How to Deal With Millions of Vulnerabilities?

Ahmet Akan
Sr. Application Security @ Trendyol

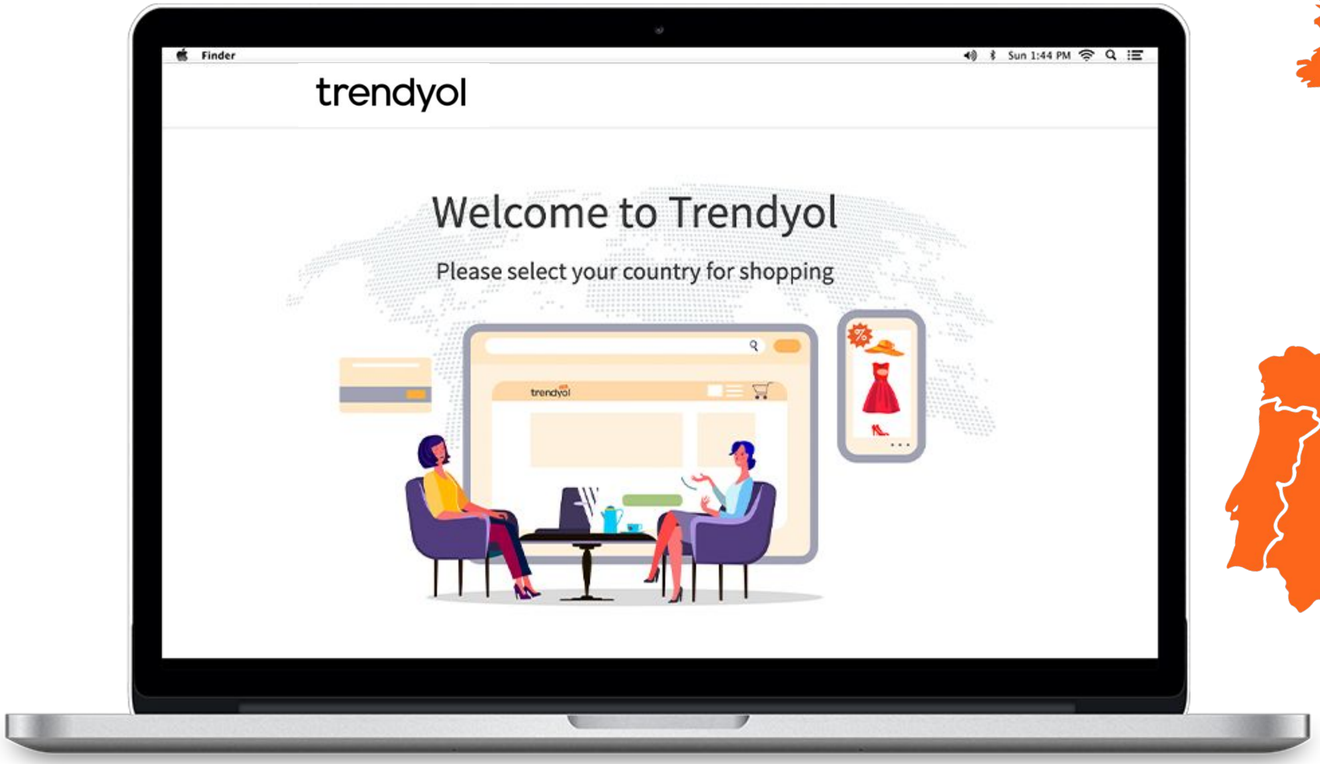
Berkay Aksaray
Sr. Application Security @ Trendyol

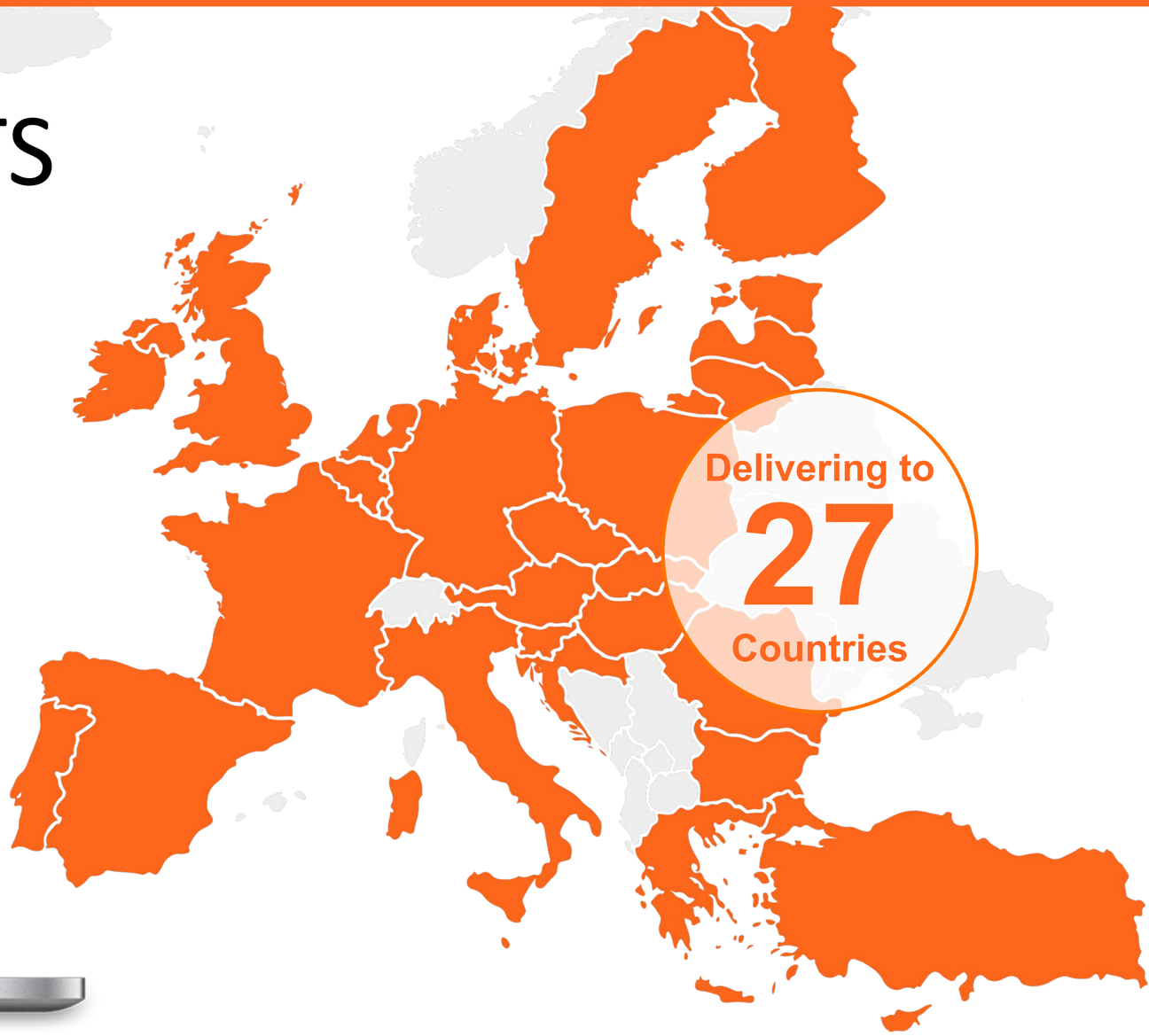
AGENDA

1. Introduction
    a. Trendyol Highlights
    b. Appsec Team
    c. Trendyol Ecosystem
    d. Problem
    e. Solution Plan
2. Manuel & Automation Tests
    a. DevSecOps
    b. Source Code Analysis
    c. External Scan
    d. Bug Bounty
3. Vulnerability Management
    a. Issue Reporter
    b. Re-Test Checker
4. Product Security
    a. Threat Model
    b. Partnership
    c. Awareness
5. K8S & Container Security
    a. Secure Image
    b. Policy Management
    c. Cloud Native App Protection Tool
6. Future Plan
    a. What is the Next
7. Q&A

INTRODUCTION

TRENDYOL HIGHLIGHTS





APPSEC TEAM

1. 8 Appsec Researchers
2. Contact with us any time
   - ahmetakan
   - berkay-aksaray


Ahmet Akan


Berkay Aksaray


Emre Durmaz


Nuri Yavuz


Enes Bulut


Fatih Çelik

Kürşat Oğuzhan Akıncı


Talha Karakumru

TRENDYOL ECOSYSTEM

trendyol

1000+ Developers

inframetrics.trendyol.com

App Metrics

20K+
Projects

2M+
Running Pipelines
(last 6 months)

