Detection As Code: Scaling SOC Operations

Aaron Wilkinson

Shoutouts

Alex Teixiera

Threat Detection Engineering SME

https://opstune.com/

David French

Staff Security Engineer @ Google

https://github.com/threat-punter

About Me

- 7 Year Cyber Security Veteran
- Experienced in Incident Response, Digital Forensics, Threat Hunting and Detection Engineering across a range of industries.
- Currently serving as a Lead Incident Response Analyst alongside running the Threat Detection program at Orbia.
- I find Missing People in my spare time #osintforgood
- Lover of automation and memes.

What's the Problem?

- Scalability
- Inconsistent Detections
- Costs and Overhead
- Fragmentation of Tools
- 'Paper-Trails'
- Outdated Detection Capabilities

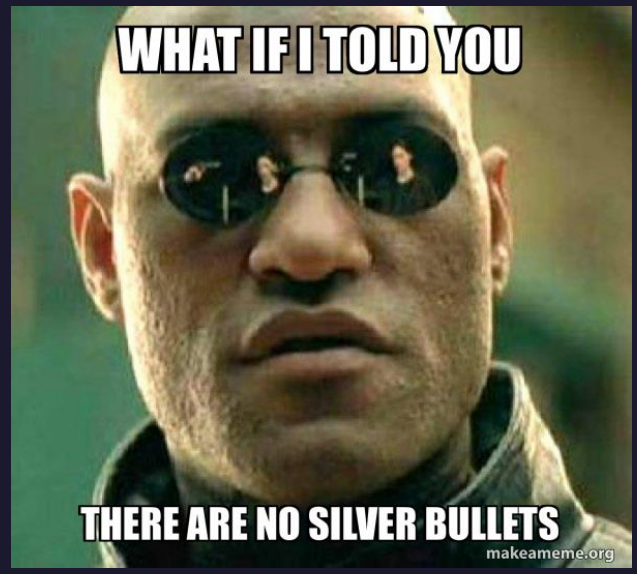
Hello, Detection as Code!

WHAT IS IT?

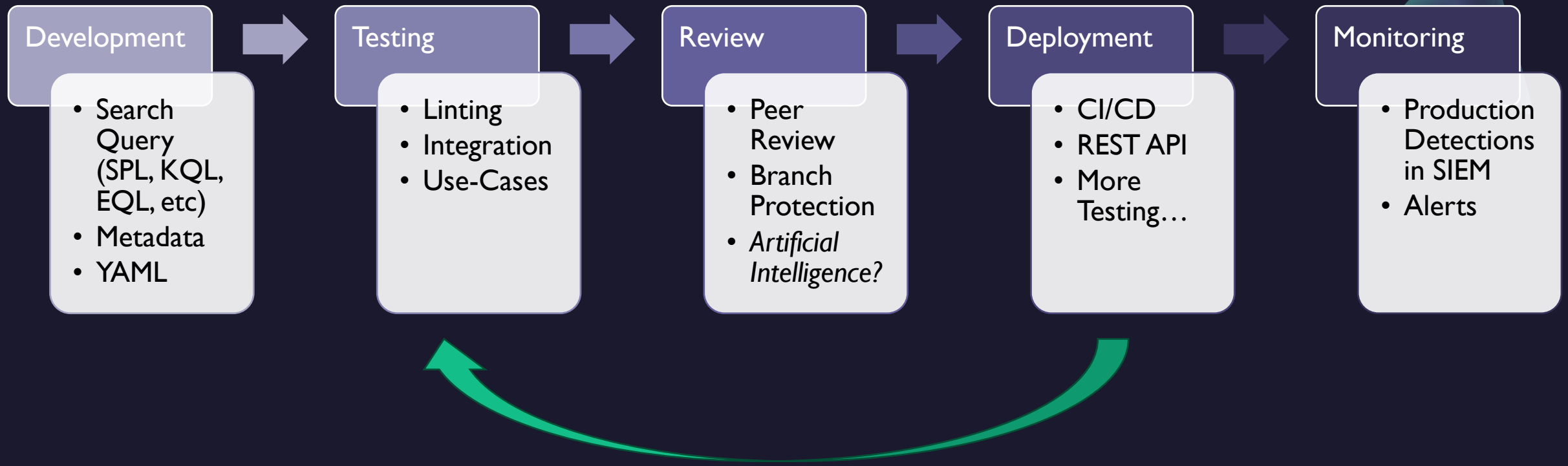
"A structured and strategic approach to integrating Detection Content with the Software Development Lifecycle (SDLC)."

WHAT IS IT NOT?

- A silver bullet solution to your Detection problems.



Example DaC Workflow



Technology Requirements

- Version Control System (VCS)
- CI/CD Pipeline
- Automated Testing Framework(s)
- Low-Code / SOAR Platform (Optional)

Detections in YAML

For Detection-as-Code to be consistent , a well-defined YAML Schema needs to exist!

