Xebia

Fixing past mistakes

A story on DevSecOps

Dave van Stein | PVIB April 2024

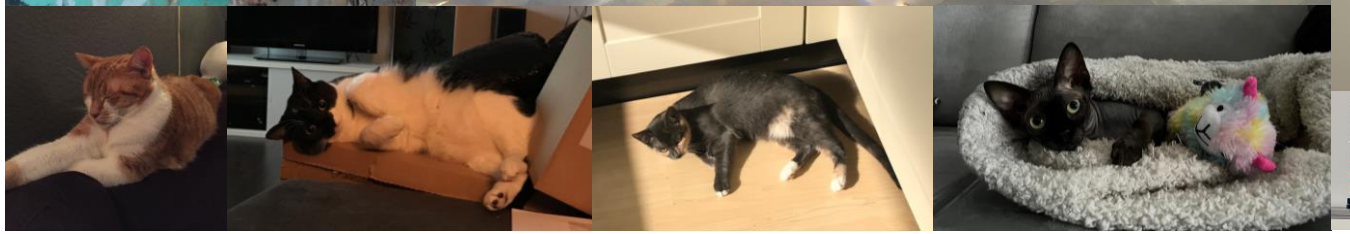
Xebia

Dave.vanstein@xebia.com

@Dave_von_S@infosec.exchange

nl.linkedin.com/in/dvstein

github.com/davevs



CSSLP

CIPP E iapp

CIPT iapp

GRC Professional

GRC Auditor

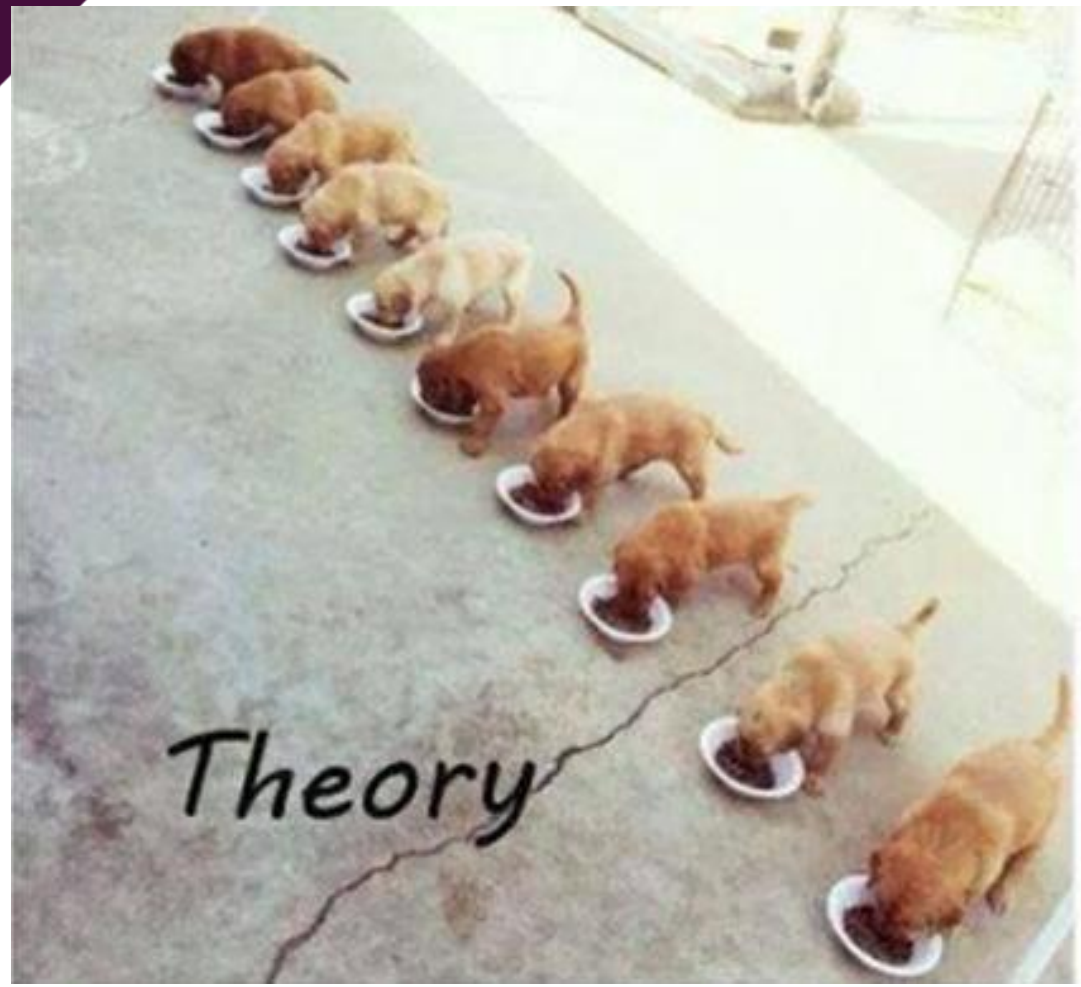
Integrated Audit & Assurance Professional

DASA DEVOPS AGILE SKILLS ASSOCIATION

6σ BLACK BELT

Certified Facilitator LEGO® SERIOUS PLAY®

Disclaimer



A small trip on memory lane

1996: first CISO

The classic approach

Xebia

Security Fixed!

