Public, Private, and Hybrid Cloud: What's the difference?

Cloud Hosting

Trying to understand the cloud hosting landscape can be a nightmare. The internet is full of blog posts, whitepapers, eBooks, and marketing sites about cloud tools and services. This makes it hard to figure out where to begin. As an enterprise software developer or development manager, understanding the environment your applications will be deployed in is critical, because it can impact everything from the tools and libraries you use to your application's architecture.

To help cut through the noise, we'll start at a high level and examine three major categories of cloud services: public, private, and hybrid. We'll look at what each one is, what the differences are, and how they vary from the perspective of application development and deployment.

As an enterprise software developer or development manager, understanding the environment your applications will be deployed in is critical.

Public Cloud

Public cloud services are what usually come to mind when discussing 'the cloud'. In public cloud services, *everything* runs off-site, on shared physical hardware operated by the cloud provider. Public cloud providers include widely known services such as Amazon Web Services, Microsoft Azure, and Google Cloud Platform, as well as other providers like Rackspace, Softlayer, and Hostway.

At a minimum, each provider offers virtual machines (VMs) running Windows Server or Linux. Developing self-contained applications that can run on a cloud VM isn't much different than developing applications that run on traditional on-premise servers. Each VM is capable of running nearly any application that can run on a physical server. Developers will have to pay extra attention to security, however, because traffic to your public cloud applications will have to traverse the internet. Many cloud providers offer secure solutions that allow you to treat your cloud applications as if they're part of your corporate network; Azure Virtual Network and AWS Virtual Private Cloud are both good examples. You won't get virtual networks by default, though; you must explicitly set them up, and then ensure your applications are deployed to VMs within the virtual network.

Public cloud VMs have the advantage of being easy to scale as an application's usage grows. Migrating an application that outgrows the physical server it is installed on can be painful. Resizing a cloud VM is a relatively pain-free process. Additionally, most cloud providers offer a block storage service that can be easily detached from one VM and attached to another. Examples of block storage include AWS Elastic Block Storage, Azure Drives, and Google Cloud Persistent disks. From a developer's perspective, block storage is usually transparent. The operating system sees block storage as an attached drive, so as long as the application doesn't depend on being deployed to a specific drive, application developers shouldn't have to worry about it.

Life in the public cloud becomes more difficult when you start using proprietary services that are specific to a single provider. A good example of this is AWS S3 storage, which can be used for storing static website assets and other files that don't change often. While other cloud providers have similar services, they each have a different API, and once you have developed an application that uses one of them, it becomes difficult to move your application to another platform if it ever becomes necessary.

Public cloud offerings also include Platform-as-a-Service (PaaS) vendors, such as Heroku, Google App Engine, and the Azure Web App Service. These products are great if you want to be able to deploy and scale your applications easily without having to devote any do any system administration. Deploying new apps usually only requires making a commit to a git repository. The downside to PaaS is that your application usually must be tailored to the requirements of the platform you're deploying to. Moving to a new platform isn't impossible, but may require you to rewrite significant parts of the application you are trying to move.

Private Cloud

Private cloud environments try to offer the scalability and ease-of-use of public cloud services while giving enterprises control over the hardware and software used to run their applications. This approach can be attractive to security conscious companies who are looking to modernize their approach to application deployment without trusting a third party to keep their applications and data secure.

For application developers, private cloud hosting is as easy to use as public clousing hosting - in most cases, application deployment and scaling is easier than in traditional on-premise server environments. Some public cloud providers also offer private versions of their environment that can be deployed in-house; the Azure Stack is a good example of this, and several vendors offer private cloud solutions based on OpenStack.

Private cloud offers a greater degree of control and certainty. While public cloud providers usually have good uptime SLAs, when things go wrong there is usually no insight into what is causing the problem, or how quickly it can be resolved. Although SLAs provide some recourse after the fact when downtime occurs, the compensation usually isn't enough to cover your losses. And no SLA can fix the angry users you'll have to face after service outages.

The downside to private cloud solutions is that they are are expensive to deploy and deliver. By moving to public cloud solutions, companies can reduce or eliminate the number of servers they must purchase and maintain in-house. Servers expensive to buy, and ongoing support is expensive due to the personnel and physical space required to run a datacenter.

Scaling in a private cloud is also more difficult. When using a public cloud provider, the ability to scale an application quickly is usually only limited by a company's ability to pay for the cloud services needed. In a private cloud environment, the ability to scale applications is limited by the quantity and power of available servers. While new servers can be added to increase the private cloud's capacity, doing so is expensive and relatively slow.

Hybrid Cloud

Hybrid cloud environments attempt to bridge the gap between public and private cloud, offering some of the advantages of both.

Many companies have some sensitive applications and data that absolutely cannot be run in a public environment, but also have other applications without this restriction. In these situations, a hybrid cloud is often an ideal solution. Companies can run their security sensitive applications in the private part of their hybrid cloud solution, and reduce their hardware and IT staffing costs by running most other applications in the public part of their hybrid cloud. This can usually be done in a way that is transparent to end users so they don't have to know anything about where the application is running.

Hybrid cloud environments are usually more complex for developers than fully public or fully private clouds. Developers have to keep in mind which part of the hybrid cloud their applications will be running in, and need to consider scenarios where a private application needs information from a public application. When this is required, developers must take steps to ensure that private applications do not leak any sensitive data to applications running in the public part of the cloud. A large part of this is governance related; it isn't always obvious which data must remain private at all times, so it's important to meet with stakeholders throughout the company to clarify security requirements.

Despite the additional complexity hybrid cloud environments do offer, for many companies, the best of both worlds. Highly secure applications and sensitive data can remain on-premise, and less sensitive applications and data can be offloaded to the public cloud to reduce costs.

Conclusion

As we've seen, moving to the cloud has some significant advantages - and some drawbacks. There's no one size fits all solution. The type of cloud hosting that is best for your company depends heavily on the needs of your users and your applications. It's important to do a thorough analysis of an application's performance, storage, and security requirements before deciding how to host it.

Links

Cloud Providers

Amazon Web Services - https://aws.amazon.com
Microsoft Azure - https://azure.microsoft.com
Google Cloud Platform - https://cloud.google.com

Rackspace - https://www.rackspace.com
Softlayer - http://www.softlayer.com
Hostway - https://hostway.com