WEB DAY 2022

ONLINE CONFERENCE

L'IMPATTO DELLA SICUREZZA SU DEVOPS

GIULIO VIAN

PRINCIPAL ENGINEER

UNUM

@GIULIO_VIAN

10 MARZO 2022

#WEBDAY2022

Kudos to Sponsors

Seguici su Telegram: https://t.me/ugidotnet

What it's all about

The **environmental pressure** on software has dramatically changed in few years.

In quality and quantity.

Mainly security concerns.

Pressure impact

How we automate.

How we plan, budget

I suggest to introduce a new term: **Technical Inflation**.

Inflation differs from Technical Debt.

Software value decrease (even drops) over time without intervention.

We won't address today

~~Infrastructure security~~

~~Explain SCA, SAST, DAST, IAST, ...~~

~~Secure SDLC~~

~~Secrets Management~~

~~Governance~~

...

Giulio Vian

unum

Principal DevOps Engineer

First computer

Hardware spec:
1 KB RAM
4 KB ROM

Past employers

buildit @ wipro digital

GLASS LEWIS

AGT International

Microsoft

INFERENTIA

ARTEL

Communities

MVP Microsoft Most Valuable Professional

getlatestversion

@giulio_vian
giuliovdev@hotmail.com

Agenda

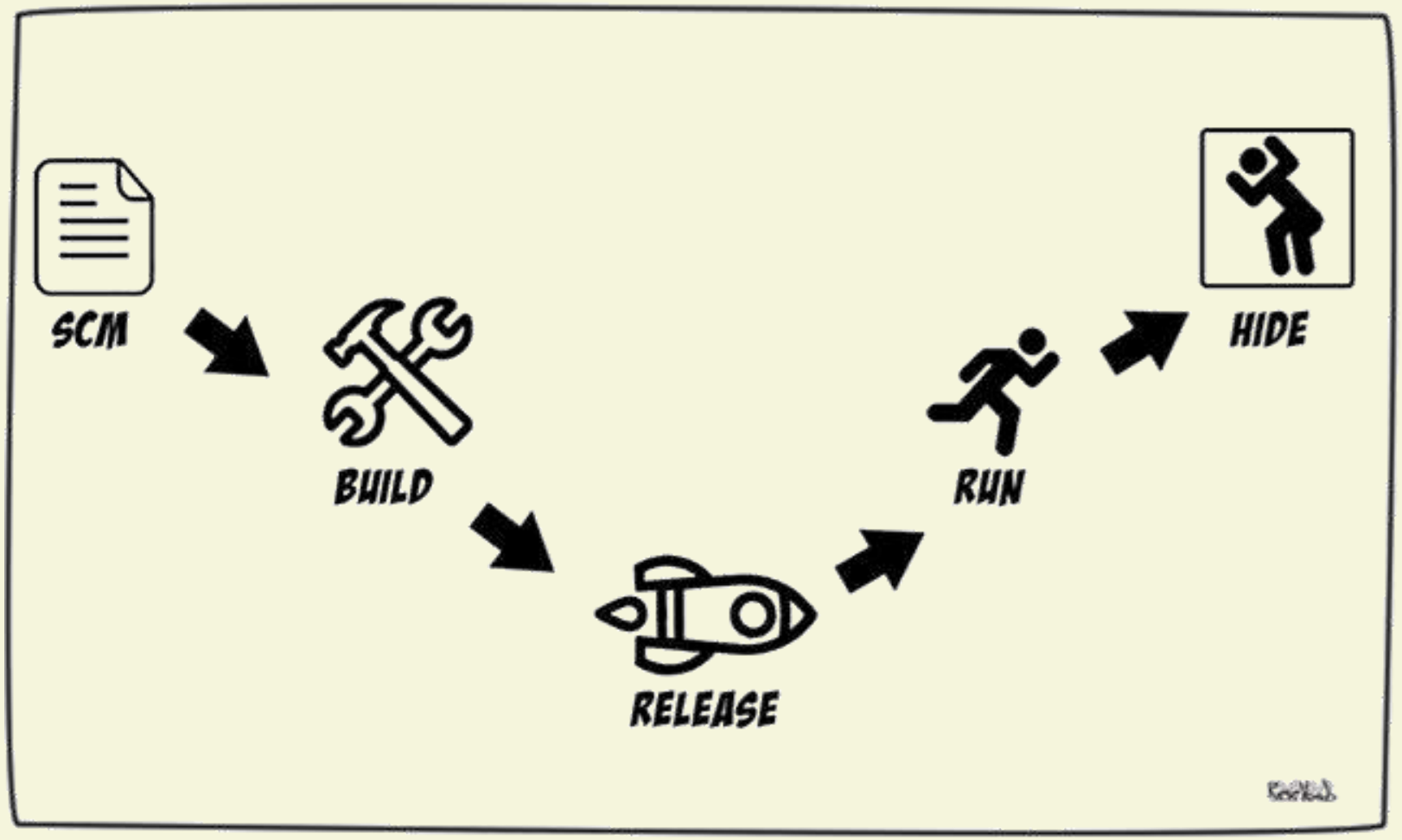
DevOps & Security
Why should you care?
Consequences

Image source: Reddit

DevOps & Security

Background Information

What is DevOps?

