CLOUD DETECTION ENGINERING IN AZURE

Day Johnson

$~: WHOAMI

- Detection Engineer at Datadog (Cloud SIEM Product)
- Information Technology at WGU
- Cybersecurity Content Creator on Youtube (13k+ Subscribers)
- Founder & Community Manager at Cyberwox Academy
- AWS Community Builder
- Cybersecurity Tutor at Collin College

ABISOLA DAYSPRING JOHNSON (DAY)

Detection Engineer, Datadog

@daycyberwox

Day Johnson

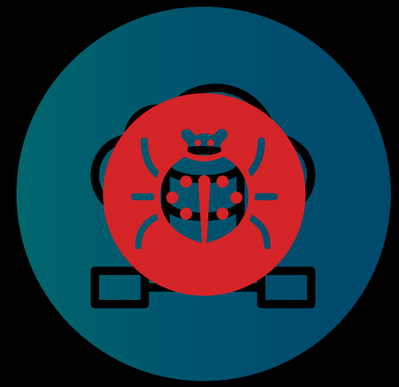
AGENDA

- **Why we need Cloud Detection Engineering**
  - The proactive approach
- **Traditional Detection Engineering vs. Cloud Detection Engineering**
- **What is Cloud Detection Engineering**
- **How to approach Cloud Detection Engineering**
  - Understanding the Adversary
  - Observability (Telemetry, Logging & Cloud Altitudes)
- **Implementing Cloud Detection Engineering**
  - Detection Engineering Lifecycle
  - Detection-as-Code with Sigma
- **Operationalizing Cloud Detection Engineering**
  - Detection Resilience
  - Democratizing your Detection Function
- **Putting it all together**

WHY DO WE NEED CLOUD DETECTION ENGINEERING?

- Adversaries have typically been ahead of defenders
- Defenders tend to be more reactive than proactive
- Traditional IDS/IPS systems are no longer enough
- Defenders need to know when preventative controls fail

