# DEVSECOPS BOOTCAMP

## BUILDING RUGGED SOFTWARE

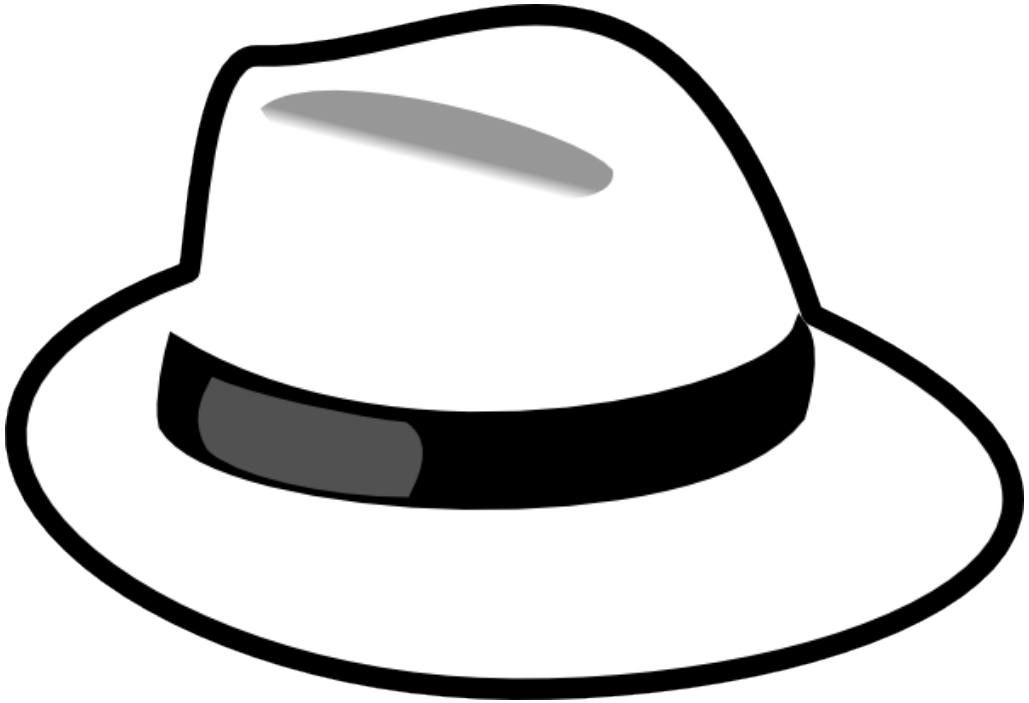
YEAR ONE / WEEK ONE / LESSON TWO

# Anatomy of an attack

- Attackers are a constant threat to online business and software
- Attackers identify gaps in security controls by running attacks in a specific order to gain access, pull data and accomplish their mission.
- Attackers can get lucky or more targeted in their attacks.
- Attacks can be slow and persistent or fast to get to a breach.

# Getting into the mind state of an attacker

- How would they attack us?
- Why would they attack us?
- What do we have that is valuable to an attacker?



# Motivations of an attacker

- Recreational
- Monetary
- Fraud
- Data
- Computing power
- Political

# Attack Map Introduction

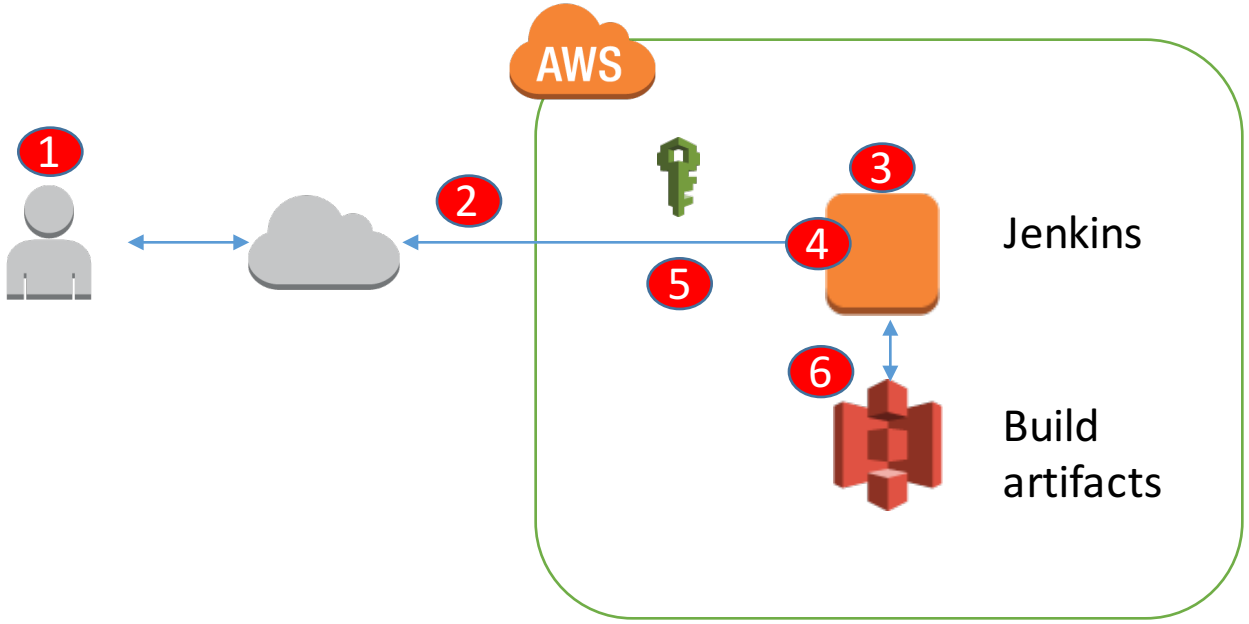
- An attack map is a graphical representation of the attack surface of an application and or environment
- Helps developers get ahead of attackers by understanding our attack surface
- Helps the Red Team to quickly verify vulnerability remediation and mitigations
- Allows developers to understand the weaknesses within their software, areas of attack and address the most important weaknesses quickly/efficiently
- Enables developers to design their applications to be resilient to attacks