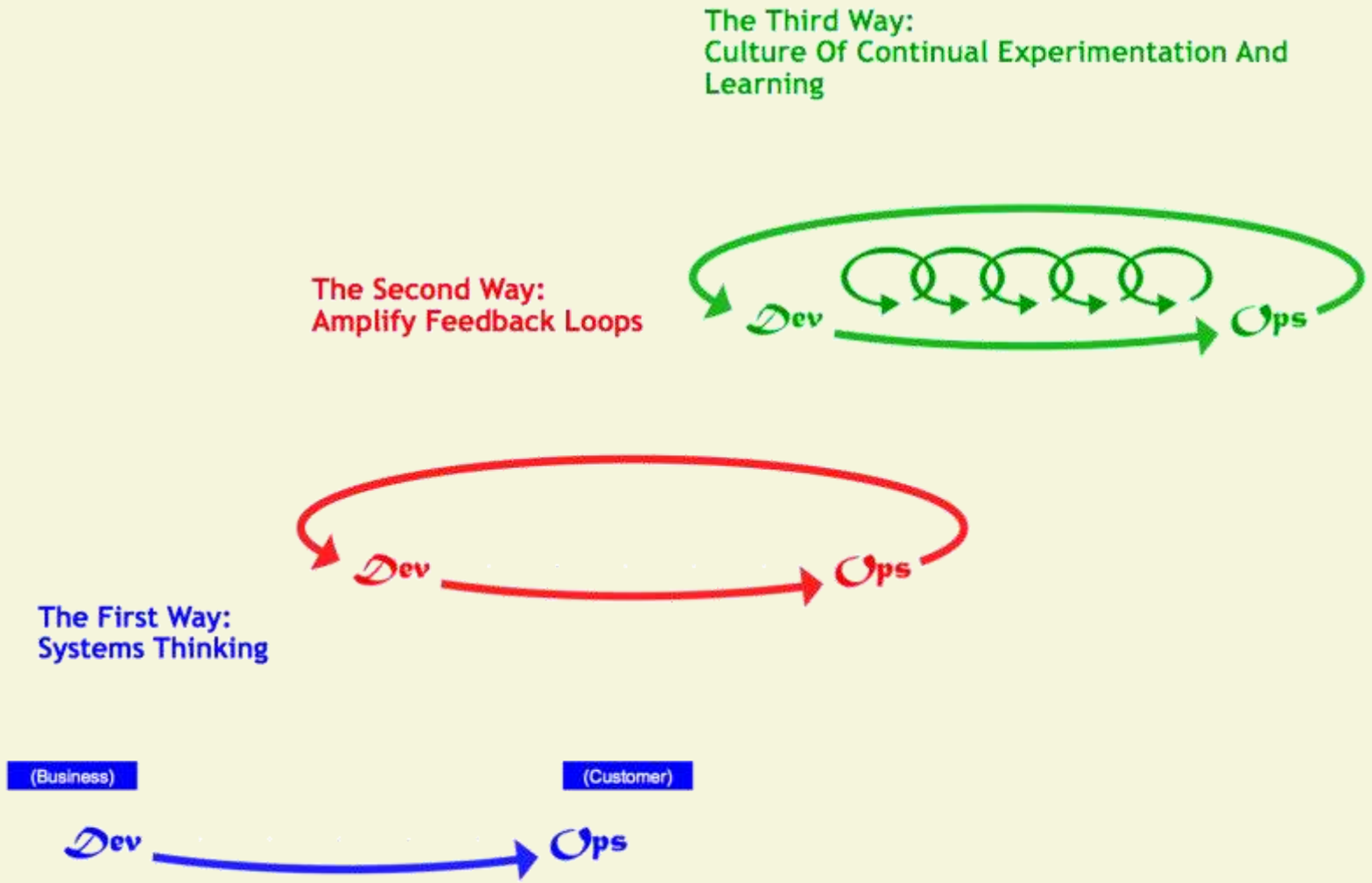


Cartoon by ROELBOB

What is DevOps?

«The result of applying **Lean principles** to the **technology value stream**»

The DevOps Handbook, Gene Kim et al., 2016



The Three Ways: The Principles Underpinning DevOps

Achievement unlocked!

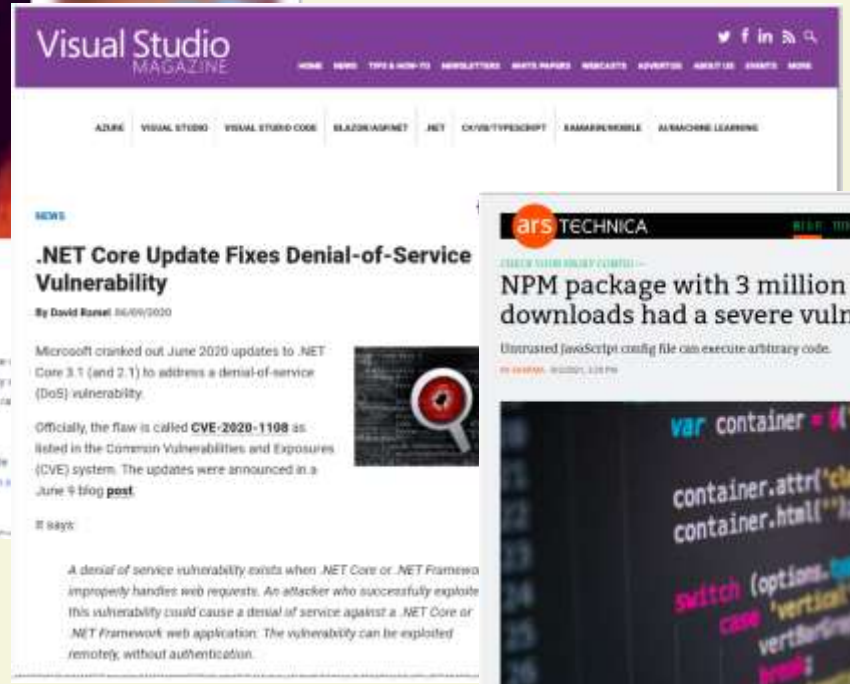
Zero-bugs!

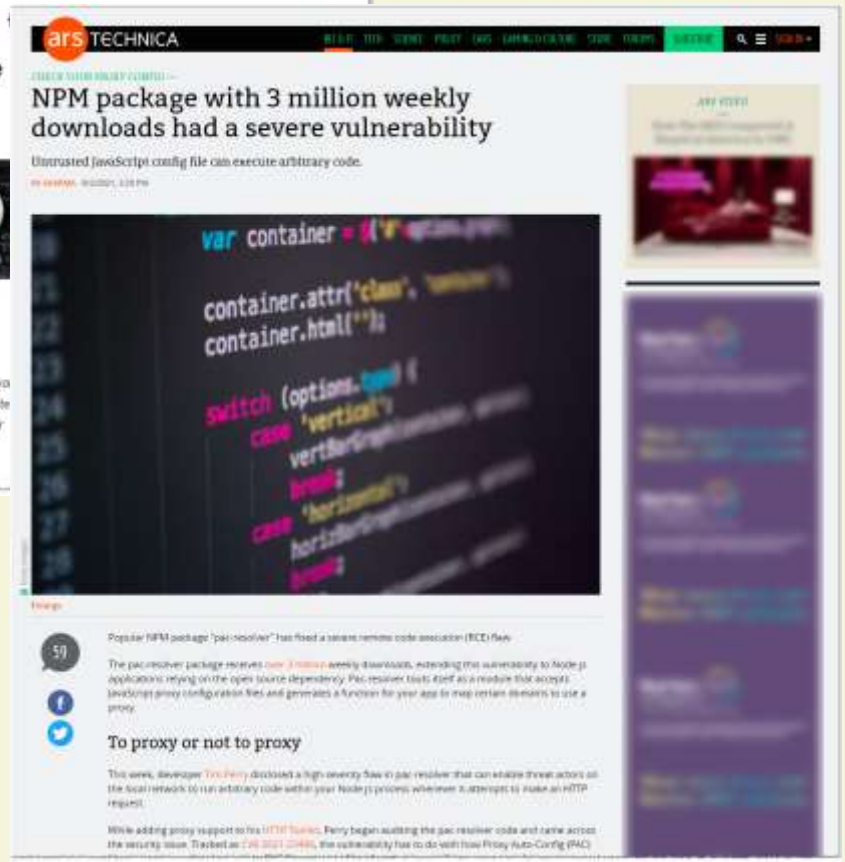
No known security issues in code!

