



Simon Skelton

Platform & Operations Manager
John Lewis & Partners



Chris Rutter

Principal Consultant
Equal Experts

Paving a Secure Road at John Lewis & Partners



JOHN LEWIS | WAITROSE

Adapting to Change - Building on a Strong Heritage



John Spedan Lewis believed in “fairness and humanity” & **experimented** to create the largest **employee owned business** in UK



JOHN LEWIS

WAITROSE



The ‘official start’ of Christmas?!



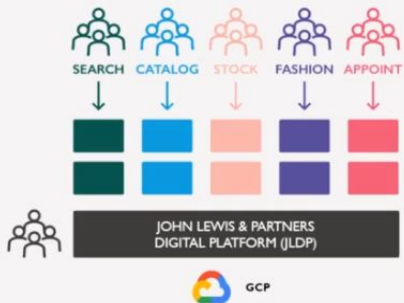
DevOps Enterprise Summit Europe 2021

Operability and You Build You Run It at John Lewis & Partners

Operability and You Build It You Run It at John Lewis & Partners

ENTERPRISE SUMMIT
AN AI REVOLUTION EVENT

2017 - 2020 timeline



- Nov 2017 - 3 pilot product teams**
Commitment to digital platform and digital services. Pilot cross-functional product teams start.
- Nov 2018 - 4 product teams**
Cloud search goes to 1% live traffic on JLDP. Pilot teams deemed successful. More teams start.
- Nov 2019 - 25 product teams**
Multiple teams on-call for 100% live traffic. New propositions emerge. JLDP wins awards.
- Nov 2020 - 30 product teams**
40 digital services, 100 microservices. Record levels of Black Friday traffic, no major incidents.

John Lewis Partnership | John Lewis & Partners | Waitrose & Partners

GET TOGETHER
GO FASTER

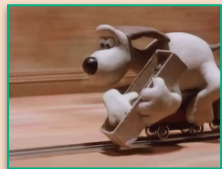
VIRTUAL-EUROPE | 18-20 MAY 2021
#DOES2021 #TREVDOES

06:16

DEVOPS ENTERPRISE SUMMIT



Our “Paved Road” Journey Continues



List of features is vast - two Platform sub-teams look after >300 git repositories that makes the platform tick

✓ Simple config-driven “paved” GCP resource provisioning

✓ Scalable, automated observability - dashboards + alerts

✓ Day 2 Tools - Dev Portal, Delivery Metrics, Runbooks & Incidents ...

✓ Range of security tools in pipeline and at runtime

✓ Environments - Namespaces + GCP projects “owned” by Teams

✓ Simplified Kubernetes resource creation. Full power there if needed

✓ Avoidance of wheel reinvention - ingress & egress, caching, SSO ...

DevOps Excellence Awards Winners 2023:

