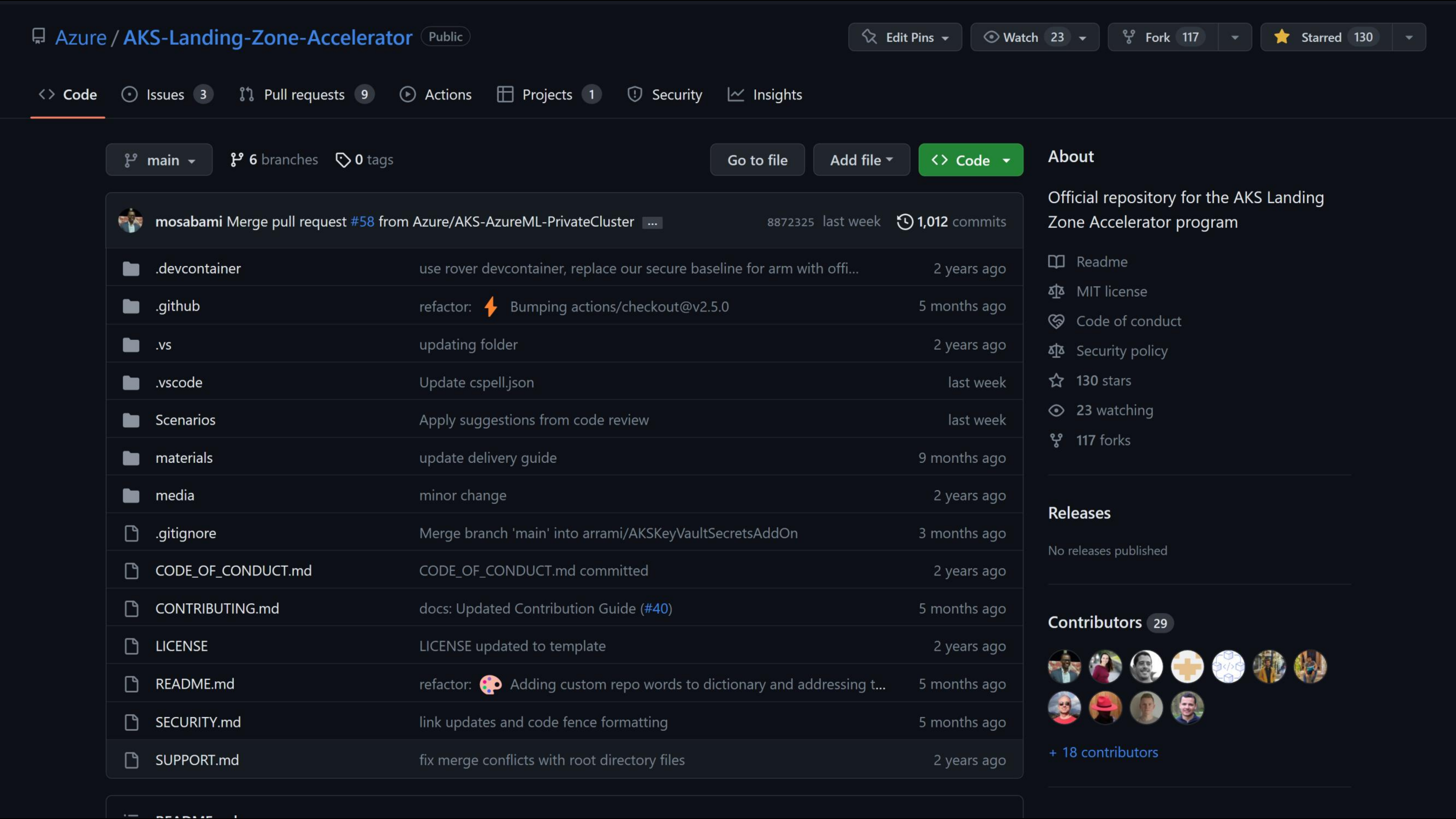
AKS Landing Zone using Terraform

Houssem Dellai



mosabami Merge pull request #58 from Azure/AKS-AzureML-PrivateCluster … 8872325 last week 1,012 commits

.devcontainer	use rover devcontainer, replace our secure baseline for arm with offi…	2 years ago
.github	refactor: ⚡ Bumping actions/checkout@v2.5.0	5 months ago
.vs	updating folder	2 years ago
.vscode	Update cspell.json	last week
Scenarios	Apply suggestions from code review	last week
materials	update delivery guide	9 months ago
media	minor change	2 years ago
.gitignore	Merge branch 'main' into arrami/AKSKeyVaultSecretsAddOn	3 months ago
CODE_OF_CONDUCT.md	CODE_OF_CONDUCT.md committed	2 years ago
CONTRIBUTING.md	docs: Updated Contribution Guide (#40)	5 months ago
LICENSE	LICENSE updated to template	2 years ago
README.md	refactor: 🎨 Adding custom repo words to dictionary and addressing t…	5 months ago
SECURITY.md	link updates and code fence formatting	5 months ago
SUPPORT.md	fix merge conflicts with root directory files	2 years ago

About

Official repository for the AKS Landing Zone Accelerator program

- Readme
- MIT license
- Code of conduct
- Security policy
- 130 stars
- 23 watching
- 117 forks

