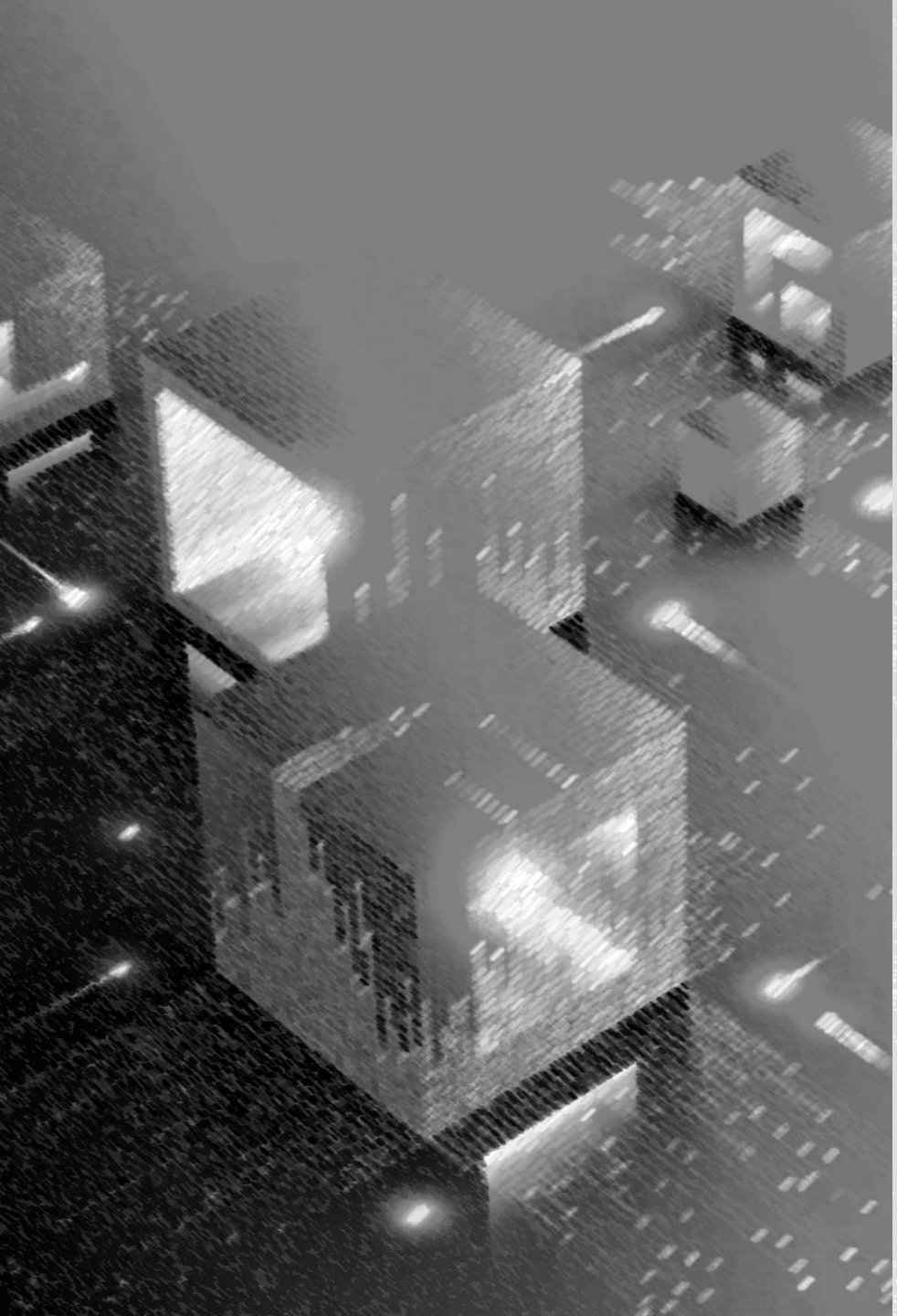
# CLOUD-ARCHITEKT PRAXISBEISPIELE

MSC, WAEL AMER

UMSETZUNG VON CLOUD-INFRASTRUKTUR-LÖSUNGEN

# AGENDA

- Einführung
- Aufgabe 1: Terraform - Bereitstellung einer Azure Web App
- Aufgabe 2: Azure DevOps - CI/CD Pipeline
- Aufgabe 3: Netzwerkkonfiguration
- Aufgabe 4: Speicheraufgabe - Sicherer Azure Storage Account



# EINFÜHRUNG

- Die Aufgabe Fragen sind im GitHub-Repository hochgeladen.
  GitHub - waelamer/Azure-Cloud-Praxisbeispiele
- Infrastructure as Code (IaC) wird für alle Aufgaben bereitgestellt:
  - **Q1**: Terraform für die Bereitstellung einer Azure Web App.
  - **Q2**: CI/CD-Pipeline mit Azure DevOps.
  - **Q3**: Netzwerkkonfiguration in Azure.
  - **Q4**: Sicherer Azure Storage Account.

