

Applicatie analyse Get-It

Inhoudsopgave

Samenvatting	3
Huidige situatie	3
Datastromen schematische weergave.....	3
Weergave tbv Spoke inrichting, technische componenten Productieomgeving.....	4
Operationele aspecten.....	4
Beschikbaarheid.....	4
Beheerprocessen (logging, monitoring).....	4
Disaster Recovery inrichting	5
Operating Model.....	Fout! Bladwijzer niet gedefinieerd.
Ontwikkel Test Acceptatie	5
Gewenste doelarchitectuur en migratie.....	5
User Story.....	5
Definition of Done.....	5

Samenvatting

De applicatie Get-It zorgt ervoor dat een bezoeker van de website bibliotheek.nl verwezen wordt naar zijn of haar lokale bibliotheek. Get-It haalt alleen de gegevens, middels HTTP verbindingen uit G!DS. Het systeem van de lokale bibliotheken wordt ook wel de ILS systemen genoemd.

In de nieuwe situatie, op het Azure platform, wenst de KB het Functioneel Applicatie Beheer te continueren en het technische applicatie beheer als PaaS dienst van Azure af te nemen. De door Get-It geleverde API wordt dan aangeboden via de API-Gateway.

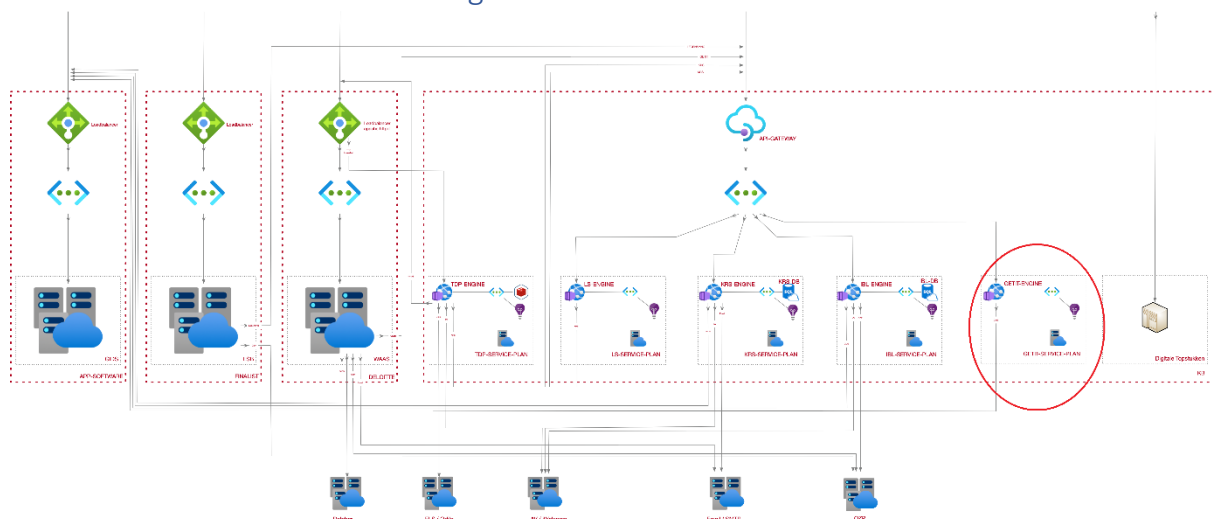
Huidige situatie

De Get-It engine is een REST interface dat en wordt op dit moment via de huidige API gateway aangeroepen door de WaaS implementatie die het core systeem levert voor de Content van bibliotheek.nl. De WAAS gebruikt Get-It om voor een gebruiker de dichtstbijzijnde lokale bibliotheek te vinden op basis van postcode.

Authenticatie van de WaaS requests gebeurt op de API-gateway middels een API-key.

Het is een eenvoudige engine die een kritische functie vervult.