452
Jira Board

Infra Metrics

3K+
Cluster

13K+
Microservices

191K+
Running Pods

Tribes

Customer Services

Checkout

Product

Member

Wallet

Order

Seller

Mobile

Payment

Other Services 15+

trendyol.com

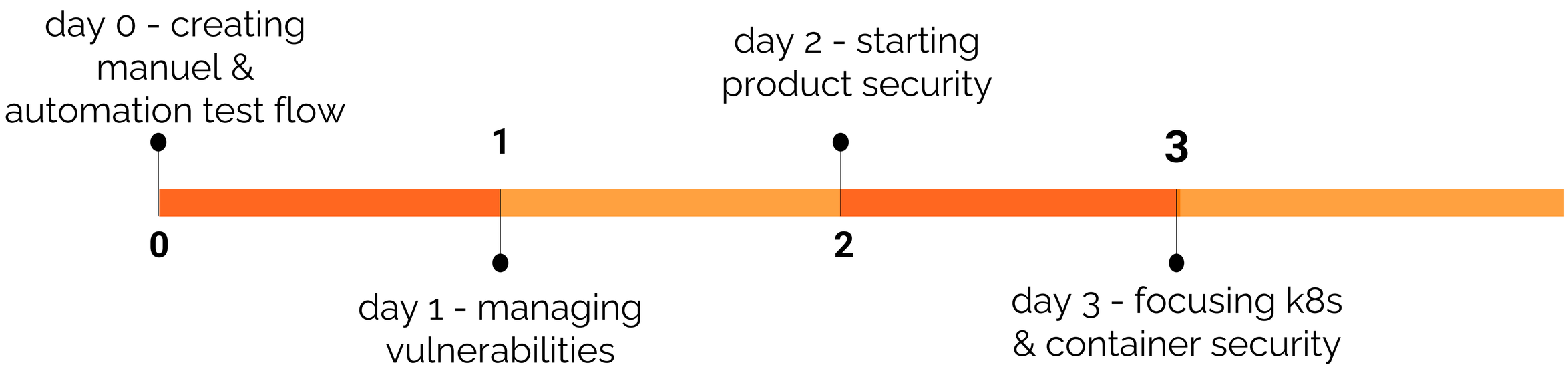
PROBLEM

We need ensure our applications' security under below conditions:

- continuously raising of project count
  - more than 6K new in one year (10673 to 16708 just one gitlab)
- continuously raising of deployment count
  - more than 160K merge request
  - more than 2M pipeline running in one year
  - deployment frequency: 180 per/day
- more than 70 different languages & frameworks usage
- using countless 3rd party dependencies, general purpose container images
- and more …