No known security issues in infrastructure!


The Hacker News
Extremely Critical Log4J Vulnerability Leaves Much of the Internet at Risk
LOG4J

...except...


Visual Studio
MAGAZINE
.NET Core Update Fixes Denial-of-Service Vulnerability
Microsoft cranked out June 2020 updates to .NET Core 3.1 (and 2.1) to address a denial-of-service (DoS) vulnerability.
Officially, the flaw is called CVE-2020-1108 as listed in the Common Vulnerabilities and Exposures (CVE) system. The updates were announced in a June 9 blog post.
It says:
A denial of service vulnerability exists when .NET Core or .NET Framework improperly handles web requests. An attacker who successfully exploited this vulnerability could cause a denial of service against a .NET Core or .NET Framework web application. The vulnerability can be exploited remotely, without authentication.


ars TECHNICA
NPM package with 3 million weekly downloads had a severe vulnerability
Untrusted JavaScript config file can execute arbitrary code.
var container
container.attr('class',
container.html("");
switch (options.
case 'vertical':
case 'horizontal':
To proxy or not to proxy

High-level process

triggers instructs fix code deploy

Finding code

Which code matches production?
- `master` `main` `release/*`
- `v*` tags

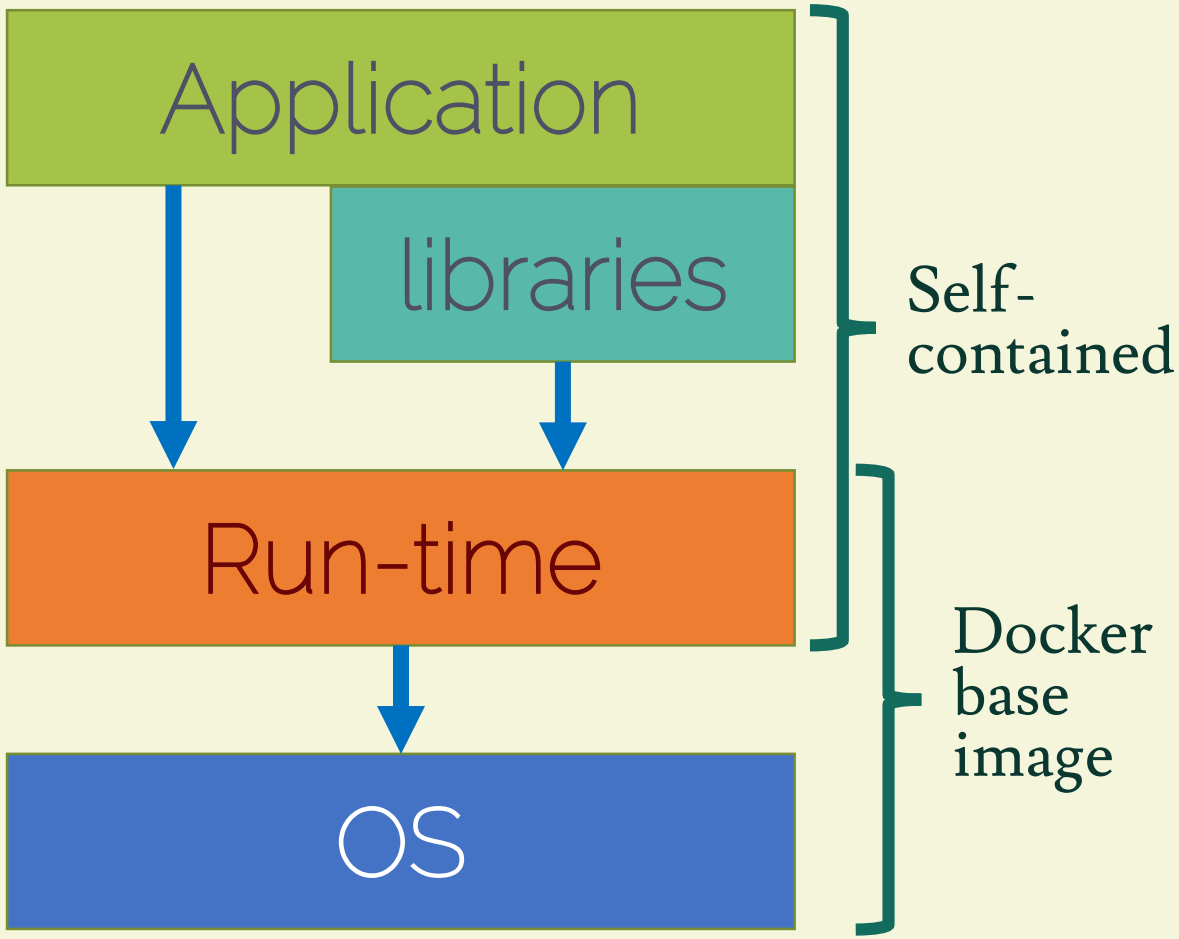
Multiple production branches
- `release/*` and `hotfix/*`

Untagged releases

SCA tools pipeline-bound
- Rarely built code
- Pipeline does not work anymore

Vulnerability may affect

- Application stack
  - Container images
  - Virtual Machine images
- Application itself
  - Application code
  - Libraries
    - Internal
    - 3rd party
  - Self-contained run-time



