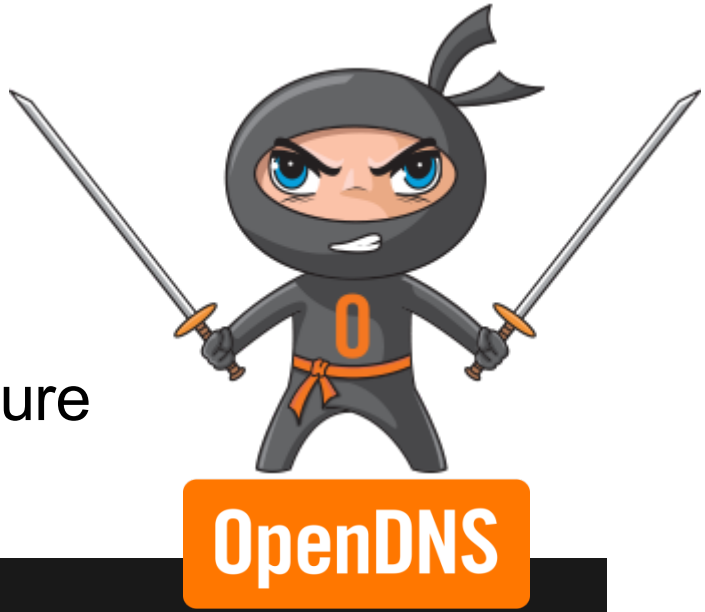
PLEASE GO TO THE FOLLOWING URL AND SETUP THE PRE-REQS + SECTION 0 IF YOU HAVEN'T ALREADY!!

https://www.chrisdorros.com/2016/oscon



Don't Fix It, Throw it Away!

Introduction to Disposable Infrastructure

2016-05-17, OSCON 2016

Chris Dorros

OpenDNS is now part of Cisco.

Tentative Schedule

1:30	was a few minutes ago
2:00ish	Section 1
2:30ish	Section 2
3:00-3:30	break
3:30	Section 3
4:00ish	Section 4
5:00	end time, stick around for questions/help

Infrastructure as Code

- Defining **infrastructure** (servers, networking, operating systems, processes) **as code**, and **provisioning those components automatically** from the code
- **Declarative** language (in general)

Disposable? Immutable?

- *Disposable* - throw your server away, make a new one
  - (recycle/compost if you're in SFO)
- *Immutable* - no changing the server after it's already deployed.

“Pets vs Cattle”

rack n’ stack,
cloud GUI

entire infra in code;
deployments via API

Why?

- Higher confidence in changes
  - since we're switching to new infrastructure for each change, it's very simple to rollback
  - great for security patches!
- Documentation without knowing we're documenting!
  - infrastructure as code, instead of infrastructure as procedures and tribal knowledge

This shouldn’t scare you!

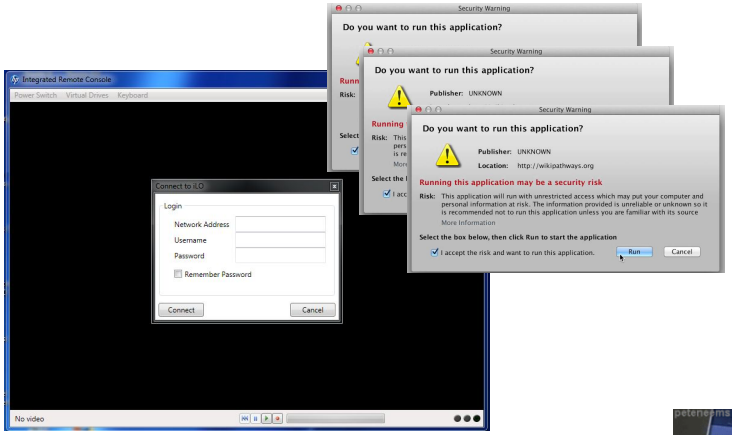
---------- Forwarded message ----------
From: **'Amazon EC2 Notification'**
Date: Wed, Dec 31, 2014 at 5:59 AM
Subject: [eng.aws-role] Amazon EC2 Maintenance

Dear Amazon EC2 Customer,

One or more of your Amazon EC2 instances is scheduled for maintenance on 2015-01-06 for 2 hours starting at 14:00 UTC. During this time, the following instances in the us-west-1 region will be unavailable and then rebooted:

