Canadian Digital Service
Service numérique canadien

Canada

DevSecOps at CDS

Launching a PBHH service in 45 days and running it over 18 months with a team of two

Who are we?

Introductions

Calvin Rodo

Max Neuvians

Covid Alert Server

What are we going to be talking about?

Two parts to the presentation:

1. How we were able to launch the service in 45 days and get a PBHH assurance

2. How we scaled, secured, and kept the service reliable with a team of two developers

What is Covid Alert Server

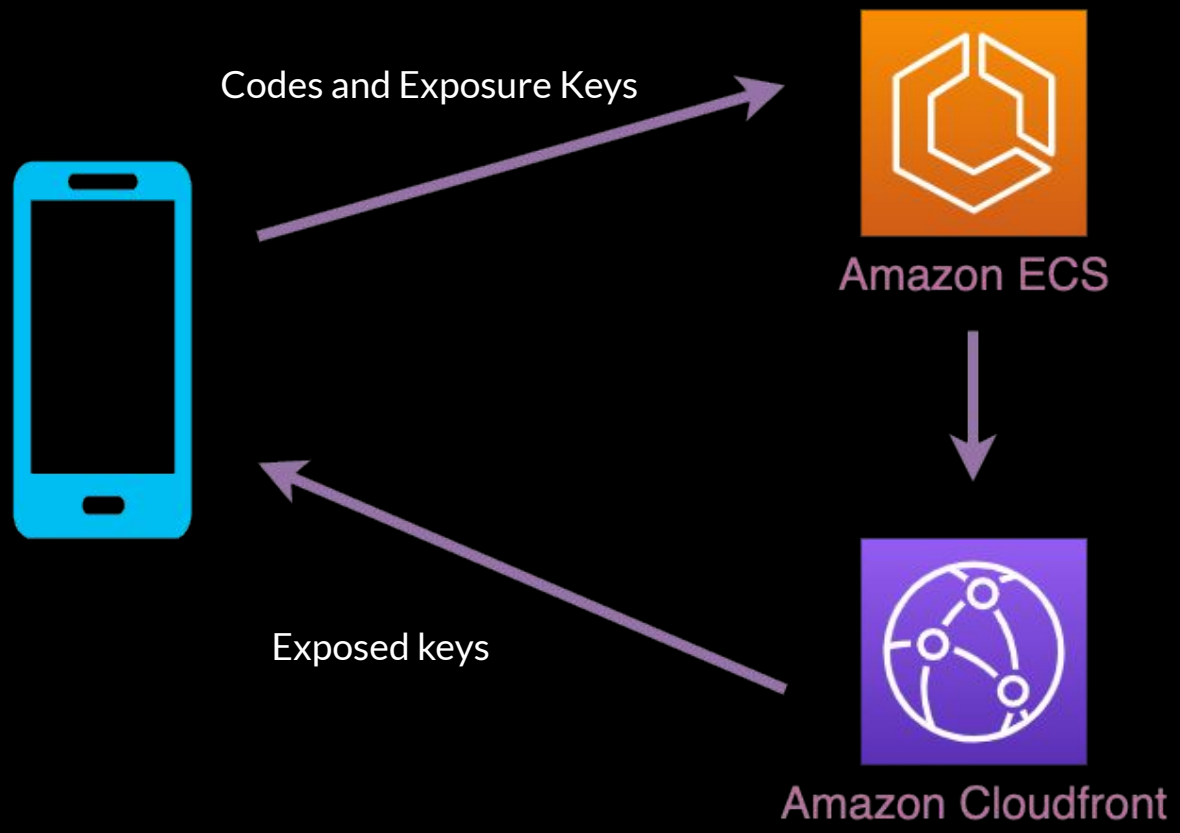
Canada's implementation of the Google / Apple Exposure Notification Framework.

Consists of three components:

- Mobile App
- Server infrastructure
- Portal for healthcare professionals.

We will focus on backend infrastructure and what we put in place so it could be supported by a small team of 1 to 2 developers.

How it works


Codes and Exposure Keys
Amazon ECS
Exposed keys
Amazon Cloudfront

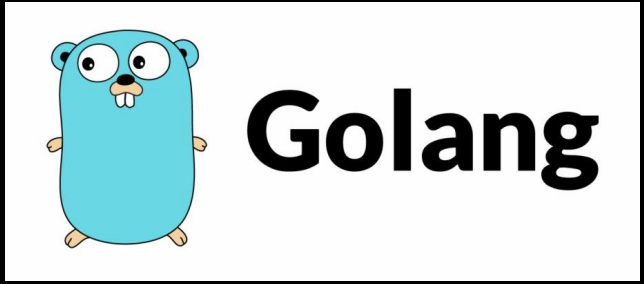
Part 1: Launch challenges

Challenge #1

We inherited proof-of-concept code from Shopify in unfamiliar tech

Solution:

- Rigorous unit testing
- Fuzzing API endpoints
- Writing external test harnesses
- Pull requests with branch testing
- Code coverage
- Static code analysis



Challenge #2

We needed to meet the Protected B, High Availability, High Reliability level

Requirements:

- Encryption
- Boundary protection
- Available
- Reliable
- Responsive

Solution:

- Managed keys and certificates
- WAF, DDoS, no network egress
- Multiple zones, microservices
- Blue / Green deploys
- Global content caching

Challenge #3

Limited developer capacity (1-2 full time developers)

- Outside developers need to be able to validate the the code based on consistent automation.
- Review environment that exist inside the branch to be reviewed (Heroku)
- Dev container which allow other developers to work with the same tools

8 checks passed

- ✓ key-submission
- ✓ security
- ✓ testing
- ✓ key-retrieval
- ✓ [container-scan] key-retrieval:2500a38495f5b4d29dd6437e60354ec2d222fe18 Container sc...
- ✓ [container-scan] key-submission:2500a38495f5b4d29dd6437e60354ec2d222fe18 Container ...
- ✓ license/snyk (ExposureNotification) No license issues in 7 tests
- ✓ security/snyk (ExposureNotification) No manifest changes detected in 7 projects

maxneuvians deployed to covid-shield-heroku-pr-anqbete 15 months ago

View deployment

Challenge #4

External review from CCCS and Blackberry

- Agreed to only address “Critical” and “High” vulnerabilities before launch
- All issues managed transparently and publically through GitHub issue
- Proof of remediation demoed in review / developer environments
- Most common issues were Time of Check / Time of Update (TOCTOU) bugs
- Hired external pen tester (caveat)

Challenge #5

Working completely in the open and transparently

- We offered people to use alias GitHub accounts in case the felt uncomfortable
- We added mechanisms for the public to provide feedback in our pull requests
- We provided Codes of Conducts and vulnerability disclosure policies

Challenge #6

Fast release process to coordinate with other components (app, portal, provinces)

- Staging environment that was an exact replica of production (same infrastructure as code, same types of resources)
- Sample code to interact with the API
- Safe secrets sharing (https://secret.cdssandbox.xyz/)

Part 2: Keeping it alive

First some numbers...

- Over a billion API requests served

  (20 million hits a day)

- 150 Terabytes of data sent
- Over 130 changes to production environments
- 0 downtime during releases
- ~~40~~ 41 Incident Reports generated

Challenge #7

Everything is broken, and that's Okay.

- What matters to us is how broken are things.
- We create Service Level Indicators (SLI) things we can measure things in our system that matter to us.
- We define Service Level Objectives (SLO) or target values that tell us what acceptable values are for our SLIs
- These aren't set in stone

Challenge #8

No, really it's normal for things to break

- Treat every incident as a learning experience
- Document incidents as they happen in an incident report
- Review every single incident report as a team
- Incident Reviews result in items being added to the backlog
- Incident Reports should be available to everyone.

"It's not Jim's fault he deleted the database, he should never have been able to delete it in the first place."

Useful Resources

- Increment Magazine:
  - https://increment.com/reliability/
- Google SRE Books
  - https://sre.google/books/

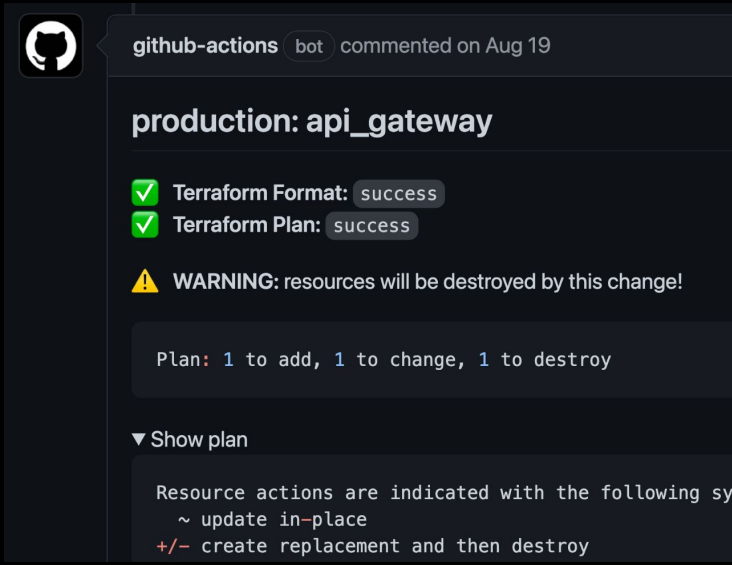
Challenge #9

Making changes to infrastructure is risky.