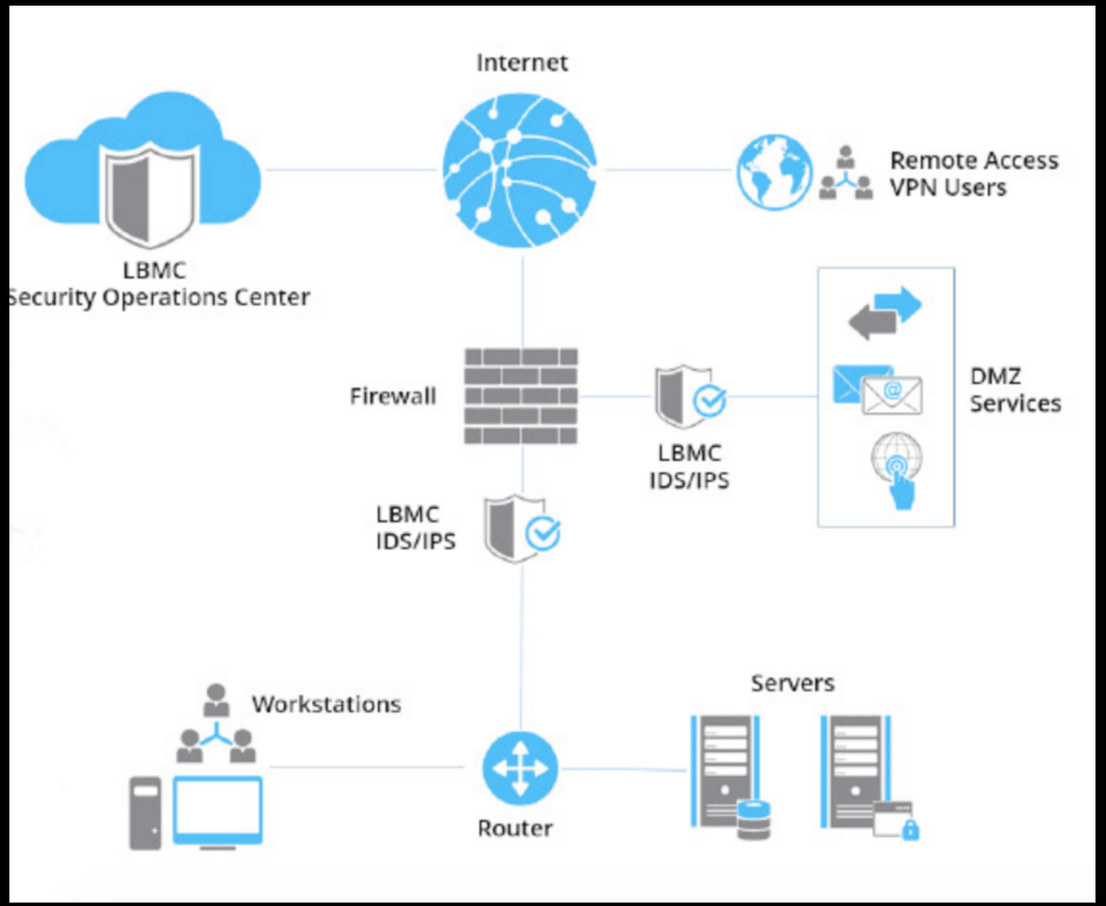
ADVERSARY



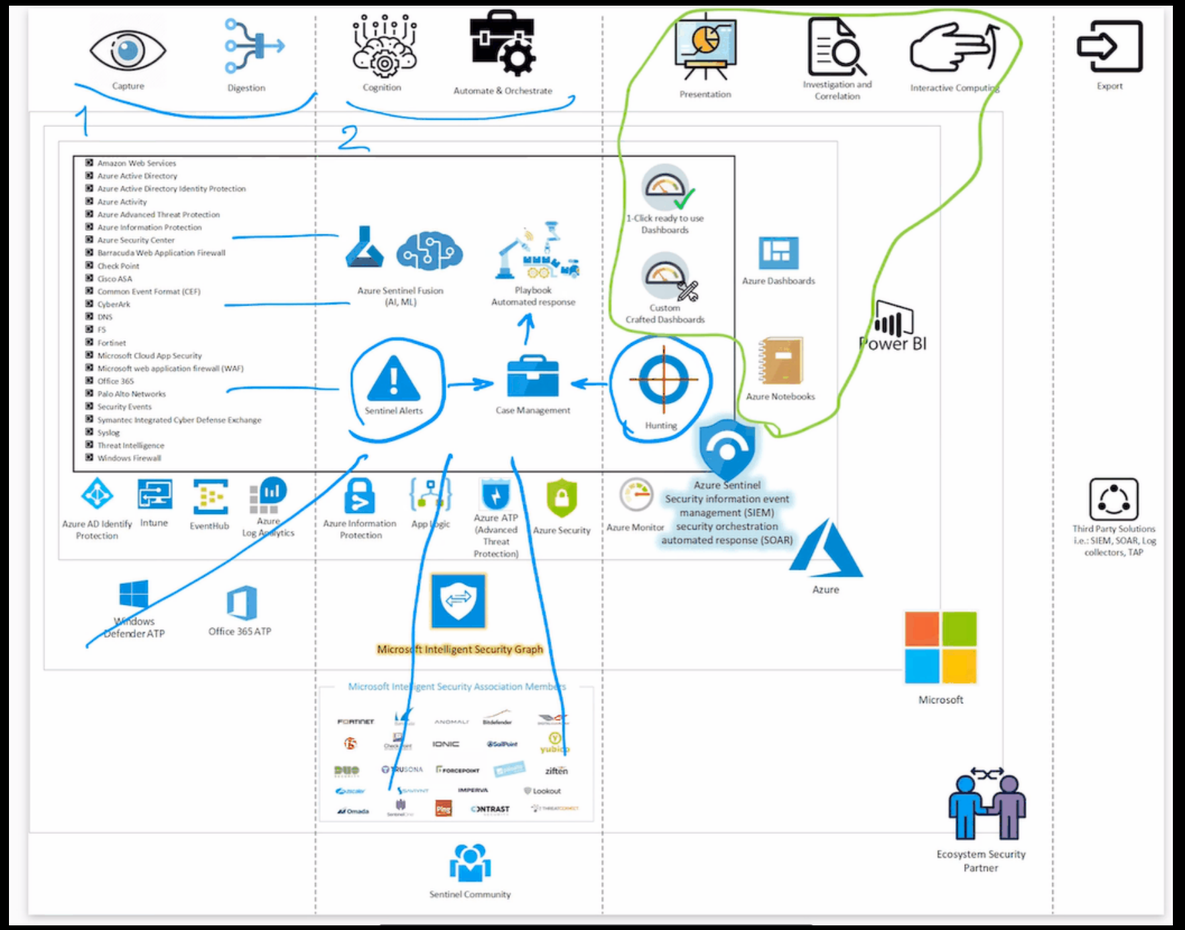
VICTIM CLOUD

TRADITIONAL DETECTION ENGINEERING
VS
CLOUD DETECTION ENGINEERING

TRADITIONAL



CLOUD



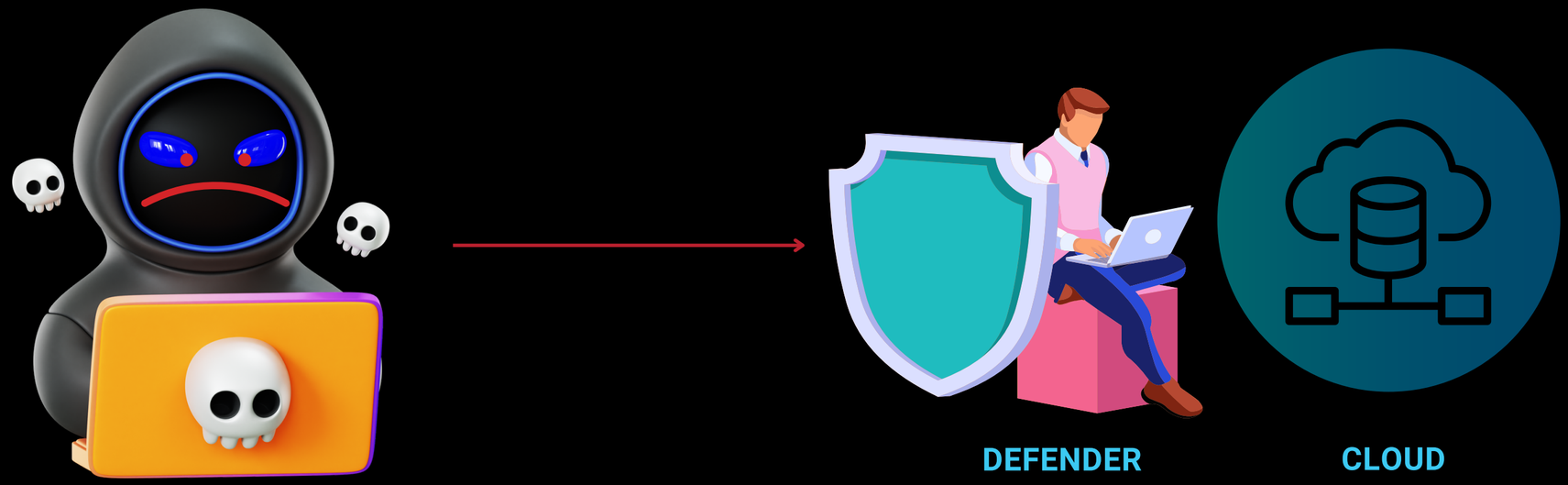
Source:https://www.lbmc.com/blog/ids-vs-ips/

Source:https://www.ais.com/tag/threat-detection/

THE PROACTIVE APPROACH

What if?

