techworld FROM IDG Insights

Issue: **January 2019**

THE COMPLETE GUIDE TO SERVERLESS COMPUTING

Credit: iStock

Produced by IDG

Credit: iStock

Cloud cover

Serverless computing is a cloud computing methodology which appears to be gaining great momentum in the market, particularly among Amazon Web Services (AWS) customers, as the vendor is extremely bullish on the architecture.

In short, serverless computing is the idea of further abstracting infrastructure concerns away from developer's lives, helping them focus on their applications and not provisioning or maintaining servers. It relies on stateless functions to execute code automatically in the cloud and allows the end user to only pay for the compute they use.

AWS pioneered serverless with its Lambda function-as-a-service offering back in 2014, but that element has since been replicated by all the other major cloud platforms, with Google Cloud Functions, Microsoft Azure Functions, IBM/Apache's OpenWhisk, and Oracle Cloud Fn.

This guide will focus on serverless from a predominantly AWS lens, primarily because they are the vendor most willing to talk about it and, most importantly for us, to put actual customer stories out into the wild. Here we run through the basics of a serverless architecture, its trajectory in the market and a case study from a very early adopter. **Scott Carey**

Contents

4 Serverless computing and the firms using it

10 Trustpilot's 'serverless first' approach to engineering

Credit: iStock

Serverless computing and the firms using it

Serverless computing has been fast gaining momentum over the past couple of years, with AWS in particular talking up enterprise adoption

Serverless computing is an architecture where code execution is fully managed by a cloud provider, instead of the traditional method of developing applications and deploying them on servers. It means developers don't have to worry about managing, provisioning and maintaining servers when deploying code. Previously a developer would have to define how much storage and database capacity would be needed pre-deployment, slowing the whole process down.

What are functions?

Serverless relies on functions, or more specifically functions-as-a-service, where developers break down their applications into small, stateless chunks, meaning they can execute without any context regarding the underlying server.

One of the most popular function-as-a-service offerings is AWS Lambda from the market-leading cloud vendor Amazon Web Services (AWS). Launched all the way back in 2014, Lambda allows developers to do just this: run code without provisioning or managing servers. AWS charges you for the compute power you use according to 100-millisecond increments. Developers can therefore focus on their code and event triggers, and AWS takes care of the rest.

Events could include changes to data in an Amazon S3 bucket or an Amazon DynamoDB table; in response to HTTP requests using Amazon API Gateway; or using API calls made using AWS SDKs. For example, when a user requests a car on a ride sharing app this could trigger the code which is written to fetch a car, or clicking the 'buy' button on an app will trigger that buying process.

"Lambda is an event-driven execution environment," explains Ian Massingham, chief evangelist at AWS. "So in very simple terms you have code and events, when the event arrives the code is executed for you automatically. So you don't have to pre-position resources or have any standing infrastructure to provide the execution environment."

He adds that when the event is triggered, a piece of infrastructure is allocated dynamically to execute the code: "What happens under the covers is a Linux container is started on a machine and details

– metadata about the event – is passed into the container at the point of execution."

This is opposed to even the 'traditional' deployment model within AWS itself, where "EC2 [Elastic Compute Cloud] instances run web or app servers that wait around for requests and when they come they service them. That gives you floor costs, with Lambda your cost of execution with no traffic is zero and as you start to get traffic you scale up. It is way more cost-effective at low levels of usage and way more scalable at high levels of usage, so benefits at both ends of the scale."

Massingham says that it is important to note that Lambda doesn't automatically equate to serverless, however. "Lambda is the execution part and serverless is a little bit bigger," he says. "Beyond compute you also want to run things like your data stores in a way which doesn't require you to operate infrastructure, a way to do identity management that doesn't require you to operate infrastructure."

Amazon CTO Werner Vogels used an old favourite metaphor during his 2016 keynote: "Before, your servers were like pets. If they became ill you had to nurture them back to health. Then with cloud they were cattle, you put them out to pasture and got yourself a new one. In serverless there are no cattle, only your application. You don't even have to think about nurturing back to health or getting new ones, all the execution is taken care of."

Serverless isn't just available to AWS customers, though. IBM OpenWhisk is an alternative event-based architecture. Google has Cloud Functions for "developers to create single-purpose, stand-alone functions that respond to cloud events without the need

to manage a server or runtime environment". Similarly, Microsoft has Azure Functions.

