Logius
Ministerie van Binnenlandse Zaken en Koninkrijksrelaties

# Aansluitvoorwaarden Kubernetes Applicaties - Standaard Platform

## Aansluitvoorwaarden Kubernetes Applicaties - Standaard Platform

Deze pagina beschrijft de aansluitvoorwaarden voor de uitrol van Cloud Native Kubernetes Applicaties [CNCF, Kubernetes] op het Standaard Platform (SP).

Het Standaard Platform wordt gerealiseerd en beheerd door Logius.

### Versiegegevens

**Publicatiedatum:** 19 januari 2021
**Versie:** 1.0

## Inleiding

### Leeswijzer

De eerste 7 hoofdstukken beschrijven aansluitvoorwaarden voor het realiseren, aanleveren, uitrollen, testen, beheren en beveiligen van applicaties. De aansluitvoorwaarden zijn voorzien van een nummering (Containerized Applicaties/CNA), zodat deze kunnen worden hergebruikt als referentie in andere documenten. Hoofdstuk 8 besluit met een categorisering van de verschillende soorten componenten die op het SP (kunnen) draaien.

In dit document wordt uitgegaan van kennis van de [SP Solution Architecture].

### Doelgroep

De doelgroep van dit document bestaat uit klanten, brokers, leveranciers en beheerders die Cloud Native applicaties op basis van Kubernetes willen realiseren, aanleveren, uitrollen, draaien en beheren op het SP:

| Doelgroep | Doel |
|---|---|
| Klant | Laat applicatie realiseren, uitrollen, draaien en beheren op het SP. |
| Broker | Voert regie op realisatie, uitrol, draaien en beheer. |
| Applicatieleverancier (leverancier | Realiseert applicatie, en voert onderhoud en vernieuwing uit. |
| Applicatiebeheer (beheer) | Rolt applicatie uit naar Productie en voert operationeel beheer op de uitgerolde/draaiende applicatie. |

### Referenties

In dit document wordt door middel van [Vierkante Haakjes] gerefereerd naar de volgende documenten:

| Referentie | Document |
|---|---|
| AVG | https://autoriteitpersoonsgegevens.nl |
| BIO | BIO (Baseline Informatiebeveiliging Overheid) |
| BIR 2017 | BIR (Baseline Informatiebeveiliging Rijksdienst) |
| Client Certificate Authentication | https://en.wikipedia.org/wiki/Client_certificate |
| CNCF | https://www.cncf.io |
| CNCF Landscape | https://landscape.cncf.io/ |
| DKIM-record | DomainKeys Identified Mail |
| DMZ | Demilitarized zone (informatica) |
| Docker | https://www.docker.com/ |
| GitHub | https://github.com |
| I-strategie Rijk 2019-2021 | I-strategie Rijk 2019-2021 |
| Kubernetes | https://kubernetes.io |
| Kubernetes Ingress | Kubernetes Ingress |
| Kubernetes Pods and Containers Configuration | https://kubernetes.io/docs/tasks/configure-pod-container/ |

| | |
|---|---|
| Kubernetes Resource Quotas | https://kubernetes.io/docs/concepts/policy/resource-quotas/ |
| Let’s Encrypt | https://letsencrypt.org |
| MX-record | https://nl.wikipedia.org/wiki/MX-record |
| NCSC Beveiliginsrichtlijnen voor Webapplicaties | NCSC Beveiliginsrichtlijnen voor Webapplicaties |
| NIST Application Container Security Guide | Application Container Security Guide |
| oAuth | https://nl.wikipedia.org/wiki/OAuth |
| OWASP | https://www.owasp.org/ |
| OWASP HSTS | HTTP Strict Transport Security Cheat Sheet |
| PKIo | https://www.pkioverheid.nl |
| SCA | OWASP Dependency-Track |
| SP Beleid Acceptabel Gebruik | SP Beleid Acceptabel Gebruik |
| SP Solution Architecture | SP Solution Architecture Document v1.0.pdf |
| SPF-record | https://nl.wikipedia.org/wiki/Sender_Policy_Framework |
| The Twelve Factor App | https://12factor.net/ |

