



AKS Security Simplified

Protecting your Kubernetes Environment

Wolfgang Ofner

Agenda

Authentication and Authorization

Entra Workload ID

Private AKS Cluster

Integrate with a private ACR

Azure Key Vault Provider for Secrets Store CSI Driver

Further security topics



Wolfgang Ofner

Freelance Cloud Architect, Toronto, Canada

Focus on Azure, Kubernetes, and DevOps

<https://programmingwithwolfgang.com>

<https://www.linkedin.com/in/wolfgangofner>

[https://www.youtube.com/
@programmingwithwolfgang](https://www.youtube.com/@programmingwithwolfgang)





Authentication and Authorization

Authentication

Local Accounts with Kubernetes RBAC

Entra ID Authentication in Kubernetes RBAC

Entra ID Authentication with Azure RBAC

Create Kubernetes cluster



Automatic upgrade scheduler

Every week on Sunday (recommended) ▾

Start on: Sat May 10 2025 00:00 +00:00 (Coordinated Universal Time)
[Edit schedule](#)

Node security channel type ⓘ

Security channel scheduler

Node Image ▾

Local accounts with Kubernetes RBAC
Use built-in Kubernetes role-based access control for authorization checks on the cluster.

Microsoft Entra ID authentication with Kubernetes RBAC
Use Microsoft Entra ID for authentication and Kubernetes native RBAC for authorization.


Microsoft Entra ID authentication with Azure RBAC
Use Azure role assignments for authorization checks on the cluster.

Local accounts with Kubernetes RBAC ▾

Choose between local accounts or Microsoft Entra ID for authentication and authorization needs.

Authentication and Authorization ⓘ

ⓘ Once the cluster is deployed, use the Kubernetes CLI to manage RBAC configurations. [Learn more](#) ↗



Local Account with Kubernetes RBAC

Local Accounts with Kubernetes RBAC

Default authentication mode for AKS

No link between Microsoft Entra and AKS

Use K8s build-in authentication

Local Accounts with Kubernetes RBAC

Only recommended when none of the users are in Microsoft Entra

User Management can become very challenging

Local account should be disabled for better security

Token is stored unencrypted in .kube config

```
apiVersion: v1
clusters:
- cluster:
    certificate-authority-data: LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUU2RENDQXRZDZ0F3SUJBZ0lR
    server: https://cyberwisecon-local-accounts-aks-dns-r0x89dn5.hcp.canadacentral.azmk8s.io:443
    name: CyberWiseCon-local-accounts-aks
contexts:
- context:
    cluster: CyberWiseCon-local-accounts-aks
    user: clusterUser_Demo_CyberWiseCon-local-accounts-aks
    name: CyberWiseCon-local-accounts-aks
current-context: CyberWiseCon-local-accounts-aks
kind: Config
preferences: {}
users:
- name: clusterUser_Demo_CyberWiseCon-local-accounts-aks
  user:
    client-certificate-data: LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUZIakNDQXdhZ0F3SUJBZ0lRSQU1
    client-key-data: LS0tLS1CRUdJTiBSU0EgUFJJVkFURSBLRVktLS0tLQpNSU1KS2dJQkFBS0NBZ0VBNmd6a0xSQVlR
    token: cv86e8y88i83s1eft9kvkixpycr4n40ro6rv8nxu9nln2o5ke5tn2y1z0vq25hzwe7adydp7kvo7n6sa5zcg9u
```



Entra ID Authentication with Kubernetes RBAC



Entra ID Authentication with Kubernetes RBAC

Authentication via Microsoft Entra

Authorization via Kubernetes RBAC

Entra ID Authentication with Kubernetes RBAC

Admin creates a role binding between K8s role and Entra user or group

Entra user or group needs “Azure Kubernetes Service Cluster User” role to download .kube config

Entra ID Authentication with Kubernetes RBAC

Easier user management than local accounts

Choose this option to have a “portable” cluster

- Cluster contains all roles and role binding definitions

Entra ID Authentication with Kubernetes RBAC

Auditing access to the cluster can be cumbersome

- Access can be given to Entra users and groups
- Groups and users are managed with their Entra IDs

kind: Role

apiVersion: rbac.authorization.k8s.io/v1

metadata:

name: reader

namespace: read

rules:

- apiGroups: [""]

- resources: ["pods", "services", "endpoints", "persistentvolumeclaims",

- verbs: ["get", "list", "watch"]

- apiGroups: ["apps"]

