pinger

Pinger, Inc

The 40 second prod environment

You don’t have to share anymore!

Adam Robertson, Head of DevOps
Kristian Windsor, DevOps Engineer

DEVOPS WORLD
by CloudBees

pinger

Question of the day.

How do we provide the devs and QA folks the environments they need to do their jobs?

DEVOPS WORLD
by CloudBees

Often heard around the office...

“Well it worked on my machine!”

“Who deployed over my code?!”

“How could it fail on prod, it worked on Staging?!”

“I can’t test it until QA is freed up”

"It's a config issue. Code is fine"

"See this? This is why we can't have nice things."

Typical set of environments

1. Local (Vagrant)
2. Dev Server (Bare Metal)
3. Build/Test System (Docker Container in Jenkins)
4. QA Server (Bare Metal)
5. Staging Server (Bare Metal)
6. Production (Tons o' bare metal)

6 Different environments!

Tensions were high...

The testing environments had a few issues

- Multi-tenant
- Unique little snowflakes
- Shared with team on same sprint
- Set up manually, years ago
- Limited/fixed resources
- Broken for one, broken for all
- Change is difficult

- Developer context switching
- Fragile
- Code being overwritten
- Config creep over time
- Stale data in database
- Hardware issues
- Inconsistent starting state

We wanted something better...

What are our "Northstar" requirements?

- Stability
- Reproducible
- IAC - Infrastructure As Code
- Scalable
- Single tenant
- Customizable
- Automated
- Fast, like really fast
- Easily to maintain and control
- Consistent starting state

pinger

What was our solution at Pinger?

Let’s go back to the blissful days of getting what we want, when we want it without having to share it...

DEVOPS WORLD by CloudBees

The 40 second prod environment

- Independent Service
  - Can be deployed via pipeline or manually
- Isolated
  - Not fighting for resources
  - Not affected by busy neighbors
  - Distinct: one time use
- Scalable and Highly Available
  - Built on Kubernetes on AWS Cloud
  - No limit (except our wallet)
  - Cleans up after itself

- Consistent
  - Build from the same image
  - Repeatable
- Fast
  - Containers that launch in seconds
- Easy to use
  - Customizable
  - Can be integrated into testing pipelines
  - Environment can correlate with a feature branch
  - Manually launched to investigate bugs

Demo time!

As easy as 1, 2, CURL!

40s prod demo recap:

- Kubernetes Resources
  - Deployment, service, ingress, persistent volumes
- Parameterized config
  - Tailor the service for your situation
- Public DNS
  - Service available on all devices (mobile!)
- Stateful Service
  - Service runs independent of the API
  - Service dies on TTL expiry

What is the value of such a system?

Yeah, it's cool.

But how is this valuable to my team?

Benefits

pinger

Benefits

- Increases
  - Build frequency
  - Developer velocity
  - Consistency
- Decreases:
  - Individual build times
  - Overall pipeline execution time
  - Human touchpoints

- Let's not forget sidecars
  - Monitoring
  - Telemetry
  - Logging
  - Stats
- This one's mine! They are all mine!
  - Each dev can have any number of custom tailored instances
  - No sharing!!

*** At Pinger we launch on average 200 of these production environments each week.**

'Gotchas' and pain points...

- Cloud provider limitations are your limitations
- Kubernetes is hard at scale
- Your users will want to abuse it as much as you let them

Conclusion

- No more sharing!
- No more waiting on “your turn”
- Focus on code, not environment

DEVOPS WORLD by CloudBees

Now you know how to spin up a production like personal QA in 40 seconds!

40 Second Prod

Q&A