# NEW SECURITY MODELS FOR CLOUD-NATIVE ARCHITECTURE

JAMIE LEWIS

VENTURE PARTNER, RAIN CAPITAL*

NOVEMBER 13, 2019

*Any investments disclosed

TECHVISION
CHRYSALIS

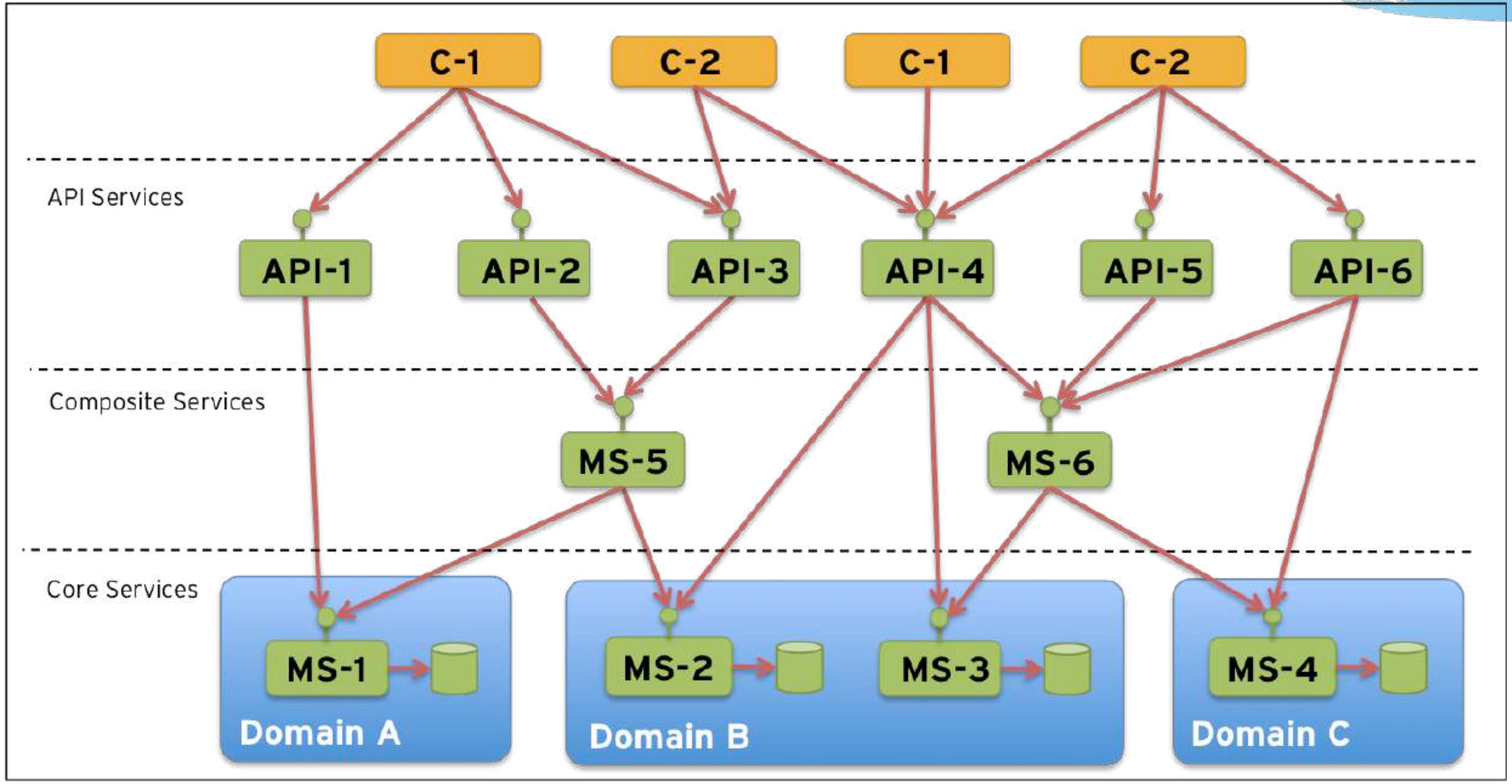
# The move to cloud-native has a profound impact on security posture and operations.

# Containerization

# Serverless computing

# CI/CD and automation

# Microservices



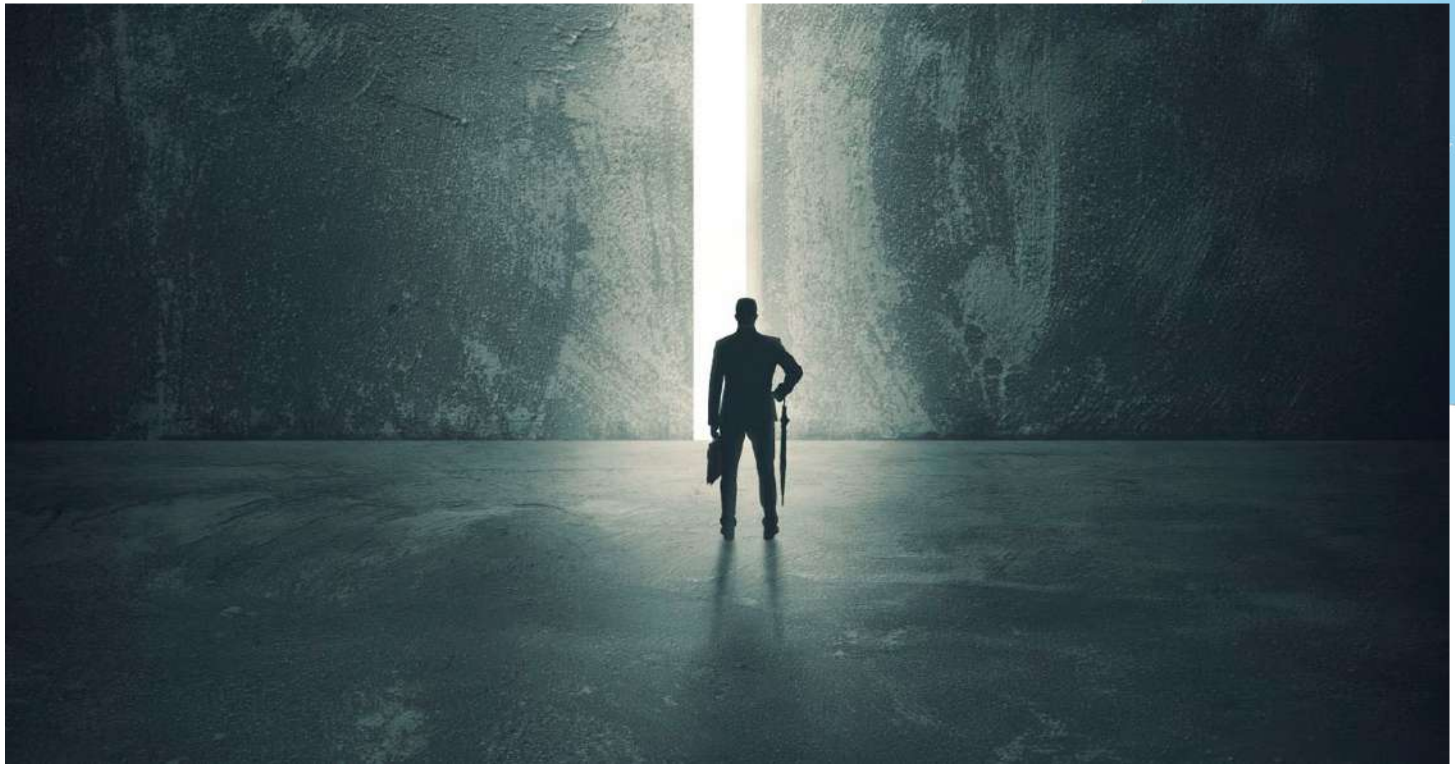
TECHVISION CHRYSALIS®

Ephemeral workloads.

