

What Does This Button Do?

Eric Johnson

Sr Developer Advocate – Serverless Applications, AWS

goto; chicago



**Click 'Rate Session'
to rate session
and ask questions.**

A close-up portrait of a man with a bald head, a full grey goatee, and black-rimmed glasses. He is wearing a dark brown sweater and has a slightly pensive or tired expression. The background is a solid dark red color.

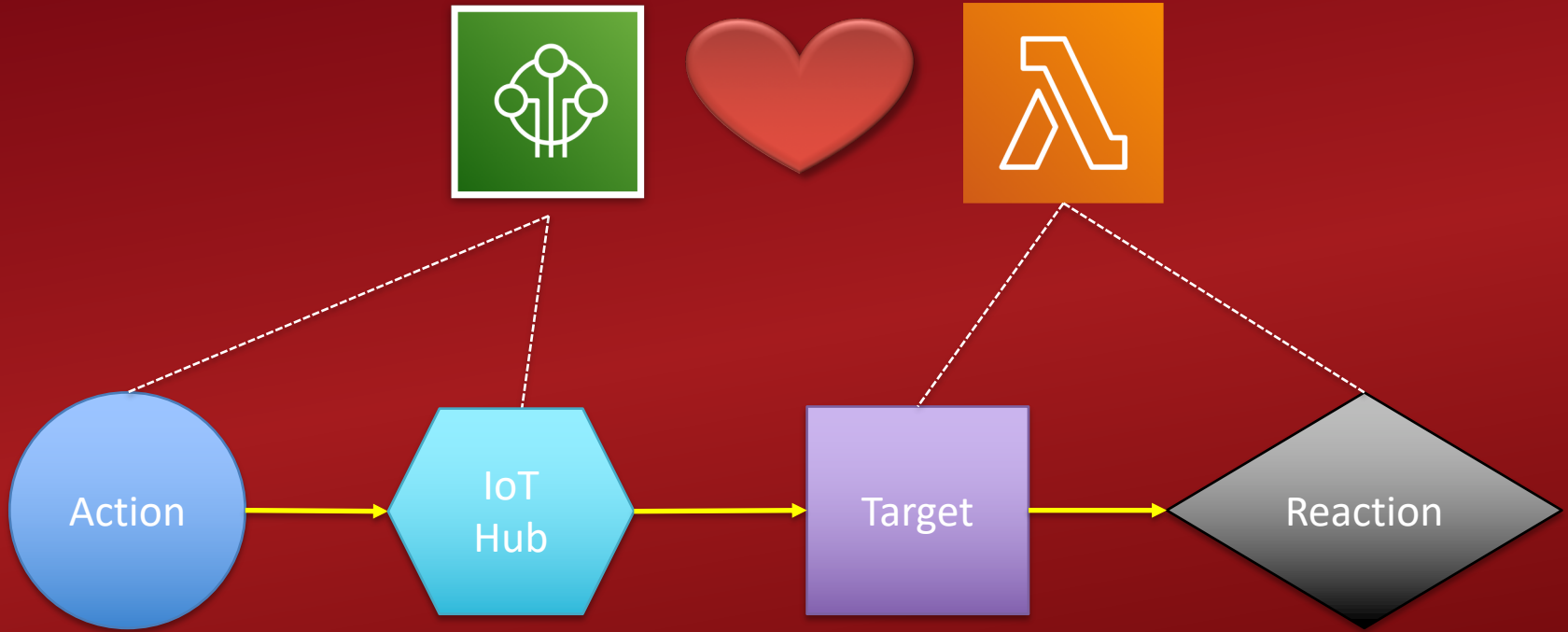
Who Am I?

- Sr. Developer Advocate
 - Serverless, AWS
- Dad of five
- Husband of one
- ~~Musician~~
- Drummer

Today's Agenda

- IoT and Serverless
- Architecture
- Tools
- Demo
 - Build and Deploy Application Backend
 - Deploy Application Front End
 - Activate an IoT 1-Click Button

IoT Love's Serverless



Things To Do



- Start an AWS service
- Send an SMS
- Send an Email
- Turn on/off a light
- Alert your family
- Start a car
- Transfer Money

- Retrieve your Tesla
- Retrieve your kids
- Close a garage
- Put house on vacation
- Feed your dog
- Start sprinklers
- Call the police



