

Monitoring and Securing Amazon Translate



Fernando Medina Corey

CLOUD ARCHITECT

@fmc_sea www.fernandomc.com



Overview



Monitoring basics

- Why use monitoring?
- AWS monitoring tools

Demo – Monitoring Amazon Translate Metrics in the AWS Console

Metrics and alerting

- Amazon Translate metrics
- AWS alerting options

Demo – Creating CloudWatch Alarms for Amazon Translate Errors

Access control

Demo – Configuring an Amazon Translate User in the AWS console



Monitoring Basics



Why Use Monitoring?

Error detection

Issue investigation

Cost control and
management

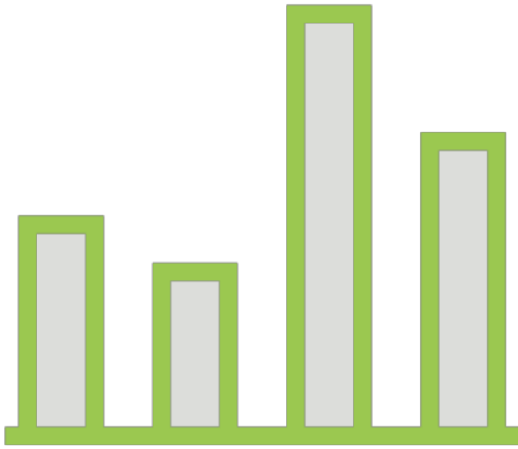
Abuse detection

Product planning

System
optimization



Monitoring Tools



CloudWatch metrics

Gather data from various AWS services



CloudWatch alarms

Configure alarms based on the data you collect

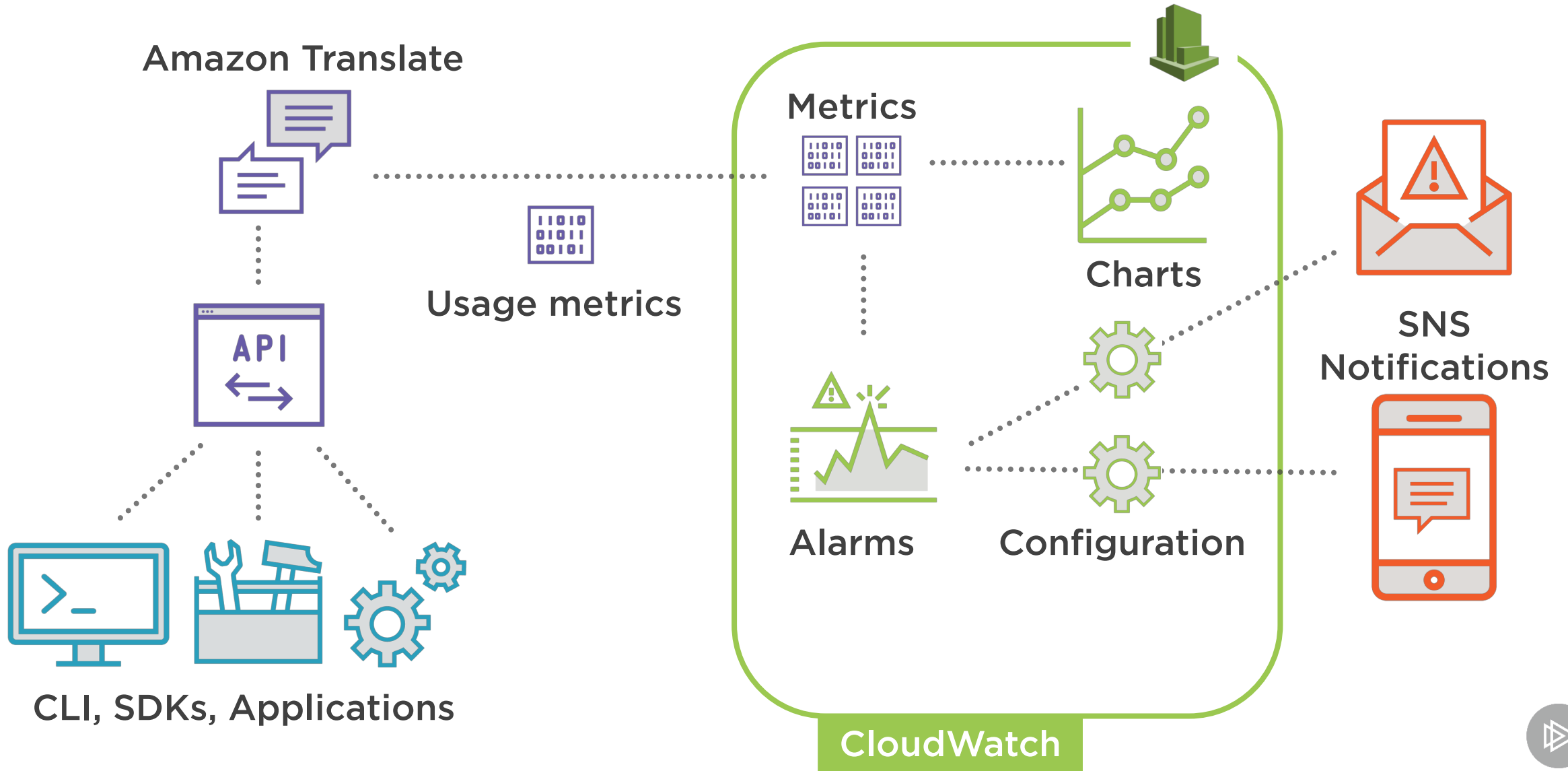


Alerting integrations

Control various forms of alerting based on the alarms you set



AWS Monitoring Explained



How to Start Monitoring Amazon Translate



Start using Amazon Translate



Review the Amazon Translate metrics in the AWS console



Think about additional things to monitor later on



Demo



Monitoring Amazon Translate Metrics in the AWS Console

- Run a script to use Amazon Translate
- Script will generate metrics for us to view
- Review several different metrics