- resources: ["deployments", "daemonsets", "replicasets", "statefulsets"]

- verbs: ["get", "list", "watch"]

- apiGroups: ["batch"]

- resources: ["jobs", "cronjobs"]

- verbs: ["get", "list", "watch"]

- apiGroups: ["extensions"]

- resources: ["ingresses"]


```
kind: RoleBinding
apiVersion: rbac.authorization.k8s.io/v1
metadata:
  name: reader-user-binding
  namespace: read
subjects:
  - kind: Group
    name: 24975d09-19e9-47a5-aa3b-e952c693c016 # Entra ID
    namespace: read
roleRef:
  kind: Role # or ClusterRole
  name: reader
  apiGroup: rbac.authorization.k8s.io
```

Entra ID Authentication with Kubernetes RBAC

```
PS C:\Demo> kubectl get all -n read
```

NAME	READY	STATUS	RESTARTS	AGE
pod/nginx	1/1	Running	0	4m32s

```
Error from server (Forbidden): replicationcontrollers is forbidden:  
: User "demo.user@programmingwithwolfgang.com" cannot list resourc  
e "replicationcontrollers" in API group "" in the namespace "read"
```

```
Error from server (Forbidden): horizontalpodautoscalers.autoscalin  
g is forbidden: User "demo.user@programmingwithwolfgang.com" canno  
t list resource "horizontalpodautoscalers" in API group "autoscali  
ng" in the namespace "read"
```



Entra ID Authentication with Azure RBAC

Entra Authentication with Azure RBAC

Manage access to the cluster with Azure only

Use Azure RBAC roles to manage permissions inside the cluster

Recommended way to manage AKS cluster

Entra Authentication with Azure RBAC

User or group needs the “Azure Kubernetes Service Cluster User” role to download the .kube config

Assign built-in or custom roles

Namespace specific permissions can only be assigned using the Azure CLI



Entra Workload Identity

Entra Workload Identity

Always use identities over username/password

Azure resources can have managed identities

AKS can have an identity

- Identity is assigned not the pod

Entra Workload Identity

Entra Workload Identity gives a pod an identity

- Pods can access Azure resources with this identity
- `azure.workload.identity/use` label needed on pod
- OIDC Issuer must be enabled for the cluster