# The Service Mesh,
# AKA distributed systems.

# Implications.

# The notions of an “application” running on a “machine” in a persistent “state” are obsolete.

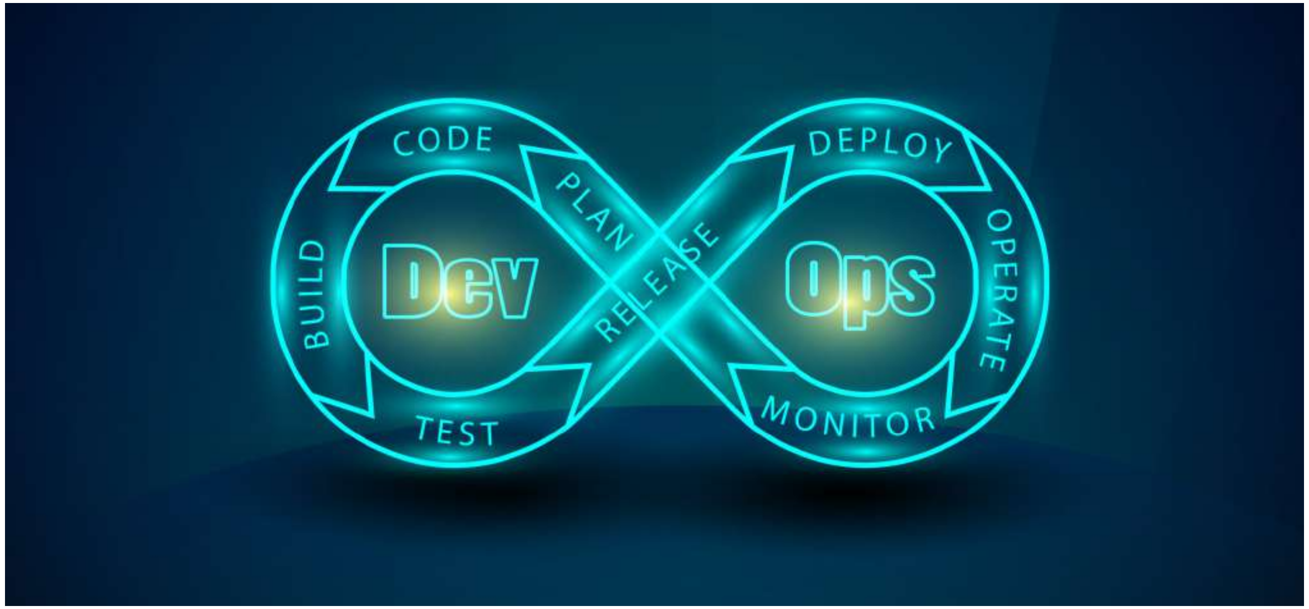


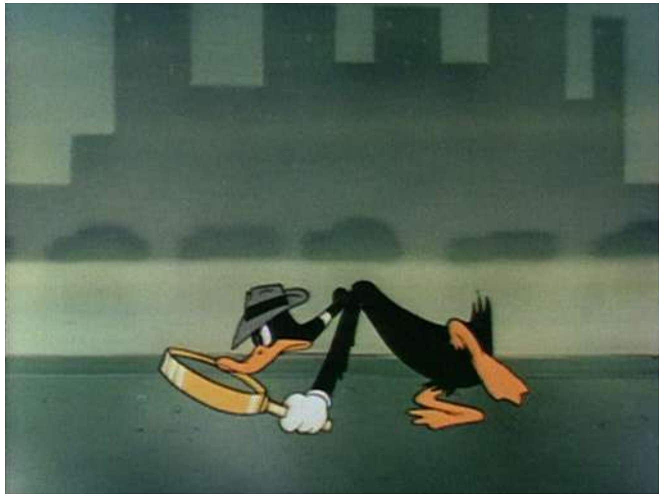
Distributed architectures introduce unpredictable dynamics and unanticipated failure modes.

# Security policies that are blind to service-to-service communications will fail.

Controls that lack horizontal scalability cannot keep pace.

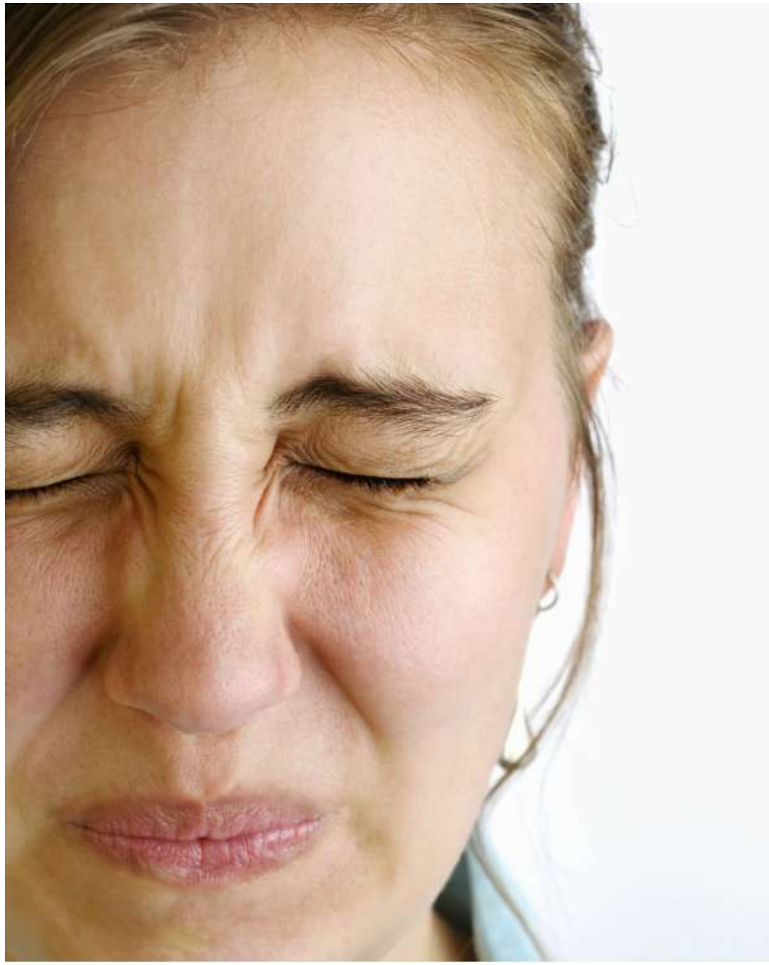
# DevOps is challenging organizing principles that have driven enterprise security operations for years.

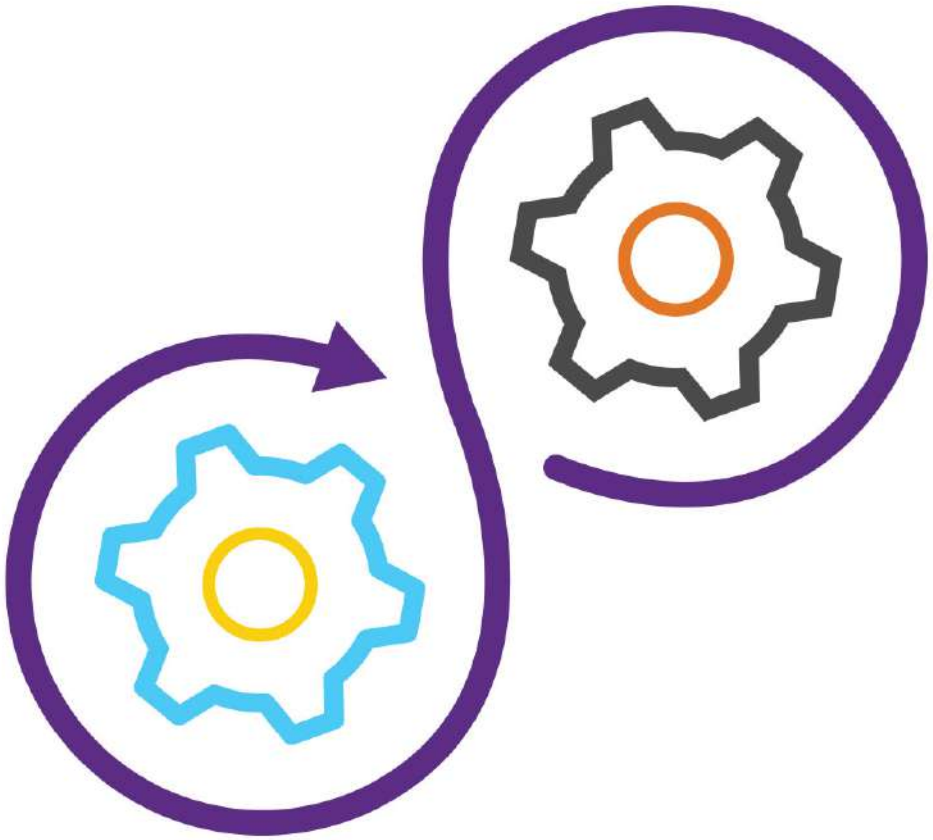




The logical conclusion?