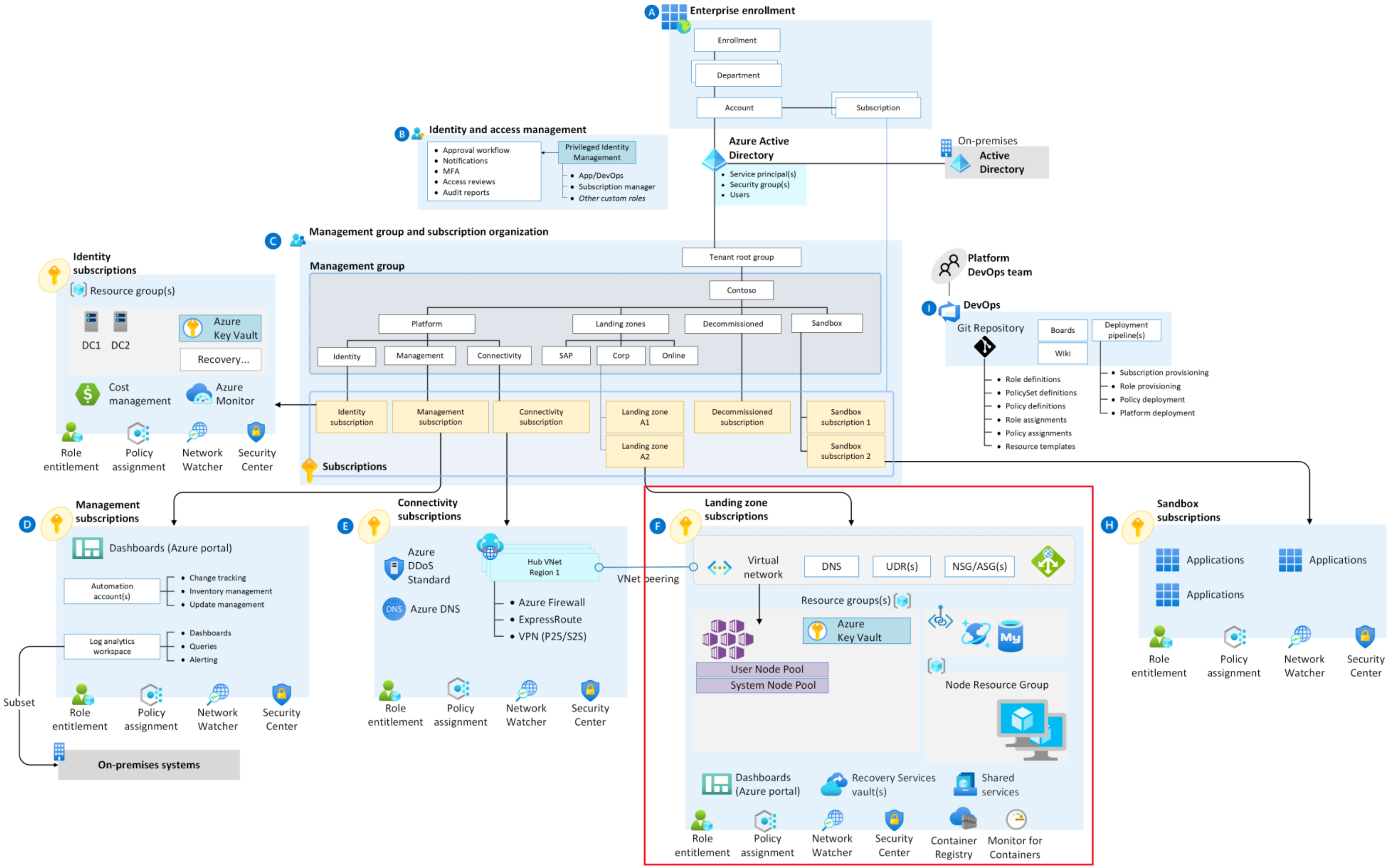
Releases

No releases published

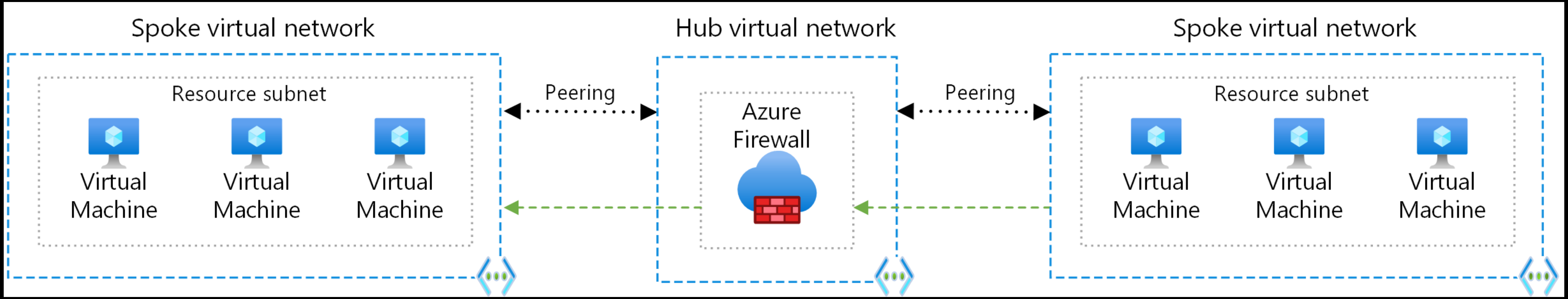
Contributors 29

+ 18 contributors

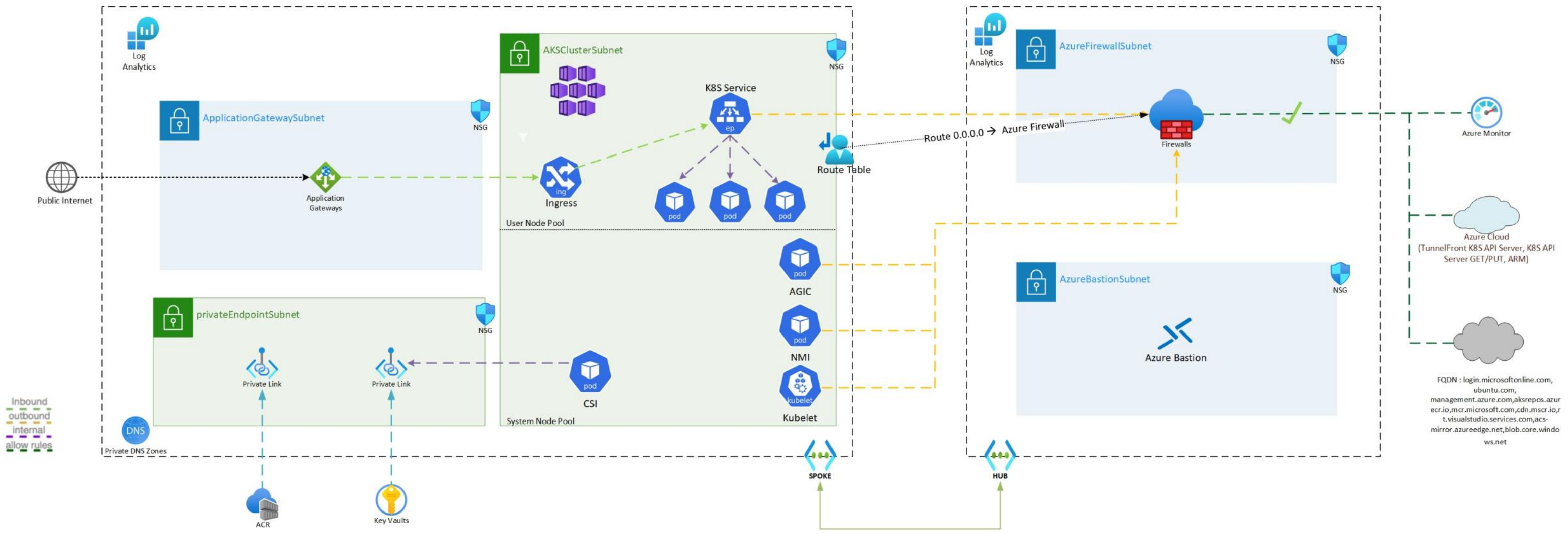
Platform Enterprise Scale & Application Landing Zone