# Applicatie Realiseren

Het SP ondersteunt applicaties op basis van Docker containertechnologie [Docker].

| Eis | Omschrijving |
|---|---|
| CNA01 | De Applicatieleverancier bezit aantoonbare kennis van- en ervaring met Docker en Kubernetes Hiermee kan het realisatietraject efficiënt worden gestart. Het opleiden van de Applicatieleverancier in Docker- en Kubernetes technologie valt buiten de scope van de dienstverlening van SP/CS. |
| CNA02 | Applicaties worden ontwikkeld in de CI/CD-omgeving van het SP, of in een eigen ontwikkelomgeving van de Applicatieleverancier Indien een applicatie wordt ontwikkeld in een eigen ontwikkelomgeving, dan dienen de applicatie artefacts (broncode, documentatie, automatische tests, ..) alsnog te worden aangeleverd naar de CI/CD-omgeving (GitLab) van het SP. Van daaruit kunnen de applicatie artefacts worden gecontroleerd op kwetsbaarheden en uitgerold naar Test en Productie. |
| CNA03 | Applicaties worden als Docker images gerealiseerd Alleen images die draaien in Kubernetes/Docker Runtime op een Linux host zijn op het SP toegestaan. |
| CNA04 | Applicaties voldoen aan de principes van “The Twelve-Factor App” Zodat deze op generieke wijze kunnen worden geschaald en hoog-beschikbaar kunnen worden uitgerold [The Twelve-Factor App]. |
| CNA05 | Applicaties zijn gebaseerd op SP-componenten of op “Bring Your Own Component”. Zie hoofdstuk Categorisering van Componenten. De Applicatieleverancier blijft aansprakelijk voor de toepassing van de componenten in de eigen applicaties. |
| CNA06 | “Bring Your Own Componenten” worden up-to-date gehouden De Applicatieleverancier zorgt ervoor dat de toegepaste “Bring Your Own Componenten” up-to-date worden gehouden, zodat beveiliging en overdraagbaarheid zijn geborgd. Zie verder hoofdstuk 8 > Categorisering van Componenten. |
| CNA07 | Applicaties maken alleen gebruik van de faciliteiten die aangewezen zijn door het SP en diens Architectuur Het SP/CS wilt graag de “cloud agnosticiteit” van het platform waarborgen. Dit biedt de mogelijkheid om het platform over meerdere onderliggende Cloud Providers gemakkelijk uit te rollen. Het kan zijn dat naast de tools en faciliteiten die door het SP/CS worden aangeboden er ook alternatieven zijn die bij een specifieke onderliggende Cloud Provider aanwezig zijn. Deze alternatieven dienen vermeden te worden in het kader van de “cloud agnosticiteit”. Bijvoorbeeld: het SP/CS schrijft voor om applicaties uit te rollen vanuit de CI/CD- omgeving. Een voorbeeld van te vermijden alternatieve functionaliteiten is de OpenShift Commandline (“oc”) voor het uitrollen en beheren van applicaties, of de OpenShift Service Catalog voor het handmatig uitrollen van voorgedefinieerde OpenShift componenten. |

# Applicatie Aanleveren

Een Applicatie wordt als Docker Image (Image) aangeleverd naar een Private Docker Registry (“SP Image Registry”). De applicatie-uitrol wordt gedefinieerd d.m.v. een Applicatieconfiguratie. Deze configuratie wordt separaat aangeleverd. De wijze waarop dit plaatsvindt wordt in dit hoofdstuk beschreven.