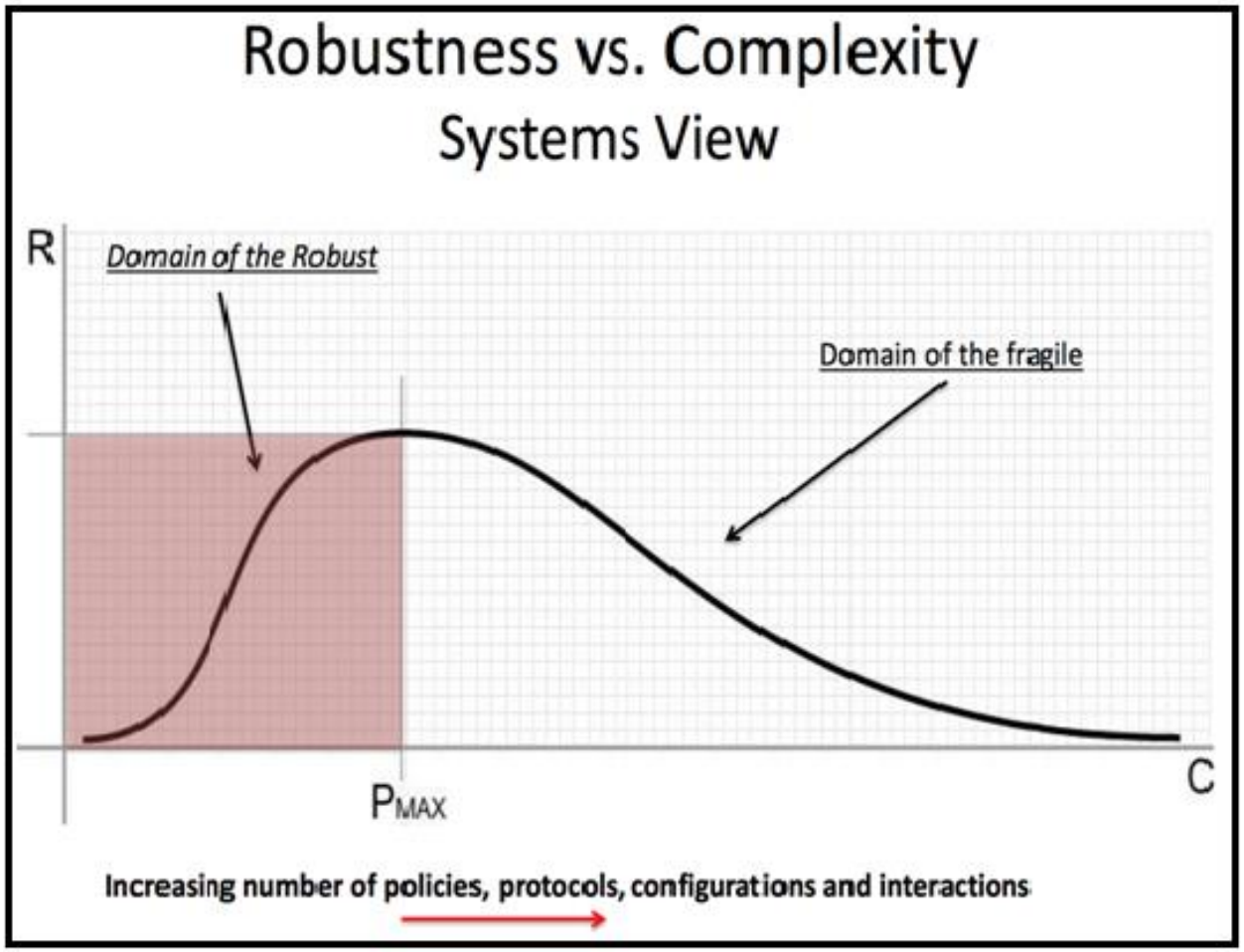
Xebia

Pentests at the end of sprints

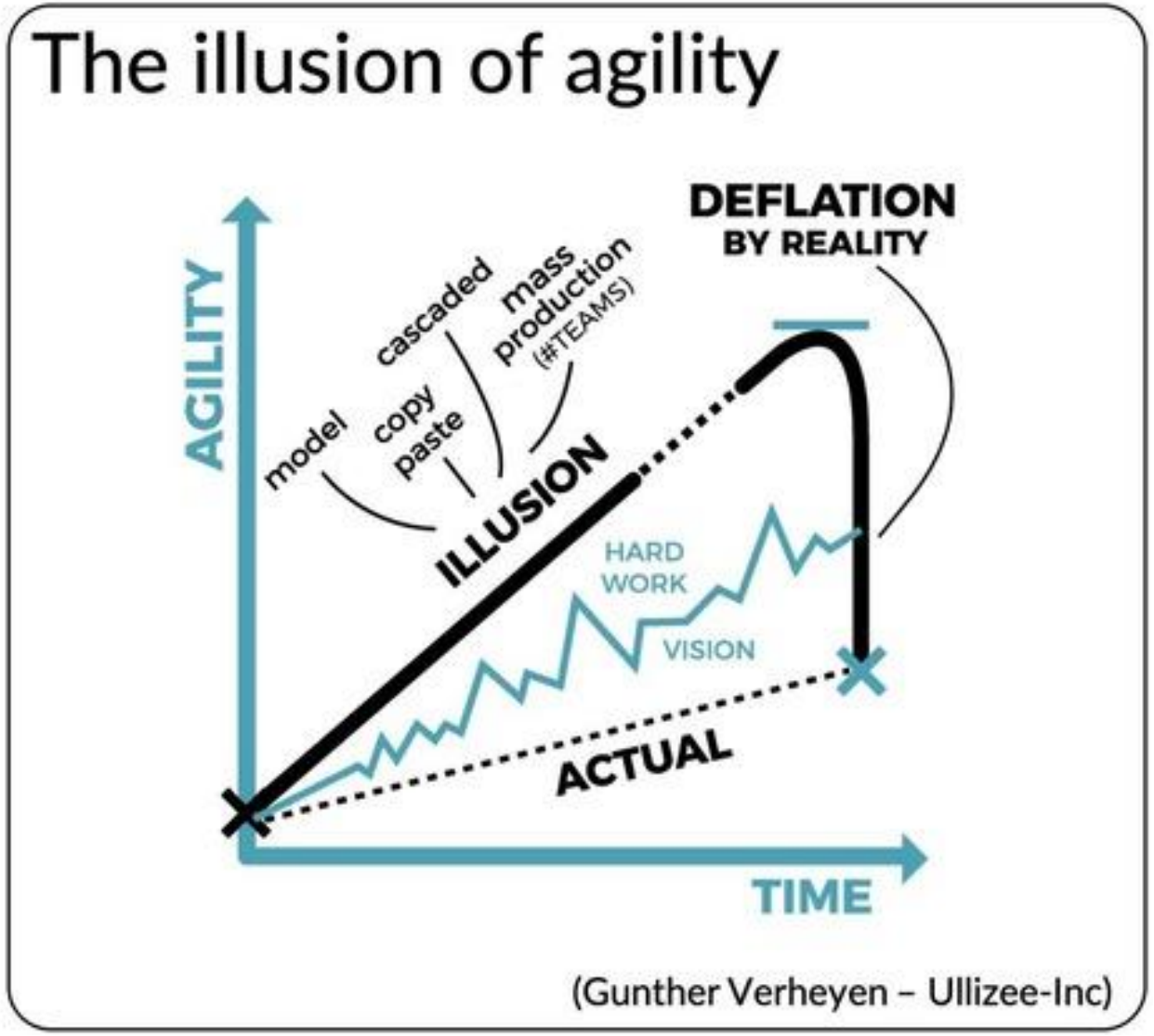
Mandatory policy checks

Development and Security: how we view ourselves

Why security is failing



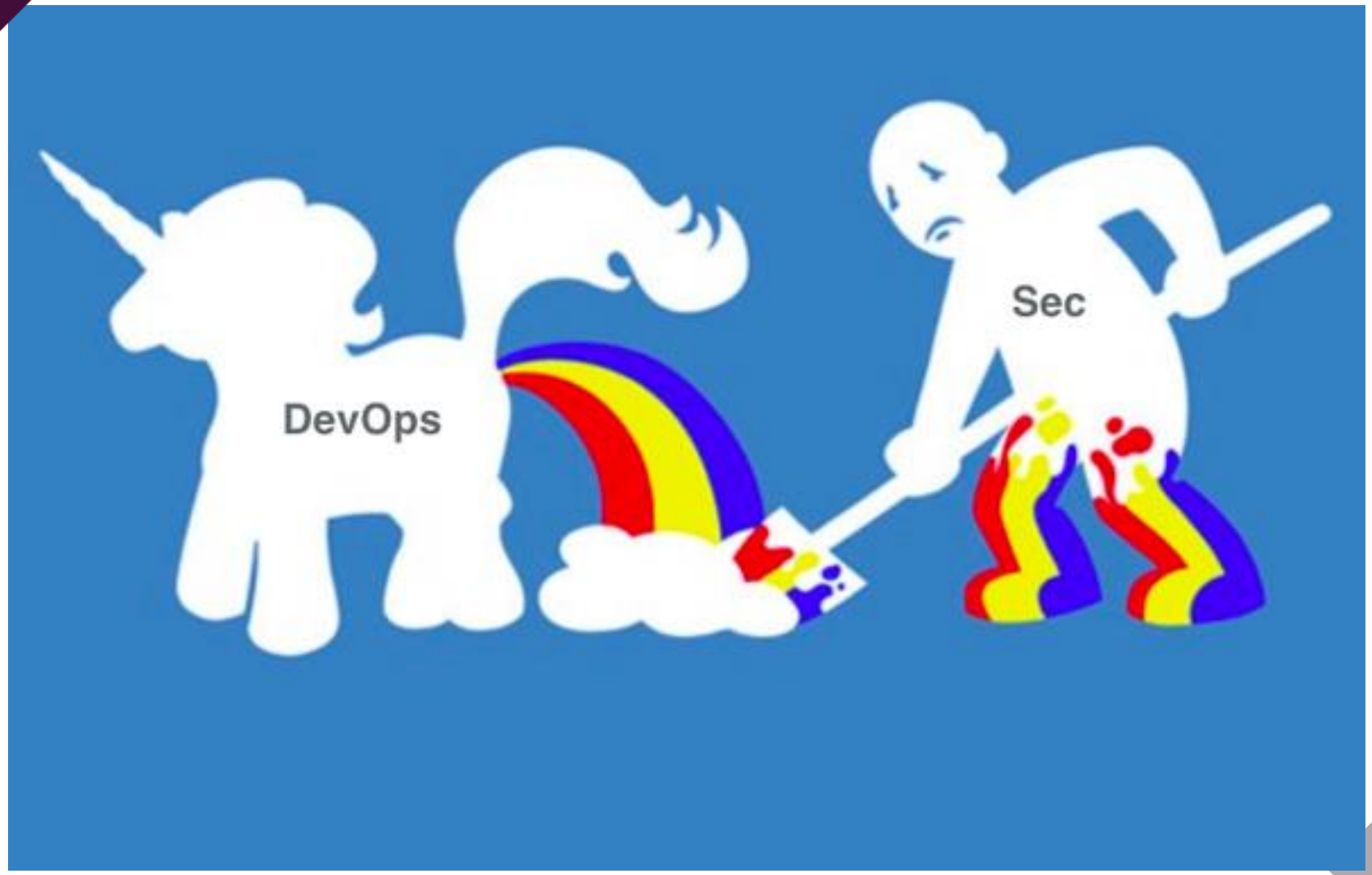
Why software development is failing



Development and Security: how we view the other

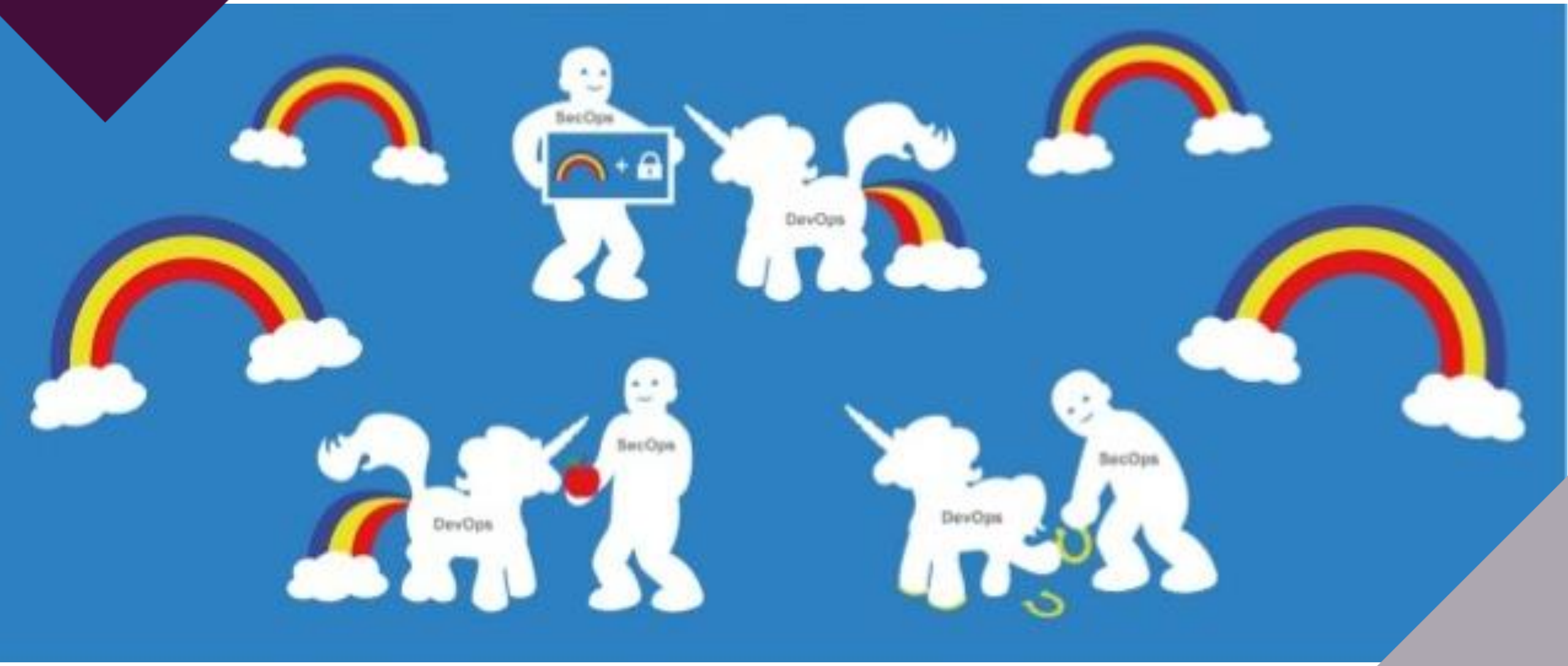
The Result

One way to do it



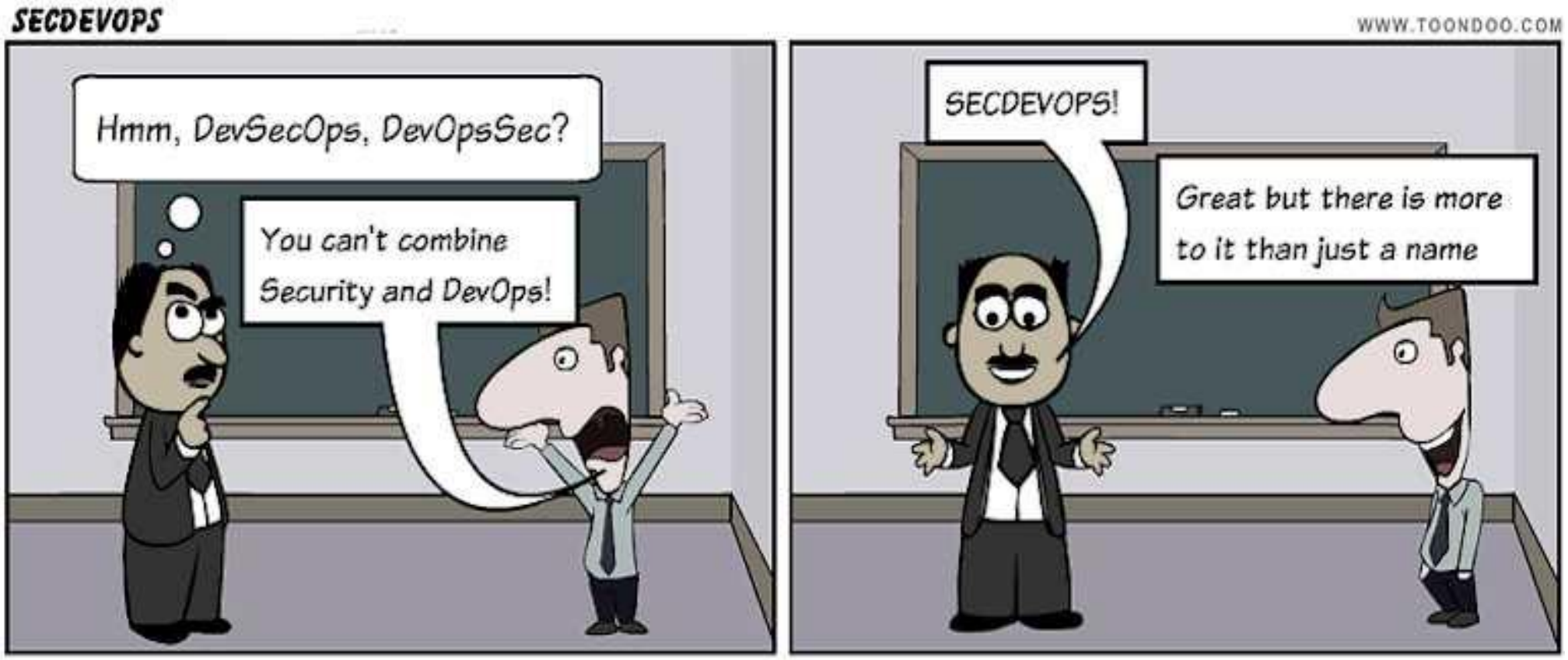
Xebia

A better idea



Xebia

DevSecOps



DevOps is not just putting everything in 1 team!

Giulio Carrara • 3rd+
Software Engineer
1d •

Dear recruiters,
if you are looking for:

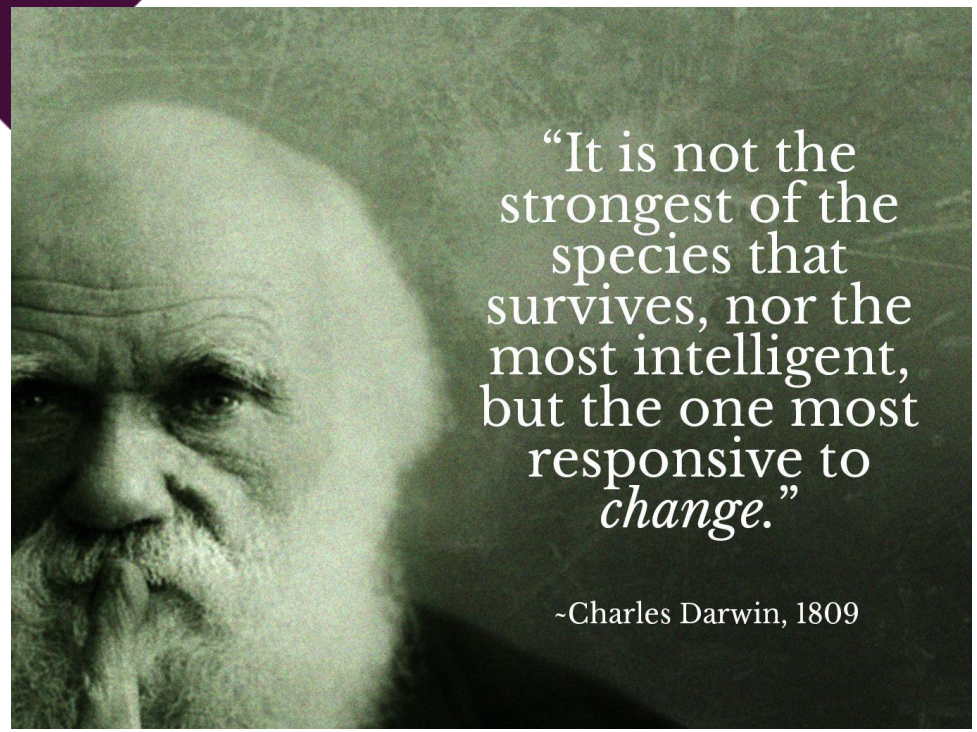
- Java, Python, PHP
- React, Angular
- PostgreSQL, Redis, MongoDB
- AWS, S3, EC2, ECS, EKS
- *nix system administration
- Git and CI with TDD
- Docker, Kubernetes

That's not a Full Stack Developer.
That's an entire IT department.

Yours truly

8,801 391 comments

Security & Agile?



Why Security?

Xebia

100% secure?