i-982a65c5

Deployment Workflow: Pre-Cloud

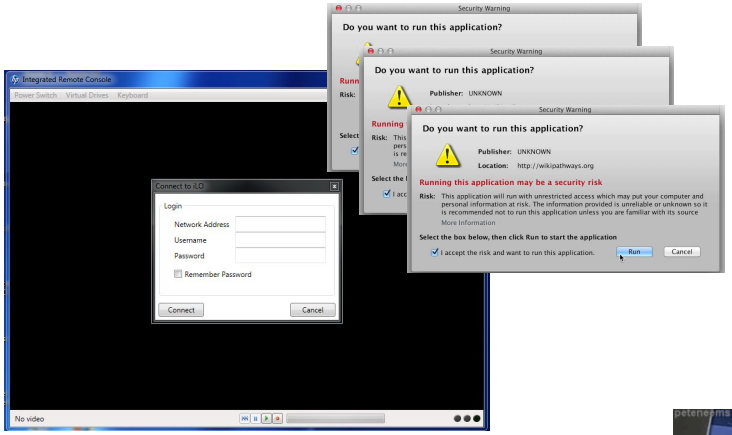


Remote KVM

$ uptime 12:40 up 4 years 22 days

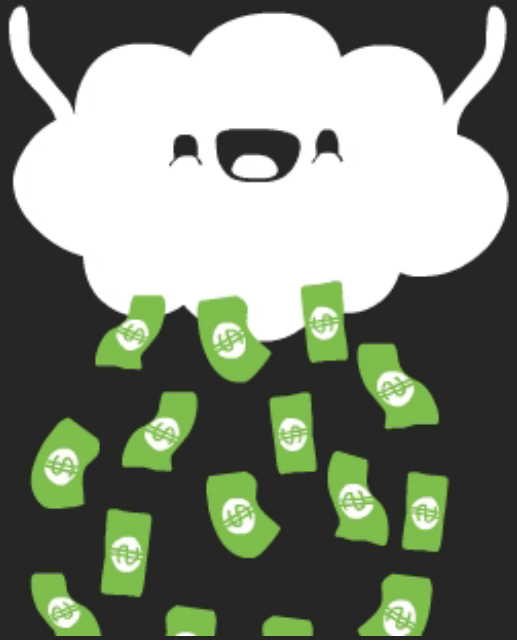
Configuration Management

Deployment Workflow: Pre-Cloud

