Analytics in Azure – week 4

In de vierde week van het leertraject Analytics in Azure gaan we aan de slag met Infra-as-Code. We hebben veel nieuwe dingen gezien op het gebied van IaC en beheertools als *az cli*.

De belangrijkste opdracht voor deze week is het uitrolbaar maken van je infrastructuur: idealiter zou je een nieuwe test-omgeving ook qua infrastructuur eenvoudig willen kunnen “uitrollen”. Ook willen we deze uiteraard goed beveiligen met de nieuwe kennis over Virtual Networks en Private Endpoints, maar uiteraard het liefst vanuit Terraform, bicep of ARM.

Ook is er deze week weer flink wat verdieping mogelijk op het gebied van Azure ARM (probeer je dataplatform met een ‘deploy to Azure’ button uitrolbaar te krijgen), Azure bicep, Terraform (probeer je terraform-scripts modulair te maken). Leuke bonus is natuurlijk om een bestaande productiedatabase bij een nieuwe uitrol ook automatisch te kunnen “restoren”.

Volgende week hebben we een totaal nieuw onderwerp: Streaming Data. De kennis die we tot nu toe opgedaan hebben, kunnen we dan uitbreiden op elk niveau.

Links van vandaag:

* [Azure Data Factory Naming Conventions - Erwin & BUSINESS ANALYTICS (erwindekreuk.com)](https://erwindekreuk.com/2020/07/azure-data-factory-naming-conventions/)
* [Azure Managed Identity Authentication for Azure SQL DB using Azure data factory | by Ashish Shukla | Analytics Vidhya | Medium](https://medium.com/analytics-vidhya/azure-managed-identity-authentication-for-azure-sql-db-using-azure-data-factory-d8b13afb19e7)

# Opdracht

* Breng je Proof of Concept onder in IaC. Het gaat hier niet primair om de ontwikkelomgeving, maar het liefst wil je in elk geval een testomgeving kunnen weggooien wanneer deze niet voldoet – en weer neerzetten met een druk op de knop.
* Probeer dit te doen in Azure DevOps Pipelines. Je kunt een demo-pipeline met Terraform vinden in het project ’04-data-project-pipelines’ (de pipeline in kwestie heet “infra-synapse-terraform.yml”).
  + In je Azure DevOps heb je hier de “Azure Pipeline Terraform Tasks” extensie voor nodig van Charles Zipp. Deze vind je kosteloos in de Marketplace.
* Wanneer de basis staat, kijk dan of je deze kunt uitbreiden:
  + Voeg een Azure Key Vault toe in je PoC, en breng hier je secrets in onder. Koppel deze aan de ADF.
  + Probeer nu je Terraform uit te breiden met Azure KeyVault. Zorg ervoor dat er in elk geval dummy-waarden met de juiste keys aanwezig zijn.
* Wanneer je ook deze koppeling op zijn plaats hebt en kunt gebruiken, zou je ook kunnen kijken naar het opzetten van Private Endpoints. [Het is eenvoudiger dan je denkt in Terraform](https://registry.terraform.io/providers/hashicorp/azurerm/latest/docs/resources/data_factory_managed_private_endpoint)!

# Verdieping / verbreding

## Az CLI

De Az CLI is kort aan bod gekomen vandaag, maar erg handig voor allerlei aanhangend werk: configuraties die je bijvoorbeeld niet in ARM, Bicep of Terraform voor elkaar krijgt. Of voor ad-hoc deployments waar je “even” een paar storage accounts nodig hebt. Of op diverse omgevingen een firewall moet inrichten.

* Op [Create a storage account - Azure Storage | Microsoft Docs](https://docs.microsoft.com/en-us/azure/storage/common/storage-account-create?tabs=azure-cli) staat eenvoudige start: het aanmaken, bekijken en verwijderen van een storage account.
* [Get started with Azure Command-Line Interface (CLI) | Microsoft Docs](https://docs.microsoft.com/en-us/cli/azure/get-started-with-azure-cli) biedt wat meer overzicht
* Als je toe bent aan iets stevigers, kun je op [Azure CLI example: Create a single database - Azure SQL Database | Microsoft Docs](https://docs.microsoft.com/en-us/azure/azure-sql/database/scripts/create-and-configure-database-cli) lezen hoe je een database-server, -firewall en -database kunt configureren en aanmaken.
  + Er zitten hier wat zaken als variabelen in het shellscript als location="East US". Lees er dus even rustig doorheen of je begrijpt wat er gebeurt
* Wil je iets fundamenteler zien wat je in bijna alle “az” commando’s kunt? Op deze website staat een prima artikel dat je bij de hand neemt: [Getting Started with the Azure CLI (adamtheautomator.com)](https://adamtheautomator.com/azure-cli/)