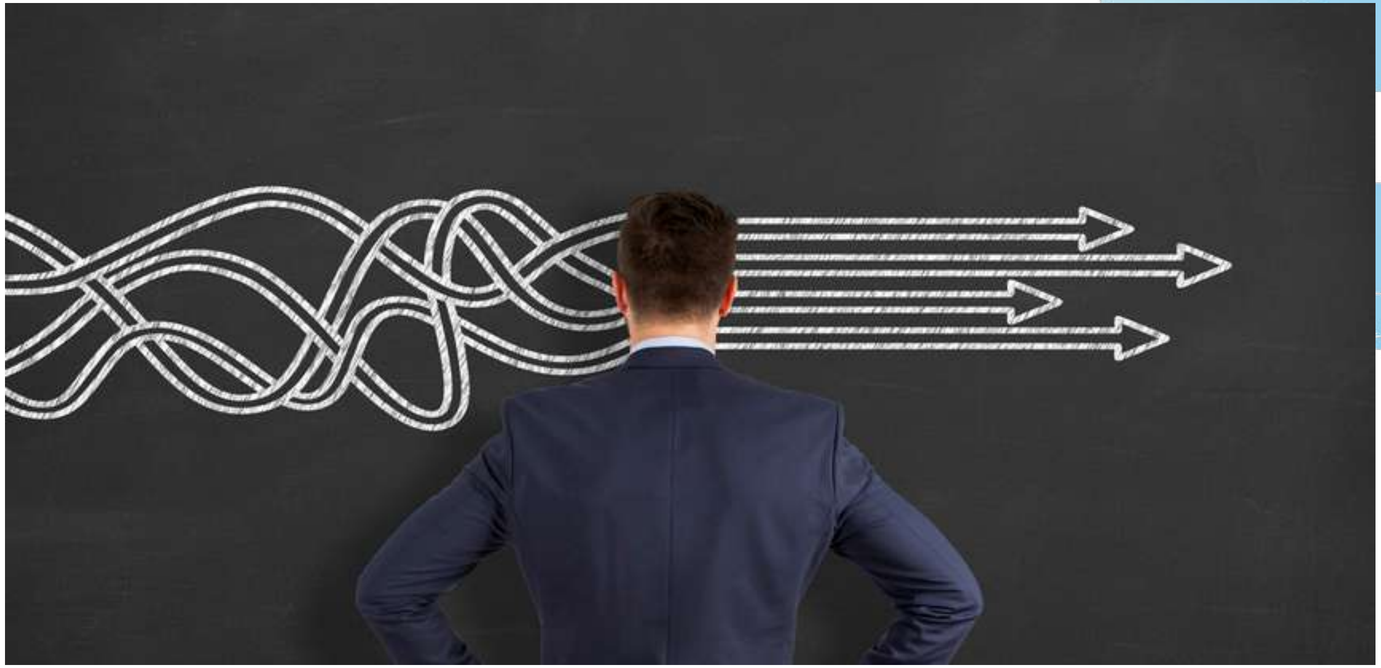
# “DevSecOps”.
# Or some variation on that theme.





Continuous security pipelines.

# It’s also an opportunity.



Bring security into architectural and organizational alignment with the systems they protect.

# Technical and Operational (Cultural)

NETFLIX

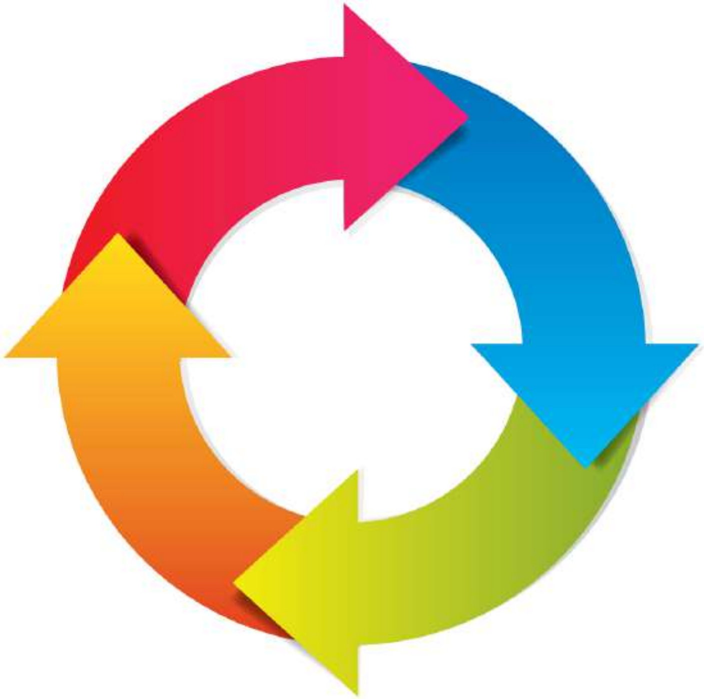
STITCH FIX™

By looking at pioneers, we can see where enterprises need to head.

# What are the requirements?

Extensive, real-time observability.

# Rapid, iterative feedback loops.

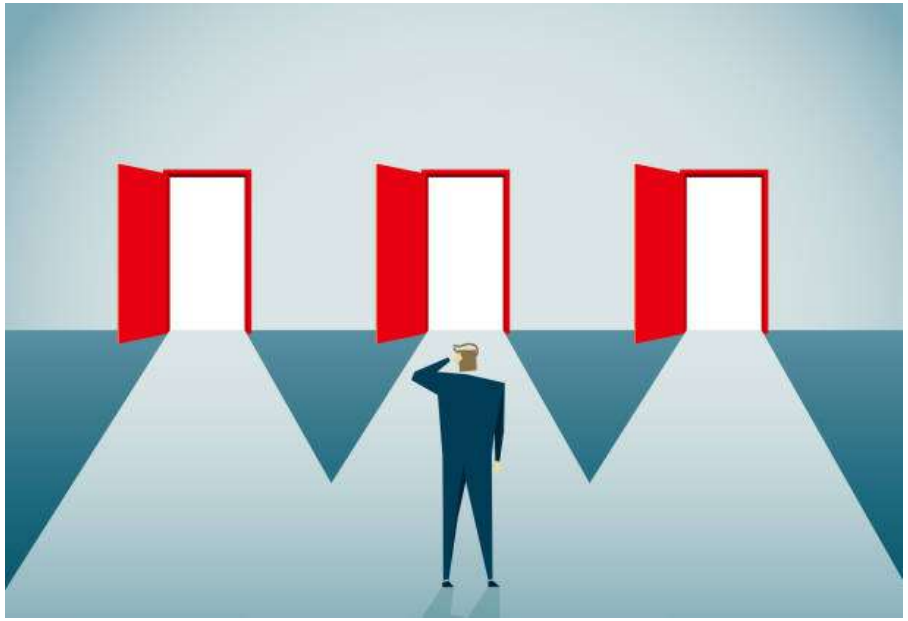


Controls that adapt.

# An engineering approach to security problems.

```
    mirror_mod.use_y = True
    mirror_mod.use_z = False
elif _operation == "MIRROR_Z":
    mirror_mod.use_x = False
    mirror_mod.use_y = False
    mirror_mod.use_z = True

    #selection at the end -add back the deselected mirror
mirror_ob.select= 1
modifier_ob.select=1
bpy.context.scene.objects.active = modifier_ob
print("Selected" + str(modifier_ob)) # modifier ob is the active
```



Several examples:
Envoy Proxy
Chaos Engineering
Detection Engineering

# Detection Engineering*

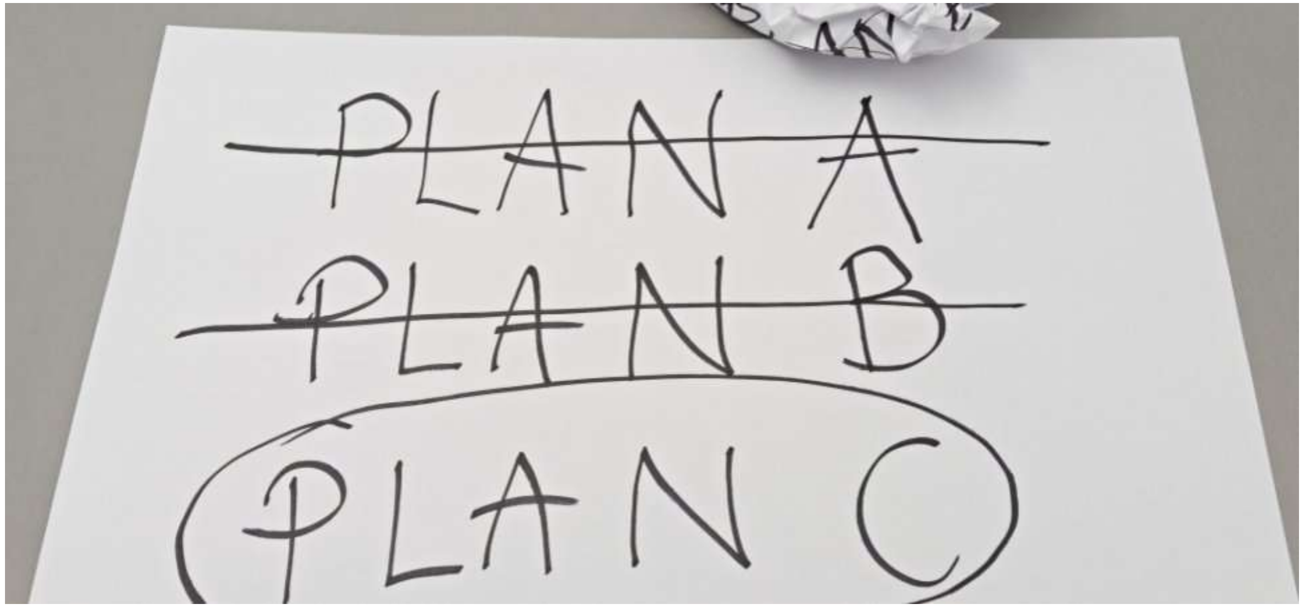
*See Rain Capital paper for deeper detail and case study

# Detection

Most enterprises have invested significantly more in prevention than they have in detection.

