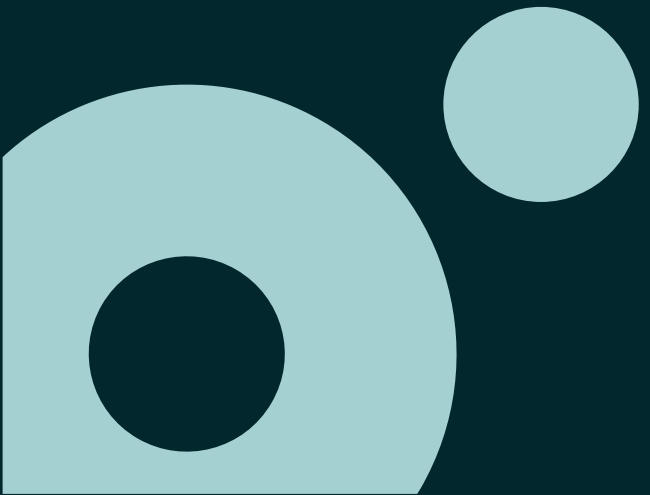
DevOps Fundamentals

Secure DevOps



Agenda

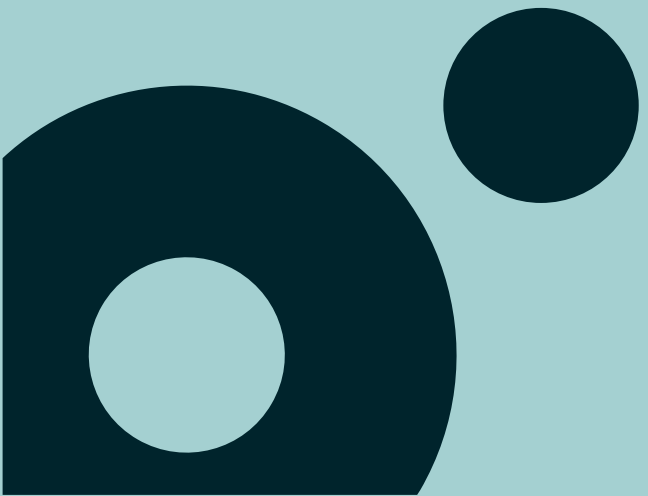
Why Secure DevOps?

Secure DevOps Practices

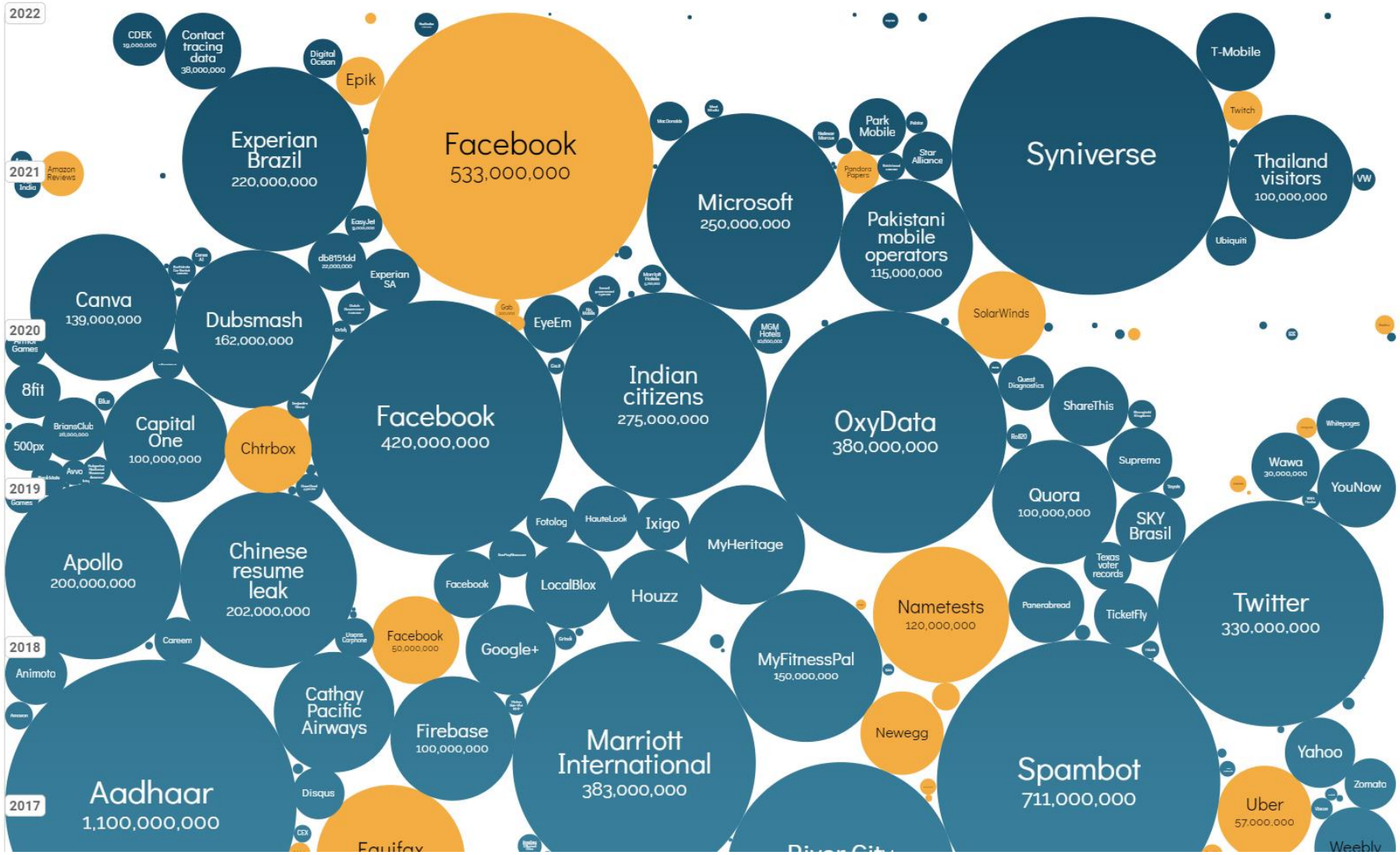
GitHub Advanced Security

moOngy.

Why Secure DevOps?



World's Biggest Data Breaches & Hacks



[World's Biggest Data Breaches & Hacks]

moOngy.