- Alle Fragen und Lösungen wurden in meiner eigenen Azure-Umgebung getestet.

# AUFGABE 1 - TERRAFORM: BEREITSTELLUNG EINER AZURE WEB APP

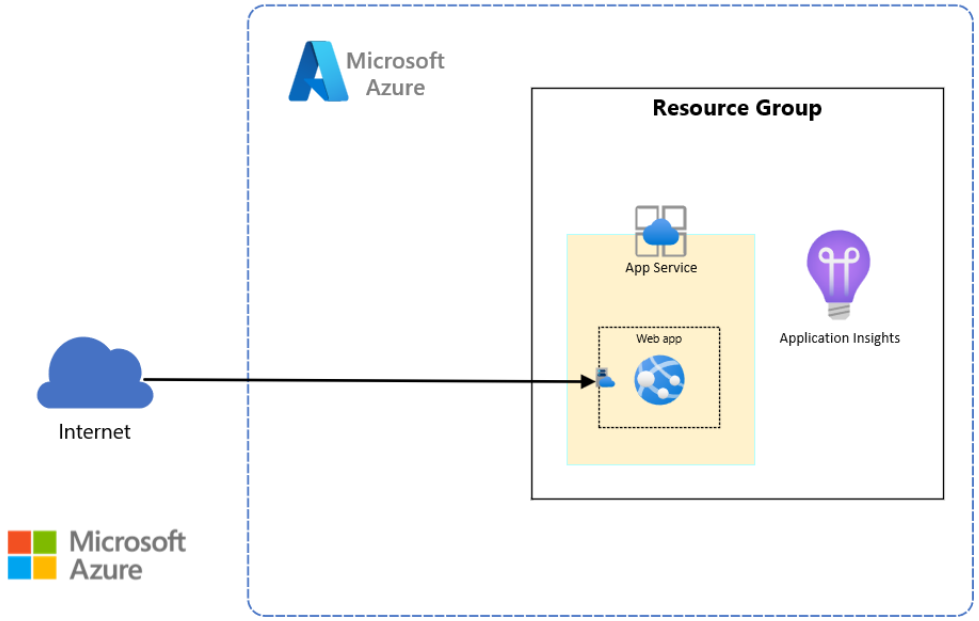
## Ziel:

- Verwenden von Terraform, um eine Web App in Azure bereitzustellen.

## Anforderungen:

- Erstellen einer neuen Resource Group.
- Bereitstellung eines App Service Plans im kostenfreien Tarif.
- Bereitstellung einer Web App, die eine einfache Webseite hosten kann.
- Verwendung von Variablen zur Konfiguration der App (z.B. App-Name, Region).

# LÖSUNGSANSATZ Q1.




Gemäß den Anforderungen für die Bereitstellung einer Azure Web App mit Terraform habe ich die Konfiguration wie folgt umgesetzt:

- **Ressourcengruppe**: Dynamisch erstellt, basierend auf dem App-Namen (var.web_app_name) und dem Standort (var.location).
- **App Service Plan (F1)**: Free-Tier Plan auf Linux-Basis, flexibel über die Variable var.sku_name konfigurierbar.
- **Web App Hosting**: Bereitgestellt mit azurerm_linux_web_app, parametrisiert für App-Name, Standort und Service-Plan. Skalierbar und anpassbar.
- **Konfigurationsvariablen**: Flexibel für web_app_name, location und sku_name. Output gibt den Hostnamen zurück.
- Zusätzlich habe ich Application Insights integriert und HTTPS erzwungen (https_only = true) für Sicherheit und Monitoring.