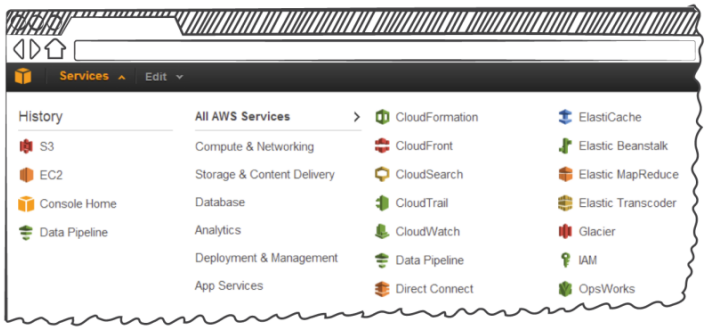


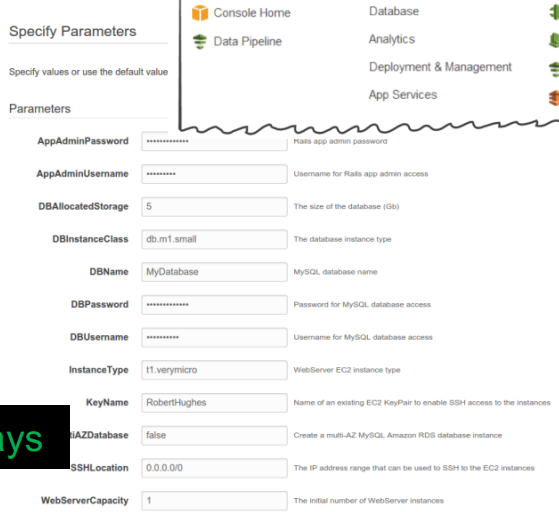
Remote KVM

$ uptime 12:40 up 4 years 22 days



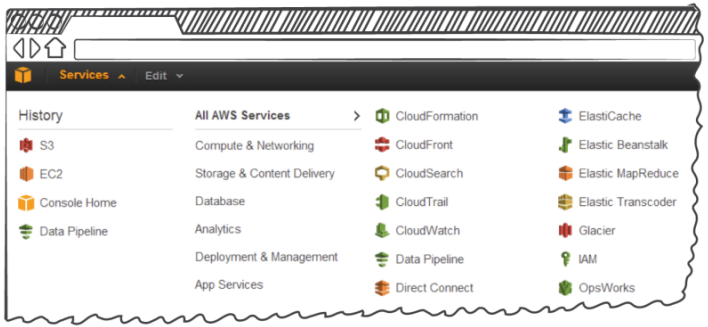
Deployment Workflow: Post-Cloud