## Bicep

Wanneer je weet hoe je in Terraform een deployment doet, is bicep niet heel veel moeilijker: de syntax lijkt er vrij veel op. Kijk bijvoorbeeld eens hier: [Bicep language for deploying Azure resources - Azure Resource Manager | Microsoft Docs](https://docs.microsoft.com/en-us/azure/azure-resource-manager/bicep/overview?tabs=bicep#get-started). Check ook even de aannames en claims van mij – kan Bicep bestaande resources inderdaad niet updaten? En is het alleen een “compile-slag” naar ARM? Of is er inmiddels nieuwe ontwikkeling op dit gebied? (Vertrouw nooit blind op de trainer ;-))

.. en, als je Bicep draaiend hebt, dan kun je eens kijken [of je een bestaande ARM-template kunt ‘decompilen’](https://docs.microsoft.com/en-us/azure/azure-resource-manager/bicep/decompile?tabs=azure-cli).

## Deploy to Azure button

Wanneer je je ARM template onderbrengt in een openbare Git repository (bijvoorbeeld op GitHub), is het eenvoudig om er een “Deploy to Azure” van te maken die iedereen kan gebruiken.

Maak een eenvoudig ARM template (bijvoorbeeld van één storage account), en probeer dit uit te rollen via “Deploy to Azure”. Hier staat beschreven hoe je dat doet: [Deploy to Azure button - Azure Resource Manager | Microsoft Docs](https://docs.microsoft.com/en-us/azure/azure-resource-manager/templates/deploy-to-azure-button)

## Terraform – modules

Wanneer je start met Terraform is het verleidelijk om je omgeving in één script te vangen. De kracht van Terraform is echter ook dat het modulair is. Kijk hoe je je script verder kan opbreken, en hoe je de modules makkelijk kunt inzetten op andere projecten. Het zou kunnen dat je hier wat (kleine) nieuwe Terraform-concepten moet uitzoeken, dus zorg ervoor dat je in elk geval de eerste opdracht klaar hebt en een beetje thuis bent in Terraform!

[Creating Modules | Terraform by HashiCorp](https://www.terraform.io/language/modules/develop)

## Restore van een bestaande productiedatabase

Infrastructuur is één ding, maar data een tweede belangrijke. Kun je bij IaC ook een database standaard laten inrichten vanuit een bestaande database?

Azure heeft hier mogelijkheden voor, bijvoorbeeld door een *restore* vanuit een backup, of een database-*kopie*. Beide functies zijn in elk geval binnen *az sql* beschikbaar:

* Copy: [az sql db | Microsoft Docs](https://docs.microsoft.com/nl-nl/cli/azure/sql/db?view=azure-cli-latest#az-sql-db-copy)
* Restore: [az sql db | Microsoft Docs](https://docs.microsoft.com/nl-nl/cli/azure/sql/db?view=azure-cli-latest#az-sql-db-restore)

(Copy is in mijn persoonlijke ervaring sneller, maar ik heb niet heel veel moeite gedaan om de restore te optimaliseren)

Implementeer dit in je DevOps pipeline, zodat je testomgeving na deployment ook vult met inhoud vanuit een bestaande backup.

Kun je deze stap ook integreren met Terraform of ARM/bicep? (Voor een restore weet ik dat het kan in ARM, maar ik heb zelf nog niet uitgezocht of het kan in Terraform, of dat de copy in ARM of Terraform beschikbaar is)

## Security

Echt weten hoe het nu zit met security en netwerken? Ook daar kun je je natuurlijk verder in verdiepen. Op <https://docs.microsoft.com/en-us/azure/virtual-network/tutorial-filter-network-traffic> staat een tutorial waarin je twee VM’s aanmaakt om vervolgens het netwerkverkeer ertussen te filteren.

Ook zijn we niet heel ver gegaan op het gebied van subnets, NSG’s, en zaken als VNet peering: als je hier meer feeling mee wilt krijgen én je Azure-skills verbeteren is het altijd leuk om hier kleine testjes voor te bouwen:

* Als ik de twee VM’s uit het voorbeeld hierboven in verschillende subnets onderbreng, kan ik ze dan alsnog met elkaar laten praten?
* Kan ik vanuit een VM een database benaderen zonder public endpoints? (Tip: hoe configureer ik mijn VNet, en waar heeft mijn VM zijn netwerkadapters nodig)
* Kan ik een VPN Gateway opzetten? (Prijs voor basic: € 0,04 / uur)
* Lukt het om een “transit netwerk” [in te richten](https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-peering-gateway-transit)?

.. en uiteraard, wanneer je hier een mooie “secure” oplossing voor gevonden hebt: is dit in IaC onder te brengen?