# AUFGABE 2 - CI/CD PIPELINE IN AZURE DEVOPS

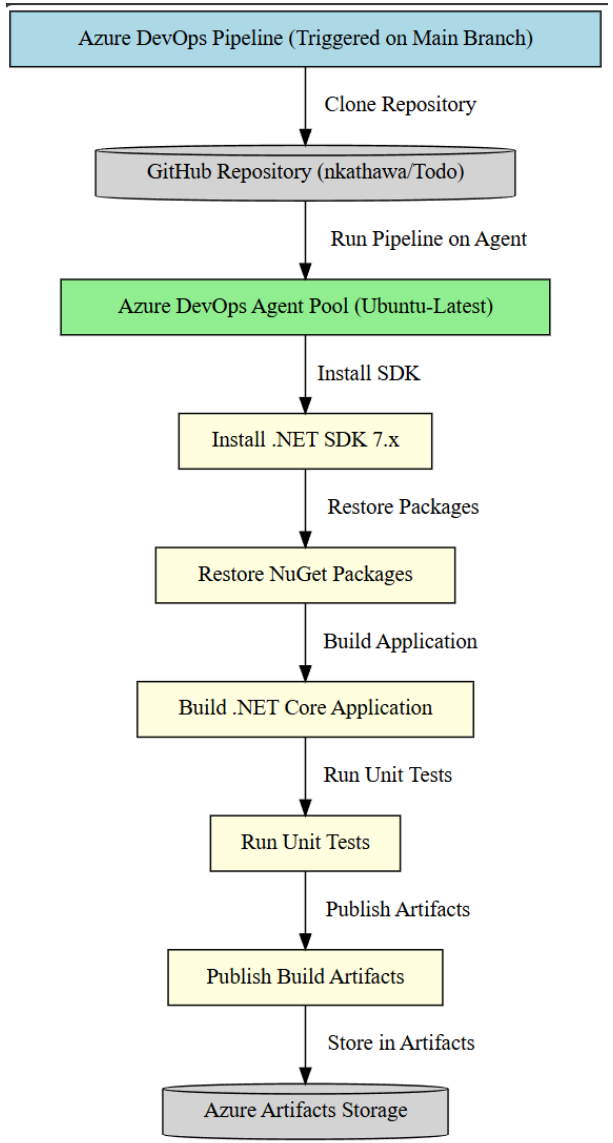
## Ziel:

- - Implementierung einer YAML-basierten CI/CD Pipeline für eine Web App.

## Anforderungen:

- Abrufen des Codes aus einem Git-Repository.
- Durchführen des Build-Prozesses.
- Bereitstellung der Anwendung in einer Azure Web App.
- Integrieren von Unit-Tests und Bereitstellung in Staging- und Produktionsumgebungen.