EXAMPLE

Metadata
Detection Name
Description
Author
Severity
References (Website, MITRE ATT&CK)
Version
Tags

Detection
Detection Query
Event Name
Next Steps
Drilldown Information

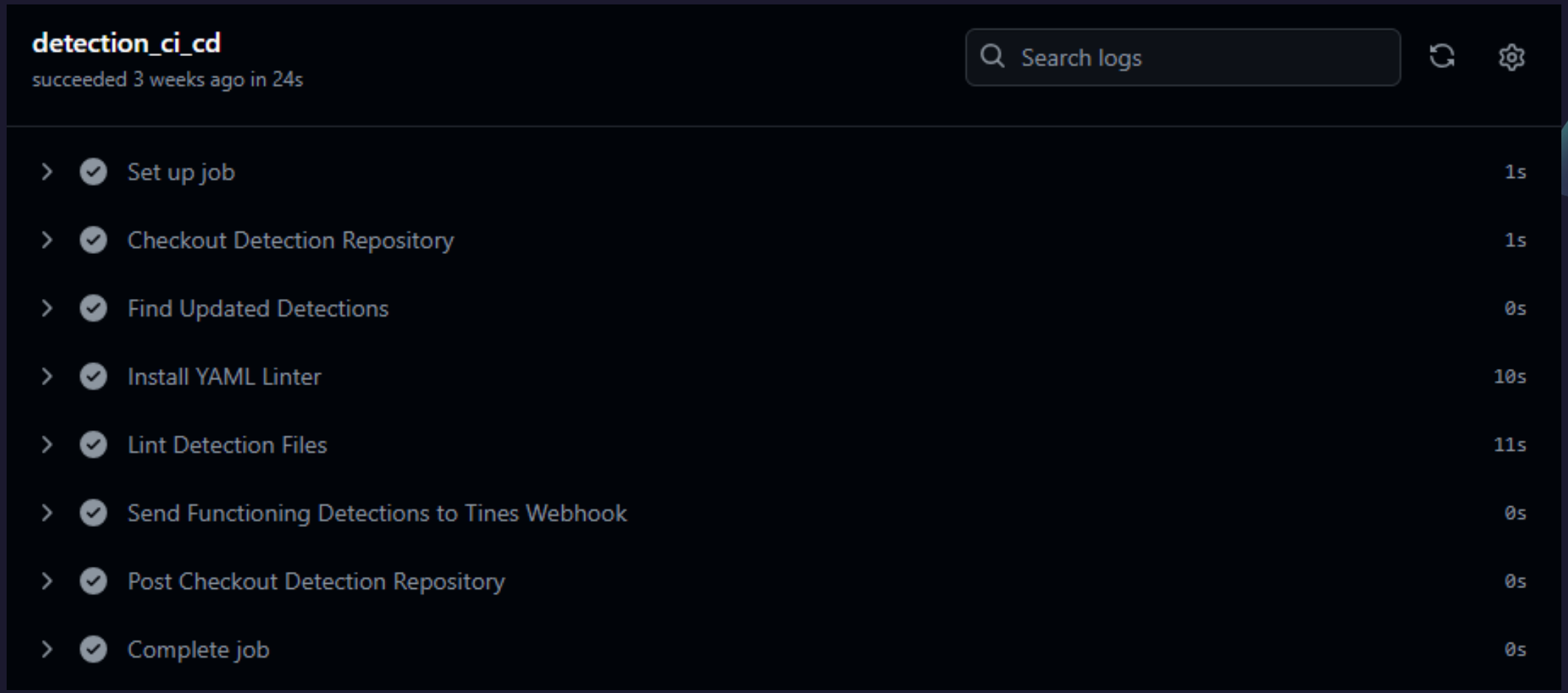
Timestamps
Creation / Last Updated Time
Query Start/End Time
Scheduler Type
Review Cycle
Cron Schedule

CI/CD with GitHub Actions

WHAT IS IT?

An automated workflow platform that can trigger based off various ‘Actions’.

CI/CD with GitHub Actions



CI/CD with GitHub Actions

```yaml
name: Detection CI/CD Pipeline

on:
  push:
    paths:
      - '*.yml'
jobs:
  detection_ci_cd:
    runs-on: ubuntu-latest

    steps:
      - name: Checkout Detection Repository
        uses: actions/checkout@v4

      - name: Find Updated Detections
        id: detections
        uses: actions/github-script@v7.0.1
        with:
          script: |···

      - name: Install YAML Linter
        run: |···

      - name: Lint Detection Files···


      - name: Send Functioning Detections to Tines Webhook
        run: |
          yml_files_base64=${{ steps.lint.outputs.result }}
          yml_files=$(echo "$yml_files_base64" | base64 --decode | jq -r '.[]')
          for file in $yml_files; do
            if [ -f "$file" ]; then
              echo "Sending file: $file"
              curl -X POST -H "Content-Type: multipart/form-data" -F "file=@$file" "https://
              [redacted].tines.com/webhook/3[redacted]/
              [redacted]"
            else
              echo "File not found: $file"
            fi
          done
```