Top 15 Cyber Threats

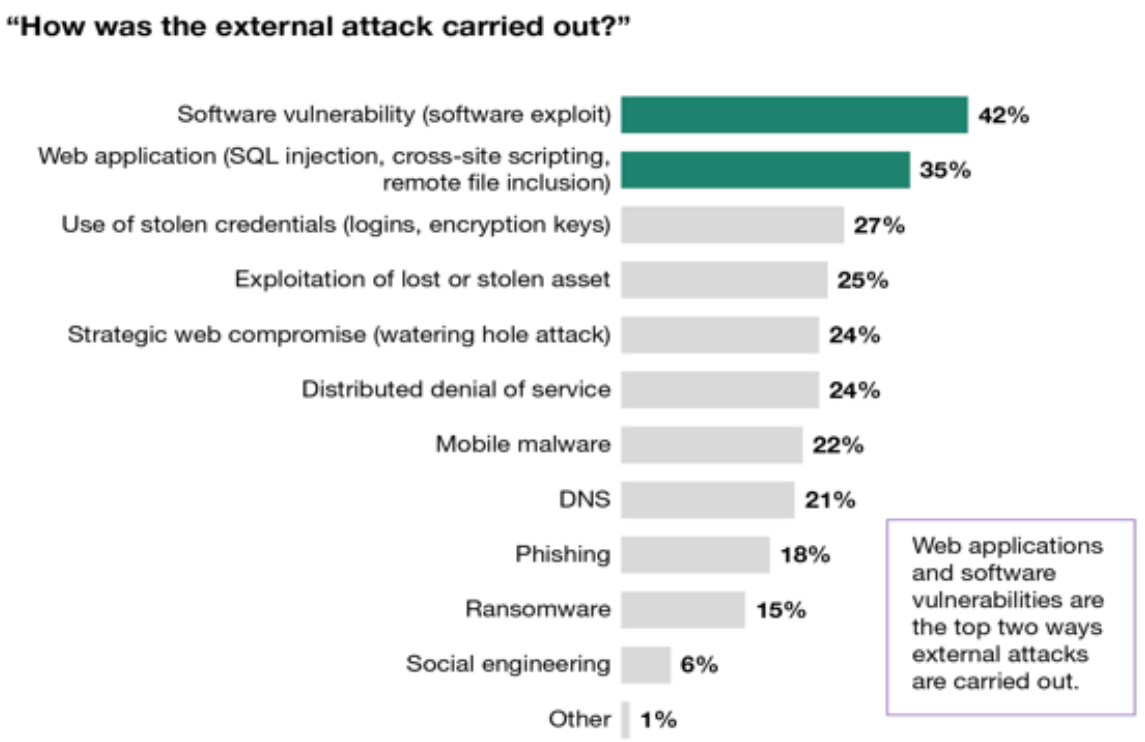
Application as attack vector

Applications remains the most common attack vector

Implementing security means be secure in several layers to make it harder to be breached

Crucial to understand and control all attack vectors

“You are only as secure as your weakest link”



Assume Breach

"Fundamentally, if somebody wants to get in, they're getting in…accept that. What we tell clients is: Number one, you're in the fight, whether you thought you were or not. Number two, you almost certainly are penetrated."

Michael Hayden, CIA

"There are only two types of companies: those that have been hacked, and those that will be."

Robert Mueller, FBI Director

You need to change your mindset (and your tools) to practice this approach.

First, you need to clearly understand your security vulnerabilities!

Impact caused by a security breach

Costs and efforts related with response and notification

Lost employee productivity

Lawsuits and settlements

Regulatory fines and response

Cost of fixing infrastructure

Brand recovery costs & liabilities

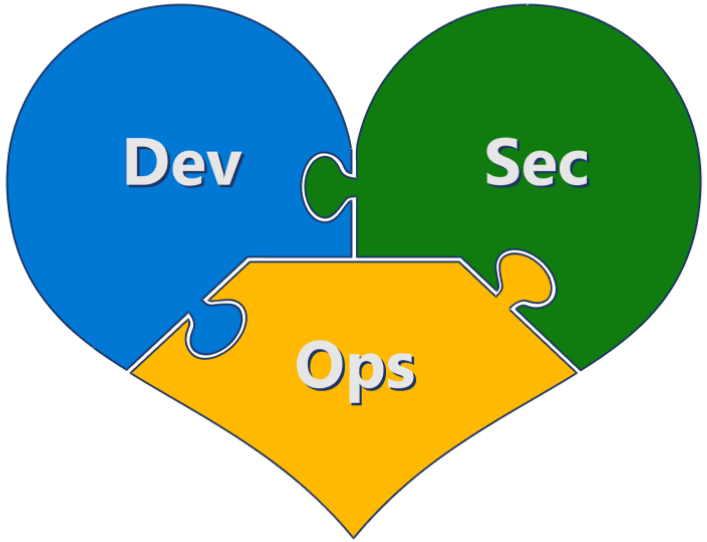
What is security?

"Security is application of technologies, processes, and controls to protect systems, networks, programs, devices, and data from unauthorized access or criminal use."

Security provides **confidentiality**, **integrity**, and **availability** assurances against deliberate attacks and abuse of valuable data and systems.

What is Secure DevOps?

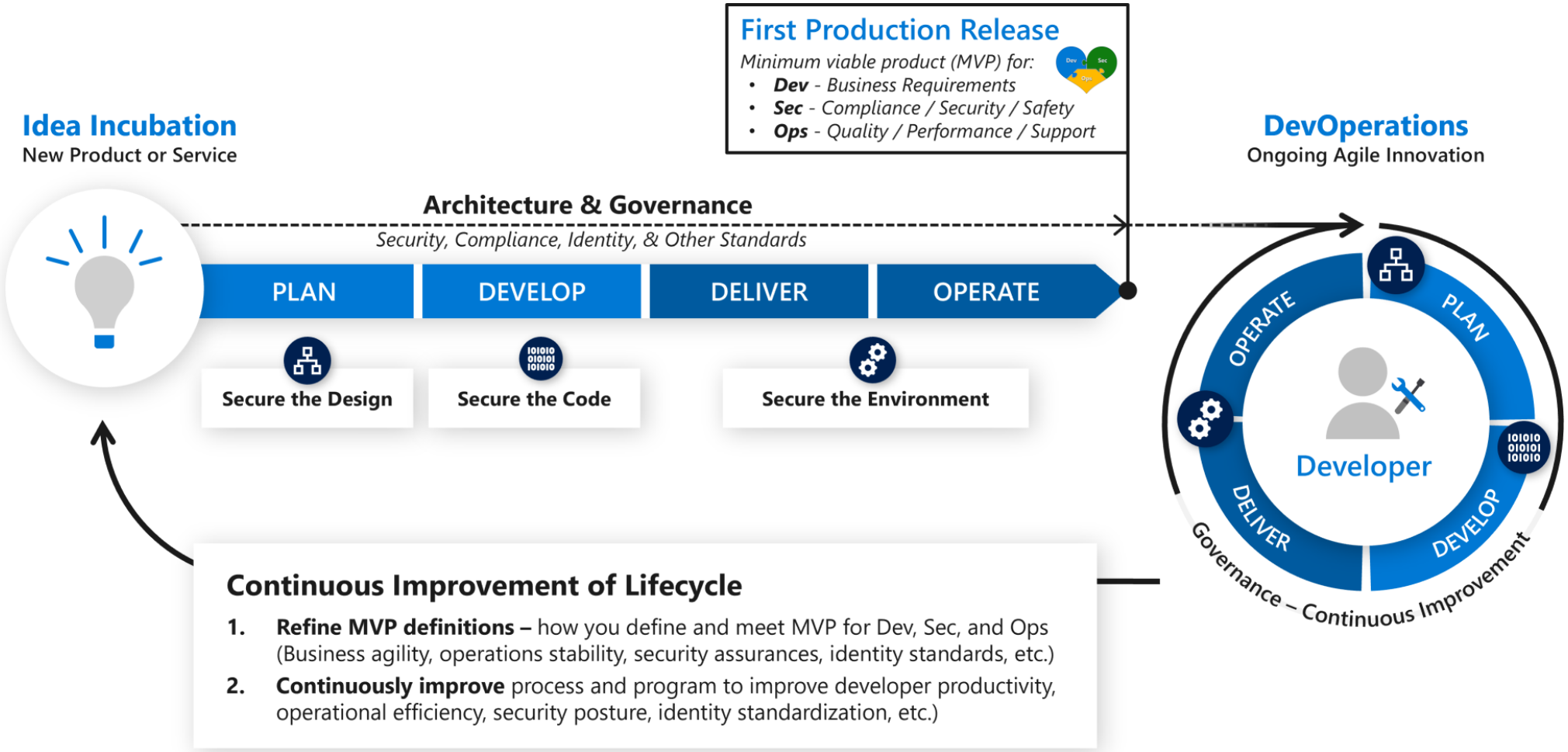
Responsive to Needs
Meets business and customer requirements for market relevance



Safe and Secure
Provides confidentiality, integrity, & availability + regulatory compliance

Quality and Performance
meets the quality, speed, scalability, reliability, and other expectations

What is Secure DevOps?



