INTRAFFIC
mastering mobility

DevSecOps

Een buzzword of toch een noodzakelijke stap richting Secure DevOps?

Rachid Kherrazi

10-10-2018

Even voorstelen

Rachid Kherrazi

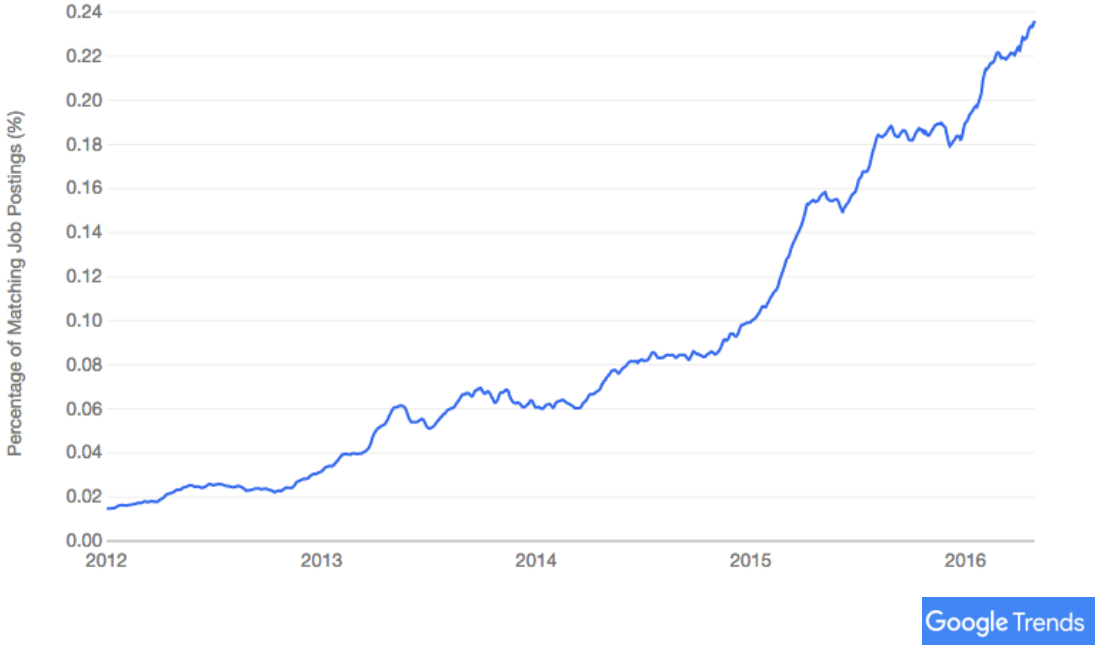
Test Manager @ InTraffic in Nieuwegein

18 jaar werkervaring bij o.a.:

PHILIPS ASML PROMEDICO@ Beter voor iedereen INTRAFFIC

Sinds 2012 lid van TestNet
Trekker van TestNet Werkgroep Model Based Testing

Wie kent DevOps?



Er is een toename in de DevOps populariteit
In aantal vacatures en zoekopdrachten!

Agenda

1. Wat is DevOps?
2. Wat zijn uitdagingen van DevOps?
3. Wat is DevSecOps?
4. Voorbeeld uit praktijk: Apotheek Informatie Systeem (AIS)
5. Mijn antwoord op de vraag: Is DevSecOps een buzzword of toch een noodzakelijke stap richting Secure DevOps?

INTRAFFIC

In het begin was er ...waterval

- Lange releasecycli
- Veel "WIPs“
- Veel silo's
- Onbuigbaar

INTRAFFIC

... toen was er Agile

- Kortere releasecycli
- Kleine leveringen
- Cross-functionele teams
- Flexibel

INTRAFFIC

En dan plotseling was Ops de bottleneck!

Uitdaging was in de samenwerking tussen Dev en Ops.

Uiteindelijk heeft dit geleid tot de eerste DevOpsDays in 2009 in Ghent, Belgium door Patrick Debois.

