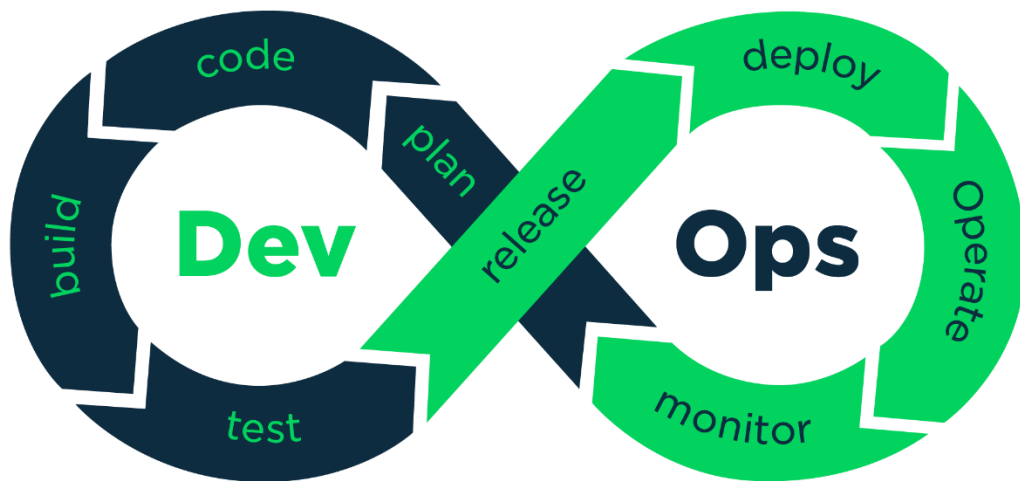


DevOps Engineer day-to-day Activities | DevOps Engineer Responsibilities

Agenda for Today

1. Who is a DevOps Engineer?
2. Role and Responsibilities of a DevOps Engineer
3. DevOps Skills you must possess
4. DevOps Engineer daily tasks or day to day activities

Who is a DevOps Engineer?



Before looking at a DevOps engineer, let us look first at what DevOps is.

DevOps is a software development strategy (SDLC) that bridges the gap between the developers and the IT staff (people operating the software).

This strategy is very advantageous because DevOps can help an organization release small features of the software and incorporate the feedback received very quickly.

Let's talk about the benefits of DevOps.

1. It has fewer software failures than the traditional SDLC methodology such as waterfall and agile.

2. The lead time between the fixes of the software issues is significantly reduced.
3. DevOps is vital because it overcomes all the limitations available in the traditional waterfall methodology.
4. The DevOps processes involve a lot of software testing and many new technologies involving different tools that are used to enhance the automation of the CI/CD pipelines.

DevOps Processes



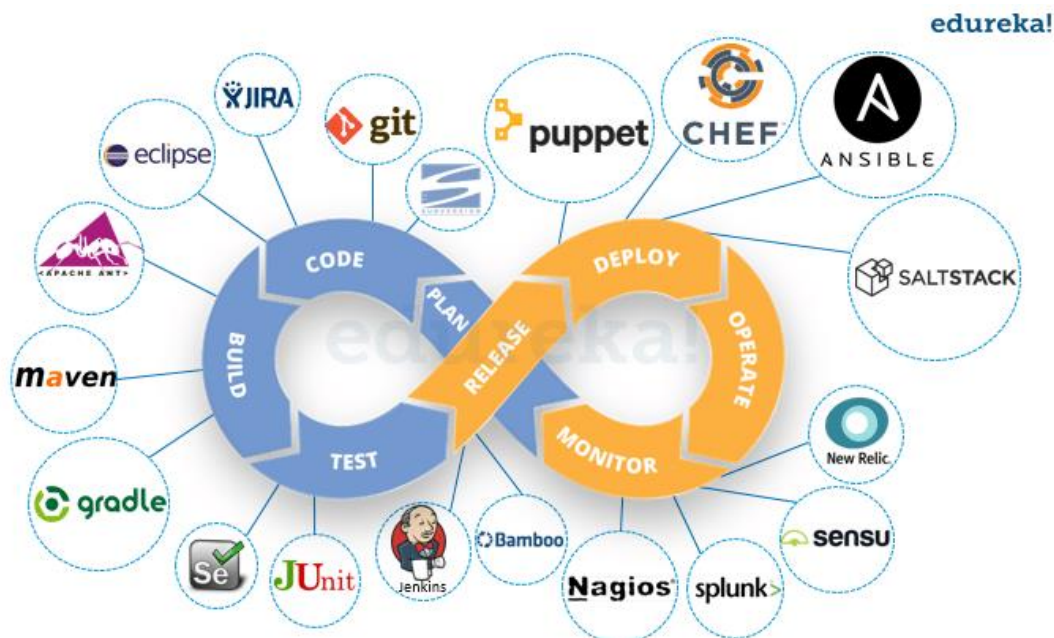
In this order, the DevOps process steps are:

- **Plan:** This is the part of the project where you **organize the tasks schedule** and set up your project management tools. The idea is to plan tasks using the **user story** process from the agile methodology. Writing tickets in the form of a user story will allow developers and Ops to understand what development needs to be done and why. A perfect user story as a what (who, where, trigger), a why, and acceptance criteria. *(Example: As a user, when I click on "option" in my customer account, I open a popup and change my name and first name).*
- **Code:** Here, the developers are doing **code development and code review**. When the code is ready, they merge it. It is essential to share a code tool between Ops and developers' teams in DevOps practice like Github or Gitlab.
- **Build:** This step is the first one toward automation. The goal is to **build the source code** into one desired format, compile, test, and deploy in particular infrastructure. Once this step sets up, the **continuous integration** (CI) and delivery (CD) tools can check and verify the source code from Source Code Management and build it.
- **Test:** The continuous testing process **reduces risks**. Automatic tests ensure that no bugs will be implemented in production. You have to implement testing tools in your workflow to ensure the best development quality for your software.
- **Release:** The code has passed the testing (continuous integration) process and is **ready to be deployed**.
- **Deploy:** The operational team is deploying the new feature in production. But as automation is one of the DevOps principles, it is possible to set up continuous deployment.
- **Operate/ configure infrastructure:** The Ops build or maintain a **scalable infrastructure**, infrastructure as code, and check security issues and log management.
- **Monitor:** Monitoring is an essential step as it **allows you to fix incidents faster** and create a better experience for your end-user.

As DevOps aims to increase the satisfaction of your customers significantly, naturally, your teams start the steps again with a new feature for your software or application. That is why we always **draw the DevOps as an endless loop**.

Let us continue

Let us look at the tools that are used in DevOps.



Source code management or version control systems – git and Github

Pull/merge requests are created if you work in a feature branch and want to merge your change in the main branch(e.g. master branch).

Jenkins – inbuilt plugins (CI/CD server that is used in creating the DevOps CI/CD pipeline)

Selenium – testing (unit testing,

Docker – used to containerize the applications (software containerization platform)

Puppet, Chef, and Ansible – configuration management and deployment tools

Kubernetes – which is the container orchestration tool

Nagios – used for continuous monitoring

Terraform – automate the provisioning of the infrastructures in cloud providers

Sonarqube - SonarQube is an open-source platform for continuous inspection of code quality.

Slack – real-time communication

Slack - Slack is a cloud-based, real-time messaging and notification system. Slack provides **persistent chat**, a more interactive alternative to email for team collaboration. You can communicate with your team on a dedicated channel or on a set of channels related directly to your work.

Service now - managing the change requests. We cannot ever merge QA into Production because there will always be changes in QA that are not ready for deployment, so we must have two separate flows.

Now let's talk about who is DevOps Engineer.

A DevOps engineer understands the software development lifecycle and has an outright understanding of various automation tools for developing the CI/CD pipeline.

DevOps engineers work with developers, and IT staffs to oversee the code releases

Role and Responsibilities of a DevOps Engineer

Roles present in DevOps

DevOps Evangelist – This is the person that is responsible for implementing the entire DevOps lifecycle, from planning, coding, testing, and monitoring

As an evangelist, you have responsibilities to implement the entire CI/CD pipeline.