What is Secure DevOps?

Practice that assures security is a crucial part of software delivery lifecycle

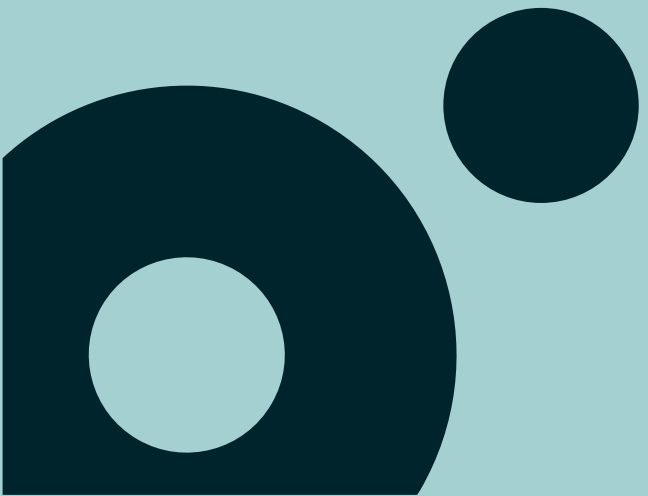
Secure DevOps have 3 main principles

- Security culture
- Secure software delivery
- Secure infrastructure

Secure DevOps have 3 main practices to implement its principles

- Shifting Left
- Continuous improvement
- Automation

Security Culture



Security Culture

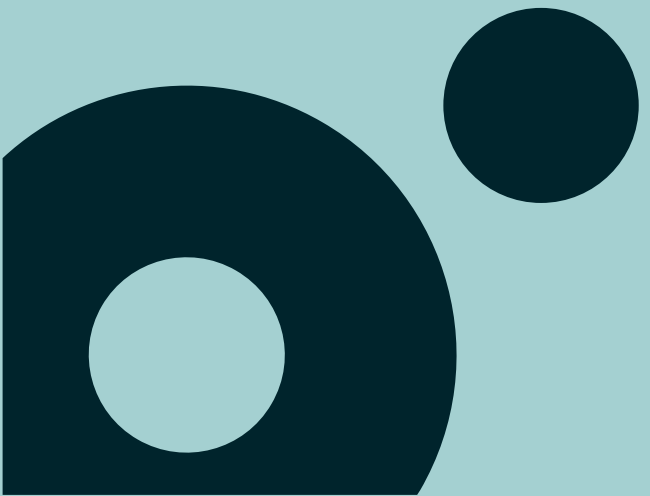
"Security culture refers to the set of values, shared by everyone in an organization, that determine how people are expected to think about and approach security."

CPNI UK (Centre for the Protection of National Infrastructure)

How to promote security culture?

- Promote awareness and training
- Build your principles
- Put principles in practice

Secure Software Delivery



Secure Software Delivery

Makes mandatory that everyone learn about basic security practices

3 main goals

- Reduce the number of vulnerabilities
- Reduce the severity of vulnerabilities you miss
- Prepare for potential attacks

SDL practices are implementing through 3 types of learning

- Learning by doing
- Learning by simulating
- Learning by playing

SDL Practices

Provide security training for all your organization (developers, operations, …)

Define security requirements with metrics & compliance reporting

Perform threat modeling (OWASP at least)

Define and use cryptography standards

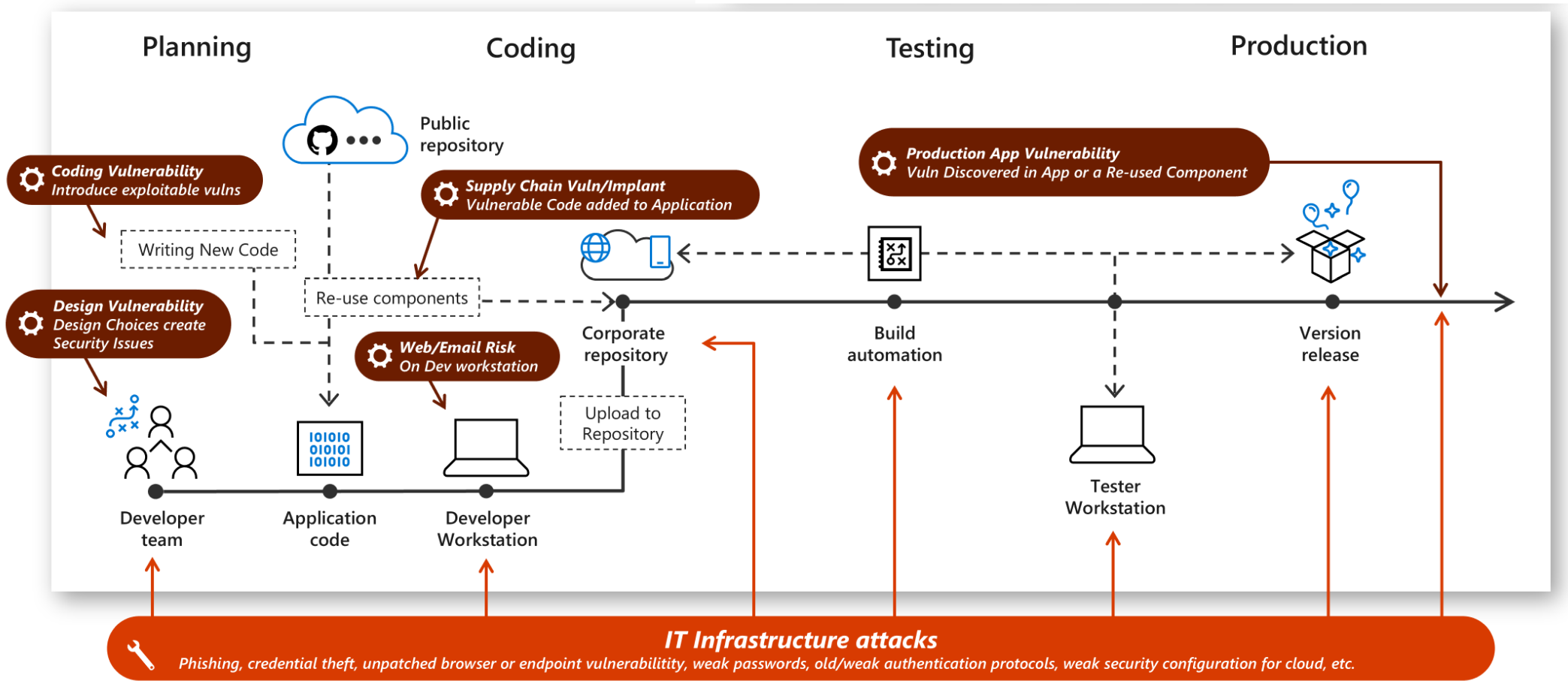
Manage security risk of using 3rd party components

Perform static analysis security testing (development time), dynamic analysis security testing (testing time) and penetration testing (production time)