A change from this

To this

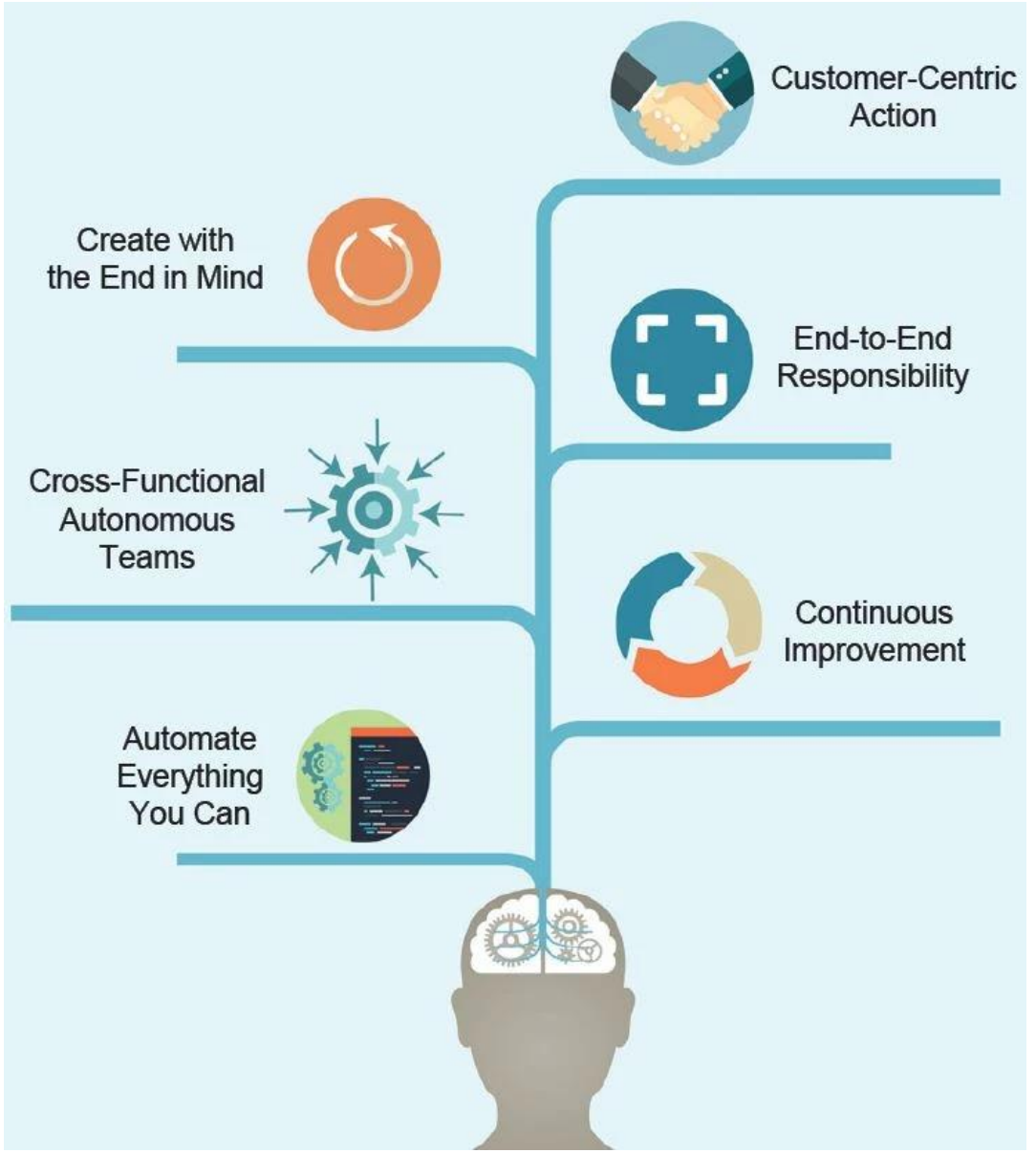

30
km/h
TX-95-FN


A8
Coenplein
S116
S103
S106
info A10
A4
Amstel
A1

While preventing this

The real goal

DevOps Core Principles



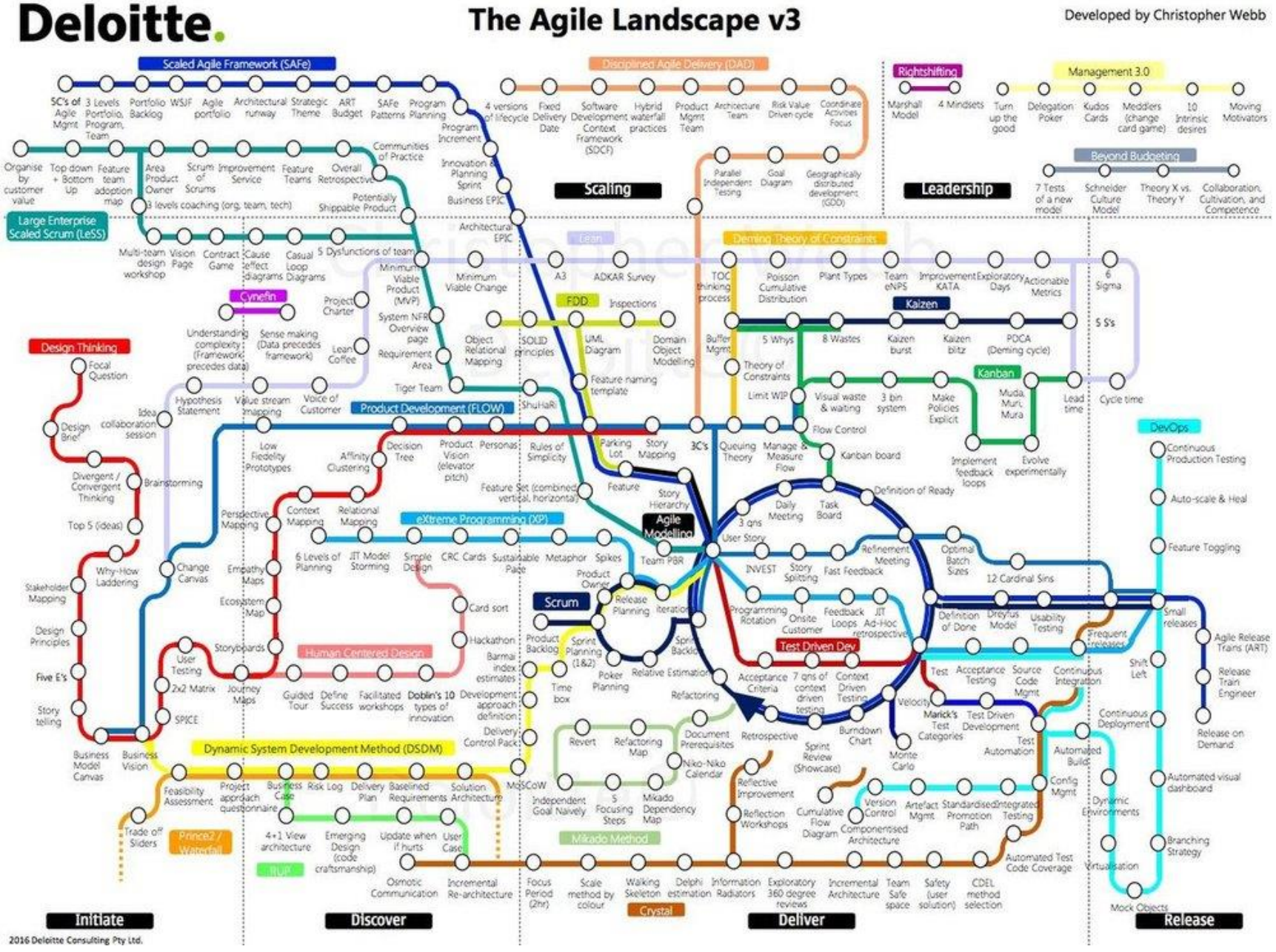

SLOW
DOWN
BUMP
AHEAD
SLOW
DOWN
BUMP
AHEAD


SLOW
DOWN
BUMP
AHEAD

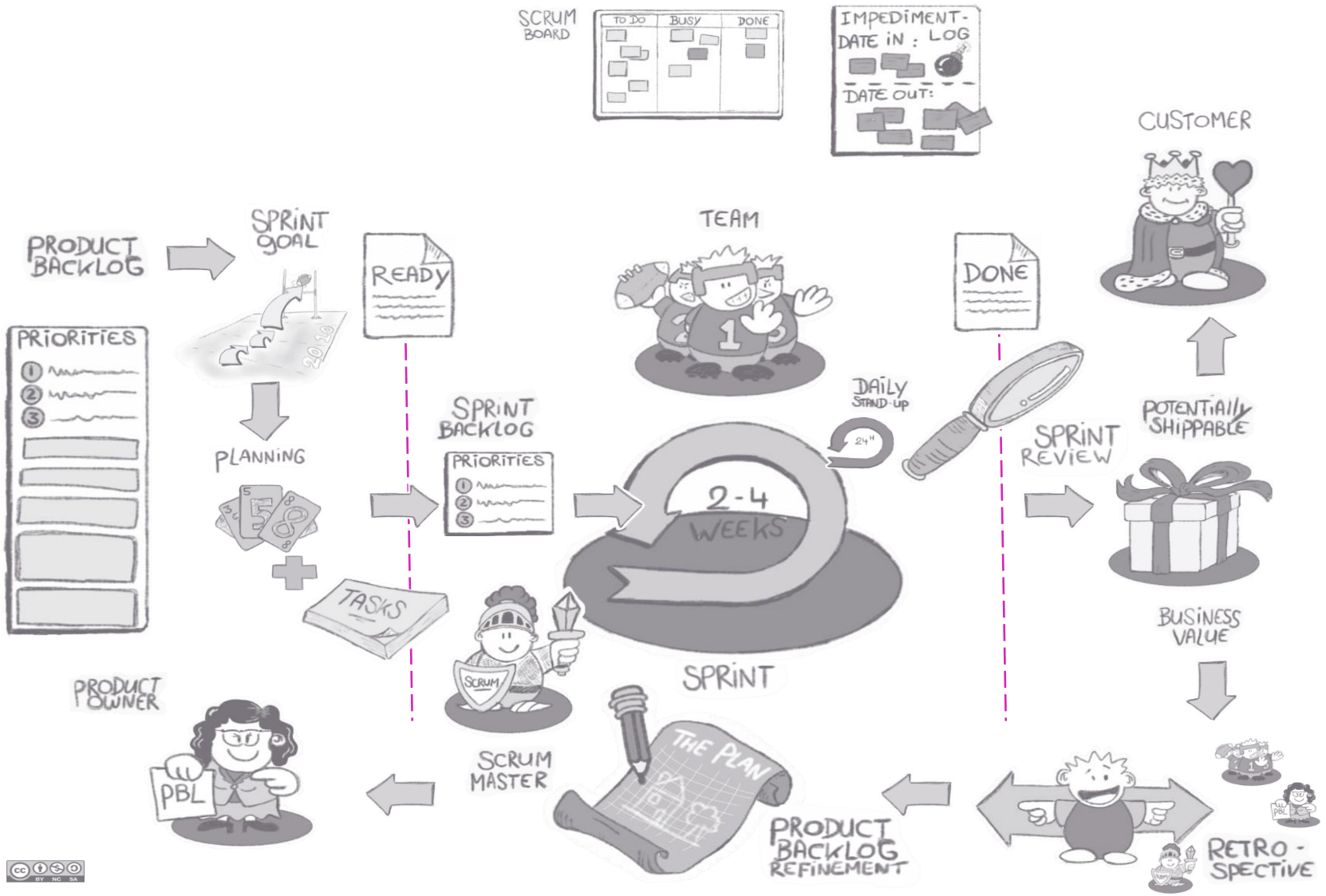
Xebia

DevSecOps

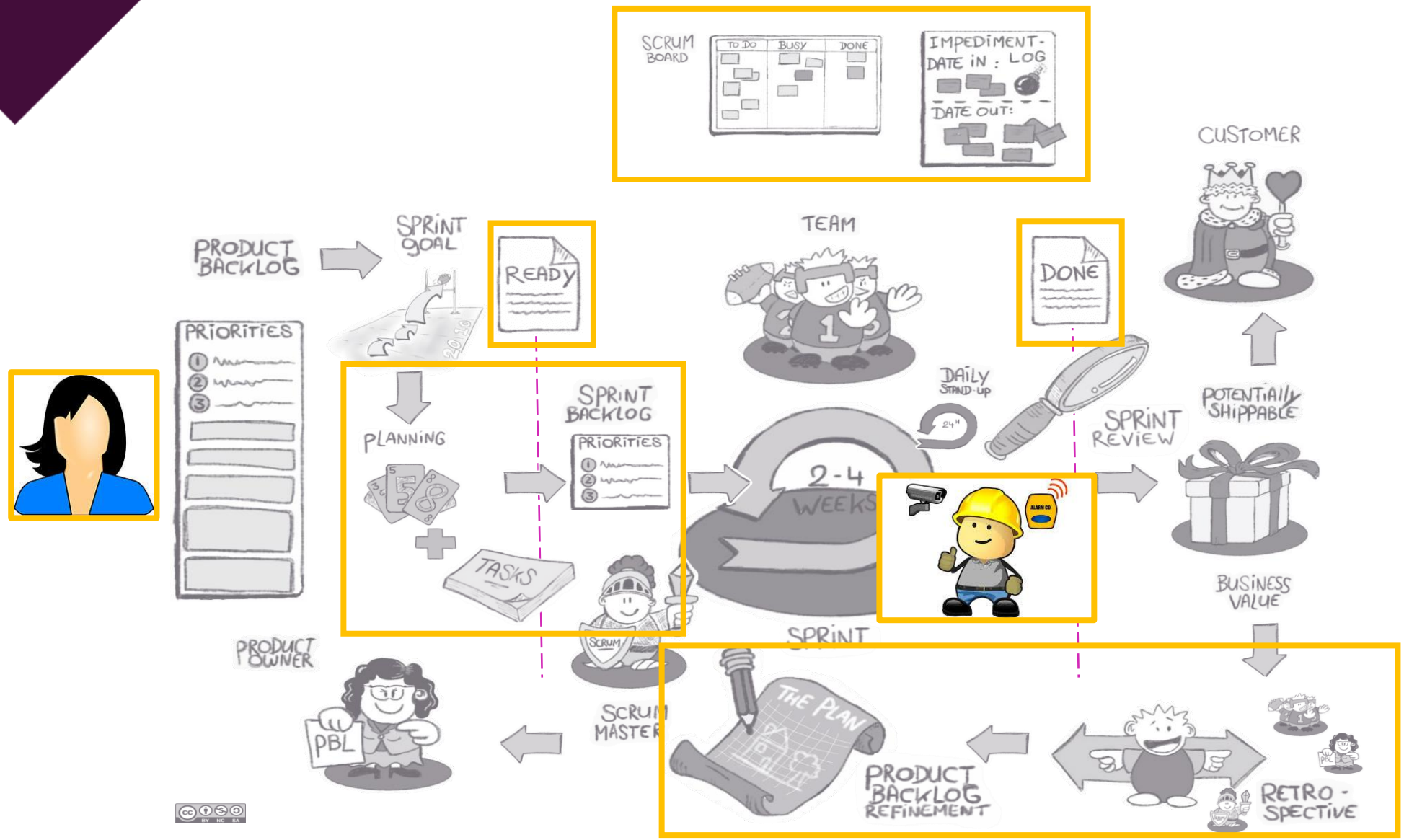
Agile? DevOps? DevSecOps?