Tools to Identify Vulnerabilities

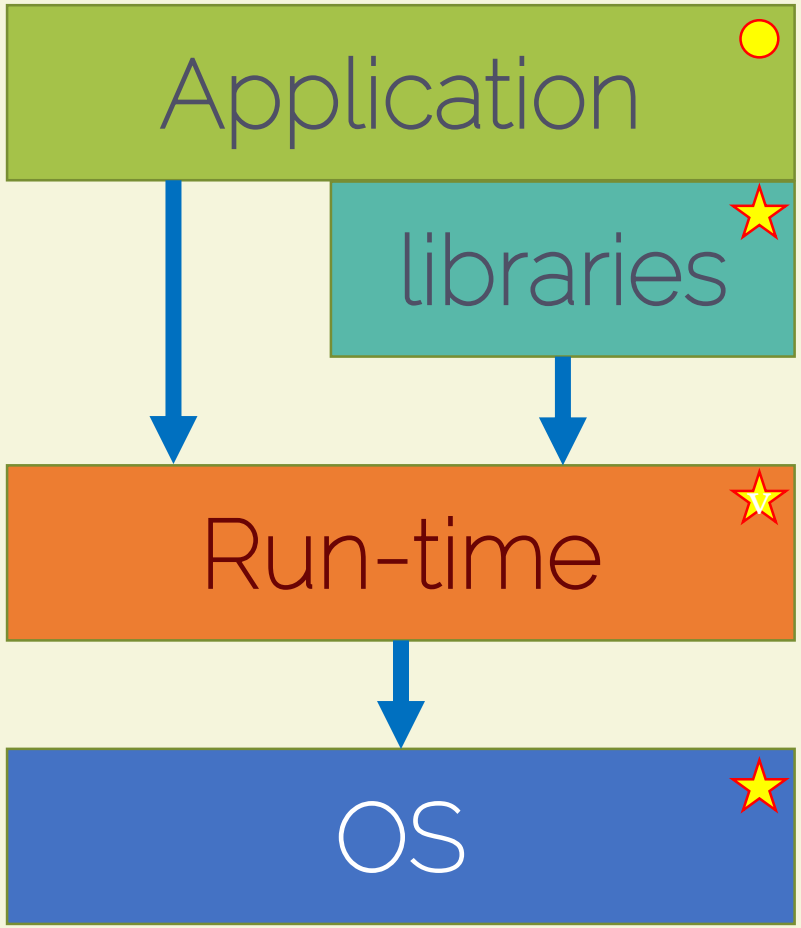
- ● Static Application Security Testing (SAST)
- ★ Software Composition Analysis (SCA)

Commercial

Synopsys Black Duck, Snyk, WhiteSource Bolt, Sonatype Nexus Platform, JFrog Xray

OSS

npm audit, OWASP Dependency Check, GitHub dependabot, Trivy



Fixing code

Scan multiple repositories
Patch code
Regression test

} Can be automated?

Trivial case

Mono-repo

Unified pipeline

Image: clutter by Ashton

Everyone else

Many teams

Many repos

My company has 3,000 repos across 100 teams, storing over 13 million lines of code, and using 2,800 pipelines

A single vulnerability may affect 10s teams and 100s of repos

Image: The Crowd For DMB 1 by Moses

Redeploy. Every. Day.

- Simplest pattern
- Once automated patching is in place
- Zero-downtime deploy in place
- Consider pipeline resources

Image: the gerbil wheel pose by dbgg1979

Setup a Code Metabase

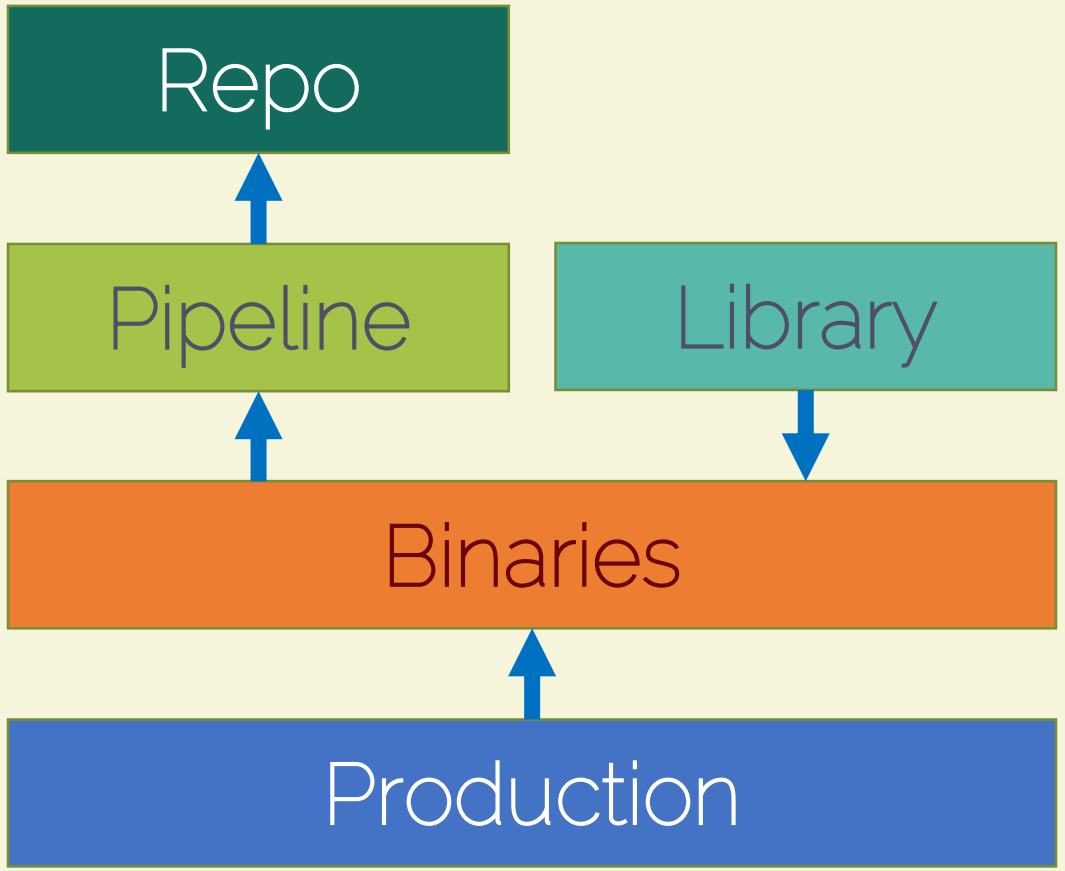
Reverse indexes

Library → Binaries [SCA tool]

O.S. API → Binaries [SAST tool]

Binary → Pipelines [artifact store]

Pipeline → Repo(s) [pipeline tool]



Expedite pipelines

- Separation of Duties
  - Regulation / audit requirement
  - Slows 0-day patching
- Tightly controlled usage
  - Automated checks
    - Single commit with limited churn
- Additional approvers for quick turnaround

Image courtesy of SpaceX

Breadth of change

Fix impacting many systems at once

- Hundreds of concurrent pipelines

Can your build & deploy tool auto-scale?

