

#XIVJORNADASCCNCERT

SRTTTools

Serverless Red Team Tools





Manuel Fernández-Aramburu

Entelgy Innotec Security

-> [linkedin.com/in/manuelaramburu](https://www.linkedin.com/in/manuelaramburu)

***Ingeniero en Informática por la
Universidad de Sevilla.***

Pentester / Consultor Ciberseguridad

***Jugador de CTFs, uno de los ganadores
de Hacker Arena organizado por CCN-CERT
en 2019. Segundo clasificado en Hackthón
en Cybercamp 2019 por el desarrollo de
una herramienta de Red Team.***



Daniel Sesé

Entelgy Innotec Security

-> [linkedin.com/in/daniel-sesé-benjumea](https://www.linkedin.com/in/daniel-sesé-benjumea)

***Estudiante de Ingeniería en Informática
por la Universidad de Sevilla.***

Pentester / Consultor Ciberseguridad

***Jugador de CTFs, uno de los ganadores
de Hacker Arena organizado por CCN-CERT
en 2019. Segundo clasificado en Hackthón
en Cybercamp 2019 por el desarrollo de
una herramienta de Red Team.***



Daniel Sesé

Entelgy Innotec Security

-> [linkedin.com/in/daniel-sesé-benjumea](https://www.linkedin.com/in/daniel-sesé-benjumea)

Ingeniero en Informática, Master en Ciberseguridad y estudiante de Doctorado en la universidad de Sevilla .