Agile: the important bits



Agile Security

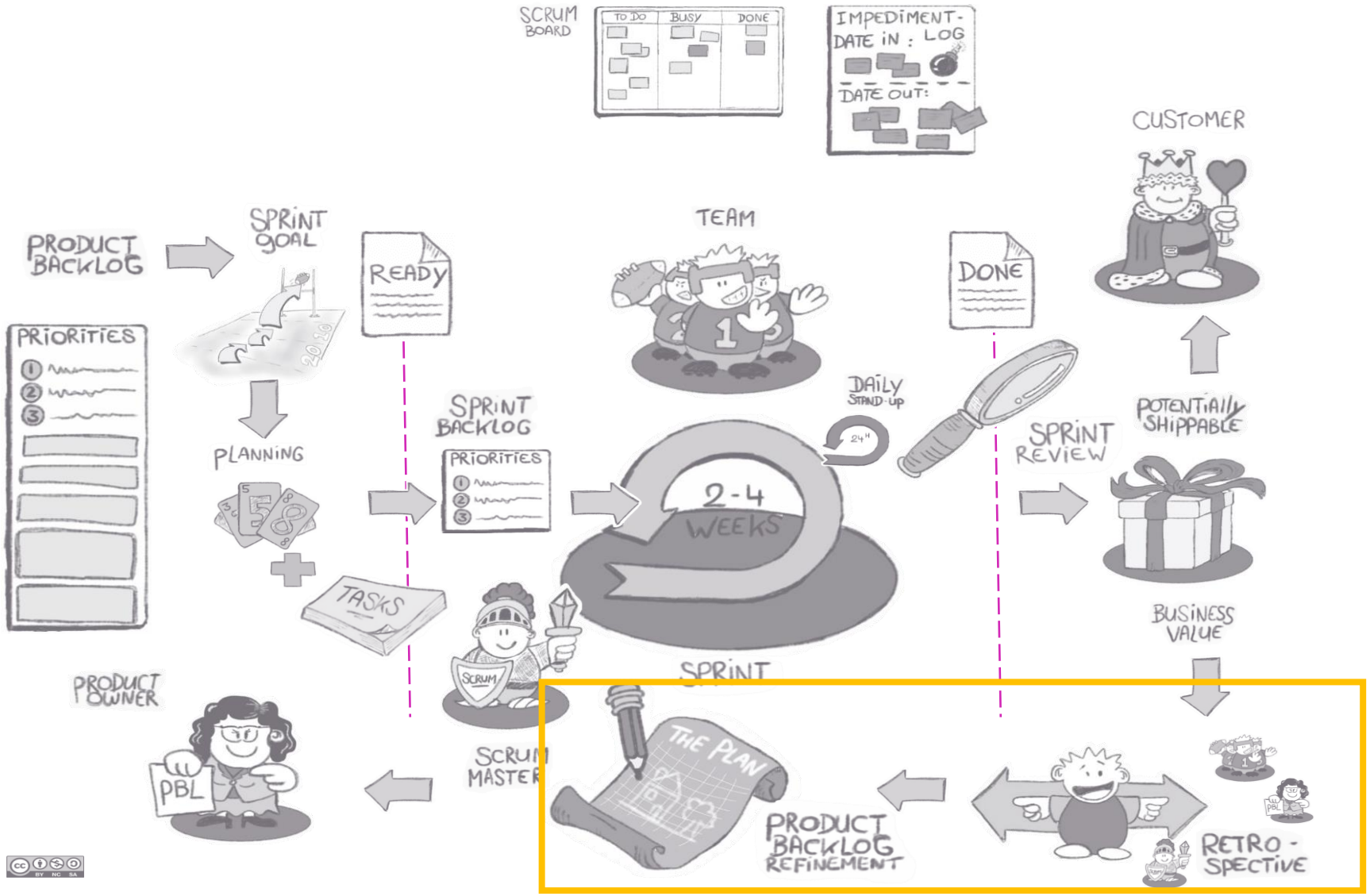


Xebia

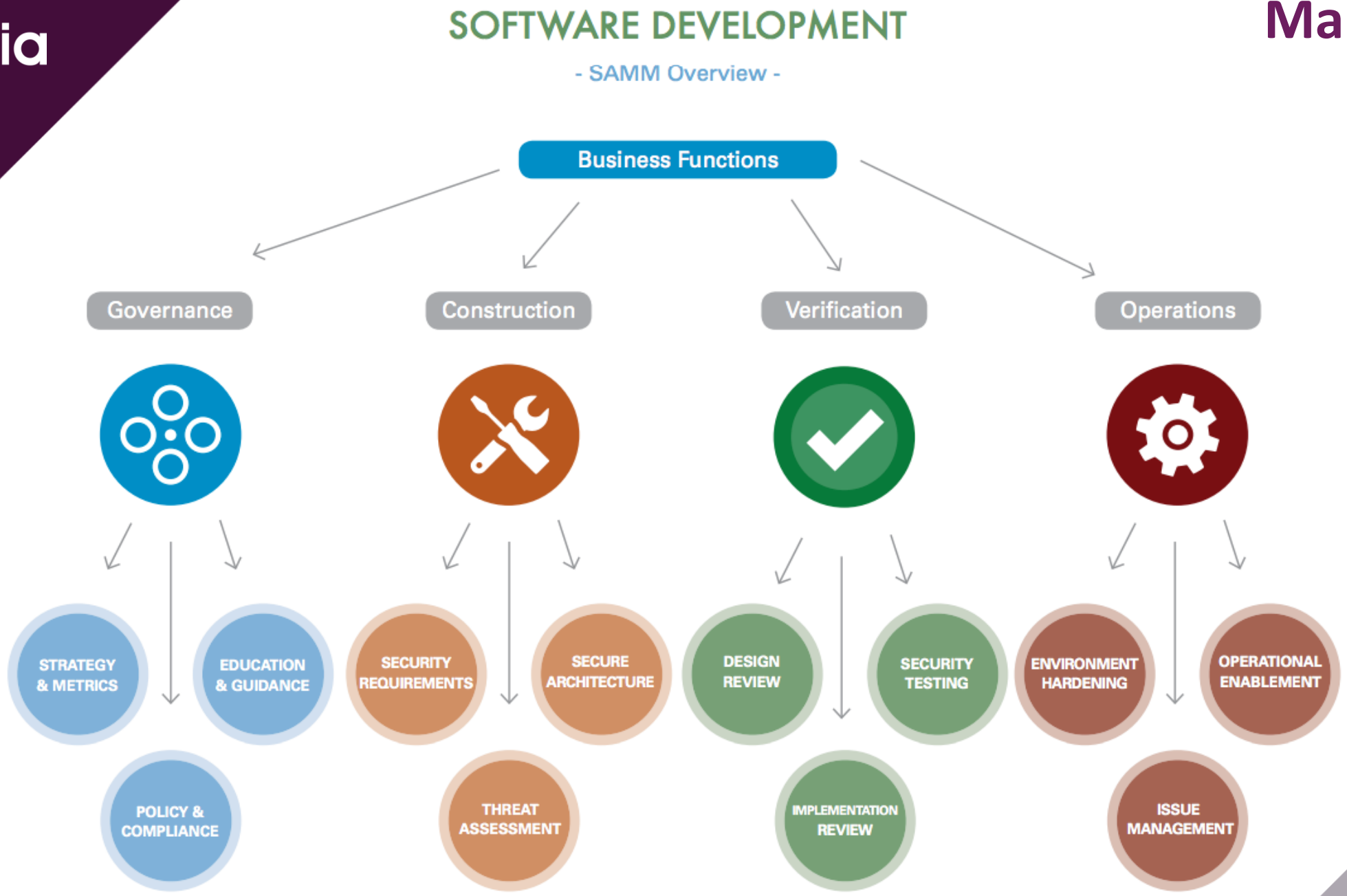
How is Secure Agile Development Different?

Traditional / Waterfall		Agile
• Distinct security-focused project phases, often at beginning and end of project.	Security Timing	• Every iteration considers security, but is not limited by it.
• Security skills brought in from outside project, often disconnected from dev/test resources.	Security Resources	• Every team member is responsible for security. Security skills are embedded in the team.
• Specific security testing phase, often at end of project.	Security Validation	• Hybrid security and functionality testing, throughout project.

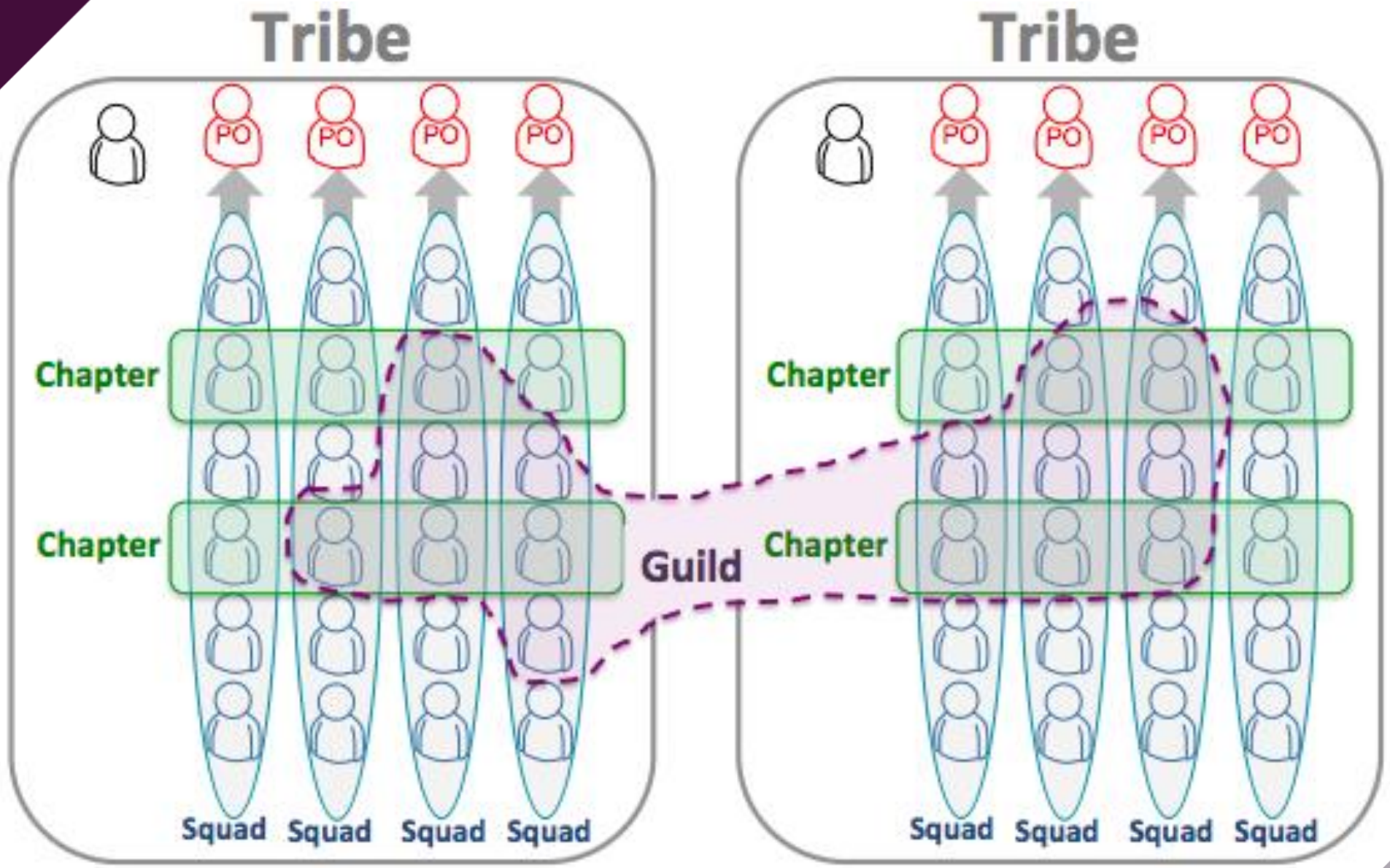
Security Education



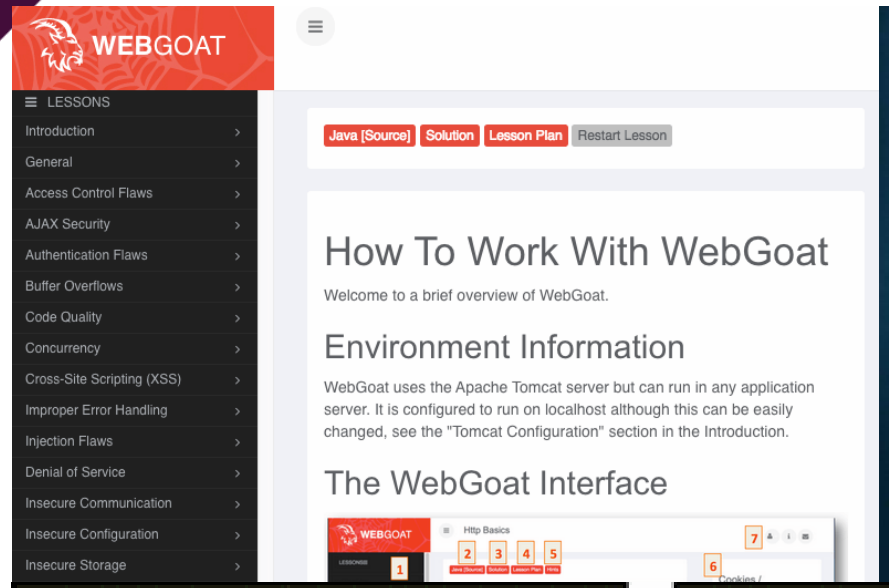
Make a plan

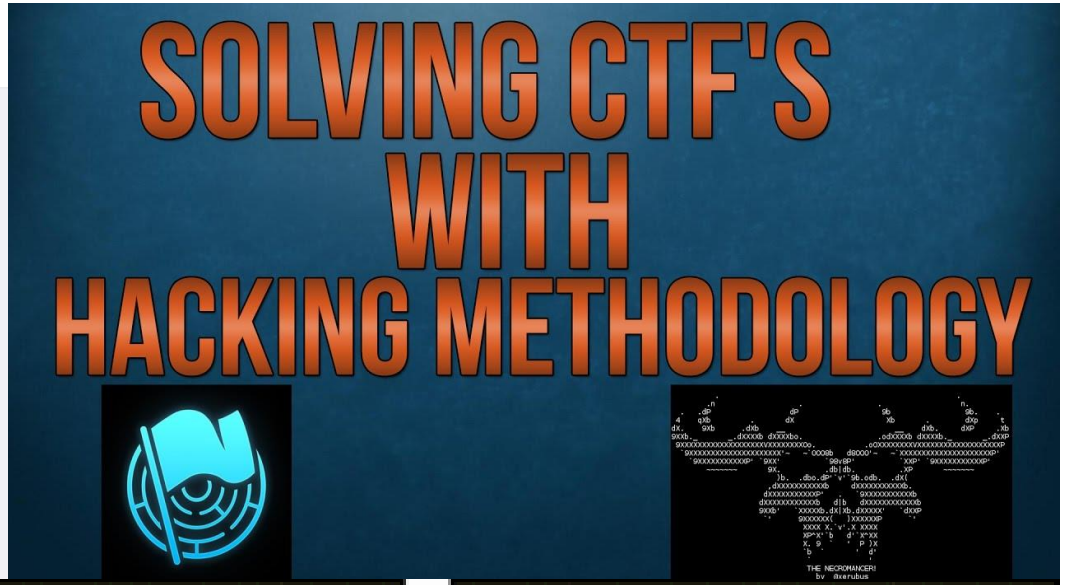

SOFTWARE DEVELOPMENT
- SAMM Overview -
Business Functions
Governance
Construction
Verification
Operations
STRATEGY & METRICS
EDUCATION & GUIDANCE
POLICY & COMPLIANCE
SECURITY REQUIREMENTS
SECURE ARCHITECTURE
THREAT ASSESSMENT
DESIGN REVIEW
SECURITY TESTING
IMPLEMENTATION REVIEW
ENVIRONMENT HARDENING
OPERATIONAL ENABLEMENT
ISSUE MANAGEMENT