Kubelet



AKS workload



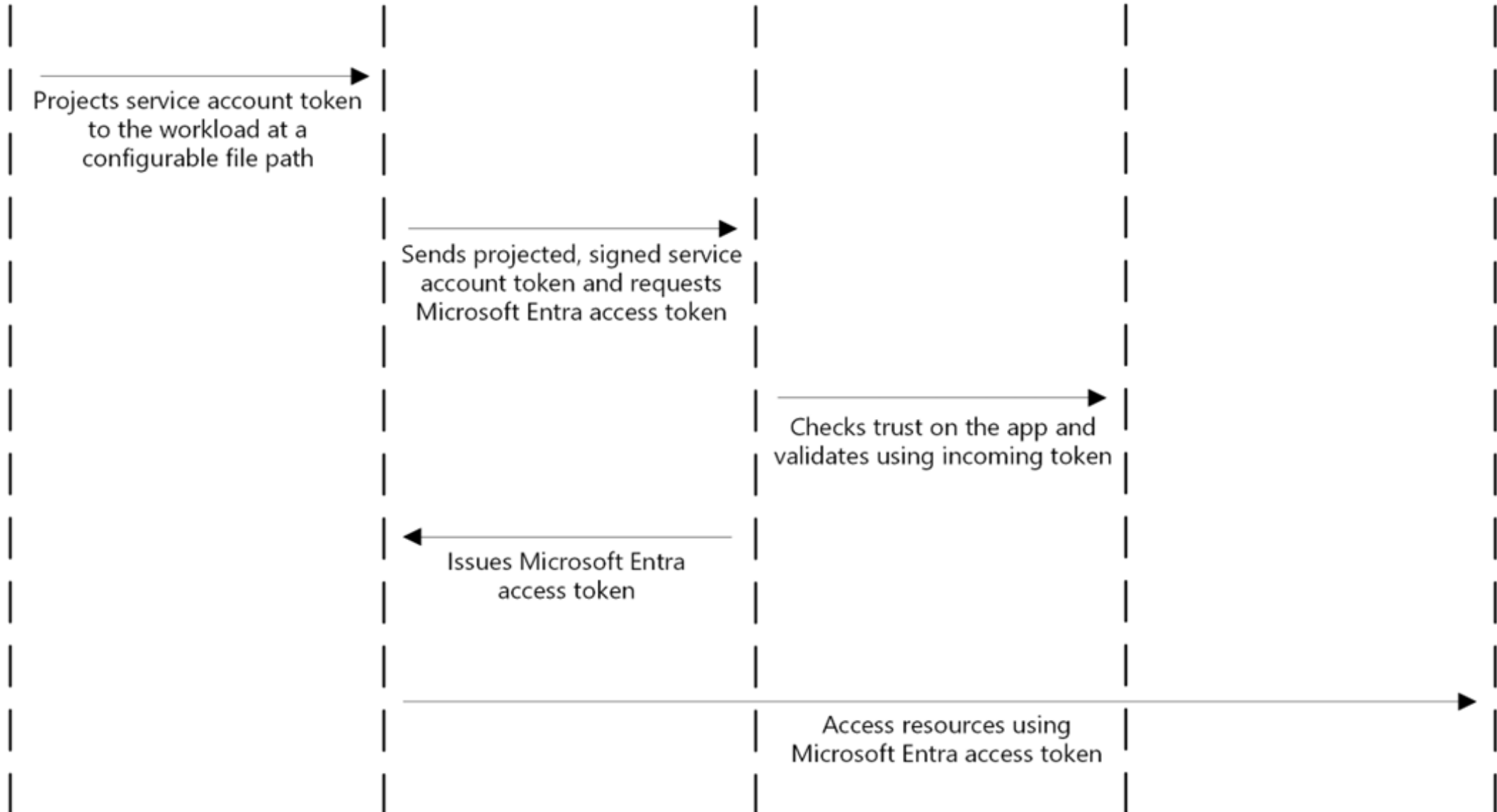
Microsoft Entra ID



OpenID Discovery Document

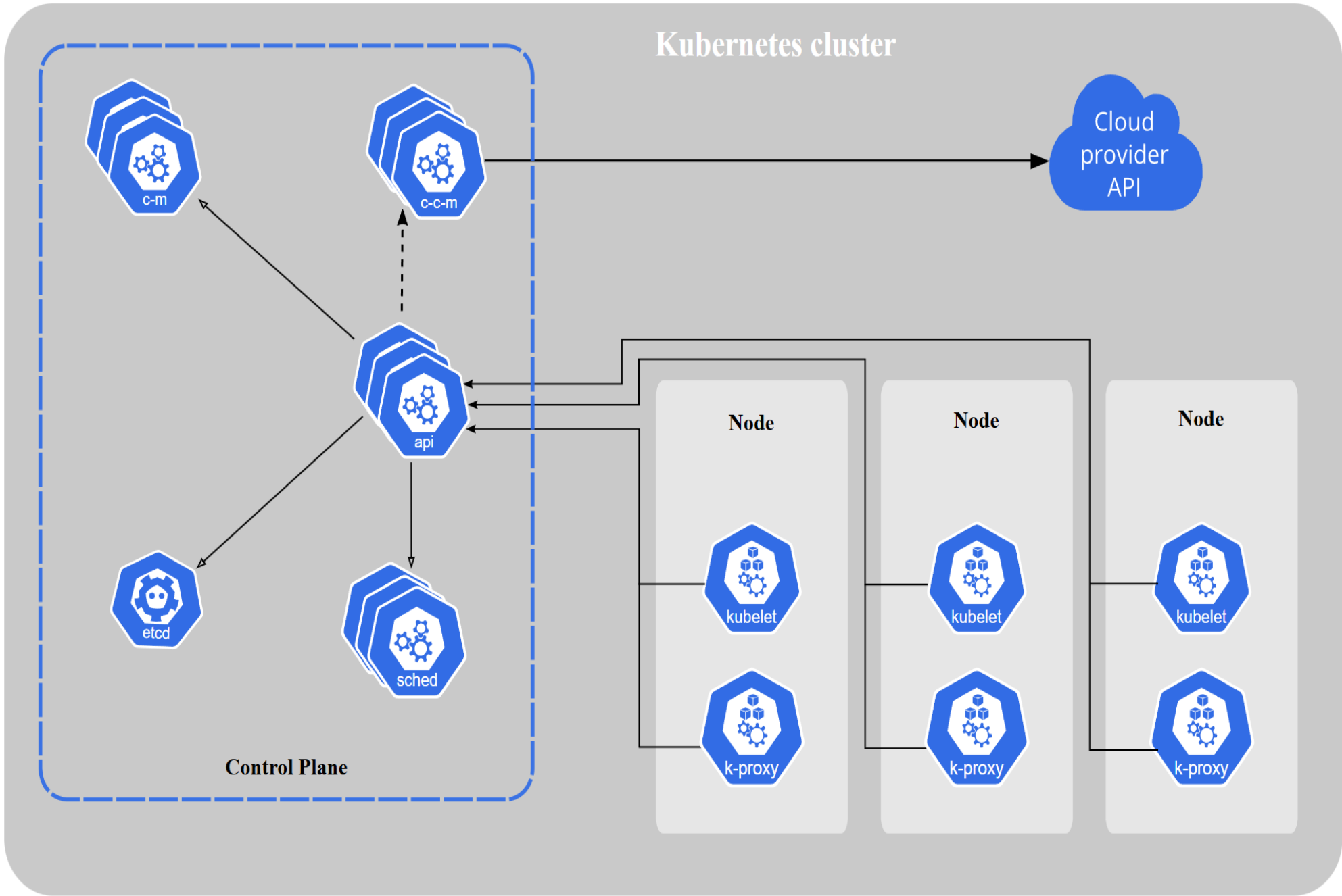











Azure resources





Private AKS Cluster



- API server 
- Cloud controller manager (optional) 