INTRAFFIC

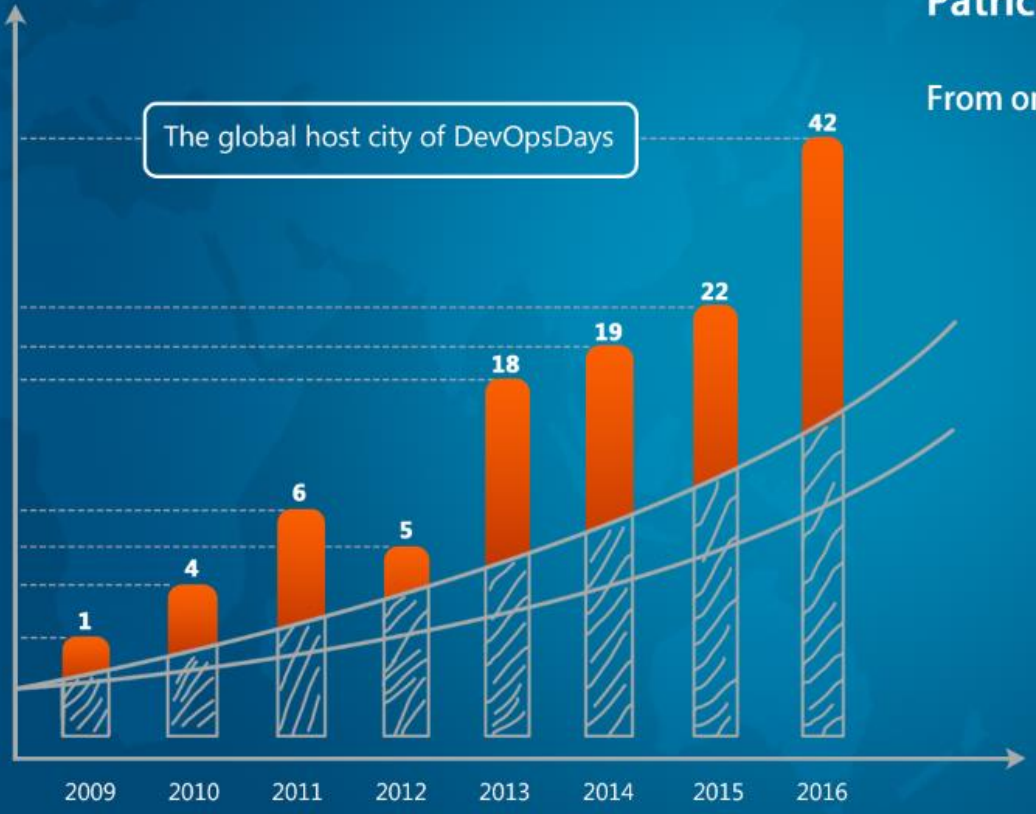
https://www.bagevent.com/event/devopsdays-beijing2017en

From **2009** to **2016**

Patrick Debois initiated the DevOpsDays campaign

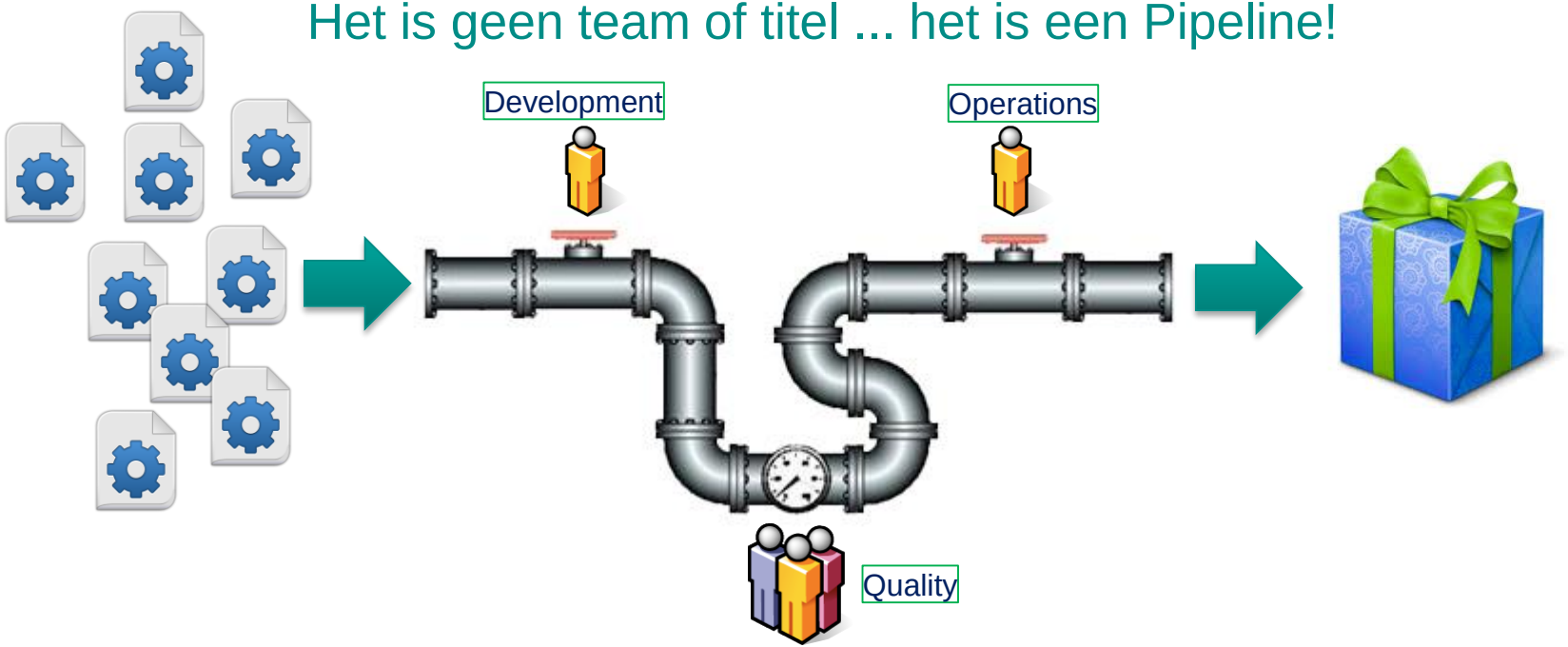
From only **1** city — **Ghent**

to **42** citys today



Wat is DevOps?

Het is geen team of titel ... het is een Pipeline!