Oracle also announced the serverless Fn project at the JavaOne 2017 conference. In his blog post on the subject Johan Vos, co-founder of Gluon and LodgON, writes: "One of the key characteristics of Fn is that although it is intended to run in cloud environments, it is not tied to a specific cloud vendor. The platform itself can be hosted on any cloud environment that supports Docker. That means you can run it on Oracle Cloud, but you can also run it on your own infrastructure or on other cloud systems, for example, Amazon Web Services (AWS), Google Cloud Platform, Microsoft Azure, and so on.

Serverless momentum

Speaking on stage for his 2018 re:Invent keynote in November, Amazon CTO Werner Vogels talked about the trajectory of serverless computing, particularly with enterprises.

"We normally expect younger, tech-oriented businesses as the first ones to try this out, but what we are actually seeing is large enterprises are the ones that are really embracing serverless technology," he revealed. "The whole notion of only having to build business logic and not think about anything else really drives the evolution of serverless."

Vogels was joined on stage by guitar maker Fender, which has certainly swallowed the serverless message whole. The firm uses Lambda triggers to underpin its digital content pipelines and talking up its ambition to free up its developers to focus on its digital products and not infrastructure.

"Even traditional organizations, like Fender, are all going serverless," Vogels added. "The advantages are obvious, there is nothing to provision, it scales automatically, it's highly available and secure, and most importantly you only have to pay for what you use."

Going into more detail, Holly Mesrobian, director of engineering for AWS Lambda said: "Today we talk a lot about scalability, reliability, performance, security, and cost. As we build out AWS Lambda we optimize for all of that in a serverless way."

Of those enhancements the one that got developers excited during re:Invent 2018 was the open source release of Firecracker, which allows for secure serverless development that doesn't cut back on performance. "We don't want our customers to make hard decisions between security and functionality," Mesrobian said.

Expanding on how it works, she added: "Firecracker provides secure and fast microVMs for serverless computing... To enable security from the ground up, it is built with speed by design. Initiating code in less than 125 milliseconds and a creation rate of 150 microVMs per second, per host, it ensures scale and efficiency, with low memory overhead of less than 5MB memory footprint per microVM and thousands of microVMs on each host."

The reason this got people so excited is because it brings a performance step change from Fargate, an AWS compute engine that allows customers to run containers without having to manage servers or clusters.

In his Twitch demonstration of the technology, Anthony Liguori launched 4,000 virtual machines, with the slowest taking 219 milliseconds.

"With Firecracker, you can see we are making the same deep investments in our infrastructure to support

serverless computing as we have with EC2 instances," Mesrobian added.

Serverless vendor pricing

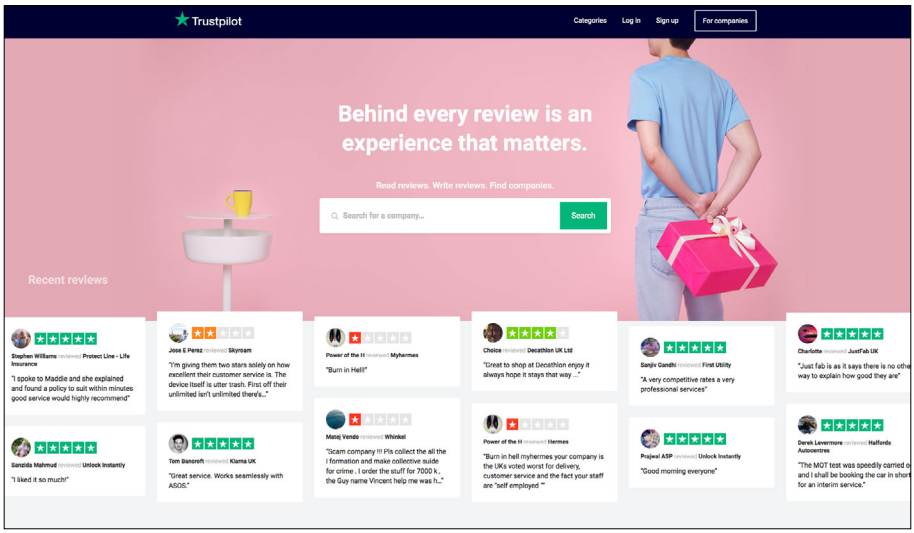
In an analysis of serverless cloud pricing (£), Owen Rogers at 451 Research found that serverless offers a lower cost of ownership (TCO) than virtual machines (VMs) and containers for the majority of new applications.

The main cost saving comes in the form of developer time as there is no need to provision, configure and manage infrastructure, and in increased utilization as users are only charged for the time they are actively using the platform.

The report compares serverless offerings from four main cloud providers – AWS, Google, Microsoft and IBM – and concludes that IBM offers the least expensive service, with Microsoft leading when it comes to certain configurations. IBM also stood out for Rogers because it allows users to choose exact memory requirements, where other providers round figures up.