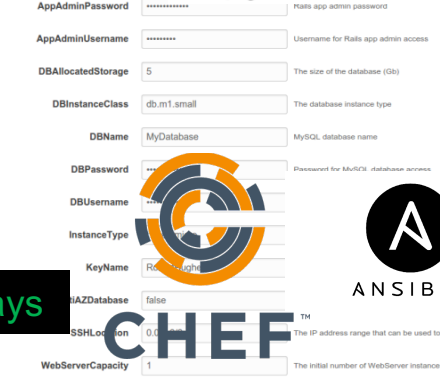




$ uptime 12:40 up 4 years 22 days

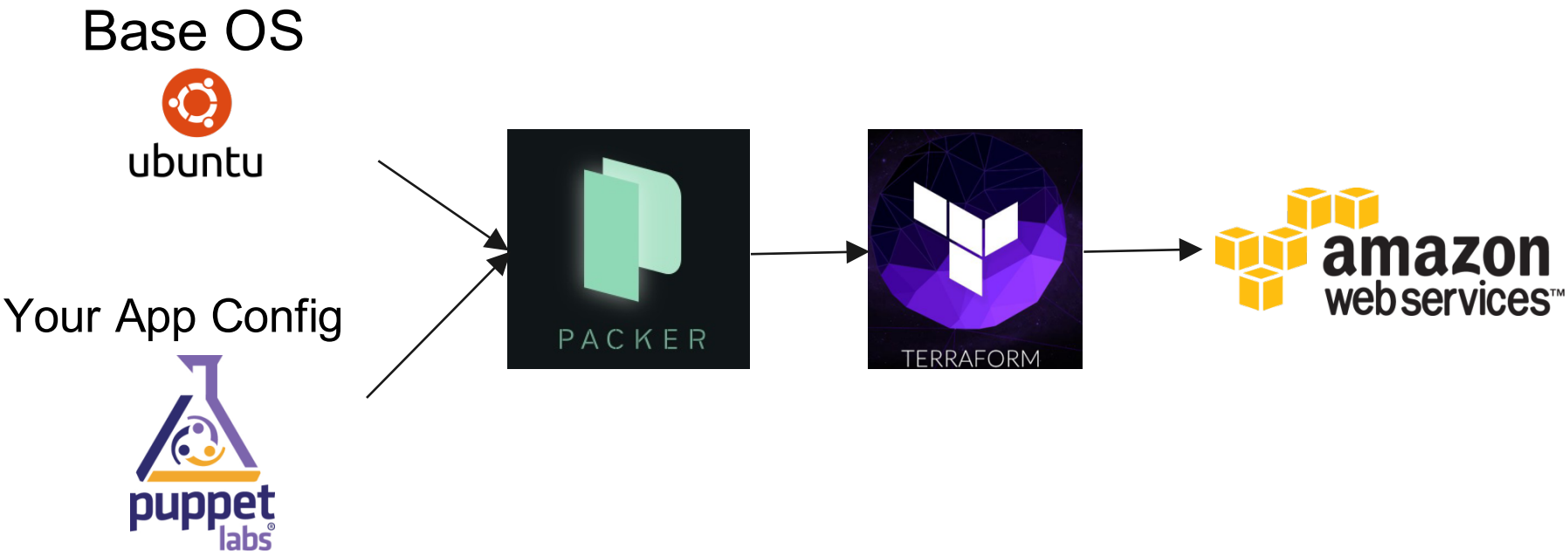
Deployment Workflow: Post-Cloud



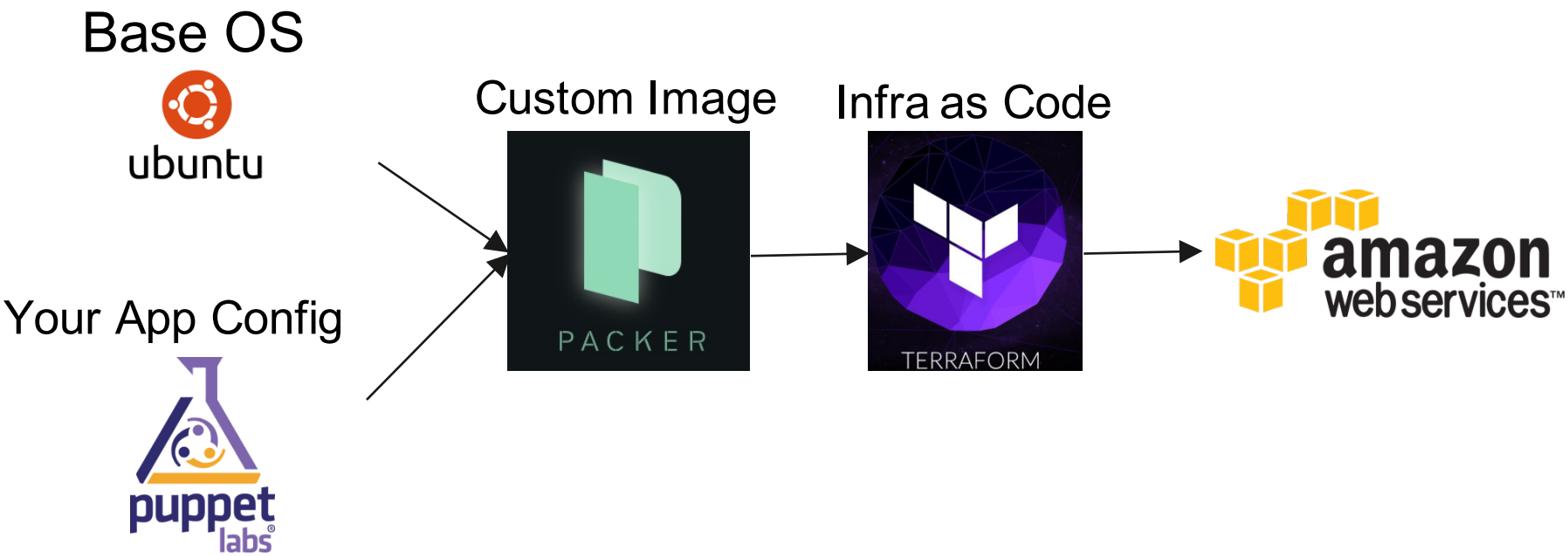