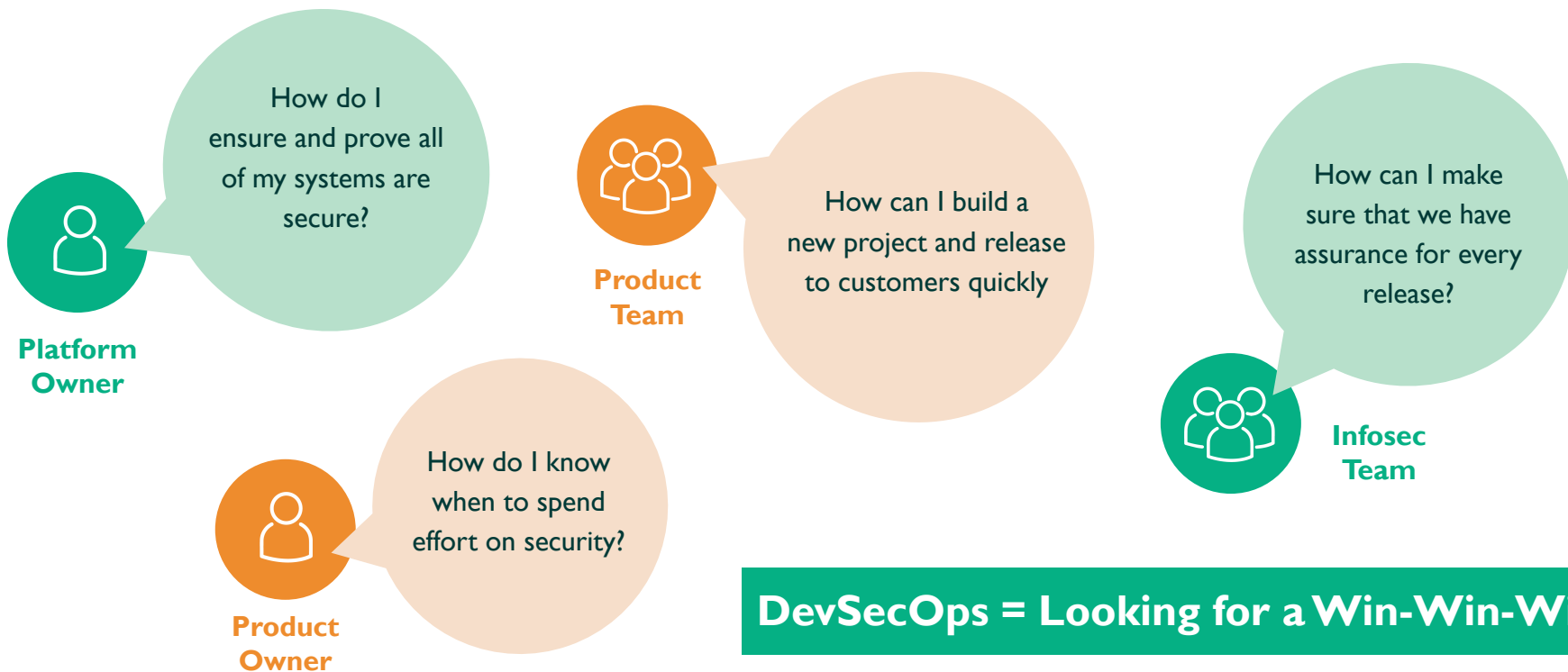
“Best Implementation of DevSecOps”



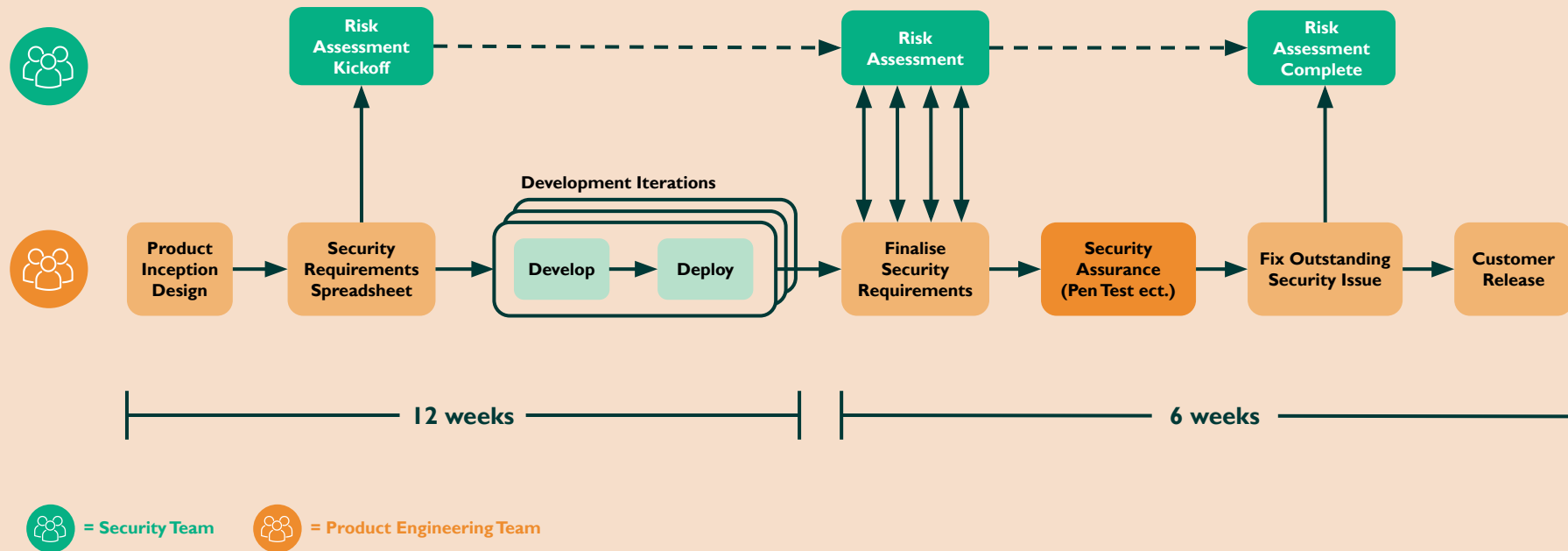
We ♥ DevSecOps!

computing
DevOps Excellence Awards

What Were Our Security Challenges?

