DevOps for Startups

HashiCorp

Armon Dadgar

@armon

What is DevOps?

DevOps Definitions

- “DevOps is you have developers do everything”
- “DevOps is you get rid of operations”
- “DevOps is a cultural movement”
- “DevOps is …”

HashiCorp

Delivering an Application

- Software organization is a *system* like any other
- Composed of people, processes, and tools
  - Processes used to organize people
  - Tools used to support people and process
- Output is applications

People

Specialized Knowledge
Limited time

People

Developer

Programming Languages
Frameworks
Design Patterns
Application Architecture
...

People

Operator

Cloud APIs
System Administration
Infrastructure Architecture
Networking
...

Developer

Programming Languages
Frameworks
Design Patterns
Application Architecture
...

Security

Threat Modeling
Cryptography
Security Patterns
Compliance
...

HashiCorp

Unicorn Developers

- “Developers should do it all!”
- Unicorns are in short supply, not a good business decision
- Specialization of Knowledge is real
- Some knowledge can be outsourced, still exists!

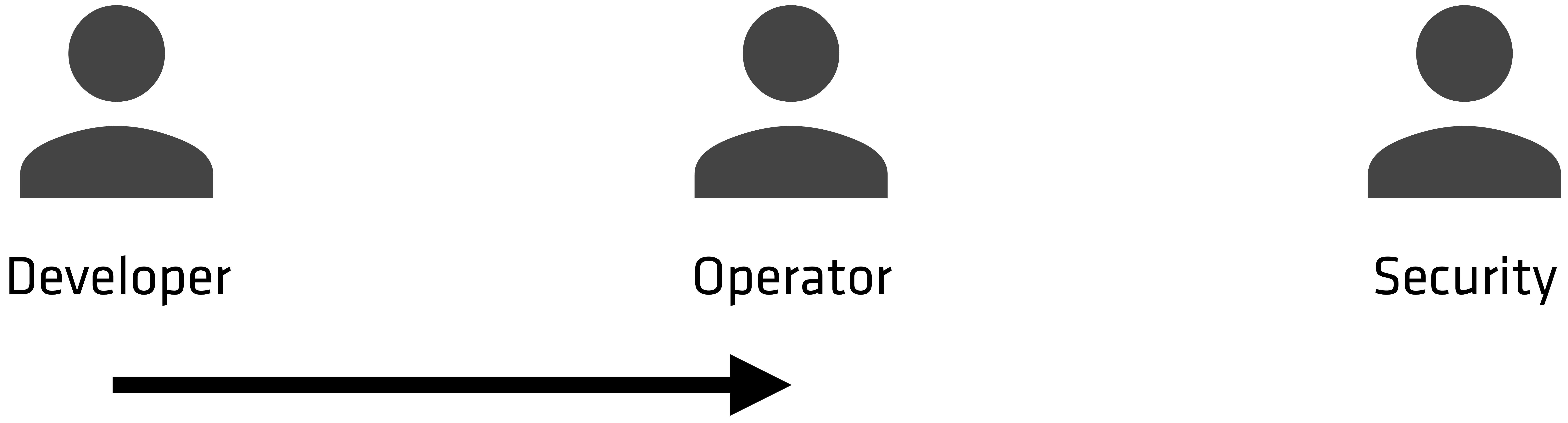
Process

Developer

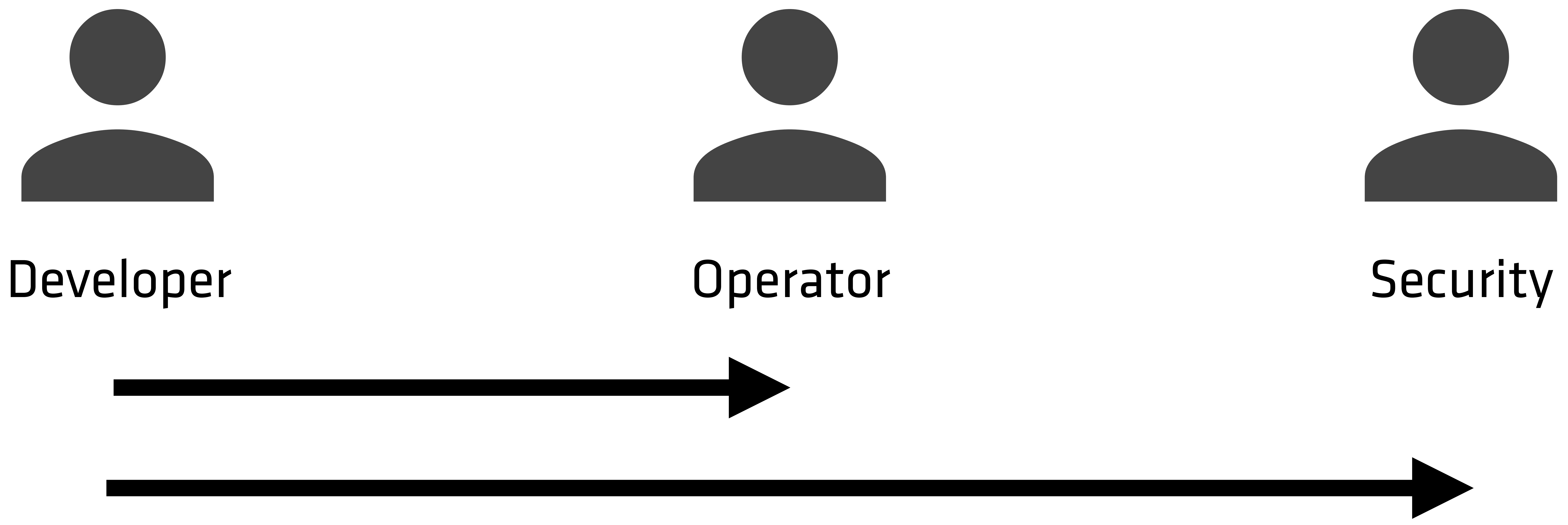
Operator

Security

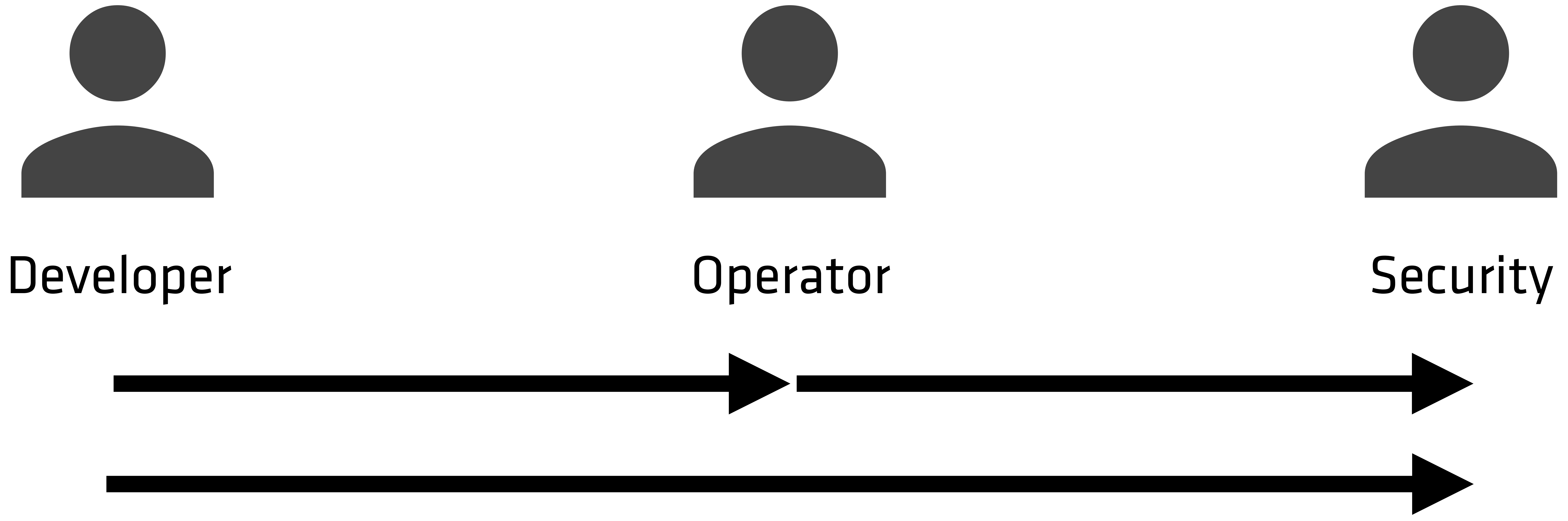
Process



Process



Process



Amdahl’s Law

- The theoretical throughput of a system is limited by *serial* latency
- Organization is a system that is creating an application
- Output is limited by serial coordination
- Empowering individuals to work **independently** improves throughput