- Controller manager 
- etcd (persistence store) 
- kubelet 
- kube-proxy 
- Scheduler 
- Control plane 
- Node 

Private AKS Cluster

Disable public access to API

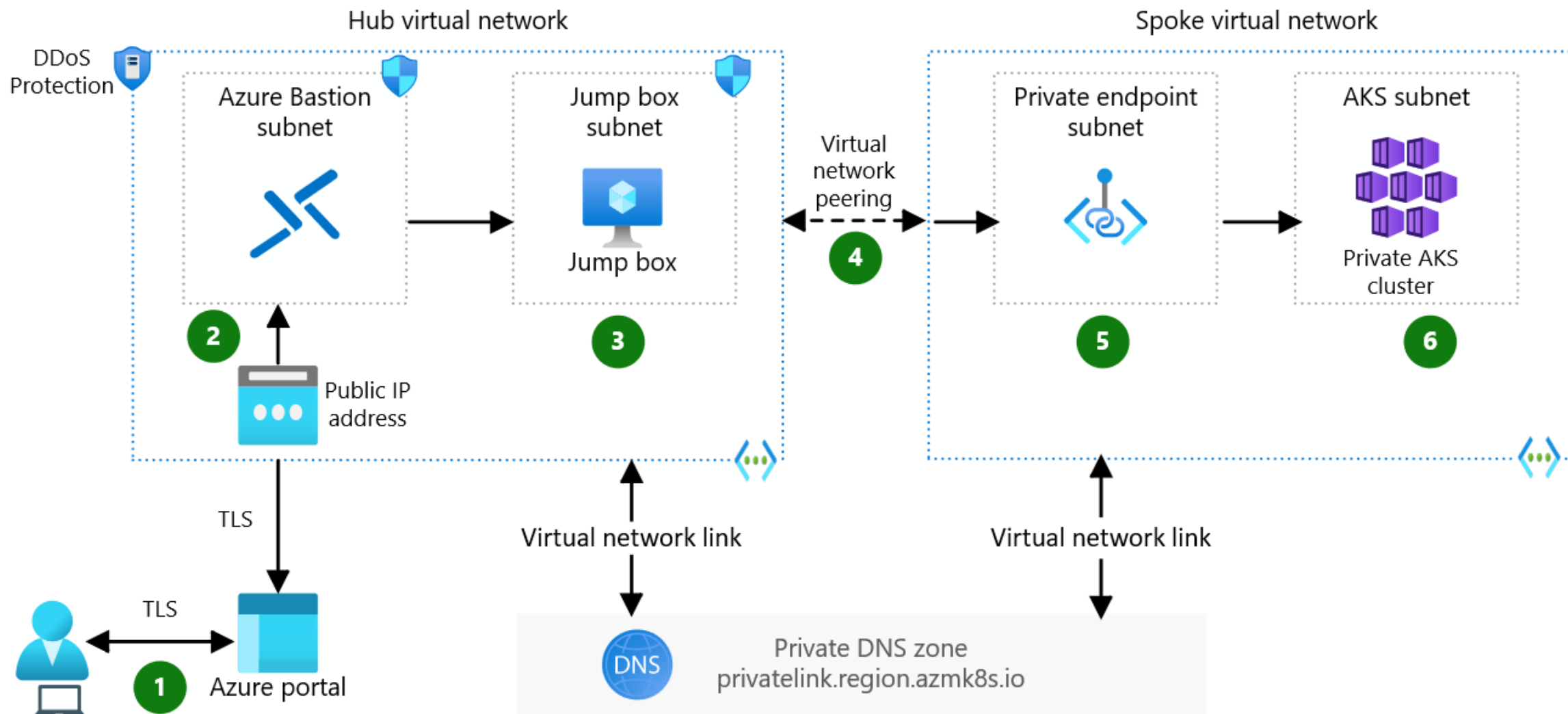
Communication between Worker and Master nodes
over private connection