Start a Security Guild

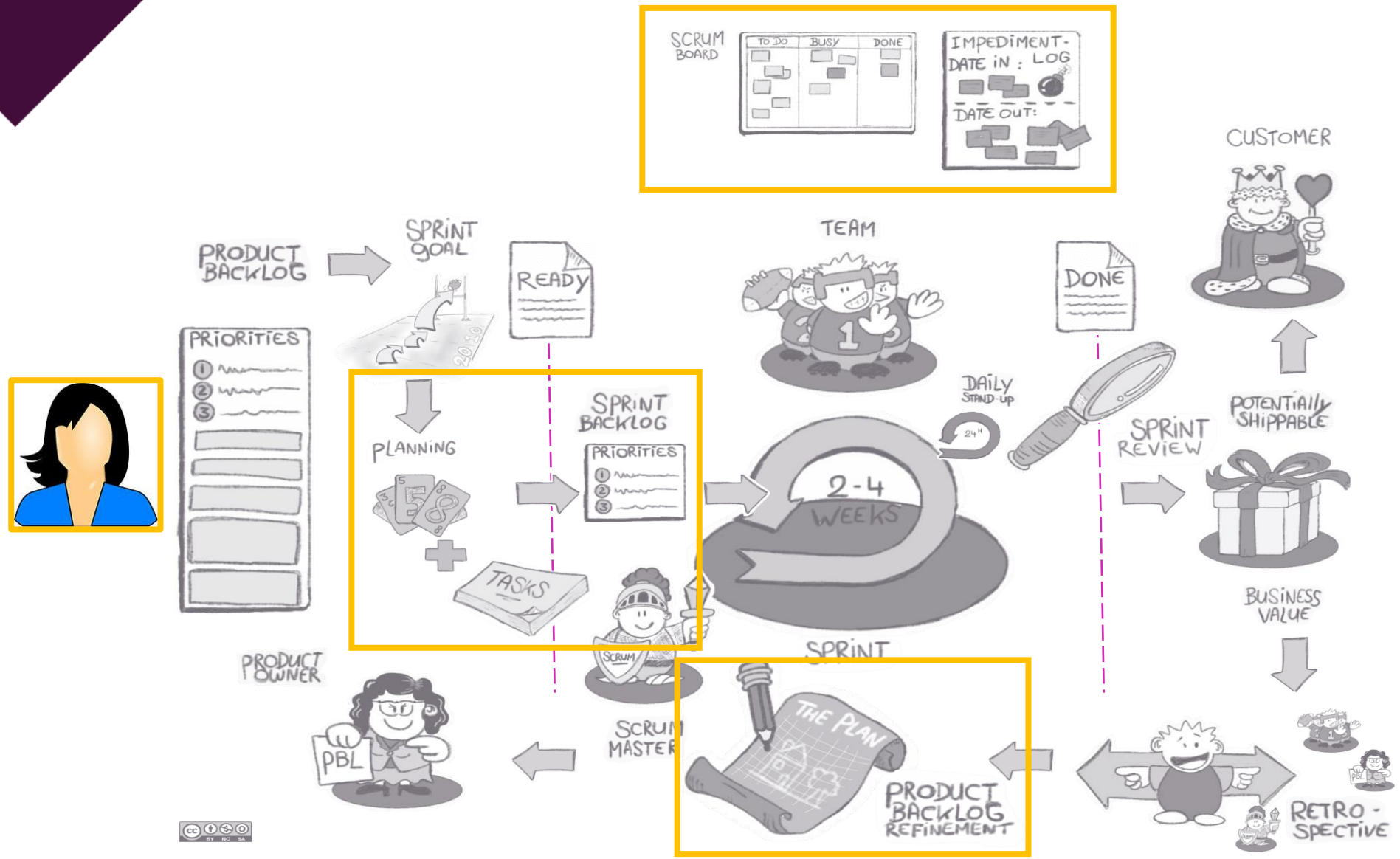


Start hacking yourself

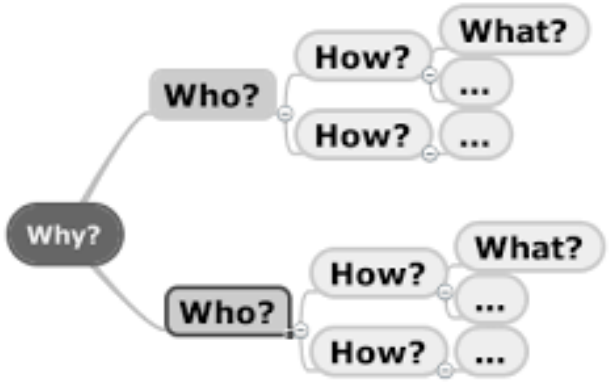




Security Stakeholdership

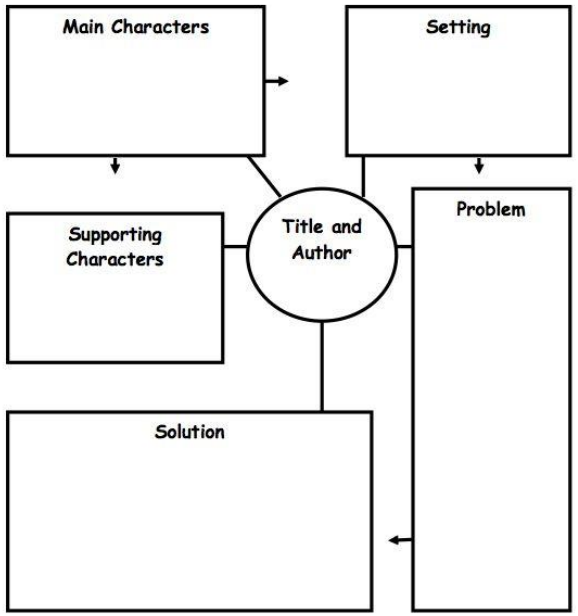


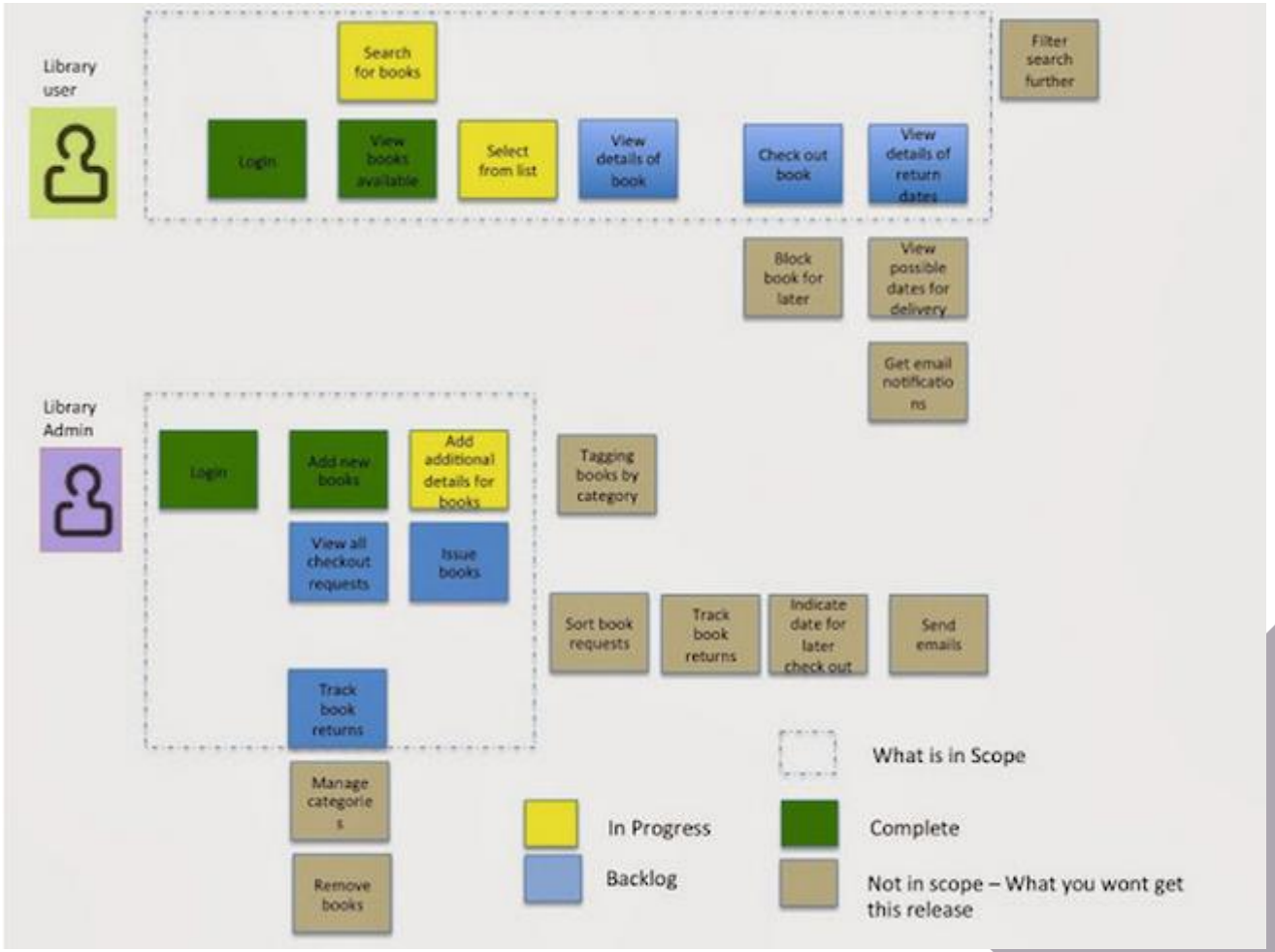
Story mapping



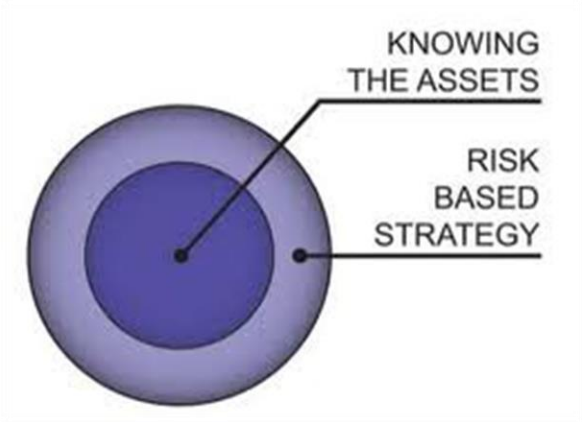
Name _____ Date _____

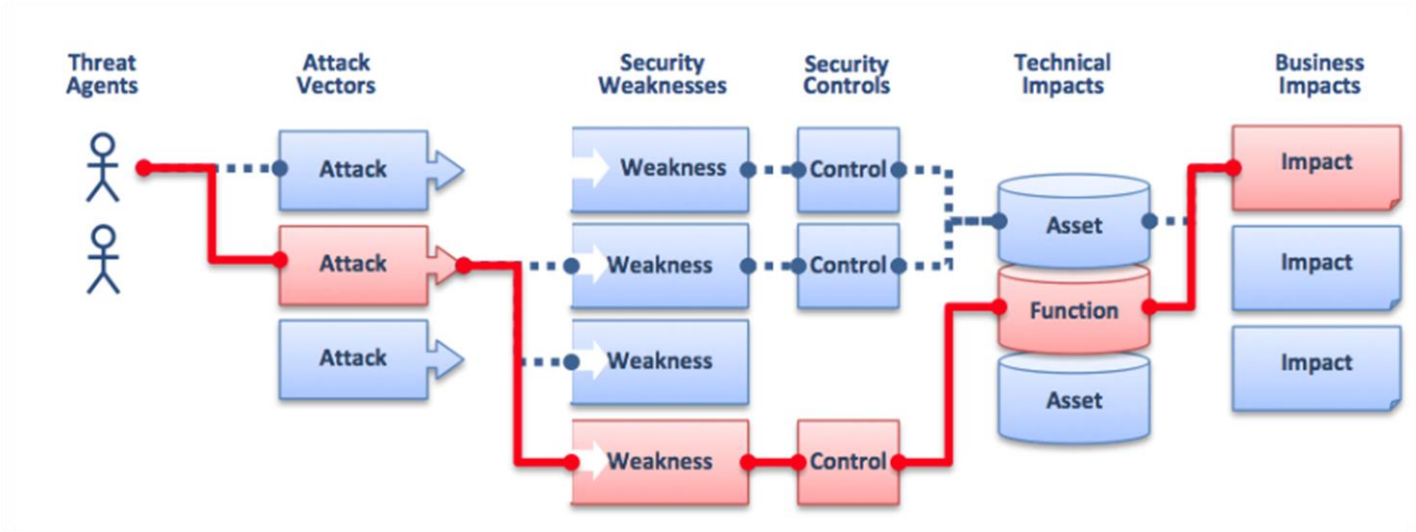
STORY MAP



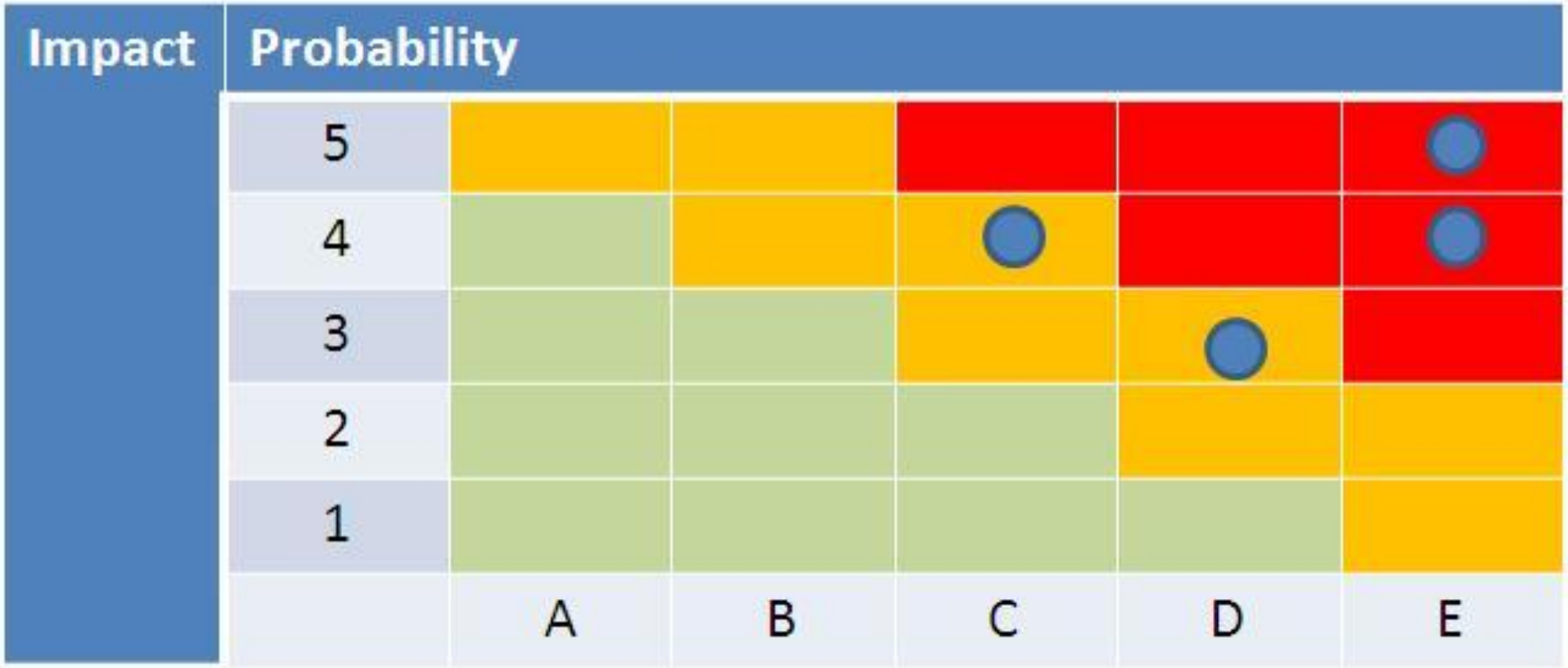


Threat model the story map





Agile Risk Self Assessment



Adding Security User Stories

User Story

As a customer, **I want to** track the shipment of my order **so that** I know when it will arrive.