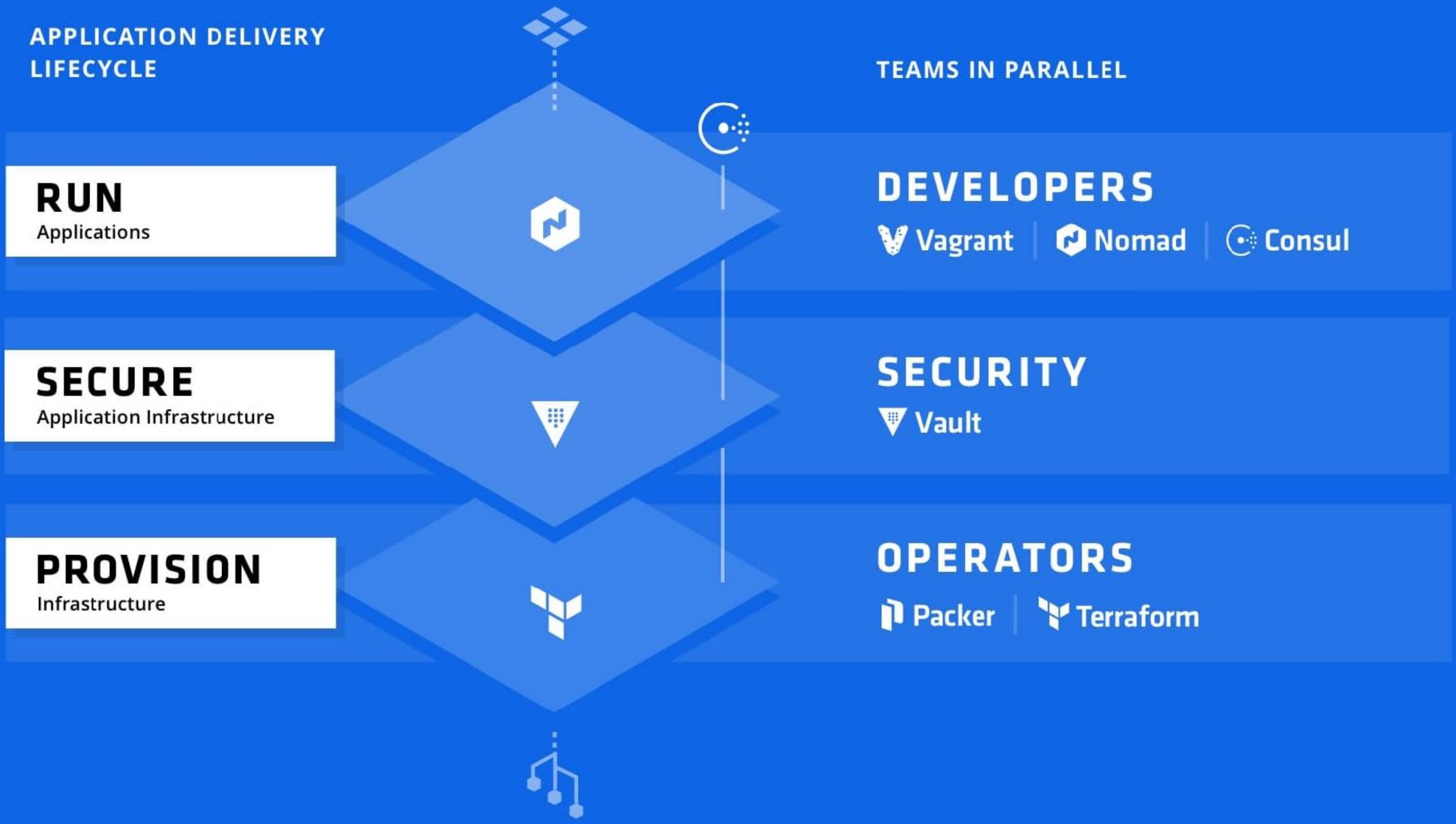
Fundamental Steps

- **Write** the application
- **Test** the application
- **Package** for staging / production
- **Provisioning** infrastructure resources
- **Deploying** an application to the infrastructure
- **Monitoring** applications and infrastructure
- **Securing** applications and infrastructure

DevOps Defined

- Process to fulfill the fundamental steps optimizing for throughput
    - Reduce coordination, empower individuals, focus on delivery time
- Use tools to coordinate between steps instead of people
- Clean separation of responsibilities