Release manager – this is someone who releases the software's new features and ensures post releases stability of the software.

Automation Engineer/Expert – this is someone responsible for the automation and orchestration of tools.

Developers and testers – develop and test the code

QA – these are the people responsible for the quality of the product, and they check whether the product is per the requirements.

Security engineers who is responsible for monitoring the health and the security of the product.

These are just some of the major roles present in DevOps

Key responsibilities of a DevOps Engineer

- Administration of the IT infrastructures
- Choosing the suitable deployment models
- Conducting the testing protocol
- Conducting the critical monitoring protocols/activities

If you go to the interview and you face the question, what is your responsibilities as a DevOps Engineer?

1. You are responsible for the build and release processes
2. Managing the user access to the resources - Managing User access on Resources (Grant / Revoke). Ensuring that access on servers is given to required users only after proper approvals.
3. Managing the various branches of git (A tool in the continuous development phase)
4. Automation of the repetitive tasks
5. Providing server and application support to both the Dev and Production teams
6. Ensuring the backup is sufficient, and if not, you are responsible for creating backup policies.
7. Take a backup of instance and restore if required.
8. Taking a backup of Prod DB and providing that DB to developers on Staging / Testing Environment to test any issue.
9. Automation setup for daily tasks like (DB/Instance/Logs/Config-Files) backup.

These are just some of the responsibilities that DevOps engineers do carry out.

DevOps Skills

This is very important because it will help you to know what you need to focus on

1. Linux fundamentals and scripting – As a DevOps engineer, there are times you will need to provision your infrastructures. The process of

provisioning an infrastructure should be automated, and as a result, you should be familiar with at least 1 scripting language. Many companies have their environment on Linux, and many of the configuration tools such as Ansible, puppet, and chef have their master nodes running in Linux. To be a successful DevOps, it is important to know the fundamentals of Linux. This means that a knowledge of one scripting language is mandatory.

2. Knowledge of various tools and technologies – DevOps involves various phases, and there are tools used in each phase. You should aim to know the development, testing, and deployment tools.
3. Continuous integration and delivery–Knowing the tools is not enough. You should have substantial knowledge of how and where to use the tools. These tools are critical and should be used to facilitate the continuous integration and continuous delivery of various features of the software.
4. In a few cases, the tools should be used to facilitate continuous deployment; however, it is important to remember that continuous deployment is not considered a good Develop practice.
5. Infrastructure as code (IAC) – this is referred to as programmable platforms. They include terraform and cloud formation in AWS. They are used by both operation and management teams to provide the code and infrastructure rapidly automatically. With the availability of IAC, the distance between the developers and operation teams is getting smaller day by day.
6. DevOps key concept – Always remember that DevOps is not a technology strategy. Instead, it is a methodology strategy. This methodology aims to bring both the Dev and the Operation together to release excellent and quality software in time. If you understand the fundamental concepts, you will be in a better position to provide solutions to various business problems.
7. Soft skills are essential skills that any DevOps must possess. Whenever developers and the operations communicate with each other clearly, they can all collaborate and release the software in time, with better

security and quality. Also, with this kind of collaboration, they can help to market the software that is competitive at a lower cost.

8. DevOps engineers also help to cultivate a good relationship with the businesses and the customers. As a result, those interested in the field need to listen attentively, negotiate, solve problems, and build teams.

DevOps Engineer daily tasks or day to day activities

What are your day-to-day activities?

This question, as per my own opinion, is weak. This is because it matters a lot with the kind of experience that you have as a DevOps engineer and depends on the organization you are working with.

When looking at the company, it boils down to how long your company has been in the process of adopting the DevOps methodology.

However, even though this is the case, there are some generalities in activities that most DevOps engineers perform daily regardless of the size/nature of their workplaces/background/experiences.

Many of the DevOps Engineers execute the following activities daily, and you also fall under this category.