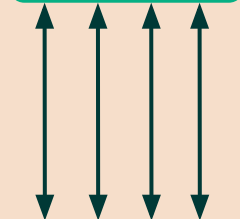


Security Workflows Were Too Slow - 3 Years Ago



Complex Manual Process Impeded Releasing New Features

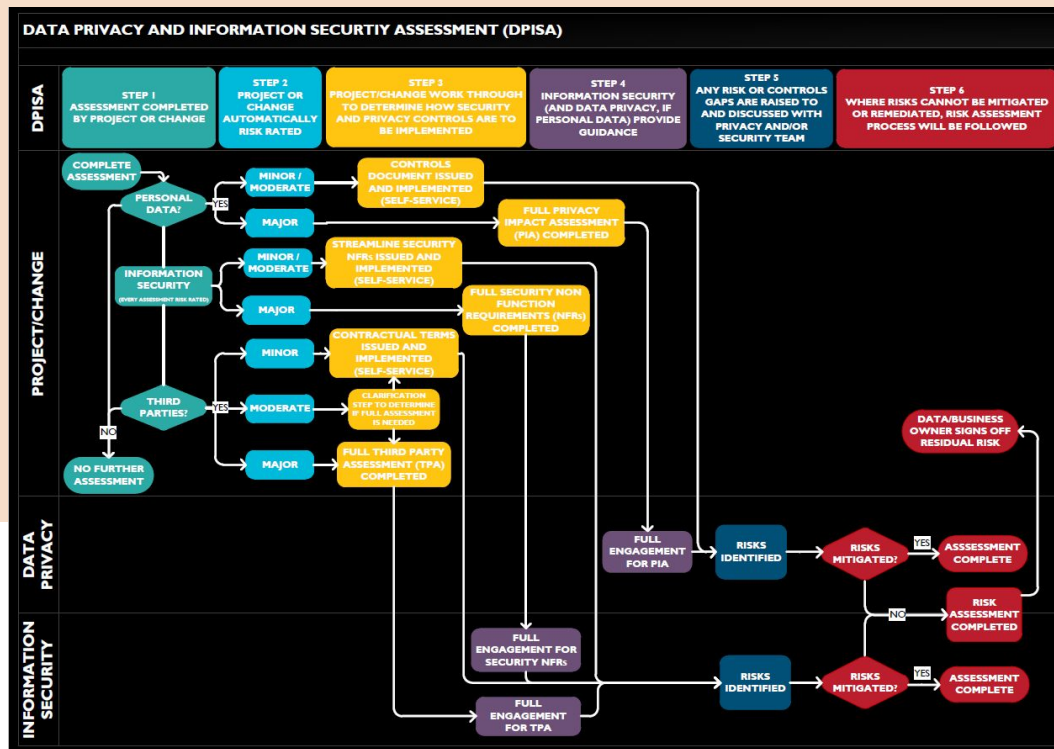
Risk Assessment



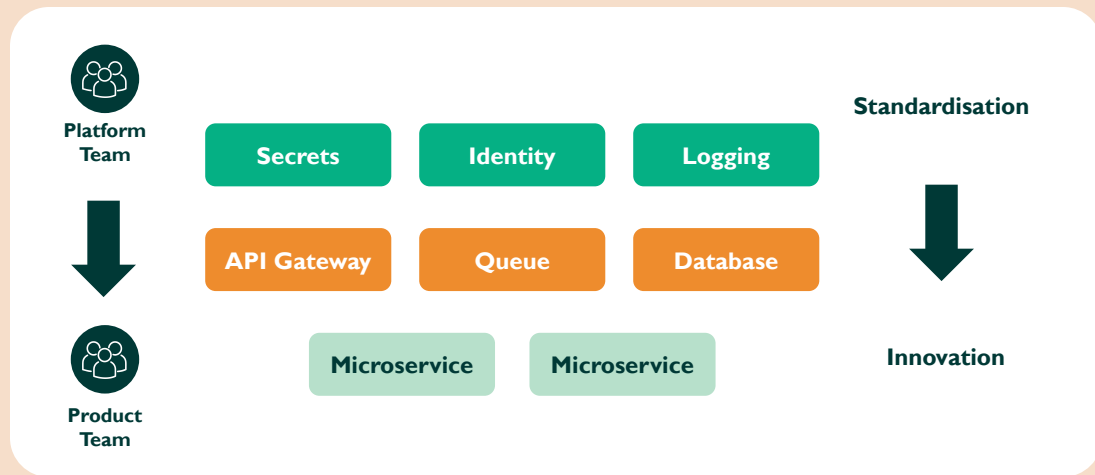
=

Finalise Security Requirements

“75% time finding out what to do and 25% actually doing it”



We Built Cross-Cutting Security Controls



- **Identified** which requirements are cross-cutting and which are unique to each team
- **Built standardised controls** that teams get “as standard” and which we know are secure
- **Shared Ownership and configuration** of components which require both

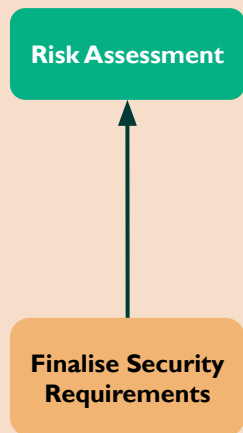
Cross-Cutting Security Controls for Free

```
service_id: browse-components
team_name: Browse
team_slack_channel: redacted-slack-channel
product_owner_email: redacted-email-address@johnlewis.co.uk
service_owner_email: redacted-email-address@johnlewis.co.uk
gcp_project_root: redacted-project-name
google_group: redacted-email-address@johnlewis.co.uk
service_level: Important
support_level: Bronze
```

```
microservices:
  redacted-service-name-1:
    core_language: "java"
    gcp_iam_roles:
      - "datastore.indexAdmin"
      - "datastore.user"
  redacted-service-name-2:
    core_language: "java"
    gcp_iam_roles:
      - 'pubsub.subscriber'
      - 'pubsub.viewer'
```

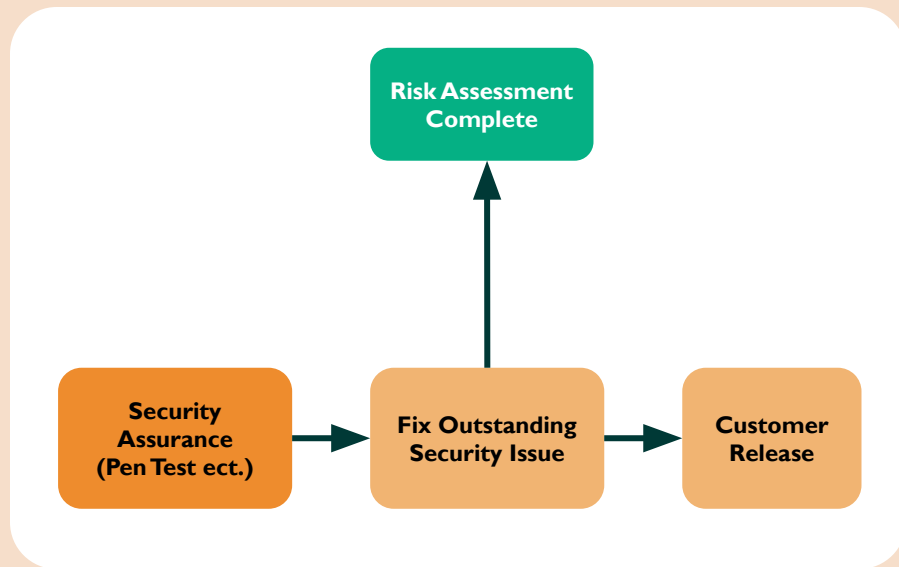
- **Identity & Access** configured for least-privilege
- **Public ingress with WAF** and enforced authentication
- **Secrets Management** using runtime injection
- **Logging** automatically configured and access locked down
- **Communication** through standardised slack channel

Much Faster Security Assessments



Requirement	Status	Detail
Identity & Access	✓	Platform Provided
Secrets Management	✓	Platform Provided
Security Logging	✓	Platform Provided
Public Ingress	✓	Platform Provided

Our Application Security Reviews - 3 Years Ago



- **Finding vulnerabilities just before release** impacted delivery timelines
- **Only 1 week of testing for months of coding** could mean potential security gaps
- **Lack of review after go-live** didn't maintain security for the life of the service

How do you Secure ALL the Things?!