HashiCorp


APPLICATION DELIVERY LIFECYCLE
TEAMS IN PARALLEL
RUN
Applications
DEVELOPERS
Vagrant
Nomad
Consul
SECURE
Application Infrastructure
SECURITY
Vault
PROVISION
Infrastructure
OPERATORS
Packer
Terraform

DevOps Process

Developer

- Write and Test
- Consume Secrets
- Deploy
- Monitor Apps

Operator

- Automate Packaging
- Provision Infrastructure
- Provide Deployment Tools
- Monitor Infrastructure

Security

- Model Organization
- Manage Secrets
- Delegate Access
- Compliance

Caveats

- With great power, comes great responsibility!
- Developers become owners of application
- More discipline around testing
- Requires investment in tooling and education

Ask your Doctor

- Every process makes assumptions and optimizes for different metrics
- DevOps optimizes for agility, assumes cost of mistake is low and risk tolerance is high
- Waterfall optimizes for risk management, assumes cost of mistake is high and risk tolerance is low
- Avionics software poorly suited for DevOps
    - Very high cost of mistakes, very low risk tolerance, low iteration speed

DevOps for Startups

HashiCorp

Scaling down DevOps

- That sounded very *Enterprise-y*
- Startups have (many) fewer people
- Impacts process and tools

HashiCorp

Startup Anatomy

- Fewer specializations and less teams
- Unlikely to have dedicated QA, Security, Compliance, etc
- In the early days, may not have any operators

Startup Constraints

- Burn rate, it’s a race against the clock!
- Focus on core product, everything else is a cost center
- Outsource where possible
- High risk tolerance, default is failure

HashiCorp

Doing DevOps

- Usually happens naturally because of lack of specialization!
    - All developers empowered to deploy in the early days
- As you start specializing, stay conscious of the delivery process

