Won't somebody please think of the maintenance?

Martin McCloskey

whoami

- Detection Engineer at Datadog
- Previously worked in threat research & Incident Response at a MSSP & internally.
- I was in Norway last month.

DATADOG

Outline

- Detection Engineering
- Why is maintenance neglected
- A simple approach to maintaining

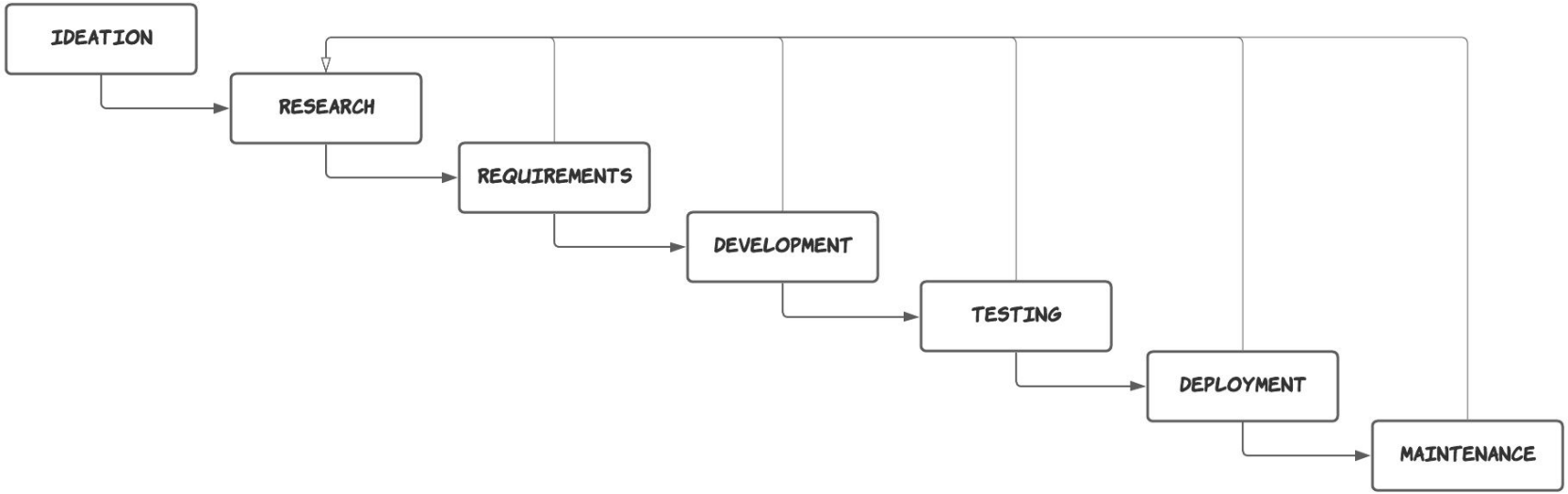
Detection Engineering

Detection Engineering

- At its core, detection engineering functions within security operations and deals with the design, development, testing, and maintenance of threat detection logic.

source: https://panther.com/cyber-explained/detection-engineering-benefits/

The Detection Engineering Lifecycle



Possible Output

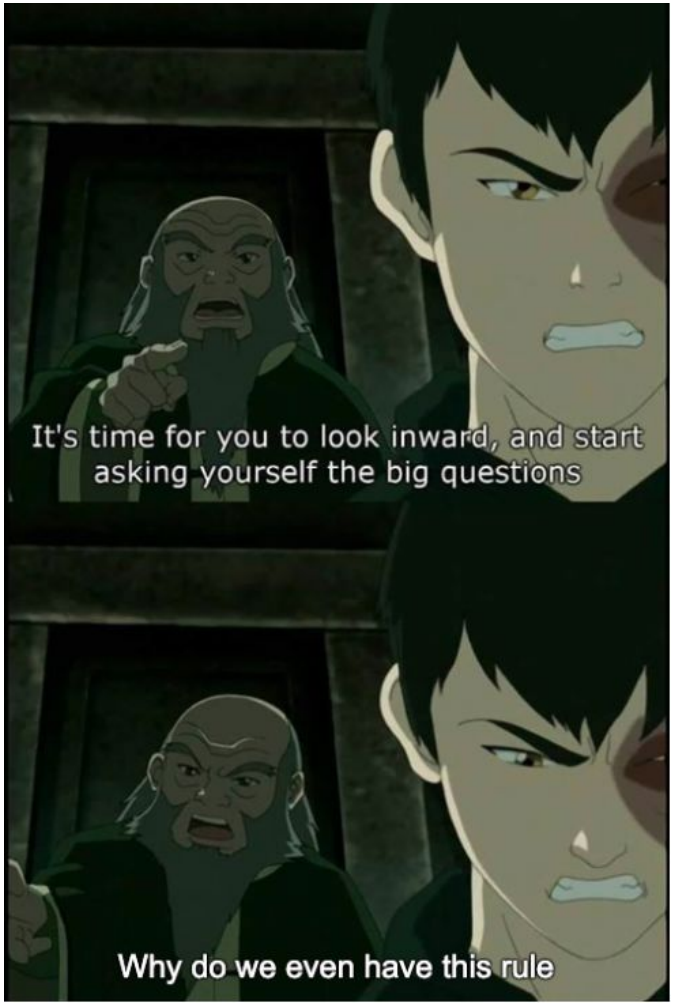
```yaml
title: Azure AD Only Single Factor Authentication Required
id: 28eea407-28d7-4e42-b0be-575d5ba60b2c
status: experimental
description: Detect when users are authenticating without MFA being required.
references:
    - https://docs.microsoft.com/en-gb/azure/active-directory/fundamentals/security-operations-user-accounts
author: MikeDuddington, '@dudders1'
date: 2022/07/27
tags:
    - attack.t1078
logsource:
    product: azure
    service: signinlogs
detection:
    selection:
        Status: 'Success'
        AuthenticationRequirement: 'singleFactorAuthentication'
    condition: selection
falsepositives:
    - If this was approved by System Administrator.
level: low
```