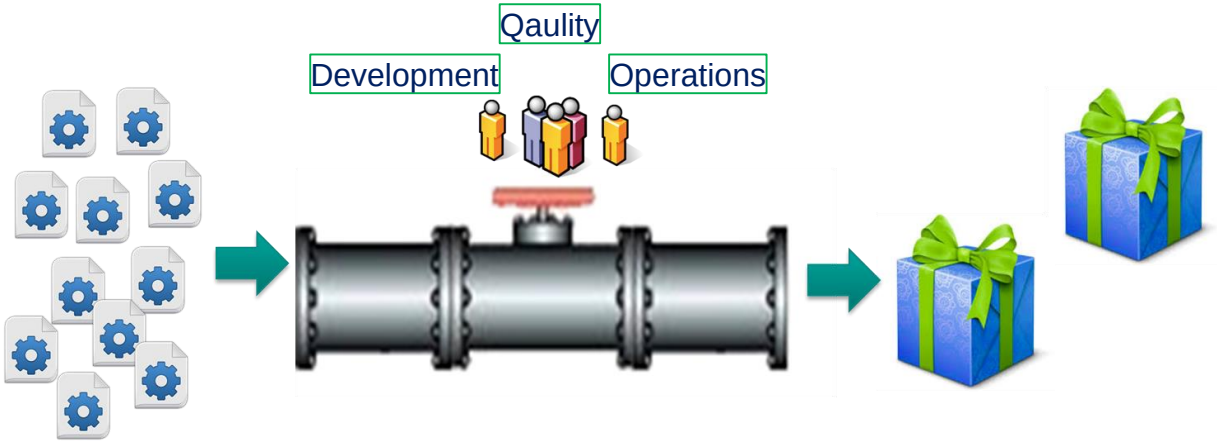


INTRAFFIC

DevOps is…

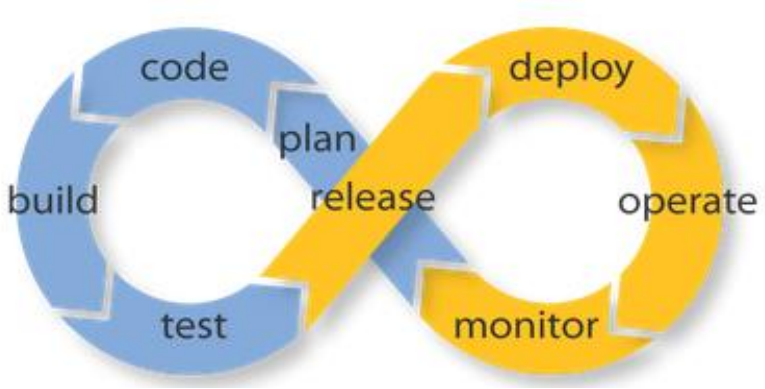
Meer samenwerking en automatisering tussen de ontwikkelings- en operationele teams = grotere Pipeline

Verwijder de bottlenecks - stroomlijn het proces



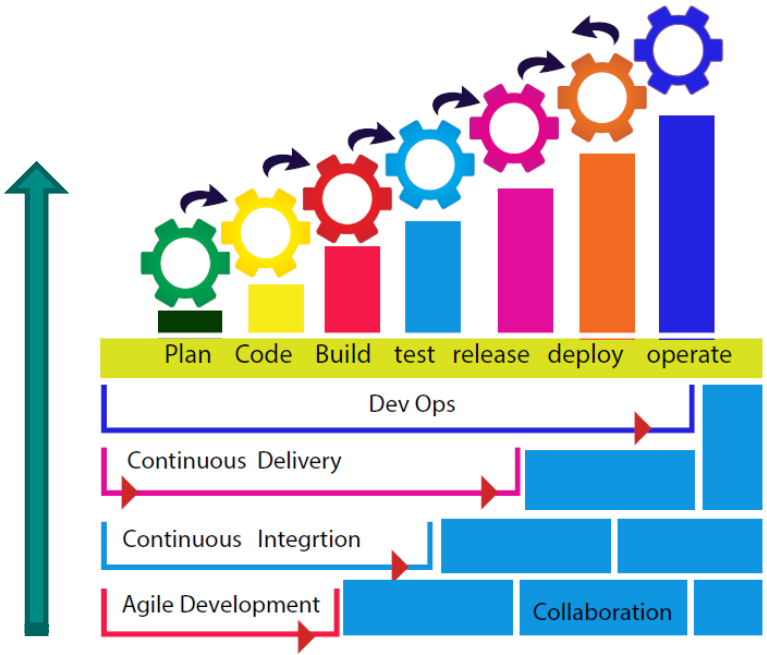
DevOps is…

DevOps richt zich op samenwerking tussen Agile Software Development, Operations en Quality Assurance.