trendyol.com

SOLUTION PLAN

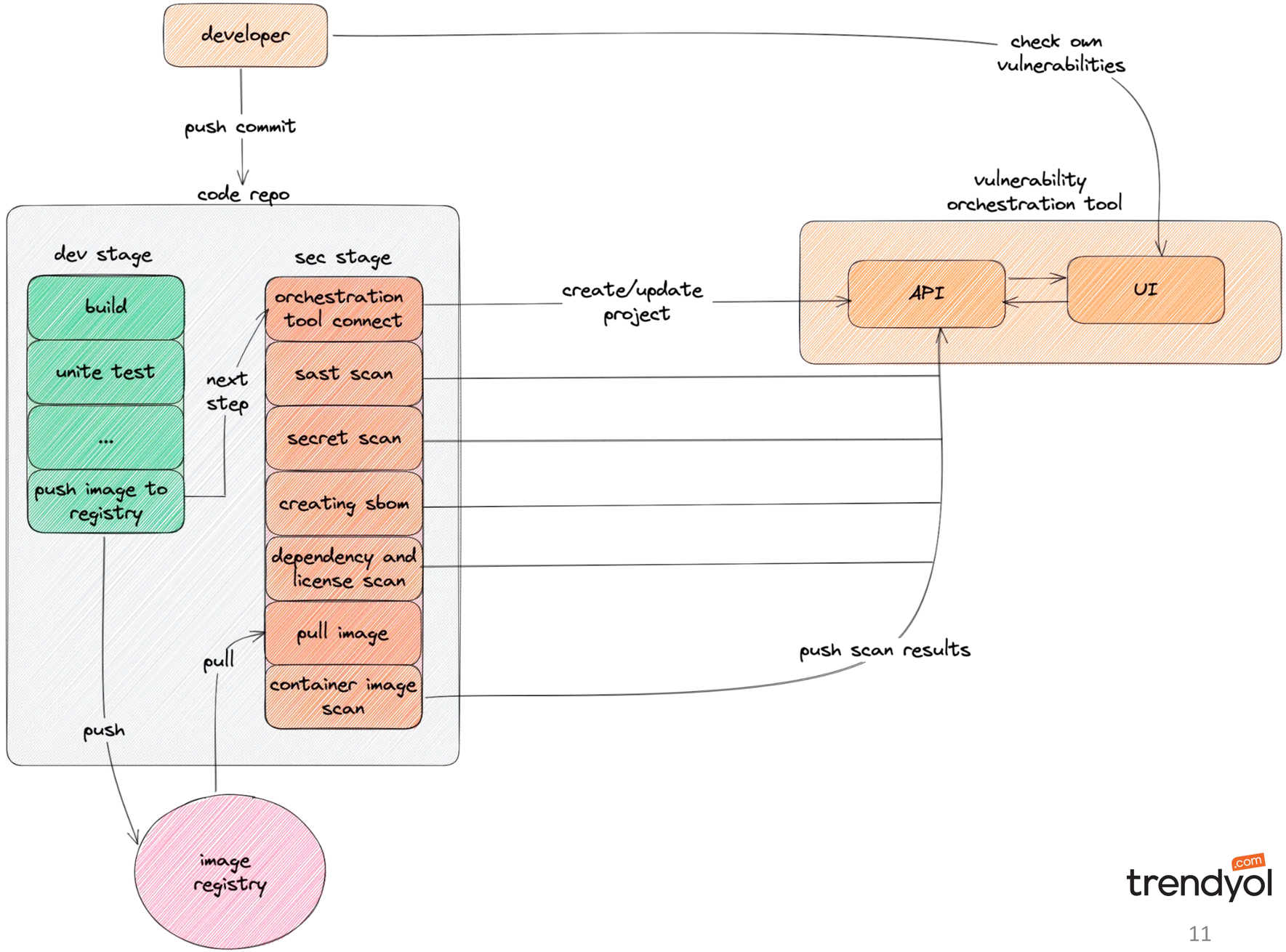


Manuel & Automation Tests

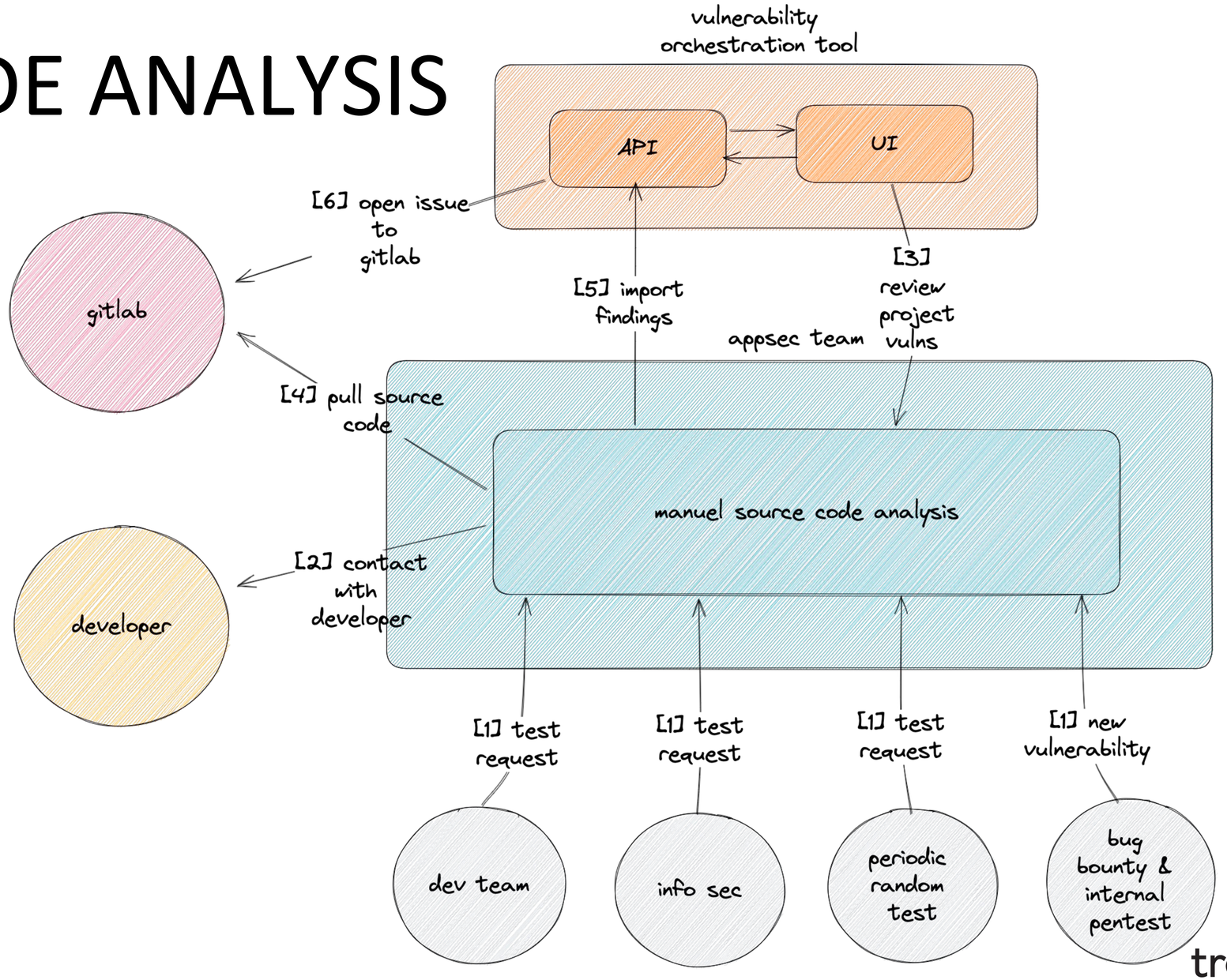
OVERVIEW

- we have a lot of applications and developers but we don’t have enough human resources
  - we absolutely need automation
- automation tools are great but not perfect. we also need cover business logic, authentication and authorization vulnerabilities
  - we need to source code analyse and dynamic web test
- we are seeing our system from inside only but hackers are looking from outside
  - we also need scan our system from outside
  - bug bounty platforms have a huge community, we must get help from community

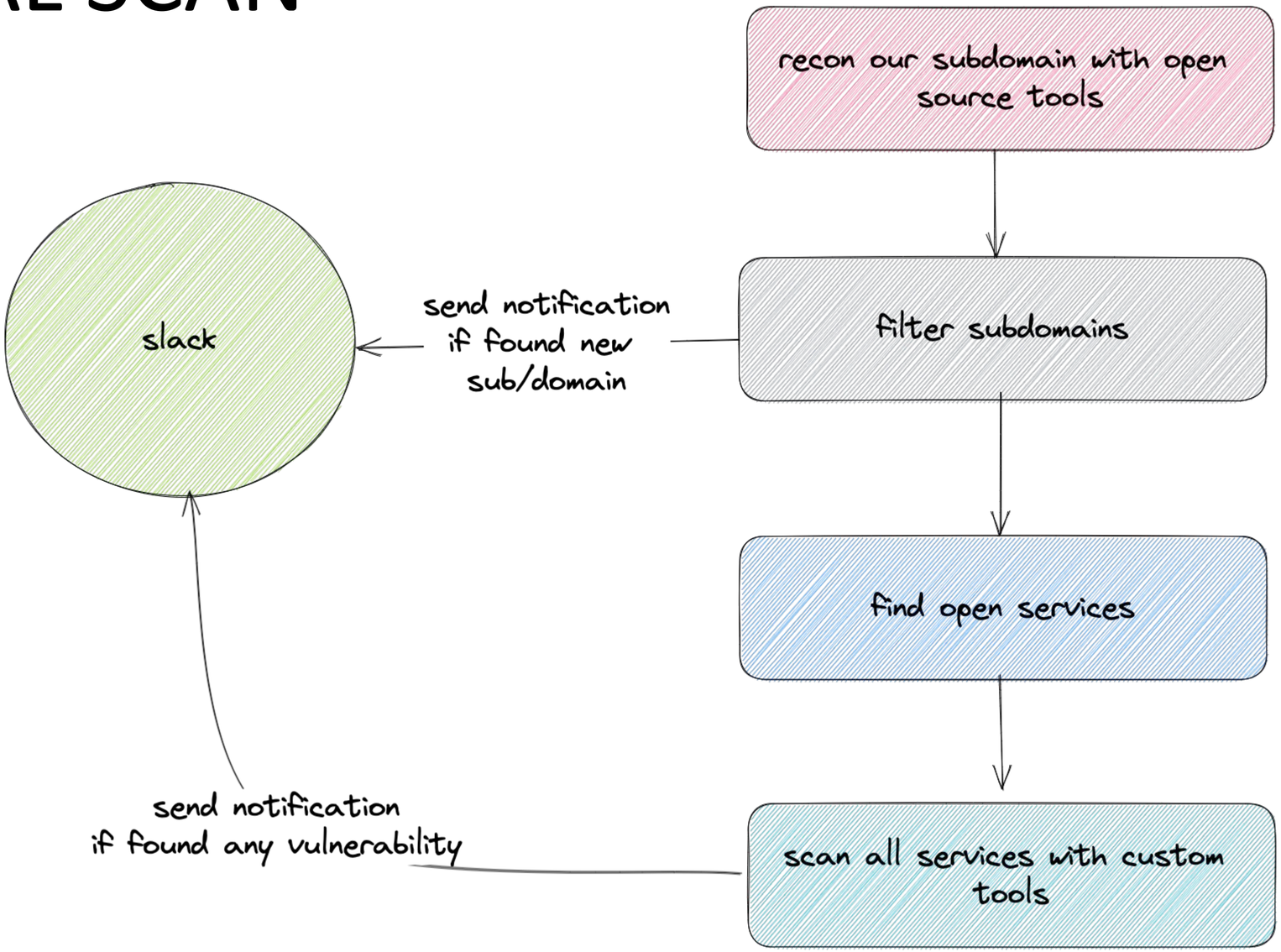
DEVSECOPS


developer
check own vulnerabilities
push commit
code repo
vulnerability orchestration tool
dev stage
sec stage
build
unite test
...
push image to registry
next step
orchestration tool connect
sast scan
secret scan
creating sbom
dependency and license scan
pull image
container image scan
create/update project
API
UI
push scan results
pull
push
image registry