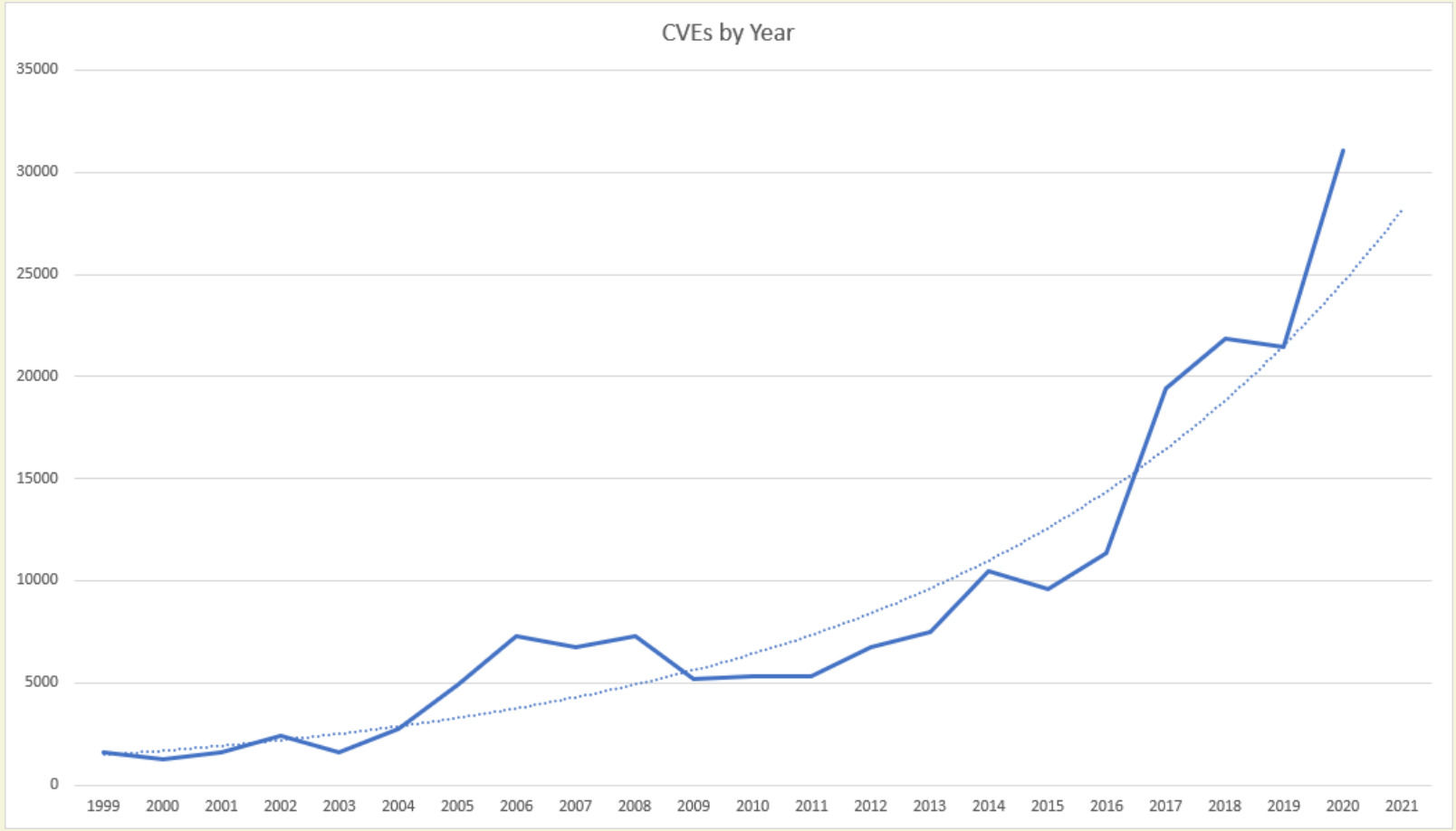
Can your approval process scale?

How fast can you rebuild a substantial portion of IT systems?