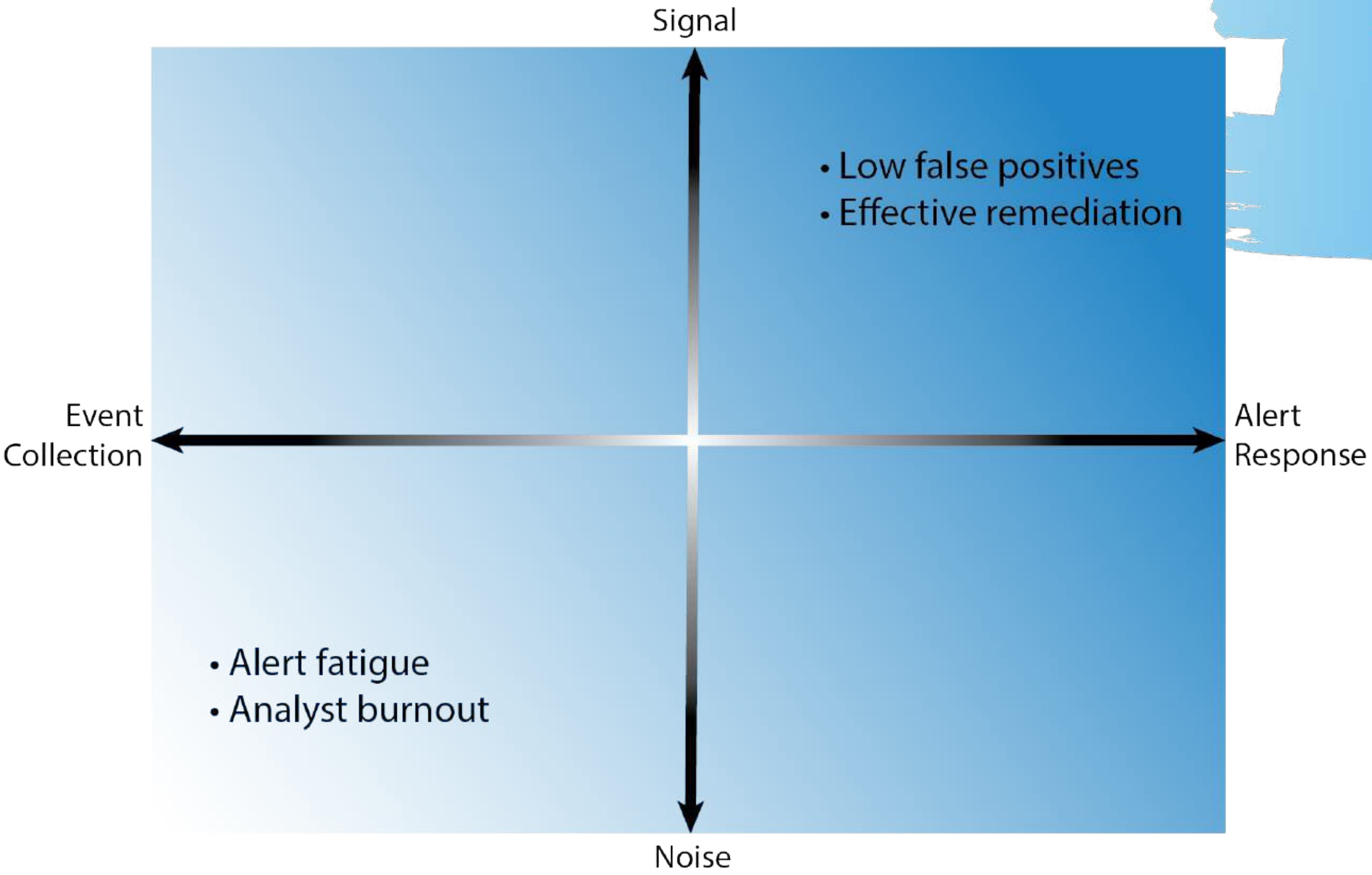
But products focused on prevention are failing.

# Assume you've been breached.



# Engineering

Massive volumes of alerts and logs make centralized, manual processing ineffective and unsustainable.



Signal

- Low false positives
- Effective remediation

Event Collection

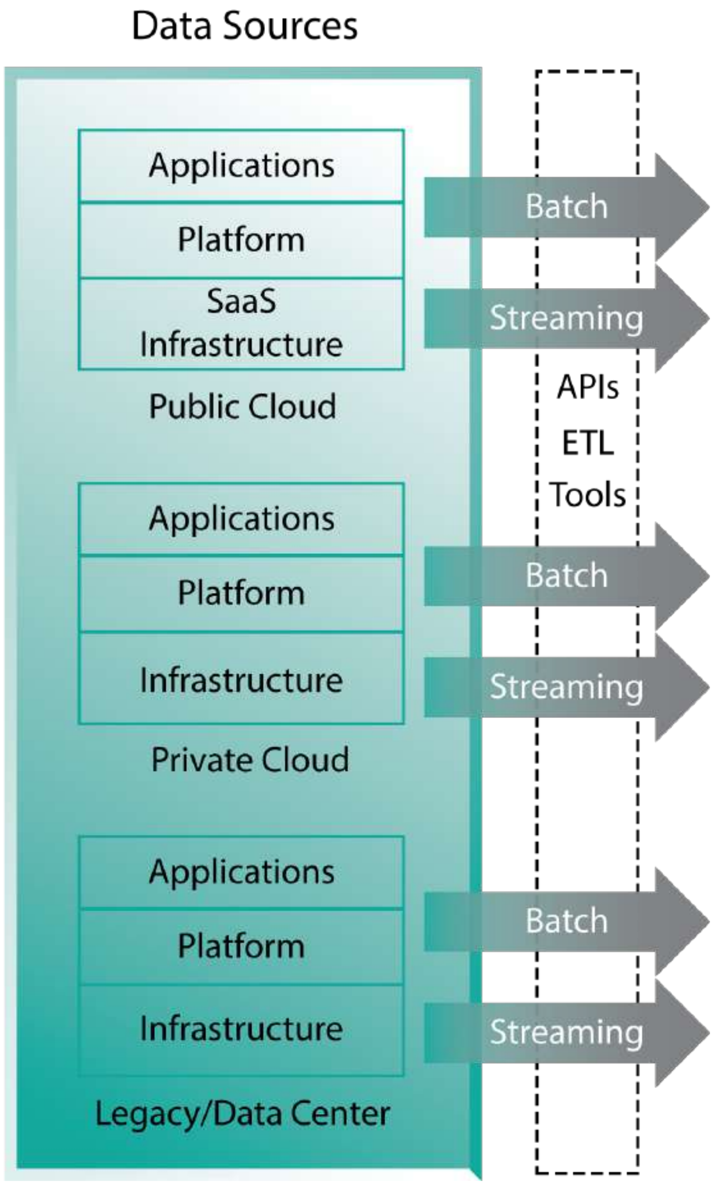
Alert Response

- Alert fatigue
- Analyst burnout

Noise

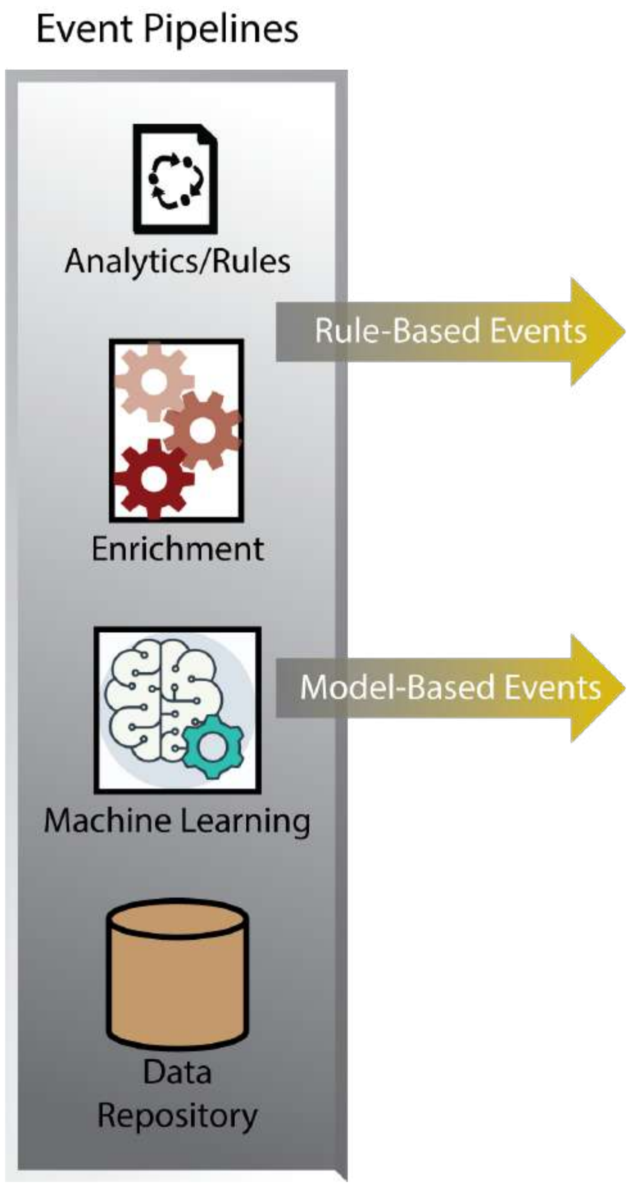
# Detection Engineering

The continuous process of deploying, tuning, and operating automated infrastructure for finding active threats and orchestrating responses.