## Build-Pipeline Steps



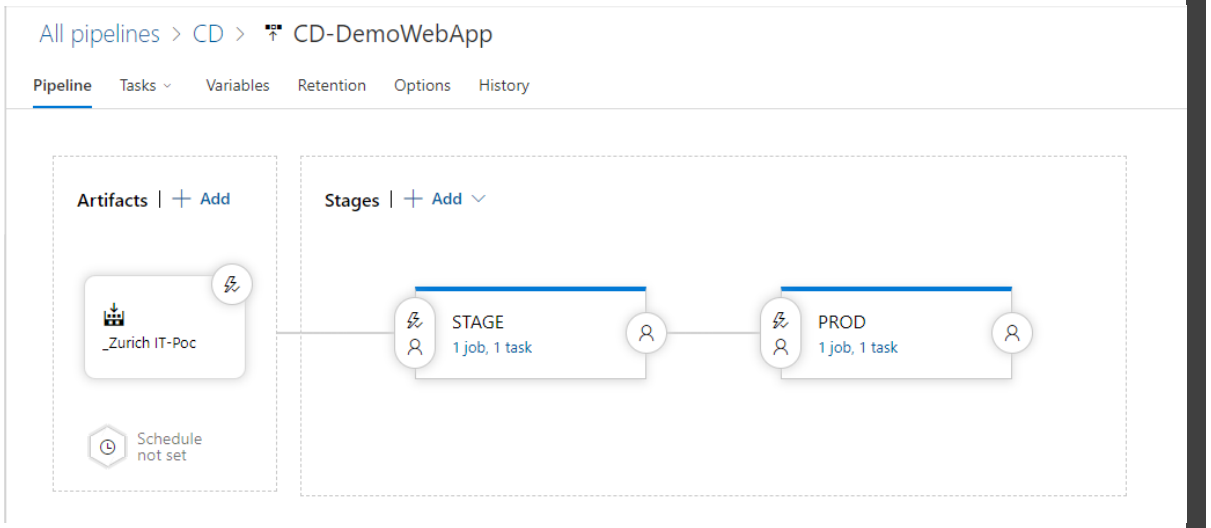

# LÖSUNGSANSATZ Q2.

**Um diese Aufgabe zu erfüllen, wird der CI/CD-Prozess in zwei Teile unterteilt: Build-Pipeline und Release-Pipeline.**

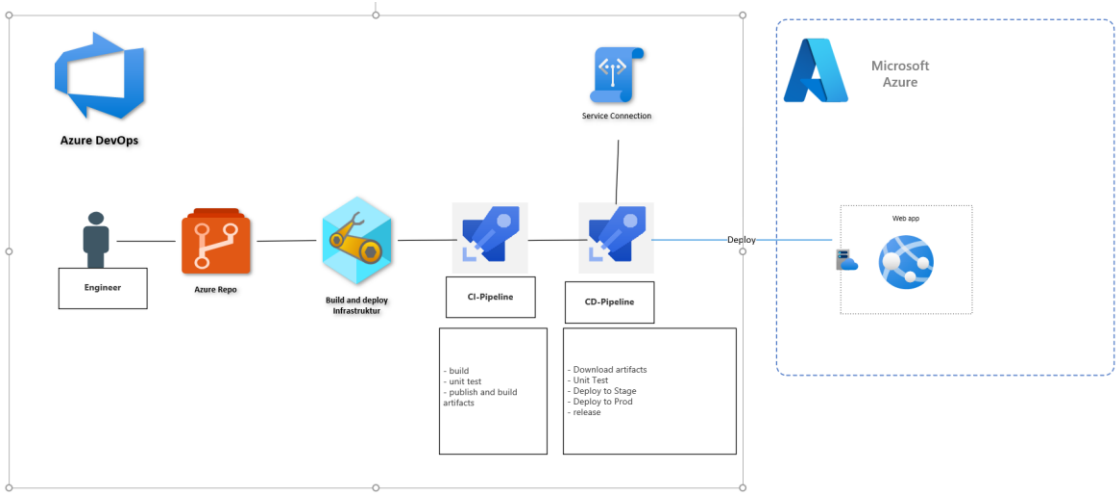
**Build-Pipeline steps:**

- **Checkout**: Holt den Quellcode aus dem Repository.
- **Clone External Repository**: Klont ein externes Repository, falls der Code nicht lokal ist.
- **UseDotNet**: Installiert die benötigte .NET Core SDK-Version.
- **Restore NuGet Packages**: Lädt Abhängigkeiten für das Projekt herunter.
- **Build**: Baut die Anwendung mit der vorgegebenen Konfiguration.
- **Test**: Führt Unit-Tests aus, um die Build-Qualität sicherzustellen.
- **Publish**: Veröffentlicht die Anwendung als Artefakte nach erfolgreichem Build.
- **PublishBuildArtifacts**: Lädt die Artefakte in ein Repository oder einen Speicherort zur späteren Bereitstellung hoch.