Security Story

As a fraudster, **I want to** see the details of an order that is not my own **so that** I can learn another person's private information.

"As an employee, I can search for other employees by their last name."

User

"As a hacker, I can send bad data in the content of requests."

"Hee hee.."

Evil User

198 Security

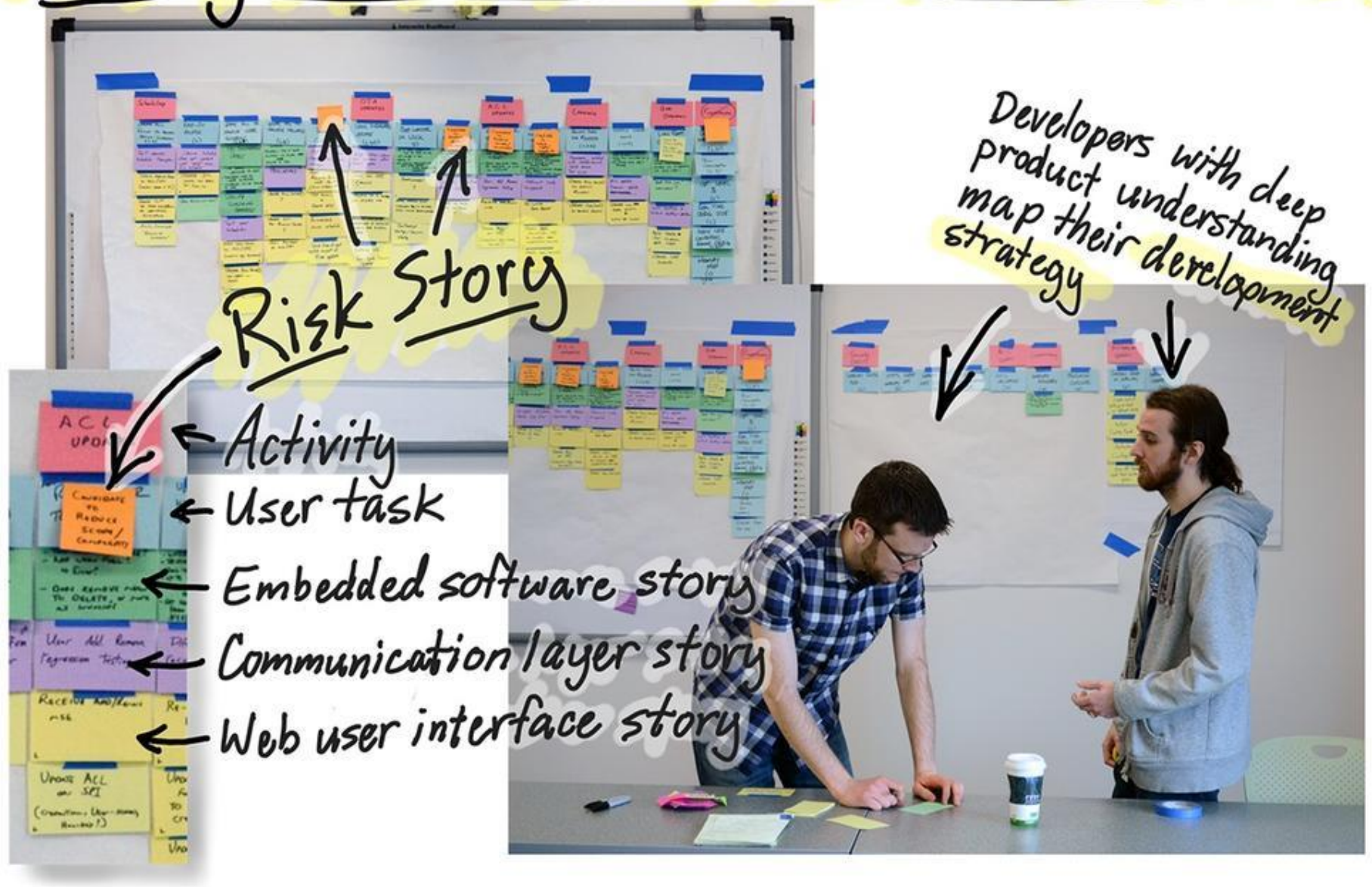
Implement Security for User Information

198

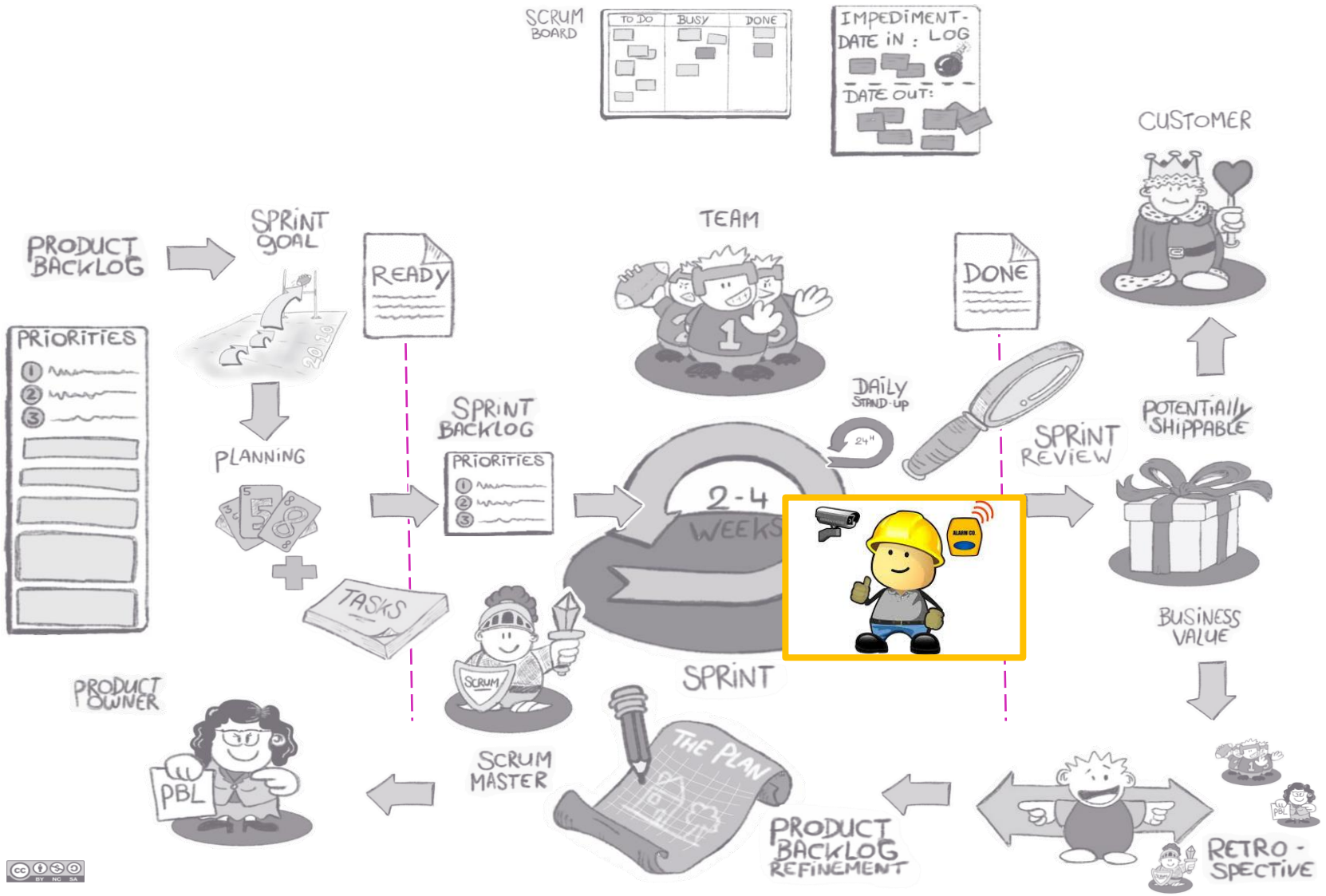
Steal Credit Card Info

As a Malicious Hacker I want to steal credit card information so that I can make fraudulent charges

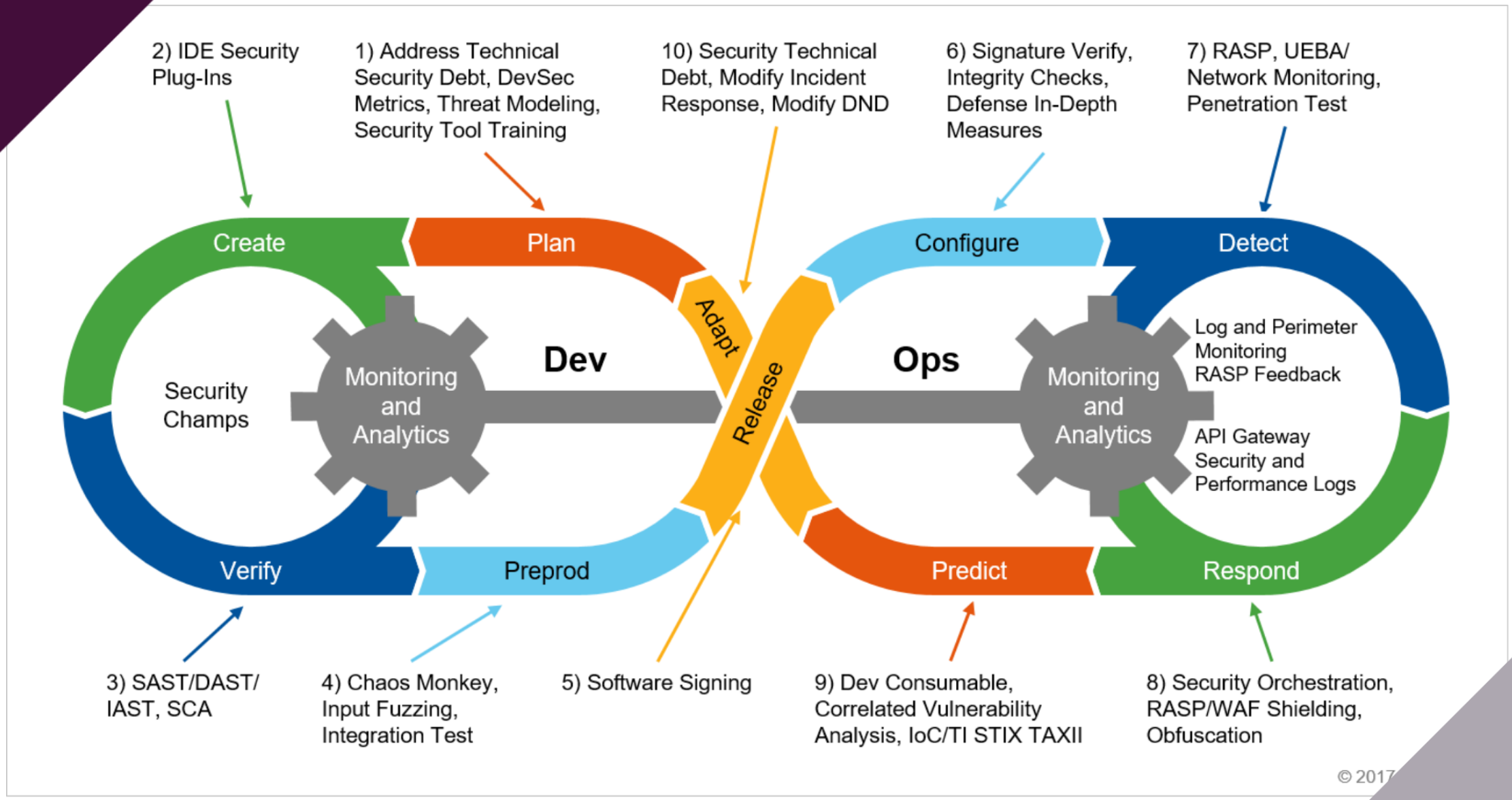
Adding risk stories to make risk visible