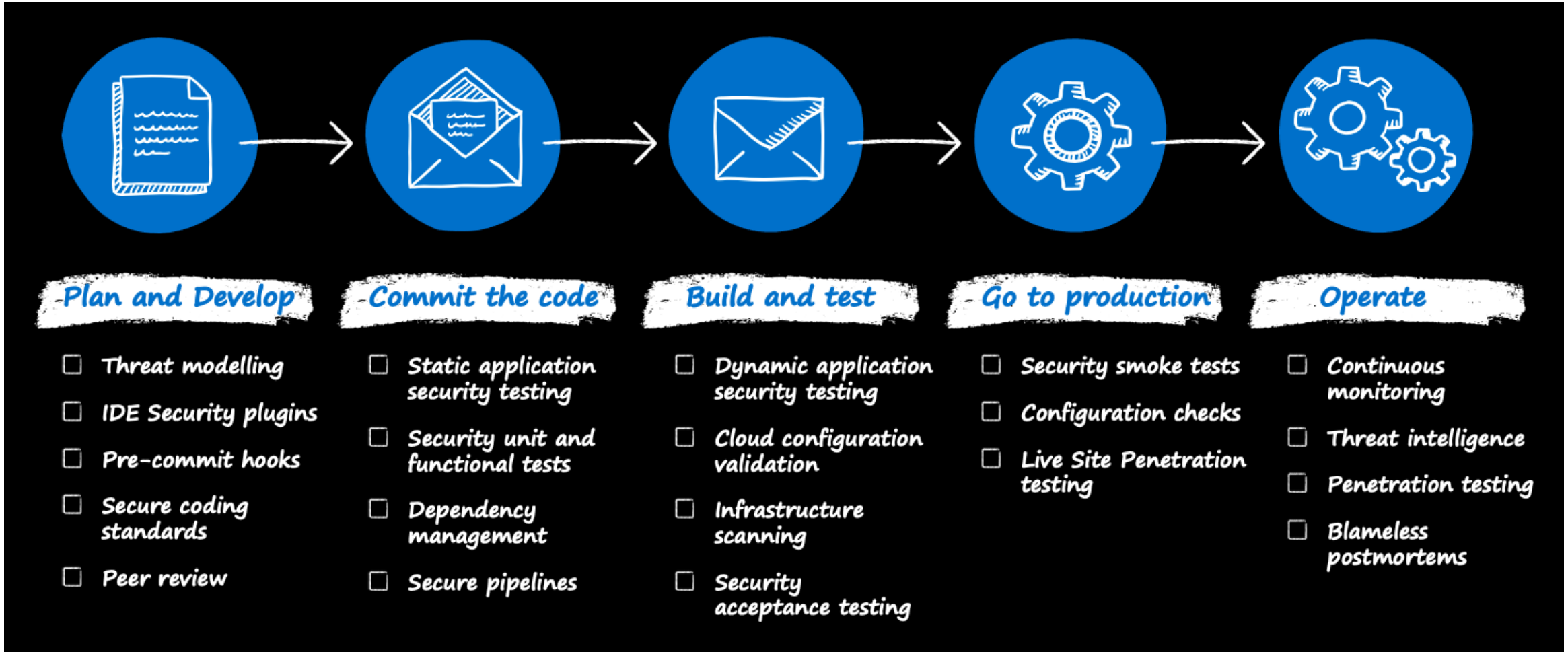
Establish standard incident response process

SDL: Attacker Opportunities & Lateral Moves

Attacker thinks in graph, defender think in lists



SDL Practices



SDL Practices: Infra as Code

Plays a very important role on recoverability

If your infra suffers an attack and you need to rebuild (or build on another place) IaC allow to be faster and secure

On organization using cloud, Infra as Code is the way to implement Disaster Recovery

SDL Practices: Static Application Security Testing (SAST)

Improve code security and quality on an easy and cost-effective way

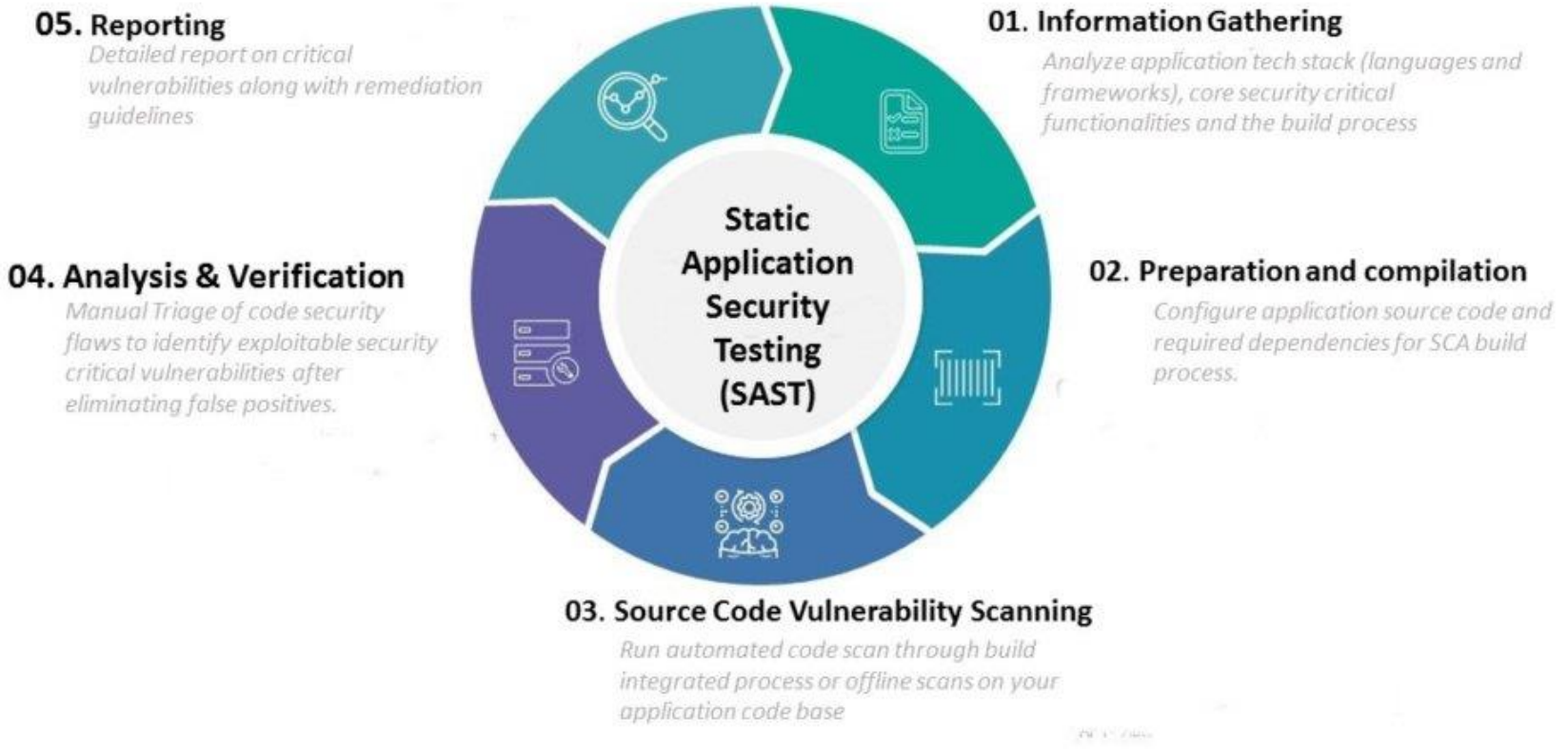
Makes an analysis on your source code and return insights about security, performance, maintainability

Fully automated and can be shift-left for developers IDE

Runs to answer by with the question “Is the code secure?”

- Is it vulnerable to injections (like SQL)?
- Does it use any weak encryption algorithms?
- Are cookies used with the right flags?