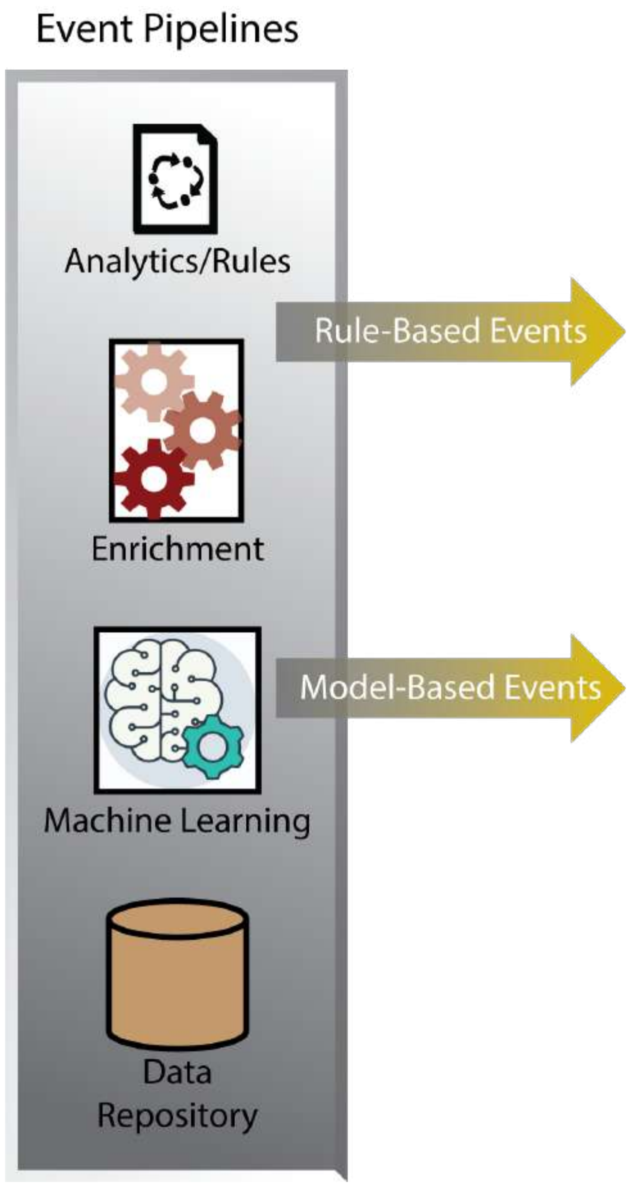
- "The person who logs the data should be the person who consumes that data."
- Cloud-native systems require new tools and instrumentation
- The Envoy Proxy* is one example
- Batch as necessary
- But instrument to stream data in real time whenever possible
- Zipkin, a distributed trace system for microservices

*Rain has an investment in Tetrate.

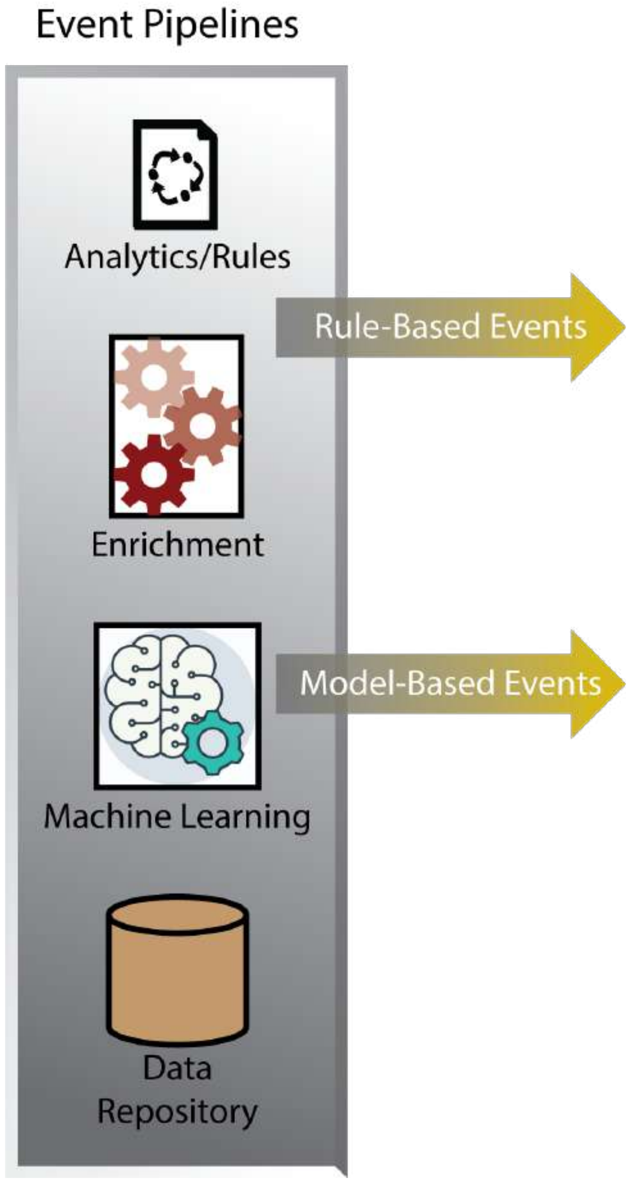


- Avoid normalizing event data
- Create pipelines for each data source, base workflow on data in that system
- Templates, reusable modules streamline work on common types

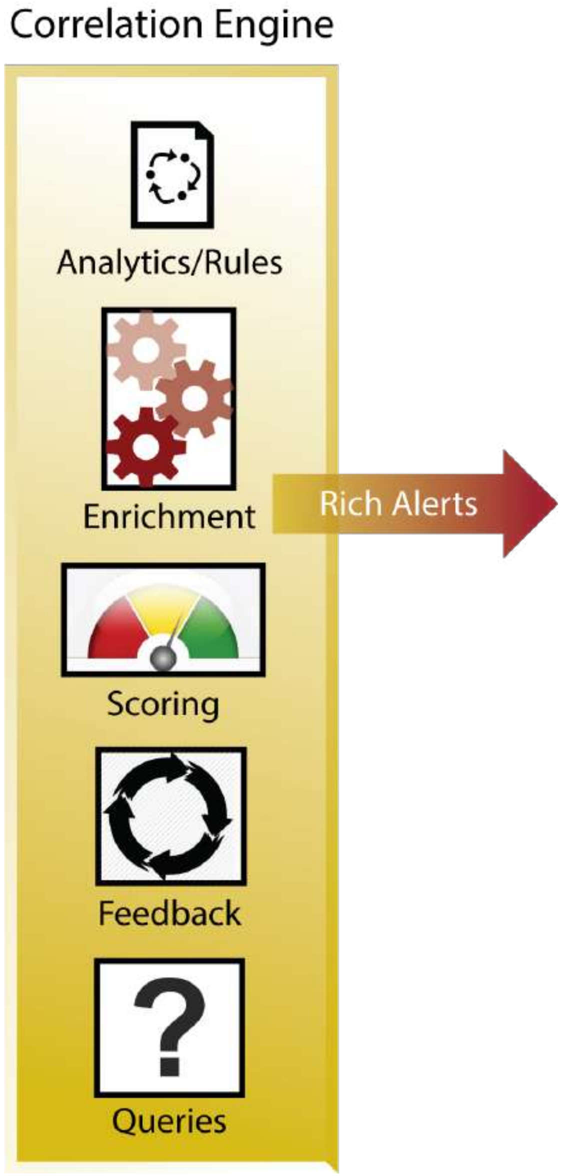
- Define a rigorous framework for events, rules, and alerts
- See Palantir's post
- Follow engineering standards for software dev: peer review, version-controlled repository
- Dev, refinement part of standard security processes, such as post-incident reviews