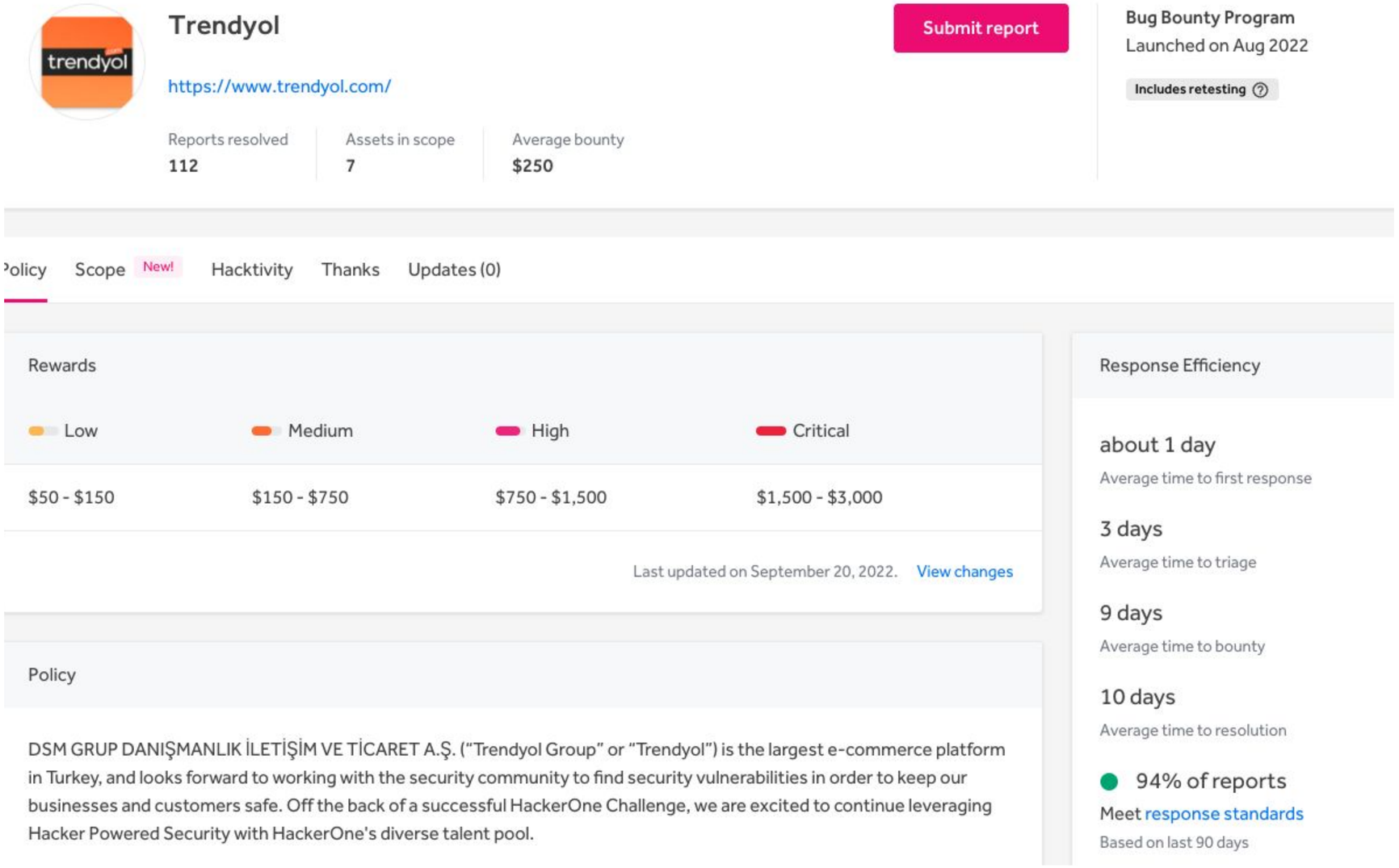
SOURCE CODE ANALYSIS



EXTERNAL SCAN



BUG BOUNTY

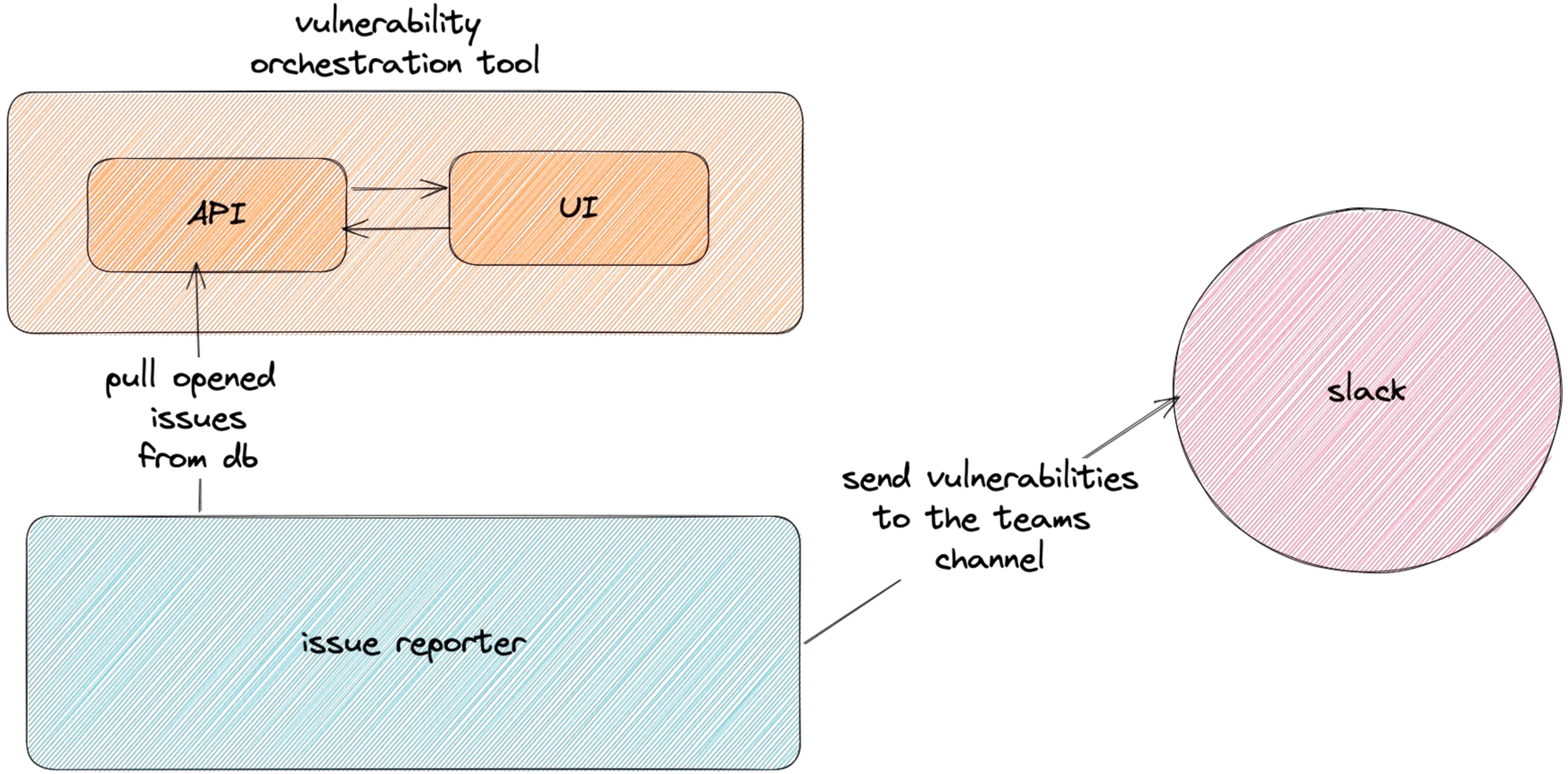

Trendyol
https://www.trendyol.com/
Submit report
Bug Bounty Program
Launched on Aug 2022
Includes retesting
Reports resolved
112
Assets in scope
7
Average bounty
$250
Policy
Scope
New!
Hacktivity
Thanks
Updates (0)
Rewards
Low
Medium
High
Critical
$50 - $150
$150 - $750
$750 - $1,500
$1,500 - $3,000
Last updated on September 20, 2022.
View changes
Policy
DSM GRUP DANIŞMANLIK İLETİŞİM VE TİCARET A.Ş. ("Trendyol Group" or "Trendyol") is the largest e-commerce platform in Turkey, and looks forward to working with the security community to find security vulnerabilities in order to keep our businesses and customers safe. Off the back of a successful HackerOne Challenge, we are excited to continue leveraging Hacker Powered Security with HackerOne's diverse talent pool.
Response Efficiency
about 1 day
Average time to first response
3 days
Average time to triage
9 days
Average time to bounty
10 days
Average time to resolution
94% of reports
Meet response standards
Based on last 90 days