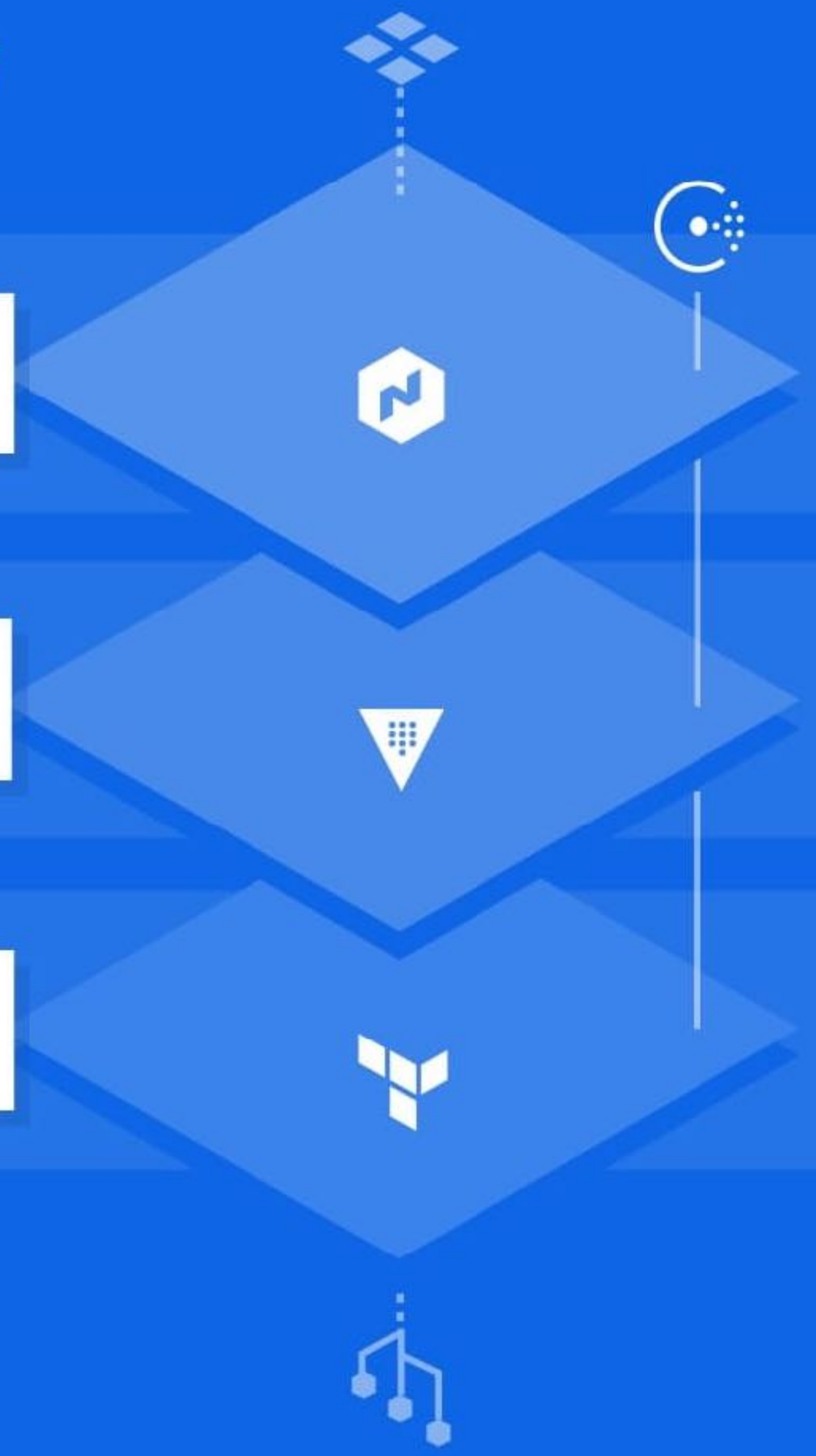
HashiCorp

Pragmatism

- Build for 1x, Design for 10x, Plan for 100x
- You are not Google, nor will you be next year
- Business and product may change, reduce sunk costs
- Stay flexible to change, without building into a cul-de-sac

APPLICATION DELIVERY LIFECYCLE

TEAMS IN PARALLEL



RUN
Applications

DEVELOPERS
Vagrant | Nomad | Consul

SECURE
Application Infrastructure

SECURITY
Vault

PROVISION
Infrastructure

OPERATORS
Packer | Terraform

Provisioning

- Pick a cloud!
  - Large credits available to incentivize usage ($10K-$100K+)
- Leverage the expertise of your people
- Make it easy to spin up multiple environments (prod, stage, dev)
  - Terraform, Infrastructure-as-Code, etc

HashiCorp

Security

- Focus on the low hanging fruit
- Enable 2FA everywhere
- Build a network perimeter (private network + bastion host)
- Avoid secrets / credentials in code (Vault)
- Encrypt sensitive data (Vault)
- Use security monitoring (evident.io)

Runtime

- Focus on developer productivity
- Cost is likely a red herring relative to payroll
- Assume ~0 operators
- Outsource logging (Cloud), metrics (NewRelic, DataDog), exception tracking (Sentry), alerting (PagerDuty)
- Leverage platforms like ECS, Nomad, K8S