| Eis | Omschrijving |
|---|---|
| CNA08 | De Applicatieleverancier levert een Applicatie aan door deze te pushen naar de SP Image Registry (docker push) De SP Image Registry is een Private Docker Registry die door SP/CS ter beschikking wordt gesteld. Voor het aanleveren dient gebruik te worden gemaakt van een VPN-verbinding met Two Factor Authenticatie (2FA). Deze wordt door SP/CS ter beschikking gesteld. |
| CNA09 | Images voldoen aan een vastgestelde naamgevingconventie De volgende naamgevingconventie is van toepassing:<br><br><SP Image Registry URL>:<poort#>/<br><klantnaam of organisatie>/<br><organisatie-unit en/of applicatiedomein>/<br><applicatienaam>:<br><versie><br><br>Bijvoorbeeld: registry.overheid.standaardplatform.rijksapps.nl/ienw/dci/voorbeeld-app:1.1 |

| | |
|---|---|
| CNA010 | Images worden door SP/CS op beveiliging gevalideerd Na deze validatie kunnen images om beveiligingsredenen worden afgekeurd. Indien een image wordt afgekeurd wordt argumentatie aan de Applicatieleverancier verstrekt en lost de Applicatieleverancier het beveiligingsissue op. |
| CNA011 | De Applicatieleverancier levert applicatieconfiguratie aan door deze te pushen naar de SP CI/CD-omgeving (git push) Een Applicatieconfiguratie definieert de uitrol van een applicatie. De Applicatieconfiguratie:<br>• Omvat de configuratie voor uit te rollen Kubernetes Objecten (Pods, Resources, Security Context, Secrets, Persistent Volume Claims, etc.).<br>• Wordt gedefinieerd d.m.v. standaard tools (Kubectl, Kustomize, Helm), eventueel aangevuld met extra scripts (Python, Shell).<br>• Wordt in de SP CI/CD-omgeving (GitLab) geïntegreerd. Hiermee kan de applicatie over meerdere OTAP-omgevingen worden uitgerold, zonder dat er wijzigingen in de applicatie zelf hoeven te worden gemaakt.<br>• Wordt afgestemd met Applicatiebeheer. |

# Applicatie Uitrollen

De applicatie en de bijbehorende configuratie wordt uitgerold naar een OTAP-omgeving. Dit vindt plaats aan de hand van Kubernetes Objecten (ConfigMaps, Secrets, Persistent Volume Claims, Network Policies, Pods, Services, Deployments, StatefulSets, etc.).

## Test

| Eis | Omschrijving |
|---|---|
| CNA012 | De Applicatieleverancier rolt de Applicatieconfiguratie uit naar Test De Applicatieconfiguratie wordt uitgerold vanuit CI/CD naar Test. Voor het uitrollen dient gebruik te worden gemaakt van een VPN-verbinding met Two Factor Authenticatie (2FA); analoog aan hoofdstuk 2. Voor testdoeleinden kan ook gebruik worden gemaakt van de Kubernetes Commandline Interface (CLI). Hiervoor is een aanvullend authenticatiemiddel nodig. Deze wordt door SP/CS ter beschikking gesteld. |
| CNA013 | De Applicatieleverancier rolt de Applicatie uit naar Test De Applicatie wordt analoog aan de Applicatieconfiguratie uitgerold. De uitrol maakt gebruik van de eerder aangeleverde Images in de SP Image Registry. |
| CNA014 | De Applicatieleverancier draagt zorg voor adequaat resourcegebruik De Applicatieleverancier definieert van tevoren de gewenste hoeveelheid resources (RAM, CPU, Storage). Het SP stelt de bijbehorende grenzen in op Namespace niveau [Kubernetes Resource Quotas]. Binnen deze toegestane grenzen kan de Applicatieleverancier resources aan Kubernetes Pods en Containers toewijzen [Kubernetes Pods and Containers Configuration]. |
| CNA015 | Applicatie uitrol maakt alleen gebruik van de faciliteiten die aangewezen zijn door het SP en diens Architectuur Zodat de “cloud agnosticiteit” eigenschap van het SP gewaarborgd blijft. |

## Productie

Uit beveiligingsoverwegingen dient de Applicatieleverancier er rekening mee te houden dat Applicatiebeheer apart is geautoriseerd.