- Defenders are steps ahead of adversaries?
- Defenders can build guardrails around their assets?
- Defenders can build custom & advanced detection mechanims?



WHAT IS CLOUD DETECTION ENGINEERING?

Cloud Threat Detection Engineering is the process of designing, implementing, and maintaining systems and controls that identify anomalous activities in cloud environments.

HOW DO WE APPROACH CLOUD DETECTION ENGINEERING?

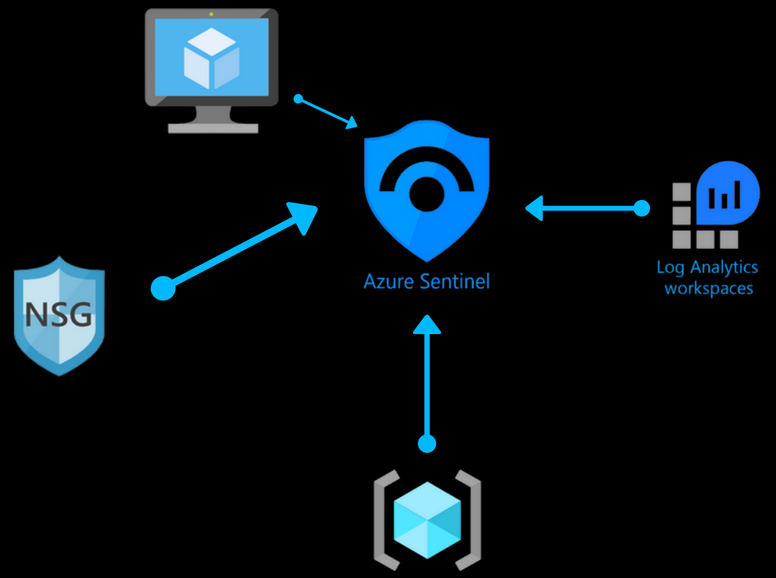
THE ADVERSARY

- **WHO IS THE ADVERSARY?**
  - The Persona - Insider Threats, APTs, etc.
- **WHAT IS THE ADVERSARY'S GOAL?**
  - The Tactic - Persistence, Initial Access, etc
- **HOW WOULD THE ADVERSARY TRY TO ACHIEVE THIS GOAL?**
  - The Technique - Azure automation accounts for Persistence, etc.
- **WHAT WOULD THE ADVERSARY USE TO ACHIEVE THIS GOAL?**
  - The Procedure - Automated tools, manual API calls, etc.

OBSERVABILITY - TELEMETRY

What to consider

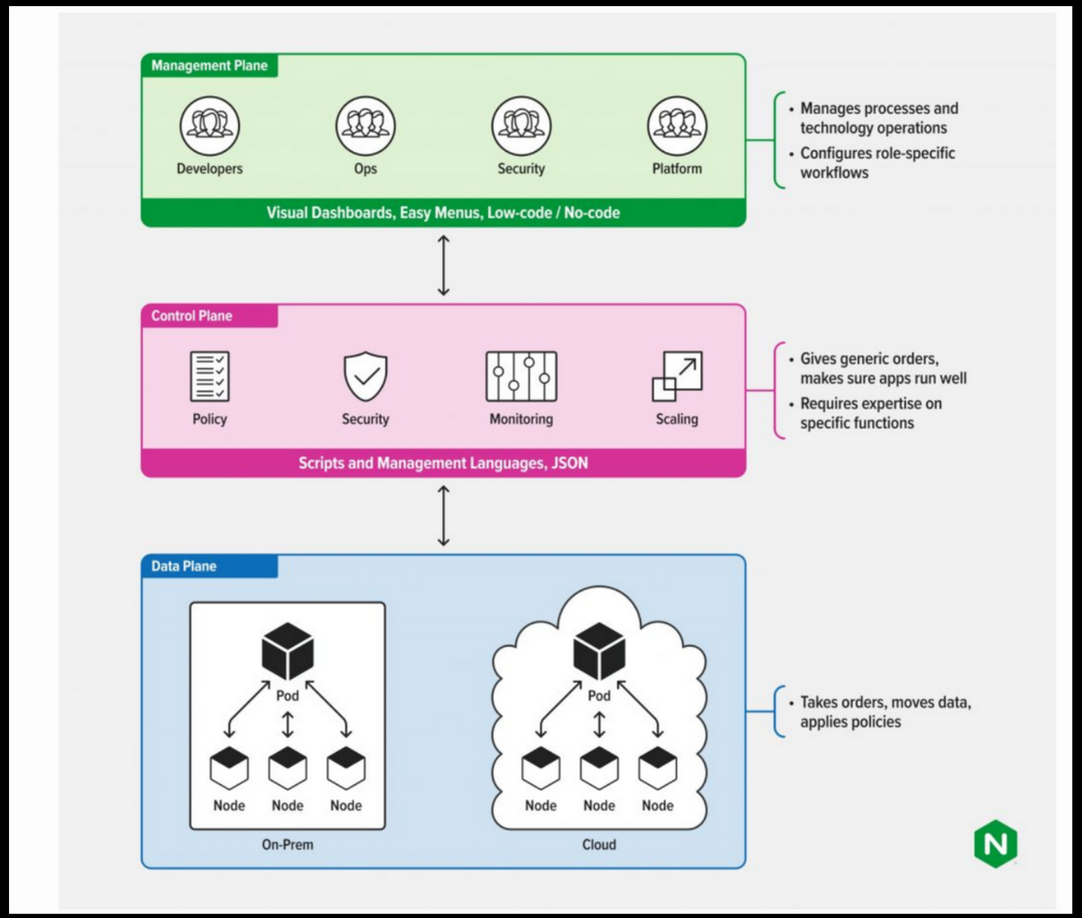
- Visibility for cloud management plane, control plane & data plane
- Baselining your environment for anomaly detection
- Cost – visibility comes at a price (Log storage, ingestion and processing)