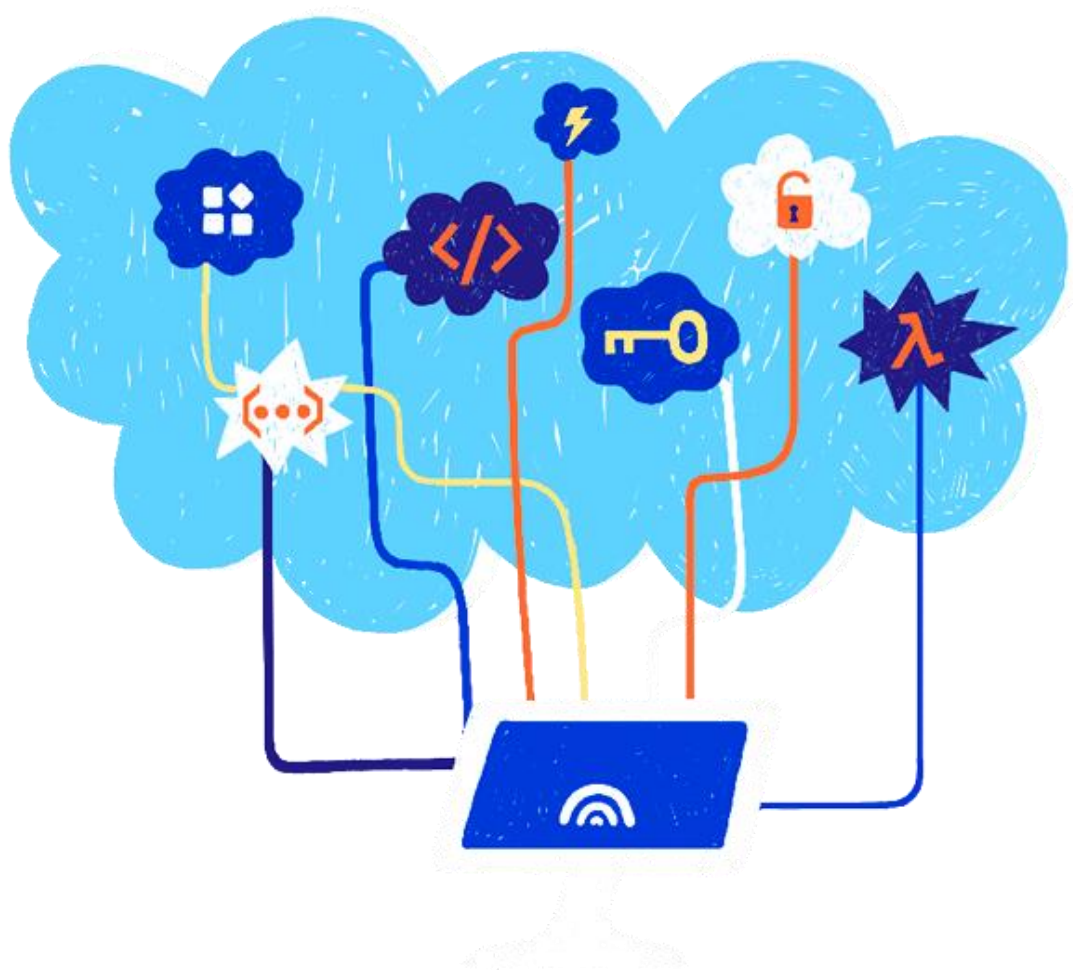
Pentester / Consultor Ciberseguridad

Cuento con 6 años de experiencia en el sector de la ciberseguridad enfocado a entornos IT y OT.

Introducción

¿Qué es la tecnología Serverless?

Si traducimos la palabra Serverless es “sin servidor” pero no es la idea exacta de esta tecnología, existe realmente un servidor que vamos a usar para nuestros proyectos pero simplemente nos olvidamos de gestionar las herramientas necesarias. Es decir, vamos a ejecutar aplicaciones en contenedores que están en servidores cloud, pero solamente funcionaran y estarán disponible mientras ejecutemos nuestras aplicaciones. Después de su uso estos contenedores se liberan y no tendremos que preocuparnos de seguir manteniéndolos.



Introducción

Que tecnología Serverless tenemos

AWS Lambda

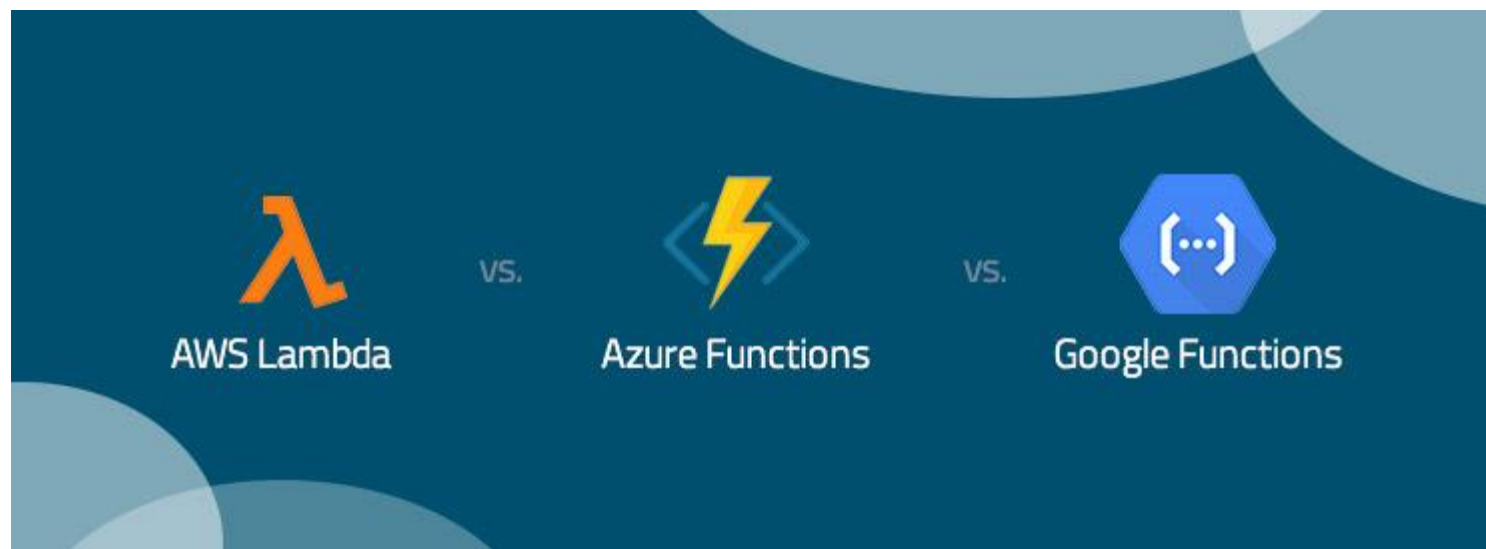
Es el más conocido y usado del mercado. Entre las aplicaciones que puede usar están Python, Go, Java, C# o Node.js.

Azure Functions

Está desarrollado por Microsoft desde hace dos años y puedes utilizar con él aplicaciones como Bash, Powershell, Java, Python, C#, F#, PHP o Batch.