| Eis | Omschrijving |
|---|---|
| CNA016 | De Applicatiebeheerder haalt de Applicatie op naar Productie.<br>De Applicatiebeheerder haalt de Images op vanuit Test en plaatst deze in Productie. |
| CNA017 | De Applicatiebeheerder haalt Applicatieconfiguratie op naar Productie.<br>De Applicatiebeheerder haalt Applicatieconfiguratie op vanuit Test en plaatst/her- configureert deze in Productie. |
| CNA018 | De Applicatiebeheerder rolt Applicatieconfiguratie uit naar Productie.<br>De Applicatiebeheerder rolt Applicatieconfiguratie uit naar Productie; analoog aan Test. |
| CNA019 | De Applicatiebeheerder rolt de Applicatie uit naar Productie.<br>De Applicatiebeheerder rolt de Applicatie uit naar Productie; analoog aan Test |
| CNA020 | De Beheeromgeving is apart geautoriseerd.<br>De uitrol naar Productie dient plaats te vinden in een apart geautoriseerde “Beheer Pipeline” (GitLab project) binnen de CI/CD-omgeving. De opslag van Applicatie en Applicatieconfiguratie, de scheiding van Test en Productie en de auditeerbaarheid van beheerhandelingen valt onder de verantwoordelijkheid van Applicatiebeheer. |
| CNA021 | Wie uitrolt, beheert.<br>Het operationele applicatiebeheer valt onder de verantwoordelijkheid van Applicatiebeheer. Dit is de partij die de applicatie op Productie uitrolt en de runtime configuratie kent. |

#### Definitie 'test'

'Test' is hier de verzamelnaam voor de OTAP-omgevingen Test, Demo en Acceptatie. Deze omgevingen worden naar behoefte door SP/CS beschikbaar gesteld.

#### Definitie 'productie'

'Productie' is hier de verzamelnaam voor OTAP-omgevingen Pre-Productie en Productie. Ook deze omgevingen worden naar behoefte door SP/CS beschikbaar gesteld.

# Applicatie Testen

De applicatie dient op diverse wijze getest te worden.

### Test

| Eis | Omschrijving |
|---|---|
| CNA022 | Automatische tests worden uitgevoerd in Test De Applicatieleverancier voert functionele-, beveiligings- en performance tests uit en lost geconstateerde fouten op. |

### Productie

| Eis | Omschrijving |
|---|---|
| CNA023 | Automatische tests worden aangeleverd naar Productie.<br>De Applicatieleverancier levert de tests op aan Applicatiebeheer. De tests worden samen met omgevingspecifieke configuratie aangeleverd. Hiermee kunnen de tests over meerdere omgevingen worden gedraaid, zonder dat er wijzigingen in de tests zelf hoeven te worden gemaakt. |
| CNA024 | Automatische tests worden uitgevoerd in Productie.<br>Applicatiebeheer voert de tests uit in Productie. Indien er zich fouten voordoen, zet Applicatiebeheer deze door/terug aan de Applicatieleverancier. De Applicatieleverancier lost de fouten op. |

## Applicatie Beheren

Het applicatiebeheer wordt getest in Test en vindt daadwerkelijk plaats in Productie.

### Test

Voor het testen van de applicatie dienen diverse SP Componenten geconfigureerd te zijn.

| Eis | Omschrijving |
|---|---|
| CNA025 | DNS in Test Het testen van een applicatie kan binnen de SP VPN of op internet plaatsvinden. Indien testen binnen VPN:<br>• De Applicatieleverancier kiest in overleg met Klant en Applicatiebeheer een subdomein onder ".rijksapps.nl".<br>• Bij uitrol van de applicatie wordt het domein automatisch doorgevoerd in de interne DNS van het cluster.<br><br>Indien testen op internet:<br>• De Applicatieleverancier arrangeert zelf in overleg met Klant en DNS-provider een publiek domein.<br>• Het publieke IP-adres waar dit domein op moet worden aangesloten wordt door SP/CS aan de leverancier aangeboden.<br>• Aanbevolen wordt om het domein voor Test zoveel mogelijk te laten lijken op het uiteindelijke domein in Productie (bijvoorbeeld: tst.<uiteindelijke domein in Productie). |
| CNA026 | Servercertificaat in Test<br>Webapplicaties en API's worden d.m.v. [Kubernetes Ingress] ontsloten via HTTPS. De Applicatieleverancier is verantwoordelijk voor het beheer- en het geautomatiseerd plaatsen van de bijbehorende (Let's Encrypt) servercertificaten. |
| CNA027 | Client certificaat in Test<br>API's die worden afgenomen kunnen op diverse manieren beveiligd zijn: niet beveiligd/open, beveiligd met oAuth over enkelzijdig SSL, of beveiligd met client certificaten over dubbelzijdig SSL [oAuth, Client Certificate Authentication]. - afname kan op beveiligd zijn met dubbelzijdig SSL ("Client Certificate Authentication)". De Applicatieleverancier is verantwoordelijk voor het beheer- en het geautomatiseerd plaatsen van de bijbehorende (PKIo) test client certificaten. |
| CNA028 | Mail Indien gebruik wordt gemaakt van een klant specifiek maildomein, dient de Applicatieleverancier bijbehorende [DKIM-], [SPF-] en [MX-records] op de DNS te gebruiken. |
| CNA029 | eHerkenning en/of DigiD Indien gebruik wordt gemaakt van eHerkenning en/of DigiD, dient de Applicatieleverancier in samenwerking met Klant en SP/CS zorg te dragen voor het inregelen van de juiste certificaten/makelaars etc. |

