



Code in the Cloud with AWS Lambda

“The magic happens at the intersection of functions, events, and data”

(everything else is glue or UI)

Why Lambda?

- Focus on what's important – your code
- Only pay for resources consumed
- Fault tolerant (code is replicated to multiple regions, automated execution retries)
- Highly available
- Monitoring (CloudWatch)
- Logging (CloudWatch Logs)
- Scales easily – no additional provisioning required

What is Lambda

- Scalable, Highly Available, Stateless, event driven computing
- Fully managed runtime environment



Python



Node.js



Java

Getting Started with Lambda – Lambda function anatomy

- **Basic Description**
 - Function Name, Description, and Runtime
- **Your code**
 - Inline
 - Zip file containing files
- **Function handler and Role**
 - Handler tells Lambda where to start execution
 - Role – IAM role, give Lambda access to services and data
- **Advanced Settings**
 - Memory Allocation
 - Timeout
 - VPC!!!!
- Event Sources
- API Endpoints

Lambda Limits - Resources

Resource	Default Limit
Ephemeral disk capacity ("/tmp" space)	512 MB
Number of file descriptors	1,024
Number of processes and threads (combined total)	1,024
Maximum execution duration per request	300 seconds
Invoke request body payload size	6 MB
Invoke response body payload size	6 MB

<http://docs.aws.amazon.com/lambda/latest/dg/limits.html>

Lambda Limits – Deployment limits

Item	Default Limit
Lambda function deployment package size (.zip/.jar file)	50 MB
Size of code/dependencies that you can zip into a deployment package (uncompressed zip/jar size)	250 MB

<http://docs.aws.amazon.com/lambda/latest/dg/limits.html>

Lambda Limits – Limits per Region

Item	Default Limit
Total size of all the deployment packages that can be uploaded per region	75 GB
Concurrent executions (see Concurrent Executions)	100


<http://docs.aws.amazon.com/lambda/latest/dg/limits.html>

Lambda Pricing

- We pay for requests, and request duration
 - First 1M requests are free, there after \$0.20/M requests
- Duration is billed in 100ms increments (rounded up)
 - Duration price changes based on allocated memory
- Other AWS charges may apply
 - S3 requests, Storage, etc.


<https://aws.amazon.com/lambda/pricing/>


Deploying Lambda – AWS Console





AWS

Services

 EC2

 VPC

 IAM

 RDS

Edit

CorpInfoMSTeam/kepstein@co...N. VirginiaSupport

Lambda > New function

Step 1: Select blueprint

Step 2: Configure function

Step 3: Review

Configure function

A Lambda function consists of the custom code you want to execute. [Learn more](#) about Lambda functions.

Name*

myFunctionName

Description

Runtime*

Node.js

Lambda function code

Provide the code for your function. Use the editor if your code does not require custom libraries (other than the aws-sdk). If you need custom libraries, you can upload your code and libraries as a .ZIP file. [Learn more](#) about deploying Lambda functions.

Code entry type

☒ Edit code inline

☐ Upload a .ZIP file

☐ Upload a .ZIP from [Amazon S3](#)

1



Deploying Lambda - CLI

```
1. kevin@kevin: / (zsh)
+ / aws lambda create-function \
>   --function-name MyFunction \
>   --description "Some Awesome Description" \
>   --memory-size 128 \
>   --timeout 300 \
>   --handler lambda.handler \
>   --runtime python2.7 \
>   --zip-file file.zip \
>   --region us-west-2
```

Deploying Lambda – Honorable mentions

- gulp-awslambda if you're deploying Node
- Jenkins or Bamboo (plugins)
- Frameworks (Kappa and Apex are interesting)
<https://github.com/garnaat/kappa>
<https://github.com/apex/apex> and <http://apex.run/>
- Roll your own custom build and deployment scripts

Triggering Lambda



S3



API Gateway



DynamoDB



AWS IoT



**Alexa Connected Home
Alexa Skills Kits**



**Cognito
Sync Errors**



Kinesis



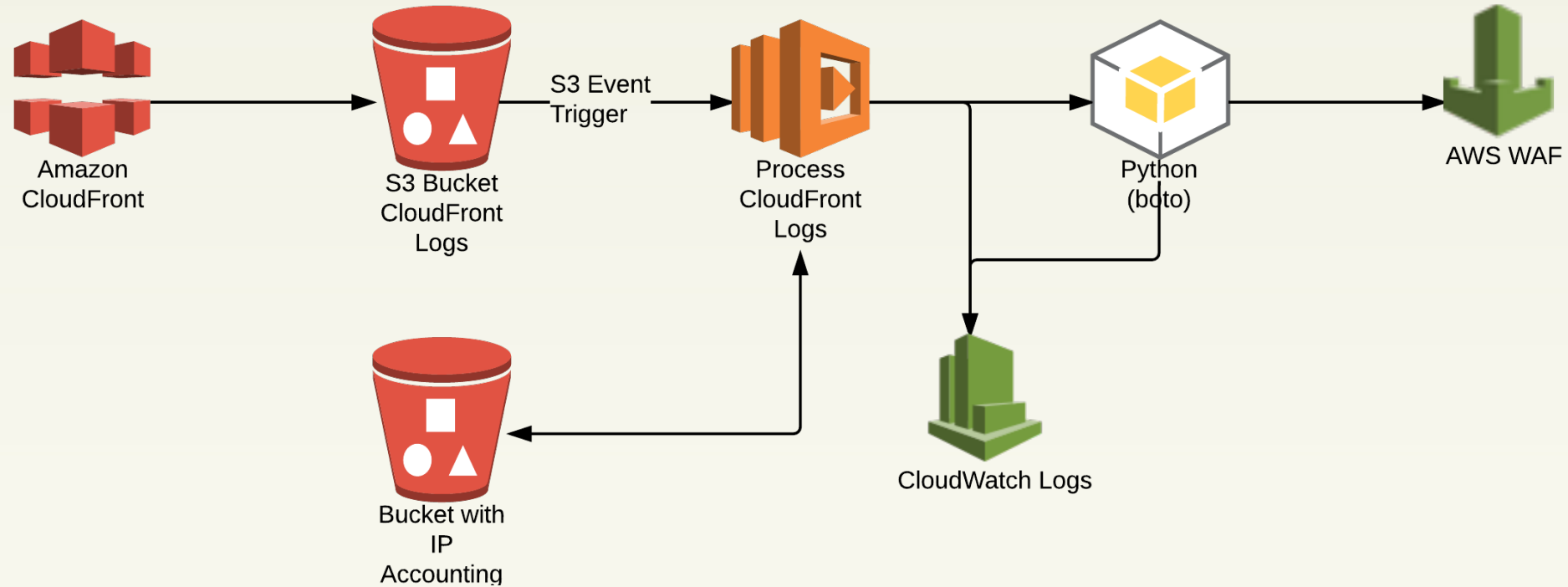
**CloudWatch Logs
CoudWatch Events**

LOS ANGELES AWS USERS GROUP

Walk through an example – Micro batch updates to Redshift



Walk through an example – Updating AWS WAF Rules



Walk through an example – Alexa Skills



When things don't work – Troubleshooting tips

- Check the CloudWatch Logs
- Make sure your execution can complete within the configured timeout
- Make sure you have enough memory allocated
 - Make sure you're not exceeding the published limits
- Make sure you're not being throttled (Error 429)
- Make sure you're loading all the necessary libraries
- Make sure you have appropriate IAM permissions
- **Specifically for Alexa Skills Kit**– Make sure you're deploying into us-east-1!