IoT Button



AWS IoT 1-Click



SERVICES AND ARCHITECTURE

Right tools for the job

AWS Services



Amazon
API Gateway



AWS Lambda



Amazon
DynamoDB



AWS IoT
1-Click

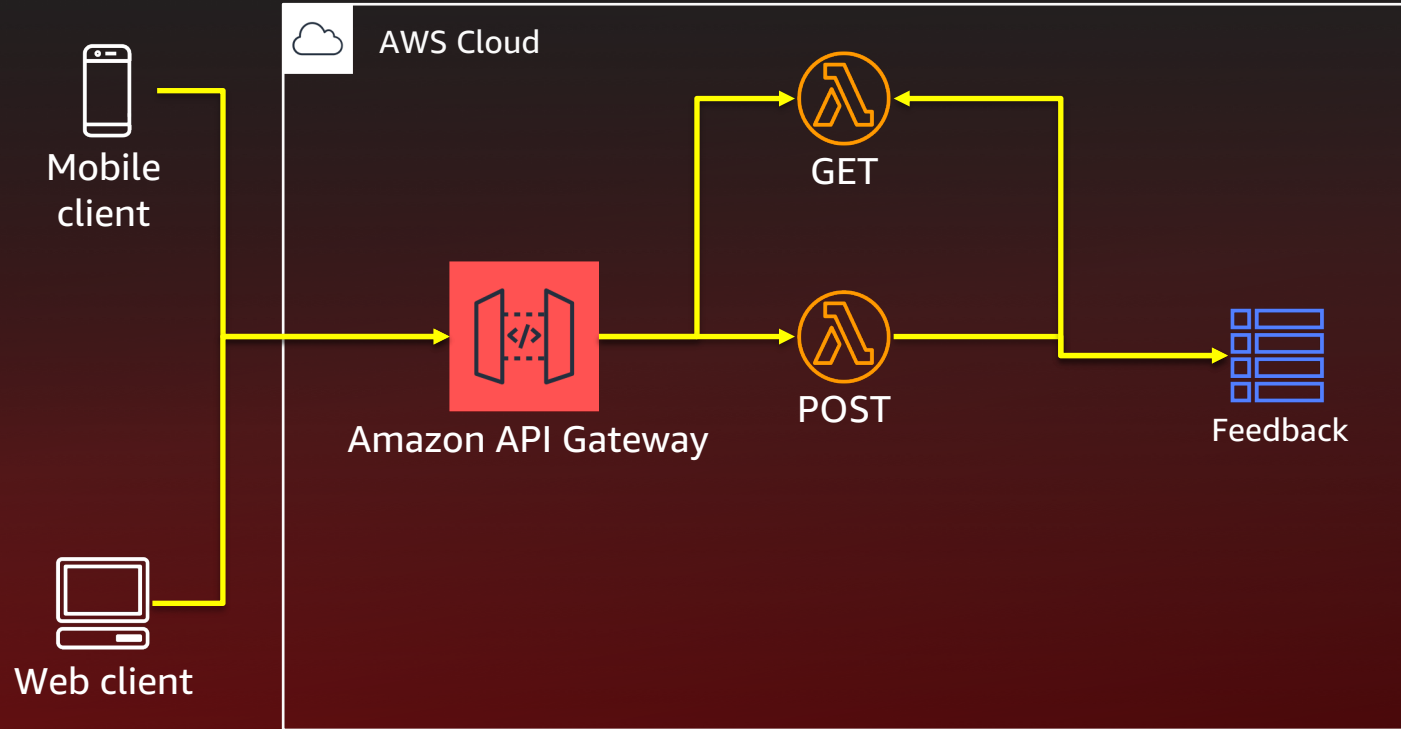


AWS
Systems
Manager

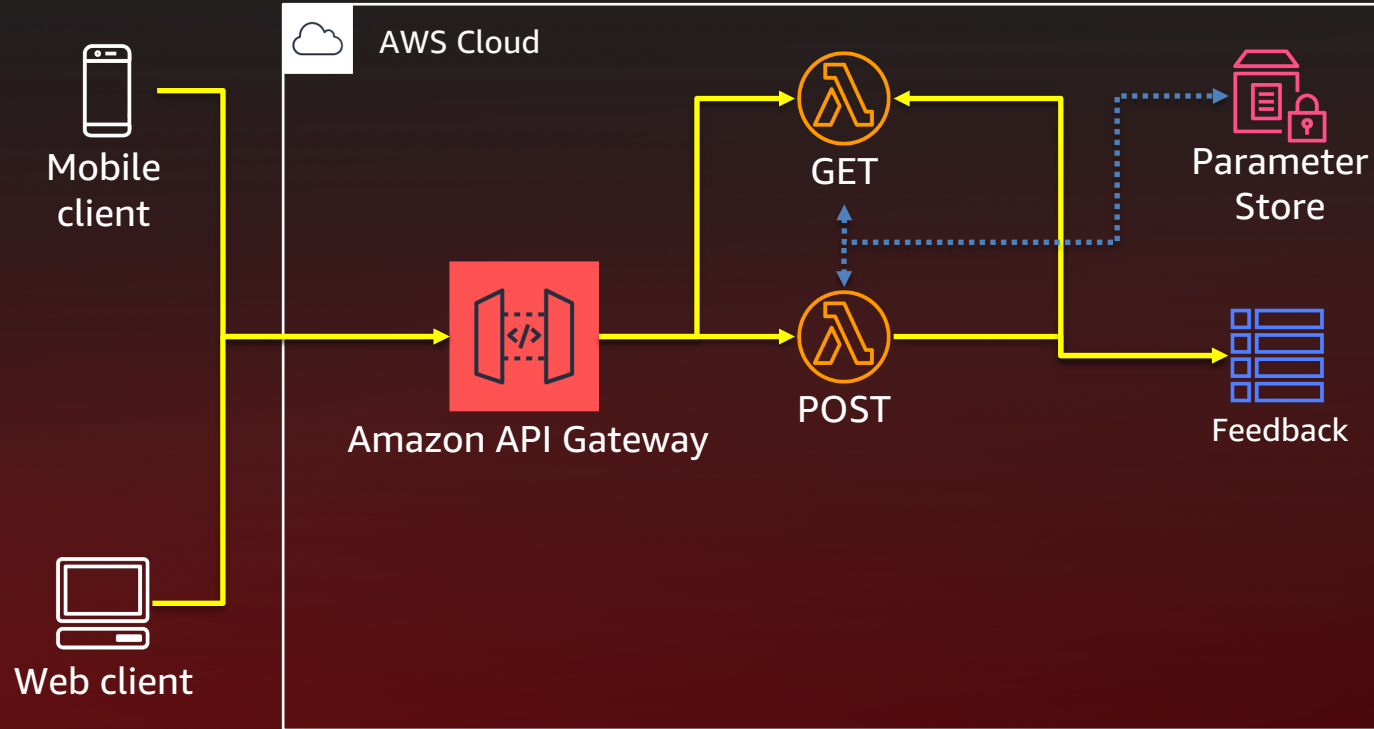


AWS Amplify

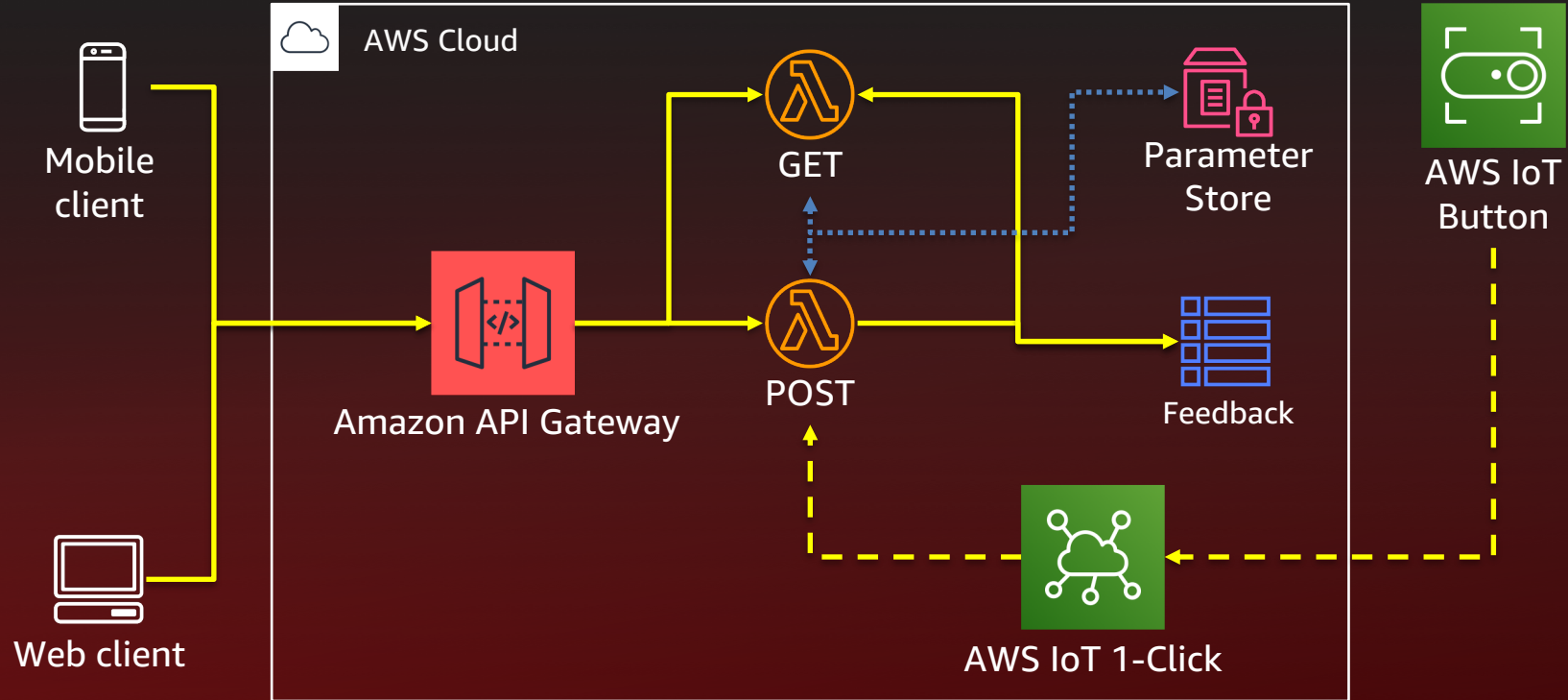
Today's Architecture



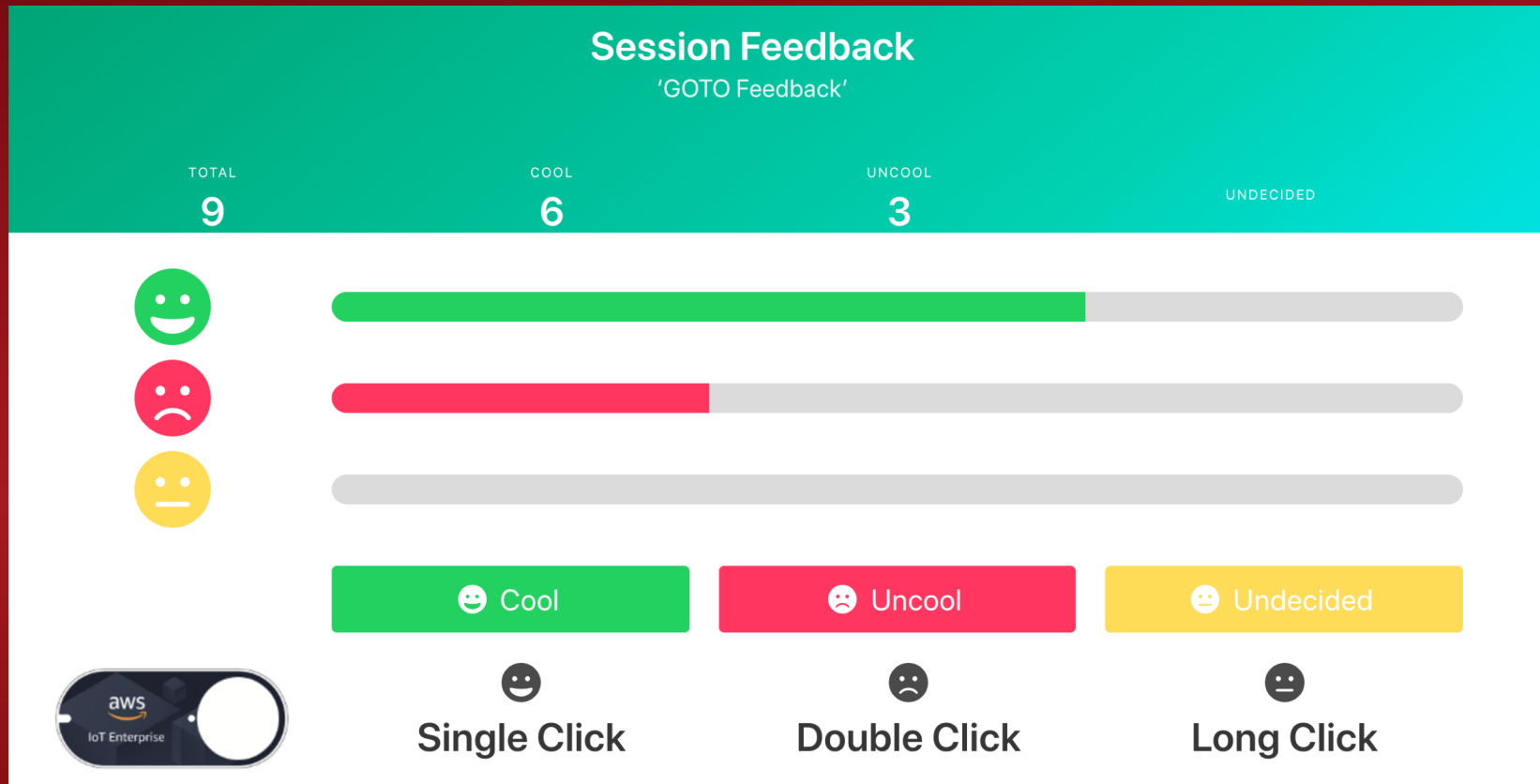
Today's Architecture



Today's Architecture



The End Goal



MEET SAM

AWS Serverless Application Model



Serverless **A**pplication **M**odel



- AWS CloudFormation extension optimized for serverless