TOOLS FOR ALL THE THINGS



Bring Your Own Scanner

Engineers

- Quality Scanning Engine
- Rapid Feedback
- Easy Onboarding

Delivery Lead / Product Owner

- Service Vulnerability Metrics
- Defined Security Policy
- Red/green Policy Check

Business Owner / Infosec Teams

- Platform-wide Metrics
- Compliance Reports
- Trend Analysis



Scan Type

contrast-library + contrast-application + secrets-auditor + image-vulnera...

Open Issue Severity

CRITICAL + HIGH + MEDIUM

Vulnerabilities Dashboard

JLDP provides several security tools to help engineering teams secure the software they build by finding vulnerabilities in their systems. To ensure the security of the platform as a whole and guide priority decisions we implement a policy for fixing these vulnerabilities within agreed timelines, based on severity and in line with the wider

Severity	Policy	Type
CRITICAL	issues must be resolved within 5 days	Enforced
HIGH	issues must be resolved within 30 days	Enforced
MEDIUM	issues should be resolved within 90 days	Advised
LOW	Issues should be reviewed	Advised

Service Status

FAIL

Service Summary

Failing Policy

1

Open

9

Critical

0

High

1

Medium

8

Low

0

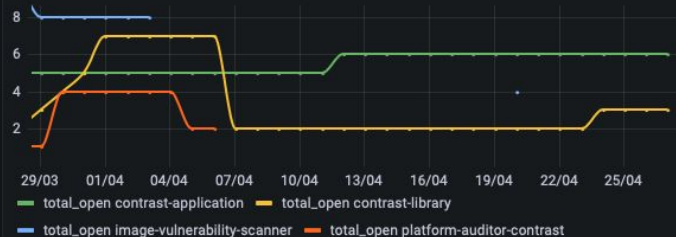
Avg Age (days)

258

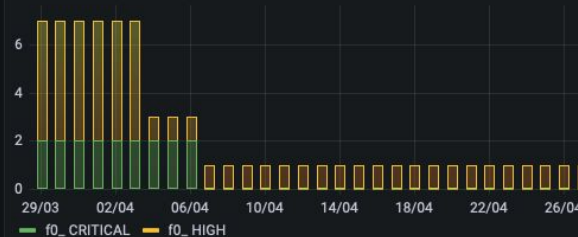
Managed

27

Open Issues By Scan Type



Issues Failing Security Policy Over Time (days)




All Open Issues

Description	Policy	Age	Type	Severity	Link	Target
appointments-appointment-bookings	OUTSIDE	446	contrast-application -	HIGH	https://www.contrastsecurity.com	appointments
appointments-appointment-bookings	INSIDE	446	contrast-application -	MEDIUM	https://www.contrastsecurity.com	appointments
appointments-appointment-bookings	INSIDE	446	contrast-application -	MEDIUM	https://www.contrastsecurity.com	appointments