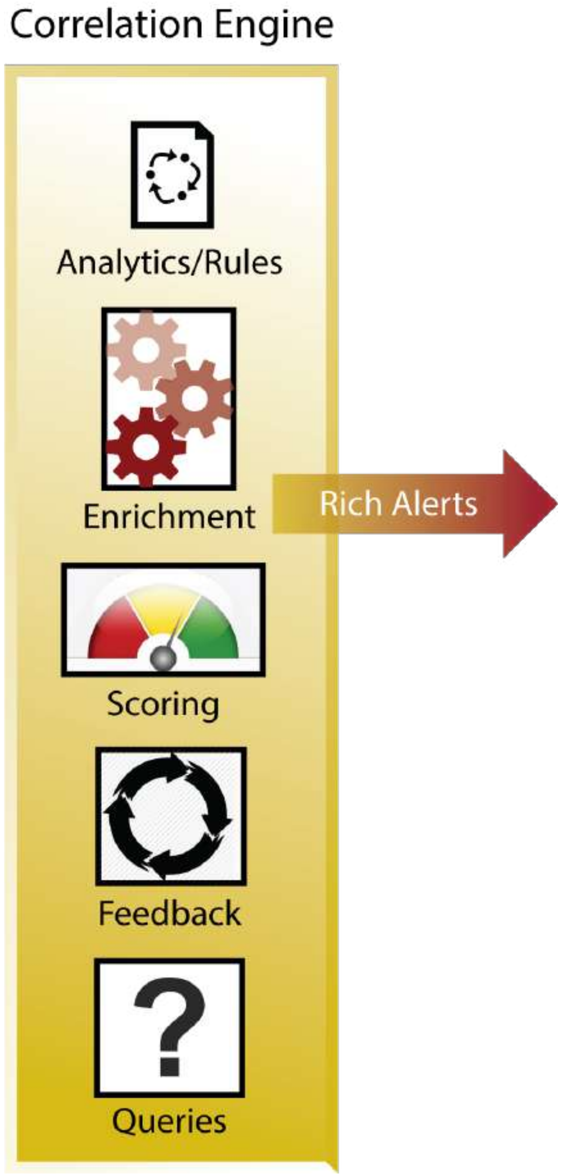
- Rules generate events
- Machine learning, feature extraction algorithms build models of actions
- Anomalies generate model-based events
- Less expensive automation enriches event data, improves quality
- Passing only events that warrant investigation



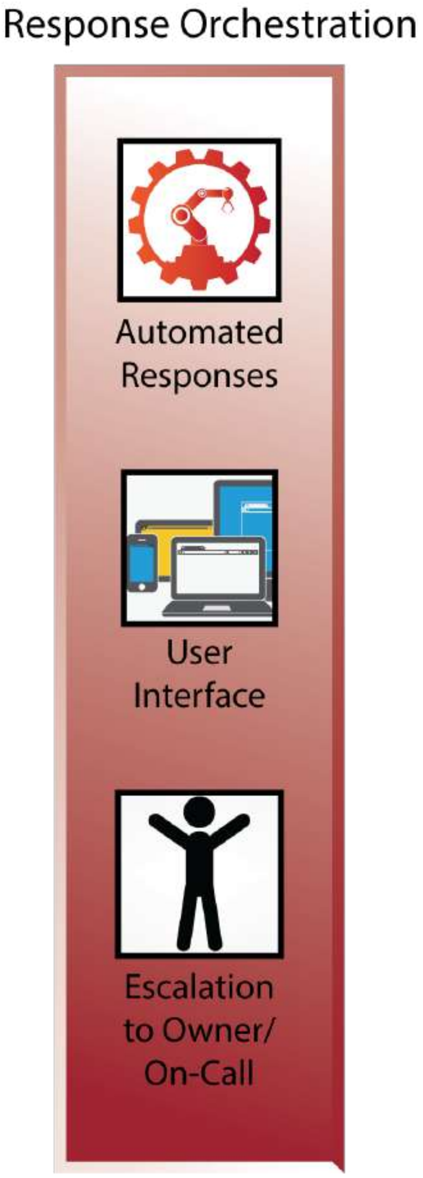
- Apache Kafka is Netflix’s event pipeline backbone
- For rules /analytics engine, Netflix uses Apache Spark, moving to Apache Flink
- Apache Hive is the data warehouse



- Typically a data analytics platform for scoring events
- Rules drive alert creation
- Creating context with more enrichment
- Accounts involved, security classification, who they work for, contact info, privilege levels, etc.
- Automated comms like Slackbots, reach out to get confirmations of activity or more info
- Rules determine response
- Alert a human, invoke automation, or both



- Netflix uses Elasticsearch (data analytics platform for its streaming service)
- Enriches event data via GraphQL queries to APIs on relevant systems



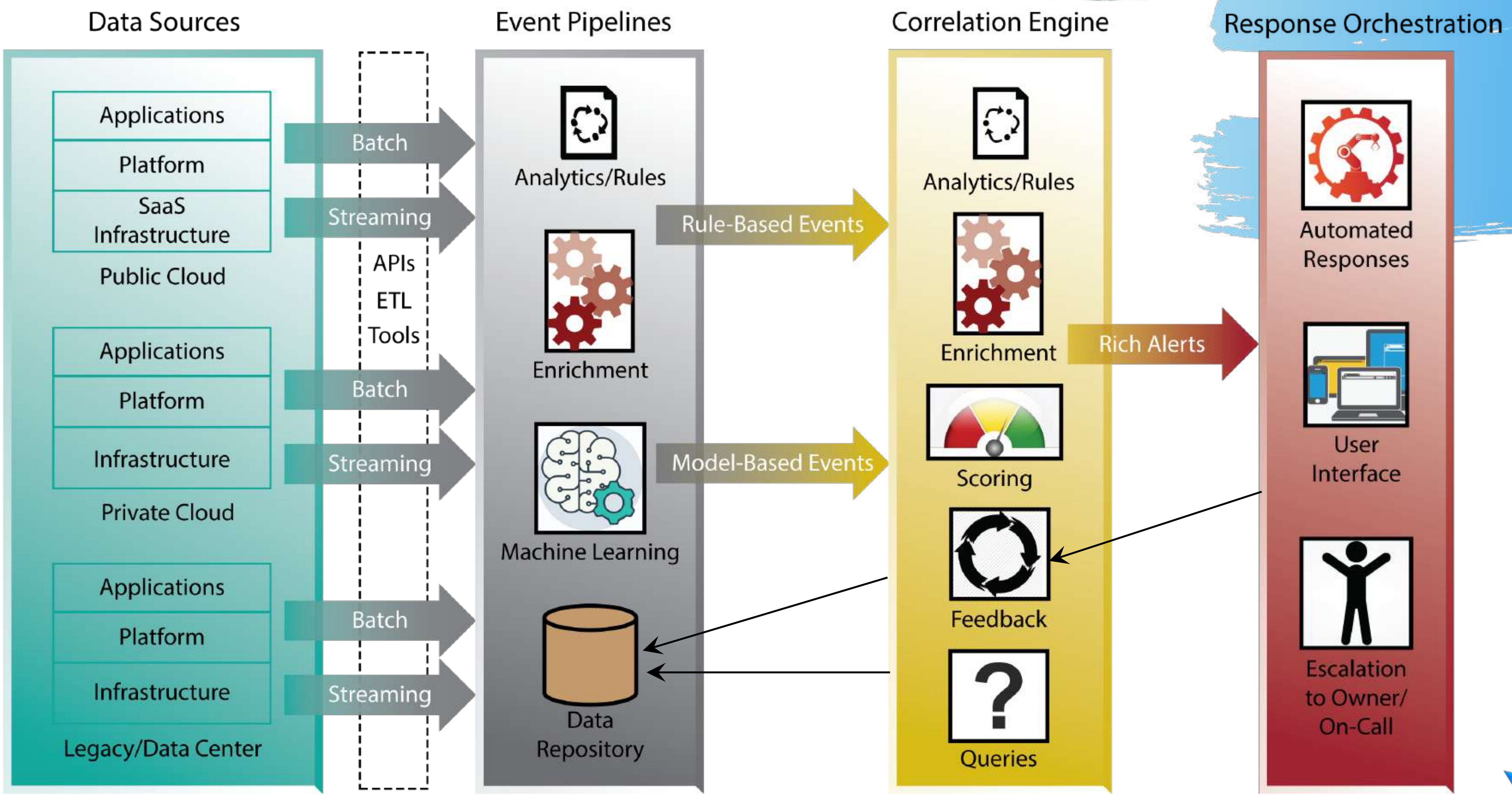
Response Orchestration

Automated Responses

User Interface

Escalation to Owner/ On-Call

- Automate common fixes
- Or, alert includes option to invoke automation
- Comms bring relevant players into the loop
- Netflix has developed automated forensics, with Diffy