### Productie

Analoog aan Test, ditmaal voor Productie.

| Eis | Omschrijving |
|---|---|
| CNA030 | DNS in Productie<br>Analoog aan Test, ditmaal uitgevoerd door Applicatiebeheer. |
| CNA031 | Servercertificaat in Productie<br>Analoog aan Test, ditmaal uitgevoerd met PKIo-certificaten door Applicatiebeheer. |
| CNA032 | Client certificaat in Productie<br>Analoog aan Test, ditmaal uitgevoerd met PKIo-certificaten door Applicatiebeheer. |
| CNA033 | Mail<br>Analoog aan Test, ditmaal uitgevoerd voor Applicatiebeheer. |

| | |
|---|---|
| CNA034 | eHerkenning en/of DigiD<br>Analoog aan Test, ditmaal uitgevoerd voor Applicatiebeheer. |

### Logging

Draaiende applicaties moeten worden gelogd.

| Eis | Omschrijving |
|---|---|
| CNA035 | Applicatieve Logging<br>Applicatieve Logging wordt door de Applicatieleverancier naar “Standard Out” op het lokale bestandssysteem weggeschreven. SP/CS draagt zorg voor het transporteren van de logbestanden naar het centrale logmanagement component. Dit component kan door Applicatiebeheer worden geraadpleegd. |
| CNA036 | Applicatieve Logrotating<br>Applicatieve Logrotating wordt door de Applicatieleverancier ingesteld. Dit voorkomt het vollopen van het lokale bestandssysteem. |

### Monitoring

Draaiende applicaties moeten worden gemonitord.

| Eis | Omschrijving |
|---|---|
| CNA03 | Applicatie “Health”<br>De gezondheid van de applicatie wordt door Applicatieleverancier & Applicatiebeheer ingeregeld door middel van de standaard aanwezige voorzieningen (Kubernetes Readiness, Kubernetes Liveliness, Prometheus AlertManager, en Grafana Dashboard). |

### Backup/Restore

Draaiende containers kunnen worden vervangen/ge-upgrade. Productiedata wordt gewaarborgd.

| Eis | Omschrijving |
|---|---|
| CNA038 | Storage<br>De Applicatieleverancier geeft aan welke storagebehoefte een applicatie heeft. |
| CNA039 | Docker Volumes<br>De Applicatieleverancier maakt gebruik van Kubernetes Persistent Volume Claims, zodat deze een container-herstart overleven. |
| CNA040 | Backup<br>De Klant geeft, in samenspraak met Applicatieleverancier, aan welke back- upbehoefte een applicatie heeft. Standaard wordt uitgegaan van back-ups op basis van Docker volumes. |
| CNA041 | Restore<br>Op verzoek van Applicatieleverancier en Klant wordt door SP/CS een restore uitgevoerd. |

### Transitie

Applicatierealisatie en Applicatiebeheer kunnen overgaan op een nieuwe Applicatieleverancier of Applicatiebeheerpartij.