Google Cloud Functions

Lo puedes encontrar en la plataforma de Google y solo se puede usar con el código Javascript para ejecutarse en un entorno Node.js.

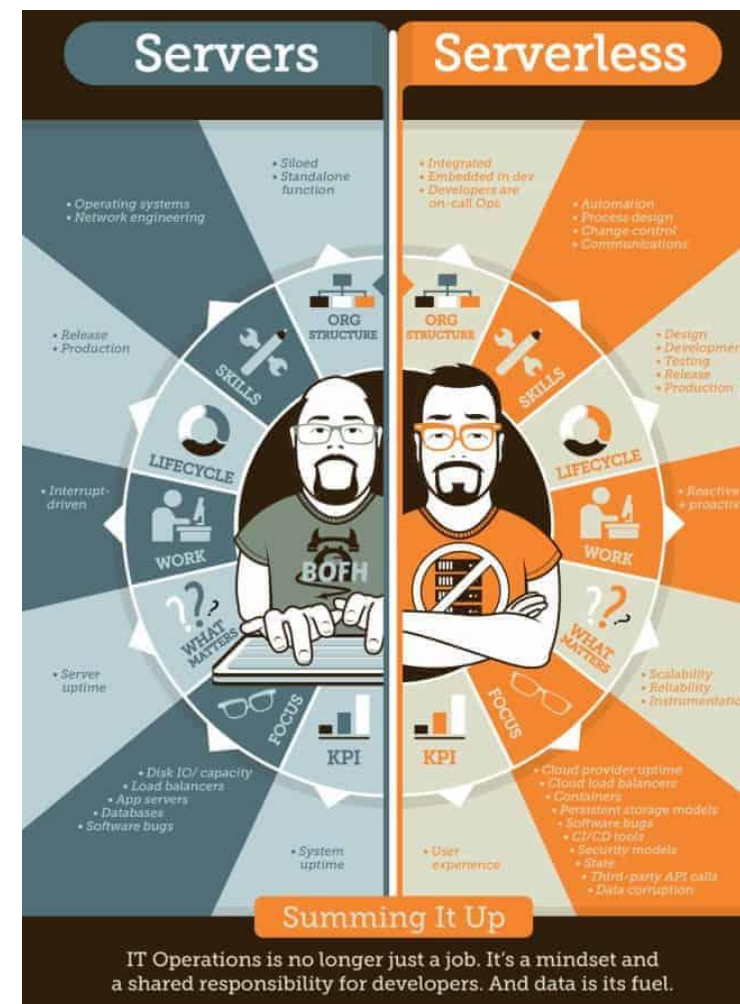


Introducción

Ventajas de usar la tecnología Serverless

Este tipo de tecnología tiene 4 ventajas que vamos a ver a continuación:

- Escalabilidad
- Mantenimiento
- Costes
- Integración



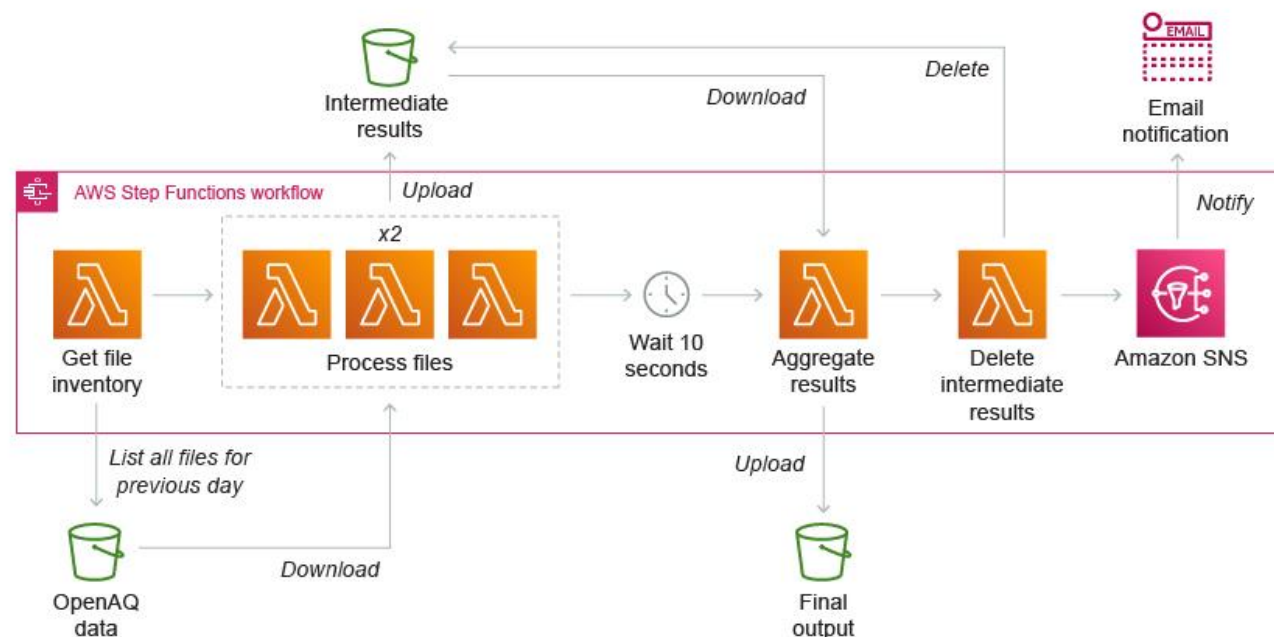
Introducción

Ventajas de usar la tecnología Serverless

Escalabilidad

En esta parte solo nos tenemos que preocupar de lanzar nuestra aplicación ya que los recursos dentro de la tecnología serverless se adapta a lo que estemos ejecutando en ese momento, aunque también podemos añadir clusters o balanceadores a nuestra api Gateway para adaptarlos a la necesidad de la aplicación que se esta ejecutando.

Como vemos en el ejemplo se puede escalar horizontalmente según su uso.



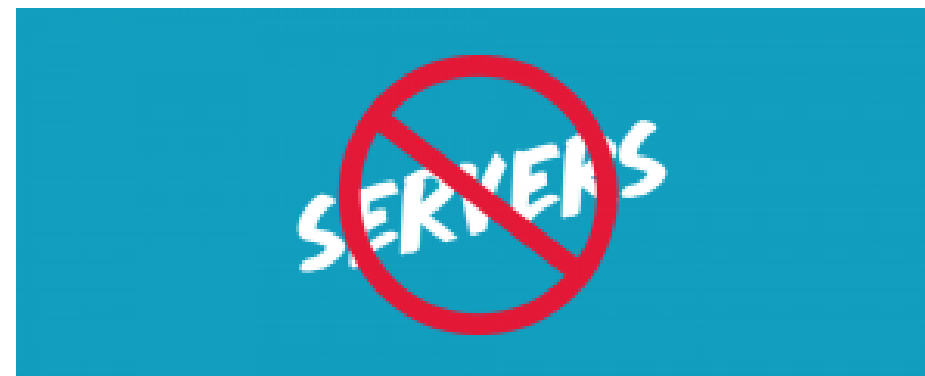
Introducción

Ventajas de usar la tecnología Serverless

Mantenimiento

No se tiene que hacer mantenimiento del servidor ya que no dispones de esa gestión, simplemente te ofrecen un entorno para usar una tecnología o lenguaje de programación en el que vas a ejecutar tu aplicación.

No hace falta instalar ningún software, actualizarlo o gestionar configuraciones de redes ya todo se encarga el proveedor de serverless por ti.



