YouTube Summarizer

Transcript:

What if I told you that a DevOps job market is rigged against newcomers? The harsh reality is that entry-level jobs are pretty much nonexistent. In this video, I'll talk about why this is, what this means for aspiring DevOps engineers, and what the future looks like. But first, how did we even get here? So I have a friend who just graduated university with a degree in computer science. Super smart guy. He told me the other day how he's looking to apply to entry-level DevOps jobs in the UK. But all of these so-called entry-level jobs are asking for like three to five years of experience. Understandably, he's feeling a bit frustrated. Now, this got me curious. How do we even get to this point? How is it possible that you hear about such a high demand for DevOps engineers? But there are hardly any actual entry-level DevOps engineering jobs. First off, it's not something that's exclusive to DevOps. It's happening all over the tech industry. In the past, companies prioritized growing their engineering workforce. However, 2020 happened and there was an economic downturn. These same companies then started firing people. And we ended up with mass layoffs from companies like Meta, Amazon, Netflix and Uber. This means that there's more senior DevOps engineers in the job market. Now, put yourself in the shoes of a hiring manager at one of these companies. There's a recession, there's more senior engineers available, and the negotiation power of these engineers is lower since they have fewer options. Are you really going to take a chance on a fresh grad versus someone with experience? But okay, does this mean that there were lots of entry-level jobs in the past? What a whole DevOps field is still relatively new. The DevOps role really only started to take off in the early 2010s. But in that time, the DevOps tooling, technologies and the skills required have gotten way more complex. DevOps engineers are expected to be experts in dozens of different tools and platforms. From CI, CD, pipelines, containers, cloud services, monitoring and more. Most entry-level candidates might have experience with one or two of those areas, like writing code or basic Linux admin tasks. But to qualify for some of these DevOps job listings, you kind of need to be a jack-of-all-trades. For example, one of the listings for a DevOps engineer role in London requires knowledge of SQL, AWS services like SNS, SQS, REST APIs, SIS admins, CI, CD-like Jenkins, networking, data warehousing and microservices. On top of that, you also need the soft skills like collaboration and communication. For a new grad or aspiring DevOps engineer, it's really hard to have practical experience in all of this. University programs or online courses teaches you the theory and fundamentals. But you don't get the same hands-on exposure to the tools and processes used in industry. So it's a classic case of a skill gap between what you learn in school versus what you actually need to know on the job. It then makes sense why companies prefer hiring experience software developers or SIS admins, who already have a background in some of these areas. They can bridge to get more easily and pick up the other DevOps skills as they go along. I read a comment on a Reddit post recently about this. We expect people who are interested in DevOps to have a drive to teach themselves. It is not a job for people who need excessive hand holding. Because a lot of the time, you'll find yourself working on a novel concept and having to think outside the box. And if you need your co-workers to do all the thinking for you, you just become dead weight. I think this comment makes a lot of sense. DevOps is a very fast-paced industry. A DevOps engineer needs to be able to learn a new tool or technology very quickly or solve problems that are unique to their task. A co-worker may not have encountered that specific problem before, and it might not be anything available online. So the people that do the best in these roles are going to be people that are able to figure things out for themselves. When you're a fresh graduate or entry level, it's very hard to demonstrate a skill compared to an experienced software engineer or IT admin. So when you put it all together, it's kind of the perfect storm. A relatively new and rapidly evolving field, an increase in the supply of senior engineers, and increasingly high expectations for skills. The end result is that it's very difficult to find an entry level DevOps job. And look, I get it from the employer's perspective. DevOps is critical to their business, and they can't afford to take chances on inexperienced new grads. The reality is, it's just much more expensive and time consuming for companies to hire junior engineers and train them from the ground up. But what does this mean for new DevOps engineers? The core issue is that companies are reluctant to hire someone who has never practiced DevOps in a real production environment before. So as an aspiring DevOps engineer, how can you demonstrate your skills and show potential employers that you have what it takes? Well, I think it really comes down to two things. On the technical side, you want to be confident in your understanding of the fundamentals. Well, what are the fundamentals? Well, I like this beginner's roadmap on roadmap.sh. I don't necessarily agree with the order that they've listed. And for some companies, certain topics might be more important than others. But this is a great starting point. It's important not to focus all of your attention on specific tools. Instead, try to understand why they are used and what problems they solve. For example, for CICD, they've listed GitHub