AKS Landing Zone uses Hub & Spoke architecture

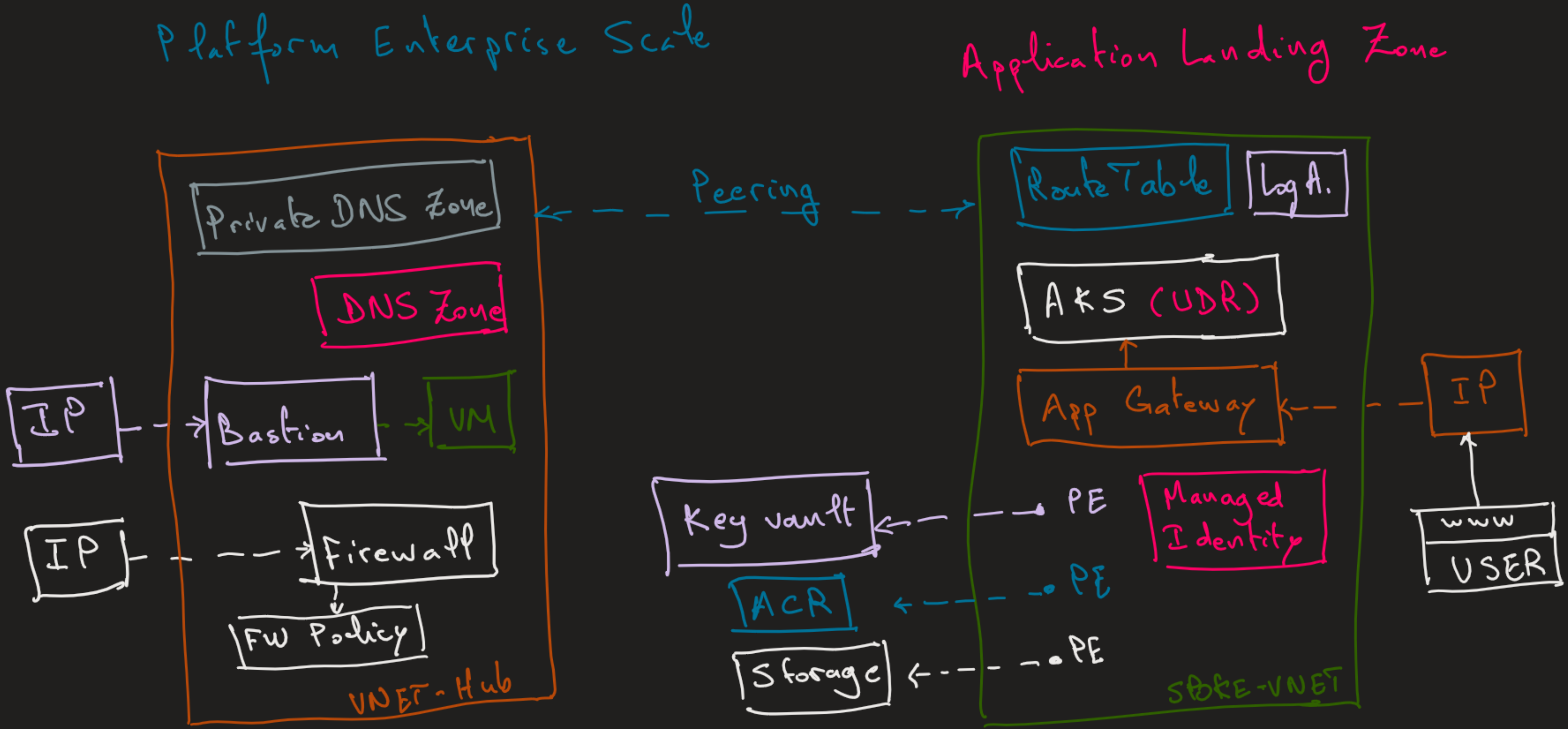


AKS Landing Zone simplified

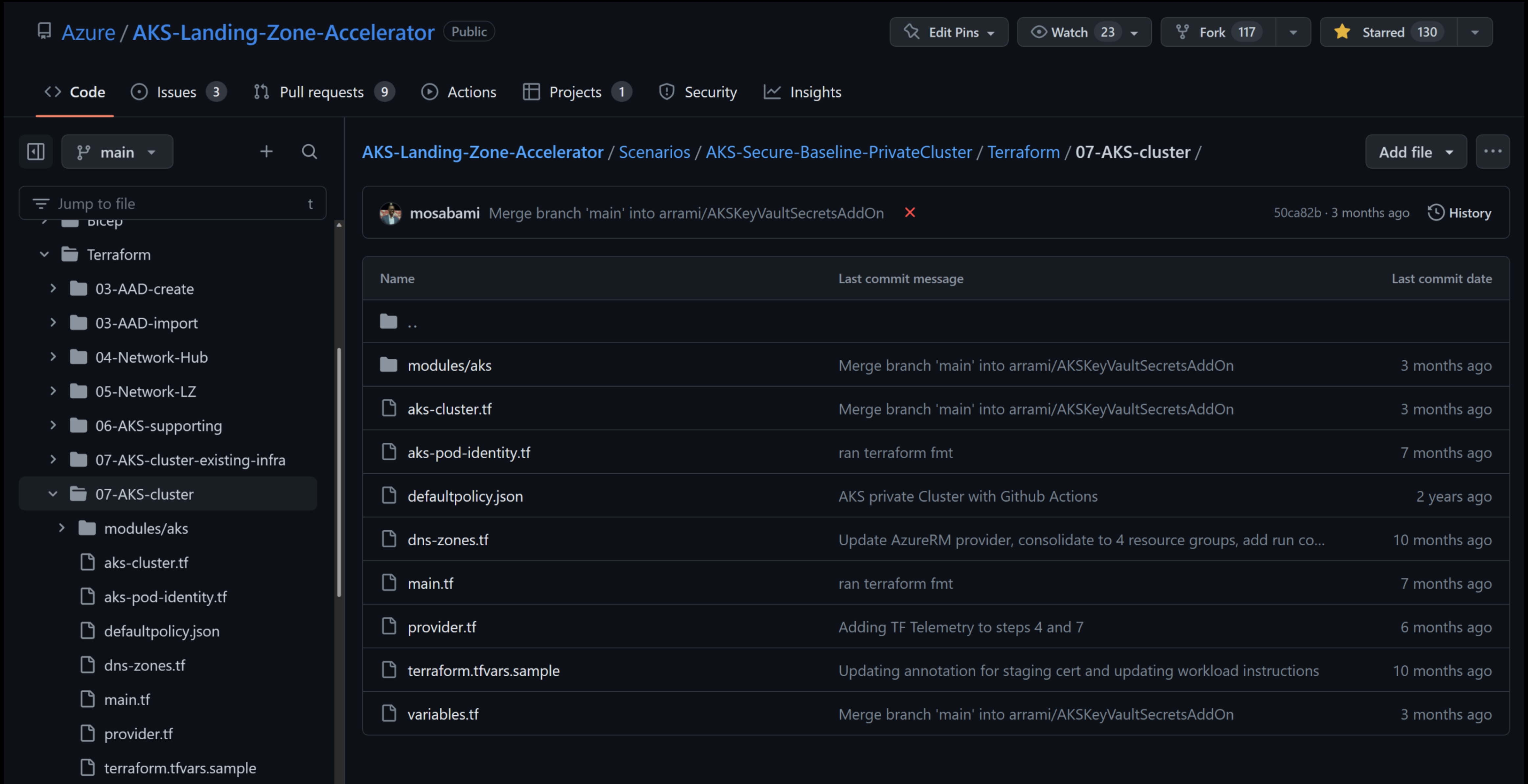


FQDN : login.microsoftonline.com, ubuntu.com, management.azure.com,aksrepos.azurecr.io,mcr.microsoft.com,cdn.mscr.io,rt.visualstudio.services.com,acs-mirror.azureedge.net,blob.core.windows.net