Private AKS Cluster

Use Azure Bastion and a Jump box

Wrap command with az aks command invoke

VPN/ExpressRoute connection



Private AKS Cluster

az aks command invoke \

--resource-group MyResourceGroup \

--name MyAks \

--command "kubectl get pod nginx -n private"



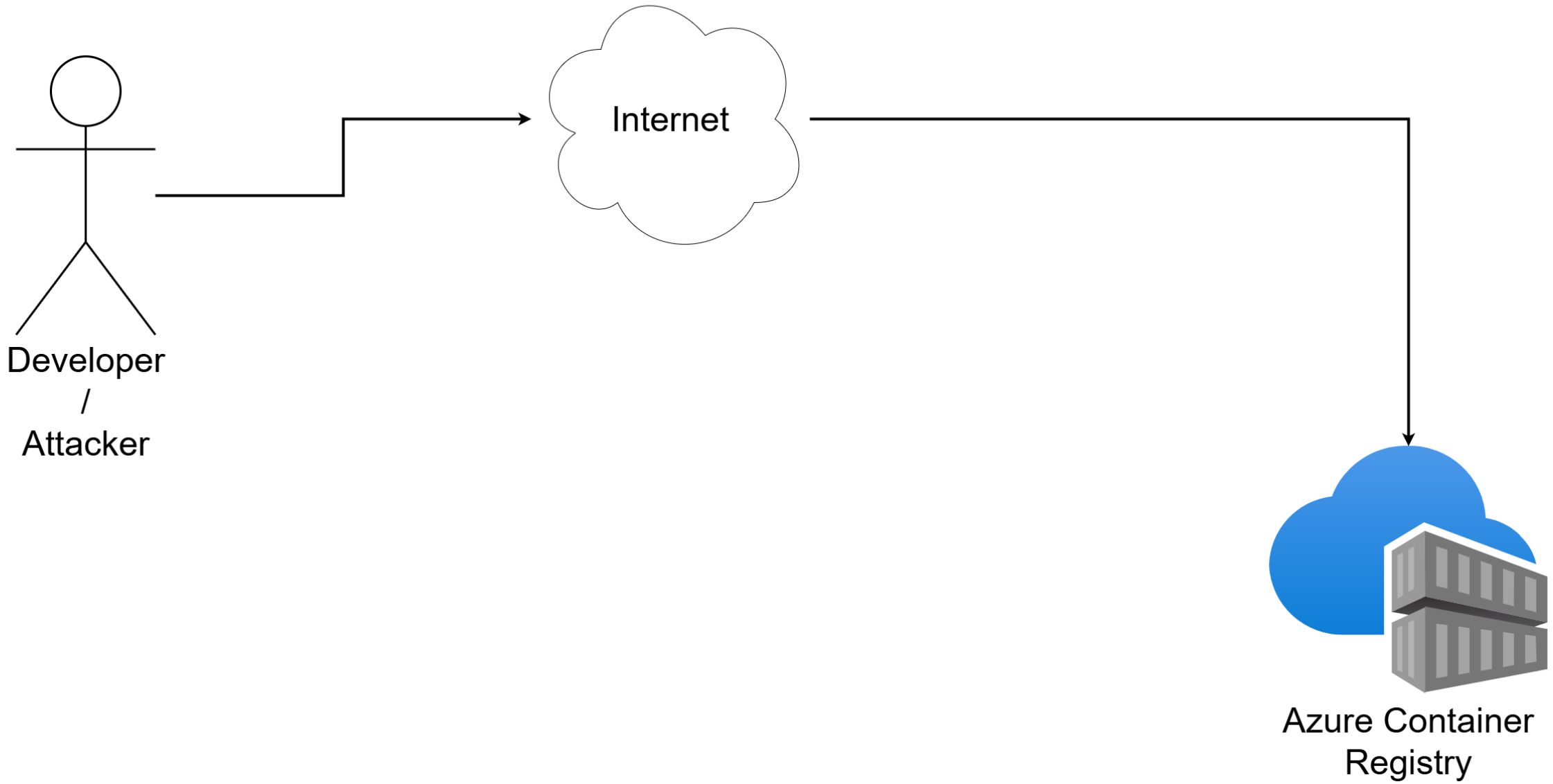
Private Azure Container Registry

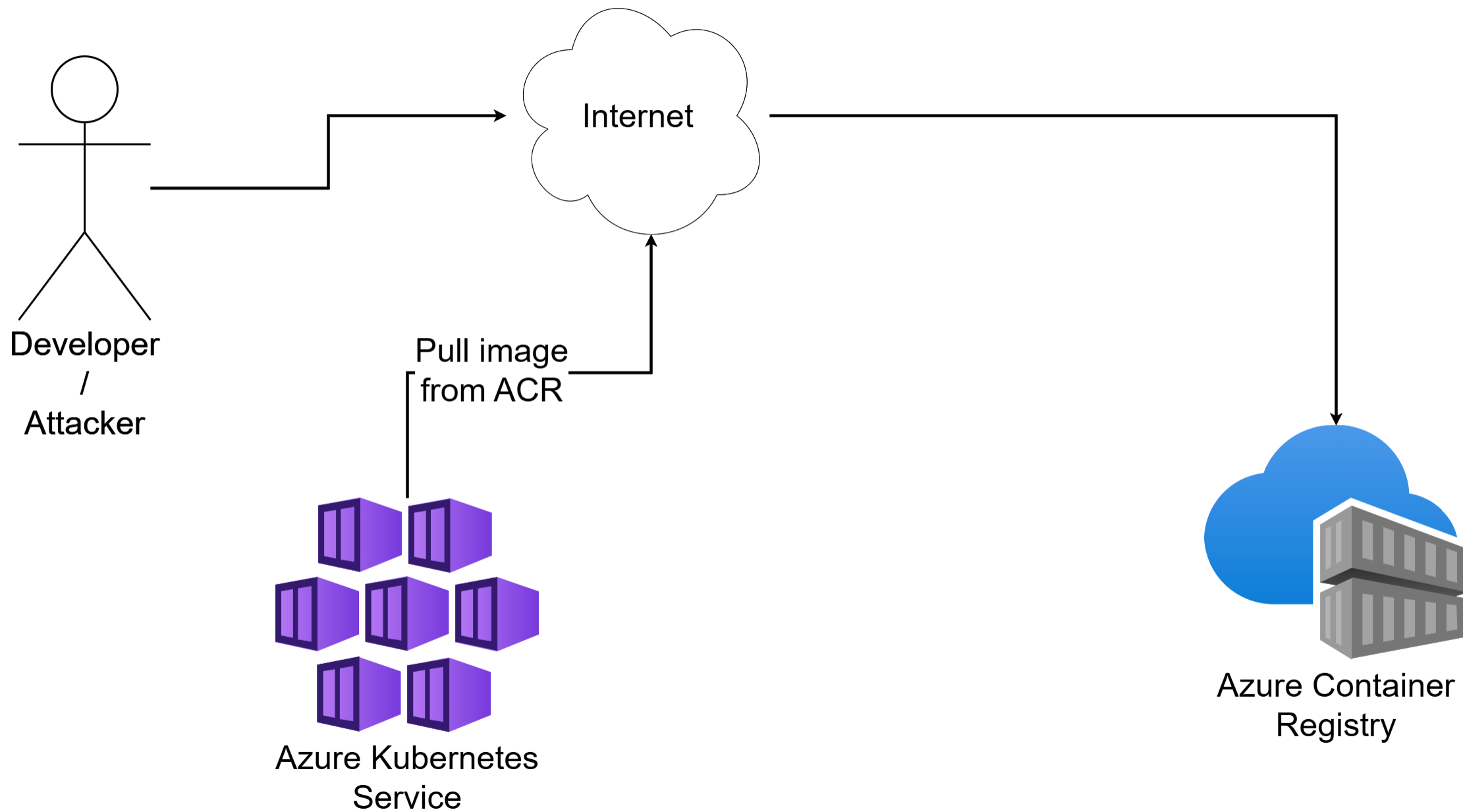
Azure Container Registry (ACR)