$ uptime 12:40 up 4 years 22 days

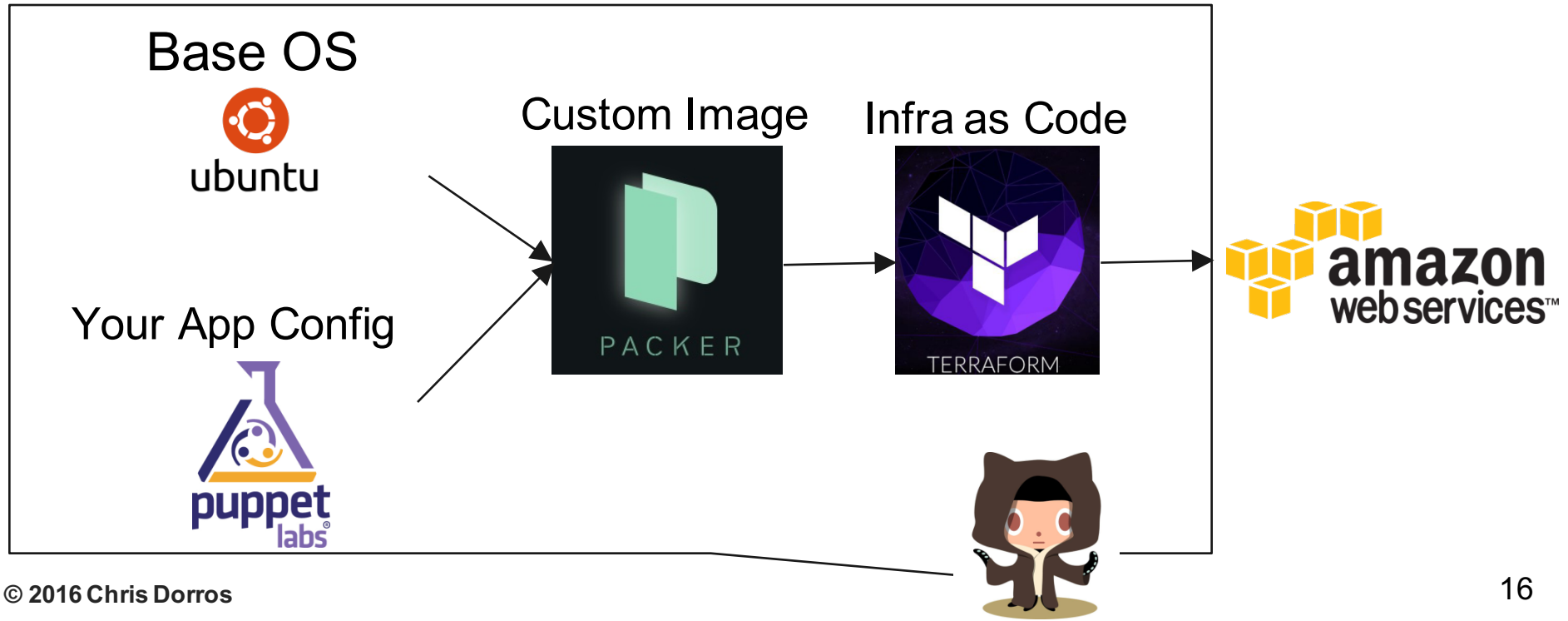
Deployment Workflow: Today's Exercise



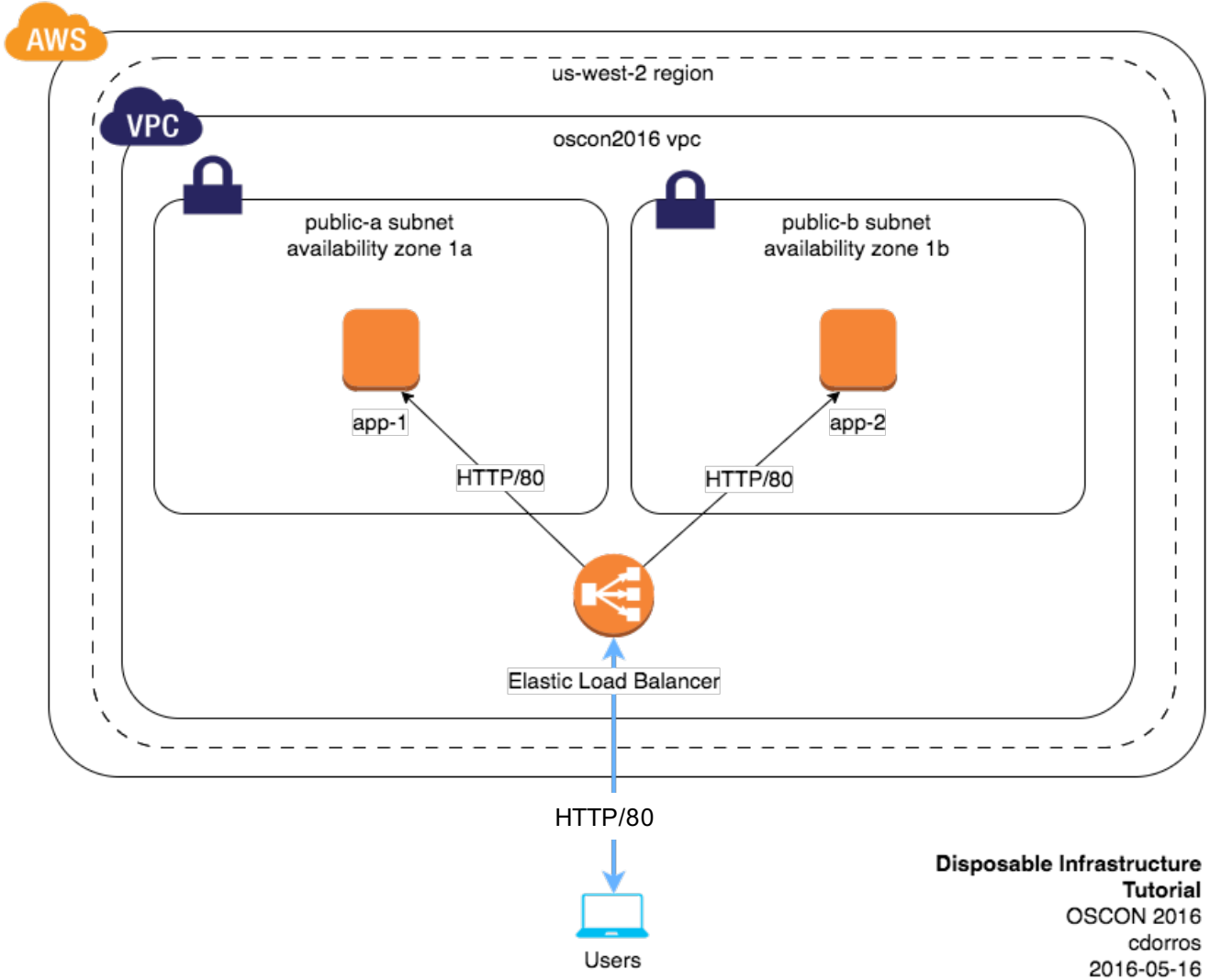
Deployment Workflow: Today's Exercise



Deployment Workflow: Today's Exercise

