5 ways to become a platform engineer

The transition to platform engineering is a knowledge-critical shift in modern IT that calls for a high demand for talent because it combines software development automation with the system reliability from infrastructure and developer operations (DevOps). Platform engineers act as force multipliers by creating self-service tools and internal developer platforms or "golden paths" that eliminate friction and help other developers build applications more safely and in less time.

1 Learn the infrastructure engine of modern applications

Platform engineering is built on the promise of infrastructure abstraction, and to be successful you must fully comprehend the fundamental technologies that power modern applications.

- **Skills to advance:** Expertise in containerization with Podman and orchestration using Kubernetes. Learn how to deploy an application, how the scheduler works, how network policies are enforced, and how storage volumes are provisioned.
- **Supporting technology:** Install Red Hat® OpenShift®, an enterprise application platform with an integrated ecosystem, for the entire software lifecycle.
- **Hero proposal:** Propose a migration path for legacy services to run on a standardized application platform, eliminating standalone workstation application issues across the development team.

2 Automate the valuestream with CI/CD and GitOps

A platform's primary job is to provide a consistent, self-service pathway from code commit to production. This job requires expert-level proficiency in continuous integration and continuous delivery (CI/CD) and GitOps.

- **Skills to advance:**
  - Treat infrastructure and deployment pipelines as code.
  - Learn Kubernetes-native CI/CD, GitLab continuous integration.
  - Demonstrate a level of mastery in GitOps principles.
- **Supporting technology:** Implement reliable, declarative CI/CD using Red Hat OpenShift Pipelines (Tekton), and integrate them with the wider Red Hat Advanced Developer Suite for supply chain security and developer self-service.
- **Hero proposal:** Introduce a single, standardized pipeline template for teams to use, reducing boilerplate and ensuring consistent security and quality checks across the organization.

3 Champion developer experience

The best internal platform is the one developers want to use. Your IT team's success hinges on providing a fantastic developer experience (DX).

- **Skills to advance:**
  - Design thinking, application processing interface (API) design, and user-friendly interface creation.
  - An advanced knowledge of the Backstage framework to implement and how to operate it through Red Hat Developer Hub.
  - Red Hat Developer Hub, a fully supported and integrated enterprise offering for creating internal developer platforms.

- **Supporting technology:** Focus on self-service automation scripts (Bash, Python) and configuration management such as Red Hat Ansible® Automation Platform. The goal of supporting your automation technology is to allow a developer to provision an entire environment (database, monitoring, application scaffold) with a single select or command.
- **Hero proposal:** Design and implement a golden path for new services. Use a documented, opinionated, and fully automated workflow that can handle the IT environment and can reduce new service setup time from days to minutes.

4 Adopt an observability mindset

A stable platform is an observable platform. Shift your focus from logging errors to understanding your organization's system health, performance, and internal user behavior in real time.

- **Skills to advance:** Implement the 3 pillars of observability (metrics, logs, traces). Become proficient with tools for collection (Prometheus, Fluentd) and visualization (Grafana, Jaeger).
- **Supporting technology:** Develop skills in defining service level objectives (SLOs) and service level indicators (SLIs). Understanding how to measure platform reliability is key to demonstrating its value.
- **Hero proposal:** Build a universal dashboard template available for every new service that automatically surfaces key operational metrics and SLO adherence, allowing IT developers to debug and monitor their services without involving an operations team.

5 Propose and document the platform

The most crucial step is articulating the value of the platform to leadership. Platform engineering is a business strategy that treats the internal developer platform (IDP) as a product and the developers as its primary customers.

- **Skills to advance:** Develop a successful business case by quantifying the platform's return on investment (ROI) (e.g., reduced deployment lead time) and qualifying the organization's service ownership.
- **Supporting technology:** Use documentation tools such as MkDocs and Confluence to enhance presentation skills. The IT platform can achieve its full value if your organization's internal customers know about its capabilities, find the interface intuitive, and understand the golden path to adoption.
- **Hero proposal:** Lead a dedicated platform product team, present a concise roadmap to accelerate business outcomes, and position yourself to advance this initiative.

Discover the benefits of becoming a platform engineer from the experts

Visit Red Hat's platform engineering page to learn more about how Red Hat OpenShift and Red Hat Advanced Developer Suite can provide platform engineering solutions for your organization's entire software development lifecycle.

redhat.com
#3358904_0126