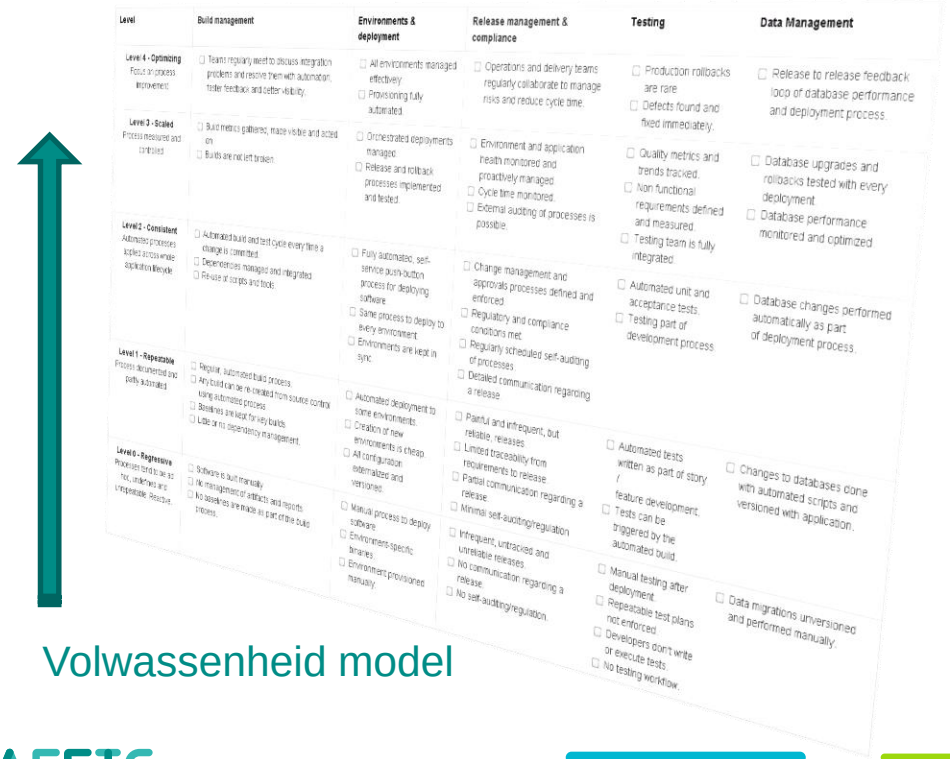


INTRAFFIC

DevOps is aanvullend aan Agile



Automatiseren



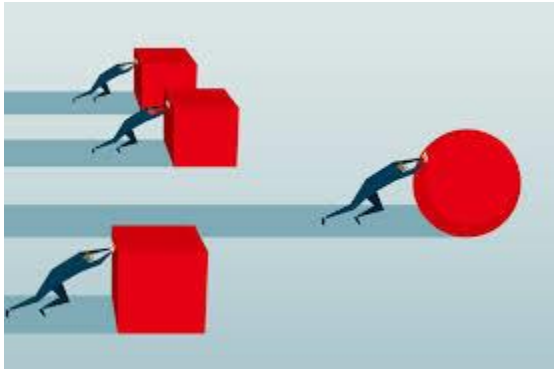
Level	Build management	Environments & deployment	Release management & compliance	Testing	Data Management
Level 4 - Optimizing Focus on process improvement	☐ Teams regularly meet to discuss integration problems and resolve them with automation; faster feedback and better visibility.	☐ All environments managed effectively ☐ Provisioning fully automated.	☐ Operations and delivery teams regularly collaborate to manage risks and reduce cycle time.	☐ Production rollbacks are rare ☐ Defects found and fixed immediately.	☐ Release to release feedback loop of database performance and deployment process.
Level 3 - Scaled Process measured and controlled	☐ Build metrics gathered, made visible and acted on ☐ Builds are not left broken	☐ Orchestrated deployments managed. ☐ Release and rollback processes implemented and tested.	☐ Environment and application health monitored and proactively managed. ☐ Cycle time monitored. ☐ External auditing of processes is possible.	☐ Quality metrics and trends tracked. ☐ Non functional requirements defined and measured. ☐ Testing team is fully integrated.	☐ Database upgrades and rollbacks tested with every deployment. ☐ Database performance monitored and optimized.
Level 2 - Consistent Automated processes applied across whole application lifecycle	☐ Automated build and test cycle every time a change is committed. ☐ Dependencies managed and integrated. ☐ Reuse of scripts and tools.	☐ Fully automated, self-service push-button process for deploying software. ☐ Same process to deploy to every environment. ☐ Environments are kept in sync.	☐ Change management and approvals processes defined and enforced. ☐ Regulatory and compliance conditions met. ☐ Regularly scheduled self-auditing of processes. ☐ Detailed communication regarding a release.	☐ Automated unit and acceptance tests. ☐ Testing part of development process.	☐ Database changes performed automatically as part of deployment process.
Level 1 - Repeatable Process documented and partly automated	☐ Regular, automated build process. ☐ Any build can be re-created from source control using automated process. ☐ Baselines are kept for key builds. ☐ Little or no dependency management.	☐ Automated deployment to some environments. ☐ Creation of new environments is cheap. ☐ All configuration externalized and versioned.	☐ Painful and infrequent, but reliable, releases. ☐ Limited traceability from requirements to release. ☐ Partial communication regarding a release. ☐ Minimal self-auditing/regulation.	☐ Automated tests written as part of story / feature development. ☐ Tests can be triggered by the automated build.	☐ Changes to databases done with automated scripts and versioned with application.
Level 0 - Regressive Processes tend to be ad hoc, undefined and unrepeatable. Reactive.	☐ Software is built manually. ☐ No management of artifacts and reports. ☐ No baselines are made as part of the build process.	☐ Manual process to deploy software. ☐ Environment-specific binaries. ☐ Environment provisioned manually.	☐ Infrequent, untracked and unreliable releases. ☐ No communication regarding a release. ☐ No self-auditing/regulation.	☐ Manual testing after deployment. ☐ Repeatable test plans not enforced. ☐ Developers don't write or execute tests. ☐ No testing workflow.	☐ Data migrations unversioned and performed manually.