Metrics and Alerting



Monitorable Amazon Translate Metrics

Character count

Server errors

Throttled requests

Response time

Successful
requests

User errors



Amazon Translate CloudWatch Dimensions

Language Pair

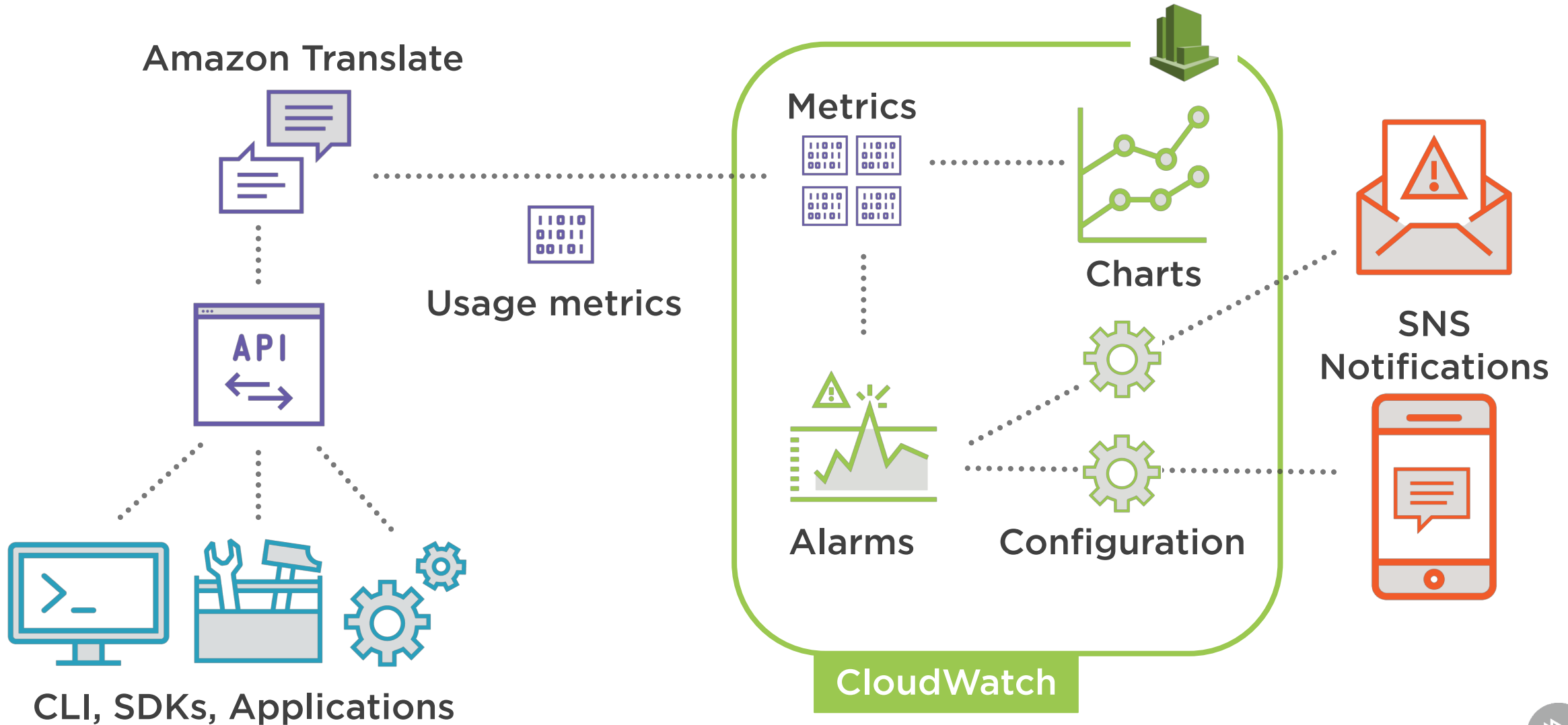
Restrict metrics viewed to those in the particular languages

Operation

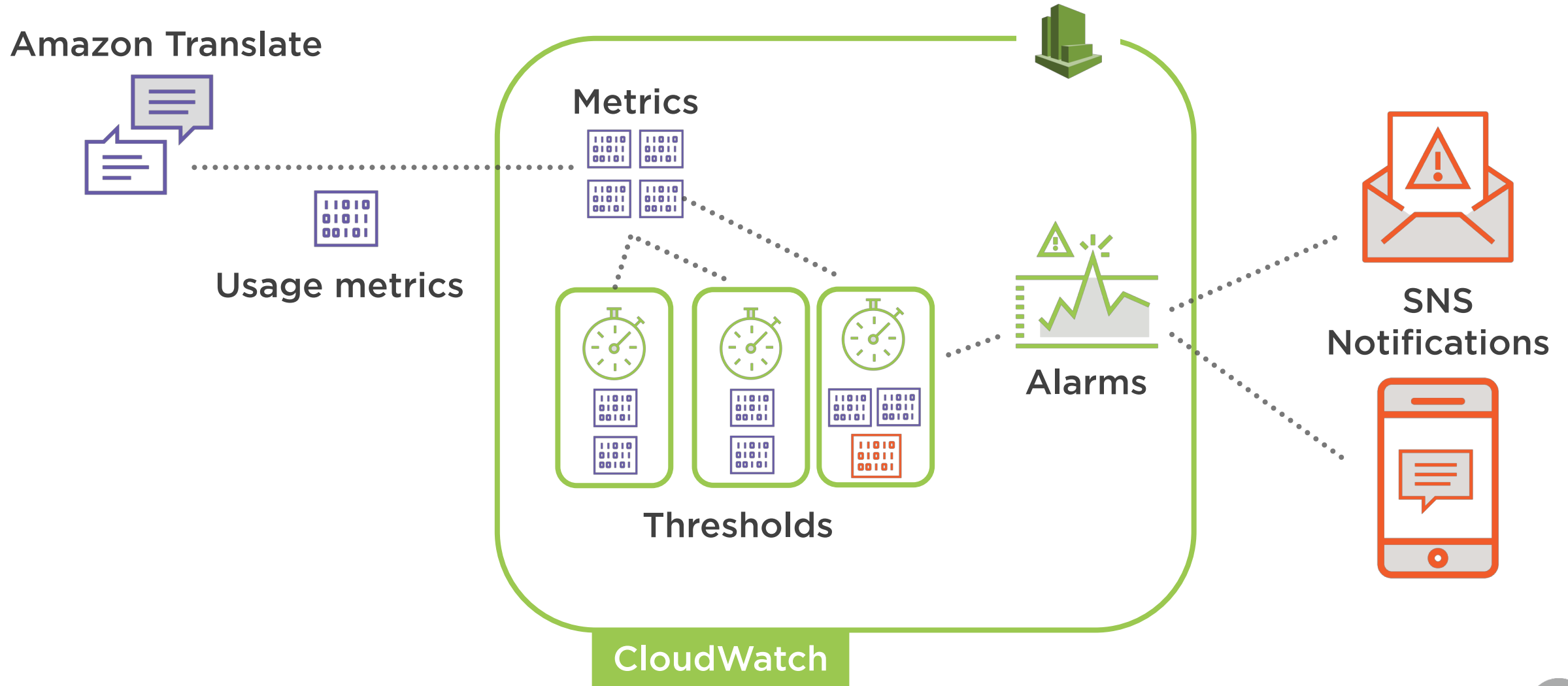
Restrict metrics to only view the ones for a specific operation



