStack ?

OpenStack is often called Cloud Operating System, and that is not far from the truth. It is the complete environment for deploying IaaS which gives you possibility of making your own cloud similar to AWS. It is highly modular and consists of many sub-projects so you can pick and chose which functionality you need. OpenStack distribution are available from Red Hat, Mirantis, HPE, Oracle, Canonical and many others. It is completely open source project but some vendors make proprietary distributions.

15) Classify Cloud Platforms anategory ?

Cloud Computing software can be classified as Software as a Service or SaaS, Infrastructure as a Service or IaaS and Platform as a Service or PaaS.

SaaS is peace of software that runs over network on remote server and has only user interface exposed to users, usually in web browser. For example salesforce.com.

Infrastructure as a service is a cloud environment that exposes VM to user to use as entire OS or container where you could install anything you would install on your server. Example for this would be OpenStack, AWS, Eucalyptus.
PaaS allows users to deploy their own application on the preinstalled platform, usually framework of application server and suite of developer tools. Examples for this would be OpenShHeroku.

16) What are easiest ways to build a small cloud ?

VMfest is one one of the options for making IaaS cloud from VirtualBox VMs in no time. If you want a lightweight PaaS there is Dokku which is basically a bash script that makes PaaS out of Dokku containers.

17) What is AWS (Amazon Web Services)? Did got chance to work on Amazon tools ?

AWS provides a set of flexible services designed to enable companies to create and deliver products with greater speed and reliability using AWS and DevOps practices . These services simplify commissioning and infrastructure management , application code deployment , automated software release process and monitoring of the application and infrastructure performance. Amazon used tools like AWS CodeCommit, AWS CodeDeploy, AWS CodePipeline etc, that helps to make devops easier.

18) What is EC2 ?

Amazon EC2 Container Service (ECS) is a highly scalable container management service and high performance that supports the Docker containers and allows you to easily run applications on a cluster managed by Amazon EC2 instances.

The EC2 service is inseparable from the concept of Amazon Machine Image - AMI . The May is Indeed the image of a virtual machine That Will Be Executed . EC2 based on XEN virtualization , that's why it is quite easy to move XEN servers to EC2 .

19) Do you find any advantage of using NoSQL database over RDBMS ?

Typical web applications are built with a three-tier architecture. To carry the load, more Web servers are simply added behind a load balancer to support more users. The ability to scale out is a key principle in the world of cloud computing, more and more important in which VM instances can be easily added or removed to meet demand.

However, when it comes to the data layer, relational databases (RDBMS) does not allow a passage to the simple scale and do not provide a flexible data model. Manage more users means adding more servers and large servers are very complex, owners and disproportionately expensive, in contrast to low-cost hardware, the "commodity hardware", architectures in the cloud. Organizations are beginning to see performance issues with their relational databases for existing or new applications. Especially as the number of users increases, they realize the need for a faster and more flexible basis. This is the time to begin to assess and adopt NoSQL database like in their Web applications.

20) What are the main SQL migration difficulties NoSQL ?

Each record in a relational database according to a schema - with a fixed number of fields (columns) each having a specified object and a data type. Each record is the same. The data is denormalized in several tables. The advantage is that there is less of duplicate data in the database. The downside is that a change in the pattern means performing several "alter table" that require expensive to lock multiple tables simultaneously to ensure that change does not leave the database in an inconsistent state.

With databases data, on the other hand, each document can have a completely different structure from other documents. No additional management is required on the database to manage changes in the schemes.

21) What are the benefits of NoSQL databases Documents ?

The main advantages of document databases are the following :

flexible data model data can be inserted without a defined schema and format of the data that is inserted can change at any time , providing extreme flexibility , which ultimately allows a significant agility to business
Consistent , high-performance Advanced NoSQL database technologies are putting cache data , transparently, in system memory ; a behavior that is completely transparent to the developer and the team in charge of operations .
Some easy scalability NoSQL databases automatically propagate data between servers , requiring no participation applications. Servers can be added and removed without disruption to applications , with data and I/O spread across multiple servers.

22) What are the main advantages of Git over CVS ?

The biggest advantage is that Git is distributed while CVS is centralised. Changes in CVS are per file, while changes (commits) in Git they always refer to the whole project. Git offers much more tools than CVS.

23) Difference between containers and virtual machines ?