Security Engineering



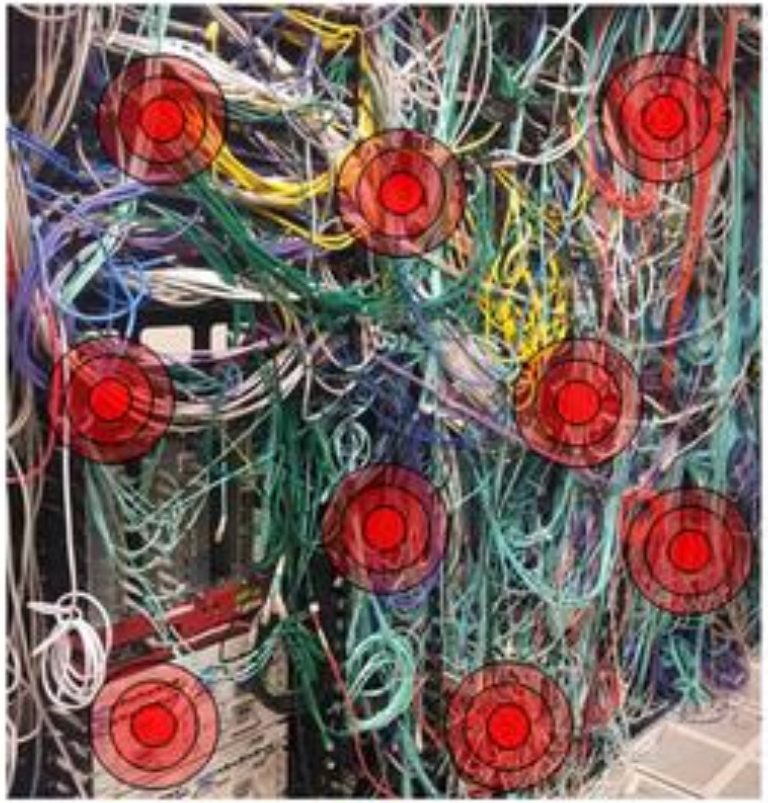
Automate everything



Why patching is hard

Clean code

Legacy code



Locations in the code base that require modification for a new feature

Time to implement a new feature

Probability of breaking existing functionality

Classical deployments

OTAP in DevOps

Xebia

Patching 2.0

Xebia

Provide sandboxes for experiments

Set some groundrules

Recipe for a Safe Kitchen

Ingredients:

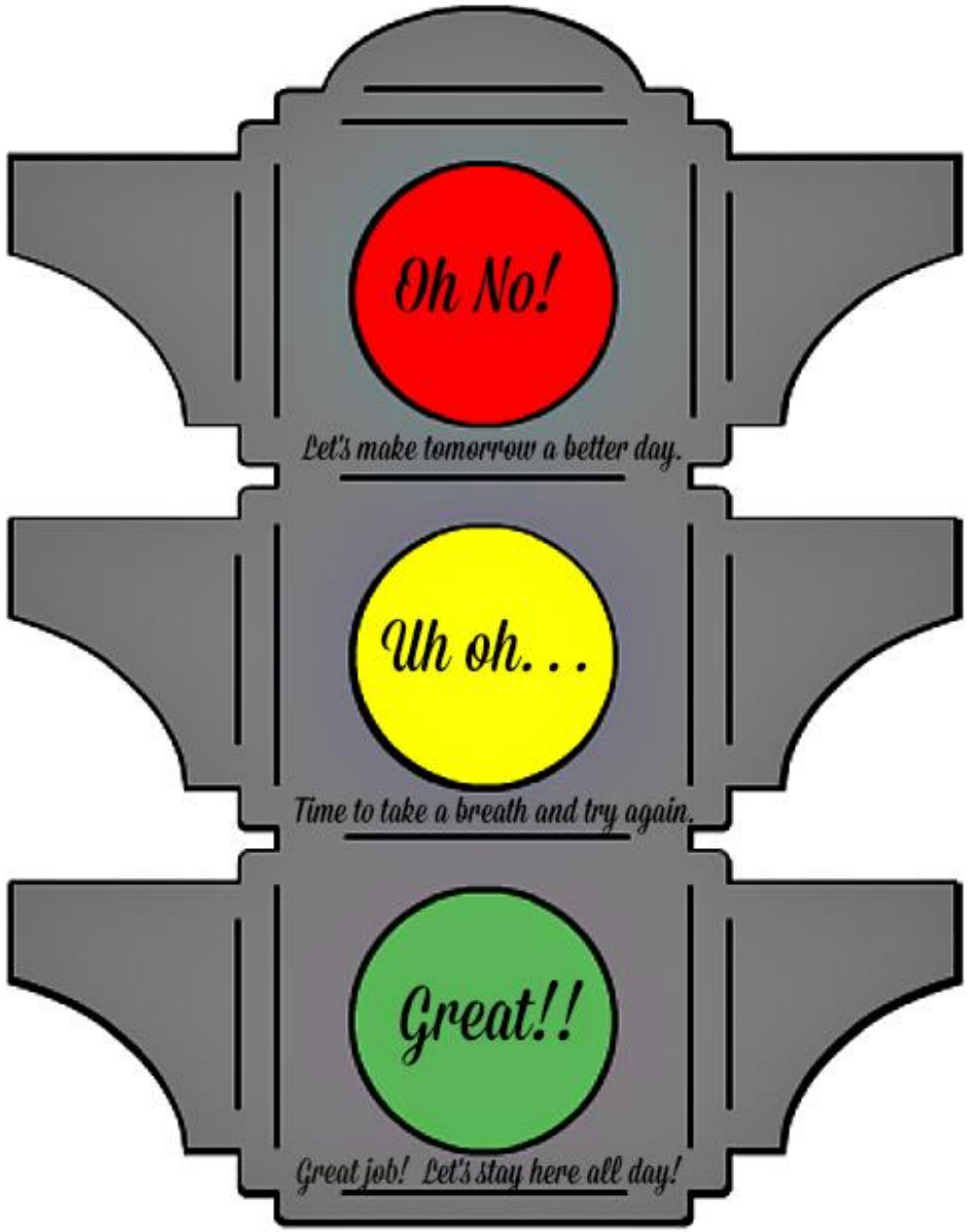
Prepare a “kid-free zone” of at least 3 feet (1 meter) around the stove.

Reduce chances of a fire. Keep anything that can catch fire away from stovetop.

Never dash out while cooking. Keep an eye on what you fry. Always cook with a lid beside your pan. If you have a fire, slide lid over pan and turn off burner.

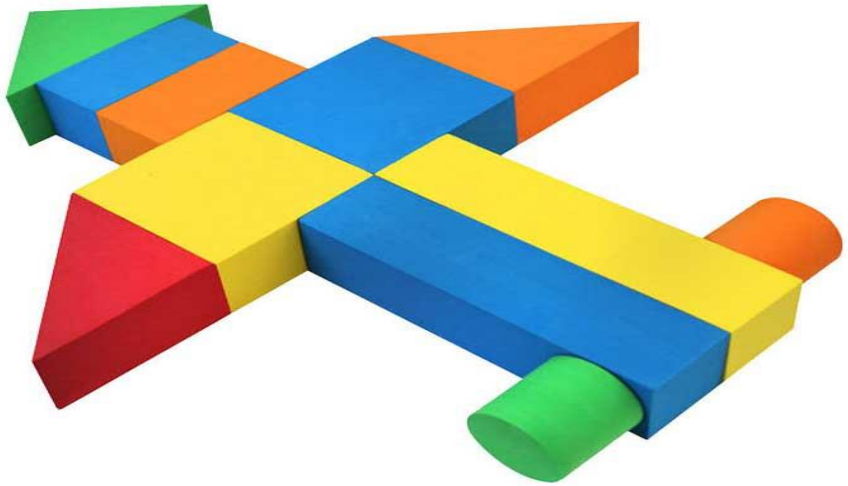
Prep your kitchen by having a working smoke alarm. Keep smoke alarms at least 10 feet (3 meters) from the stove to reduce false alarms.