Low Code with Tines

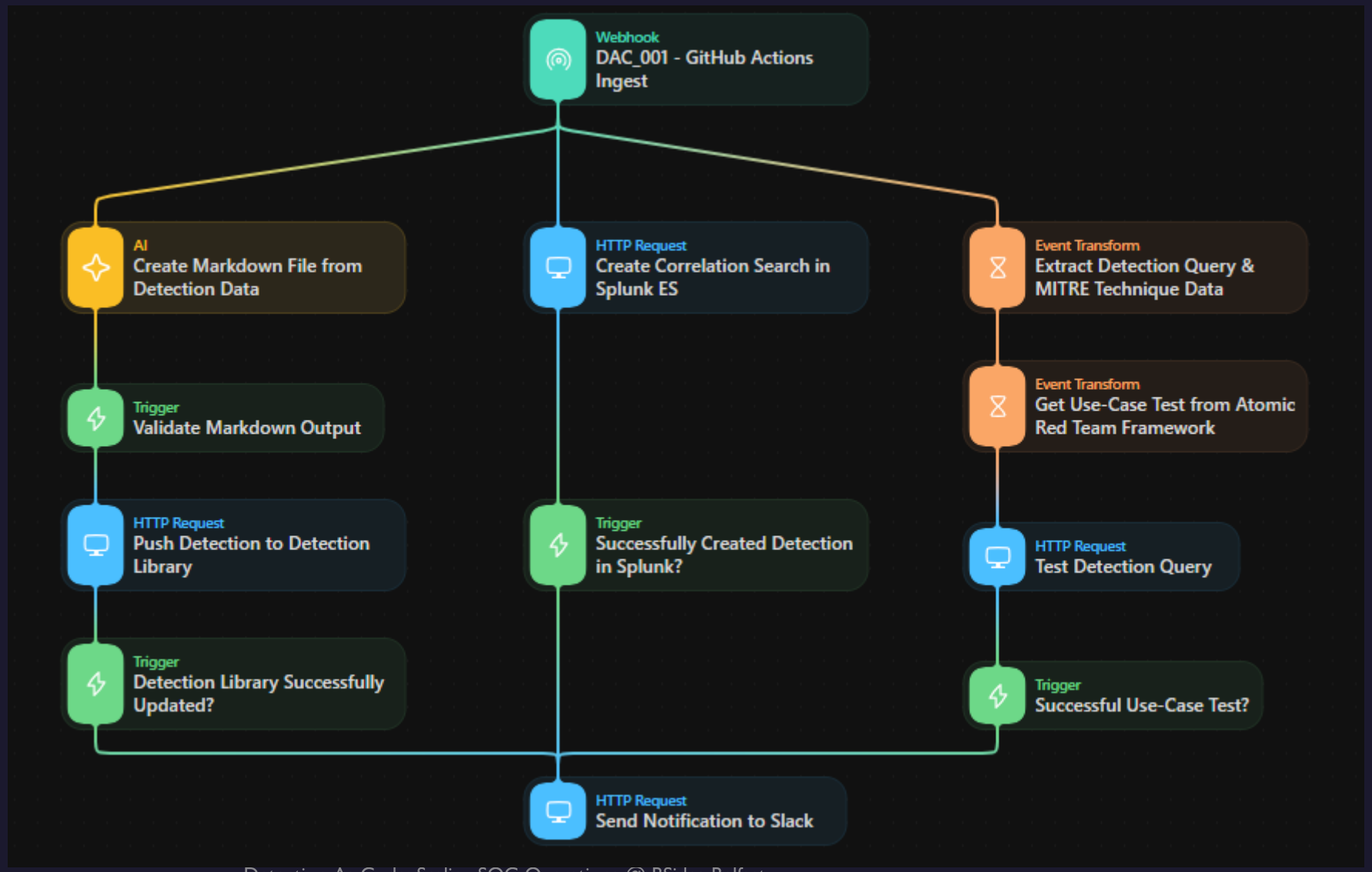
WHAT IS IT?

"It's a Low Code solution that enables teams to create efficient automative workflows."

SO WHAT? HOW CAN THIS BOOST MY DAC CAPABILITIES?

- Automated Health Checking
- Deployment Metrics
- Automatic Remediation Workflows

Low Code with Tines



Additional Resources

- GitHub
  - https://github.com
- GitHub Actions
  - https://github.com/actions
- Tines
  - https://tines.com
- YAML
  - https://yaml.org

- "From Soup to Nuts: Building a Detection as Code Pipeline" – David French
  - https://medium.com/threatpunter/from-soup-to-nuts-building-a-detection-as-code-pipeline-28945015fc38
- All Things Detection Engineering
  - https://detect.fyi
- Detection Engineering Maturity Matrix – Kyle Bailey
  - https://detectionengineering.io

Q&A