Introducción

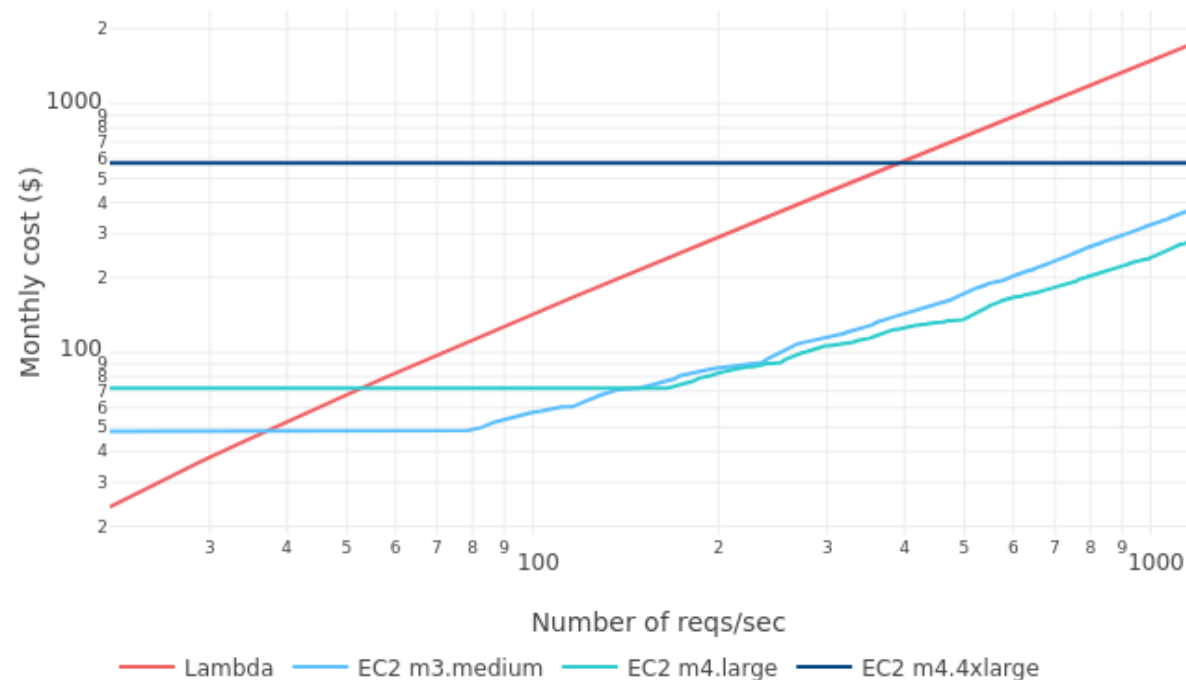
Ventajas de usar la tecnología Serverless

Costes

Al no disponer de servidores propios y simplemente va a hacer ejecución de código temporalmente en un contenedor que cuando acaba el proceso desaparece.

Es por ello que solo haces un pago por el uso que le des a esa computación, y no tener un servidor disponible para ti las 24 horas, con mucho mayor coste obviamente.