| Eis | Omschrijving |
|---|---|
| CNA042 | Transitie van Applicatieleverancier is voorafgaand aan een realisatietraject afgestemd met Klant/Broker en SP/CS<br>Wanneer een nieuwe Applicatieleverancier de (door) ontwikkeling overneemt van de bestaande Applicatieleverancier, is de bestaande Applicatieleverancier verantwoordelijk voor de overdracht. Dit is inclusief de daarbij behorende voorwaarden zoals in dit document verwoord. |
| CNA043 | Transitie van Applicatiebeheerder is voorafgaand aan een realisatietraject afgestemd met Klant/Broker en SP/CS<br>Wanneer een nieuwe Applicatiebeheerpartij het beheer overneemt van de bestaande Applicatiebeheerpartij, is de bestaande Applicatiebeheerpartij verantwoordelijk voor de overdracht. Dit is inclusief de daarbij behorende voorwaarden zoals in dit document verwoord. |

## Applicatie Beveiligen

Applicaties die op het SP als Docker Images worden aangeleverd en als Docker Containers worden uitgerold, moeten veilig zijn.

### Patches

De container “Host OS” en de Images worden op regelmatige wijze gepatcht zodat deze aan de laatste beveiligingsstandaarden voldoen.

| Eis | Omschrijving |
|---|---|
| CNA044 | “Host OS” patches<br>De Applicatieleverancier dient rekening te houden met periodieke patches op het Host OS. Deze patches worden door SP/CS aangekondigd en op geleidelijke wijze op de omgevingen van de OTAP-straat toegepast. |

| | |
|---|---|
| CNA045 | “Image patches”<br>De Applicatieleverancier is verantwoordelijk voor het adequaat upgraden/patchen/uitrollen van Images die als “Bring Your Own Component” worden toegepast, en voor het adequaat uitrollen van bijgewerkte Images die als SP Bouwsteen worden toegepast (zie ook hoofdstuk 8). |

### Toegang

Applicaties dienen op veilige wijze benaderd te worden.

| Eis | Omschrijving |
|---|---|
| CNA046 | Load Balancer De applicatie wordt via een Load Balancer op internet ontsloten. Deze Load Balancer wordt door SP/CS beheerd. De Applicatieleverancier dient rekening te houden met de volgende eigenschappen van de Load Balancer. De Load Balancer:<br>• Bevindt zich in een apart netwerksegment [DMZ], zodat het netwerk van de applicatie zelf niet direct op internet toegankelijk is.<br>• Ontsluit business applicaties naar Internet, Rijksweb en Diginetwerk.<br>• Ondersteunt alleen de netwerkprotocollen HTTP en HTTPS, op de standaardpoorten 80 en 443; elk HTTP-verzoek wordt omgezet in een verzoek voor HTTPS [OWASP HSTS].<br>• Verbiedt toegang door middel van SSH, zodat kwaadwillenden geen toegang kunnen krijgen tot draaiende VMs of containers. |
| CNA047 | Reverse Proxy De applicatie wordt door middel van een Reverse Proxy vanuit het cluster ontsloten. Deze Reverse Proxy wordt door Applicatieleverancier in Test, en door Applicatiebeheer in Productie beheerd [Kubernetes Ingress]. De Reverse Proxy wordt binnen de Namespace gebruikt voor het ontsluiten van Kubernetes Services naar de Load Balancer. |

### Kwetsbaarheidanalyse

Applicaties dienen te worden getest op kwetsbaarheden, zodat veiligheid op het SP is gegarandeerd.

| Eis | Omschrijving |
|---|---|
| CNA048 | De broncode van de applicatie wordt door SP/CS periodiek nagezien door middel van een “Software Compositie Analyse” Zodat veilig gebruik gemaakt wordt van (open source) libraries van derde partijen [SCA]. Indien er zich fouten voordoen, dan worden deze door de Applicatieleverancier opgelost. |
| CNA049 | Images worden SP/CS periodiek getest op algemeen bekende kwetsbaarheden (“Common Vulnerabilities Exposures”/CVEs) Indien er zich fouten voordoen, dan worden deze door de Applicatieleverancier opgelost. |

