Measuring Detection Engineering Teams

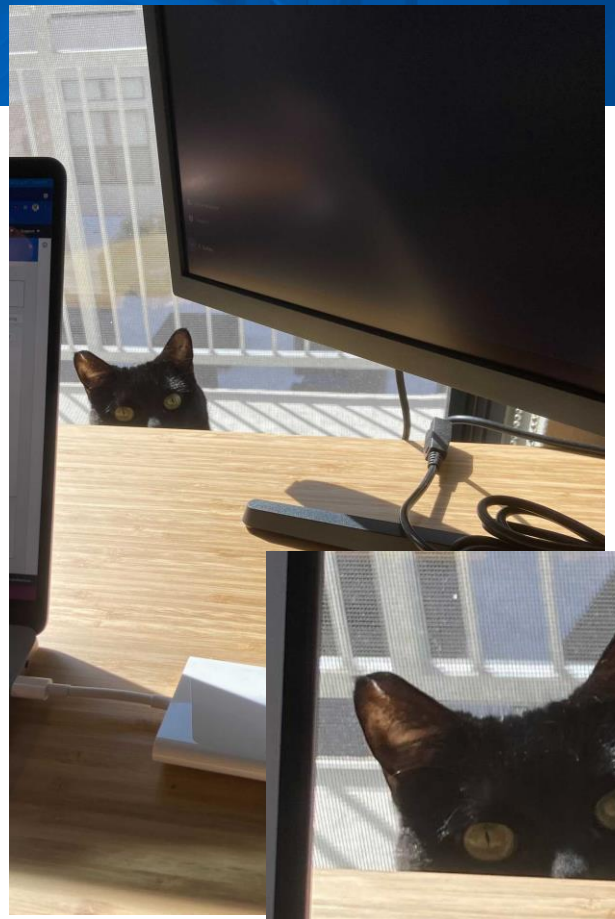
Kyle Bailey | Security Engineer | Panther Labs
@KyleBailey22

Agenda

- Detection Engineering Core Principles
- Motivation & Methodology
- Maturity Matrix
- Metrics
- Resources

About Me

- 5y managing cyber operations for USAF & CYBERCOM
- 3y as an Analyst/Engineer working in Incident Response
- 2y building & managing the detection engineering program @ Box
- Currently @ Panther labs breaking things



Detection Core Concepts

Overarching areas that make up the content of the Matrix:

Process Change

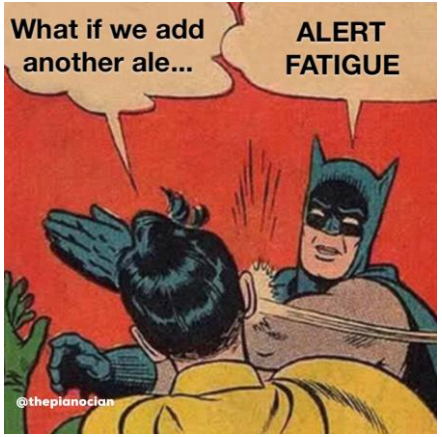
+ Technology Change

- The Incident Response Experience
  - Alerts are actionable, high fidelity and contain sufficient context
- Detection Logic & Infrastructure
  - Log visibility & logic to identify malicious behavior
- Detection-as-code
  - Treating all detection logic as code, following software engineering principles

Incident Response Experience

Do our alerts set the incident response team up for success?

- Alert Documentation & Context
  - Does IR (or another detection engineer) understand what the alert does and why it was built? Are there blindspots or assumptions that need to be known?
  - Start here: Palantir Alerting & Detection Strategy
- Alert Fidelity, Monitoring & Maintenance
  - Does IR have a dedicated resource to help with low fidelity alerts? Or the permission to self serve?



Incident Response Experience

Do our alerts set the incident response team up for success?

- Detection & Response Relationship
  - Constant collaboration to improve low value alerts, and processes for ensuring new detection is as high quality as possible.
  - When building new alerts “begin with the IR team in mind”
  - Is the alert “worth” someones time? Can it be automated? Are there tangible response actions?
- Automated resolution by the user
  - Does a secure method exist for validating actions without IR interaction? i.e. Slackbot
  - https://www.tines.com/blog/chatbots-for-security-and-it-teams-part-3-creating-a-slack-chatbot