AKS Landing Zone



What are these multiple levels in Terraform ?



Azure / AKS-Landing-Zone-Accelerator Public

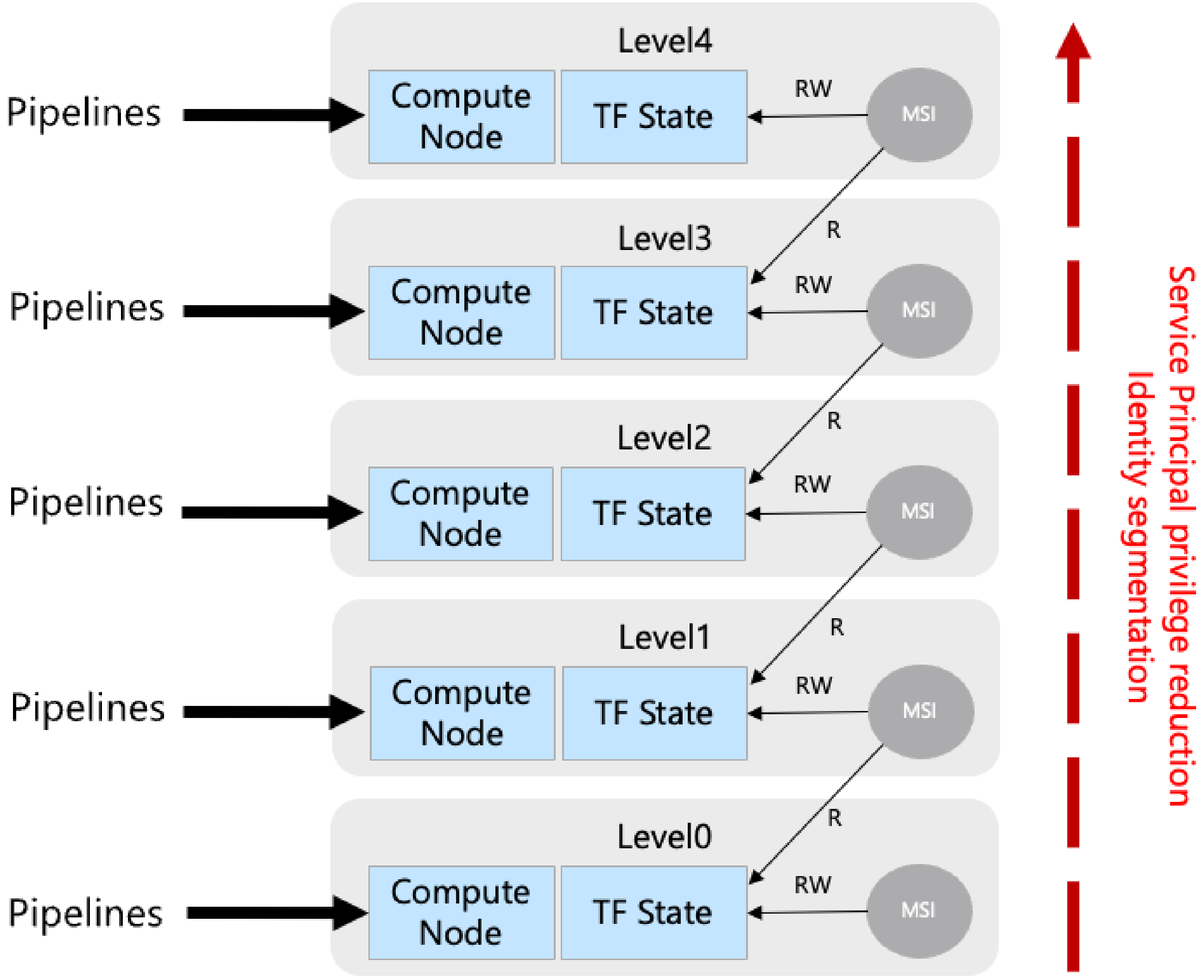
- Bicep
- Terraform
  - 03-AAD-create
  - 03-AAD-import
  - 04-Network-Hub
  - 05-Network-LZ
  - 06-AKS-supporting
  - 07-AKS-cluster-existing-infra
  - 07-AKS-cluster
    - modules/aks
    - aks-cluster.tf
    - aks-pod-identity.tf
    - defaultpolicy.json
    - dns-zones.tf
    - main.tf
    - provider.tf
    - terraform.tfvars.sample

AKS-Landing-Zone-Accelerator / Scenarios / AKS-Secure-Baseline-PrivateCluster / Terraform / 07-AKS-cluster /

mosabami Merge branch 'main' into arrami/AKSKeyVaultSecretsAddOn ✕ 50ca82b · 3 months ago History

Name	Last commit message	Last commit date
..		
modules/aks	Merge branch 'main' into arrami/AKSKeyVaultSecretsAddOn	3 months ago
aks-cluster.tf	Merge branch 'main' into arrami/AKSKeyVaultSecretsAddOn	3 months ago
aks-pod-identity.tf	ran terraform fmt	7 months ago
defaultpolicy.json	AKS private Cluster with Github Actions	2 years ago
dns-zones.tf	Update AzureRM provider, consolidate to 4 resource groups, add run co...	10 months ago
main.tf	ran terraform fmt	7 months ago
provider.tf	Adding TF Telemetry to steps 4 and 7	6 months ago
terraform.tfvars.sample	Updating annotation for staging cert and updating workload instructions	10 months ago
variables.tf	Merge branch 'main' into arrami/AKSKeyVaultSecretsAddOn	3 months ago

Multi-level Terraform configuration



Application landing zone (delegated to application teams) **platform ops solution accelerators** (AKS, App Service, Data analytics, etc.)