Rogers notes that the serverless pricing model is "essentially the same model utilized by VMs, in which size and running time are the basis for cost, with the inclusion of number of times to represent the more variable aspect of serverless. In fact, the conceptual similarity to VM pricing might aid serverless' adoption with enterprises."

There is more good news for consumers too, as the report concluded: "Considering the similarities in pricing methods and offerings between providers, 451 Research believes serverless is poised to undergo a round of price cutting this year." In short: there is no better time to go serverless. **Scott Carey**



Credit: Trustpilot

Trustpilot's 'serverless first' approach to engineering

Trustpilot aiming to go serverless with Amazon Web Services

Trustpilot has embarked on an ambitious programme to go completely serverless with Amazon Web Services, with a bold aim to completely embrace the modern architecture by the middle of this year, accounting for what the organization estimates could be a 10x saving on cloud compute costs.

The Danish web company, which collates independent reviews for online businesses, started its serverless journey in 2016, when VP of engineering Martin Buberl came back from AWS re:Invent in 2016. Speaking at re:Invent in Las Vegas in November, Buberl

said he “couldn’t have imagined standing up here” if you had asked him two years ago.

His engineering team successfully shifted to a nearly completely serverless architecture, leaning heavily on Lambda functions to reach a point where AWS is essentially fully responsible for code execution.

“Serverless was not completely new to me, but the concept of serverless compute and Lambda functions really clicked for me [in 2016],” he said.

The company had already been cloud native for five years, running a high level architecture of event driven microservices and REST APIs. Now, with the addition of serverless functions-as-a-service and event queues, he felt ready to take the engineering team to what he saw as the next level.

How did it get there?

His first move was to establish what Trustpilot calls its “engineering principles” to add “serverless first” to its architecture. That reads: “If serverless is not available or practical, containers are recommended. Virtual servers are considered legacy and should be avoided.”

Buberl admits that the day he got back from Las Vegas with grand plans to go completely serverless there were varying degrees of excitement from his engineers, and said that he may have overlooked the all important ‘why’ of Simon Sinek’s Golden Circle.

It’s the last bit of that principle which caused most of the pushback from its population of .NET developers, who remained reliant on virtual servers.

After engaging with the firm’s engineers, Buberl said: “What happened is the engineers were happier, but there were still a few folks raising their eyebrows

and not fully bought in." After heading back to the drawing board, the organization opted to move to .NET Core and Docker for that team.

As a result, the expanded principle reads: "We do this because we strongly believe that serverless (FaaS, BaaS, DBaaS) is the future of the cloud and we'd like to be on the forefront of that movement. Serverless might not necessarily be the right choice for everything today, but start your architecture discussions there. We're in the process of fading out virtual servers and want to avoid creating new ones."

Once they were happy with this principle they open sourced it on GitHub, where it joined others to code review everything, services first, build smaller things, encapsulate in contexts and expose APIs, and aim to open source.

How does this architecture look?

This new architecture relies on an API management layer and the simple notification service (SNS) pub/sub messaging service, which is tooled using GitHub and Slack. "GitHub and Slack means you can immediately start using [Lambdas]," he said. So anytime anything happens, posts are sent out using the API gateway, where Lambda subscribes and fans out triggered actions using that SNS pub/sub mechanism, broadly speaking. One example of how this is leveraged is for GDPR compliance. Data scientists were sometimes accidentally committing personally identifiable data within their training sets to GitHub, which would cause problems at audit. The answer is to bubble this up to Slack every time a potentially problematic commit is made to get that taken down as quickly as possible.

The company has moved to running 53 per cent less virtual servers, from 180 to 95 today; 283 containers, up 354 per cent from 80 in 2016, and 252 regular Lambda functions, up from 40.

Benefits

Buberl said the question he gets asked the most is whether the Lambda functions are cheaper. The problem is, he believes Lambda triggers versus traditional cloud compute is like comparing apples with bananas.

"Effort has to go in to autoscaling systems," he added. "And we see it's hard to quantify. Then if you make mistakes and the system doesn't scale that is expensive too." However, his "gut feel" is that its serverless architecture is now "10 times cheaper" thanks in large part to the reduction in operations overhead.

The other benefits of going serverless, he said, are faster development speeds, but the biggest downside has been a loss of traceability over systems.

"We're investing in this as you have lots of smaller systems," he said, with Trustpilot now running more than 500. Today his team is using Amazon X-Ray and logging to track these services, but is looking to invest in a service mesh "to bring all these systems together and map them there". **Scott Carey**