SDL Practices: Static Application Security Testing (SAST)



SDL Practices: SAST Tooling

Semmle/CodeQL

sonarcloud

snyk

klocwork

Spectral

Checkmarx

https://owasp.org/www-community/Source_Code_Analysis_Tools

SDL Practices: Static Application Security Testing (SAST)

To improve security all around a CVE (Common Vulnerability and Exposures) database was created

Defined on top of CWE (Common Weakness Enumeration) definition, being an effective instance of 1+ CWE exploit

On this database, each CVE found on open-source software are published with vulnerability description and how to fix

For each CVE a CVSS score is calculated granting a potential risk you're exposed

CVSS Score	Qualitative Rating
0.0	None
0.1 – 3.9	Low
4.0 – 6.9	Medium
7.0 – 8.9	High
9.0 – 10.0	Critical

SDL Practices: Dynamic Application Security Testing (DAST)

Type of testing that looks for security vulnerabilities by safely exploiting a running application from the outside

This type of testing is not dependent on the framework or programming language used

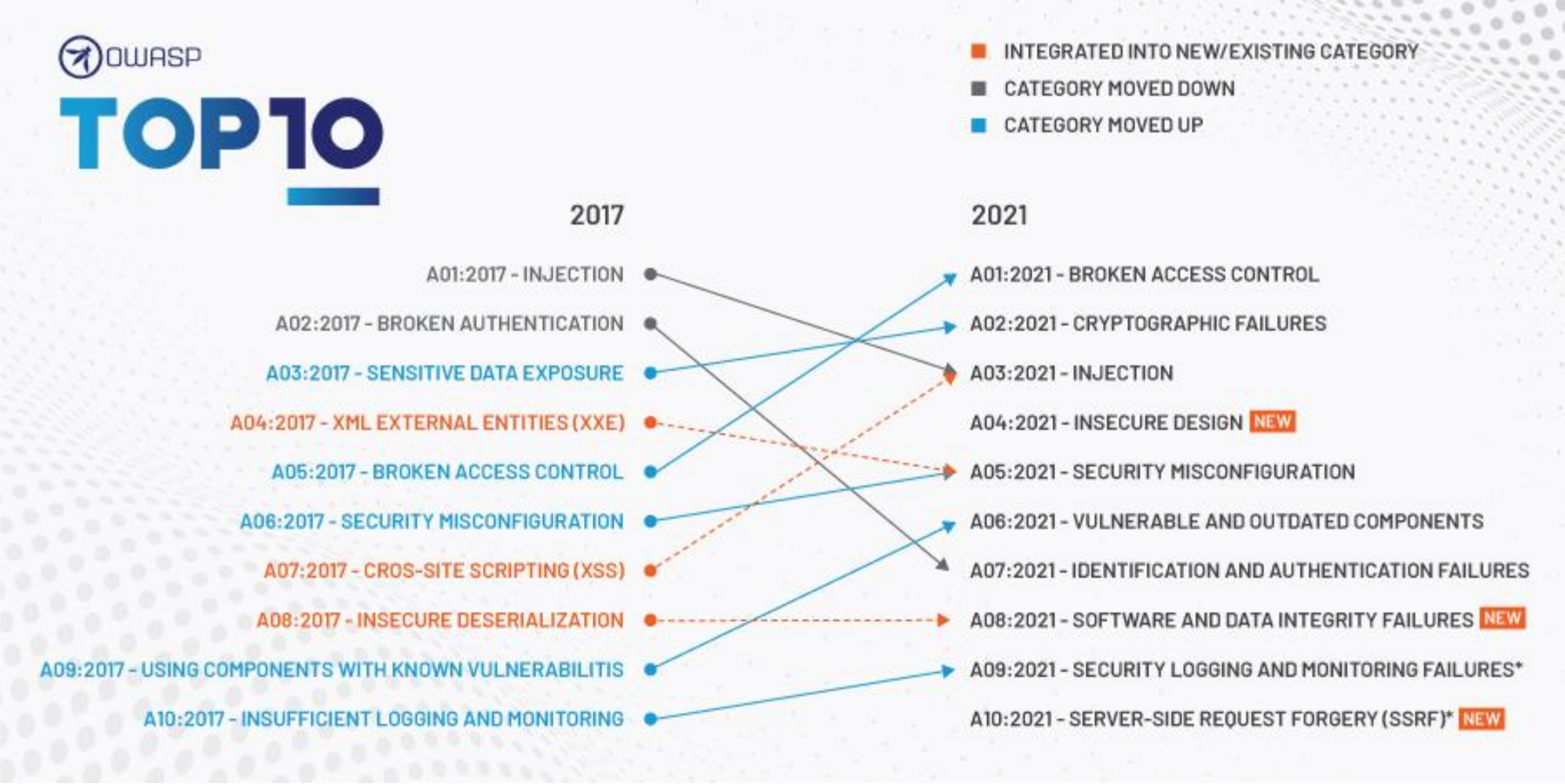
Differs from SAST because runs on top of a running application

Must of the times, a specific environment is created to run DAST tests (and then destroyed ☺)

Uses fuzzy mechanisms to test your application, like send request with body, corrupted headers, etc.

At least should run tests on your application to check OWASP Top 10 Attacks

SDL Practices: OWASP Top 10



[OWASP Top Ten Web Application Security Risks | OWASP]

SDL Practices: OWASP Top 10

If your application is not affected by OWASP Top 10 vulnerabilities, can you say your application is secure and solid?

SDL Practices: OWASP Top 10

If your application is affected by any OWASP Top 10 vulnerabilities, can you say your application is insecure?

SDL Practices: Dynamic Application Security Testing (DAST)

Ideally, and using shift-right approach, you may run it in production expecting issues to arise

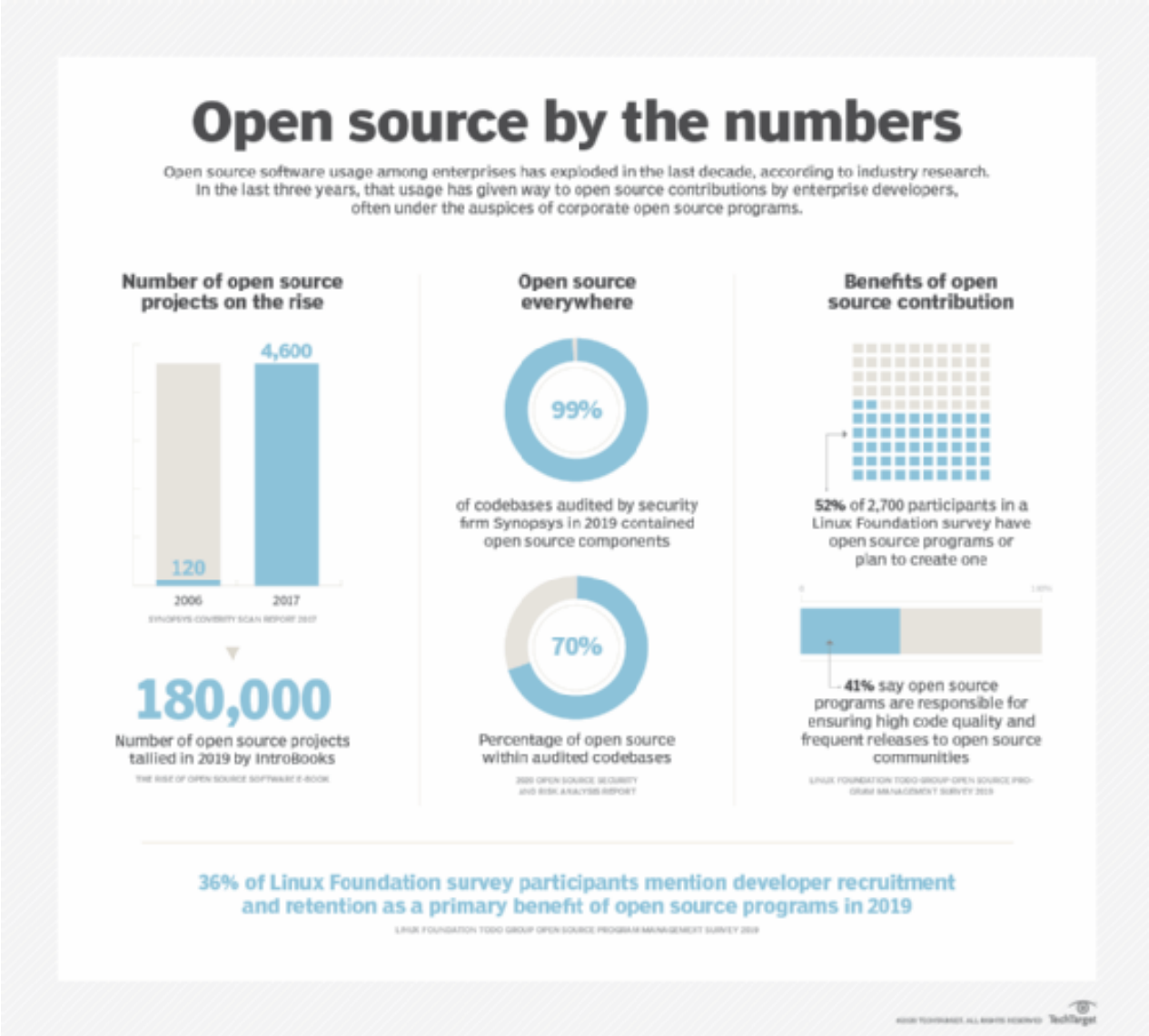
Works well together with canary/ring deployment strategies where you may test and affect only a subset of your users