AWS Monitoring Explained



AWS Monitoring Explained



Why Monitor Amazon Translate Applications?



Check for misconfigured applications
(High amounts of throttled data, or high failure rates)



Keep tabs on potentially bad user experiences
(Slow response times, errors)



Showcase new use patterns
(Why are particular language pairs suddenly popular?)



Demo



Creating CloudWatch Alarms for Amazon Translate Errors

- Create alarm for high request failures
- Create an SNS email notification



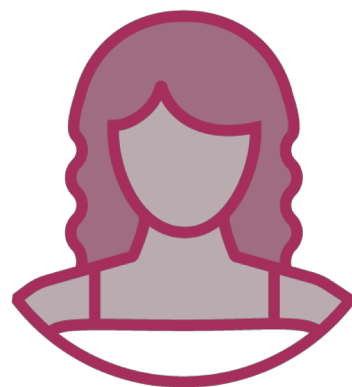
Access Control



AWS Identity and Access Management



Policies



Users



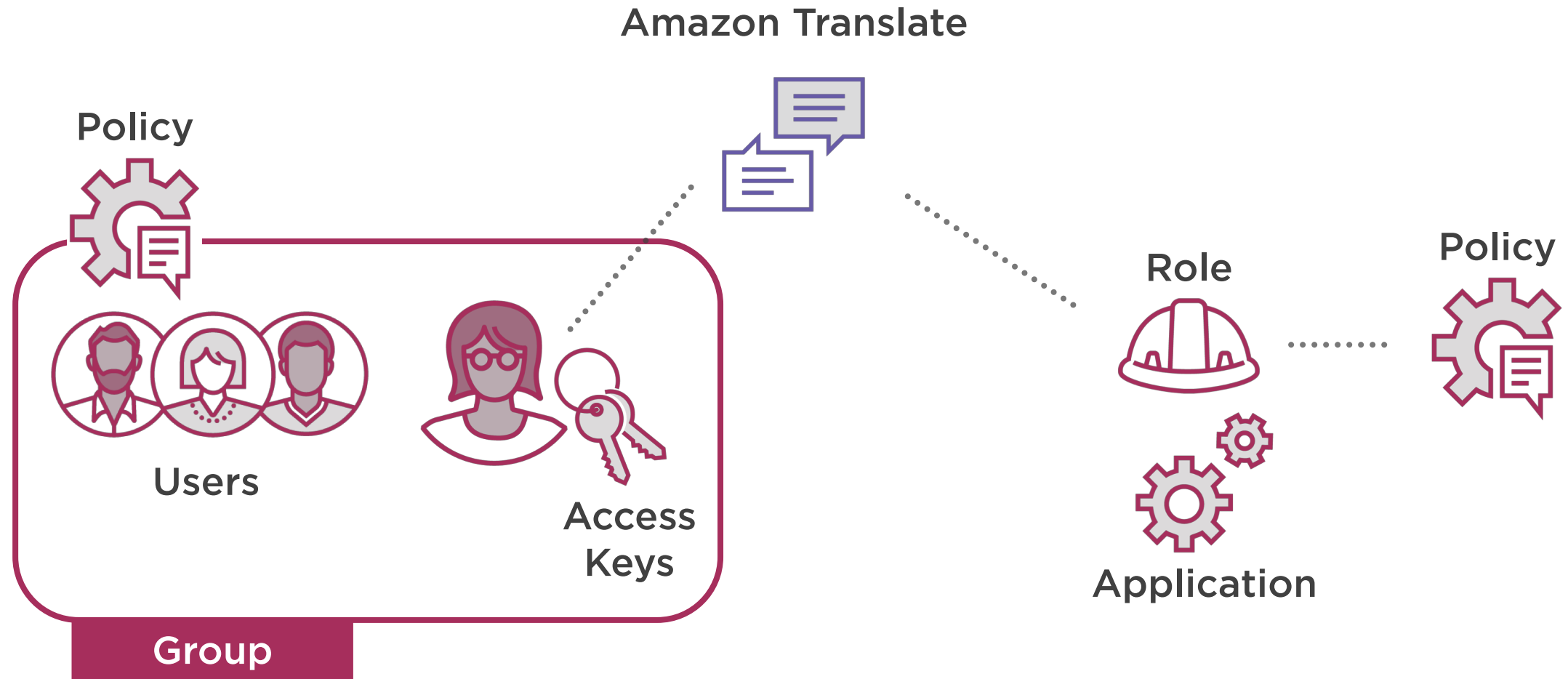
Groups



Roles



IAM Visualized



```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "translate:*",
      "comprehend:DetectDominantLanguage",
      "cloudwatch:GetMetricStatistics",
      "cloudwatch:ListMetrics"
    ],
    "Resource": "*"
  }]
}
```