Vulnerability Management

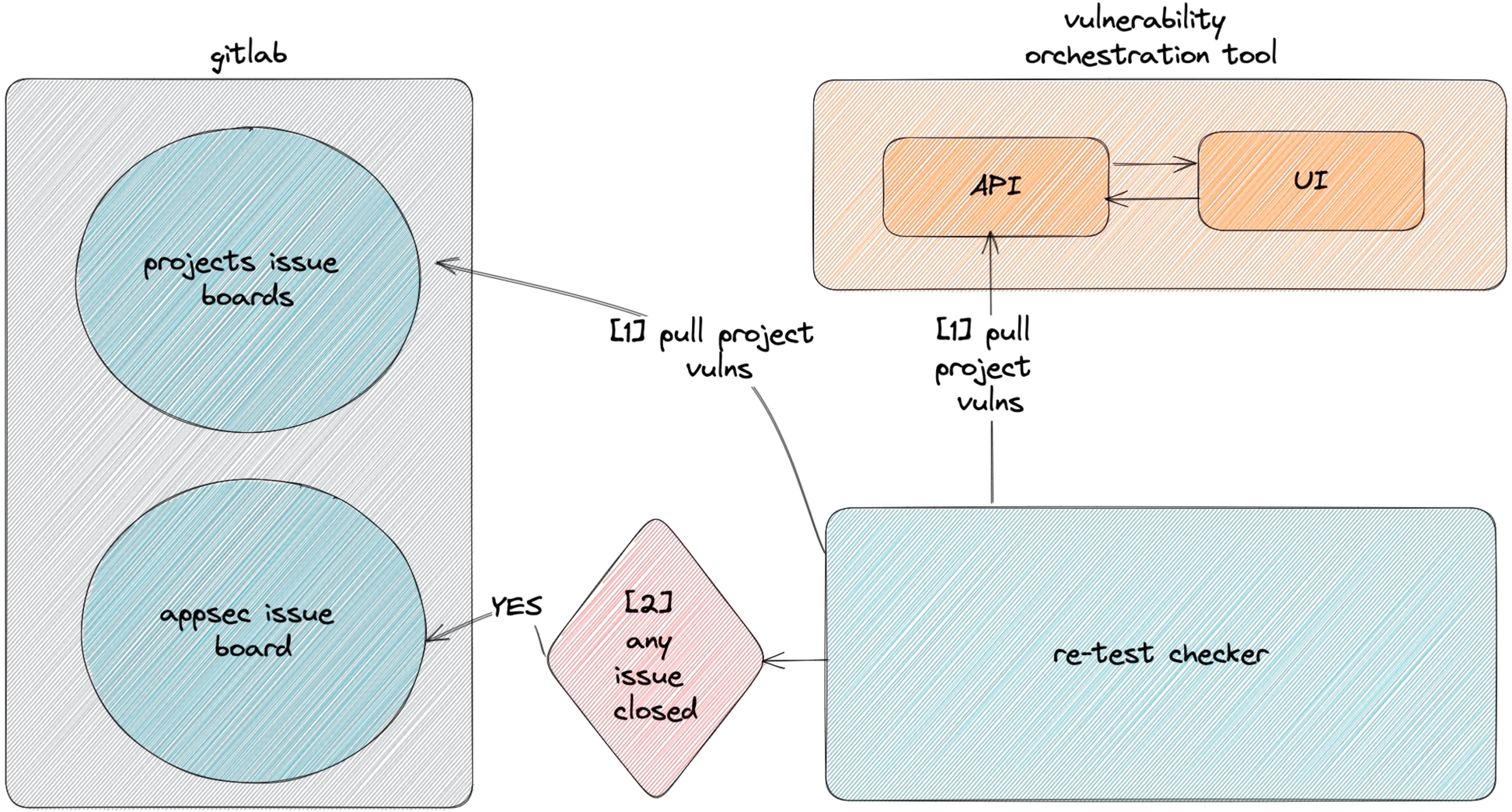
OVERVIEW

- detection of a vulnerability is important but we aim to resolve also that vulnerabilities
  - necessary to keep reminding of the vulnerabilities.not only developers but also team leaders and PO/PM
- thousands of projects means millions of vulnerabilities.
  - re-tests should be automated as well.
- a lot of tools exist but most of them are not able to solve our requirements.
  - custom tool development must

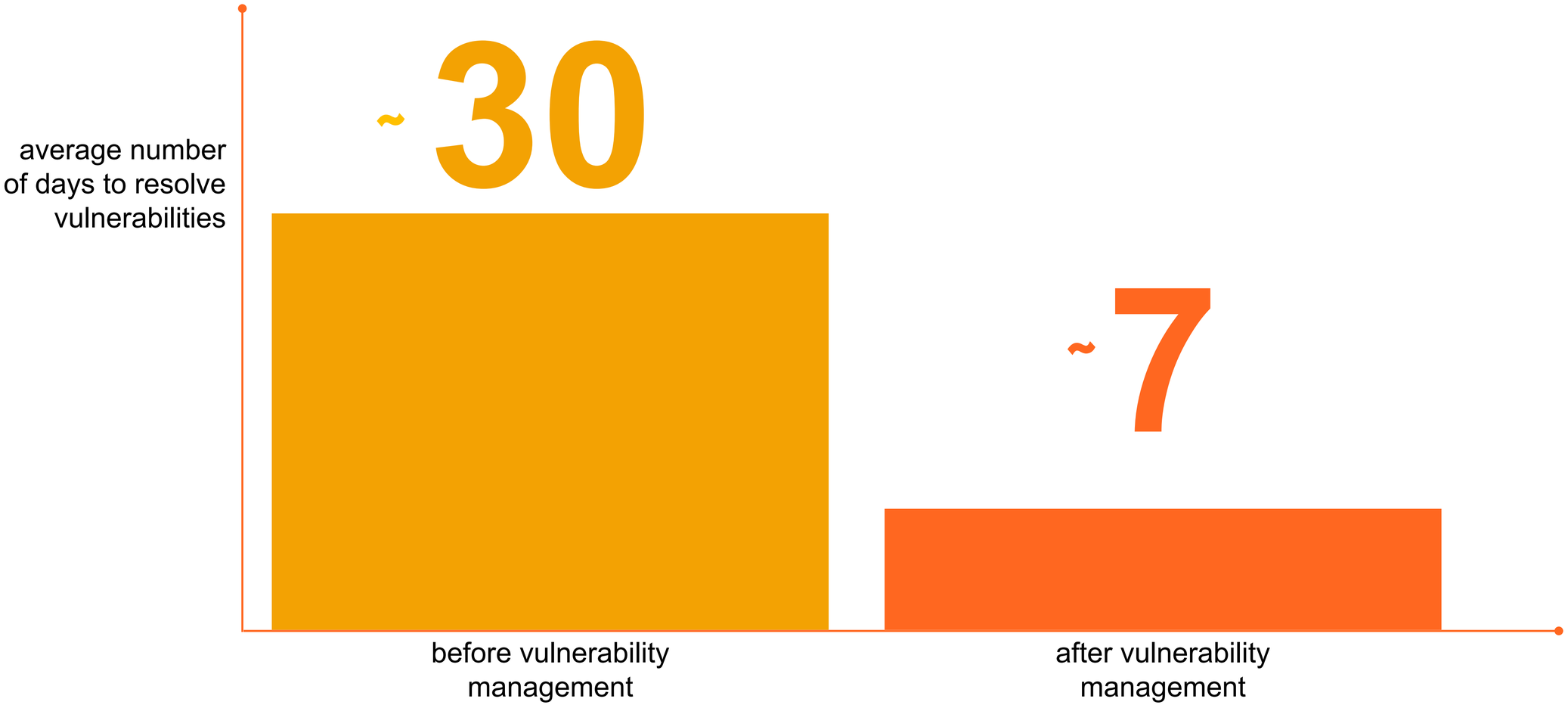
ISSUE REPORTER



RE-TEST CHECKER



RESULT

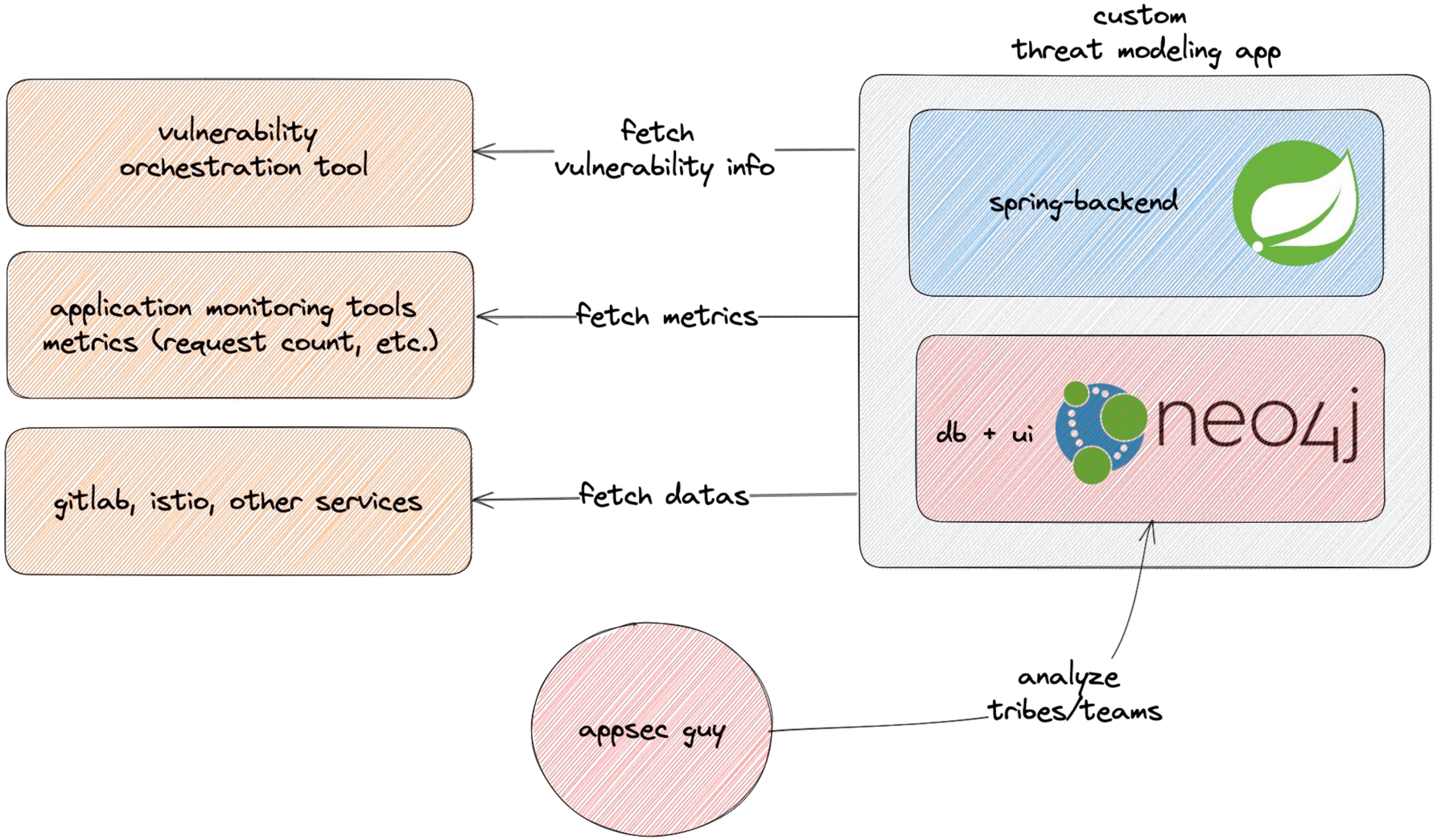

average number of days to resolve vulnerabilities
~ 30
~ 7
before vulnerability management
after vulnerability management