Image © Mediaset

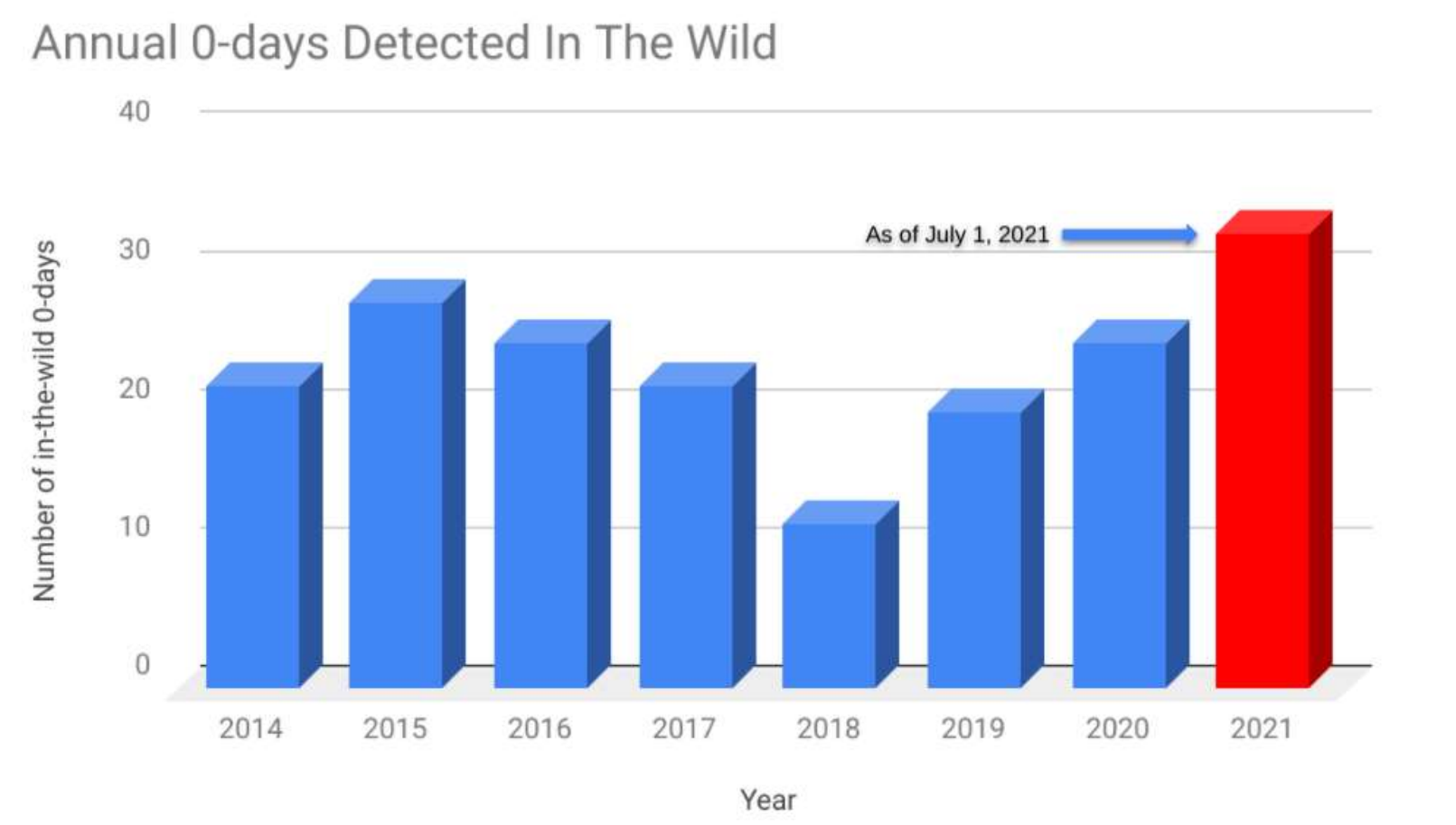
Why should I care?

Vulnerabilities over year



Source: mitre.org

Zero-days exploits are increasing



Source: Google

Dependencies

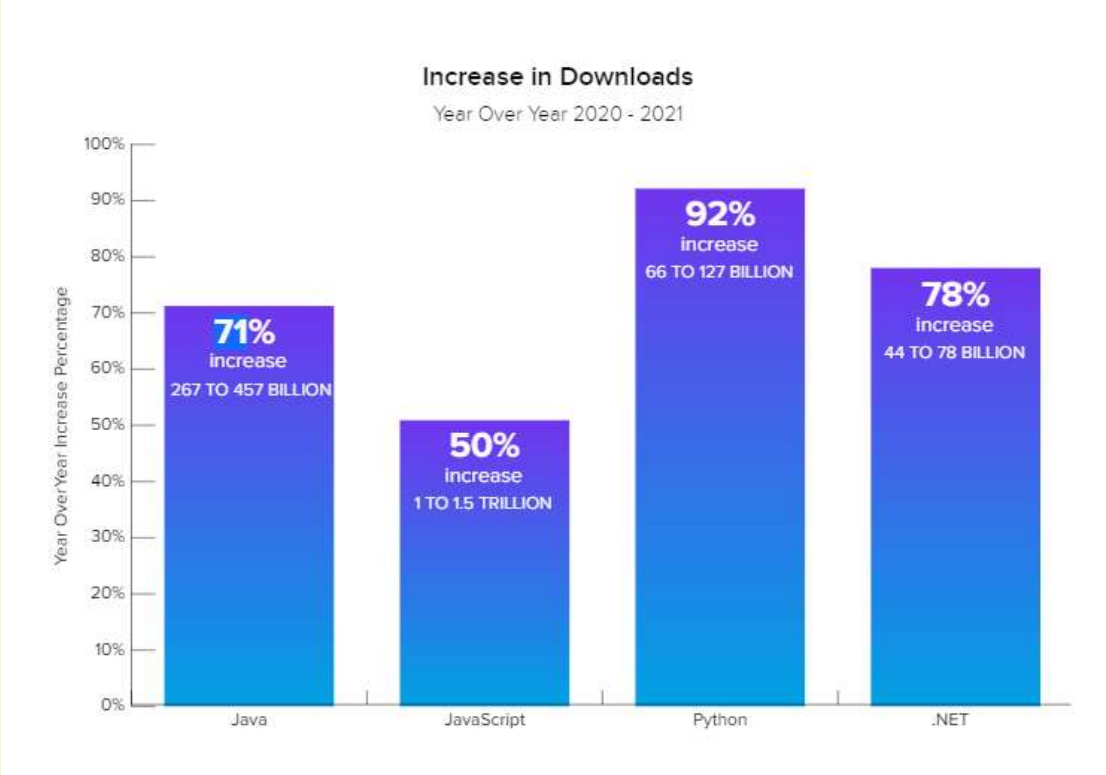
An average .NET project has 11 direct, and 76 indirect dependencies [Source: Snyk]