CLOUD ALTITUDES

EASIER

EASE OF LOGGING



EASE OF VISIBILITY

HARDER

Source:https://thenewstack.io/data-control-management-three-planes-different-altitudes/

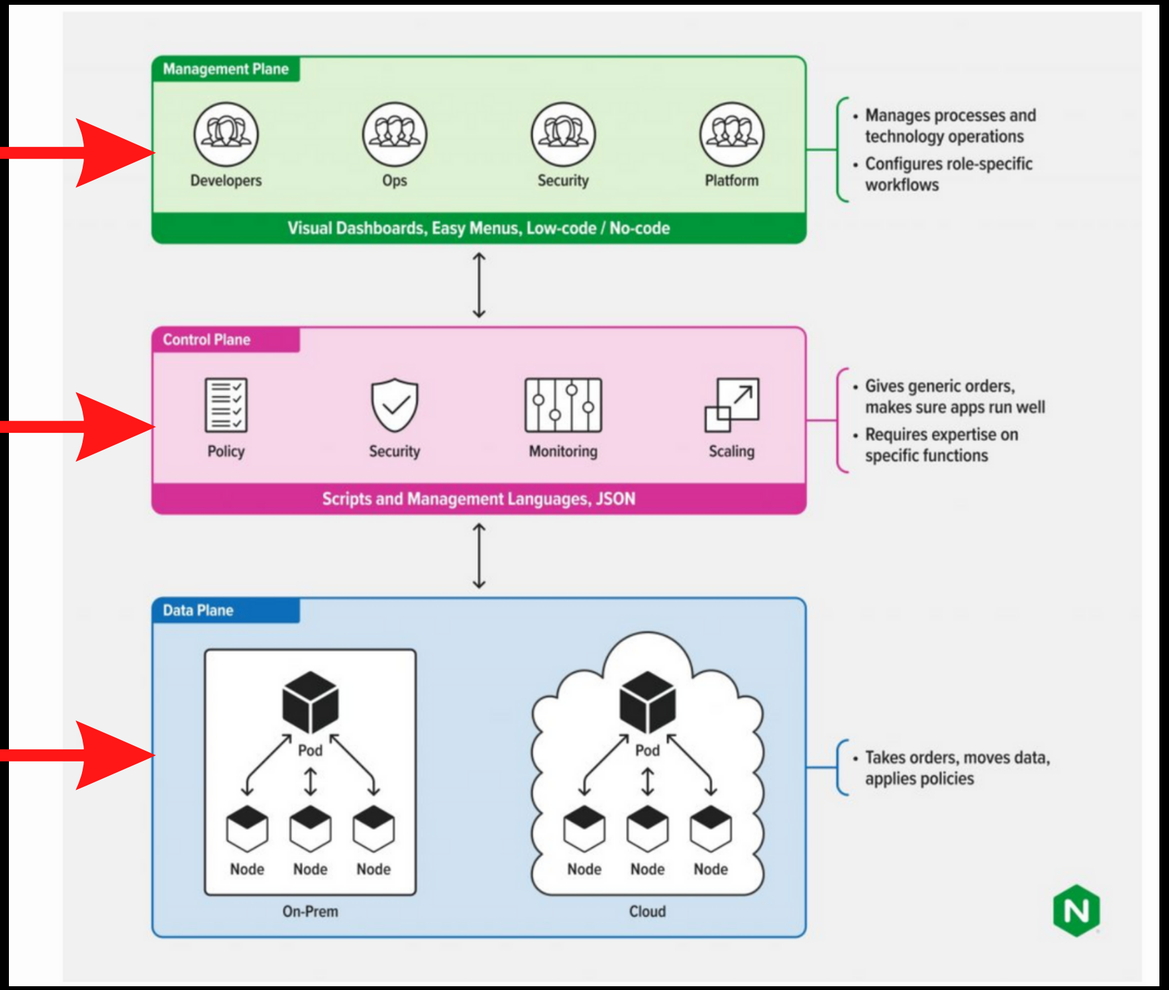
OBSERVABILITY - LOGGING FOR VARIOUS ALTITUDES