1. Making sure the pipeline is running smoothly is the most critical task in the life of a DevOps engineer and must ensure that the CI/CD pipeline is intact, and he or she should work on fixing any issues facing the CI/CD is the topmost daily priority. DevOps engineers must spend time troubleshooting, analyzing, and fixing the CI/CD pipeline issues.
2. Interaction with other teams – coordination and collaboration is the key concept of the DevOps methodology, and hence the DevOps engineer must ensure a daily integration of DevOps with the QA team, program management IT, and other stakeholders are always required.
3. Working on automation backlogs – Automation is the soul of DevOps, and therefore a DevOps engineer must plan how to enhance automation of various activities daily, meaning he or she must spend much time on the keyboard.

4. Infrastructure management - DevOps engineers are also responsible for maintaining and managing the CI/CD pipeline infrastructure. He or she must ensure that the CI/CD pipeline is up daily and is being executed optimally. This is part of the DevOps engineer daily schedule. The DevOps must work on backups and ensure the system and the tools required to run the CI/CD pipeline are highly available and set up a new environment where necessary.
5. Dealing with legacy stuff – Not every DevOps engineer is lucky to work on the latest and newest systems and tools. DevOps engineers are not exceptional to this. The DevOps must spend time daily supporting or migrating the legacy to the latest version.
6. Exploration – Devops leverage a lot from readily available tools, and there are many options as open-source, and the DevOps team must regularly check on this to make sure that all the adoptions areas required, and all adoptions are up to date. This requires a team effort, and you can say you do this regularly.
7. Removing bottlenecks -You must be able to identify the bottlenecks and manual handshakes and work on them daily. You must work with all the QA, production, and Operation teams to identify and create an automation backlog.
8. Documentations – Although both agile and DevOps stress more on documentation, it is still a vital part of the DevOps engineer to work on documentation on daily basis. Documentation should include server information or just a simple step to configure or modify the infrastructures. DevOps engineers must spend a reasonable amount of time coming up with these artifacts.
9. Training and self-development – training and self-development are essential, and many organizations encourage their employees to take their time out and train and self-develop themselves. The same holds for DevOps engineers, and you must focus on learning something new every day.
10. Continuous improvement as a practice – As a DevOps engineer, it is your daily task to build awareness on potential CI/CD pipeline and DevOps practices and build a culture leveraging it to do things better

by reducing the rework and increasing productivity and optimizing the use of existing resources. You must communicate with the Dev, QA, Production (both staging and live to create an awareness to enhance the culture of continuous improvement).

These are activities a DevOps engineer frequently does and almost on daily basis.

Real Scenarios of day-to-day activities of a DevOps Engineer

When you go to the office, these are the tasks you do

1. Start with your daily standup meeting. In these meetings, your team involved in DevOps such as Dev, QA, staging Prod, and Live Prod usually meet to discuss what needs to be done and how it needs to be done. Let, for instance, say you have a team lead which will happen. During the standup meetings, your team lead will assign you a particular task for the day, and he or she will also take the feedback of your work at the end of the day. **This is the most important and foremost thing you must do in the morning when you arrive at your workplace/office.**
2. You check your emails and Jira tickets. Jira tickets are tasks nothing but just a platform where the information of your daily task is available and not for the entire week. **You use your emails and Jira tickets to know what needs to be done.**
3. You check the notifications from the monitoring systems. Nagios and Splunk are commonly used for continuous monitoring of the IT infrastructure. What happens is that after the deployment of the application in live production, you use tools like Nagios and Splunk to enhance continuous monitoring of the health and security of running applications. If there are any notifications related to running applications, it is your responsibility as a DevOps Engineer to check and fix those notifications if there are any issues. You must always be able to fix or provide solutions to them.

4. You must ensure that you do not leave any critical issue unattended as a DevOps engineer. Any notification alert that is critical regarding the application must be solved, and in case you are not able to do so, you must notify your DevOps team. If you do not have the skills to solve the critical notification alert, you must communicate and find the person who has the skill set to solve the issue.
5. You must communicate with the developers and provide both support and help. Developers are responsible for developing the code as per the application's requirements. As a DevOps engineer, you must talk to the developers and ensure they can write the code elegantly and run the code through a continuous inspection server, which you will help to provision and, if necessary, train developers on how to use the code inspector and scanning tool.
6. Apart from supporting the developers, you must ensure the code move to further stages such as testing integration, deployment, and so on.....
7. Provide support for the production servers – Once the application is deployed on the production servers, it is your responsibility to run the application and whether it is running smoothly or not and whether the application generates their log errors.

