Putting the **Sec** in Dev**Sec**Ops

*Automating cloud security as an enabler*
Agenda

● Intro to infrastructure as code (IaC)
● Problems facing DevOps & security teams
● Sources of misconfigurations and cloud risk
● Tools to help with IaC security challenges and education
● Addressing cloud security throughout the DevOps lifecycle
● Strategy to implement a DevSecOps strategy
Cloud-native technology (and security) is evolving.
What about

- Readability?
- Debugging?
- Reusability?
- Collaboration?
- Versioning?
- Security?
resource "aws_security_group" "my-sg" {
    name             = "my-sg"
    ingress {
        from_port     = 22
        to_port         = 2
        protocol       = "tcp"
        cidr_blocks  = ["0.0.0.0/0"]
    }
}

resource "aws_instance" "web-server" {
    ami                                  = "ami-0885b1f6bd170450c"
    instance_type                  = "t2.micro"
    vpc_security_group_ids  = [aws_security_group.my-sg.id]
}
Better, but we’re still missing **security**
What are some examples of misconfigurations?

- Unencrypted databases
- Disabled logging
- Insecure protocols
Security isn’t happy

~9 Devs : 1 Sec

1 insecure IaC template → 100s of insecure resources → 1,000s of alerts

Not a real service

... Had real job postings ...

...
Waterfall

Requirements

Design

Implementation

Testing (incl. Security)

Maintenance

YOU WANT ME TO PATCH CODE FROM HOW MANY SPRINTS AGO?

Engineering isn’t happy
Responsibilities are at odds and shifting

DevOps moves faster and is more agile

Traditional security lags and doesn’t scale

Shifting security left is hard and risky
The key to developer-first security

Codified  Automated  Integrated
Tools that can help

...and tools that can help us learn!
github.com/bridgecrewio/checkov
Learn by doing
What does embedded security mean?

Every step of the SDLC
Security in the whole DevOps lifecycle

IDE | Pre-commit | Pull request | CI/CD | Runtime

Available information, representation of actual resource

Complexity, time, risk of finding and fixing errors
Security in the whole DevOps lifecycle

Available information, representation of actual resource

Complexity, time, risk of finding and fixing errors
Security in the whole DevOps lifecycle

IDE | Pre-commit | Pull request | CI/CD | Runtime

Available information, representation of actual resource

Complexity, time, risk of finding and fixing errors
Security in the whole DevOps lifecycle

Available information, representation of actual resource

Complexity, time, risk of finding and fixing errors
Security in the whole DevOps lifecycle

IDE | Pre-commit | Pull request | CI/CD | Runtime

Available information, representation of actual resource

Complexity, time, risk of finding and fixing errors
Implementing a cloud DevSecOps strategy

Crawl, walk, run
## How to roll out your IaC governance program

<table>
<thead>
<tr>
<th><strong>Experiment</strong></th>
<th><strong>Test</strong></th>
<th><strong>Scale</strong></th>
<th><strong>Govern</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>● Scan individual folders and repos</td>
<td>● Schedule scans on repositories</td>
<td>● Orchestrate scans with build jobs</td>
<td>● Audit results with all stakeholders</td>
</tr>
<tr>
<td>● Use open source containerized scanners</td>
<td>● Audit configuration methods and adjust scanners</td>
<td>● Tweak and customize policies</td>
<td>● Implement tagging strategy</td>
</tr>
<tr>
<td>● Explore various output types and their effectiveness</td>
<td>● Track errors and manage SLAs for addressing them</td>
<td>● Evaluate results against known compliance benchmarks</td>
<td>● Get code to cloud visibility</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>● Regulate non-compliance usage using VCS</td>
</tr>
</tbody>
</table>
Thank you!

Meet me in the chat lounge for questions