**Demo
Data*



Clear Security Responsibility

 JLDP Vulnerability Alerts APP 09:00

JLDP Vulnerability Digest for service: beauty



 **Open Vulnerabilities** 

image-vulnerability-scanner

- Critical: 0 High: 3 Medium: 3

No Vulnerabilities Almost Outside of Policy




 **Vulnerabilities Outside of Policy** 

image-vulnerability-scanner

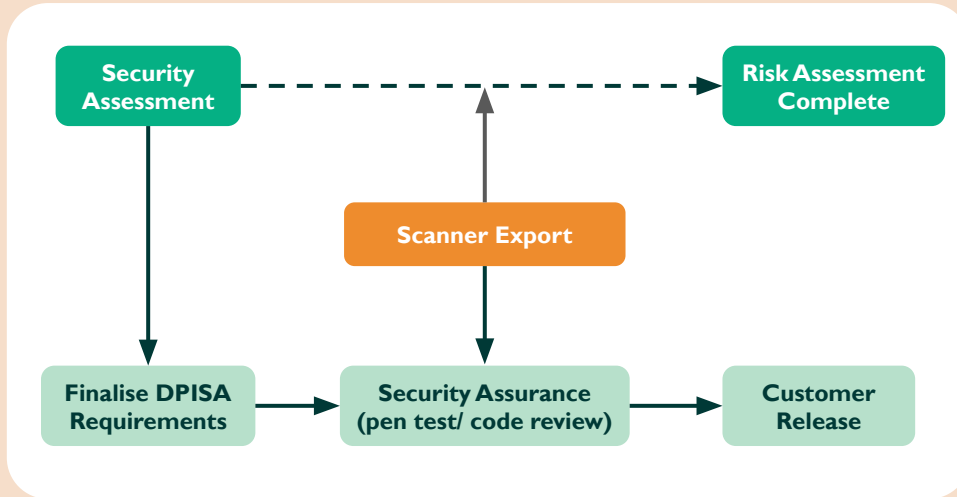
- Critical: 0 High: 3 

Visit your service's [vulnerability dashboard](#) for a detailed breakdown and links to vulnerabilities.

Please fix any vulnerabilities within policy timelines or you will be in breach of our security agreements which can result in escalation to senior business owners. More information on handling findings can be found here: [https://www.waitrose.com/.../agreements](#)

Today ▾

Integrated Security Assurance



- **Less release delay** fixing pen test issues
- **Ability to release smaller chunks** of product features to customers
- **Full coverage of all code through its lifetime**, not just a single Pen Test

Security Metrics Front and Centre

DORA METRICS

Availability

99.98%
(99.97%)



TARGET 99.9%
Based on all Services website responses aggregated across the Platform

Lead Time

7 DAYS
(6 days)



Ideally all services can deploy to production if required in between 1 and 5 days

Deployment Frequency

DAILY
(Daily)



All services are deploying to Production on a better than weekly across the Platform

Security Vulnerabilities Outside Policy

50
75



Critical or High Security vulnerabilities will be fixed within 5 days of discovery.

Major Incidents

0 (£0k)



Change Failure Rate

0.00%
0.06%



The number of production changes that have failed due to a issue are between 0-15%, measure is an average

Restore

78 Mins
(189 MINS)



Target is less than 1 hours for Critical, less than 1 day for non-critical. This measures JLDP services only and is a average.

Performance

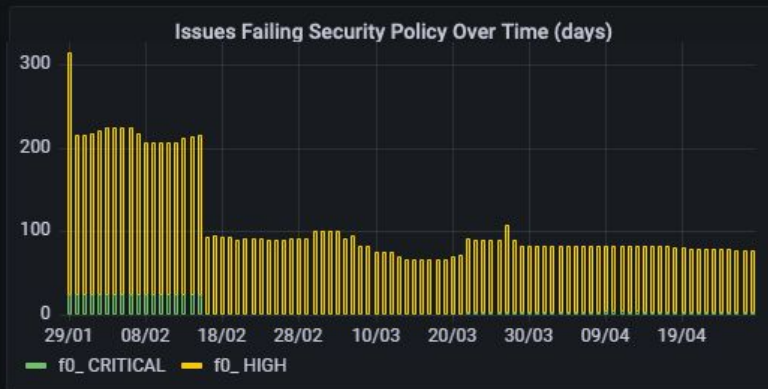
Core Web Vitals - PASS
Live Load Test - PASS

Live Load test and Core Web Vitals combined give a indicator over website performance (Pass/Fail)



**Demo Data*

Service Name All Scan Type contrast-library + contrast-application + secrets-auditor + image-vulnera... Service Level All Severity All



Per Service Summary								
Service ↑	Outside Policy	Open	Critical	High	Medium	Low	Note	Unknown
all-in-one	2	4	0	2	2	0	0	
secrets-auditor	15	45	6	9	20	10	0	
image-vulnerability-scanner	2	4	0	2	2	0	0	
contrast-app	5	16	1	4	7	4	0	
contrast-lib	3	5	0	3	2	0	0	