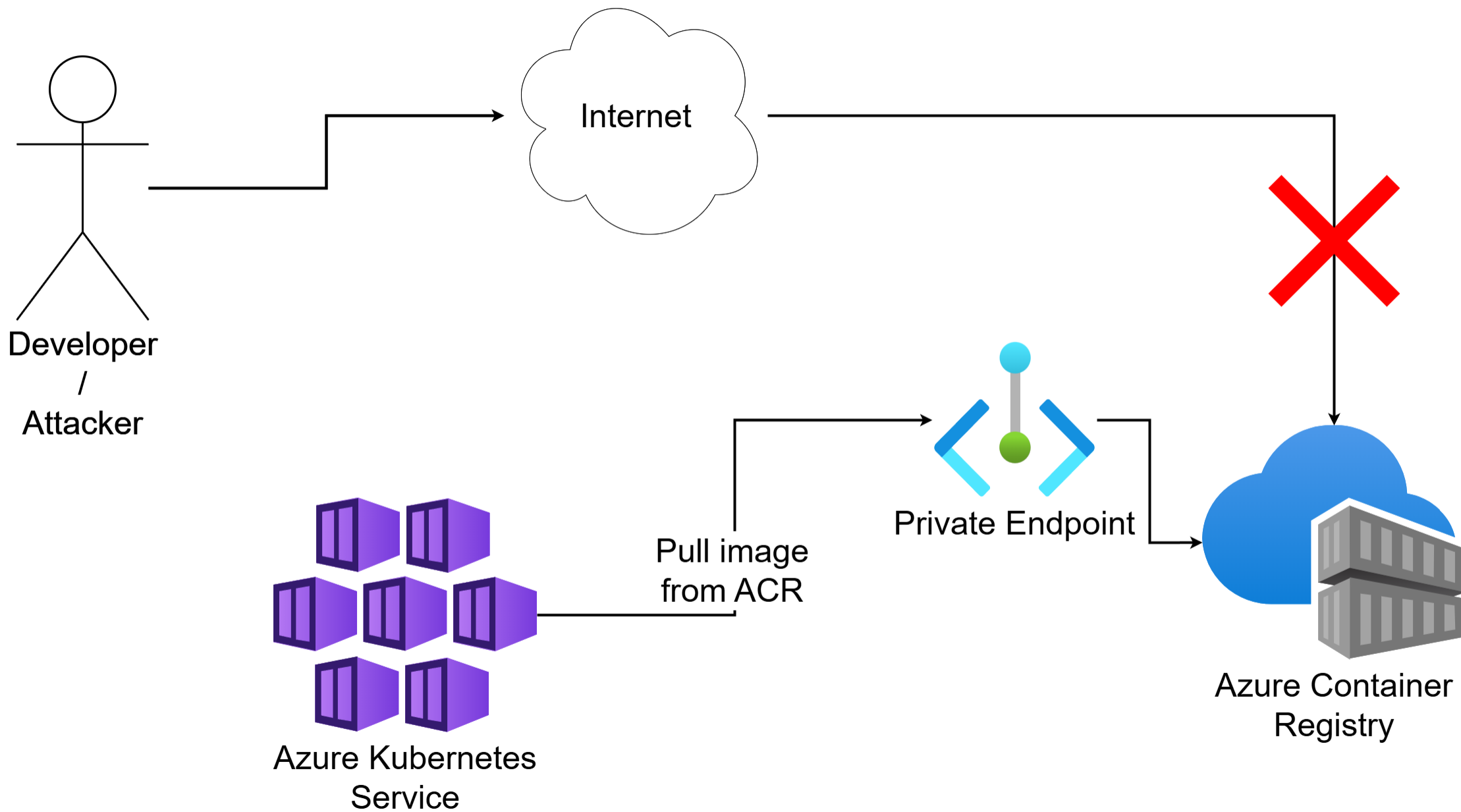
Keep everything private

Store your images in a private registry

AKS should download images over a private connection







Azure Container Registry (ACR)

