nanovms

OpenStack vs Unikernels

OpenStack

OpenStack brought cloud computing to the masses. The traditional model of I.T. service delivery was not working. Provisioning resources for new projects and new deployments was slowing down development teams. Cloud computing was born out of market pressures to bring new products and new iterations of those products into production faster. The open source suite of cloud tools known OpenStack offered a ready solution.

With compute, network, storage becoming commoditized and the open source tools to manage them easily available, organizations were able to bring new products and new features to market faster, and release bug fixes and security updates with more regularity. DevOps teams were formed which shared responsibility for Development and Operations tasks. With much of Operations now automated, the new DevOps is more Dev than Ops.

A lingering Dev challenge was that often when code was moved from the dev environment into production, there were dependency errors. Containers solve this problem by bundling the applications and all its dependencies including libraries and frameworks. Containers share a host OS.

The OpenStack tool set is based on the Linux operating system. The Linux kernel was developed before virtualization, when multiple users were the norm. It is this multi-user design which renders it vulnerable to security breaches. Within the last few years, high-profile security breaches have dominated the news. Some have resulted in extensive loss of downtime and fines.

Unikernels

Unikernels are a technology which solves some of the problems containers solve even more effectively while providing greater security. While Containers access a shared OS kernel, a unikernel includes within it the portions of the kernel which is required to run the application. Unikernels are much smaller than containers. The application doesn't need to load the OS kernel each time it is run, so runtimes are much faster.

OpenStack

Pros

- Self-service
- Open Source software is less expensive, updated frequently
- Open Source so lots of people are discovering vulnerabilities and working on patching them

Cons

- Built on the Linux kernel which is designed for multiple users and has inherent security vulnerabilities
- Security solutions are bolted on and plug specific holes but there will continue to be exploits
- Development environment different from production environment; containers attempt to solve this

Unikernels

Pros

- More secure
- Smaller and faster runtime
- The unikernel contains everything needed to run it. Dependencies are all contained and cross-compiled within the unikernel

Cons

- Every change to code has to be re-cross-compiled and redeployed into production environment (this is fast with new orchestration tools)
- Required access to orchestration tools for compiling means a change to workflows
- The public cloud providers services are not designed for unikernels. They can be an excellent option for private cloud deployments

Thousands of unikernels can share the same physical server.A big reason to consider unikernels is that by design they prevent remote code execution. Their single process design foils the shell code exploits typically employed by hackers to gain access to systems.

Unikernels are not new. The cross-compilation process to create a unikernel used to be a time-consuming, manual one. The deployment and management of unikernels was also problematic. Until recently, unikernels were not generally thought to be suitable for mission-critical enterprise applications. However, a new generation of orchestration tools have brought unikernels into a new era.

Unikernels are especially suited to applications in the financial services, healthcare, life sciences, and government sectors. The pricing and deployment models of public clouds make them less than ideal for using unikernels. Unikernels are, however, a great fit for private clouds.

NanoVMs is the Enterprise Services Unikernel Company. NanoVMs is the only production ready, fully managed unikernel platform in the industry today. We save companies money on infrastructure and ops cost while at the same time taking real proactive security measures to limit attacks.