### Wet- en regelgeving

Applicaties moeten voldoen aan wet- en regelgeving, en aan overige “best-practices” op het gebied van beveiliging.

| Eis | Omschrijving |
|---|---|
| CNA050 | Applicaties voldoen aan de [BIR 2017], en vanaf 2019 aan de [BIO] |
| CNA051 | Applicaties voldoen aan de [AVG] |
| CNA052 | Applicaties voldoen aan het [SP Beleid Acceptabel Gebruik] |
| CNA053 | Applicaties voldoen aan de [NCSC Beveiligingsrichtlijnen voor Webapplicaties] |
| CNA054 | Applicaties voldoen aan de richtlijnen van de [NIST Application Container Security Guide] |
| CNA055 | Applicaties voldoen aan de licentievoorwaarden van de toegepaste componenten van derden Dit betreft zowel open source licenties als eventuele commerciële licenties. Zie ook hoofdstuk 'Categorisering van Componenten'. |

## Categorisering van Componenten

Het SP categoriseert de verschillende soorten componenten die op het SP (kunnen) draaien: SP Basis, SP Dienst, SP Bouwsteen, SP Lab, en Bring Your Own Component (BYOC).

### Definities

- **SP Basis**: Dit zijn basis uitrol & beheer componenten die het SP standaard aflevert en die klanten verplicht moeten gebruiken. Bijvoorbeeld de tools van de CI/CD straat zoals GitLab en Harbor. Deze componenten zijn nodig voor een minimale functionaliteit van het SP.
- **SP Diens**t: Dit zijn componenten die het SP onderhoudt, uitrolt en beheert. Klanten kunnen deze componenten als dienst gebruiken zonder deze technisch te hoeven onderhouden; ook wel “as a service” genoemd.
- **SP Bouwstee**n: Dit zijn componenten waarin het SP de oorspronkelijke vorm up-to-date houdt qua upgrades & patches, maar zelf niet beheert of uitrolt. Deze zijn bedoeld om door klanten verder geconfigureerd en geïntegreerd te worden in de eigen applicaties. In het kader van zogeheten “co-creatie” is het wenselijk dat klanten bruikbare verrijkingen op deze componenten teruggeven aan de SP Community.
- **SP Lab**: Dit zijn componenten uit het Lab die bedoeld zijn om klanten een voorbeeld te geven. Deze componenten worden tijdelijk aangehouden totdat ze promoveren naar een andere vorm of niet meer nodig zijn.
- **Bring Your Own Component (BYOC)**: Dit zijn zelfstandige componenten die klanten als “Docker” image elders vandaan halen en uitrollen op het SP.

## Verantwoordelijkheden

Bovenstaande definities leiden tot de volgende verantwoordelijkheden.

Verantwoordelijkheden

| Categorie | Aanleveren, upgraden, patchen en licenties | Uitrollen | Operationeel beheren |
|---|---|---|---|
| SP Basis | SP | SP | SP |
| SP Dienst | SP | SP | SP |
| SP Bouwsteen | SP | Applicatiebeheer | Applicatiebeheer |
| SP Lab | SP | SP | SP |
| BYOC | Applicatieleverancier | Applicatiebeheer | Applicatiebeheer |

Bijvoorbeeld:

| Categorie | Voorbeeld |
|---|---|
| SP Basis | GitLab, Harbor, Prometheus/Grafana |
| SP Dienst | WSO2 IS eHerkenning, PostgreSQL |
| SP Bouwsteen | WSO2 IS, WSO2 APIM |
| SP Lab | Suite CRM, Alfresco DMS, WordPress CMS, Pega ZGW |
| BYOC | Python Django, Spring Boot, JBoss, … |

## Voorwaarden

Het SP hanteert een aantal voorwaarden bij de keuze en opname van componenten. Leveranciers en implementatiepartners van de klant kunnen onder strikte voorwaarden zelf het BYOC-concept toepassen. Een evaluatiematrix geeft aan per soort component welke voorwaarden er van toepassing zijn.