Volwassenheid model

INTRAFFIC

Waarom DevOps?

- Efficiëntie: een kortere Time-To-Market
- Voorspelbaarheid: Lagere uitvalpercentage releases
- Reproduceerbaarheid: Versiebeheer
- Onderhoudbaarheid: Snellere hersteltijd



Waarom bedrijven DevOps adopteren?

COMPANY	DEPLOY FREQUENCY	DEPLOY LEAD TIME	RELIABILITY	CUSTOMER RESPONSIVENESS
Amazon	23,000/day	minutes	high	high
Google	5,500/day	minutes	high	high
Netflix	500/day	minutes	high	high
Facebook	1/day	minutes	high	high
Twitter	3/week	minutes	high	high
Typical enterprise	once every 9 months	months or quarters	low/medium	low/medium

Source: https://xebialabs.com/assets/files/whitepapers/ITRev_DevOps_Guide_5_2015.pdf

INTRAFFIC

Wat is dan de uitdaging?

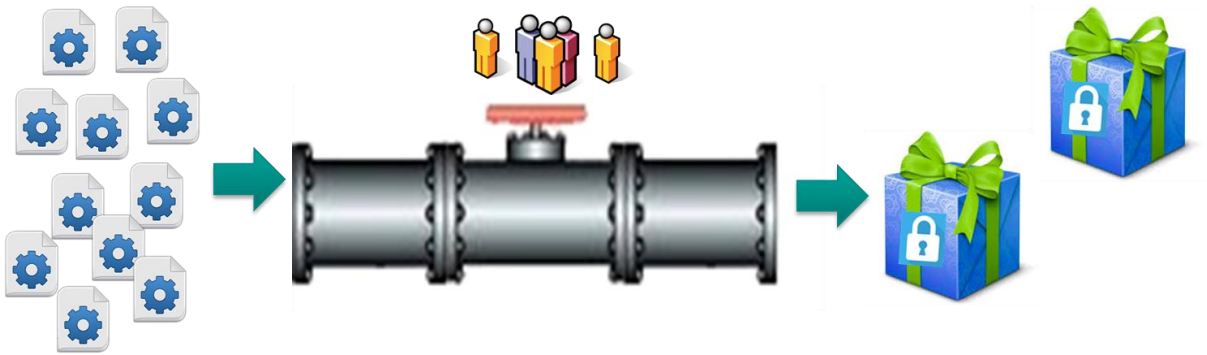
Business: Onderscheiden, Concurreren ➔ functionaliteiten sneller produceren, zonder in te leveren op stabiliteit en beschikbaarheid

Voor Development en Operations is beveiliging echter een botelnek in DevOps omgeving, ➔ Security is iets wat het proces onnodig vertraagt.

INTRAFFIC

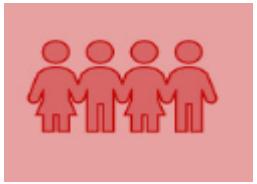
Oplossing is DevSecOps

Doel van DevSecOps is om organisaties in staat te stellen om inherent veilige software te leveren op DevOps-snelheid.



DevSecOps: wat moet geoptimaliseerd worden?