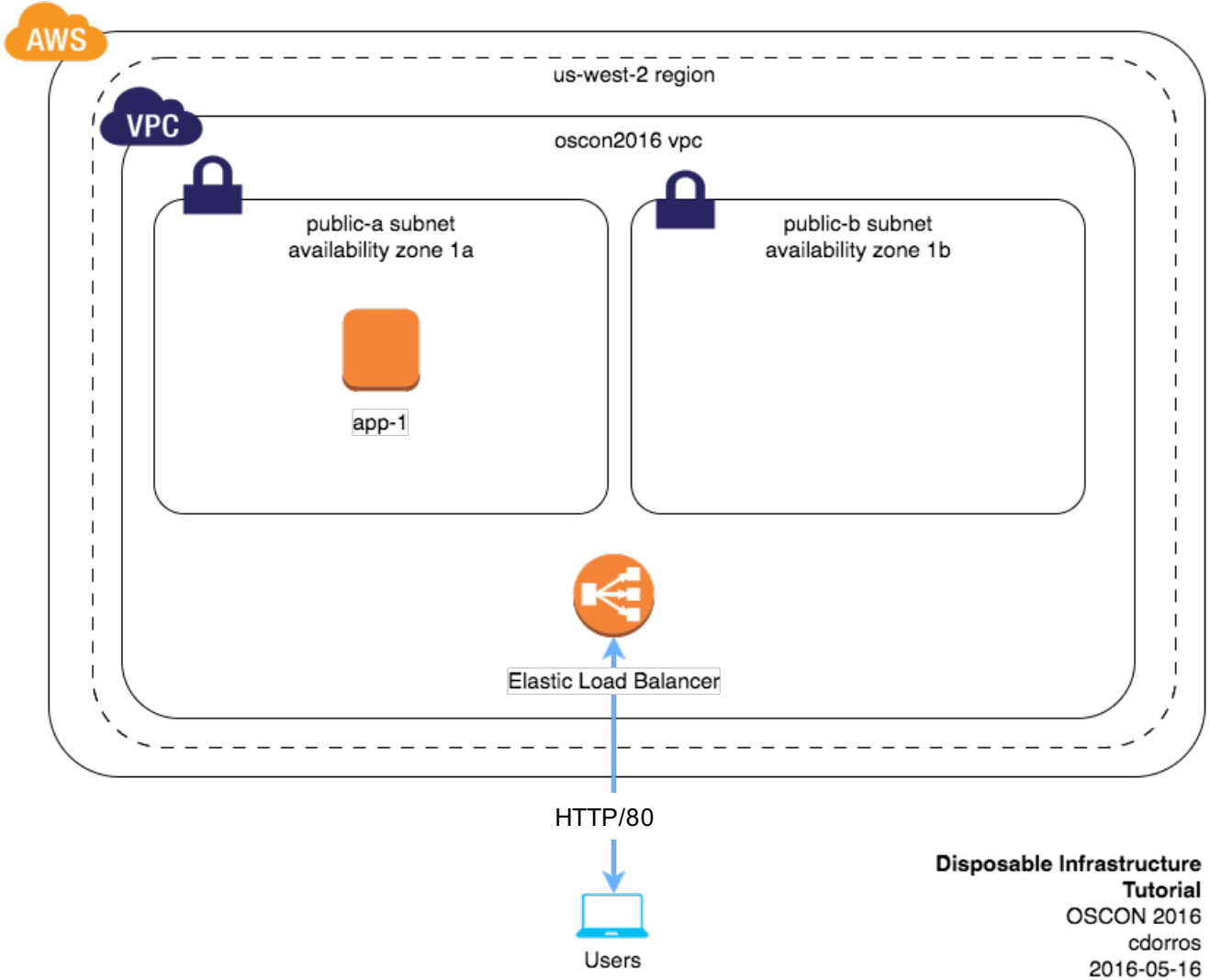


What we're building today



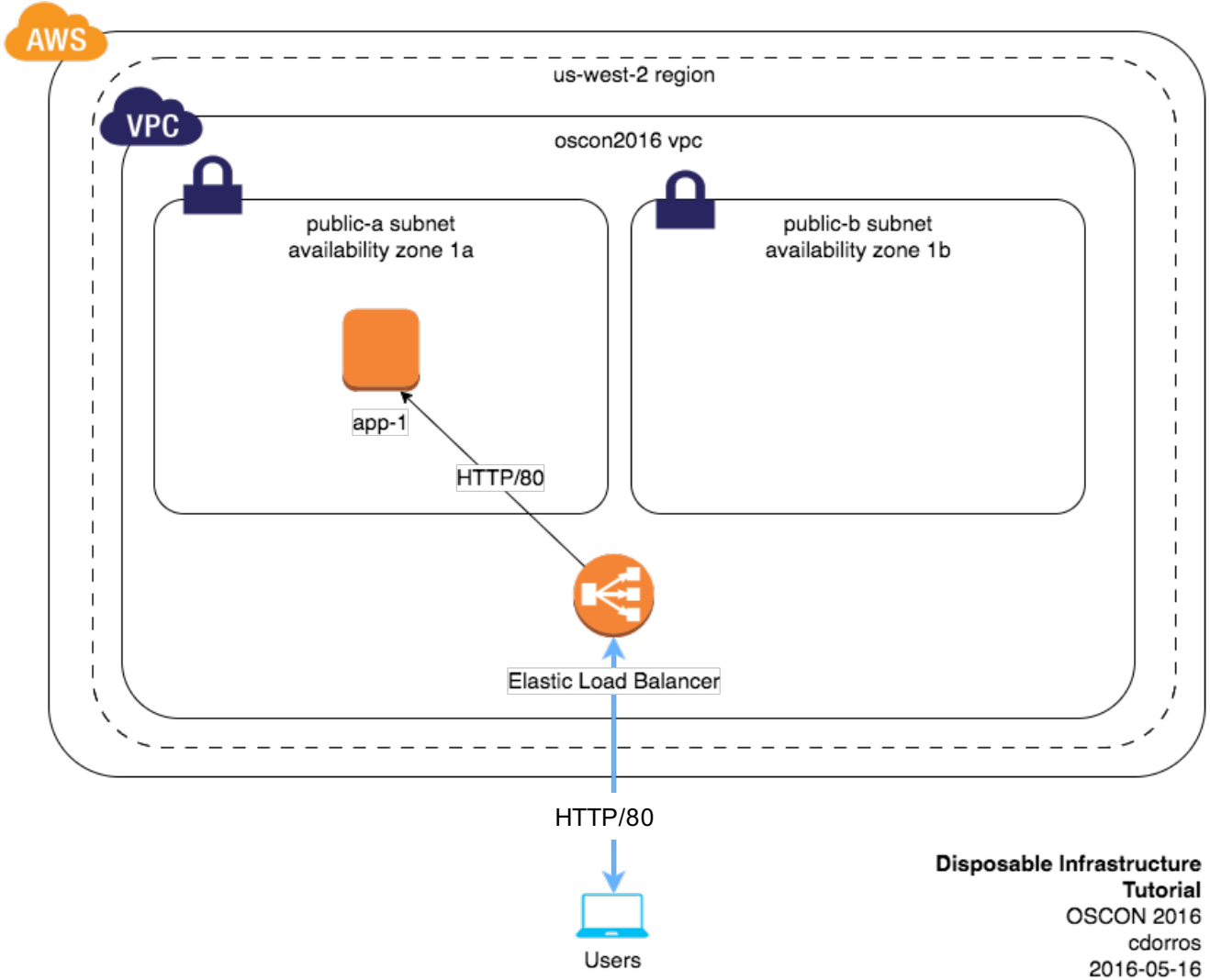
Disposable Infrastructure Tutorial
OSCON 2016
cdorros
2016-05-16

Section 3 (app-1 out of rotation)