#### Uitgangspunten

De filosofie van het SP gaat uit van standaardisatie en zo veel als mogelijk hergebruik (samengebruik). Zie hiervoor ook de [I-strategie Rijk 2019- 2021]. Daarom moedigt het SP het gebruik van standaard componenten en het hergebruik van bestaande componenten aan.

Mede om deze filosofie te waarborgen en een “wildgroei” van componenten te voorkomen, hanteert het SP de volgende uitgangspunten:

1. Hergebruik: de voorkeur van het SP ligt in het hergebruik van bestaande (reeds eerder toegepaste) componenten. Er wordt gekeken of (soortgelijke) componenten al aanwezig zijn.
2. CNCF “tenzij”: het component is bij voorkeur onderdeel van de [CNCF landscape].
3. Open Source: er wordt gebruik gemaakt van aantoonbare "courante open source, tenzij", zodat de overdraagbaarheid naar een andere leverancier of applicatie-beheerpartij gegarandeerd blijft. Met “courante open source” wordt bedoeld:
    1. Component is algemeen bekend in de markt.
    2. Component wordt al (veel) in productie gebruikt.
    3. Component wordt actief onderhouden. Dit is bijvoorbeeld te valideren door het raadplegen van recente Blogs, "Commits" op [GitHub] etc.
4. Community: er zijn voldoende softwareontwikkelaars beschikbaar, zodat de overdraagbaarheid/doorontwikkeling van het component is geborgd.
5. Expertise & capaciteit SP: er zijn voldoende softwareontwikkelaars beschikbaar bij het SP, zodat het component ondersteund kan worden.
6. Docker/Kubernetes op Linux host: component draait in Docker/Kubernetes op een Linux Host.
7. Linux Image: de Docker Image van het component zelf is Linux- gebaseerd.
8. Geen vervanging SP Basis: component vervangt geen SP Basis Component.
9. Toets SP: component wordt door het SP getoetst.
10. Beveiliging: afhankelijk van het soort component is het SP of de leverancier/implementatiepartner (bij BYOC) verantwoordelijk voor security, upgrades & patches. Ten alle tijd is dit belegd en ingeregeld.
11. Business Applicaties: het SP is bedoeld voor Business Applicaties

#### Evaluatiematrix

Bovenstaande uitgangspunten leiden tot de volgende evaluatiematrix:

| Categorie | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **SP Dienst** | E | E | E | E | E | E | E | E | E | E | E |
| **SP Bouwsteen** | E | E | E | E | E | E | E | E | E | E | E |
| **SP Lab** | - | - | A | E | - | E | - | E | E | E | E |
| **BYOC** | - | - | A | A | - | E | - | E | E | E | - |

1: Hergebruik
2: CNCF “tenzij”
3: Open Source
4: Community
5: Expertise & capaciteit SP

6: Docker/Kubernetes op Linux host
7: Linux Image
8: Geen vervanging van SP Basis
9: Toets door SP/CS
10: Beveiliging
11: Business Applicaties

E: eis
A: geen eis, wel geadviseerd
-: geen eis, neutraal

**Promotiepad**

De componenten van het SP kunnen promoveren naar een andere categorie; bijvoorbeeld een bouwsteen zou op termijn een dienst kunnen worden. Het standaard pad is BYOC > SP Lab > SP Bouwsteen > SP Dienst.

De mate van interesse in zo'n component en de mate waarin de criteria van de evaluatiematrix worden vervuld, bepalen of een component daadwerkelijk promoveert.

Een Bring Your Own Component hoeft in de evaluatiematrix aan de minste SP eisen te voldoen. Uiteraard moet een aantal zaken goed geregeld zijn, zoals beveiliging, maar de klant is hier redelijk vrij in zijn keuze. Echter, is de klant van plan om het SP te vragen om op termijn dit component te promoveren naar een Bouwsteen (om bijvoorbeeld hergebruik binnen de SP Community te promoten) dan moet het Bring Your Own Component ook voldoen aan alle eisen van een SP Bouwsteen component. Tijdens de keuze van een BYOC dient hier rekening mee gehouden te worden.