- Special serverless resource types: functions, APIs, tables, Layers and Applications
- Supports anything AWS CloudFormation supports
- Open specification (Apache 2.0)

<https://aws.amazon.com/serverless/sam>

SAM Template

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Resources:
  GetHtmlFunction:
    Type: AWS::Serverless::Function
    Properties:
      CodeUri: s3://sam-demo-bucket/todo_list.zip
      Handler: index.gethtml
      Runtime: nodejs6.10
      Policies: AmazonDynamoDBReadOnlyAccess
      Events:
        GetHtml:
          Type: Api
          Properties:
            Path: /{proxy+}
            Method: ANY

  ListTable:
    Type: AWS::Serverless::SimpleTable
```

SAM Template

```
AWSTemplateFormatVersion: '2010-09-09'
```

```
Transform: AWS::Serverless-2016-10-31
```

```
Resources:
```

```
  GetHtmlFunction:
```

```
    Type: AWS::Serverless::Function
```

```
    Properties:
```

```
      CodeUri: s3://sam-demo-bucket/todo_list.zip
```

```
      Handler: index.gethtml
```

```
      Runtime: nodejs6.10
```

```
      Policies: AmazonDynamoDBReadOnlyAccess
```

```
      Events:
```

```
        GetHtml:
```

```
          Type: Api
```

```