Monthly cost by number of requests per second



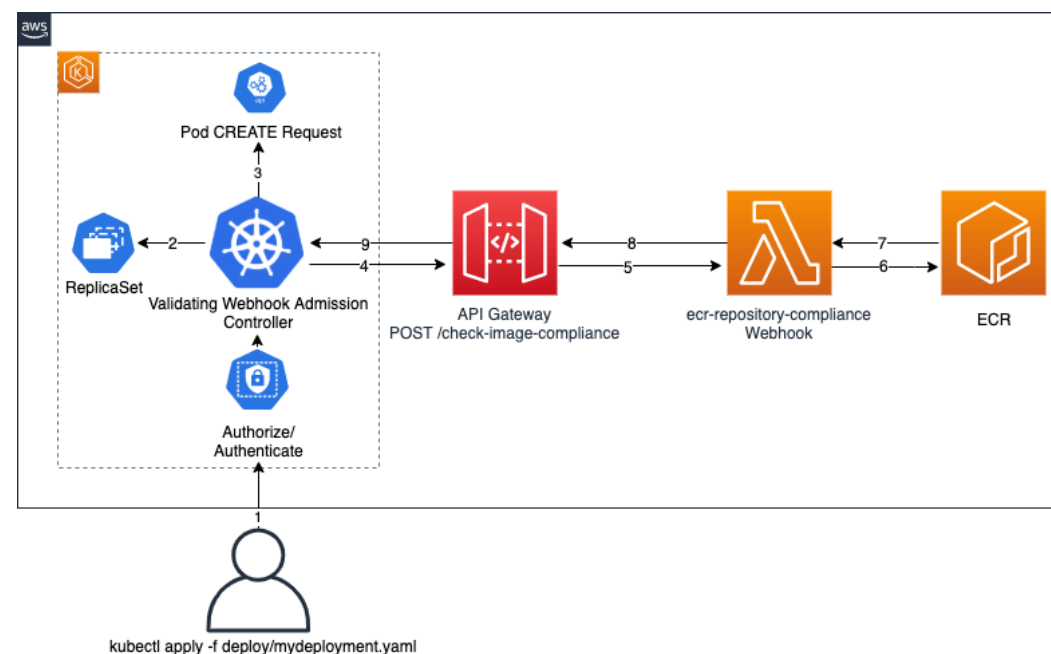
Introducción

Ventajas de usar la tecnología Serverless

Integración

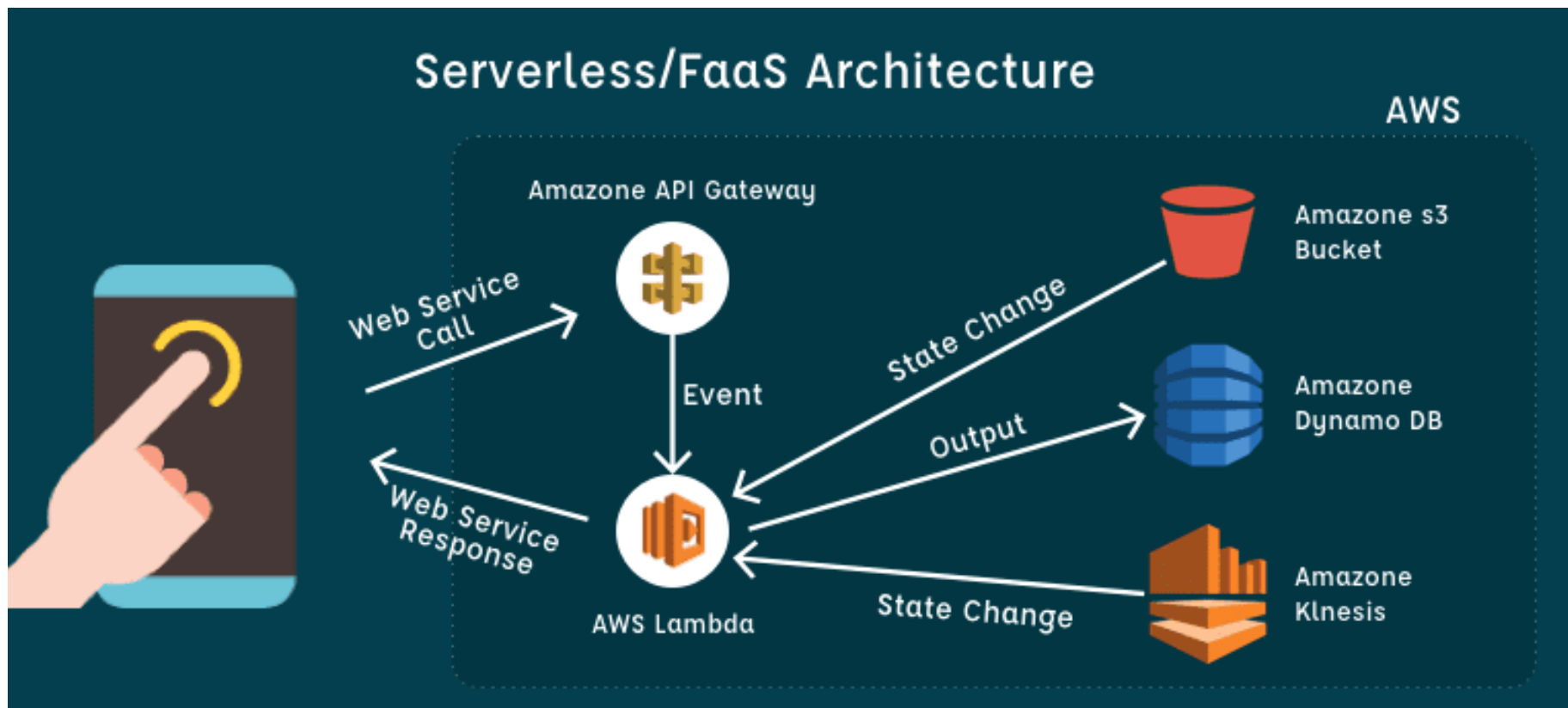
Este tipo de servicio no solo te ofrece computación, también tienes integración con otros servicios de tu proveedor o incluso con otras API o sistemas.

Todo ello está integrado de una manera fácil en la que la mayoría de veces solo tienes que añadirlo en el servicio serverless y rellenar algunos campos de configuración.



Introducción

Como lo vamos a usar



Como accedemos a la tecnología



Serverless Red Team Tools

La herramienta para ejercicios de Red Team en entornos Cloud

Os vamos a presentar como usando la tecnología anteriormente descrita nos va a facilitar el proceso de detección de vulnerabilidades en sistemas o apps o también crear un sistema de conexión de Command & Control que nos permitirá comprobar la seguridad en infraestructuras o sistemas críticos.

