Platform as a service

Previously, you learned how cloud service providers (CSPs) offer varying service models to meet users' workload needs. Each service model has a different degree of shared responsibility between the CSP and user. For example, you learned how in an infrastructure as a service (IaaS) model, the CSP provides only the underlying infrastructure for a cloud environment.

In this reading, you'll learn about platform as a service (PaaS), an organization's and CSP's shared responsibilities with PaaS, and examples of why organizations adopt PaaS.

Components of PaaS

Platform as a service, or PaaS, is a cloud service model where the CSP hosts and maintains the backend hardware and software that applications use to operate, while developers use these resources to write code and build and deploy their own applications.

PaaS includes the CSP's underlying infrastructure, middleware, and user interface. Middleware is software that facilitates communication and interaction between services and applications.

Let's break down each of these concepts:

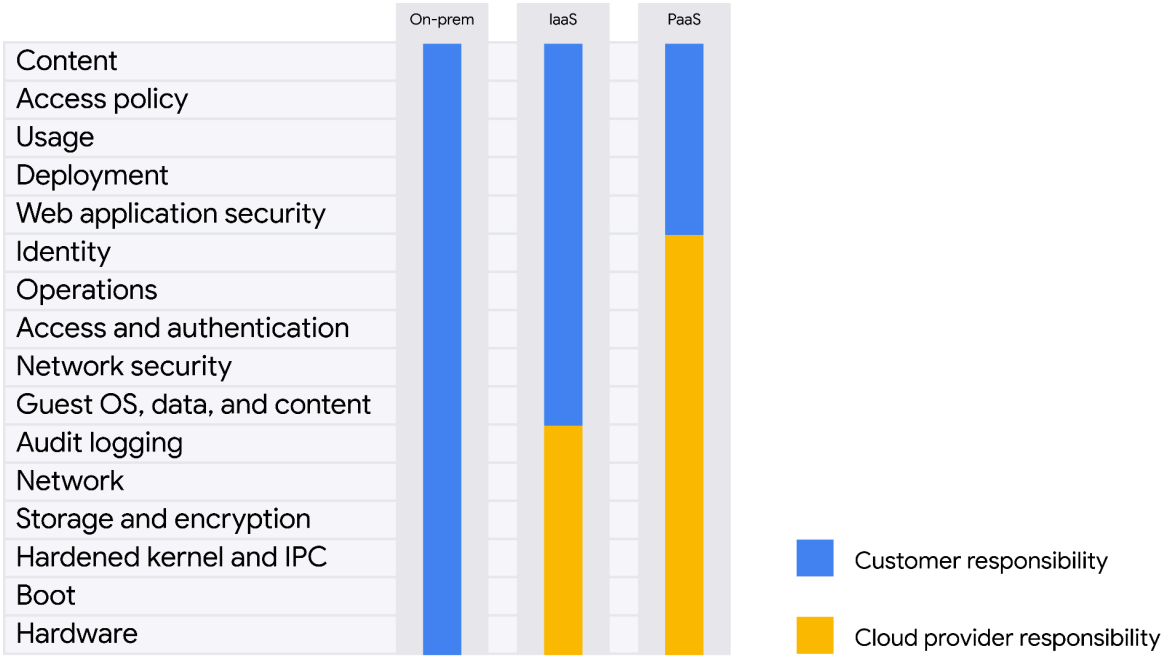
- Infrastructure: includes underlying physical infrastructure like data centers and servers
- Middleware: includes software like operatings systems and code libraries
- User interface: includes a graphic user interface (GUI) or command-line interface (CLI)

The combination of these tools and services enable developers to build and deploy applications quickly. These cloud tools also make PaaS scalable. Developers can add or subtract resources on demand as needed, and organizations only pay for the resources they use.

PaaS shared responsibilities

Like with other cloud service models, shared security responsibility is also a component of using PaaS. With PaaS, the CSP is responsible for more security components than in an IaaS model or on-premises model. You may recall that IaaS provides access to underlying infrastructure services like compute, storage, and networks. In on-premises environments, customers are responsible for securing all aspects of their infrastructure.

In a PaaS model, the CSP takes more control over software resources needed to develop applications. For example, the CSP provides the underlying infrastructure, plus operating systems, network security, and access and authentication. This leaves customers responsible for data, access policies, deployments, and web application security.



PaaS use cases

PaaS is commonly used for building applications. Developers can use the provided suite of tools to develop applications, like those commonly accessed through a website or mobile device.

PaaS can also be used as part of a hybrid cloud migration strategy. For example, when an organization is migrating to the cloud using a hybrid cloud model. You may recall that in a hybrid cloud model, an organization uses cloud infrastructure for parts of their business, but still retains some resources privately on-premises. PaaS is useful as part of the migration process since it enables developers to use cloud infrastructure while remaining in control of their application development.

Key takeaways

PaaS is a cloud service model that is ideal for developers. Cloud resources are scalable, allowing developers to experiment and only pay for the resources they use. Developers can take advantage of the CSP's predetermined security controls for infrastructure and networks,

while still maintaining control over their applications and code and who can access them. PaaS is one of the core cloud service models, so it's important to understand the distinction between the CSP's and user's security responsibilities. As a security professional, your organization may use PaaS to develop software, and an understanding of this service model will help you secure your organization's resources.

Resources for more information

The following resource provides more information about PaaS:

- Google Cloud's description of PaaS