Note on Schedulers

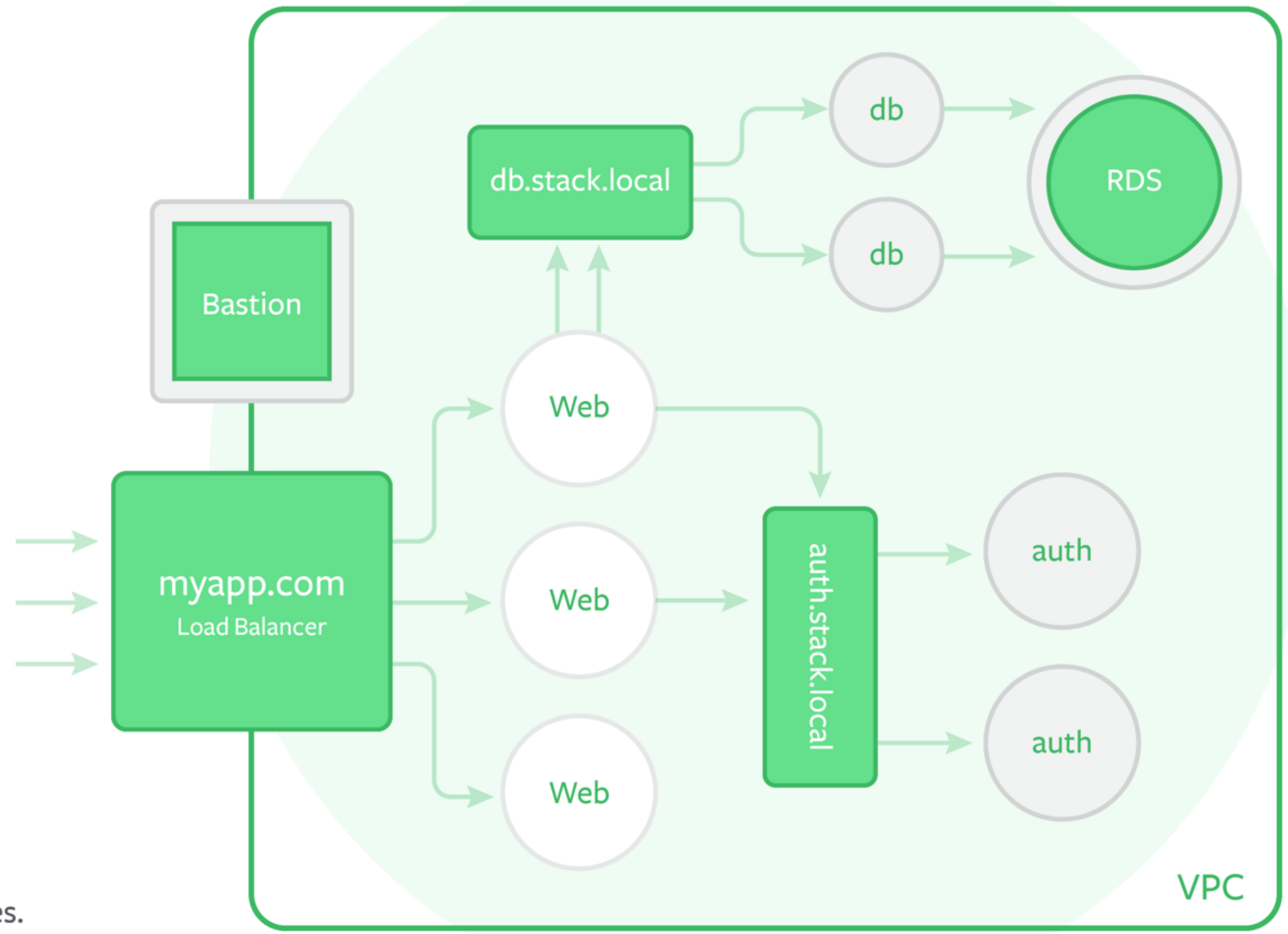
- Schedulers are fantastic, but not silver bullets
- Complex software has complex failure modes
- Keep It Simple Stupid
- Dedicated operator almost a requirement for more advanced systems
- Ask: *Why do we need it?*

Starting with Segment Stack

- Segment is a streaming analytics startup
- Published their full AWS stack configuration
    - https://github.com/segmentio/stack
- Leverages Terraform, AWS, Docker, and ECS

The Open Source

Segment Stack



Production Ready:
Highly available infrastructure across three availability zones.

Modern & Flexible:
Built on Docker, AWS, and Terraform

Easy & Fast:
Create 50 different AWS resources, in under 10 minutes.

Segment

Segment Stack Features

- Secure networking by default
- Basic auto-scaling
- Deployment handled by ECS
- Uses CloudWatch for logging and metrics
- Up and running in 10 minutes

Growing Up

HashiCorp

Growing into scale

- If all goes well, the startup will grow
- More people eventually forces a specialization of knowledge
- Starts to look more like the *Enterprise-y* process
- Allows for more sophistication if done right

HashiCorp

Dedicated Operations

- Owns the Infrastructure / Security / Runtime core
- Provide a platform to developers (write, test, deploy, monitor)
  - Evaluate fancier schedulers (Nomad, K8S, Swarm)
- Richer tooling (deployment, observability, tracing, etc)
- Performance of the infrastructure
- Blue/Green, shadow traffic, enable better testing rigor

HashiCorp

Dedicated Security Team

- Reduce the surface area of access
- Locking down SSH access
- Mutual TLS for services
- Data privacy
- Compliance

Splitting Development Teams

- Dividing the application into services owned by teams
- Reduces coordination between teams, increases operational demand
- Need better deployment and observability tooling
- More disciple around testing required, more moving pieces

Conclusion

DevOps for Startups

- DevOps is a process focused on agility, aligns with constraints of startups!
- Clouds and modern tools give you a huge amount of leverage
- Avoid BIY, almost always a cost center, doesn't add product value
- Go forth and build!

HashiCorp

Thanks!

HashiCorp

Resources

- DevOps Defined: https://www.hashicorp.com/devops.html
- Segment Stack: https://segment.com/blog/the-segment-aws-stack/
    - https://github.com/segmentio/stack
- Terraform: https://www.terraform.io