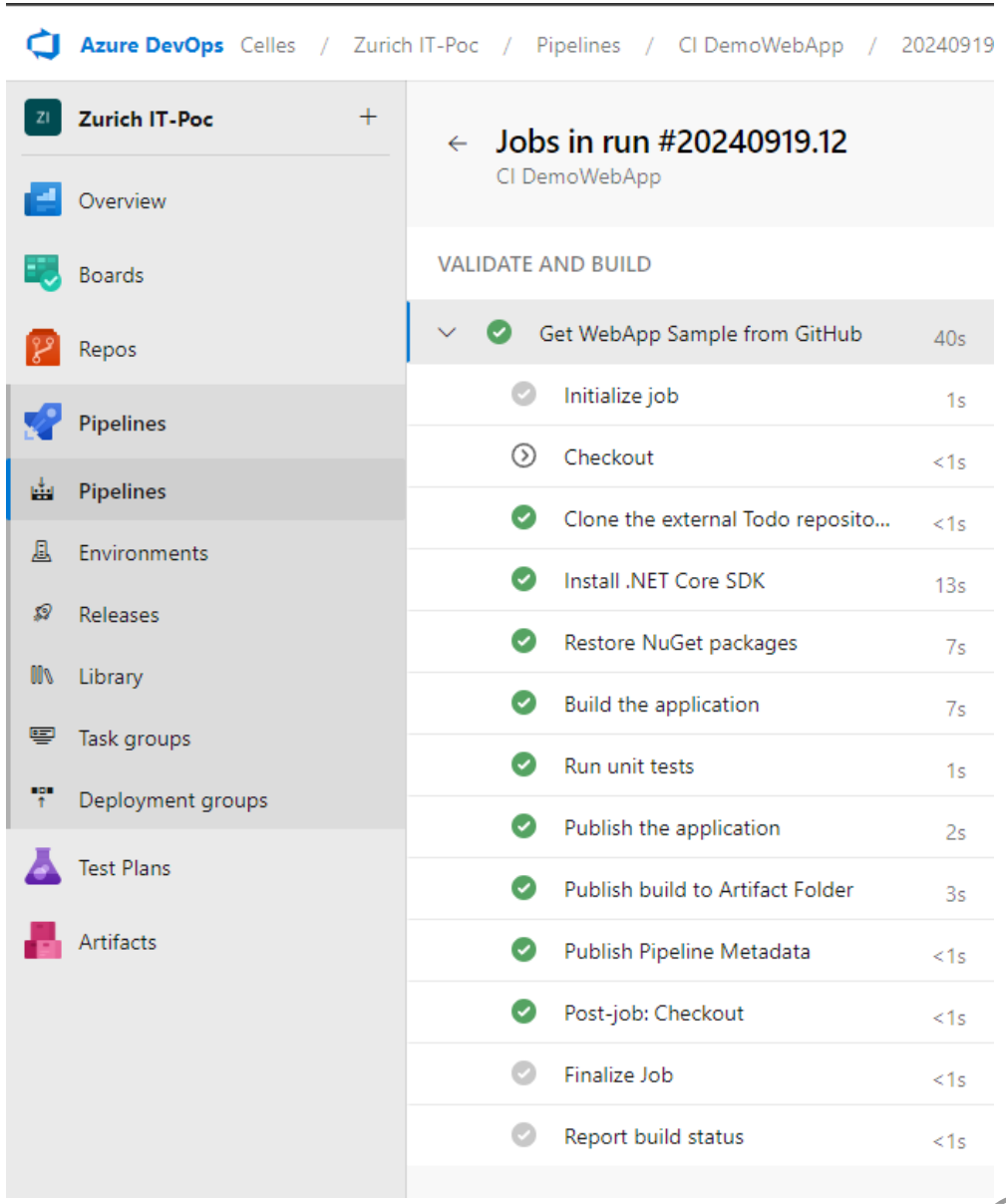
# Release-Pipeline




# Architecture




# Build-Pipeline



# AUFGABE 3 - IMPLEMENTIERUNG EINES VIRTUELLEN NETZWERKS

## Ziel:

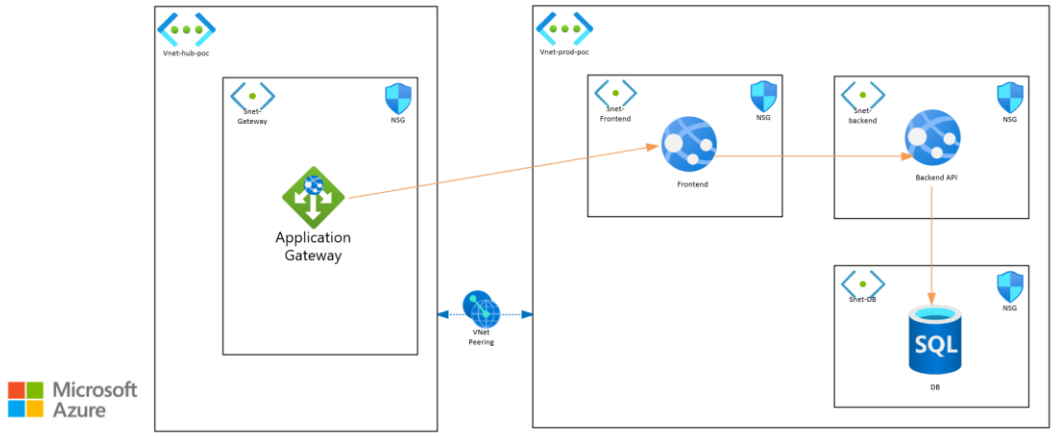
- - Erstellen eines virtuellen Netzwerks mit Subnetzen und einer Network Security Group (NSG).