Other techniques for production testing like penetration testing and chaos engineering (monkey testing)

SDL Practices: DAST Tooling

https://owasp.org/www-community/Vulnerability_Scanning_Tools

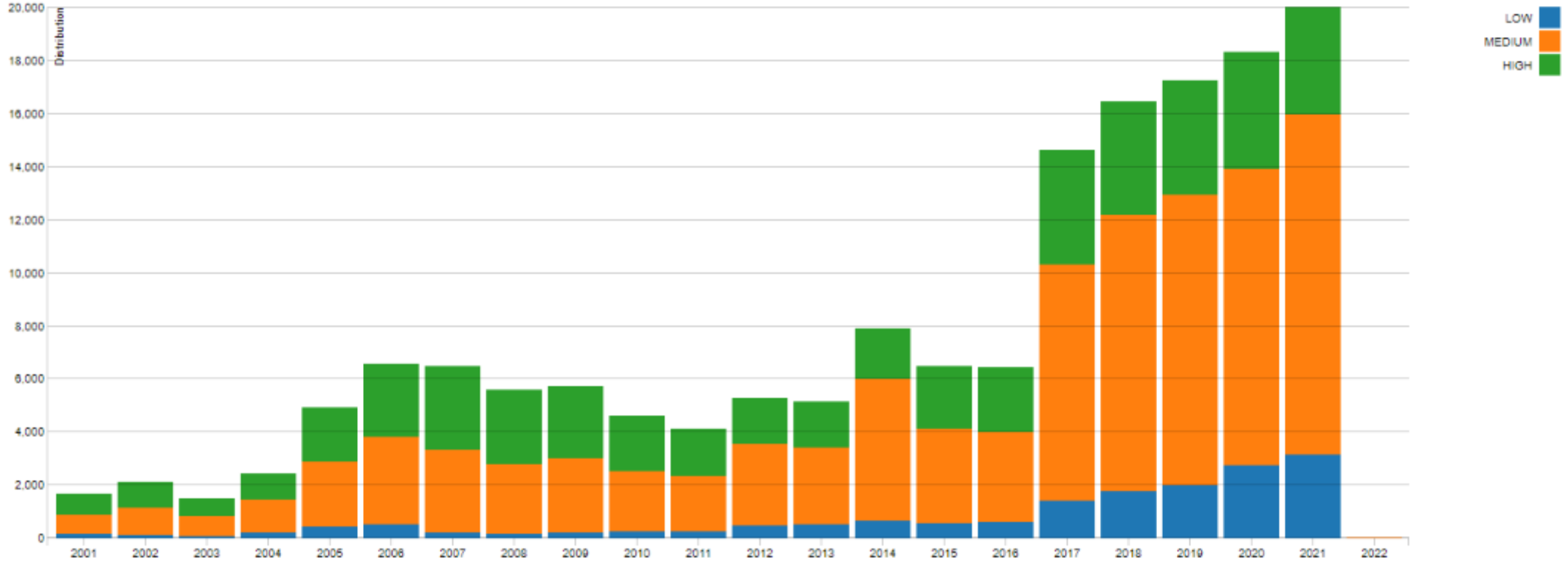
SDL Practices: Software Composition Analysis (SCA)

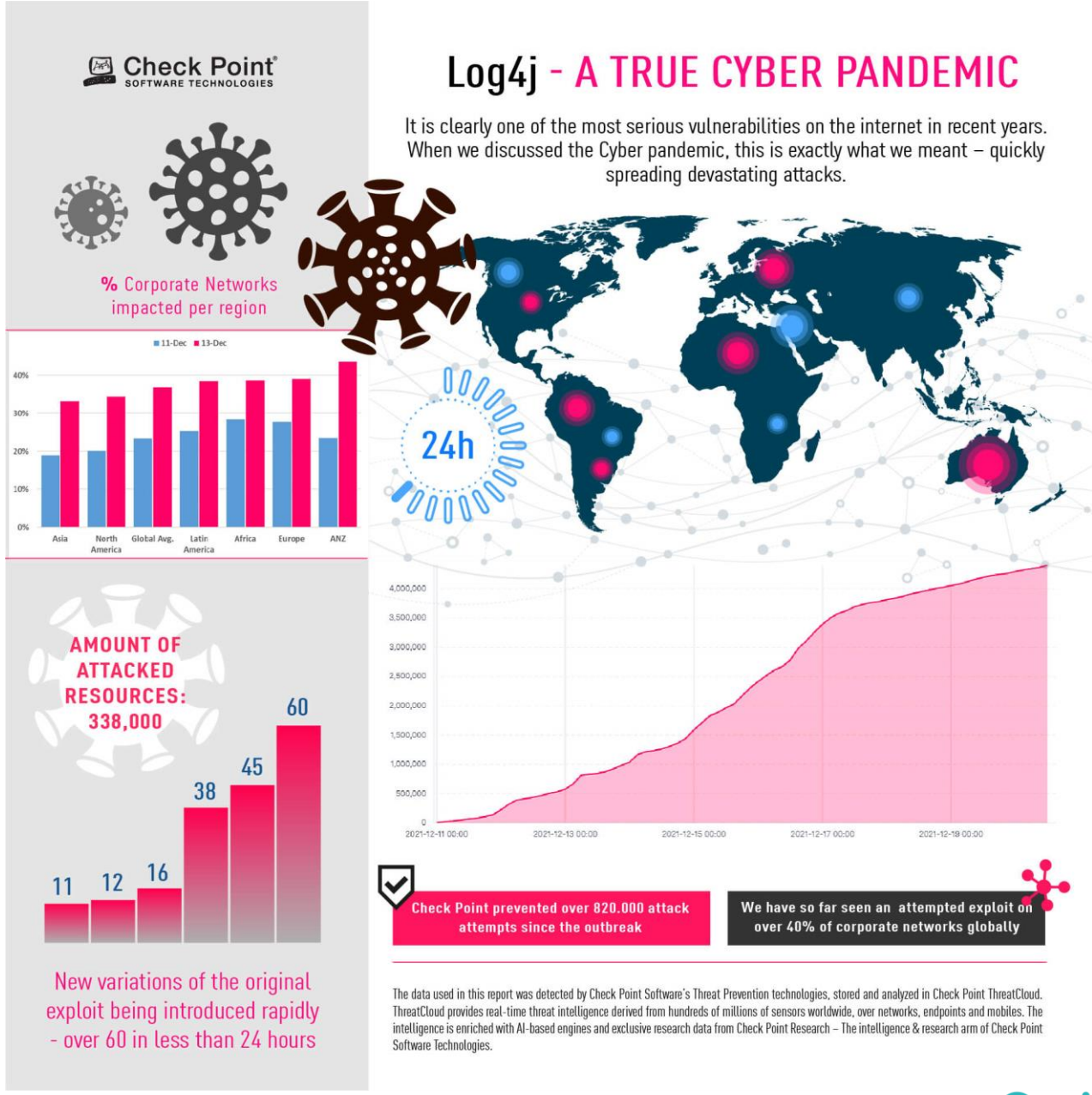


SDL Practices: Software Composition Analysis (SCA)