Application landing zone (managed by platform team)**,** **subscription vending** machine to create **application** landing zone subscriptions and base services (resource groups, Virtual Networks peering and delegated identities for level 4)

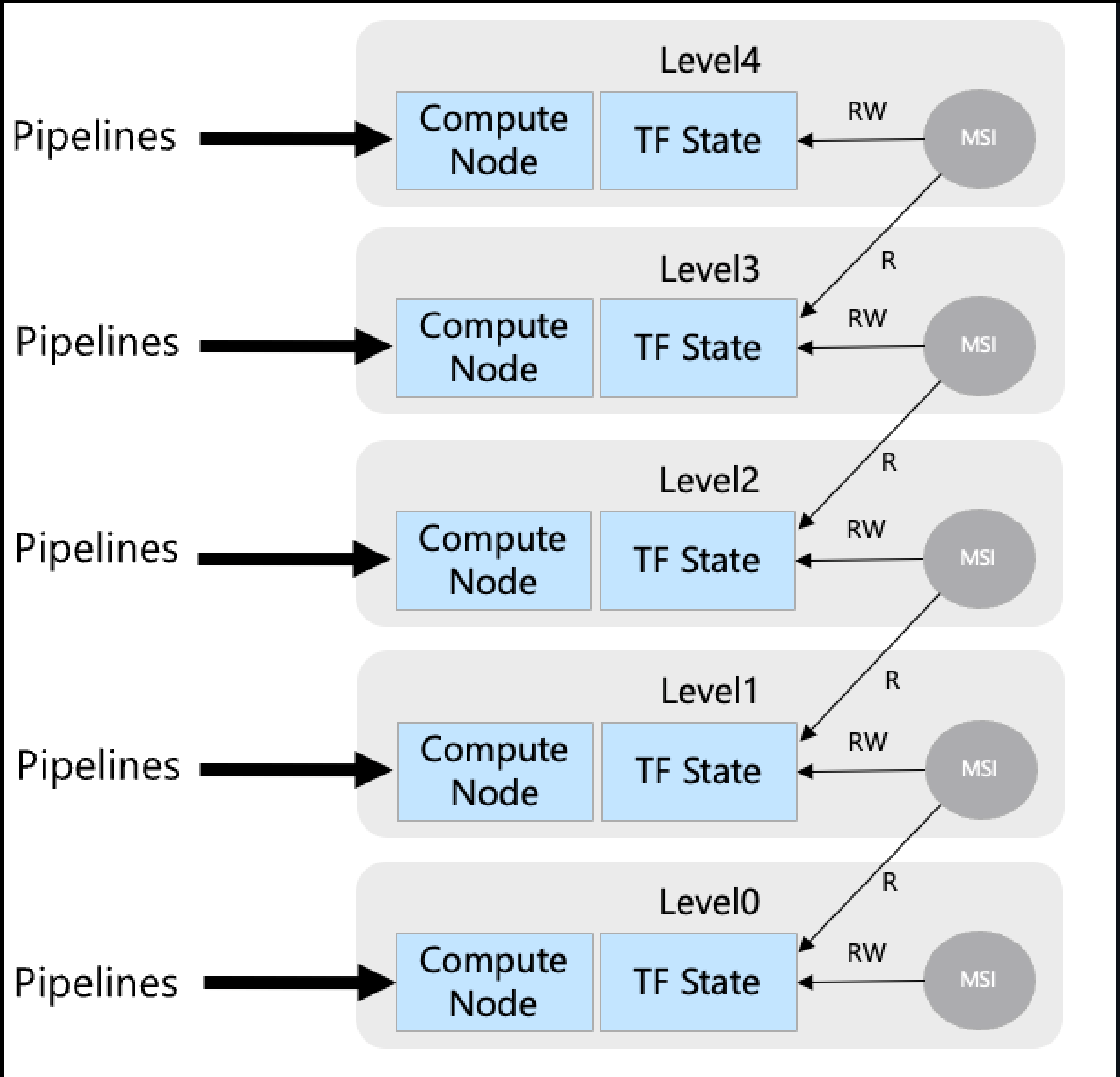
Core **platform**: **Connectivity** components for Virtual WAN, hub and spoke, ExpressRoute, etc., **identity** domain controllers, **management** services

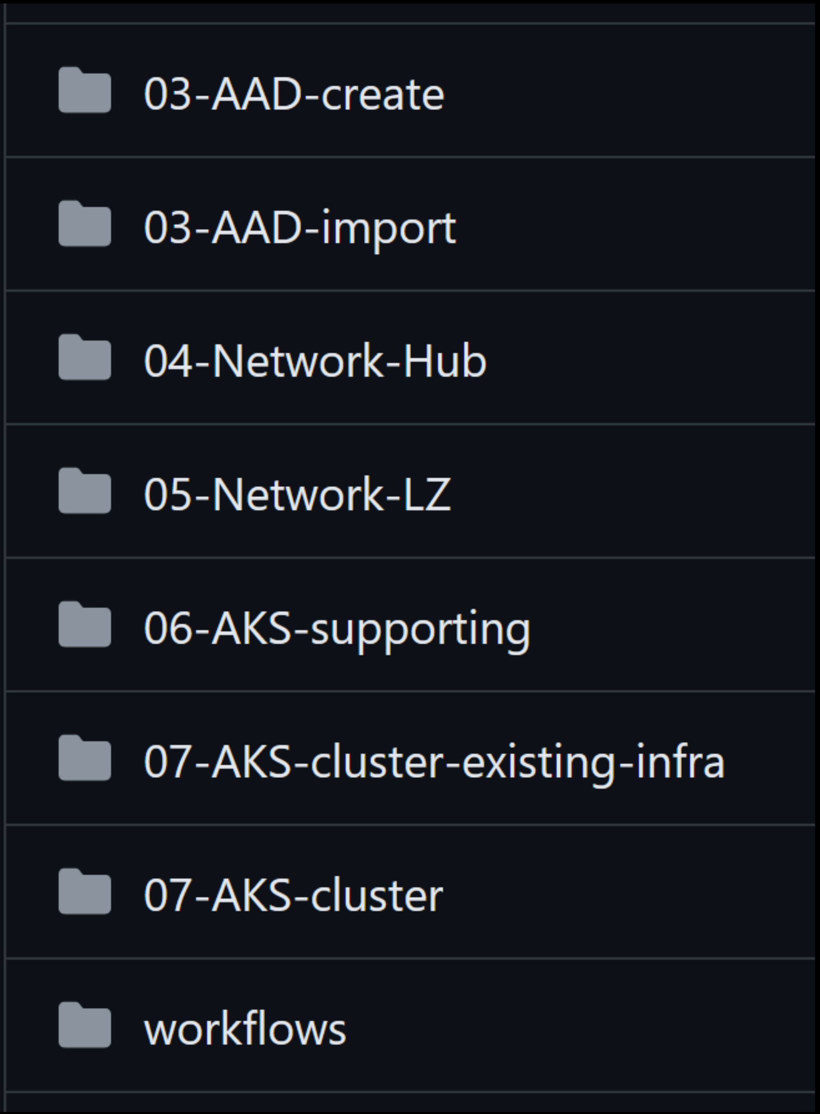
Core **platform**: **Enterprise-Scale** management groups and policies, **Identity** services, **management** services, platform **subscriptions** creation and GitOps pipelines.