          Properties:
```

```
            Path: /{proxy+}
```

```
            Method: ANY
```

```
  ListTable:
```

```
    Type: AWS::Serverless::SimpleTable
```

Tells AWS CloudFormation this is a SAM template it needs to “transform”

Creates a AWS Lambda function with the referenced managed AWS IAM policy, runtime, code at the referenced zip location, and handler as defined. Also creates an Amazon API Gateway and takes care of all mapping/permissions necessary

Creates a Amazon DynamoDB table

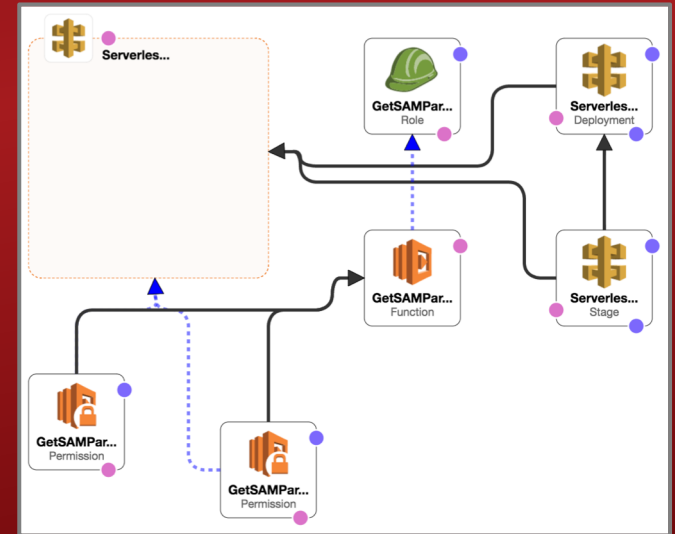
AWS SAM Templates

26 lines (25 sloc) | 827 Bytes

Raw Blame