Product Security

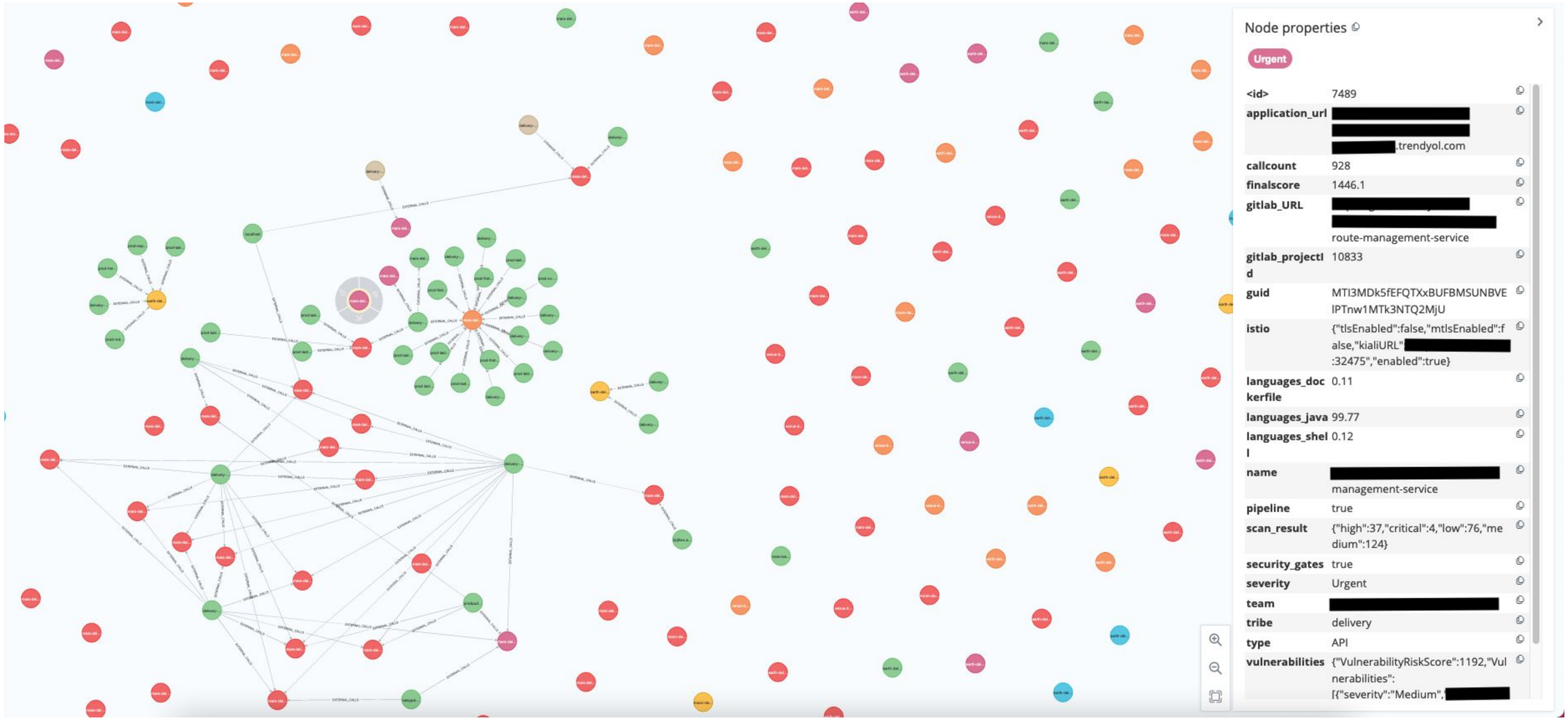
OVERVIEW

- applications are not the same. each one has a different business and security criticality level
  - thread modeling must to calculate criticality level of each MS.
- not enough appsec member to cover all tribes
  - Some critical tribes we have to work closely
- creating appsec standards
- secure software development education

THREAT MODELING



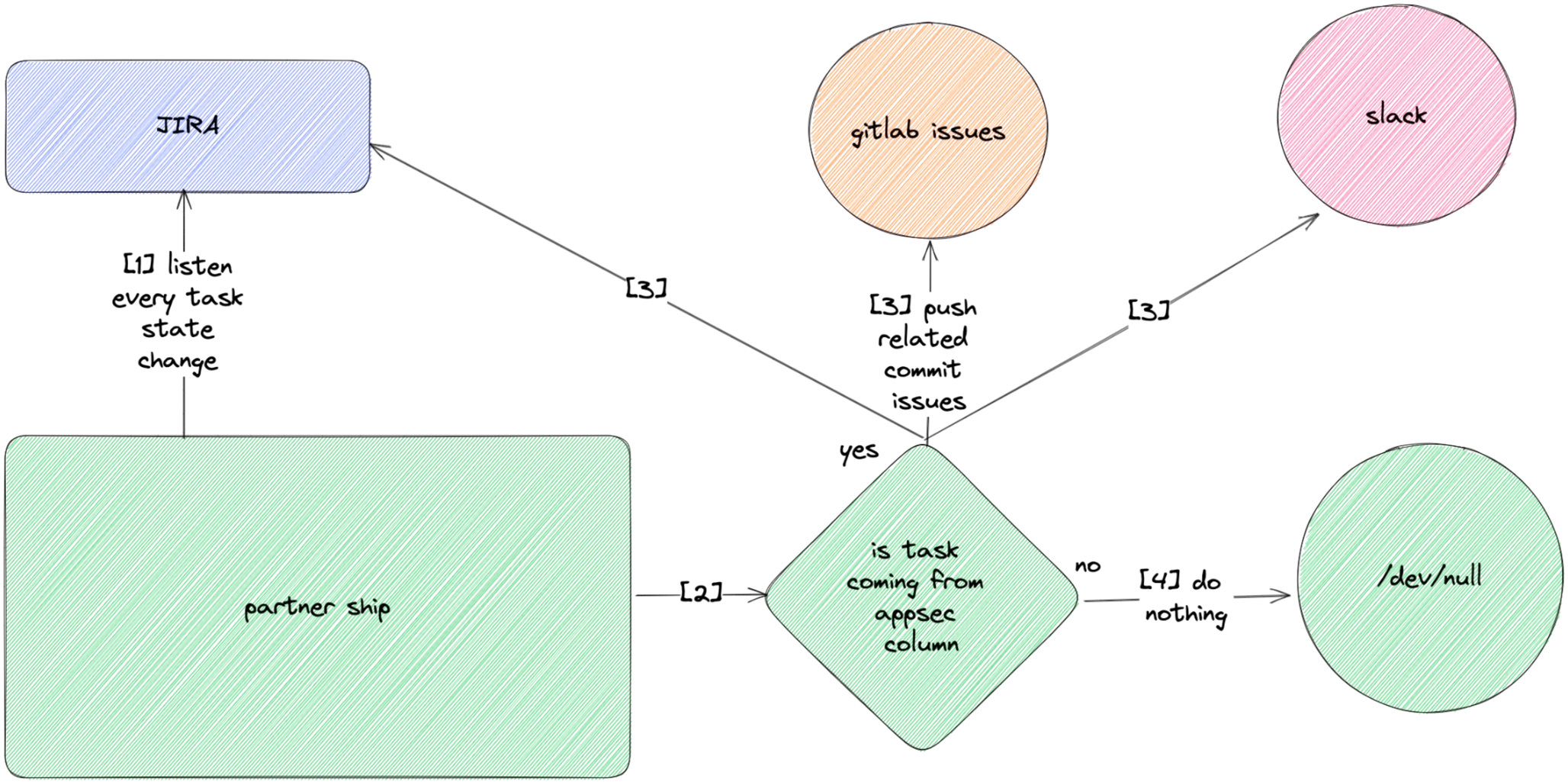
THREAT MODELING



PARTNERSHIP

- every team have different security requirements
- founding classic vulnerabilities good but we need to find more complex vulnerabilities
- we need to increase awareness about application security

PARTNERSHIP



PARTNERSHIP

AppSec - Vulnerability Summary

You can see the latest AppSec status of Gitlab repositories that I can associate with this task below.