Why is maintenance neglected?

Why is maintenance neglected

- It’s not as interesting - it’s in the name.
- It’s not seen as a valuable use of time (fighting fires, deploy and onto the next one, whack-a-mole)
- You don’t have the data/metrics to identify detections for maintenance.

A simple approach to maintaining


It's time for you to look inward, and start asking yourself the big questions
Why do we even have this rule

DATADOG

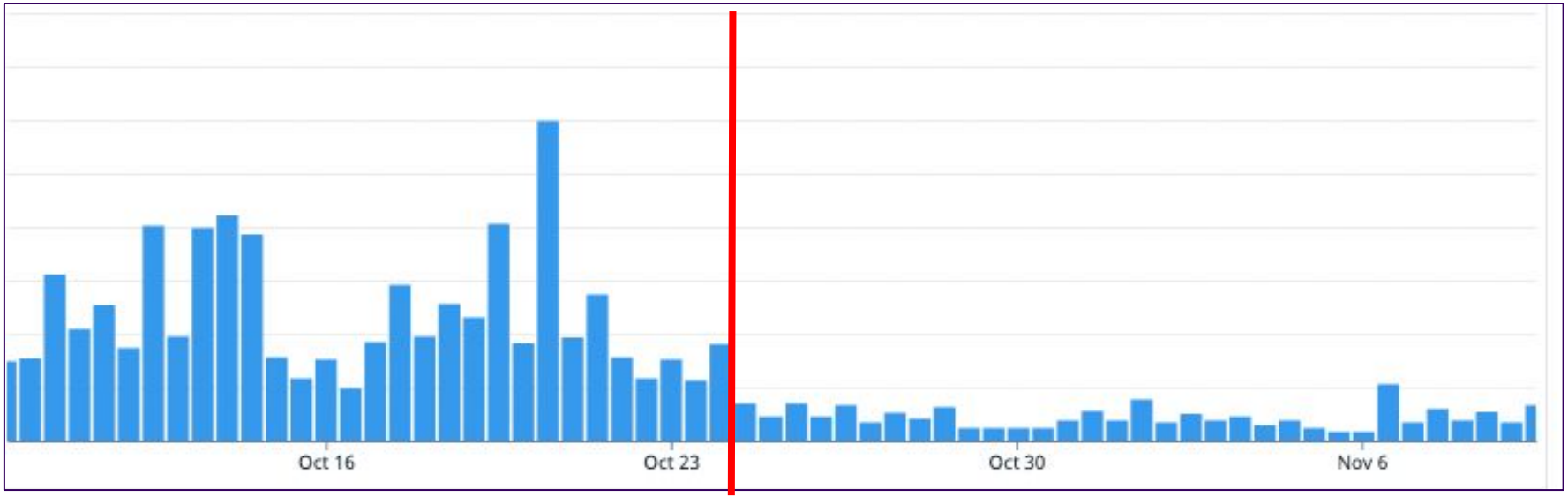
Reducing the noise

- Identify the top 10 detections with the highest volume over a period of at least 30 days.
  - Prioritise further by severity

- Number of causes behind a high rate of false positives.

- Visualisations will be very helpful to communicate the impact of the work here.

Time Series Analysis



Validating critical/high detections

- On the flipside of rules that fire all the time are ones that either hardly ever fire or never fire over a 1-3 month period.
- Tackle the highest severity detections first.
- End to end testing with adversary emulation frameworks
  - Atomic Red Team, Stratus Red Team, MITRE Caldera

https://github.com/redcanaryco/atomic-red-team

https://github.com/DataDog/stratus-red-team

https://github.com/mitre/caldera


DATADOG

Detection As Code

- A more systematic, flexible and comprehensive approach to threat detection that is somewhat inspired by software development.
- Check out “BSidesSF 2022 - Detection-as-code: Why it works and where to start”