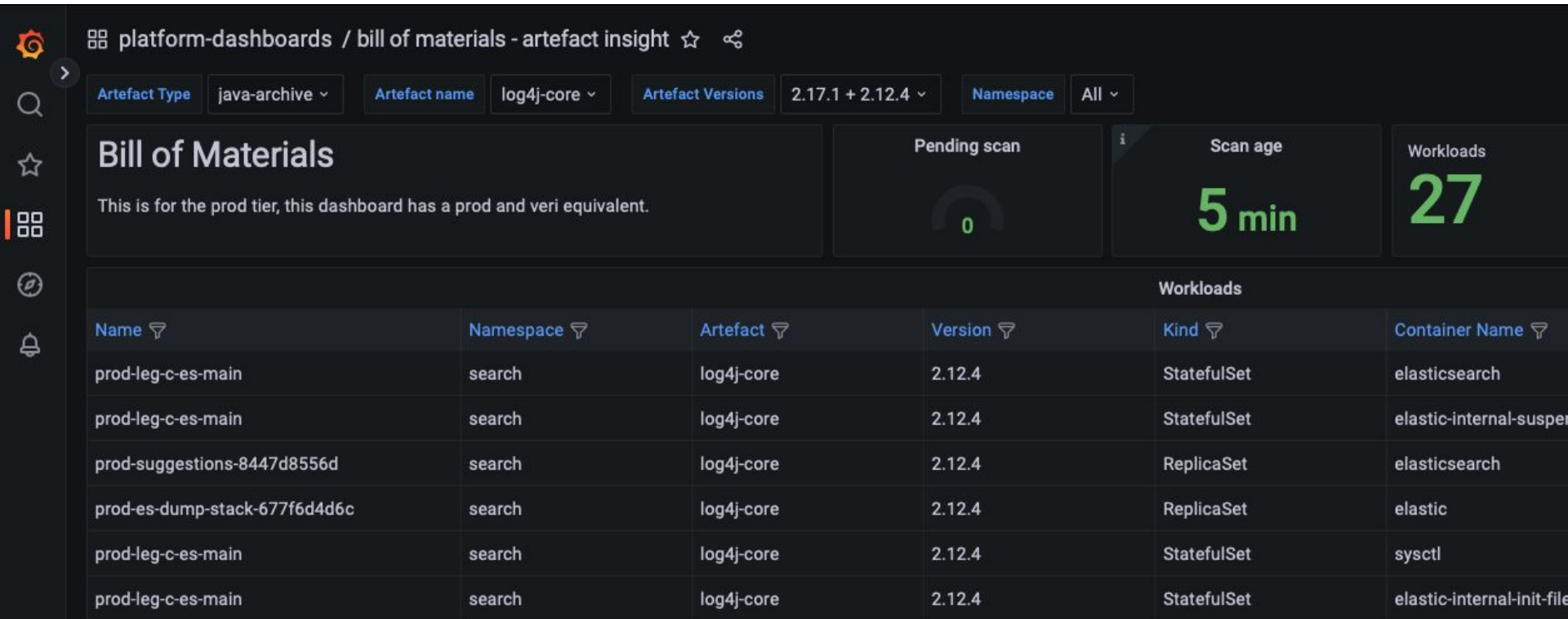
*Demo Data

How About Critical Emerging Threats?



- **Difficult to diagnose** which production systems are affected
- **Who owns what?** And how do we get in touch with them?
- **How do we track remediation?**
- **How long until we were secure?**

Bill Of Materials Dashboard



What's Next?



- **Selecting and tuning scanning tools** for all emerging technologies and frameworks
- **Building more realistic severity scoring for dependency vulnerabilities** so teams spend less time updating dependencies which may not have realistic vulnerabilities
- **Evolving security alerting and monitoring to a shared responsibility model** so service teams can respond to real-time threats directly

Takeaways

**How do you
secure over 700
microservices
releasing daily to
production...?**

- **Share security responsibility** using effective reporting and communication
- **Be opinionated and standardise** key security areas
- **Know when to be un-opinionated** and provide extensible capabilities to allow innovation
- **Understand security requirements for all stakeholders** and build workflows to support them
- **Build vs. Buy** Most effort is spent mapping tools to your domain, so build your own reporting and management “glue”