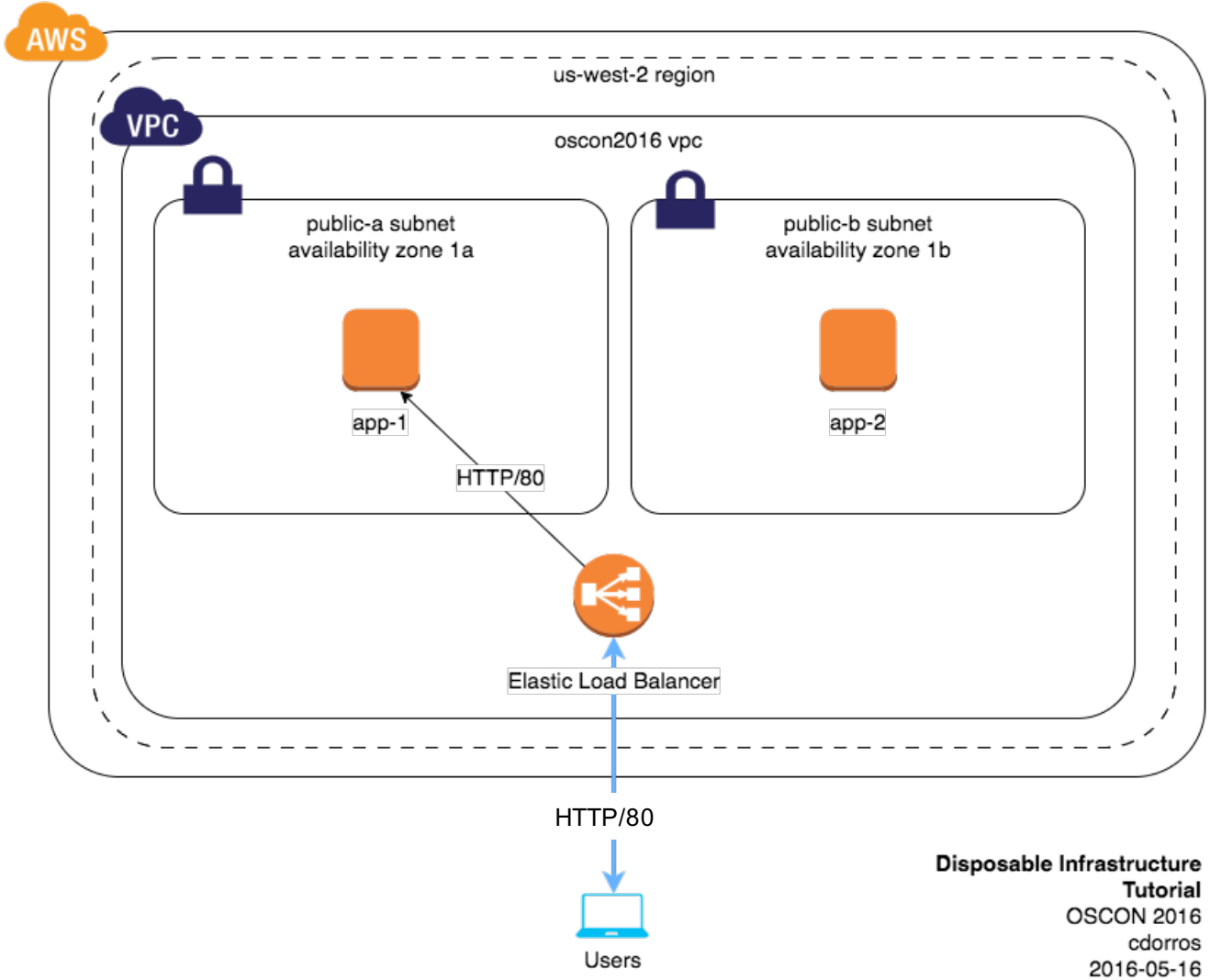
Disposable Infrastructure Tutorial
OSCON 2016
cdorros
2016-05-16

Section 3 (app-1 in rotation)



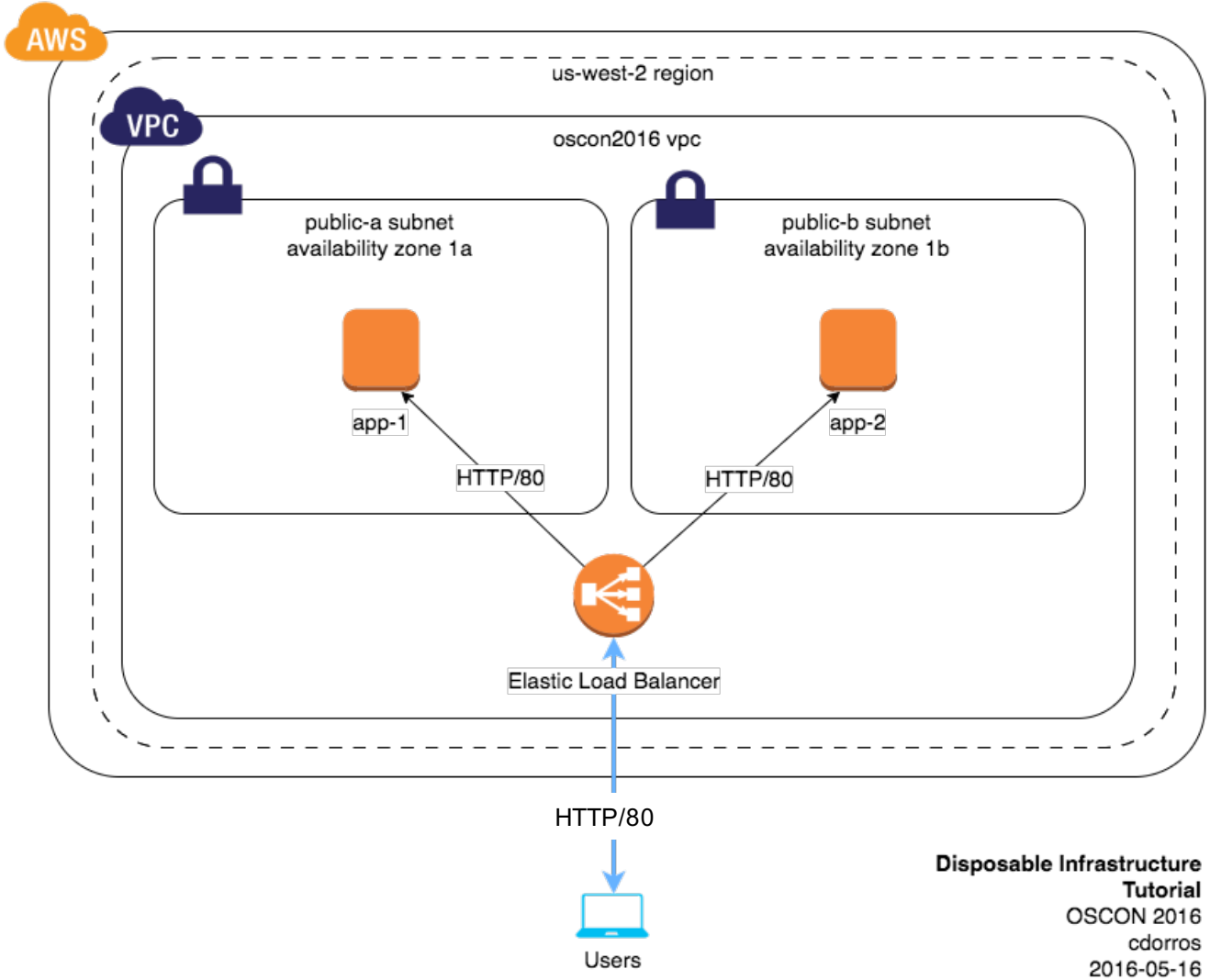
Disposable Infrastructure Tutorial
OSCON 2016
cdorros
2016-05-16