Core **platform**: Terraform State Management Fundamentals (launchpad), Billing subscription role delegation from EA or MCA.

https://aztfmod.github.io/documentation/docs/fundamentals/lz-intro/

Applying multi-level Terraform to AKS Landing Zone




03-AAD-create

03-AAD-import

04-Network-Hub

05-Network-LZ

06-AKS-supporting

07-AKS-cluster-existing-infra

07-AKS-cluster

workflows

Steps to deploy AKS Landing Zone

1. Create Storage Account for TF state
   TF state per level

2. Create Azure AD groups
   AKS dev & admin groups

3. Create Network Hub
   VNET, Firewall, Bastion, VM

4. Create Network Spoke / LZ
   VNET, Peering, RT, AppGw, Pr. DNS Zones

5. Create resources for AKS
   ACR, Key vault, PE, Public DNS Zone

6. Create AKS cluster
   AKS, LA, MI, RBAC..

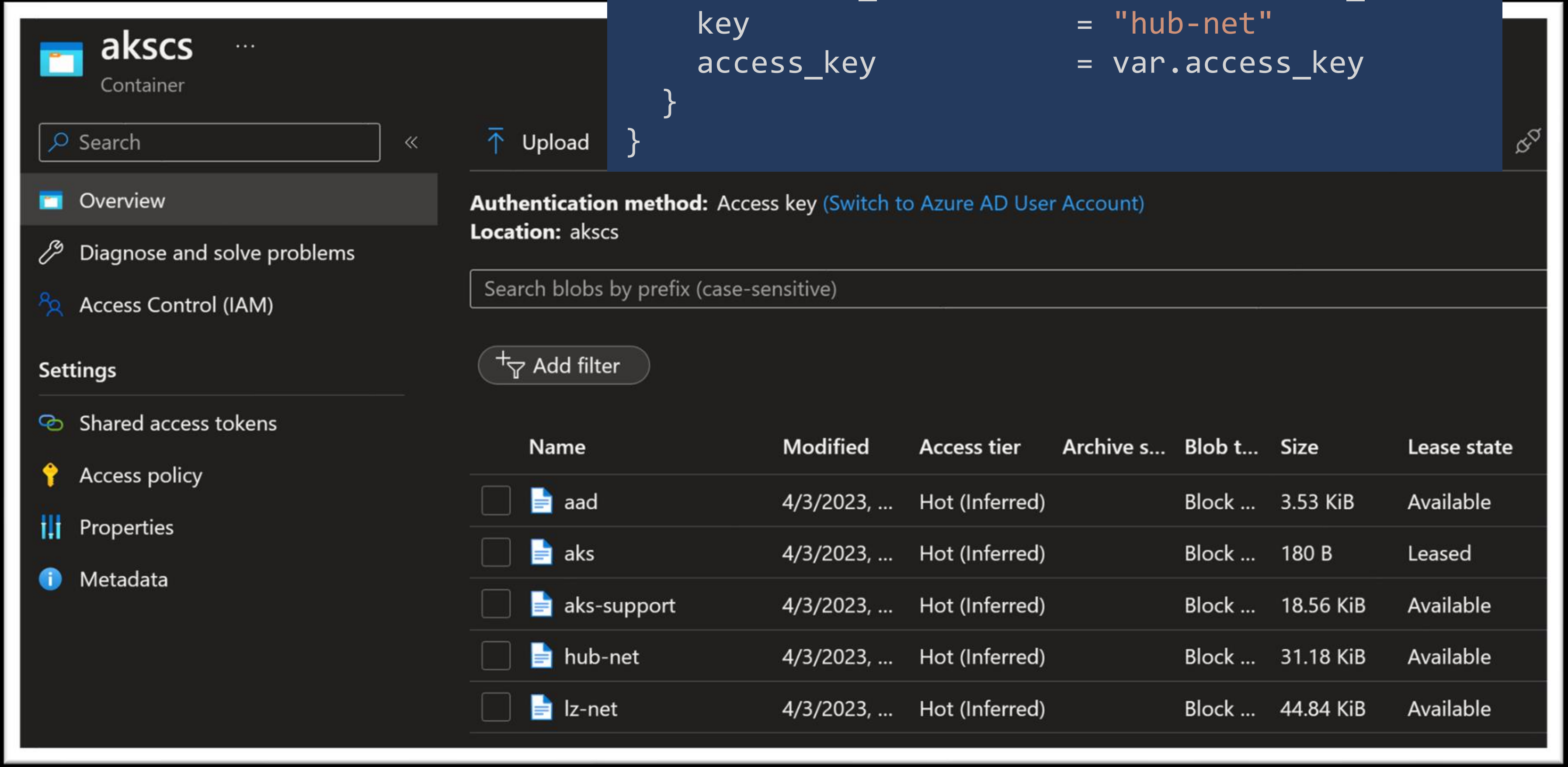
Steps to deploy AKS Landing Zone

1. Create Storage Acc for TF state

   TF state per level

```
data "terraform_remote_state" "existing-hub" {
  backend = "azurerm"

  config = {
    storage_account_name = var.state_sa_name
    container_name       = var.container_name
    key                  = "hub-net"
    access_key           = var.access_key
  }
}
```



