**Azure Functions**

[**https://docs.microsoft.com/en-us/azure/azure-functions/functions-overview#:~:text=Azure%20Functions%20allows%20you%20to,a%20specific%20type%20of%20event**](https://docs.microsoft.com/en-us/azure/azure-functions/functions-overview#:~:text=Azure%20Functions%20allows%20you%20to,a%20specific%20type%20of%20event)**.**

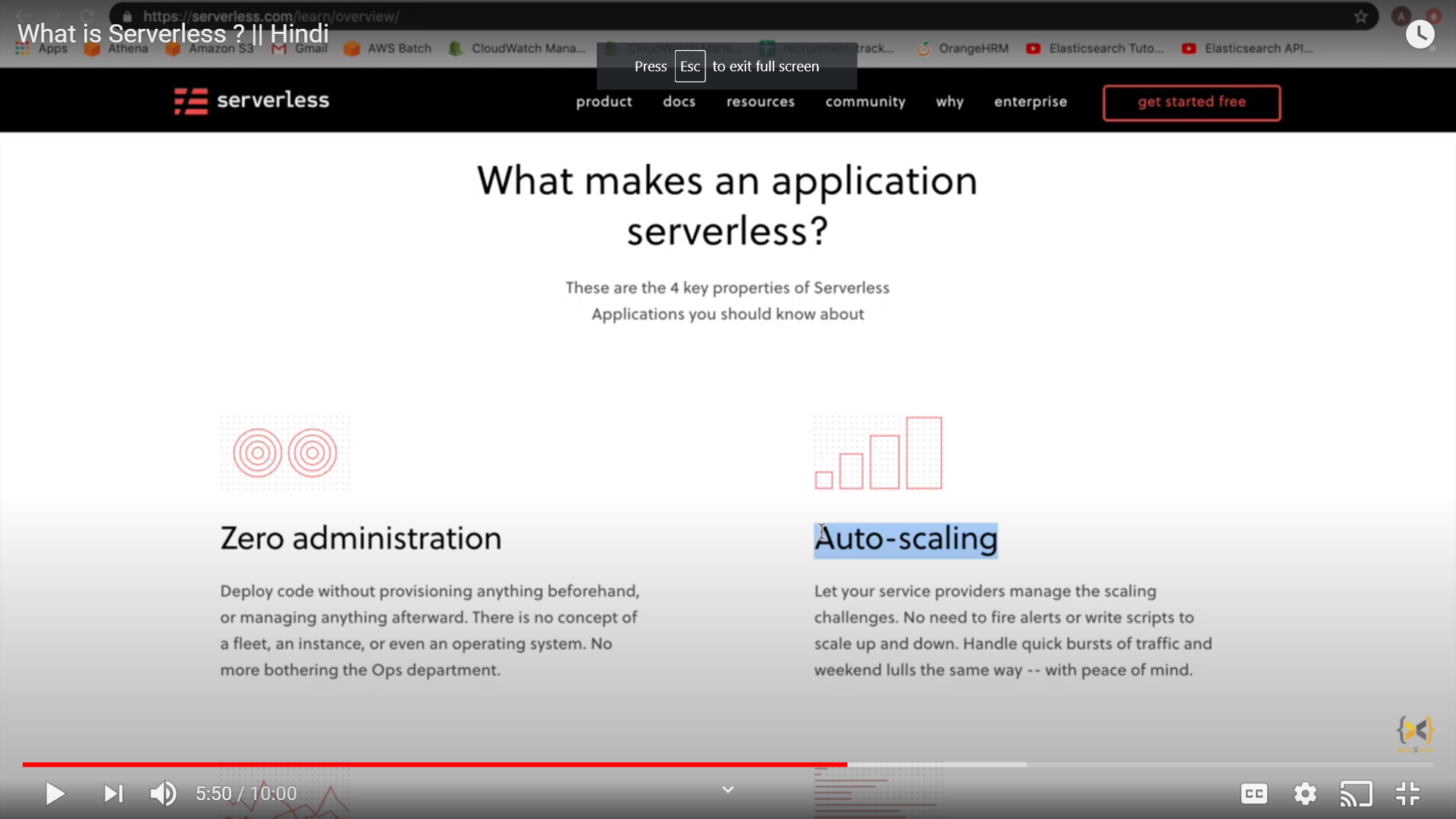
**Serverless** is the native architecture of the cloud that enables you to shift more of your operational responsibilities to AWS, increasing your agility and innovation. **Serverless** allows you to build and run applications and services without thinking about servers.