## Anforderungen:

- Erstellen eines virtuellen Netzwerks mit mindestens zwei Subnetzen (Frontend und Backend).
- Einrichten einer NSG, die nur HTTP-Traffic ins Frontend-Subnetz erlaubt.
- Backend-Traffic nur vom Frontend-Subnetz zugänglich.
- Keine eingehenden Verbindungen zu Backend-Ressourcen zulassen.

# Lösungsansatz Q3.

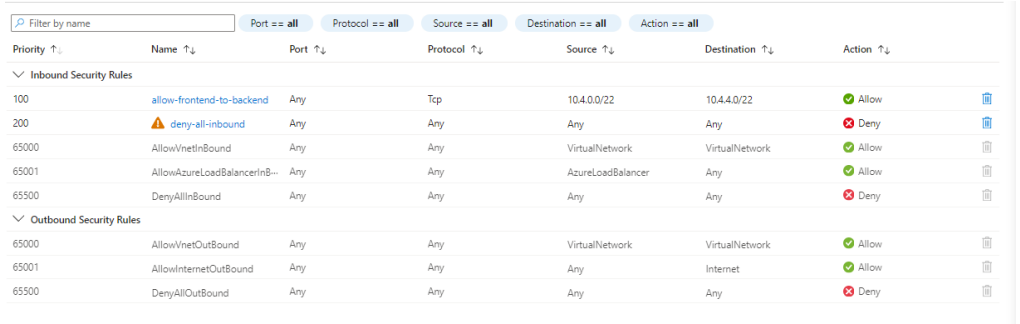
**Architecture**




- Die Terraform-Konfiguration entspricht den gestellten Anforderungen:

- 1. **Virtuelles Netzwerk und Subnetze**: Zwei virtuelle Netzwerke (Hub und Spoke) sind eingerichtet. Das Spoke-VNet enthält zwei Subnetze (Frontend und Backend), die im „prod“-Spoke-VNet angelegt sind, mit einer Verbindung zum „Hub“-Netzwerk.
- 2. **Netzwerksicherheitsgruppen**: Die NSG-Regeln sind wirksam implementiert, sodass HTTP-Traffic nur zum Frontend-Subnetz erlaubt ist, während der Zugriff auf das Backend-Subnetz nur aus dem Frontend-Subnetz gestattet wird.
- 3. **Sicherheit**: Das Backend-Subnetz ist durch eine "deny-all-inbound"-Regel geschützt, die sicherstellt, dass kein externer Zugriff möglich ist, außer vom Frontend.

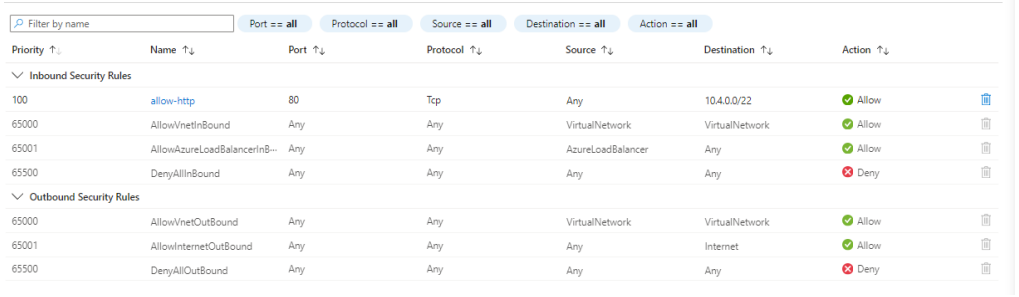
## nsg-backend-PoC



Filter by name | Port == all | Protocol == all | Source == all | Destination == all | Action == all

| Priority | Name | Port | Protocol | Source | Destination | Action |
|---|---|---|---|---|---|---|
| **Inbound Security Rules** | | | | | | |
| 100 | allow-frontend-to-backend | Any | Tcp | 10.4.0.0/22 | 10.4.4.0/22 | Allow |
| 200 | deny-all-inbound | Any | Any | Any | Any | Deny |
| 65000 | AllowVnetInBound | Any | Any | VirtualNetwork | VirtualNetwork | Allow |
| 65001 | AllowAzureLoadBalancerInB··· | Any | Any | AzureLoadBalancer | Any | Allow |
| 65500 | DenyAllInBound | Any | Any | Any | Any | Deny |
| **Outbound Security Rules** | | | | | | |
| 65000 | AllowVnetOutBound | Any | Any | VirtualNetwork | VirtualNetwork | Allow |
| 65001 | AllowInternetOutBound | Any | Any | Any | Internet | Allow |
| 65500 | DenyAllOutBound | Any | Any | Any | Any | Deny |