- ◀ Determines IAM Policy features used
- ◀ Allow these actions
 - ◀ Everything under Amazon Translate
 - ◀ Also one API call from Amazon Comprehend
 - ◀ Two API actions under CloudWatch to view metrics
- ◀ Allow these API calls on all resources



Why Are Policies Important?



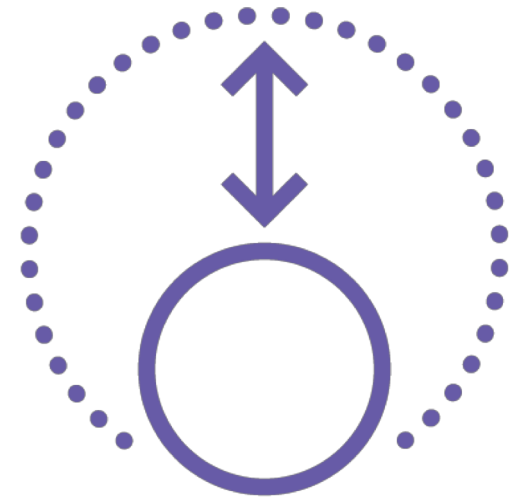
Default deny

AWS denies access to everything by default



Security

Knowing and controlling who can do what at any time



Least privilege

Reduce risk and access to AWS services and APIs



Demo



Configuring an Amazon Translate User in the AWS Console

- Create an AWS user with IAM
- Sign into AWS with that user
- Access Amazon Translate
- Attempt to access other services



Summary



Monitoring and alerting

- Why we would use CloudWatch
- How we can use metrics and alarms
- Creating alarm notifications

The importance of access control

- Basic IAM concepts
- Leveraging IAM users, groups, and policies with Amazon Translate