Normally, before it is deployed to the live production, the application is first deployed in the staging area of production, where several tests are conducted. The environment in which the application was developed in the Dev (servers) may differ from the production environment (production servers). This mismatch of the environment in the production will cause the application to fail when deployed in production servers. It is your responsibility to check that the environment mismatch does not happen, and all the dependencies issues are solved.

8. Automation of the admin repetitive tasks. As a DevOps engineer, you will always come across many repetitive activities that you must execute almost daily. As a DevOps engineer, it is your work to automate these activities by writing scripts since you are familiar with at least one scripting language. When you write the script and automate

these tasks, you as a DevOps engineer will always be relaxed, and you will have more time to focus on supporting the Dev, QA, and production servers. You will also have more time to talk to the clients and provide better business solutions.

9. You must create a Jenkins job for deployment. Jenkins is a tool used for both continuous integration and continuous delivery phases of DevOps. The basic unit of execution in Jenkins is a job. As a DevOps Engineer, you must integrate the code and prepare a CI/CD pipeline, which is the heart of DevOps, and you must be able to prepare a successful CI/CD pipeline.
10. Support existing infrastructure and create a new one if needed to support your CI/CD pipeline. The creation of new infrastructures usually involves the creation of new servers either on-premises or in the cloud. You are also responsible for maintaining the infrastructure and checking the overall status of your DevOps infrastructure to evaluate whether it is sufficient or not. If not sufficient, talk to the required authorities about the need to expand the infrastructure and, if necessary, provision new servers to solve the issue.
11. Deployment – You must always check whether the code has been deployed successfully on the production servers.
12. You must check for the proper approvals before the code can be released to production and to live production.

Example of a project – Let look at the school website software project

You have a client that wants your company to develop a secure, reliable, tested and highly available software with about 30 features. As a DevOps engineer, it is important to know that you will not develop an application with 30 features all together.

That is not a DevOps recommended practice. This application must be developed in iterations.

You start by attending daily standup meetings involving all the teams. The lead

person in the team will assign you specific tasks daily that must be accomplished to ensure the efficient development of the software.

You will start with supporting developers to develop the code for the agreed five features in the application.

You will also manage the branching strategy and, at the same time, manage the way the code will be committed through version control systems to the existing code in the master branch throughout the entire project.

Continuous code inspection for quality will be implemented. Here your support is required to create the required code quality environment. If the environment already exists, you must evaluate whether the servers are working smoothly to ensure there is no issue with code testing.

You will now ensure the code has moved from Dev to the continuous integration portion of the CI/CD pipeline by cloning the code from the master branch.

You will support the testers in the QA team to carry out the testing of the application using tools like selenium.

You can do so by ensuring the environment the testers in QA are running tools like selenium is sufficient, and if not, you can go ahead and provision servers after talking to the right authority.

Prepare the Jenkins CI/CD pipeline to execute a job

Ensure the correct release process policy is followed to deploy the build or the packaged code or artifact to the staging area of production.

Support the production servers by ensuring the environment is the same as Dev's to avoid mismatch and failure if the application.

After the application has been deployed to live production, as a DevOps engineer, you must continue to monitor the running of the application.

During the running process of the application, you are also gathering feedback, including whether the users are linking the application and how it is running. Also, you check all the 5 features to ensure they all are

running smoothly. If you get any significant changes, you work hard with your team to incorporate those changes in the application, and again you release another version of the application with added features. Let's say this time, 10 features. You repeat the whole process until all the 30 features required for the application are developed, tested (QA), and the Staging environment in production (functional, load testing, user acceptance testing, configuration), and then you monitor the deployment of the new version in live production. Again you get more feedback as you continue to monitor the application, and the process repeats until you are done.

Releasing software in multiple iterations with different features and enhanced feedback in each.

This will result in an application that is

- Quality
- Secure
- Reliable
- Low maintainability cost
- Highly available
- Delivered at a low cost
- Delivered quickly

This is our end of the day-to-day activities of a DevOps Engineer and things you do when you get to the office.