Auth & Identity - Behaviors from IAM Entities

Infrastructure - All Azure Audit Logs & Analytics

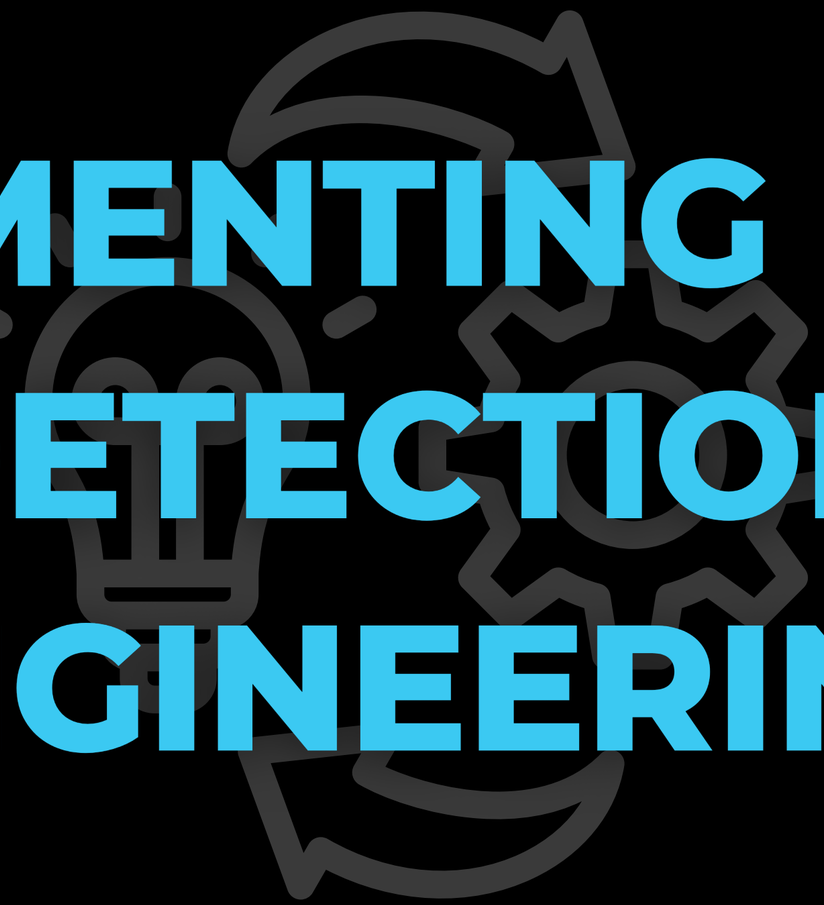
Hosts - VMs, Nodes, Containers, e.t.c

Others - Databases, Applications, CI/CD, e.t.c



Source:https://thenewstack.io/data-control-management-three-planes-different-altitudes/

IMPLEMENTING CLOUD DETECTION ENGINEERING



DETECTION ENGINEERING LIFECYCLE

- Spin off of the **SDLC** but for Detection Engineering
- Well-defined & iterative process to build and maintain detections
- Also known as **Detection Development Lifecycle**

IDEATION

- **The initial idea for the detection**
  - Threat models - partner with other security or product teams
  - Threat Intel data
  - Previous Incidents - IR Docs
  - Red Team engagement or Penetration Testing finding
  - Compliance finding, e.t.c

HYPOTHETICAL SITUATION

A recent pentest revealed that a compromised user could register a "rogue" device to your Azure AD without MFA. This allowed the adversary to easily maintain persistence in your environment.

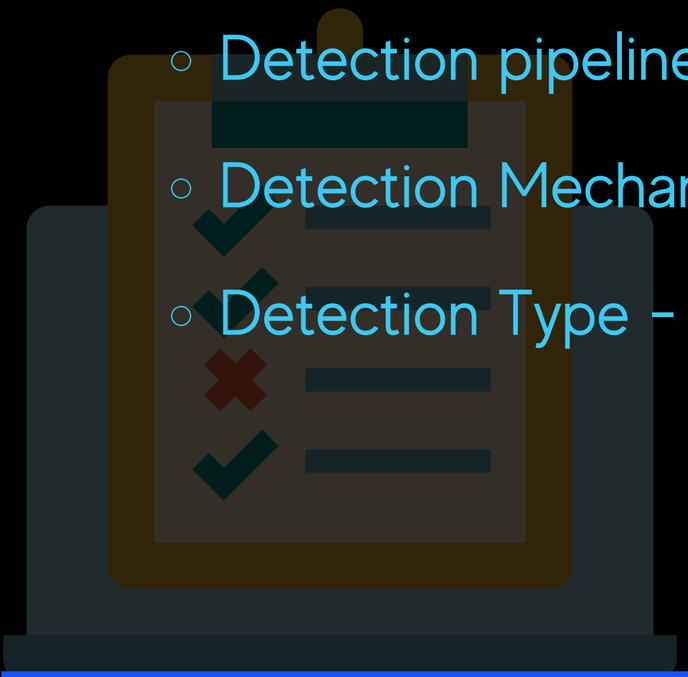
You are tasked with developing a detection mechanism for this activity.