1. Cultuur

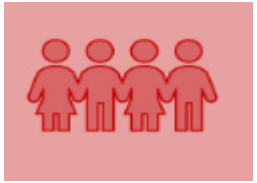


2. Processen

3. Technologie

INTRAFFIC

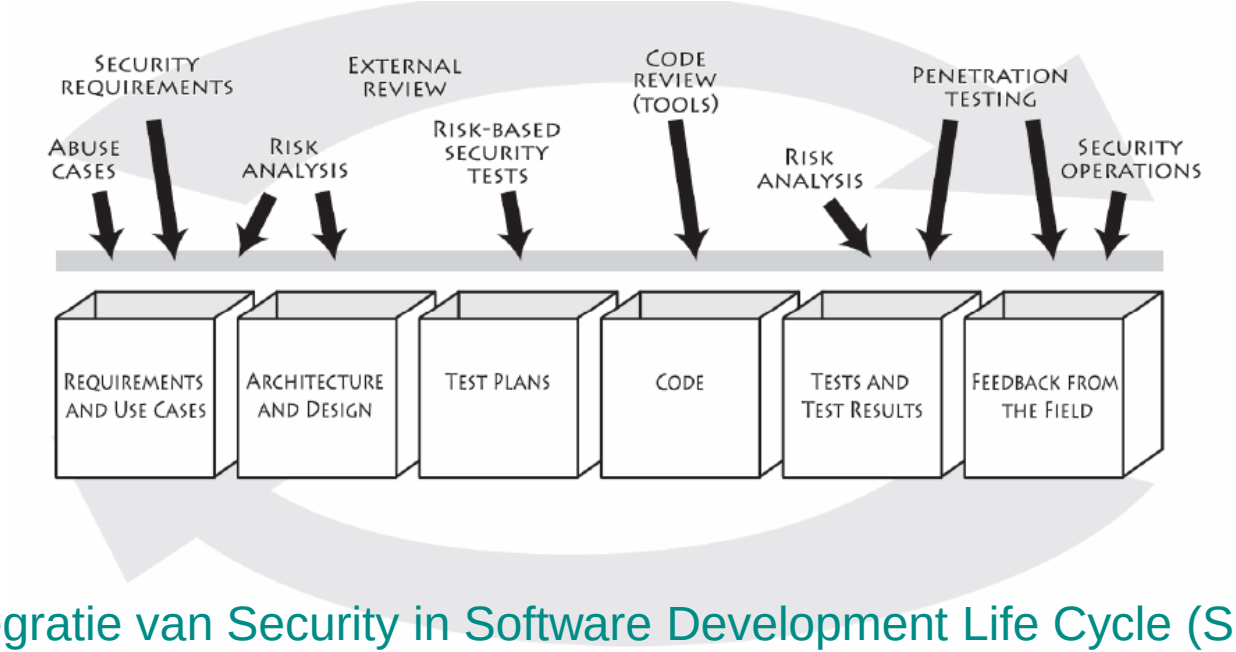
Cultuur



- Communicatie en transparantie
- Continue verbetering mentaliteit
- Iedereen is verantwoordelijk voor de beveiliging
- Automatiseren (zoveel mogelijk)

INTRAFFIC

Processen



Integratie van Security in Software Development Life Cycle (SDLC)

INTRAFFIC

Technologie

Security requirement

Security by design

Security in sprintplanning

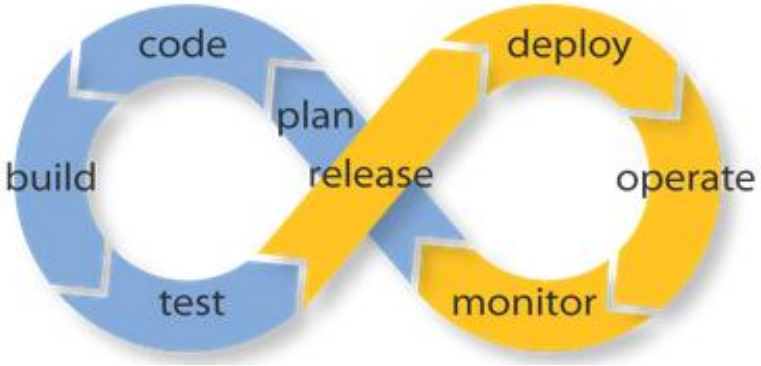
Security backlog

Security reviews

Static code analysis

Dynamic code analysis

Abuse usecases



Thread modeling

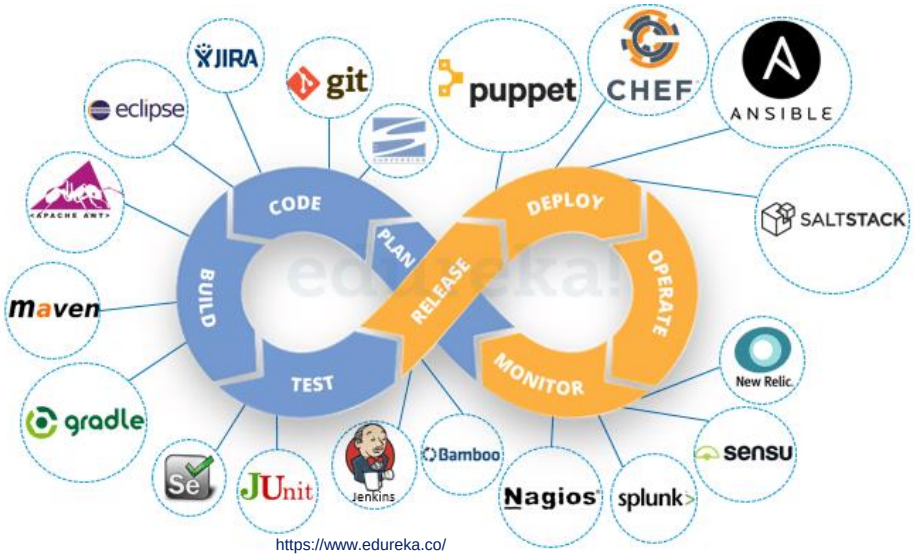
Automated
Pen test
Network scan

“Evidence based security” testing

Log monitoring

INTRAFFIC

Automatiseren is key voor Secure DevOps ofwel DevSecOps!



https://www.edureka.co/

Zichtbaar
Herhaalbaar
Consistent
Stabiel

Zonder automatisering is er geen DevSecOps

INTRAFFIC

Voorbeeld van tools voor het automatiseren van security testen in alle lagen

- Cloud infrastructure: tools zoals **Microsoft Azure Advisor**, of **evident.io** kan helpen bij het scannen van configuraties op best practices met betrekking tot beveiliging.
- Security testing: **Gauntlt** is open framework (Given When Then)
- Code-analyse - Tools zoals **Veracode** kunnen code scannen om potentiële kwetsbaarheden in code en open source-bibliotheken te vinden.
- Runtime-applicatiebeveiliging - Tools zoals **Contrast Security** worden tijdens de productie in uw toepassing uitgevoerd en kunnen in realtime beveiligingsproblemen identificeren en voorkomen