## nsg-frontend-PoC



Filter by name | Port == all | Protocol == all | Source == all | Destination == all | Action == all

| Priority | Name | Port | Protocol | Source | Destination | Action |
|---|---|---|---|---|---|---|
| **Inbound Security Rules** | | | | | | |
| 100 | allow-http | 80 | Tcp | Any | 10.4.0.0/22 | Allow |
| 65000 | AllowVnetInBound | Any | Any | VirtualNetwork | VirtualNetwork | Allow |
| 65001 | AllowAzureLoadBalancerInB··· | Any | Any | AzureLoadBalancer | Any | Allow |
| 65500 | DenyAllInBound | Any | Any | Any | Any | Deny |
| **Outbound Security Rules** | | | | | | |
| 65000 | AllowVnetOutBound | Any | Any | VirtualNetwork | VirtualNetwork | Allow |
| 65001 | AllowInternetOutBound | Any | Any | Any | Internet | Allow |
| 65500 | DenyAllOutBound | Any | Any | Any | Any | Deny |

# AUFGABE 4 - KONFIGURATION EINES SICHEREN STORAGE-ACCOUNTS

## Ziel:

- Konfigurieren eines Storage-Accounts mit Sicherheitsanforderungen.

## Anforderungen:

- Erstellen eines Storage-Accounts mit Blob-Verschlüsselung.
- Einrichtung einer Shared Access Signature (SAS) mit 1 Stunde Gültigkeit und Lesezugriff auf eine bestimmte Datei.
- Erstellen einer Lifecycle-Policy zur Archivierung älterer Blobs.

Microsoft Azure

Home > Microsoft Azure Sponsorship Startup | Resource groups > rg-zurichq4-poc > stzurichq4poc | Lifecycle management >

## Update a rule

Details | Base blobs | Filter set

A rule is made up of one or more conditions and actions that apply to the entire storage account. Optionally, specify that rules will apply to particular blobs by limiting with filters.

Rule name *

archive-old-blobs-policy

Rule scope *

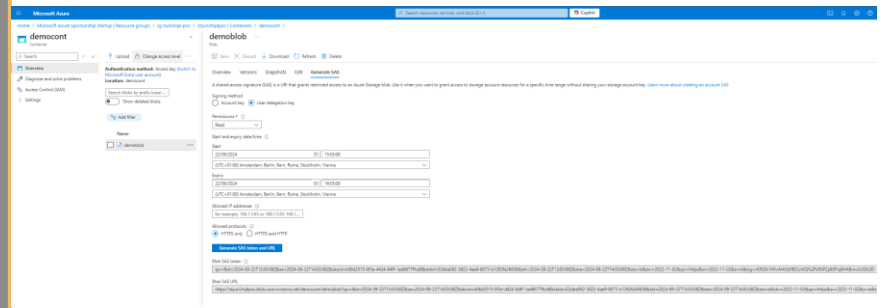
- ○ Apply rule to all blobs in your storage account
- ◉ Limit blobs with filters

Blob type *

- ☑ Block blobs
- ☐ Append blobs

Blob subtype *

- ☑ Base blobs
- ☐ Snapshots
- ☐ Versions



Microsoft Azure

Home > Microsoft Azure Sponsorship Startup | Resource groups > rg-zurichq4-poc > stzurichq4poc

## stzurichq4poc | Encryption

Storage account

Search

- Data storage
  - Containers
  - File shares
  - Queues
  - Tables
- Security + networking
  - Networking
  - Front Door and CDN
  - Access keys
  - Shared access signature
  - Encryption
  - Microsoft Defender for

Encryption | Encryption scopes

Storage service encryption protects your data at rest. Azure Storage encrypts your data as it's written in our datacenters, and automatically decrypts it for you as you access it.

Please note that after enabling Storage Service Encryption, only new data will be encrypted, and any existing files in this storage account will retroactively get encrypted by a background encryption process. Learn more about Azure Storage encryption

### Encryption selection

| | |
|---|---|
| Enable support for customer-managed keys | Blobs and files only |
| Infrastructure encryption | Disabled |
| Encryption type | ◉ Microsoft-managed keys<br>○ Customer-managed keys |

# Danke für Ihre Aufmerksamkeit