La herramienta que presentamos tiene 2 principales uso:

- Collaborator HTTP
- C2 (Command & Control)

SRTTTools



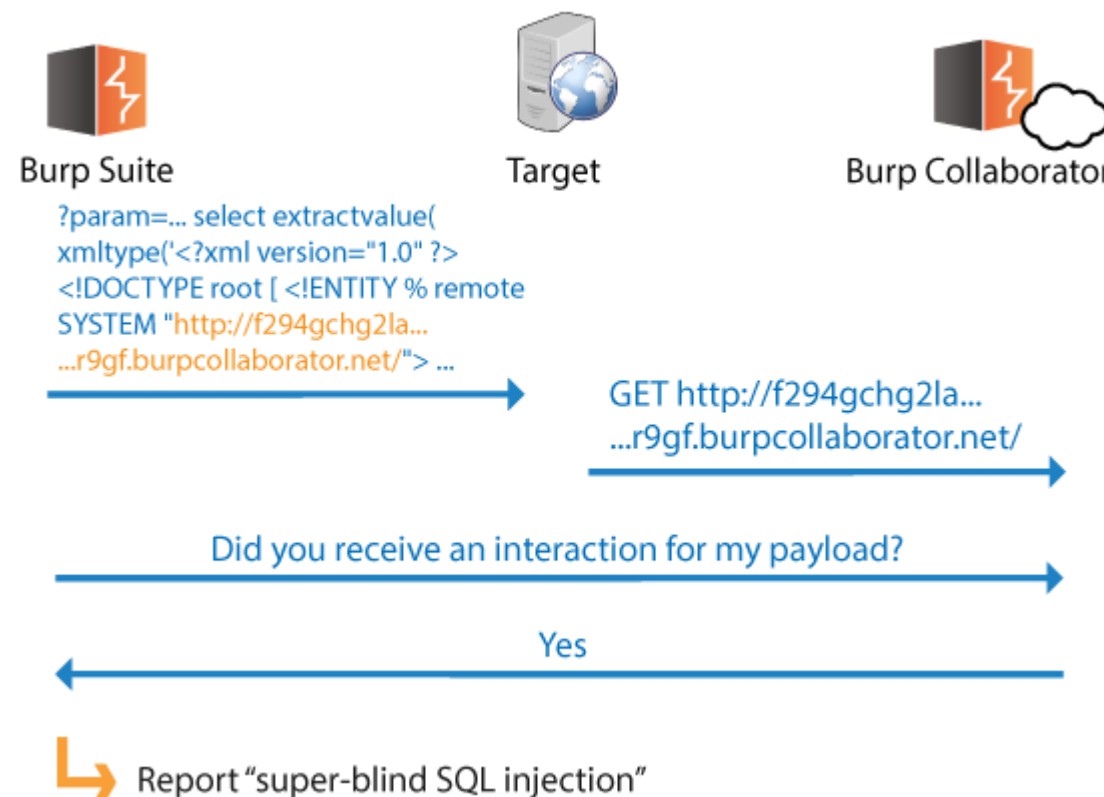
Serverless Red Team Tools

Collaborator HTTP

En el caso de tener la necesidad de comprobar si algún sistema tiene una inyección de código y nos permite la ejecución de comandos o solicitar un recurso por HTTP, en el que algunos casos no conocemos si esta inyección se esta ejecutando con éxito.

Es por ello que hemos creado una herramienta que podemos integrar con un proxy web y nos permite recibir la información si esta inyección de código se ha realizado con éxito.

Para este funcionamiento nos hemos basado en la herramienta de PortSwigger como es Burp Collaborator.