CVSS Severity Distribution Over Time

This visualization is a simple graph which shows the distribution of vulnerabilities by severity over time. The choice of LOW, MEDIUM and HIGH is based upon the CVSS V2 Base score. For more information on how this data was constructed please see the NVD CVSS page .




Check Point
SOFTWARE TECHNOLOGIES
% Corporate Networks impacted per region
11-Dec
13-Dec
40%
30%
20%
10%
0%
Asia
North America
Global Avg.
Latin America
Africa
Europe
ANZ
AMOUNT OF ATTACKED RESOURCES: 338,000
11
12
16
38
45
60
New variations of the original exploit being introduced rapidly - over 60 in less than 24 hours
Log4j - A TRUE CYBER PANDEMIC
It is clearly one of the most serious vulnerabilities on the internet in recent years. When we discussed the Cyber pandemic, this is exactly what we meant – quickly spreading devastating attacks.
24h
4,000,000
3,500,000
3,000,000
2,500,000
2,000,000
1,500,000
1,000,000
500,000
0
2021-12-11 00:00
2021-12-13 00:00
2021-12-15 00:00
2021-12-17 00:00
2021-12-19 00:00
Check Point prevented over 820.000 attack attempts since the outbreak
We have so far seen an attempted exploit on over 40% of corporate networks globally
The data used in this report was detected by Check Point Software's Threat Prevention technologies, stored and analyzed in Check Point ThreatCloud. ThreatCloud provides real-time threat intelligence derived from hundreds of millions of sensors worldwide, over networks, endpoints and mobiles. The intelligence is enriched with AI-based engines and exclusive research data from Check Point Research – The intelligence & research arm of Check Point Software Technologies.

SDL Practices: Software Composition Analysis (SCA)

So, Open Source is a bad and dangerous thing? Of course not!

But you need to use it careful and mostly you need to clearly know what are you using!

Constantly run a scan on your dependencies is crucial to understand known vulnerabilities on your supply chain

Knowing the vulnerabilities and their severity you may define you plan to fix them

Know what you're using means knowing your dependency graph! Your direct dependency have its own dependencies. That dependencies have their own dependencies and so on…

Dependency Graph

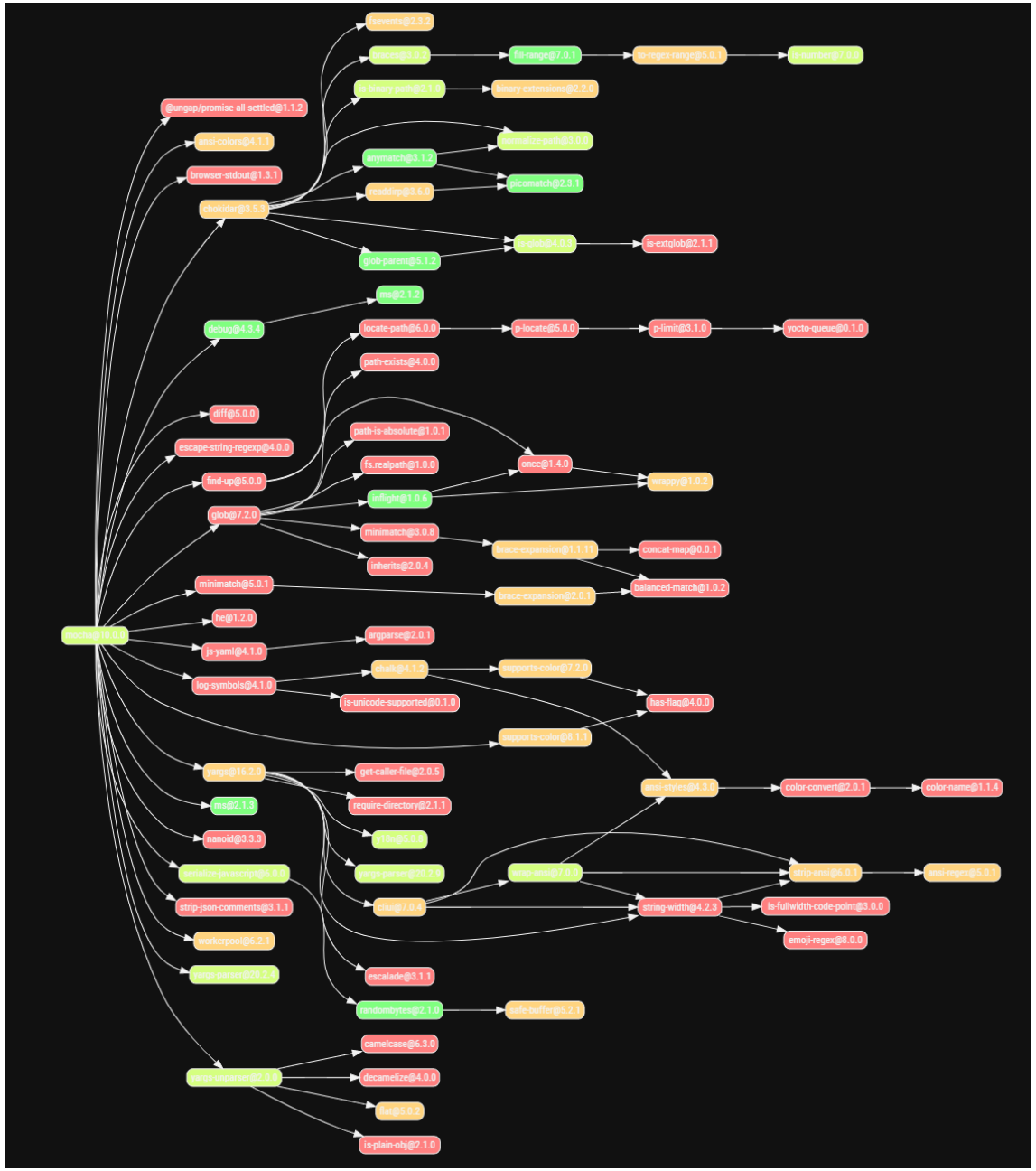
npm package Mocha

You select only one package but look to your attack surface!

Another risk is about how open source project is maintained

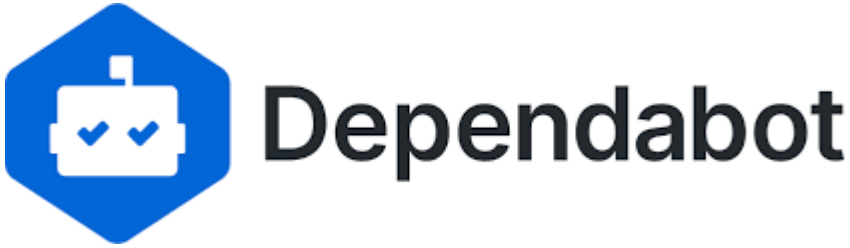
On image, red means only one maintainer.

