

# Jeison Vergara Vargas

DOI: 10.29236/sistemas.n168a3

*Profesional con amplia experiencia para abordar diferentes asuntos relacionados con el “Paradigma Serverless”.*

Sara Gallardo M.

En su amplio recorrido como arquitecto de software en los sectores público y privado del país, Jeison Vergara Vargas ha dejado huella. Así mismo en el exterior como Doctorat en Informatique et Architectures Numériques, Université Bretagne Sud, Francia; profesor visitante en Université de Poitiers, Francia (2017) y profesor visitante, EPI-TECH París, Francia (2019).

Considera una fortuna ser egresado de la Universidad Nacional de Colombia, en donde se desempeña como profesor auxiliar del Departamento de Ingeniería de Sistemas e Industrial, sede Bogotá. Además, es investigador doctoral del laboratorio IRISA (Institut de Recherche en Informatique et Systèmes Aléatoires) en Francia.



Sus áreas de interés contemplan arquitectura de software, lenguajes de dominio específico, métodos formales e ingeniería de software dirigida por modelos. Así mismo es líder de la línea de investigación en Arquitectura de Software del Grupo de Investigación en Ingeniería de Software (CoISWE).

A las actividades profesionales les suma un permanente deseo por conocer diferentes culturas del mundo y su gastronomía, combinado con otras lecturas sobre arte, astronomía y un espacio para la natación y los juegos de mesa.

**Revista Sistemas:** ¿Cómo se podría materializar el concepto de ciberseguridad en el paradigma "serverless" cuando sólo existen "funciones" que se usan? ¿Sería sólo lo correspondiente al desarrollo de aplicaciones o funciones seguras?

### **Jeison Vergara Vargas**

La computación tipo «serverless» es un modelo basado en la nube, en donde los recursos de infraestructura (cómputo y red, principalmente) son gestionados por el proveedor del servicio, ocultando su identidad a los clientes. En este modelo, los clientes no tienen la responsabilidad de la creación, configuración y mantenimiento de estos recursos de infraestructura.

La ciberseguridad es un área que busca garantizar con énfasis la confidencialidad, la integridad y la disponibilidad de los recursos manejados desde una perspectiva computacional: datos, software y hardware. Independientemente del uso del modelo *serverless*, la seguridad mantiene retos que abarcan, entre otras cosas:

- Seguridad a nivel de la arquitectura del software: diseño estructural del sistema, seguridad a nivel de componentes y conectores.
- Seguridad a nivel de diseño detallado del software: aspectos de seguridad durante la codificación de los componentes del sistema, incluyendo FaaS (funciones como servicio).

- Seguridad a nivel de flujos de datos: aspectos de seguridad cuando los datos viajan de un componente a otro, o de un sistema a otro.

**RS:** *Se ha venido trabajando en los últimos años en arquitecturas basadas en microservicios como estrategia arquitectural para que las soluciones sean más escalables y flexibles. ¿Cómo encaja una arquitectura serverless en esta estrategia?*

**JVV:** Las arquitecturas de software basadas en microservicios se caracterizan por su estructuración a partir de un conjunto de componentes lógicos; es decir, componentes con responsabilidad de lógica de negocio dentro de un sistema de software, los cuales se implementan por medio de algún lenguaje de programación de propósito general. Estos componentes son pequeños y poseen una responsabilidad funcional única. En este sentido, dado que el modelo *serverless* promueve el uso de funciones como servicio, estas funciones pueden ser usadas para la implementación de componentes de tipo microservicio, por lo cual se convierten en una alternativa bastante atractiva para el diseño de arquitecturas de software distribuidas y basadas en la nube.

**RS:** *¿Qué opina acerca de la dependencia que se genera entre la arquitectura de un sistema bajo el paradigma *serverless* y los proveedores de soluciones de nube?*

**JVV:** La dependencia que se pueda generar es relativa. En el caso de un buen diseño arquitectónico, dado que la arquitectura de un sistema debe ser agnóstica a la tecnología, la dependencia se puede reducir. Una de las formas más comunes es generando redundancia entre los componentes. De esta forma, un componente puede estar ejecutándose como una función en el proveedor A y, a su vez, como otra función en el proveedor B. Las dos funciones son equivalentes, pero se ejecutan de manera independiente.

**RS:** *¿Es posible no tener que "cajarse" con un proveedor de nube específico para una solución bajo el paradigma *serverless*?*

**JVV:** Sí, existen varias alternativas a partir de variaciones en el diseño de la arquitectura del sistema. Un ejemplo es el descrito en la pregunta anterior.

**RS:** *¿Cuáles son las mayores ventajas del paradigma *serverless* con respecto a los paradigmas tradicionales?*

**JVV:** Definitivamente la agilidad en la construcción del componente.

**RS:** *Los desarrolladores de software ya no tendrían el control del uso de los recursos de hardware en el paradigma *serverless*, por lo cual, los atributos de calidad arquitecturales relacionados con desempeño ya no son gobernados por los desarrolladores y arquitectos. ¿Cuáles son los retos a los que*

*deben enfrentarse quienes construyen soluciones de software bajo este nuevo concepto?*

**JVV:** Hay varios retos a tener en cuenta a la hora de diseñar y construir sistemas de software bajo el modelo *serverless*, entre ellos:

- Seguridad durante la ejecución de componentes basados en funciones.
- Gestión eficiente en el uso de recursos proveídos por el proveedor de nube.
- Proceso de pruebas adecuado: no se tiene control sobre pruebas de nivel 3 (sistema – no funcionales).
- Latencia: tiempos de respuesta variables.
- Manejo de estado.

**RS:** *¿Este paradigma *serverless* es una evolución normal de la arquitectura de software? ¿O es una "moda"? ¿Cómo se ve este nuevo paradigma ante los ojos de la comunidad académica de arquitectura de software e ingeniería de software?*

**JVV:** No, el modelo *serverless* no corresponde a una evolución normal de la arquitectura de software.

La evolución de la arquitectura se fundamenta sobre cambios a nivel estructural; es decir, a nivel de los componentes y conectores que conforman el sistema, y cómo estos se comunican entre sí; no en términos de cambios a nivel de despliegue. El uso del modelo *serverless* no tiene trascendencia a nivel estructural, pero sí a nivel de despliegue.

**RS:** *¿Qué tipo de soluciones de software se ven beneficiadas de este nuevo paradigma, ¿cuáles no?*

**JVV:** Sistemas de software de procesamiento de datos en tiempo real, tareas en segundo plano, algoritmos de inteligencia artificial que automaticen procesos. Sistemas que requieran distribución en su arquitectura (microservicios y brokers de mensajes). 🌐

**Sara Gallardo M.** Periodista comunicadora, universidad Jorge Tadeo Lozano. Ha sido directora de las revistas Uno y Cero, Gestión empresarial y Acuc Noticias. Editora de Aló Computadores del diario El Tiempo. Redactora en las revistas Cambio 16, Cambio y Clase Empresarial. Coautora del libro "Lo que cuesta el abuso del poder". Ha sido corresponsal de la revista Infochannel de México; de los diarios La Prensa de Panamá y La Prensa Gráfica de El Salvador y corresponsal de la revista IN de Lanchile e investigadora en publicaciones culturales. Se ha desempeñado también como gerente de Comunicaciones y Servicio al Comensal en Inmaculada Guadalupe y amigos en Cía. S.A. (Andrés Carne de Res) y editora de Alfaomega Colombiana S.A.; en la actualidad es asesora y editora en escritura y producción de libros. Es editora de esta revista.