Demo Collaborator

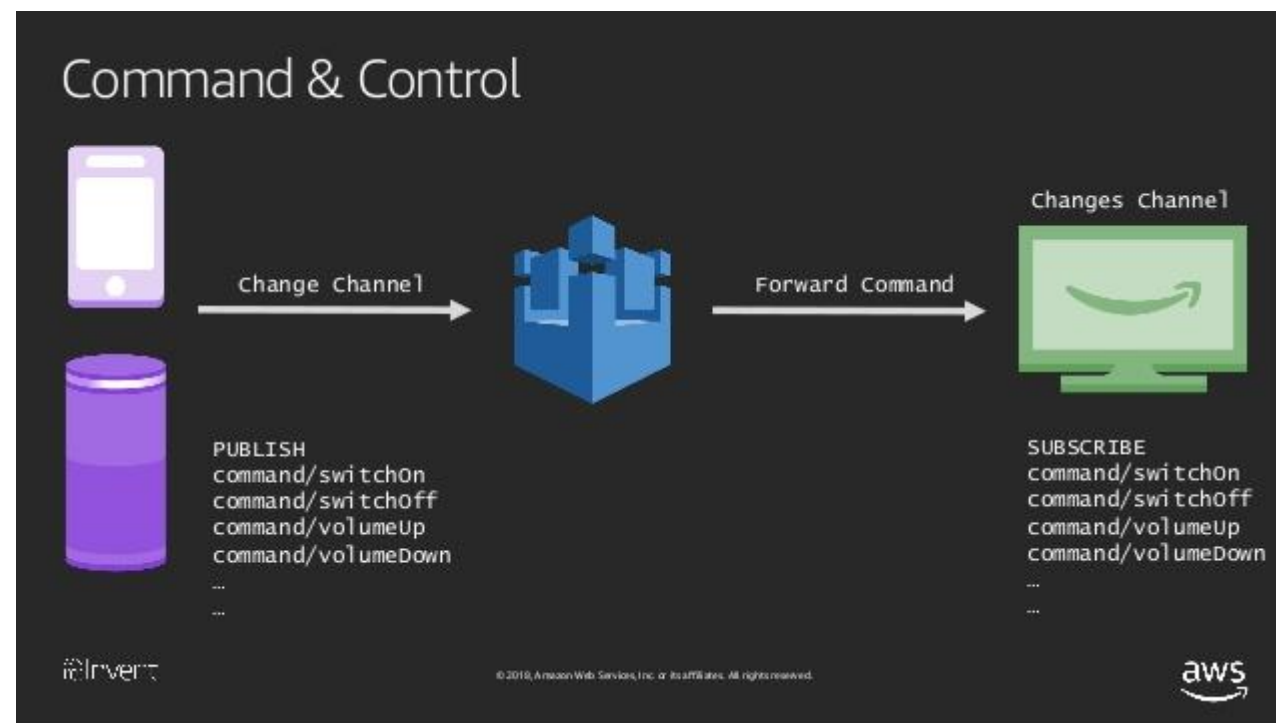


Serverless Red Team Tools

C2

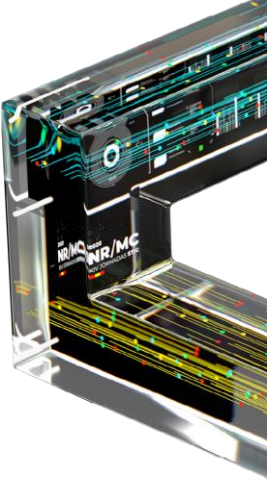
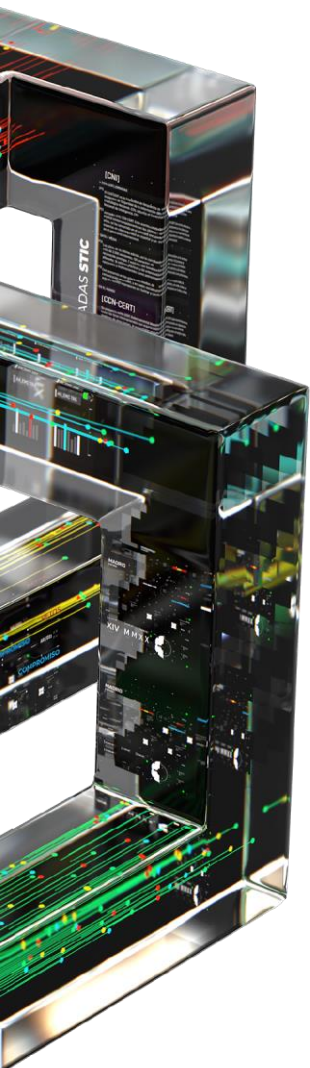
En este caso hemos creado una herramienta en el que se ejecuta un cliente en PowerShell en el sistema o infraestructura a realizar la prueba para establecer una comunicación con la tecnología serverless.

Todo ello va a través de una navegación por websockets y en el que además podemos establecer un proxy en el que comprobar las redes del servidor en el que se han establecidos las pruebas.



Demo C2





MUCHAS GRACIAS

#XIVJORNADASCCNCERT