INTRAFFIC

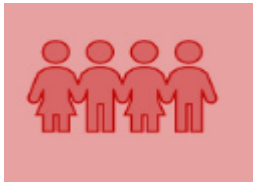
Mijn ervaring met DevSecOps

Product: Apotheek Informatie Systeem (AIS)
Bedrijf: Promedico in Utrecht, Project APRO 2016-2018

1. 2016-Q1: BIA (Business Impact Analysis) ➔ focus op Cultuur
    1. Time-to-Market (go live Q4-2017)
    2. Privacy en Security beleid en strategie (medische en persoonsgegevens)
    3. Besef van urgentie en noodzaak, awareness en training
    4. DevSecOps overleg voor gestroomlijnde en betere communicatie

2. 2016-Q2: Externe Audit ➔ focus op Processen
    1. RAD: Agile /scrum
    2. Privacy en security requirements (abuse usestories, Security by Design)
    3. Integreren van Security in Software Development Life Cycle (SDLC)

3. 2016-Q3: Interne Audit ➔ focus op Technologie
    1. RAD: Low Code platform (Outsystems)
    2. DevOps: CI/CD, Jenkins, Finesse, Kibana
    3. DevSecOps: geautomatiseerd security testing in toolchain
        - Evidance based security testing
        - Security monitoring
        - Automatische security scans



Tools

kibana

JIRA

Xray for JIRA

FitNesse

OWASP ZAP

Se

Jenkins

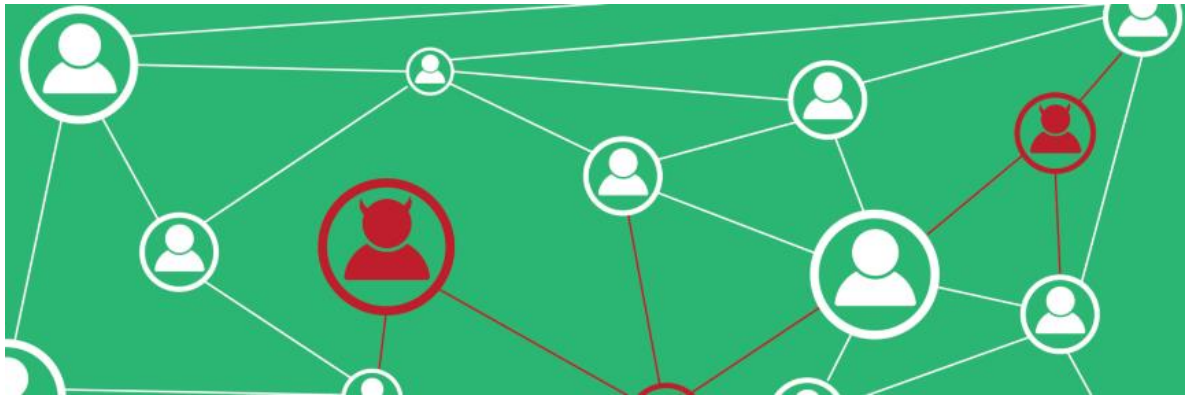
Maven

git

outsystems

INTRAFFIC

Abuse usecases



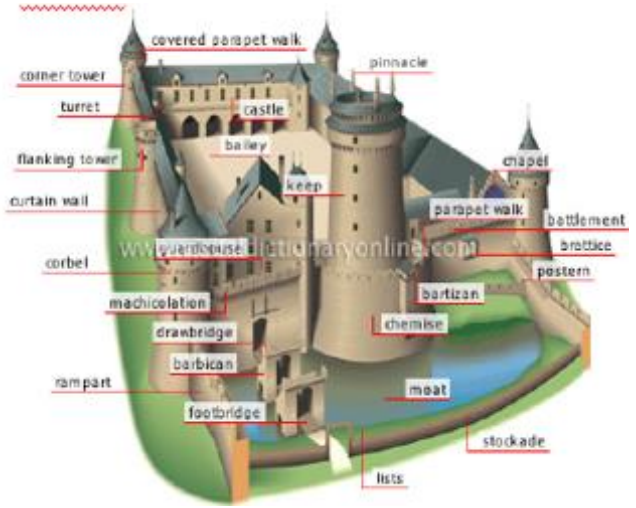
"Abuse usecases" zijn user story’s gericht op functioneel misbruik van de applicatie: Denk als een hacker

INTRAFFIC

Security-by-design

Attack Surface Analysis en Reductie
- Overzicht van mogelijke deuren in applicatie
- Minder "deuren" (toegangspunten) betekent meestal minder risico
- Big Attack Surface = Big Security Work = Big Security Problems

Threat Modeling
- Identificeert de meest kritieke bedreigingen en risico's
- Maatregelen nemen gebaseerd op risico analysis