No changes for pull operation necessary

Private DNS-Zone resolves public FQDN

Build agent needs to push images over PE



Azure Key Vault provider for Secrets Store CSI Driver



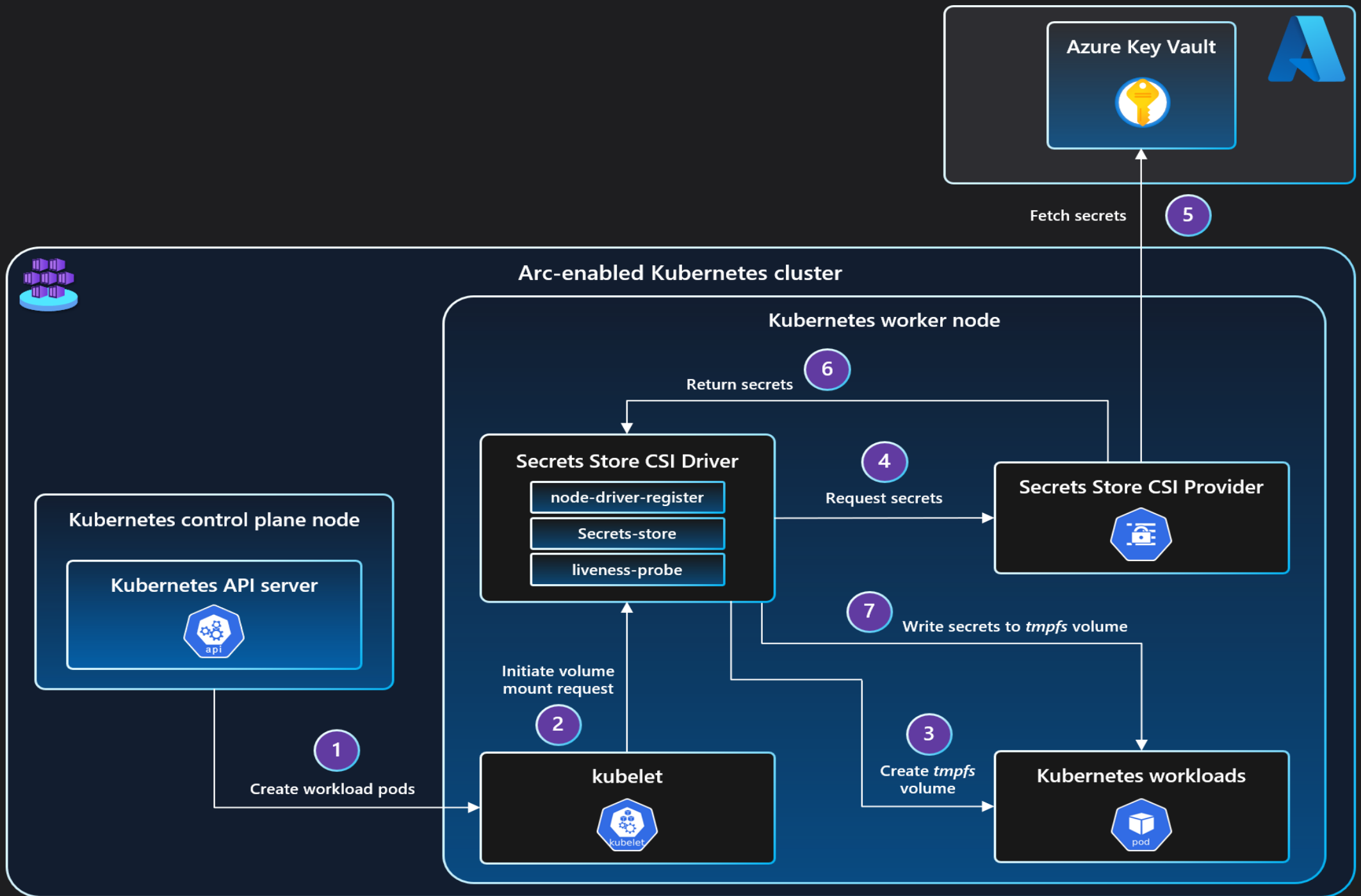
Azure Key Vault provider for Secrets Store

Mount secrets, keys, and certificates to pods

Auto-rotate secrets

Sync Azure Key Vault with Kubernetes secrets

Separation of concerns





Further Security Topics

Further Security Topics

Use Azure Linux as your node OS

Disable SSH access

Disable local account

Install the Azure Policy addon

Microsoft Defender for Containers

Setup your cluster using AKS Automatic

Further Security Topics

Only run signed images

Validate image integrity

Limit the pod privileges

Set the security context

Reduce the pod capabilities

Configure seccomp (Secure computing)



AKS Security Simplified

Protecting your Kubernetes Environment

Wolfgang Ofner