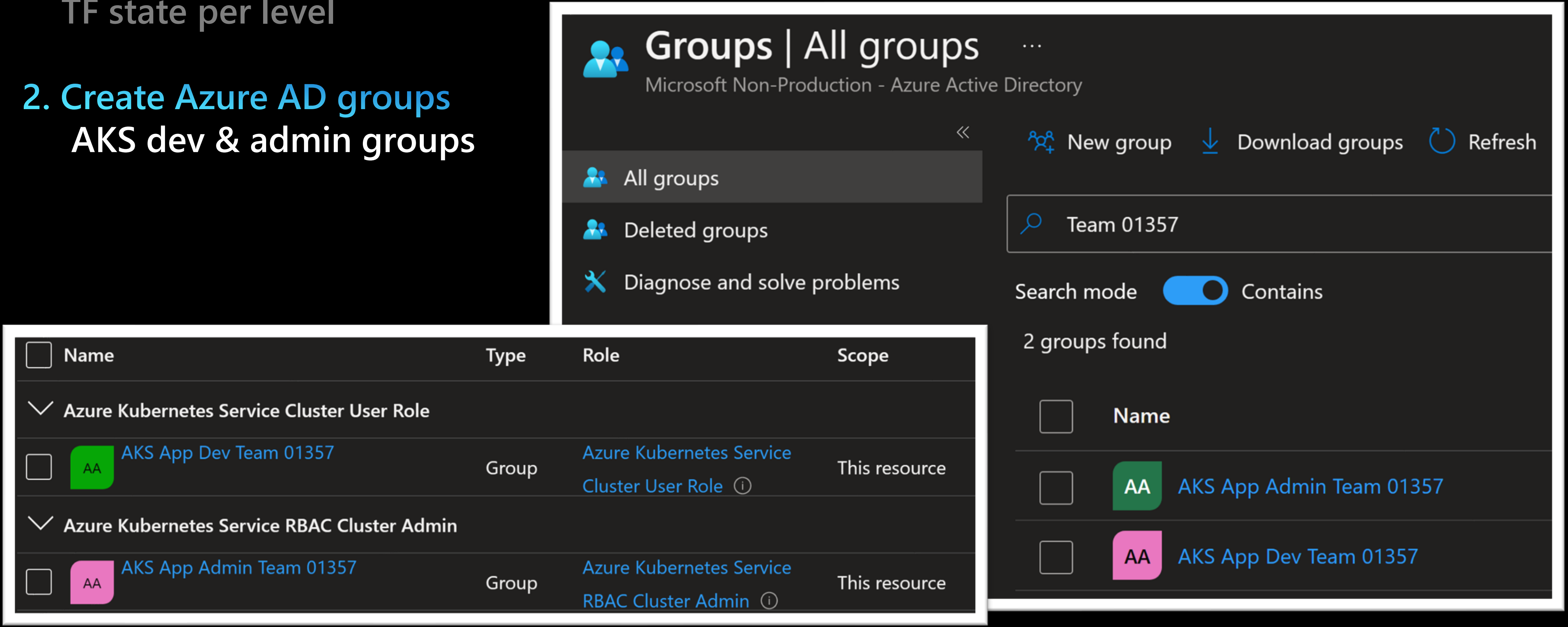
Authentication method: Access key (Switch to Azure AD User Account)
Location: akscs

	Name	Modified	Access tier	Archive s...	Blob t...	Size	Lease state
□	aad	4/3/2023, ...	Hot (Inferred)		Block ...	3.53 KiB	Available
□	aks	4/3/2023, ...	Hot (Inferred)		Block ...	180 B	Leased
□	aks-support	4/3/2023, ...	Hot (Inferred)		Block ...	18.56 KiB	Available
□	hub-net	4/3/2023, ...	Hot (Inferred)		Block ...	31.18 KiB	Available
□	lz-net	4/3/2023, ...	Hot (Inferred)		Block ...	44.84 KiB	Available

Steps to deploy AKS Landing Zone

1. Create Storage Acc for TF state
   TF state per level

2. Create Azure AD groups
   AKS dev & admin groups



Groups | All groups ···

Microsoft Non-Production - Azure Active Directory

Team 01357

Search mode: Contains

2 groups found

□	Name
□	AA AKS App Admin Team 01357
□	AA AKS App Dev Team 01357

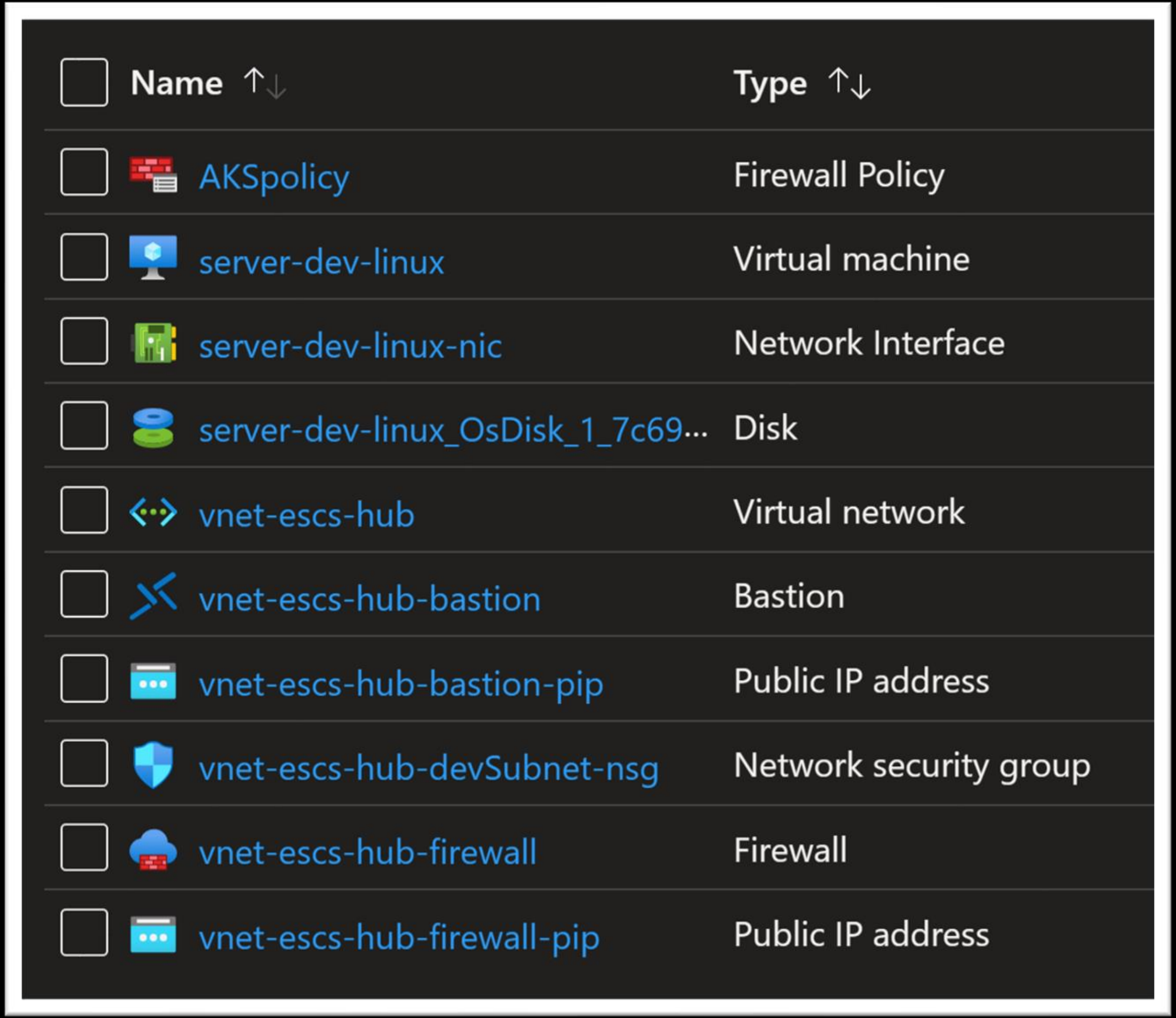
□	Name	Type	Role	Scope
	Azure Kubernetes Service Cluster User Role			
□	AA AKS App Dev Team 01357	Group	Azure Kubernetes Service Cluster User Role ⓘ	This resource
	Azure Kubernetes Service RBAC Cluster Admin			
□	AA AKS App Admin Team 01357	Group	Azure Kubernetes Service RBAC Cluster Admin ⓘ	This resource