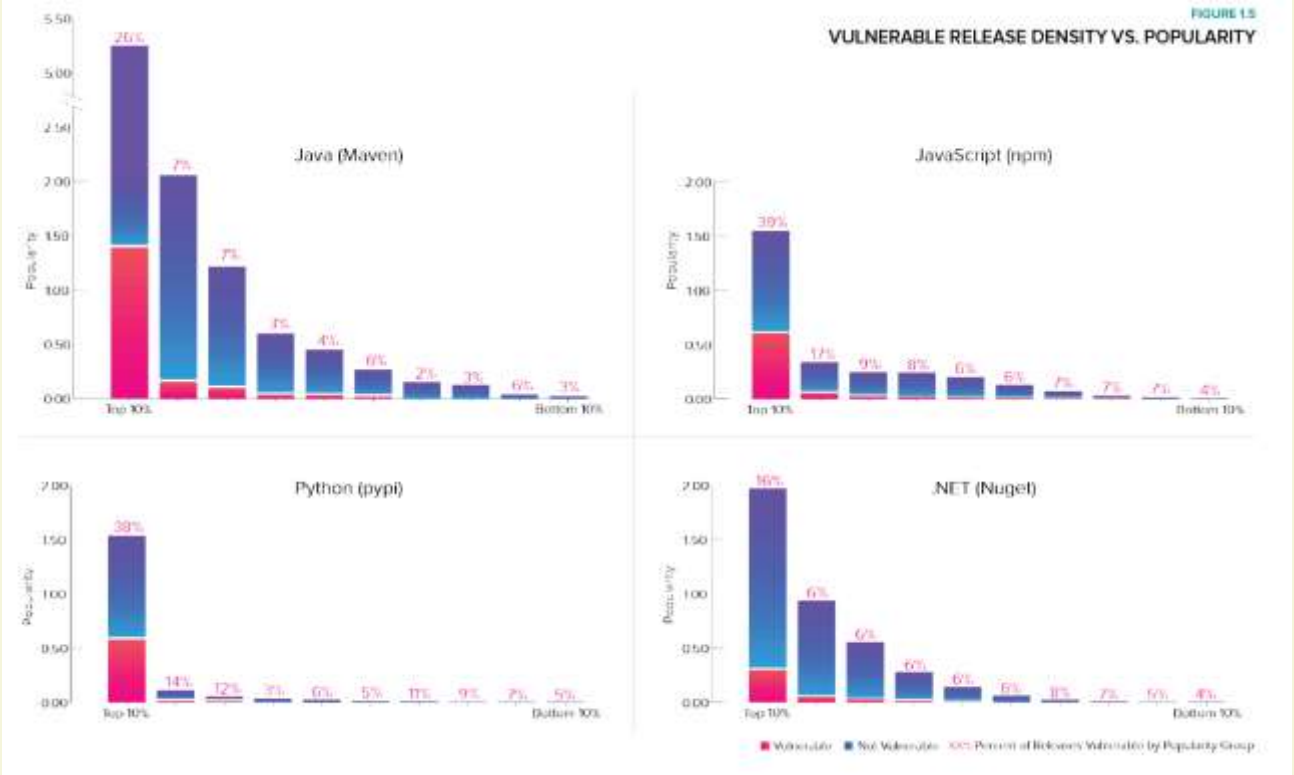
Project == nuget.org package

The average application contains **118** open-source libraries [Source: Contrast Security]

Application: Java/.NET/NodeJS

Open source dependency & vulnerability





Source: Sonatype

App Platform shift

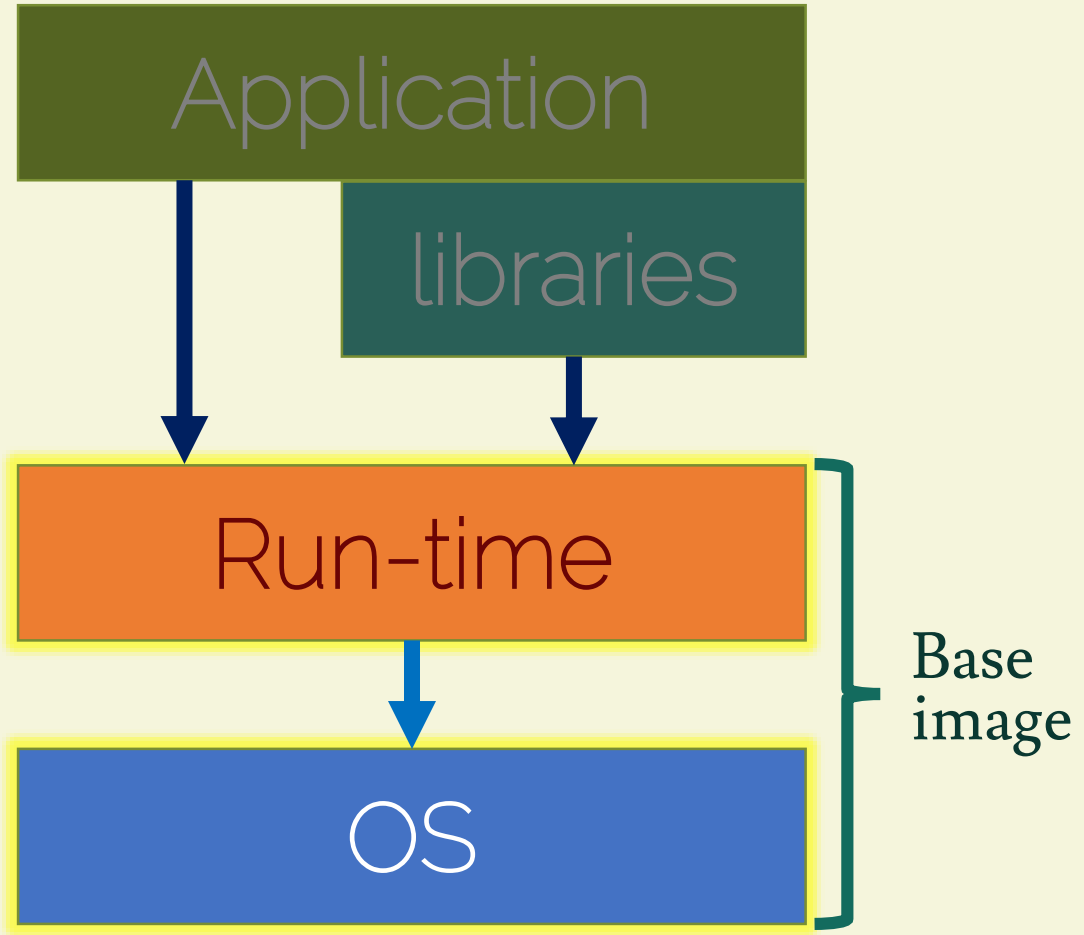
Chrome	1 month	patched after **14 days**
Node.JS	30 months (LTS) 6 months	patched every **25 days**
Go	6 months Two major releases supported.	patched every **26 days**
MongoDB	30 months	patched every **5 weeks**
.NET	3 years (LTS) 18 months	patched every **6 weeks**
Java	3 years (LTS) 6 months	patched every **12 weeks**

Base images

vmdk, VHD, VDI, OVA, ...

AMI , VHD

Docker, OCI, ACI, ...



Security SLA

Mean Time to Patch

- Single component
- Multiple components at once!
- In Production

Consequences

Technical Debt

«describes the consequences of software development **actions** that intentionally or unintentionally prioritize client value and/or project constraints such as delivery deadlines, over more technical implementation and design considerations.»

Holvitie J., Licorish S.A., et al. - *Technical debt and agile software development practices and processes* – Information and Software Technology, iss. 96 (2018) p.142

Image by ThoBel-0043

Technical Inflation

Unintended reduction in value of a software product over time, independent of source code changes.