RESEARCH

- Getting a better understanding of the threat
  - The attack surface
  - Azure AD Device Registerations
  - MFA Policies
  - Reading the pentest report
  - Reading the IR report
  - Tactic & Techniques Used

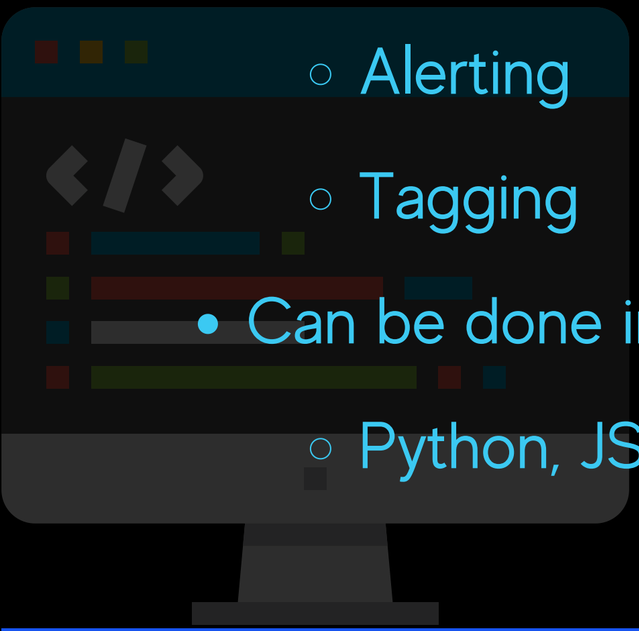
REQUIREMENTS

- What do we need to build this detection?
  - Logs & Data sources - AzureAD, Sign-in or M365 logs
  - Detection pipelines - log parsers, event parameters
  - Detection Mechanism - platform or agent
  - Detection Type - Threshold, Anomaly



DEVELOPMENT

- Building the Detection as code
  - Query
  - Logic
  - Detection
  - Alerting
  - Tagging
- Can be done in
  - Python, JSON, YAML (Sigma, Sentinel & Splunk)



DETECTION-AS-CODE IN PYTHON

```python
from panther_base_helpers import deep_get, okta_alert_context


def rule(event):
    return (deep_get(event, 'outcome', 'result') == 'FAILURE' and
            event['eventType'] == 'user.session.start')


def title(event):
    return 'Suspected brute force Okta logins to account {} due to [{}]'.format(
        deep_get(event, 'actor', 'alternateId'),
        deep_get(event, 'outcome', 'reason'))


def alert_context(event):
    return okta_alert_context(event)
```

Copy

Modules

Event Parameters

Alert

Okta Brute Force Login Rule in Panther

Source:https://panther.com/cyber-explained/detections-as-code/

TESTING

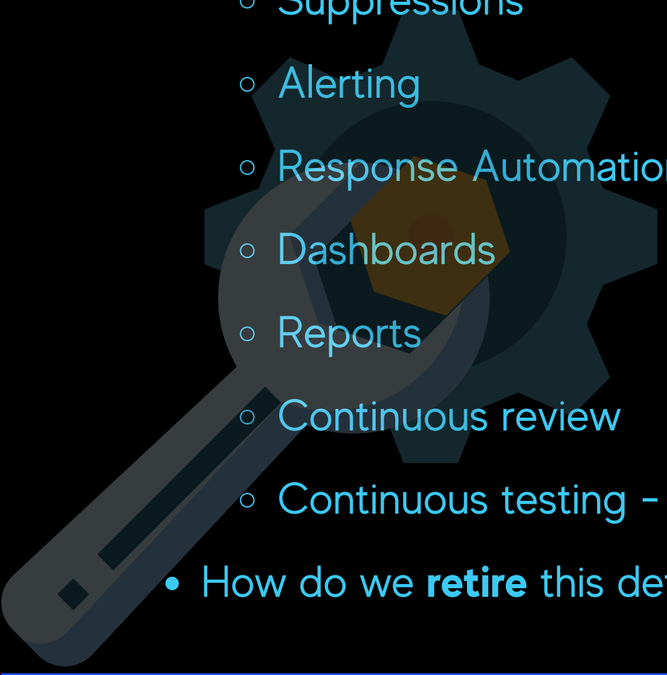
- Manual detection validation
- Automated detection validation
  - Stratus Red Team, Atomic Red Team
  - Writing your own scripts
- Unit Testing
  - Testing for proper query syntax
  - Testing for appropriate code format
  - Testing for appropriate tagging
  - Testing for necessary detection parameters

DEPLOYMENT

- Detection is pushed to production
  - Peer review from other detection engineers and security peers (IR, Internal Sec, e.t.c)