Define thresholds

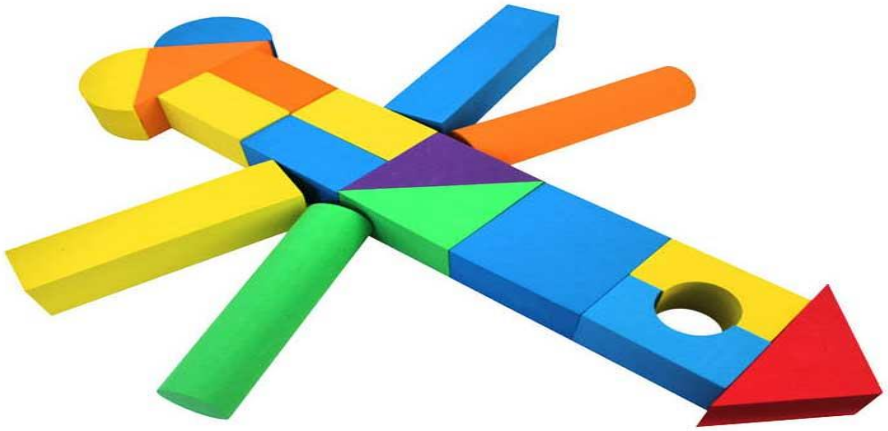


Set some groundrules

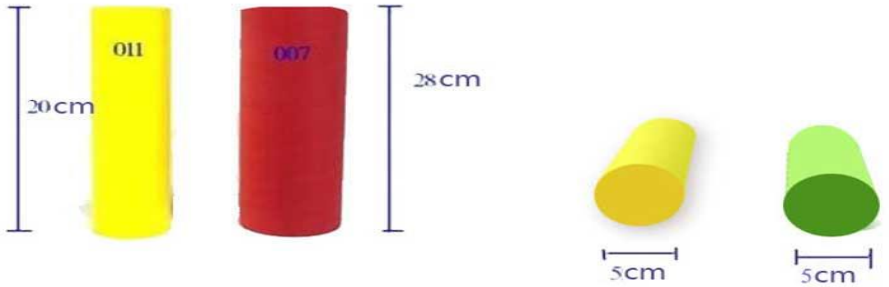
Plane



Dragonfly

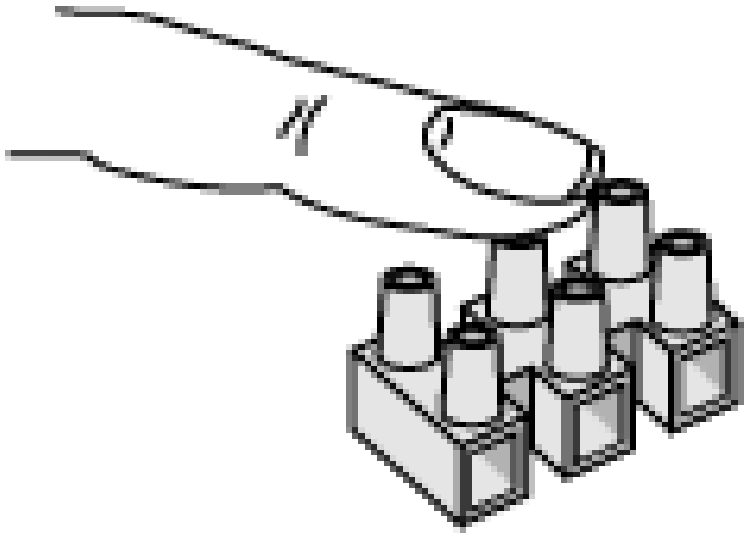


Gun

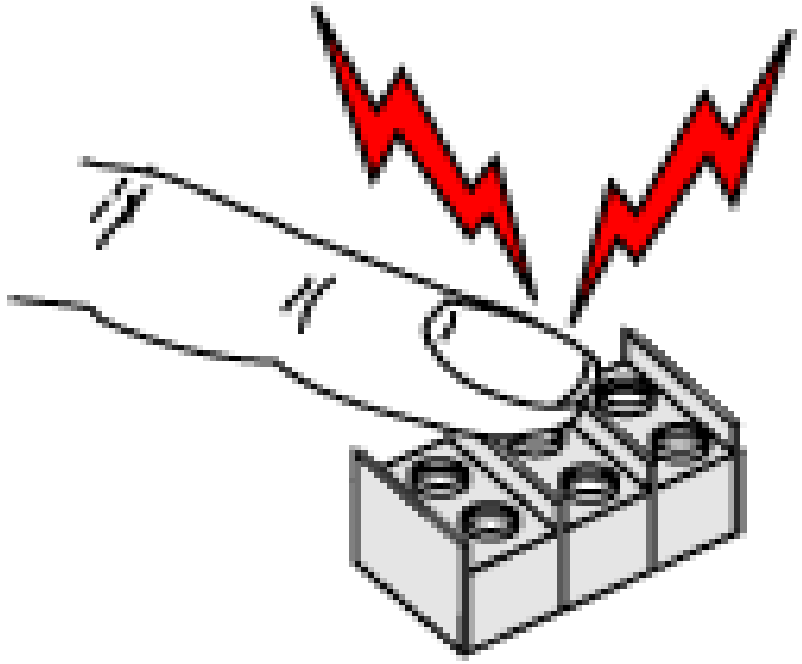


Assess new blocks

The International terminal block has recessed screws and terminals

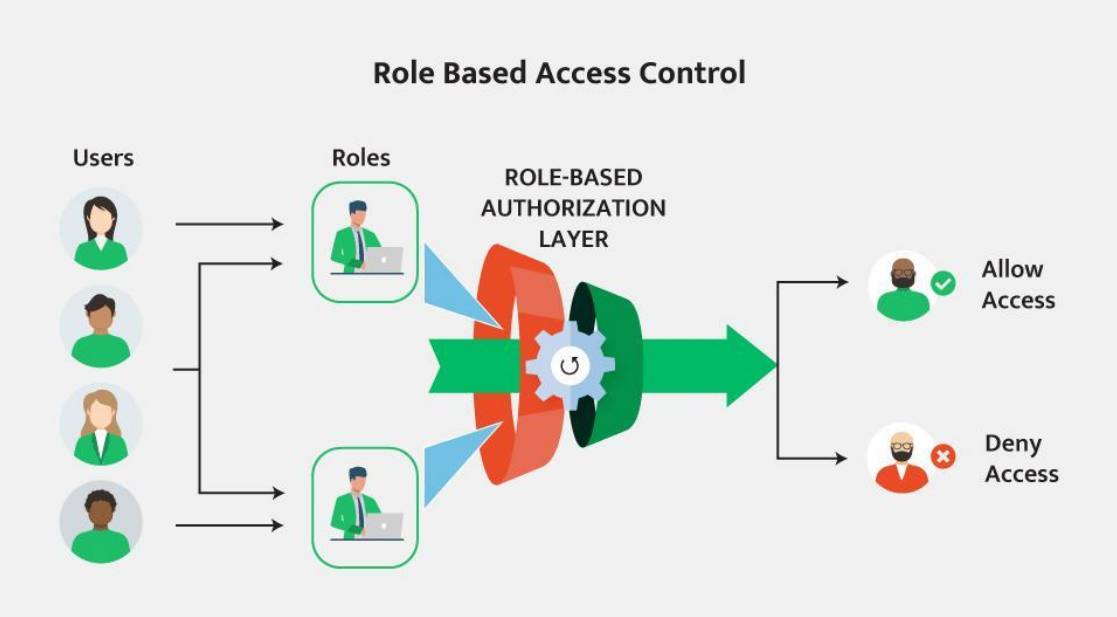


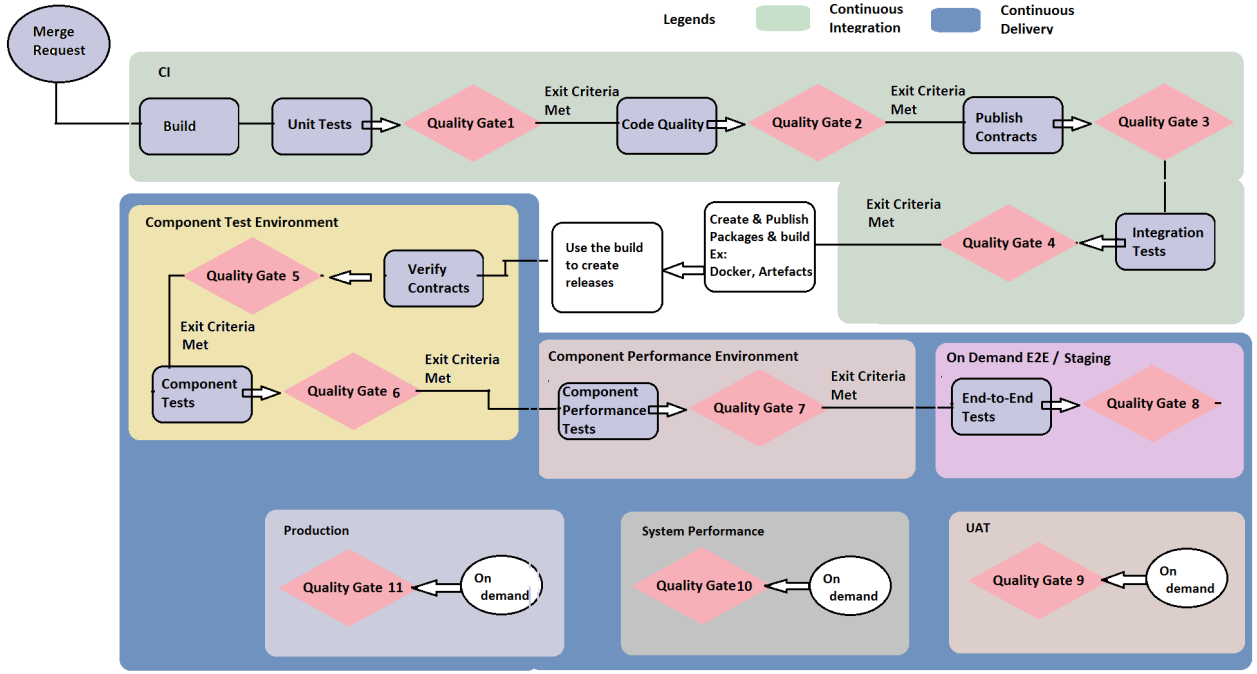
International terminal block



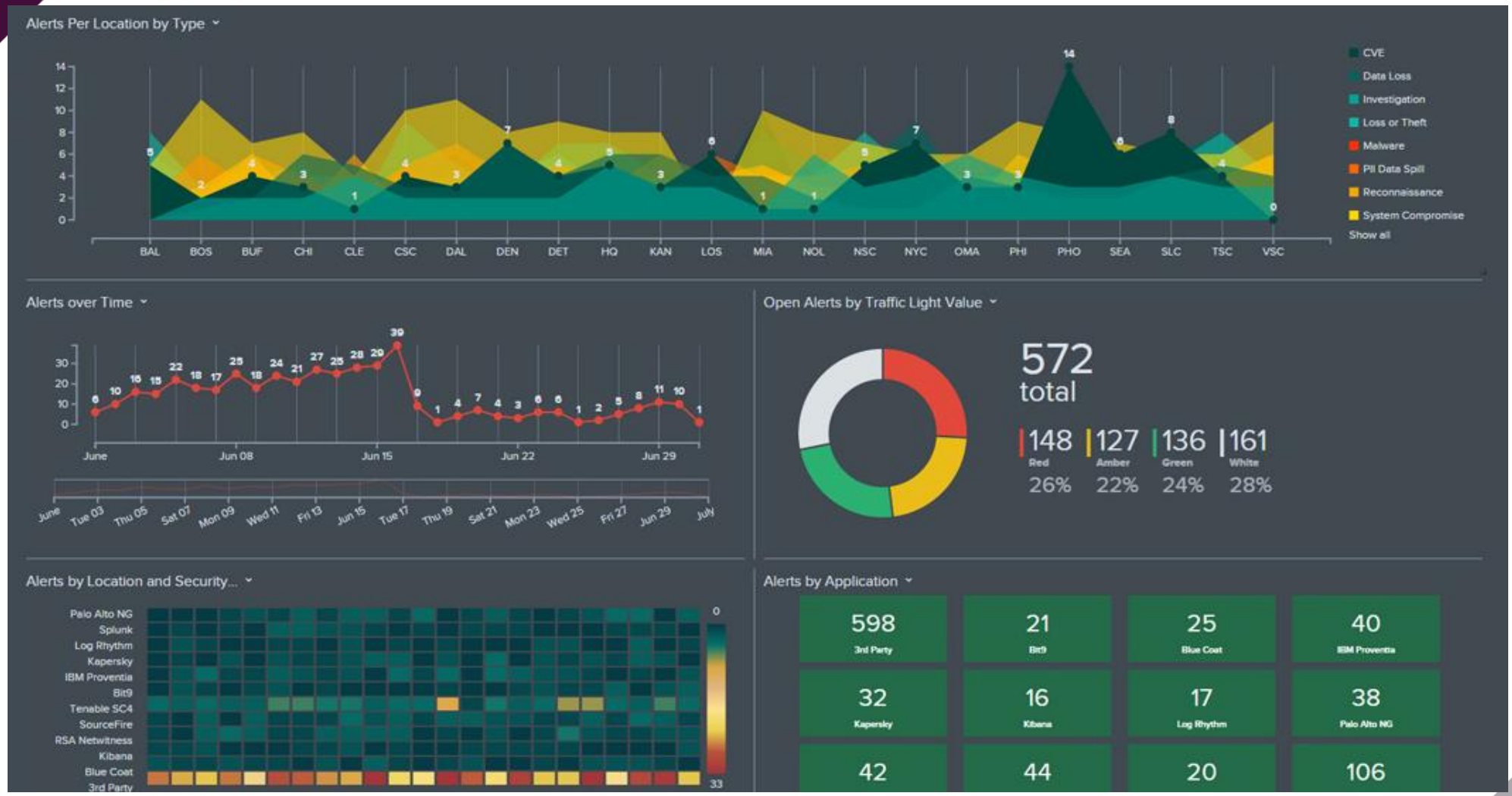
North American terminal block

Segregation of duties new style





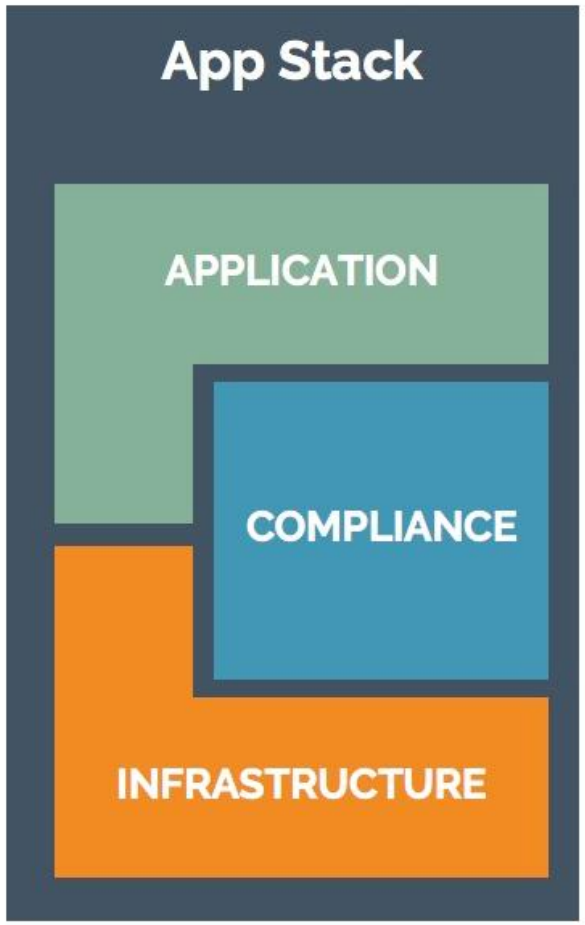
Monitoring


Alerts Per Location by Type
BAL BOS BUF CHI CLE CSC DAL DEN DET HQ KAN LOS MIA NOL NSC NYC OMA PHI PHO SEA SLC TSC VSC
CVE
Data Loss
Investigation
Loss or Theft
Malware
PII Data Spill
Reconnaissance
System Compromise
Show all
Alerts over Time
June Jun 08 Jun 15 Jun 22 Jun 29
Open Alerts by Traffic Light Value
572
total
148 Red 26%
127 Amber 22%
136 Green 24%
161 White 28%
Alerts by Location and Security...
Palo Alto NG
Splunk
Log Rhythm
Kapersky
IBM Proventia
Bit9
Tenable SC4
SourceFire
RSA Netwitness
Kibana
Blue Coat
3rd Party
Alerts by Application
598 3rd Party
21 Bit9
25 Blue Coat
40 IBM Proventia
32 Kapersky
16 Kibana
17 Log Rhythm
38 Palo Alto NG
42
44
20
106

Make it a feedback loop

Xebia

Managing the Application stack as code



Complete application lifecycle managed as code with a frictionless path from laptop to production

Codify how the application is built, how it runs, and all of its dependencies to free the app from underlying infrastructure

Define policies as code to detect issues before production and discover non-compliance for fast remediation

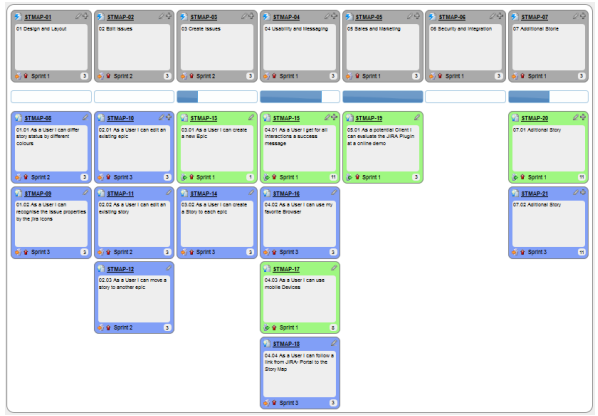
Manage infrastructure as code to provision, harden, and maintain configuration state

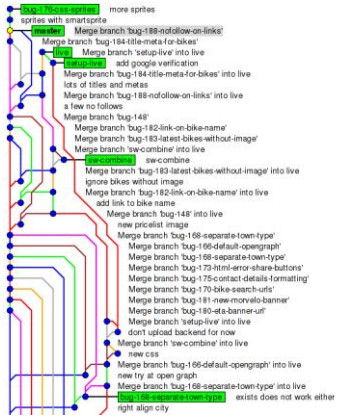
Audit Automation

Work items

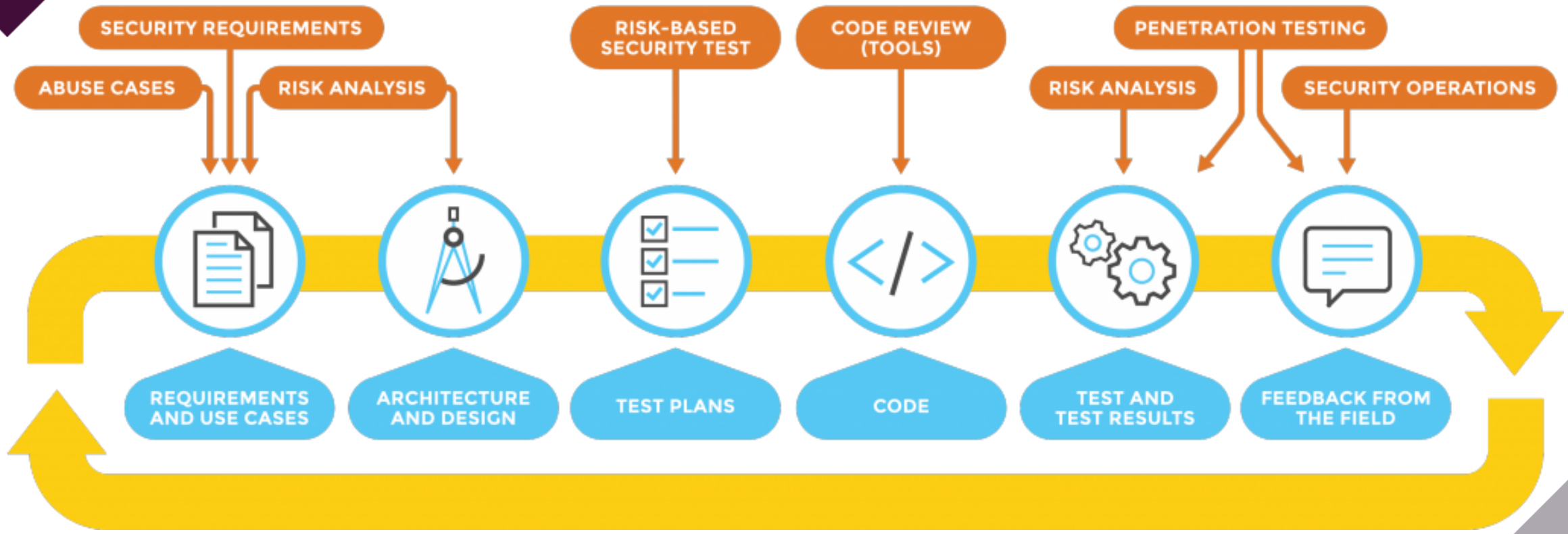
Version control

Continuous Delivery





Continuous security



Xebia

DevSecOps: summary

- Align Dev, Sec, Bus, and Ops
- Standardize and simplify
- Automate, automate, automate
- Know your value
- Attack yourself
- Learn, teach and train

More information!

- **Whitepaper: ‘The IT manager guide to DevOps’**

  → **https://xebialabs.com/resources/whitepapers/the-it-managers-guide-to-devops/**

- **Whitepaper: ‘Becoming an agile security officer’**

  → **https://pages.xebia.com/becoming-an-agile-software-security-officer**

Xebia