- SLA for Issues

Vulnerability Summary:

SAST: Category of vulnerabilities detected in source code scans of your applications (XSS, SQLi, Code Execution etc.)

DAST/Manuel: Category with vulnerabilities discovered as a result of dynamic tests performed on your applications

SCA: Category with security or license issues detected in open source libraries your applications use

CS: Category with vulnerabilities discovered as a result of scanning the container images you use to deploy your applications

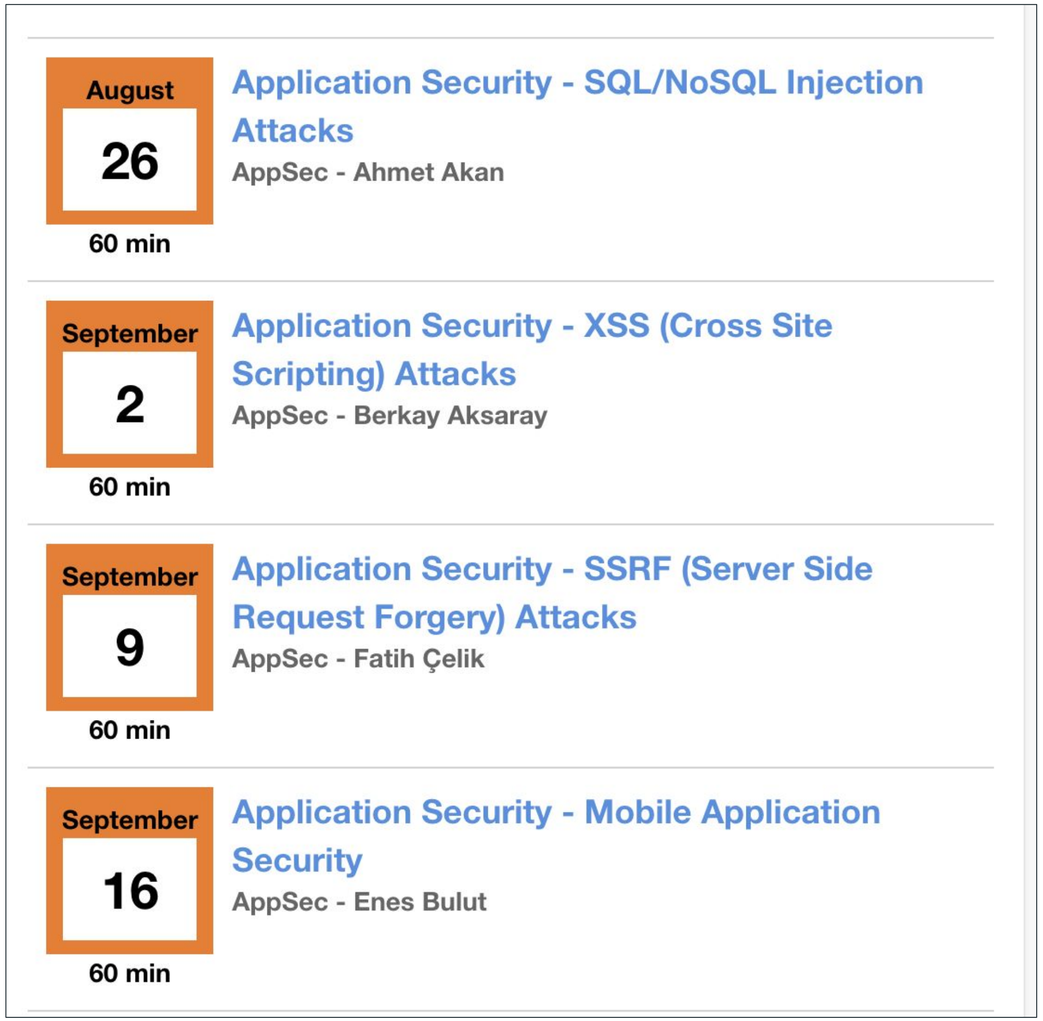
ISSUES: Category with the findings discovered in your projects and sent to you by AppSec for you to fix

Project Name: [redacted]
[redacted]**Address:** Project Link

Severity	SAST	DAST	SCA	CS	ISSUES
Critical	0	0	0	12	0
High	9	0	0	39	0
Medium	2	0	0	62	0
Low	0	0	0	5	0

AWARENESS

- creating appsec standards for teams. best practice, mitigation methods, etc.
- internal education that created with vulnerabilities which detected during development process
- external education platform for improve the secure software development skills that will affect developer’s annual success grade



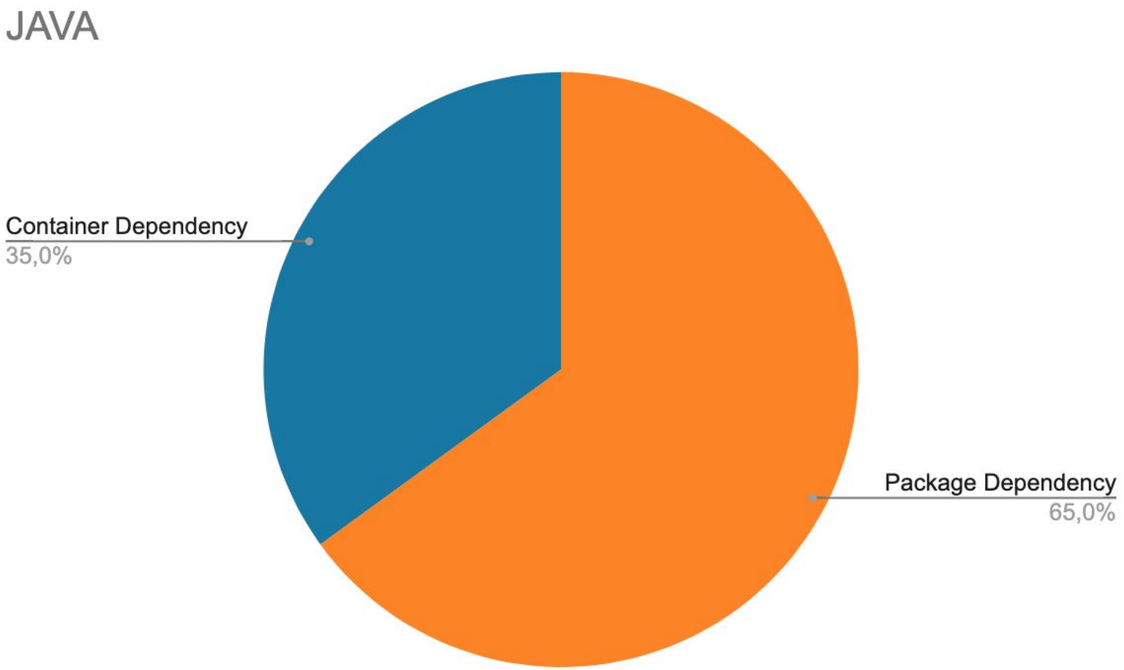
K8S & Container Security

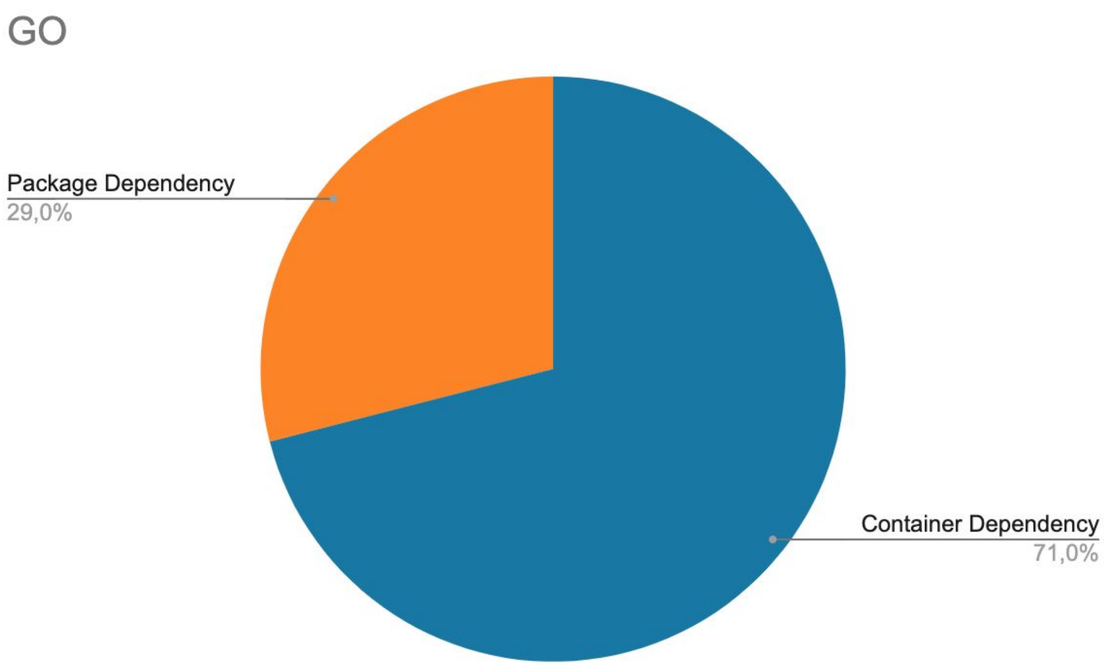
OVERVIEW

- 75% of all vulnerabilities caused by 3rd party application and container dependencies.
  - it’s a common problem in company. we need to provide secure container image to dev teams.
- our assets are applications and it work on k8s cluster. we need to know that our application deployed securely
  - we need to learn how we manage k8s configurations during deployment