Anton Chuvakin

Sep 21, 2020 · 5 min read · Listen

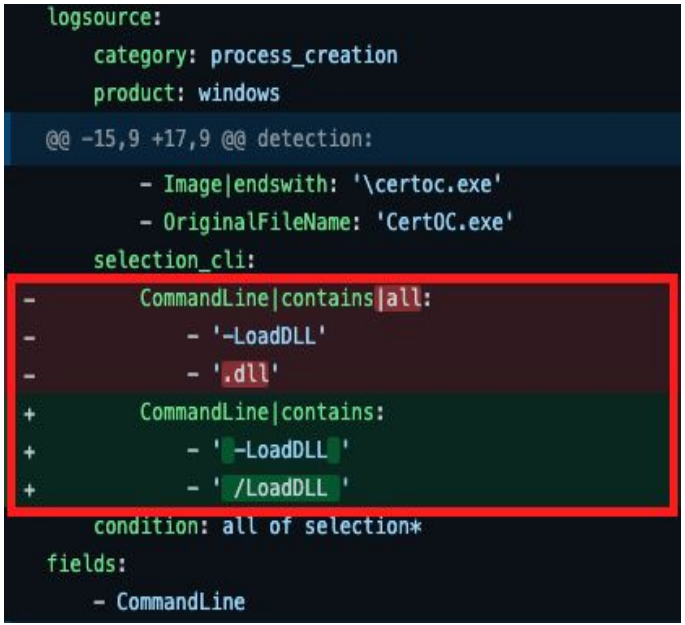
Can We Have “Detection as Code”?

One more idea that has been bugging me for years is an idea of “**detection as code.**” Why is it bugging me and why should anybody else care?

DATADOG

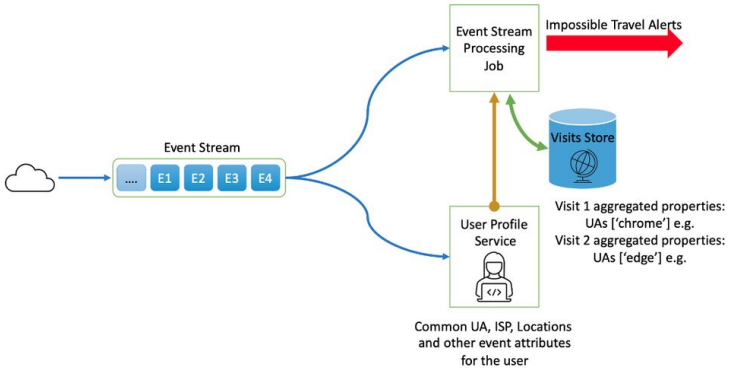
Version Control

- Easily track changes to rules.
- Helpful to identify why a rule is noisy.



Context

- Adding context to detections can aid in providing a higher level of fidelity in the alert, reducing noise.
- Not just threat intelligence
- Identity & asset tagging can be useful.



source:https://techcommunity.microsoft.com/t5/microsoft-365-defender-blog/detecting-and-remediating-impossible-travel/ba-p/3366017

Adjusting Severity

This PR fixes some false positives found in testing

- Reduced the level of the following three rules due to untunable FP. By that, I mean that when the rules trigger the event doesn't contain enough information on its own to let the analysts know whether the action was malicious or not. It needs to be correlated with other events such as process creation for example in order to obtain useful information. Since I was able to trigger these rules in a benign way. I think it's better to reduce the level to medium. At least until correlation is introduced to SIGMA. (Example of FP was discussed in keybase)
  - b439f47d-ef52-4b29-9a2f-57d8a96cb6b8
  - 06ce37c2-61ab-4f05-9ff5-b1a96d18ae32
  - ec1d5e28-8f3b-4188-a6f8-6e8df81dc28e
- Added FP to AV DLL Sideloading Rule based on DELL SAR processes
- Other FP found during testing which are staright forward to understand from the change itself.

Detection Logic

```yaml
title: Azure AD Only Single Factor Authentication Required
id: 28eea407-28d7-4e42-b0be-575d5ba60b2c
status: experimental
description: Detect when users are authenticating without MFA being required.
references:
    - https://docs.microsoft.com/en-gb/azure/active-directory/fundamentals/security-operations-user-accounts
author: MikeDuddington, '@dudders1'
date: 2022/07/27
tags:
    - attack.t1078
logsource:
    product: azure
    service: signinlogs
detection:
    selection:
        Status: 'Success'
        AuthenticationRequirement: 'singleFactorAuthentication'
    condition: selection
falsepositives:
    - If this was approved by System Administrator.
level: low
```

Detection Logic

- Duo/OKTA have bypass policies for - Windows-AzureAD-Authentication-Provider/1.0
- MFA requirement satisfied/skipped by….
- Trusted devices
- Trusted network locations (check your conditional access policies)

Key takeaways

- A rigorous maintenance process allows us to better manage the risk of false negatives while improving the efficiency of our detections and reducing toil on false positives.

- Identify top 10 highest volume and lowest volume detections in your organisation, and apply a maintenance strategy.

- Be open to having your work and thinking challenged, if you are on the receiving end.

Thank you!