TECHVISION CHRYSALIS®

# All well and good, but we can’t build our own.

🤔

CAPSULE8

elastic

mistnet
Enterprise security. Demystified.

splunk>

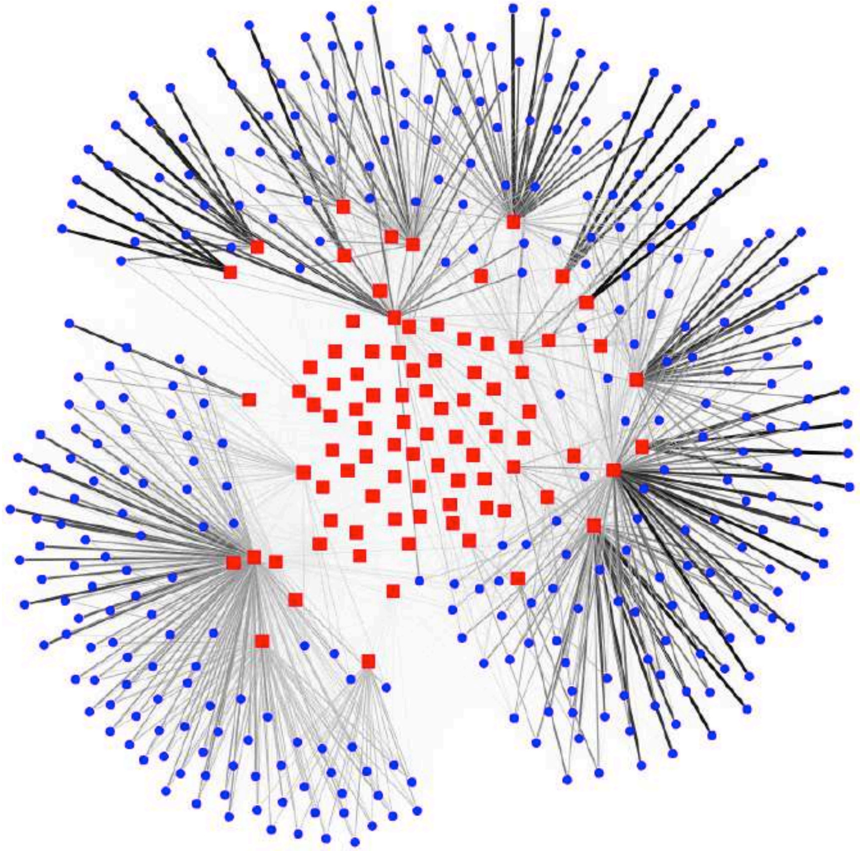
Candidates include Capsule8*, the Elastic Stack (ELK), MistNet, and Splunk.

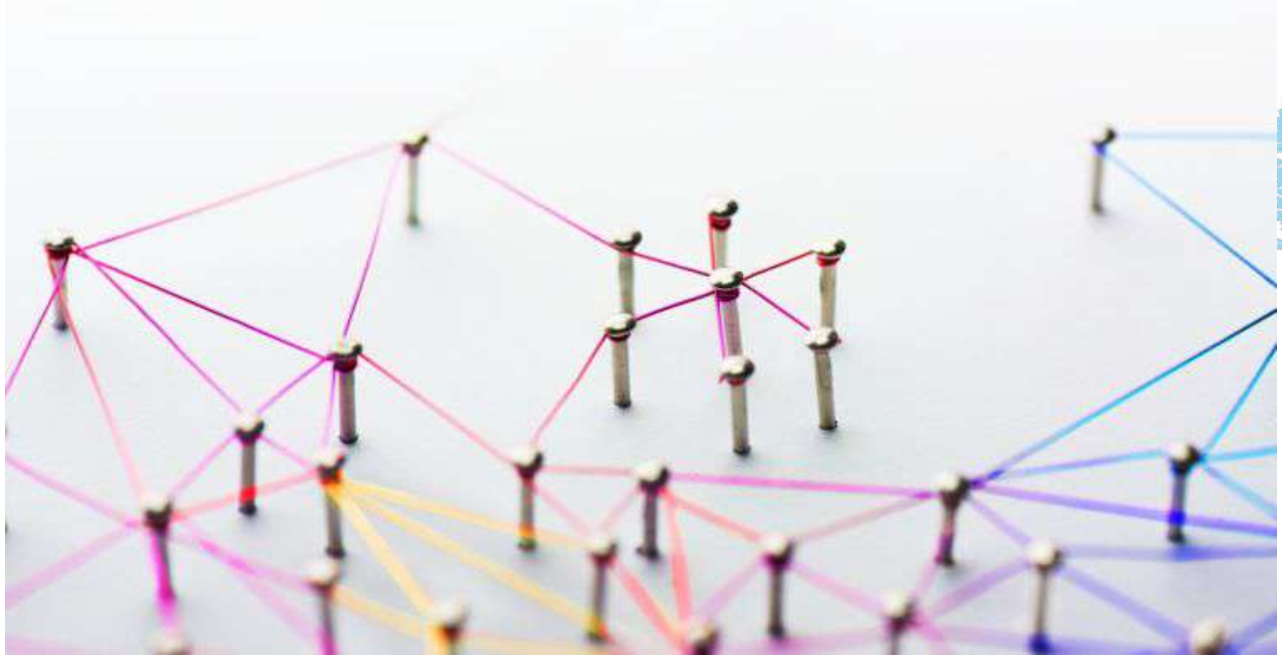
*Rain has an investment in Capsule 8

# Operational Implications:
# DevOps, culture, and decentralization.

Requires taking a hard look at the traditional SOC.

# Assumption: Centralized analyst team understands most (or all) of an organization's systems . . .



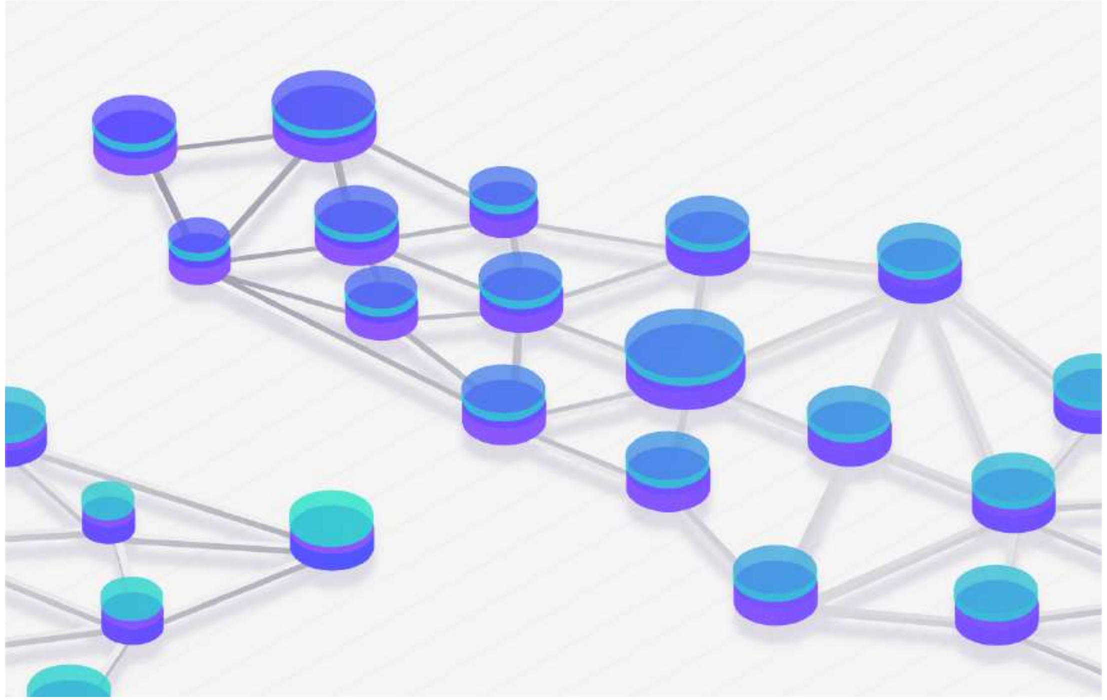


. . . has experience with most (or all) of the organization's security systems . . .