Each VM instantiation requires starting a full OS. VMs take up a lot of system resources. This quickly adds up to a lot of RAM and CPU cycles. Container host uses the process and file system isolation features of the linux kernel.

24) What is CoreOS, and what are alternatives ?

CoreOS is striped down linux distribution meant for running containters, mainly with its own rkt format but others are also supported. It was initially based on ChromeOS and supported Docker. The alternatives to this are canonical's ubuntu snappy or red hat enterprise linux atomic host. Of course, Containers can also be ran on regular Linux system.

25) What is Kickstart ?

It is a way to install Red Hat based systems by automated way. During manual install process, Anaconda installer creates file anaconda-ks.cfg which then can be used with system-config-kickstart tool to install same configuration automatically on multiple systems.

26) What are tools for network monitoring? List few

For example, Nagios, Icinga 2, OpenNMS, Splunk and Wireshark. Those tools are used to monitor network traffic, network quality and detect network problems even before they arise. Of those listed, only Splunk is proprietary other are open source.

27) What is Juju ?

Juju is orchestration tool primarily for ubuntu for management, provision and configuration on Ubuntu systems. It is was initially written in Python and since have been rewritten in Go.

28) Give me an examples of how you would handle projects ?

As a DevOps engineer, I would demonstrate a clear understanding of DevOps project management tactics and also work with teams to set objectives, streamline workflow, maintain scope, research and introduce new tools or frameworks, translate requirements into workflow and follow up. I would resort to CI, release management and other tools to keep interdisciplinary projects on track.

29) What is post mortem meetings ?

It is a meeting where we discuss what went wrong and what steps should be taken so that failure doesn't happen again. Post mortem meetings are not about finding the one to be blamed, they are for preventing outages from reoccurring and planing redesign of the infrastructure so that downtime can be minimised. It is about learning from mistakes.

30) What you know about serverless model ?

Serverless refers to a model where the existence of servers is hidden from developers. It means you no longer have to deal with capacity, deployments, scaling and fault tolerance and OS. It will essentially reducing maintenance efforts and allow developers to quickly focus on developing codes.

Examples are Amazon AWS Lambda and Auth0 serveless platform.

Devops Example : Deploying Applications with Ansible

Ansible is a lightweight, extensible solution for automating your application provisioning. Ansible has no dependencies other than Python and SSH. It doesn’t require any agents to be set up on the remote hosts and it doesn’t leave any traces after it runs either. It allows you to significantly simplify our operations by creating easy YAML based playbooks. It’s good for configuration automation, deployments and orchestration.

Components of Ansible

Playbooks : Ansible playbooks are a way to send commands to remote computers in a scripted way. Instead of using Ansible commands individually to remotely configure computers from the command line, you can configure entire complex environments by passing a script to one or more systems.

Ansible playbooks are written in the YAML data serialization format. If you don't know what a data serialization format is, think of it as a way to translate a programmatic data structure (lists, arrays, dictionaries, etc) into a format that can be easily stored to disk. The file can then be used to recreate the structure at a later point. JSON is another popular data serialization format, but YAML is much easier to read.

Let's look at a basic playbook that allow us to install a web application (nginx) in a multiple hosts :

```
hosts: webservers
tasks:
- name: Installs nginx web server
apt: pkg=nginx state=installed update_cache=true
notify:
- start nginx

handlers:
- name: start nginx
service: name=nginx state=started
```

The hosts file : (by default under /etc/ansible/hosts) this is the Ansible Inventory file, and it stores the hosts, and their mappings to the host groups (webservers ,databases etc)

```
[webservers]
10.0.15.22
# example of setting a host inventory by IP address.
# also demonstrates how to set per-host variables.

[repository_servers]
example-repository
#example of setting a host by hostname. Requires local lookup in /etc/hosts
# or DNS.
[dbservers]
db01
```

The SSH key : For the first run, we'll need to tell ansible the SSH and Sudo passwords, because one of the thing that the common role does is to configure passwordless sudo, and deploy a SSH key. So in this case ansible can execute the playbook’s commands in the remote nodes (hosts) and deploy the web application nginx.

Conclusion

Those are some of the questions you might encounter during the interview but when learning about DevOps concepts you by no means should only concentrate on those read everything and anything related to Linux and open source and try any software you might be of any use to you. This article hopefully gives idea where to start. Thank you for reading