Section 4 (app-1 in rotation)



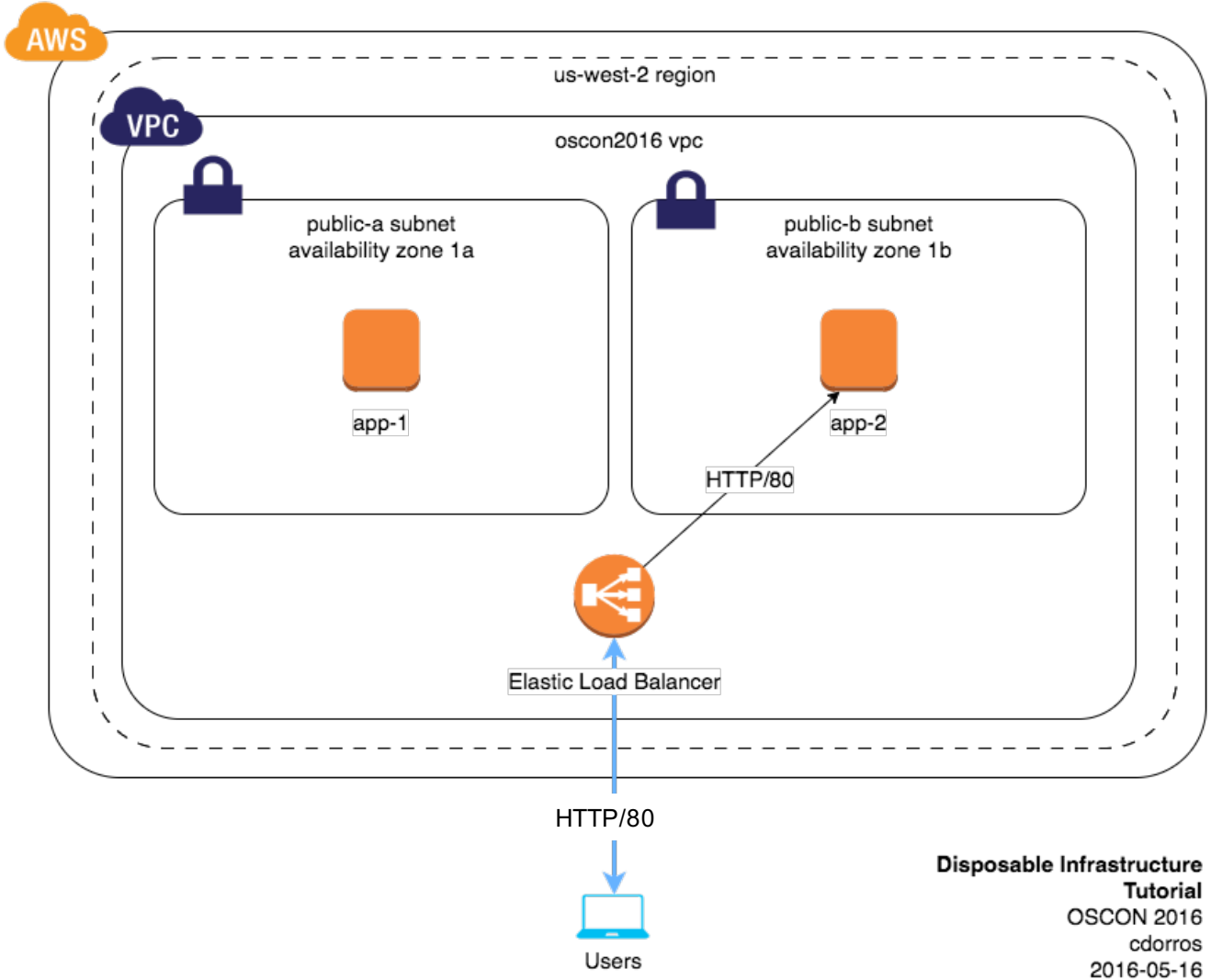
Disposable Infrastructure Tutorial
OSCON 2016
cdorros
2016-05-16

Section 4 (both versions of the app in rotation)



Disposable Infrastructure Tutorial
OSCON 2016
cdorros
2016-05-16

Section 4 (app-2 in rotation instead of app-1)



Disposable Infrastructure Tutorial
OSCON 2016
cdorros
2016-05-16

Where do I go from here

- Experiment with it!
  - Try adding a fake SSH key to the AWS EC2 instance during build to force yourself not to login
- Dev / Prod parity
  - Re-using Terraform configuration across both
    - 1. “terraform apply” in dev
    - 2. switch AWS keys to point to prod
    - 3. “terraform apply” in prod

Thank you!

Feedback, or just to update on applying these techniques back @ work:

chris[at]chrisdorros[dot]com