Actions. And yes, it's a great beginner-friendly tool. But try not to spend all of your time studying GitHub Actions and think broadly about how CICD is used, where it's used, and why. These tools vary from company to company and also change a lot over time. But the underlying skills and knowledge are transferable. You also want to get practical hands-on experience with these technical skills, wherever you can. Contributing to open-source projects is a great way to do this. Find a project that interests you and start making pull requests. Fix bugs, add new features, and improve the documentation. You'll learn a huge amount and have some impressive talking points in an interview. Building your own personal projects is another great option. Take a simple app or service that you can build and try to devopsify it. A good example for this is the Cloud Resume Challenge. Although it's focused mainly on AWS skills, you can still add a lot of devops-related features. For example, automate the deployment, build a CICD pipeline for it, or add monitoring and logging. Then write up a case study or blog explaining what you've learned. Not convinced? On the Cloud Resume Challenge website, you can find the testimonial from Brooke, who switched from Verizon analysts to associate devops engineer, just one to two months after completing the challenge. I relied heavily on the Cloud Resume Challenge during my interview. But technical skills alone won't get your job, unless you also work on this. So much of devops is about breaking down silos and getting different teams and roles to work together smoothly. Companies want to see that you understand and embrace the devops philosophy. And the best way to show this is by using your soft skills. You want to demonstrate that you can communicate and collaborate effectively. But how? Unlike learning technical skills, there isn't a defined pathway. A lot of this involves actively seeking out opportunities. For example, writing clear and concise documentation in your personal projects, volunteering for a local meet-up or conference, or offering to give a lunch and learn session at your current job. Anything that shows you can work well within a team and communicate effectively. But I think the most important thing you can do is contribute to discussions in the devops community. You can do this online, but I would suggest trying to find local communities and meet-ups in person. You can use sites like Meetup.com or look around in Facebook groups in your area. Not only does this show employees that you're serious about learning devops, it also builds your personal brand and widens your network. But is it even worth doing all of this? Would the market continue to be like this? What does the future look like for devops engineers? Look, it's not all doom and gloom. Although the current market heavily favors experienced candidates, I don't think it will always be like this. The irony is that as more and more companies adopt devops, the demand for these roles keeps growing. But the amount of experience devops engineers can't keep up with this demand. Soon enough, I think companies will be forced to try and recruit and train entry-level engineers, with the hope that they'll eventually turn into senior engineers and fill their open positions. Otherwise, the devops skill gap will just keep increasing. Another trend that I think makes the future of devops exciting is how it's expanding into other areas. We're seeing devops principles being applied in areas like security with devs' setups, data science with data ops, machine learning with ML ops. So the opportunities aren't limited to just one niche. But what about AI? We've already got intelligent systems that can provision and configure servers, monitor applications for issues, and even self-heal when problems arise. It's pretty wild when you think about it. So does this mean that devops jobs are going to disappear? I don't think so. But I do believe the role is going to evolve quite a bit. In self-spending time on manual repetitive tasks, devops engineers will likely shift to more of an oversight and architecture kind of role. They'll be responsible for designing the overall devop strategy, choosing the right tools and platforms and ensuring everything integrates properly with AI. Of course, no one knows exactly how things will play out. But I do believe that devops as a practice is here to stay. We just look a bit different than it does today. So if someone asks me, should they even bother learning devops? I'd say definitely yes. Those skills are hugely valuable. But I'd encourage them to focus more on the fundamentals, the operating systems, the networking, the core computer science concepts. Because the specific tools and platforms will evolve, but the fundamentals will always be relevant. On top of this, try to meet people and network as best as you can. But you need to consider whether or not you even want to devops job in the first place. You've probably heard about all the positive aspects posted online. But it's not all glitz and glamour. There is a dark side, and you can find out the harsh reality of being a devot engineer in this video.

Summary

DevOps engineers are expected to be experts in dozens of different tools and platforms . To qualify for some of these DevOps job listings, you kind of need to be a jack-of-all-trades . For a new grad or aspiring DevOps engineer, it's really hard to have practical experience in all of this . We expect people who are interested in DevOps to have a drive to teach themselves. It is not a job for people who need excessive hand holding. And if you need your co-workers to do all the thinking for you, you just become dead weight .