Depreciation does not capture two elements:

- Unintentionality
- Value can be restored

Image source: Max Pixel

1974

Continuing Change law

«A[n E-type] system must be **continually adapted** or it becomes progressively less satisfactory.»

Image source: WikiMedia

Restoring Value

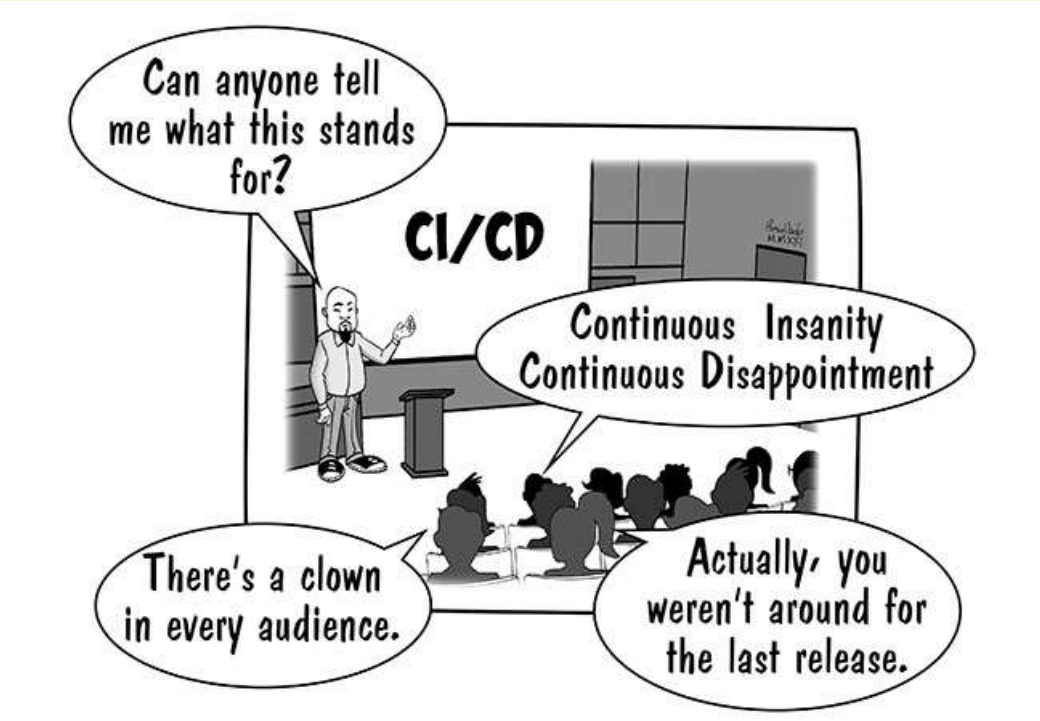
At most two platform versions

Zero-(security-)issues policy

Expedite pipelines

Image by Marek Ślusarczyk

Act!



Cartoons by: ROELBOB

Change


INSTEAD OF RISKING ANYTHING NEW, LET'S PLAY IT SAFE BY CONTINUING OUR SLOW DECLINE INTO OBSOLESCENCE.
TOM FISH BURNE
© marketoonist.com

Never forget about consequences

Image by Lionel Allorge

Kudos to Sponsors

Seguici su Telegram: https://t.me/ugidotnet

Thank you!

@giulio_vian

giuliovdev@hotmail.com

Slides that follow list bibliographic references

References (1/4)

https://www.sonatype.com/resources/state-of-the-software-supply-chain-2021

https://blog.chromium.org/2021/03/speeding-up-release-cycle.html

https://nodejs.org/en/about/releases/

https://chromium.googlesource.com/chromium/src/+/refs/heads/main/docs/process/release_cycle.md

https://support.google.com/chrome/a/answer/6220366

https://dotnet.microsoft.com/en-us/platform/support/policy/dotnet-core

https://docs.fedoraproject.org/en-US/releases/lifecycle/

https://www.oracle.com/java/technologies/java-se-support-roadmap.html

https://kubernetes.io/releases/release/

https://www.mongodb.com/support-policy/software

References (2/4)

https://heartbleed.com/

Why Every Business Is a Software Business — Watts S. Humphrey Informit, Feb 22, 2002
http://www.informit.com/articles/article.aspx?p=25491

https://en.wikipedia.org/wiki/Watts_Humphrey

https://www.sonatype.com/resources/state-of-the-software-supply-chain-2021

https://www.shopify.com/enterprise/global-ecommerce-statistics

https://blog.cloudflare.com/popular-domains-year-in-review-2021/

https://radar.cloudflare.com/year-in-review-2021

https://snyk.io/blog/net-open-source-security-insights/

https://www.contrastsecurity.com/the-state-of-the-oss-report-2021

https://octoverse.github.com/static/github-octoverse-2020-security-report.pdf

References (3/4)

https://www.soa.org/globalassets/assets/files/resources/research-report/2020/quantification-cyber-risk.pdf

https://www.soa.org/globalassets/assets/files/resources/research-report/2020/exposure-measures-cyber-insurance.pdf

https://www.csis.org/programs/strategic-technologies-program/significant-cyber-incidents

https://www.verizon.com/business/resources/reports/dbir/

https://www.accenture.com/us-en/insights/security/cost-cybercrime-study

https://www.ibm.com/security/data-breach

https://libraries.io/data

https://go.snyk.io/SoOSS-Report-2020.html

https://www.amazon.co.uk/Accelerate-Software-Performing-Technology-Organizations/dp/1942788339

References (4/4)

https://www.sciencedirect.com/science/article/abs/pii/0164121279900220

https://daverupert.com/2020/11/technical-debt-as-a-lack-of-understanding/

https://wiki.owasp.org/images/b/bd/Software_Composition_Analysis_OWASP_Stammtisch_-_Stanislav_Sivak.pdf

https://googleprojectzero.blogspot.com/

https://blog.google/threat-analysis-group/how-we-protect-users-0-day-attacks/

https://github.com/nodejs/node/blob/master/doc/changelogs/CHANGELOG_V14.md

https://dotnet.microsoft.com/en-us/download/dotnet/3.1

https://docs.mongodb.com/upcoming/release-notes/5.0/

https://itrevolution.com/the-three-ways-principles-underpinning-devops/

https://www.devsecops.org/