Detection Logic & Infra

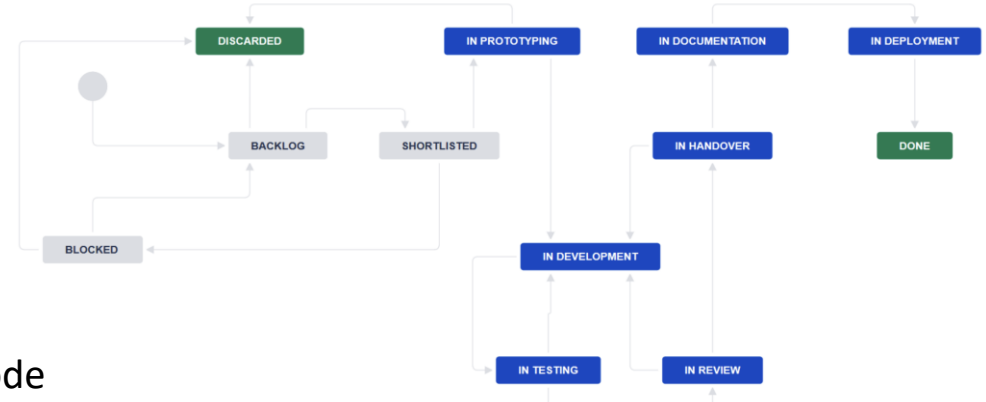
How good is our threat visibility & ability to detect those threats?

- Log visibility & timeliness
  - If malicious activity occurs can we see it in the log data we collect? (OS logs, EDR, network, cloud management planes, cloud services, application layer, etc.)
  - Does data reach the detection logic in near real time?
  - Are we removing unnecessary log data from the SIEM?
- Mitre Att&ck
  - Integrating att&ck TTP tracking into detection processes and documentation
- Purple teaming
  - Validating the detection your team built is working as intended (and continues to work)
  - Start here: 1-hour purple teaming, Ben Goerz https://www.youtube.com/watch?v=fNQ7EQjd3Zg
- Threat Intelligence
  - Focusing on att&ck TTP's that are probable, not possible. Varies based on your threat model.

Detection-as-Code

Are we using engineering processes and principles?

- Agile Processes
  - The team follows scrum, kanban, etc.
  - Task visualization, prioritization and tracking, work sizing.
- Code Reuse
  - Breaking query logic into functional pieces and creating “functions” for code that is the same across multiple detections.



Ref: https://ateixei.medium.com/jira-workflow-for-detection-engineering-teams-a7433f4c2a9f

Detection-as-Code

Are we using engineering processes and principles?

- Version Control
  - The source of truth for your detection logic: Diffs, enforced peer review, CI/CD integration, rollbacks.
  - Check with your SIEM provider, they may have prebuilt code to help you get started.
- CI/CD
  - Automatically deploy approved changes, enforce testing.
  - Someone in your org is probably already using Jenkins, CircleCI, etc. Enlist their help.
- Static & Dynamic Testing
  - Ensuring errors don't exist and the logic will catch malicious activity.
  - Linting, running static tests, dynamic tests, looking for an excessive number of results or errors, even expected true positives.
  - Start here: https://atomicredteam.io/ manual testing is a great place to start.

Motivation and Methodology

- I wanted to capture the core concepts in a simple & consumable format.
- I couldn’t find a maturity matrix specific to detection engineering
- I really like redteams.fyi (@ok_bye_now)
- A successful detection engineering team is highly dependent on other teams (Threat Intel, IR, RedTeam, Engineering, etc.)

Matrix

	Defined	Managed	Optimized
People	The team owning (creating & maintaining) the detection logic & infra, leadership awareness and support for properly resourcing the detection function.		
Process	The processes used to prioritize new detection, track detection work, monitor existing detection, track metrics and work with the IR team.		
Technology	SIEM and surrounding the components that give your team log visibility. Detection-as-code tools to enable the detection team to follow those principles.		
Detection	The robustness and fidelity of detection logic and the catalyst for building new use cases (threat intel, purple team). Mitre Att&ck integration.		

https://github.com/k-bailey/detection-engineering-maturity-matrix

Some Metrics

- Detection Performance
  - % False positives, % “Expected behavior”
  - % of alerts resolved by an end user
  - Mean time to detect
  - % of “probable” att&ck TTPs covered by detection logic
  - % of detection catalogue tested by a purple team exercise
- Detection-as-code
  - % of detection logic in version control
  - % of logic with test coverage
  - % of deployments with errors

Resources

- Palantir Alerting & Detection Strategy
  - https://github.com/palantir/alerting-detection-strategy-framework
- Jira Workflows for Detection Engineering Teams
  - https://ateixei.medium.com/jira-workflow-for-detection-engineering-teams-a7433f4c2a9f
- 5 Benefits of Detection-as-Code
  - https://blog.runpanther.io/detections-as-code/
- Google Cloud Security Podcast
  - Episodes 17 & 24 https://cloud.withgoogle.com/cloudsecurity/podcast/
- More on detection as code - testing:
  - https://kyle-bailey.medium.com/detection-as-code-testing-c03b0eea7fb8

SANS BLUE TEAM SUMMIT & TRAINING

Live Online

Summit: September 9–10, 2021