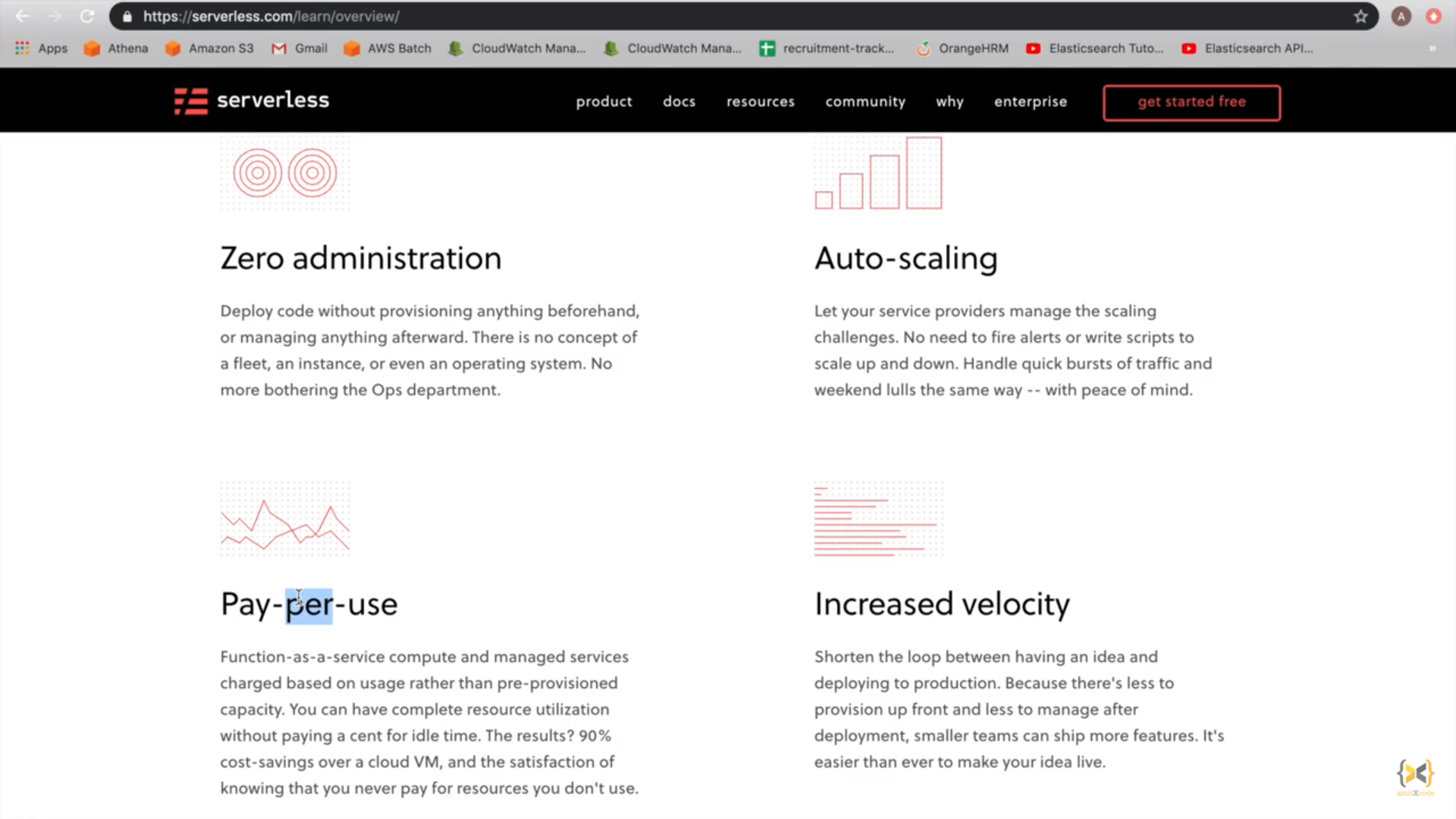
**Serverless** computing offers a number of advantages over traditional cloud-based or server-centric infrastructure. For many developers, **serverless** architectures offer greater scalability, more flexibility, and quicker time to release, all at a reduced cost.

**When to use serverless architecture**

* High latency background tasks like multimedia or data processing.
* Client-heavy applications where most of the logic can be moved to the client.
* Applications with an unpredictable amount of server load.

**Serverless** architectures are **application** designs that incorporate third-party “Backend as a Service” (BaaS) services, and/or that include custom code run in managed, ephemeral containers on a “Functions as a Service” (FaaS) platform.





**Azure Functions Keys**

[**https://vincentlauzon.com/2017/12/04/azure-functions-http-authorization-levels/**](https://vincentlauzon.com/2017/12/04/azure-functions-http-authorization-levels/)

**Create functions through different ways**

[**https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-first-azure-function-azure-cli?tabs=bash%2Cbrowser&pivots=programming-language-csharp**](https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-first-azure-function-azure-cli?tabs=bash%2Cbrowser&pivots=programming-language-csharp)

**Logic App and Function app combined**

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-twitter-email>

Web Jobs

If the JOB is set to run continuously, once the process exits (say you are polling an queue and it's empty) the job shuts down and **status** changes to "**pending restart**". **Azure** Scheduler will typically **restart** the process in 60 seconds

A popular comparison states that **Azure Functions** is code being triggered by an event, whereas **Logic Apps** is a workflow triggered by an event. This is reflected in the developer experience. Azure Functions are completely written in code, with currently supports JavaScript, C#, F#, Node.js, Python, PHP, batch, bash and PowerShell. In Logic Apps, workflows are created with an easy-to-use visual designer, combined with a simple workflow definition language in the code view. Each developer has of course his/her personal preference. Logic Apps is much simpler to use, but this can sometimes cause limitations in complex scenarios. Azure Functions gives a lot more flexibility and responsibility to the developer.