- Infrastructure as Code
  - Terraform
  - Terragrunt
- Version our infrastructure
- Use the tools we are used to
- Code review on all changes
- Use automated checks to validate our infrastructure

Helpful tools

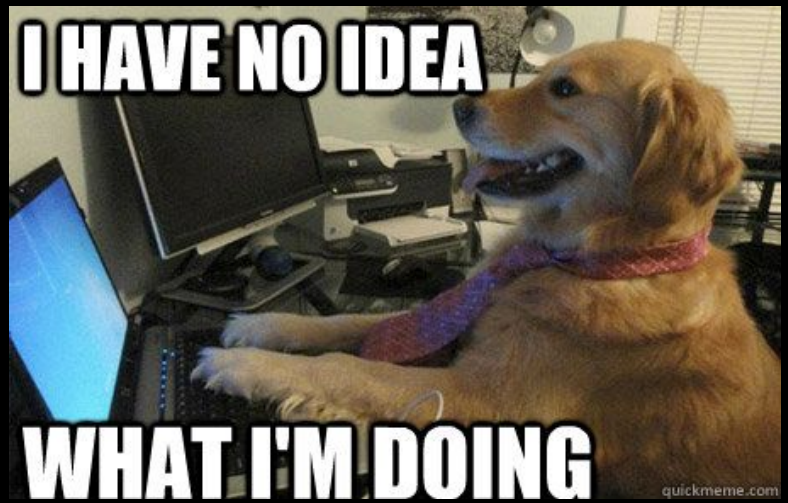
- Checkov
- TFSec
- CDS Terraform Plan Action
  - github.com/cds-snc/terraform-plan
- Awesome Terraform List
  - github.com/shuaibiyy/awesome-terraform



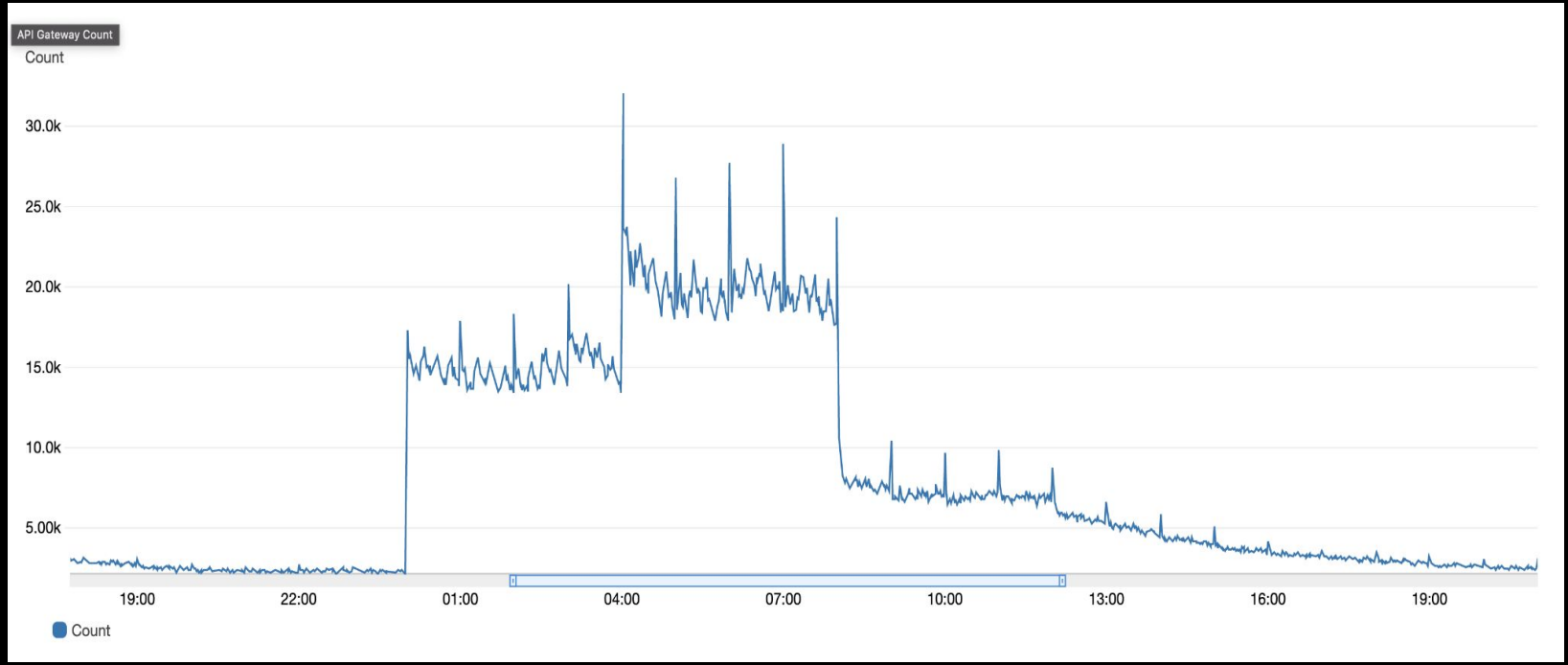
Challenge #10

We aren't that good at running infrastructure

- We are developers not server administrators
- CSPs are a lot better than we are at running infrastructure.
- Reduces the number of security controls we need to deal with.
- Reduces the need to come up with a patching strategy.



The Metrics Roller Coaster



Recommendations

- Build feedback loops
- Automate everything you can
- Continuously pay down tech debt
- Run an incident anytime something breaks
- Take advantage of managed services whenever possible.

CDS
SNC

Thank you!

Merci!

digital.canada.ca · @CDS_GC | numerique.canada.ca · @SNC_GC