Is a risk you may want to take, but you have to clearly know it!

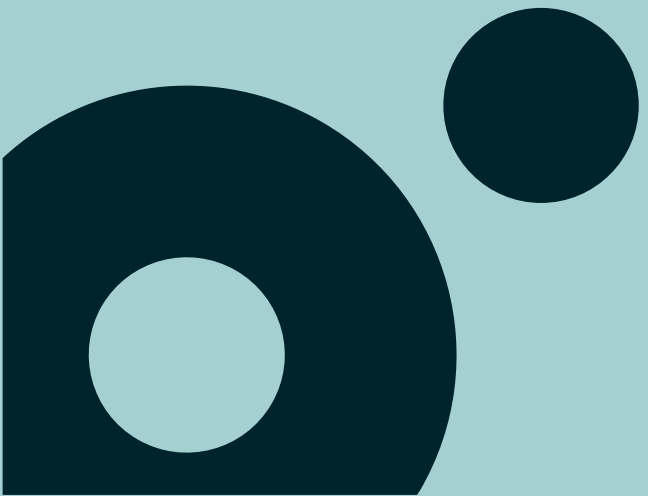


moOngy.

SDL Practices: SCA Tooling



Secure Infrastrucure

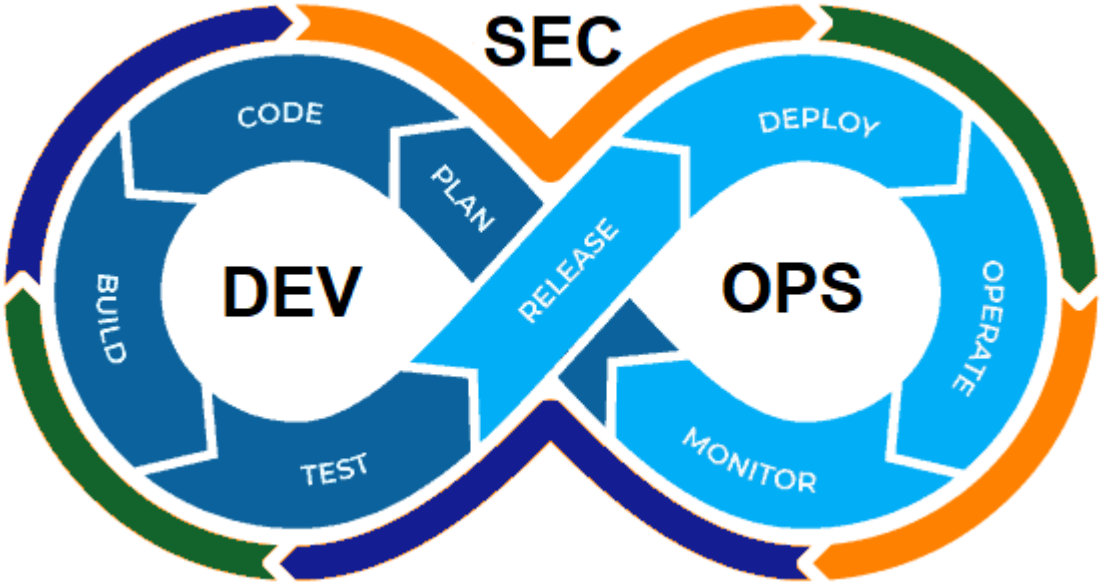


Secure Infrastructure

Secure DevOps proposal is to add security layer on top of any phase of DevOps infinite loop

Any phase uses infrastructure to execute expected outcomes

What means infrastructure in DevOps?



What means infrastructure in DevOps?

For plan, your PO, designer or architect workstations are DevOps Infra

On build phase, developers and any want producing code workstation is DevOps Infra

If you use any internal repository, is DevOps Infra

During CI/CD, your runners are DevOps Infra, even more if you don't control them directly and you're doing deploys on your infra

On testing phases, testers and even customers workstations are DevOps Infra

During operation, all your operations and infra team workstations are DevOps Infra

Oh! And your production (all) environments are DevOps Infra too! ☺

Secure Infra in DevOps

Vulnerable workstations open doors for lateral moves

Constantly update your machines

Zero trust principles, grant access to everything is needed but nothing more

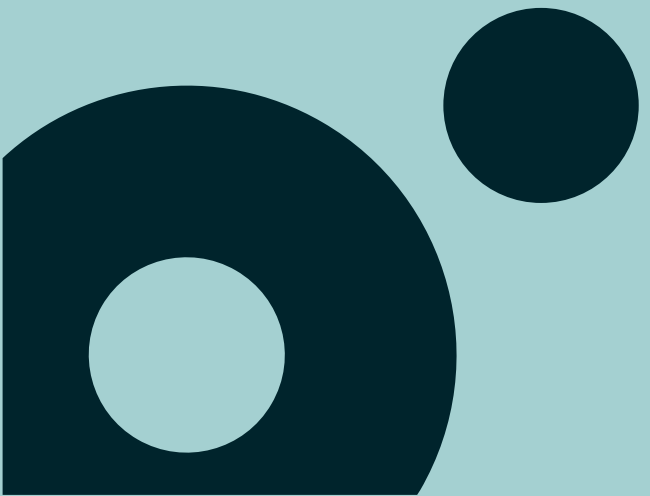
Repository access sharing credentials and adding to the repos

Reuse of credentials without rotation

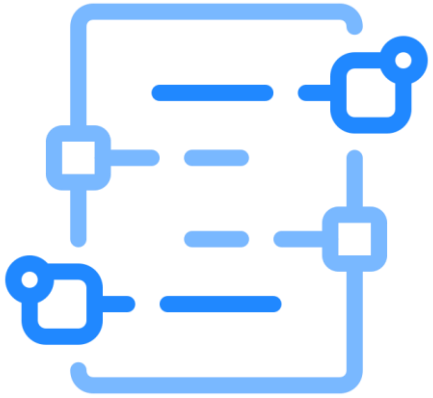
Isolate your environments to make harder to do lateral moves

Upskill your collaborators and make surprise tests for common tasks, like email phishing

GitHub Advanced Security



Secure software lifecycle with GitHub



Dependency Scanning

Code Scanning

Secrets Scanning

Dependency Scanning

Dependency graph: See the packages your project depends on, the repositories that depend on them, and any vulnerabilities detected in their dependencies.

Dependabot alerts: Get notified when there are new vulnerabilities affecting your repositories. GitHub detects and alerts users to vulnerable dependencies in public and private repos.

Dependabot security and version updates: Keep your supply chain secure and up-to-date by automatically opening pull requests that update vulnerable or out-of-date dependencies.

About supply chain security - GitHub Docs

Code Scanning

Find and fix vulnerabilities fast, before they are merged into the code base with automated CodeQL scans.

Community of top security experts produce CodeQL queries to empower every project with a world-class security team. You can even create your own custom queries.

Integrated with developer workflow for a frictionless experience and faster development, beginning with IDE integration and automate on GitHub Actions

Extensible as you may plug other SAST tools into the same developer workflow.

Secret Scanning

Identifies secrets as early as possible, since the moment they are pushed to GitHub and immediately notifies developers when they are found. Scan your entire git history

Community of secret scanning partners, for every commit made to your repository, and its full git history, we'll look for secret formats from secret scanning partners

Secret scanning watches both **public and private repos** for potential secret vulnerabilities.

Protection from exposed secrets, automatically disable or suspend secrets from 100+ service providers as soon as they are committed

Demo: GitHub Advanced Security

moOngy.
Minds on the move

Rua Sousa Martins, nº 10
1050-218 Lisboa | Portugal