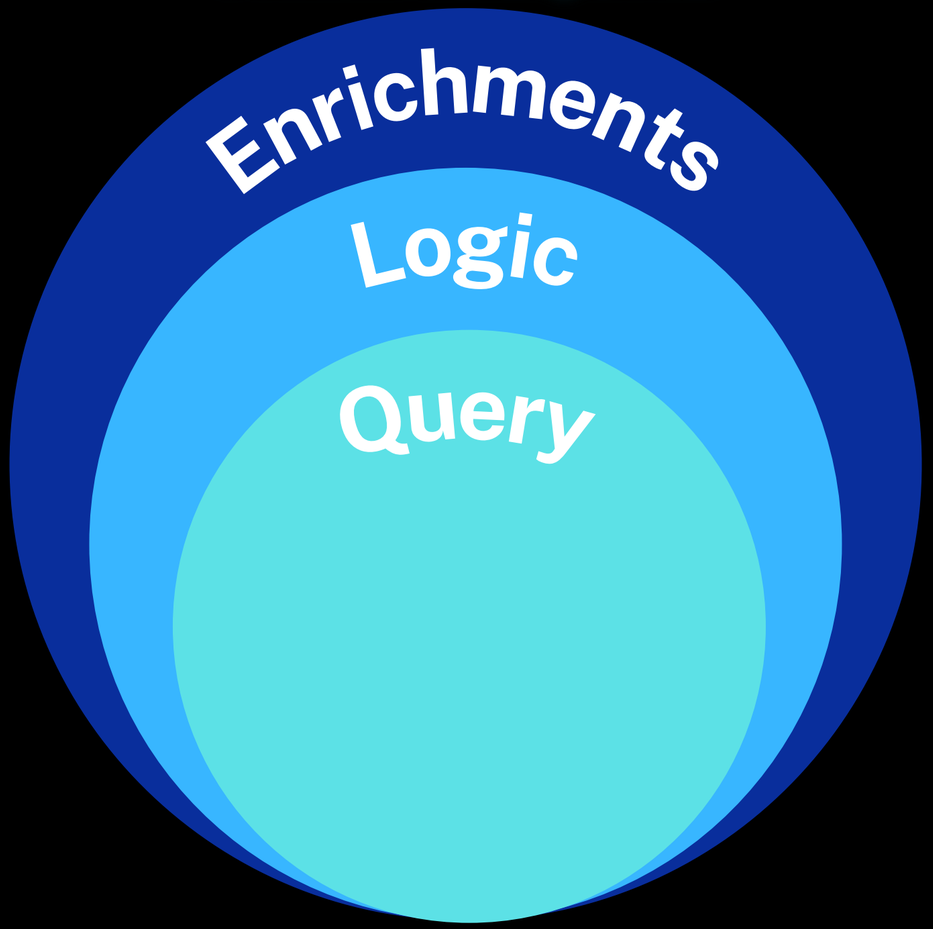
MAINTENANCE

- Adding and managing version control
- Response and triage elements for analysts
- How do we improve this detection?
  - Enrichment - threat intel
  - Tuning - thresholds, queries, alerts
  - Suppressions
  - Alerting
  - Response Automation & Playbooks/Runbooks
  - Dashboards
  - Reports
  - Continuous review
  - Continuous testing - purple team exercises
- How do we **retire** this detection if necessary?



DETECTION-AS-CODE

Detection Building Blocks



Query

The foundation of your detection. Specific attributes & parameters that define the adversary's activity.

Logic

Defines the methodology of your detection. Here you define rules, thresholds, baselines and logic statements (AND/OR).

Enrichments

Additions like metadat, tagging, exclusions, suppressions, alerting, threat intel and more are considered enrichments.

DETECTION-AS-CODE WITH SIGMA

Device Registeration or Join Without MFA

```yaml
title: Device Registration or Join Without MFA
id: 5afa454e-030c-4ab4-9253-a90aa7fcc581
description: Monitor and alert for device registration or join events where MFA was not performed.
author: Michael Epping, '@mepples21'
date: 2022/06/28
references:
  - https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/security-operations-devices#device-registrations-and-joins-outside-policy
logsource:
  product: azure
  service: signinlogs
detection:
  selection:
    ResourceDisplayName: 'Device Registration Service'
    conditionalAccessStatus: 'success'
  filter_mfa:
    AuthenticationRequirement: 'multiFactorAuthentication'
  condition: selection and not filter_mfa
falsepositives:
  - Unknown
level: medium
status: experimental
tags:
  - attack.valid_accounts
  - attack.t1078
```

Source:https://github.com/SigmaHQ/sigma/blob/master/rules/cloud/azure/azure_ad_device_registration_or_join_without_mfa.yml

DETECTION-AS-CODE WITH SIGMA

QUERY

```yaml
title: Device Registration or Join Without MFA
id: 5afa454e-030c-4ab4-9253-a90aa7fcc581
description: Monitor and alert for device registration or join events where MFA was not performed.
author: Michael Epping, '@mepples21'
date: 2022/06/28
references:
  - https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/security-operations-devices#device-registrations-and-joins-outside-policy
logsource:
  product: azure
  service: signinlogs
detection:
  selection:
    ResourceDisplayName: 'Device Registration Service'
    conditionalAccessStatus: 'success'
  filter_mfa:
    AuthenticationRequirement: 'multiFactorAuthentication'
  condition: selection and not filter_mfa
falsepositives:
  - Unknown
level: medium
status: experimental
tags:
  - attack.valid_accounts
  - attack.t1078
```

Source:https://github.com/SigmaHQ/sigma/blob/master/rules/cloud/azure/azure_ad_device_registration_or_join_without_mfa.yml

DETECTION-AS-CODE WITH SIGMA

RULE LOGIC

```yaml
title: Device Registration or Join Without MFA
id: 5afa454e-030c-4ab4-9253-a90aa7fcc581
description: Monitor and alert for device registration or join events where MFA was not performed.
author: Michael Epping, '@mepples21'
date: 2022/06/28
references:
  - https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/security-operations-devices#device-registrations-and-joins-outside-policy
logsource:
  product: azure
  service: signinlogs
detection:
  selection:
    ResourceDisplayName: 'Device Registration Service'
    conditionalAccessStatus: 'success'
  filter_mfa:
    AuthenticationRequirement: 'multiFactorAuthentication'
  condition: selection and not filter_mfa
falsepositives:
  - Unknown
level: medium
status: experimental
tags:
  - attack.valid_accounts
  - attack.t1078
```

LOGIC

Source:https://github.com/SigmaHQ/sigma/blob/master/rules/cloud/azure/azure_ad_device_registration_or_join_without_mfa.yml