Datastromen schematische weergave



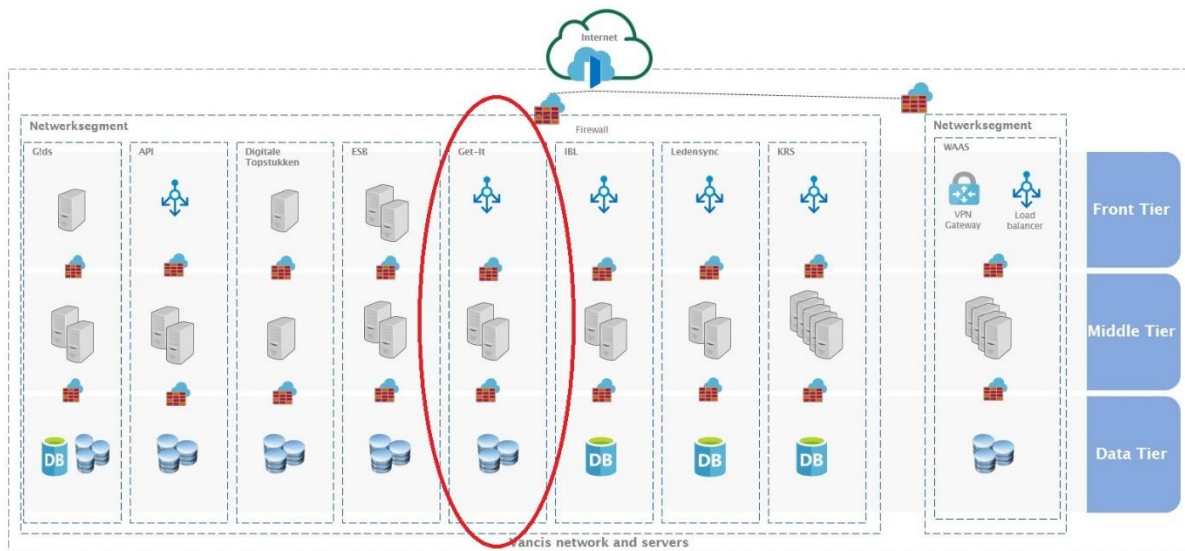
De Get-It engine kan niet vanaf alle locaties worden aangeroepen. Iedere aanroep gaat via de API-gateway realiseert de restricties op de toegang vanaf een locatie. Op dit moment is de WAAS omgeving degene die via de API-gateway een directe verbinding met Get-It heeft.

De Get-It engine verzorgt dan (technisch) de afhandeling van de requests door data uit de G!DS applicatie op te halen, lokaal te cachen, en uiteindelijk op te leveren aan de aanvrager.

Verder kunnen de beheerders vanuit de KB, middels een vaste VPN verbinding een sessie opzetten naar de servers in de middleware laag.

Alle lagen, zoals deze schematisch zijn weergegeven, worden gescheiden middels een, door Vancis beheerde, firewall.

Weergave tbv Spoke inrichting, technische componenten Productieomgeving



De Front tier wordt door Vancis geleverd. Het is een Shared Service Loadbalancer configuratie die zorg draagt voor het afhandelen van de HTTP requests.

De Middle tier laag bestaat uit 2 servers. Hierop is java als middleware laag geïnstalleerd. Dropwizard wordt gebruikt om API functionaliteit te ontwikkelen. De applicatie wordt aangeleverd als een stand-alone jar en een gescheiden configuratie.

Er is geen additionele data opslag laag, alle server data staat in de VM op het VSAN.

Operationele aspecten

Operating Model

In de huidige opzet is het Functioneel en Technisch Beheer van de applicatie belegd bij de KB en het beheer van de OS laag is belegd bij Vancis. In de nieuwe situatie is het gewenst dat de OS laag middels containers en een PaaS oplossing worden verwezenlijkt.

Beschikbaarheid.

De beschikbaarheidseisen aan de applicatie zijn niet heel scherp gedefinieerd. Vancis heeft wel servicelevels, deze gelden aan de onderliggende infrastructuur componenten:

De twee servers (VMs) zijn standaard servers, (2 vCPU, 4 GB en 50G OS storage) deze hebben een beschikbaarheidspercentage van 99,8 % (conform Vancis SLA)

In de praktijk heeft zich nog nooit een calamiteit en/of grote verstoring voorgedaan. Voor deze applicatie geldt dan ook dat dit niveau als voldoende worden ervaren.

Beheerprocessen (logging, monitoring)

Logging op de infrastructuur componenten (tot en met OS laag) wordt uitgevoerd door Vancis.