```
1 Transform: 'AWS::Serverless-2016-10-31'
2 Parameters:
3   SamMultiplier:
4     Description: "SAM multiplier. Make this really big to have a party :)"
5     Type: "String"
6   OriginUrl:
7     Description: "The origin url to allow CORS requests from. This will be the base URL of your stat
8     Type: "String"
9 Resources:
10  GetSAMPartyCount:
11    Type: AWS::Serverless::Function
12    Properties:
13      Handler: index.handler
14      Runtime: nodejs4.3
15      CodeUri: ./
16      Environment:
17        Variables:
18          SAM_MULTIPLIER: !Ref SamMultiplier
19          ORIGIN_URL: !Ref OriginUrl
20    Events:
21      GetResource:
22        Type: Api
23        Properties:
24          Path: /sam
25          Method: get
```

← This, Becomes This



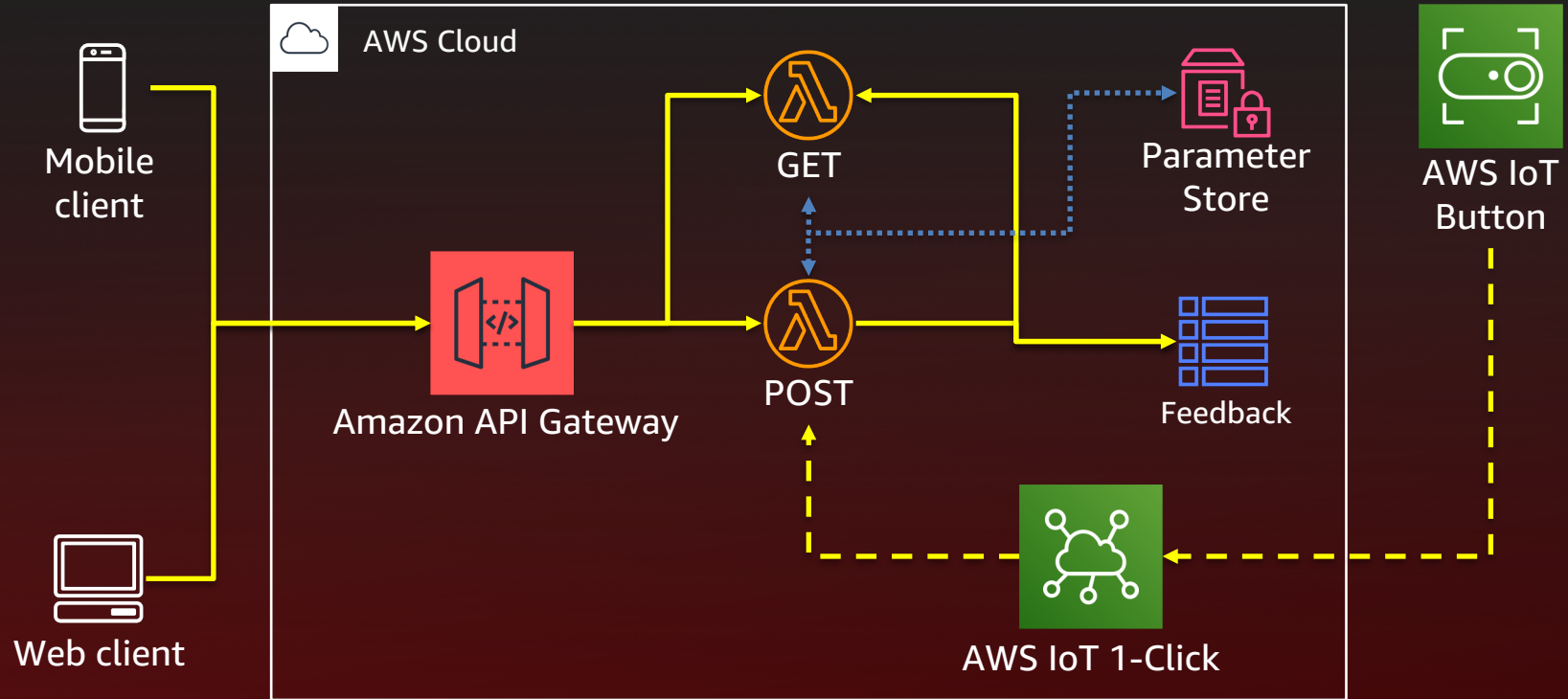
We want
code!!!



See it in Action!

<http://bit.ly/goto-feedback>

What We Accomplished (Hopefully)



Source Code

Backend:

<http://bit.ly/iot-fb-back>

Front End:

<http://bit.ly/iot-fb-front>



Please

**Remember to
rate this session**

Thank you!

goto;
chicago



Click 'Rate Session'

Rate **5** sessions to get the
supercool GOTO reward