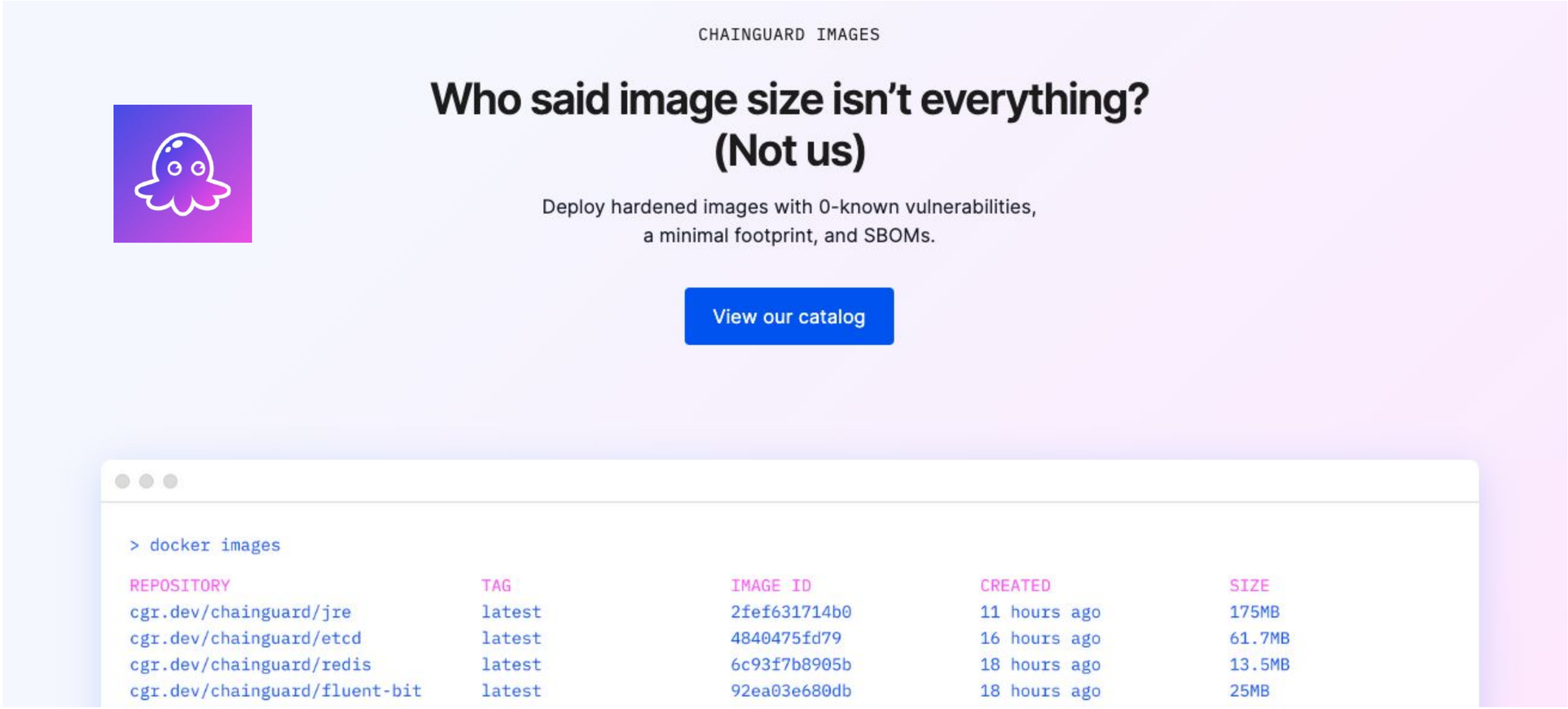
SECURE IMAGE

- 35% of java project vulnerabilities caused by container images
- 71% of go dependency vulnerabilities caused by container images

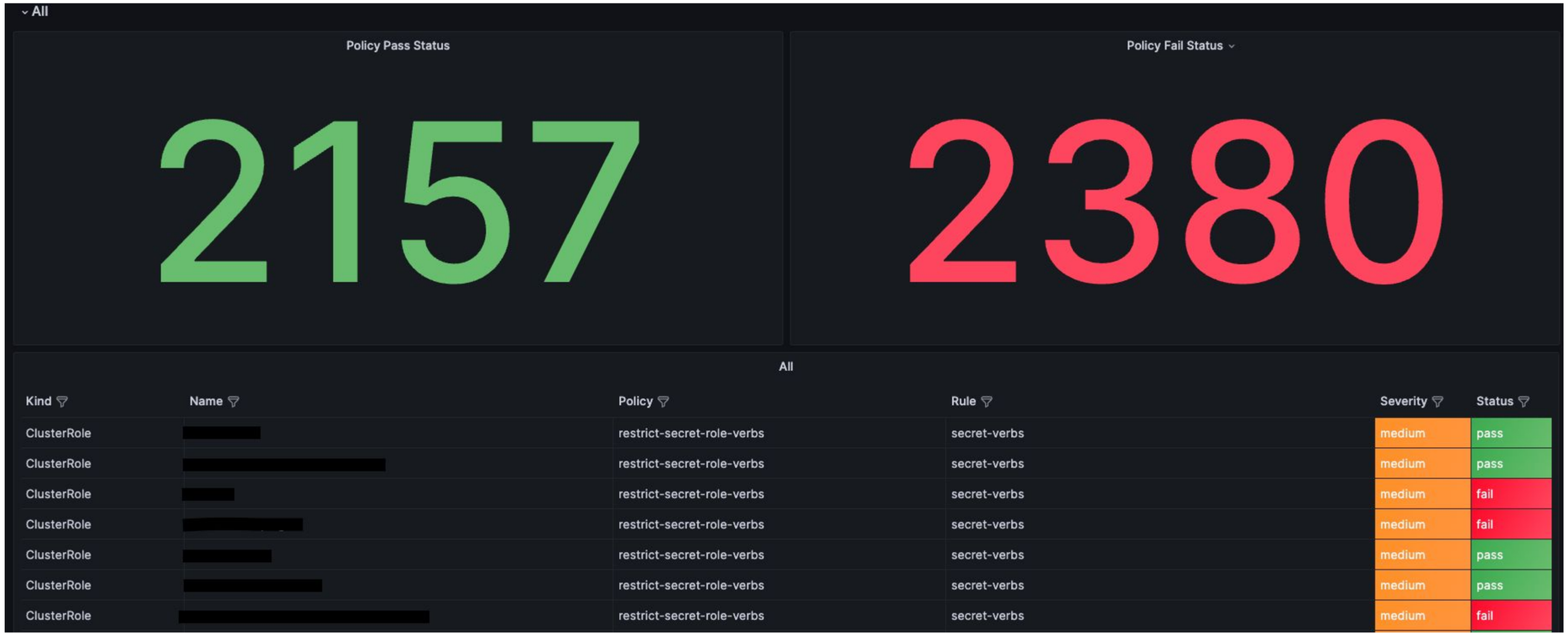




SECURE IMAGE



POLICY MANAGEMENT



Kyverno

trendyol.com

CLOUD NATIVE APP PROTECTION - CNAPP

- need to scan our clusters and containers continuously
  - host, container and app dependencies scan
- need to know our cluster compliance issues
  - pci-dss and similar regulations
- api protection & waf for container
  - api discovering
  - creating rules for any endpoints

FUTURE PLAN

WHAT IS THE NEXT

- security hero
- binary fuzzing
- ide plugin & pre-commit

Q&A

THANK YOU FOR LISTENING