DETECTION-AS-CODE WITH SIGMA

ENRICHMENTS

```yaml
title: Device Registration or Join Without MFA
id: 5afa454e-030c-4ab4-9253-a90aa7fcc581
description: Monitor and alert for device registration or join events where MFA was not performed.
author: Michael Epping, '@mepples21'
date: 2022/06/28
references:
  - https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/security-operations-devices#device-registrations-and-joins-outside-policy
logsource:
  product: azure
  service: signinlogs
detection:
  selection:
    ResourceDisplayName: 'Device Registration Service'
    conditionalAccessStatus: 'success'
  filter_mfa:
    AuthenticationRequirement: 'multiFactorAuthentication'
  condition: selection and not filter_mfa
falsepositives:
  - Unknown
level: medium
status: experimental
tags:
  - attack.valid_accounts
  - attack.t1078
```

ENRICHMENTS

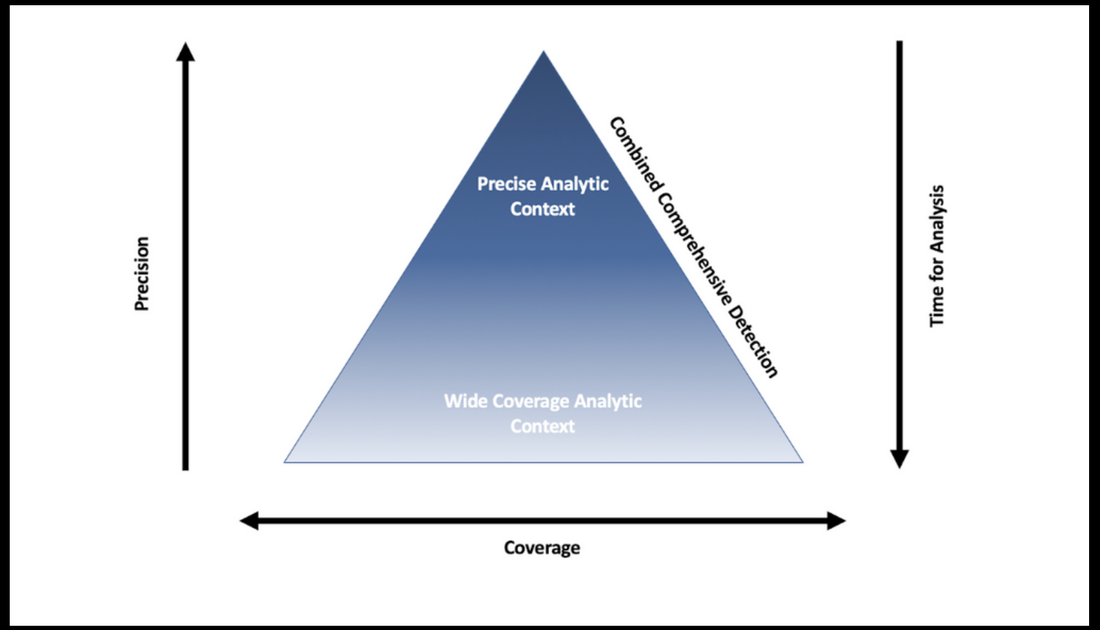
ENRICHMENTS

Source:https://github.com/SigmaHQ/sigma/blob/master/rules/cloud/azure/azure_ad_device_registration_or_join_without_mfa.yml

DETECTION RESILIENCE

DETECTION-IN-DEPTH

- Precision
- Layers
- Blind Spots
- Evasion



Source:https://posts.specterops.io/detection-in-depth-a2392b3a7e94

DEMOCRATIZING YOUR DETECTION FUNCTION

THE DETECTION CHAMPION

A play on security champion

- Technical Competency across various technologies
- Creating Awareness of your detection team
- Collaboration with Dev, Ops and other Security Teams
- Mentoring junior detection engineers

PUTTING IT ALL TOGETHER

- **Cloud Detection Engineering**
  - Identifying suspicious activities in our Cloud Environment
  - Staying ahead as defenders
- **Approaching Cloud Detection Engineering**
  - Understanding the Adversary
  - Observability Considerations- Telemetry, Logging & Cloud Altitudes
- **Implementing Cloud Detection Engineering**
  - Detection Engineering Lifecycle
  - Detection-as-Code
- **Operationalizing Cloud Detection Engineering**
  - Detection Resilience
  - The Detection Champion

SOURCES & ADDITIONAL REFERENCES

- **Detection Development Lifecycle**
  - https://medium.com/snowflake/detection-development-lifecycle-af166fffb3bc
  - https://www.securonix.com/blog/managing-security-threats-with-detection-development-life-cycle/
- **Detection-in-depth**
  - https://posts.specterops.io/detection-in-depth-a2392b3a7e94
  - https://inquest.net/blog/2020/08/28/Detection-in-Depth
- **Security Champion**
  - https://brightsec.com/blog/what-is-a-security-champion/
- **Detection Engineering & Detection-as-code**
  - https://panther.com/cyber-explained/detection-engineering-benefits/
  - https://panther.com/cyber-explained/detections-as-code/
- **Cloud Altitudes**
  - https://thenewstack.io/data-control-management-three-planes-different-altitudes/

SOURCES & ADDITIONAL REFERENCES (CONTD.)

- **IDS/IPS**
  - https://www.lbmc.com/blog/ids-vs-ips/
- **Sigma**
  - https://github.com/SigmaHQ
- **Azure Docs & Sentinel**
  - https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/security-operations-devices#device-registrations-and-joins-outside-polic
- **https://github.com/Azure/Azure-Sentinel**
  - https://www.ais.com/tag/threat-detection/
- **MITRE Cloud Matrix**
  - https://attack.mitre.org/matrices/enterprise/cloud/
- **Automated Detection Validation**
  - https://stratus-red-team.cloud/
  - https://atomicredteam.io/

Questions