INTRAFFIC

Security-by-Design

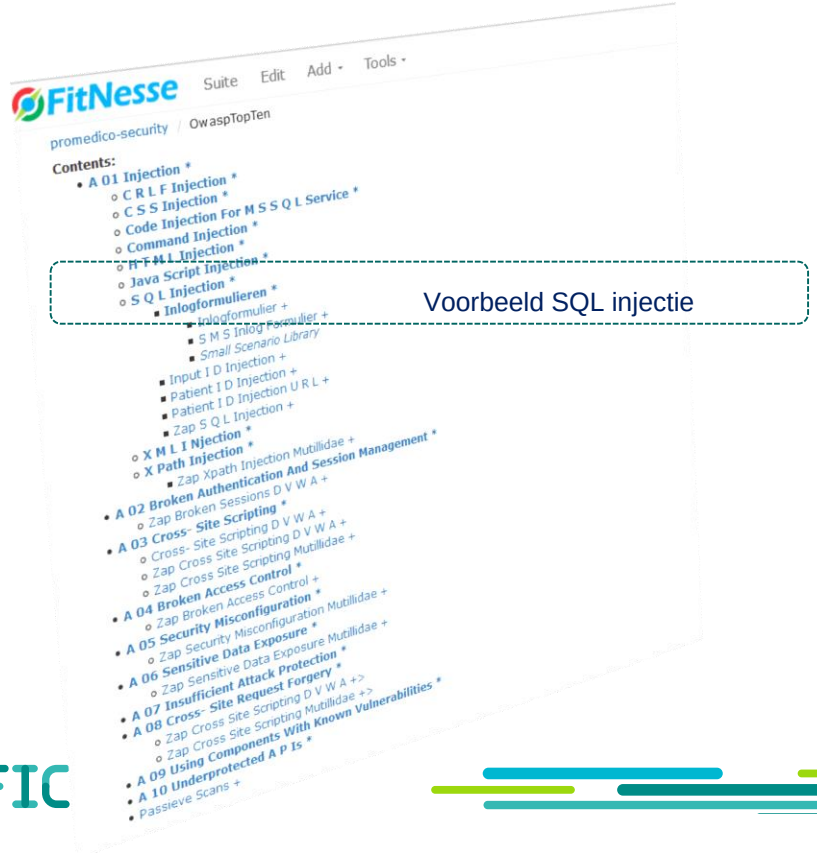
Beter voorkomen dan genezen!

- In plaats van te vertrouwen op handmatige beveiligingscontrole, eist Security-by-Design dat beveiligingscontrole ingebouwd geautomatiseerd in proces moeten worden.
    - Bouw beveiliging in elke laag
    - Infrastructuur als code
    - Handmatige processen elimineren

INTRAFFIC

“Evidence based security” testing

- Owasp top 10



- Voor elk Owasp top-10-risico is er een fitnesse-geautomatiseerde scripts gemaakt die bij elke release gedraaid wordt

OWASP ZAP FitNesse SPECIALISTERREN DE BESTE TESTERS

INTRAFFIC

Voorbeeld SQL injectie

```
SELECT * FROM users WHERE email = '$email' AND password = md5('$password') ;
```

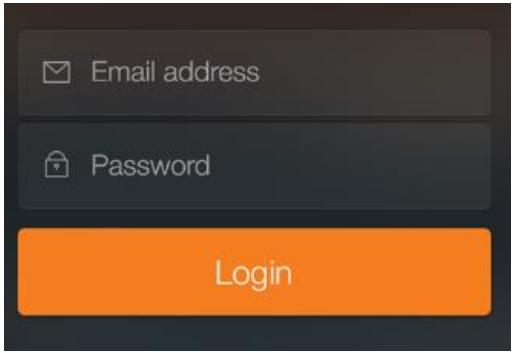
Supplied values { `xxx@xxx.xxx` `xxx') OR 1 = 1 -- ]`

```
SELECT * FROM users WHERE email = 'xxx@xxx.xxx' AND password = md5('xxx') OR 1 = 1 -- ]');
```

```
SELECT * FROM users WHERE FALSE AND FALSE OR TRUE

SELECT * FROM users WHERE FALSE OR TRUE

SELECT * FROM users WHERE TRUE
```

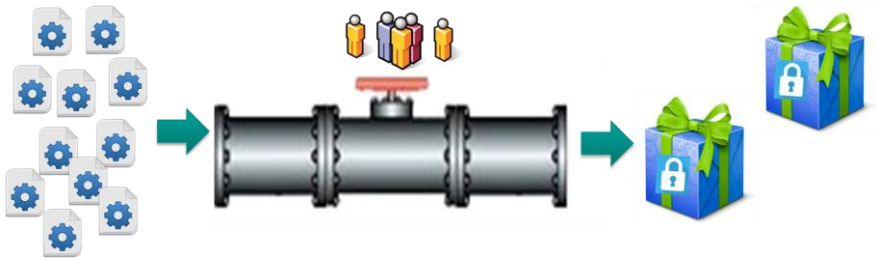


MD5 is encryptie techniek
Statement resulteert in TRUE en naar dashboard pagina

INTRAFFIC

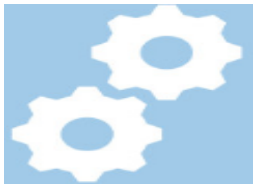
De hamvraag:
Is DevSecOps een buzzword of toch een noodzakelijke stap richting Secure DevOps?

Mijn antwoord: DevSecOps is een noodzakelijke stap om organisaties in staat te stellen inherent veilige software te leveren op DevOps-snelheid.



Focus op: Cultuur, Proces, Technologie, Automatiseren!

DevSecOps = DevOps + Security + Automatiseren



INTRAFFIC