# Attack Map Creation

- Create a graphical representation of your application including all communication flows and technologies being used
- Gather a list of potential vulnerabilities and areas of attack.
- Think about Confidentiality, Integrity and Availability for each connection/interaction within the application
- Map the attacks/vulnerabilities to the graphical representation
- Create a key that allows for mapping attack descriptions to the graphical attack map
- Include this document as an ATTACKS.md file in your repository

# Lab 2 - Attack Maps

1.
2.
3.
4.
5.
6.
7.



# Example Attack Map Key

**Data Center Threats**

1. Denial of Service of application
2. Malicious insider access to physical app server host
3. Malicious outsider access to physical app server host
4. Some AWS access keys logged
5. Some Key Encryption Keys and AWS access keys logged
6. All Key Encryption Keys compromised from Hardware Security Module
7. Untrusted employee departure

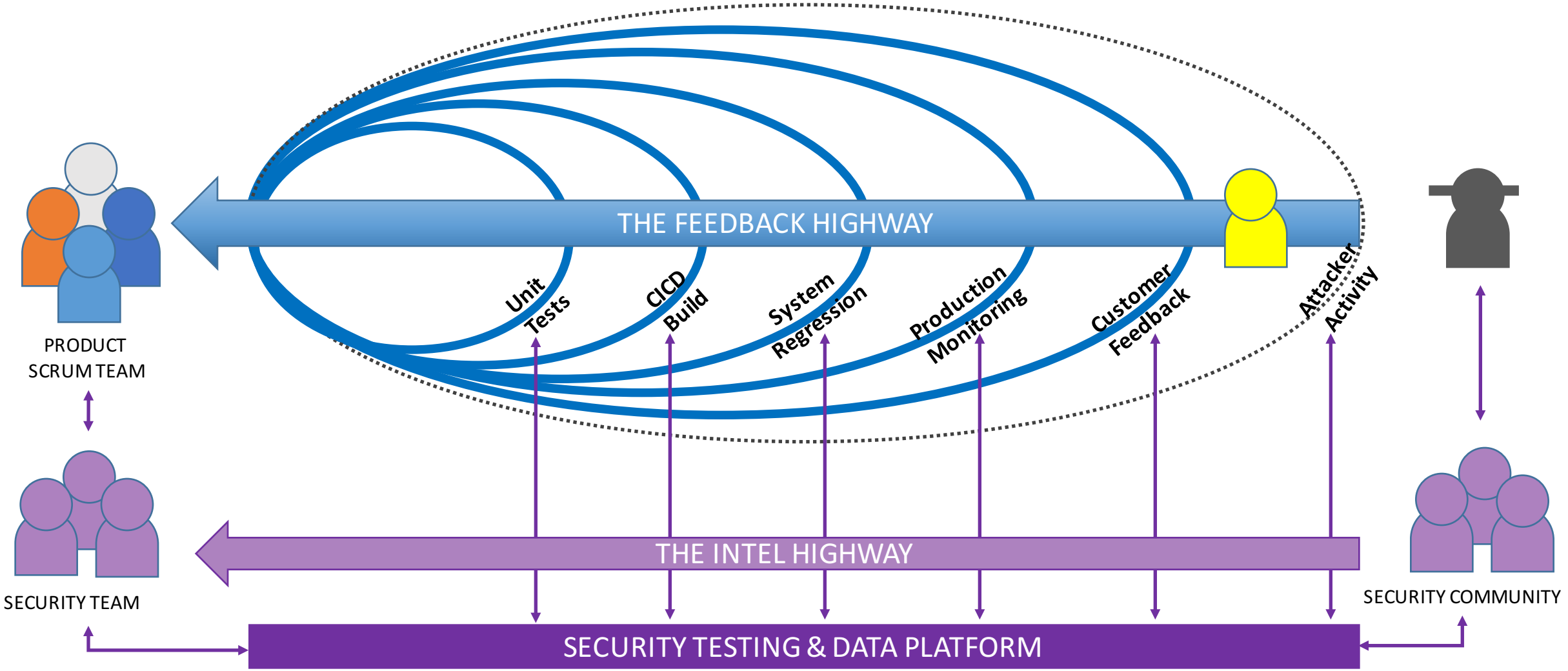
**Mixed Threats**

21. Trusted operator departure

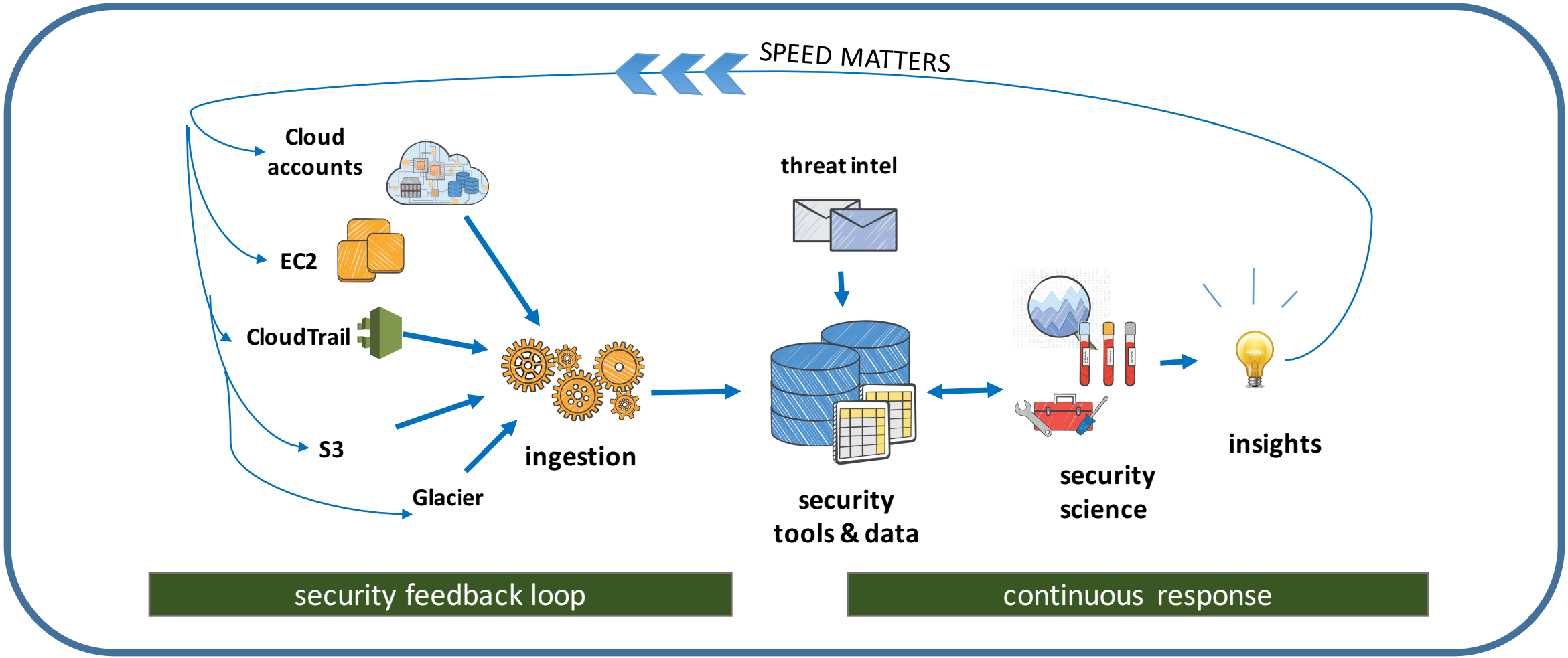
**AWS Threats**

8. Denial of Service
9. AWS IAM (app) user has more than one AWS API access key
10. EC2 host compromised
11. IAM account and bucket policy error
12. Malicious modification or delete of objects
13. Many Key Encryption Keys compromised during key rotation
14. Unexpected AWS IAM role on account
15. Access to physical media
16. Compromise of root
17. S3 object retrieved from an unauthorized IP address
18. Unexpected AWS IAM user on account
19. Untrusted employee departure
20. AWS encryption keys compromised

# The Intel Highway



# Intel Gathering Platform




Monitor & Inspect Everything

# Crawl, Walk, Run

- **Crawl** - Identifying security design constraints and controls that need to be built into the software to reduce successful attack
- **Walk** - Prioritize and build security into for issues found later in the software lifecycle
- **Run** - Build automation into script deployment to detect issues, unit testing, security testing , black box testing

# Iterative Security Controls

| Crawl | Walk | Run |
| --- | --- | --- |
| Authenticate Users to ensure authorized access to online application | Ensure each user has a role and gets assigned according to functional role | Ensure that user authentication and roles are behaviorally consistent functionally, identify anomalies and heal anti-patterns |