Containers vs. Unikernels

The Key Differences, Plus How to Improve
your IT infrastructure and Reduce Costs

nanovms

When it comes to improving your IT infrastructure, there is often confusion between containers and unikernels. There are key differences between the two when it comes to performance, security, and cutting down on infrastructure costs. Let's take a look at the two technologies to discover which is the best choice for your company.

Performance

CONTAINERS

Traditional software stacks require the generic OS so the application can draw needed functions at runtime.

Containers are a type of deployment. A container is a piece of software that includes everything required to run it, but requires placement on top of a generic operating system (OS), such as Linux.

UNIKERNELS

Unikernels differ from containers because they do not require a normal OS at all. Unikernels build the necessary functions from the OS in at the time of compilation, rather than during startup. This results in a single image that contains only what the application requires to run properly, leaving out the unnecessary libraries, utilities, etc. that are typically included. In other words, they simply require the application code and only the software from an operating system needed to run that application, making them extremely lightweight and agile.

nanovms

Security

CONTAINERS

Containers require add-on security. The add-ons are required to run the bulky operating system that containers require to run. While these security measures are useful, they are not strong enough for use over the internet and leave your data vulnerable to hackers.

In order to hack a unikernel, one must be smart enough to break the app without any tools, like using a shell, utilities programs, or password files... because there aren't any in a unikernel.

UNIKERNELS

Unikernels are more secure for several reasons:

- Unikernels minimize the size of an application to a fraction of its usual surface.

- Unikernels do not use shells (the program that allows ad-hoc commands from the user to the OS to Unikernels minimize the size of an application to a fraction of it's usual surface.

- Unlike the Linux system, unikernels use an incredibly small amount of code, meaning there is less code that can be exploited.

- Unikernels don't have users, meaning hackers cannot login and execute arbitrary code.

- Since unikernels can run only a single program, they simply cannot run code that was not intended to run. This drastically cuts down on security issues.

nanovms

Infrastructure Costs

CONTAINERS

Clouds have become the desired infrastructure method of IT. But the problem with common cloud systems is that they are overloaded with software and heavy disk images. All of these images weigh down the cloud, requiring more host servers. More storage means more resources are required, including electricity, more data center room, and more money.

UNIKERNELS

Unikernels streamline these needs. Minimal resources are required to run them, significantly cutting down on the start time and efficiency of cloud servers as well as the amount of resources required to turn them.

That means with unikernels, you can effectively reduce infrastructure storage and save money thanks to reduced power requirements.

Ready to reduce costs, improve security, and streamline your IT?

NanoVMs can help.

Unikernels are widely considered to be the next generation of cloud infrastructure for their speed and security, and at NanoVMs, we specialize in them. To schedule a demo or contact us to see how unikernels can improve your company's infrastructure, visit our website or give us a call.

https://nanovms.com/

(833) NANO-VMS

SCHEDULE A DEMO

Containers vs. Unikernels

nanovms