Indien relevant genereren zij ook incidentmeldingen welke worden geregistreerd bij de Service Desk van de KB. De KB gebruikt hiervoor op dit moment Topdesk.

De applicatie logging van de Get-It engine wordt beheerd door technisch beheer van de KB.

Daarnaast monitored de Service Desk van de KB ook de systemen, op ongeveer dezelfde paramaters (disk, CPU, processen etc.).

Er is een rudimentair SIEM proces, nog geen volledige security implementatie. Er vindt geen proactieve Informatie Beveiliging plaats.

Disaster Recovery inrichting

Er zijn geen RTO's en RPO's voor de Get-It engine gedefinieerd. Deze worden echter wel verwacht in de Azure omgeving.

De servers worden geleverd op een virtuele omgeving in een Active-Passive cluster configuratie en middels snapshots continue geborgd. Vancis heeft hier geen contractuele KPI's op afgesloten.

Alle overige zaken die bij DR horen (bijvoorbeeld gebruikersoefeningen, of applicatie beschikbaarheidsgarantie) zijn niet ingericht.

Ontwikkel Test Acceptatie

De applicatie kent een Test en Acceptatie omgeving welke gebruikt wordt voor implementatie van nieuwe API functionaliteit. De infrastructuurinrichting van deze omgeving verschilt (Test is een single instance, Acceptatie zijn twee servers) Qua grootte zijn de servers identiek aan de Productie servers.

TA en P omgevingen zijn ook gescheiden middels, een door Vancis beheerde, firewall.

Er is geen standaard test en acceptatieproces geïdentificeerd. Hiermee bedoelen we een standaard proces van wie welke aanpassing mag testen en goedkeuren voordat deze naar de volgende acceptatiestap gaat. Dit proces wordt onder begeleiding van de KB aangestuurd.

Gewenste doelarchitectuur en migratie

Voor deze applicatie wenst de KB toe te gaan naar een applicatie container oplossing waarbij de dienst als een PaaS dienst wordt aangeboden. De KB zal hier de applicatie container bestaande uit de stand-alone jar en voor T/A/P relevante configuratie aanleveren. De java code blijft in beheer bij de KB.

De KB vraagt hier aan de leverancier om hier een model te beschrijven hoe de samenwerking tussen de beide partijen in de Migratie en Managed Service fase het meest succesvol zal zijn.

De KB is primair op zoek naar een leverancier die helder communiceert wat er nodig is om succesvol in de Cloud te landen. De KB verwacht dan ook van de leverancier dat deze proactief de KB wijst op haar taken en verantwoordelijkheden, terwijl zij volledig invulling geeft aan de aan haar gestelde eisen.

User Story

In mijn rol als **API ontwikkelaar/GetIt applicatie** verwacht ik dat de Cloud diensten mij gaat ontzorgen in de ontwikkeling en deployment, door automatiseren of toepassen van best practices, zonder afbreuk te doen aan de beveiligingseisen die de Koninklijke Bibliotheek aan mij oplegt.

Definition of Done

De generieke eisen gesteld aan deze applicatie en de migratie naar de Cloud zijn de volgende:

- De applicatie levert in de nieuwe, gemigreerde, situatie dezelfde functionaliteiten als hierboven beschreven.
- Beschikbaarheid, monitoring en rapportage functionaliteiten voor deze applicatie zijn in de nieuwe, gemigreerde, situatie minimaal van hetzelfde niveau als voor de migratie.

- Logging en monitoring voor deze applicatie levert, na migratie, direct inzicht in de gemaakte kosten voor deze applicatie.
- Alle betrokken partijen, benodigd voor het goed functioneren van de applicatie, zijn bekend en op de hoogte van elkaars rol en verantwoordelijkheden.
- De rol en verantwoordelijkheden van alle betrokken partijen, benodigd voor het goed functioneren van de applicatie, zijn beschreven en overeengekomen.
- De gemigreerde applicatie is op zijn minst getest en geaccepteerd door de leverancier (Technische Applicatie Test), in dit geval de Koninklijke Bibliotheek.
- De gemigreerde applicatie is op zijn minst getest en geaccepteerd door de gebruiker (Functionele Applicatie Test), nog nader te bepalen door de product owner van de API gateway.