# . . . and can define and analyze threats.

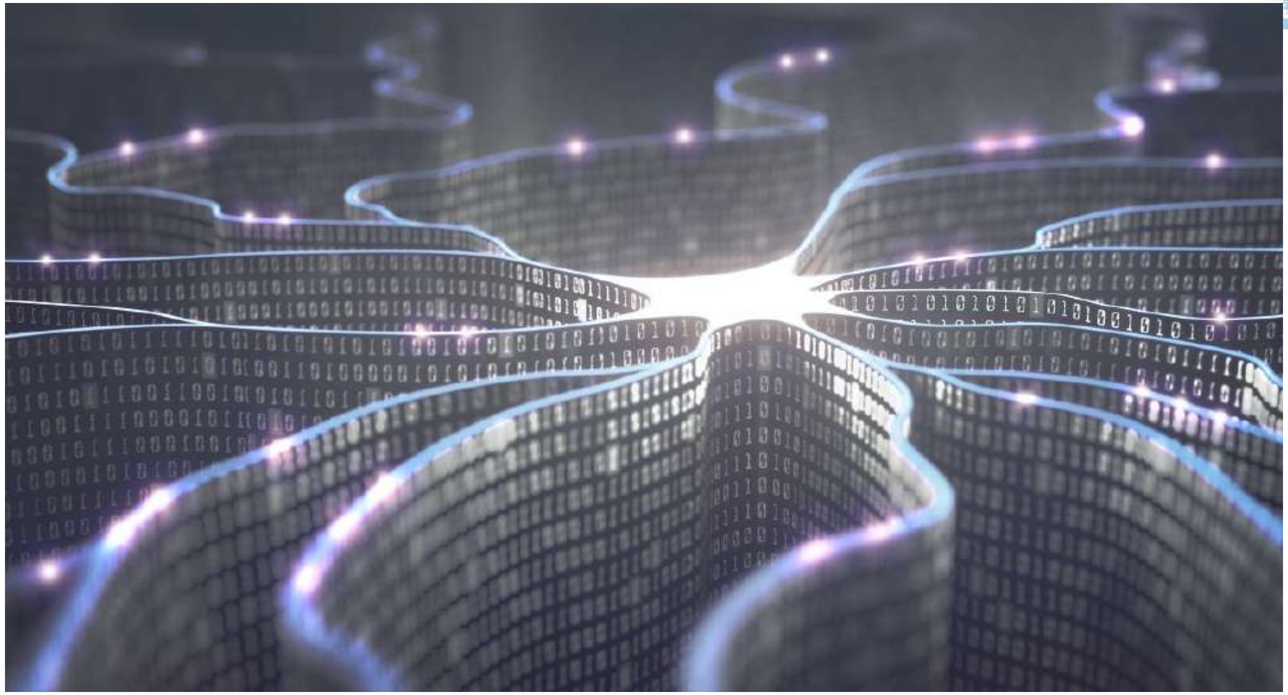
Really?

# The “SOC-less enterprise”.



It’s about decentralizing those capabilities.

# Integrating security functions and people more directly with the DevOps process.



Moving alert triage to system owner/on-call, whether a security or application team.

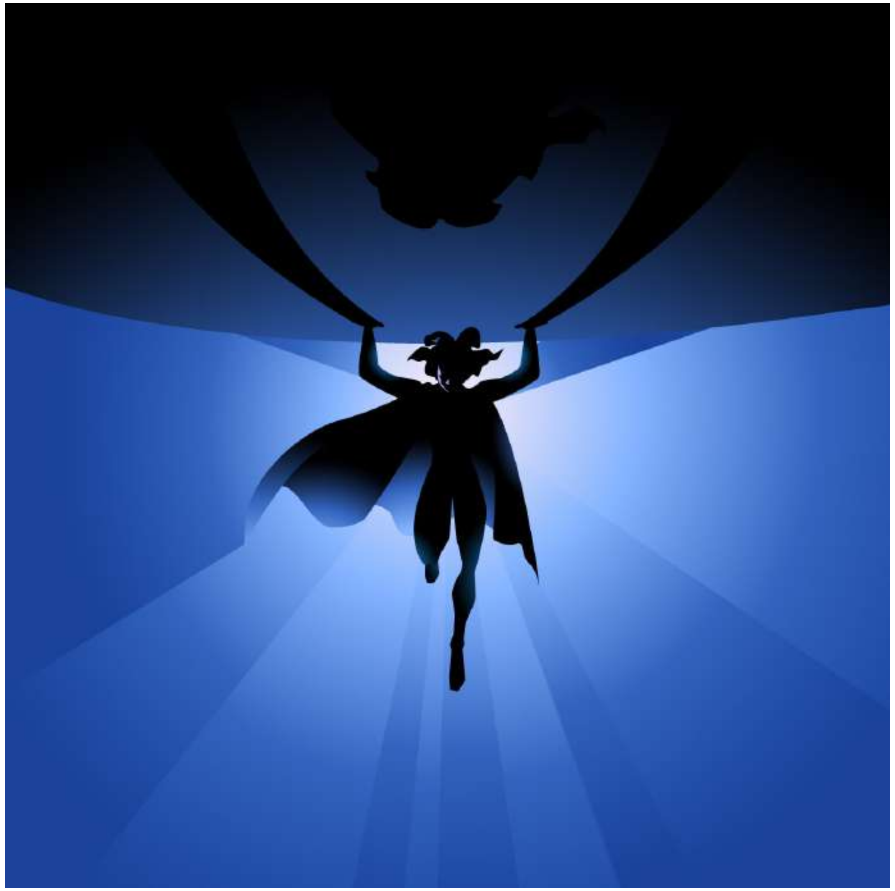
# Assumption: With good detection engineering, you can bring developers and system owners up to speed on a security alert . . .

. . . more easily than you can teach a security person the details of a production system.

With context and enrichment, alerts can tell system owners what an alert means, give them a set of response options.

Accountability for reliability extends to accountability for security.

# (But responsibility for both the quality of and the response to the alert rests with the security team.)



“That’s all great. But we’re not Netflix.”

# When and how depends on your business, your risk profile.

And how fast and far you go with cloud-native.

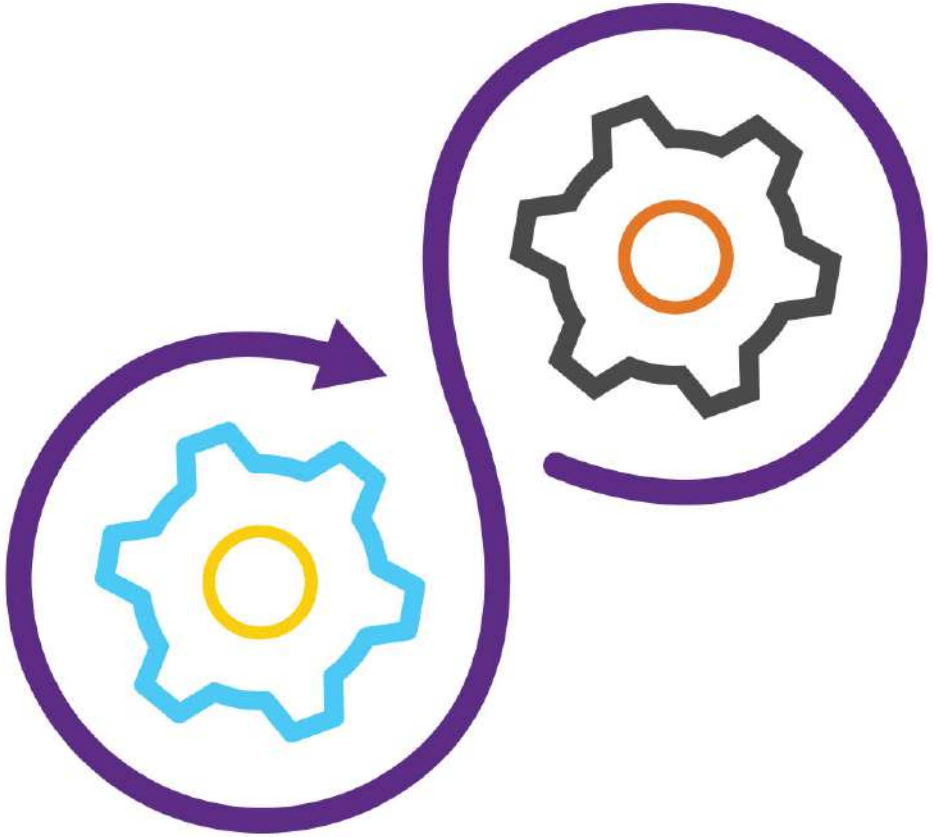
# Build a roadmap that leads to a more decentralized or hybrid security organization.

Project 5
Project 4
5
4
Project 3
3
Project 2
2
Project 1
1
1 Your description of project
2 Your description of project
3 Your description of project
4 Your description of project
5 Your description of project

TECHVISION CHRYSALIS®

In conclusion . . .

# Aligning security with the cloud-native technology stack and DevOps mindset means creating continuous security pipelines.



# Detection Engineering

One way organizations can accomplish that goal.

Enterprises moving to cloud-native technologies should consider how and when to incorporate these practices into their security programs.

Being careful to understand
the importance that culture change plays in
successfully making that transition.

# Thank you.

# Links:

- DevSecOps and Detection Engineering: New Approaches To Security (Rain Report)
- Security Chaos Engineering (Rain Report)
- Envoy Is the Real Deal (Rain Blog post)
- Palantir Alert Framework
- Netflix presentation on automated forensics

TECHVISION

CHRYSALIS