Steps to deploy AKS Landing Zone

1. Create Storage Acc for TF state
   TF state per level

2. Create Azure AD groups
   AKS dev & admin groups

3. Create Network Hub
   VNET, Firewall, Bastion, VM



□	Name ↑↓	Type ↑↓
□	AKSpolicy	Firewall Policy
□	server-dev-linux	Virtual machine
□	server-dev-linux-nic	Network Interface
□	server-dev-linux_OsDisk_1_7c69...	Disk
□	vnet-escs-hub	Virtual network
□	vnet-escs-hub-bastion	Bastion
□	vnet-escs-hub-bastion-pip	Public IP address
□	vnet-escs-hub-devSubnet-nsg	Network security group
□	vnet-escs-hub-firewall	Firewall
□	vnet-escs-hub-firewall-pip	Public IP address

Steps to deploy AKS LZ

1. Create Storage Acc for TF state
   TF state per level

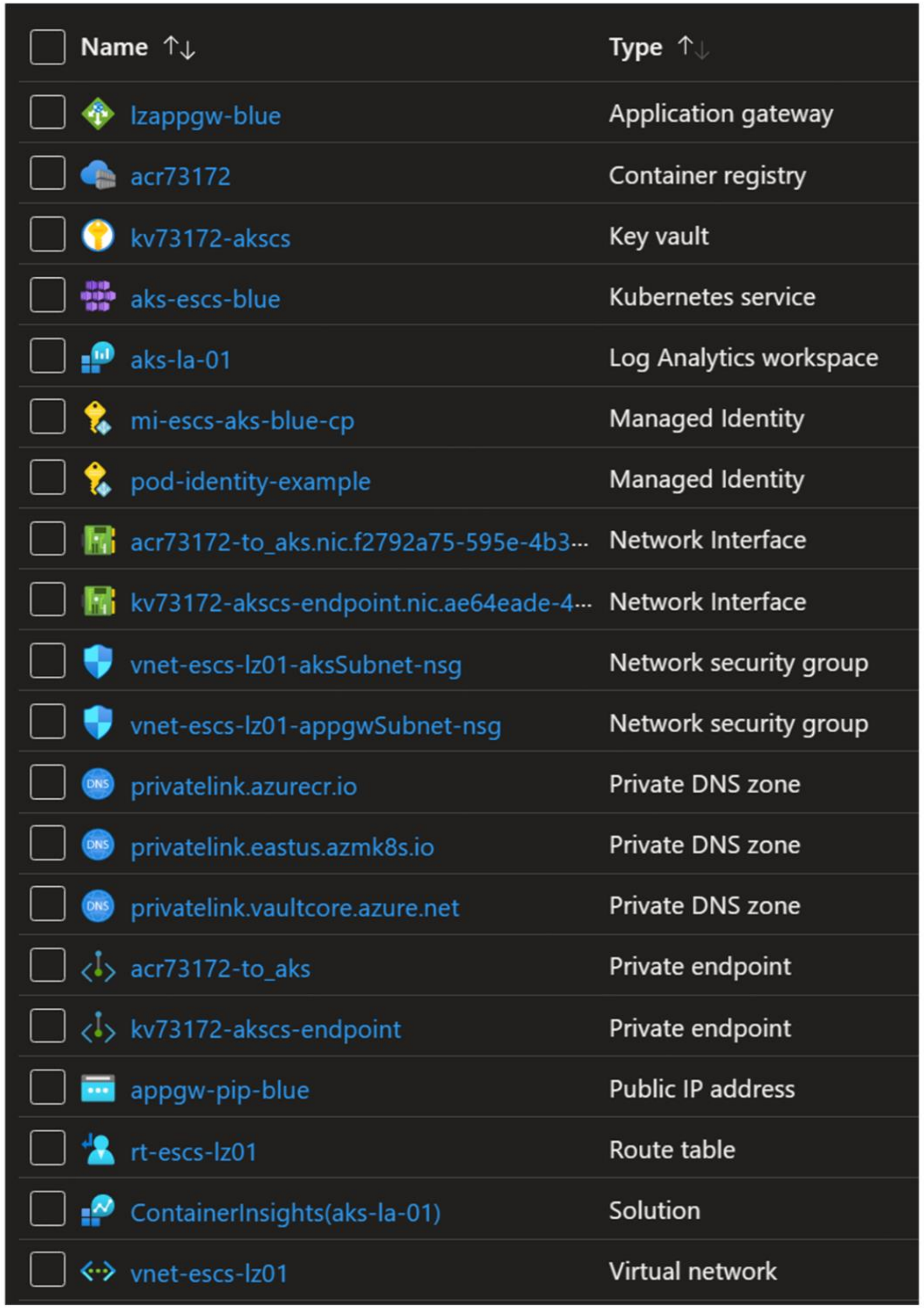
2. Create Azure AD groups
   AKS dev & admin groups

3. Create Network Hub
   VNET, Firewall, Bastion, VM

4. Create Network Spoke / LZ
   VNET, Peering, RT, AppGw, Pr. DNS Zones

5. Create resources for AKS
   ACR, Key vault, PE, Public DNS Zone

6. Create AKS cluster
   AKS, LA, MI, RBAC



□	Name ↑↓	Type ↑↓
□	lzappgw-blue	Application gateway
□	acr73172	Container registry
□	kv73172-akscs	Key vault
□	aks-escs-blue	Kubernetes service
□	aks-la-01	Log Analytics workspace
□	mi-escs-aks-blue-cp	Managed Identity
□	pod-identity-example	Managed Identity
□	acr73172-to_aks.nic.f2792a75-595e-4b3...	Network Interface
□	kv73172-akscs-endpoint.nic.ae64eade-4...	Network Interface
□	vnet-escs-lz01-aksSubnet-nsg	Network security group
□	vnet-escs-lz01-appgwSubnet-nsg	Network security group
□	privatelink.azurecr.io	Private DNS zone
□	privatelink.eastus.azmk8s.io	Private DNS zone
□	privatelink.vaultcore.azure.net	Private DNS zone
□	acr73172-to_aks	Private endpoint
□	kv73172-akscs-endpoint	Private endpoint
□	appgw-pip-blue	Public IP address
□	rt-escs-lz01	Route table
□	ContainerInsights(aks-la-01)	Solution
□	vnet-escs-lz01	Virtual network

Steps to deploy AKS LZ

1. Create Storage Acc for TF state
   TF state per level

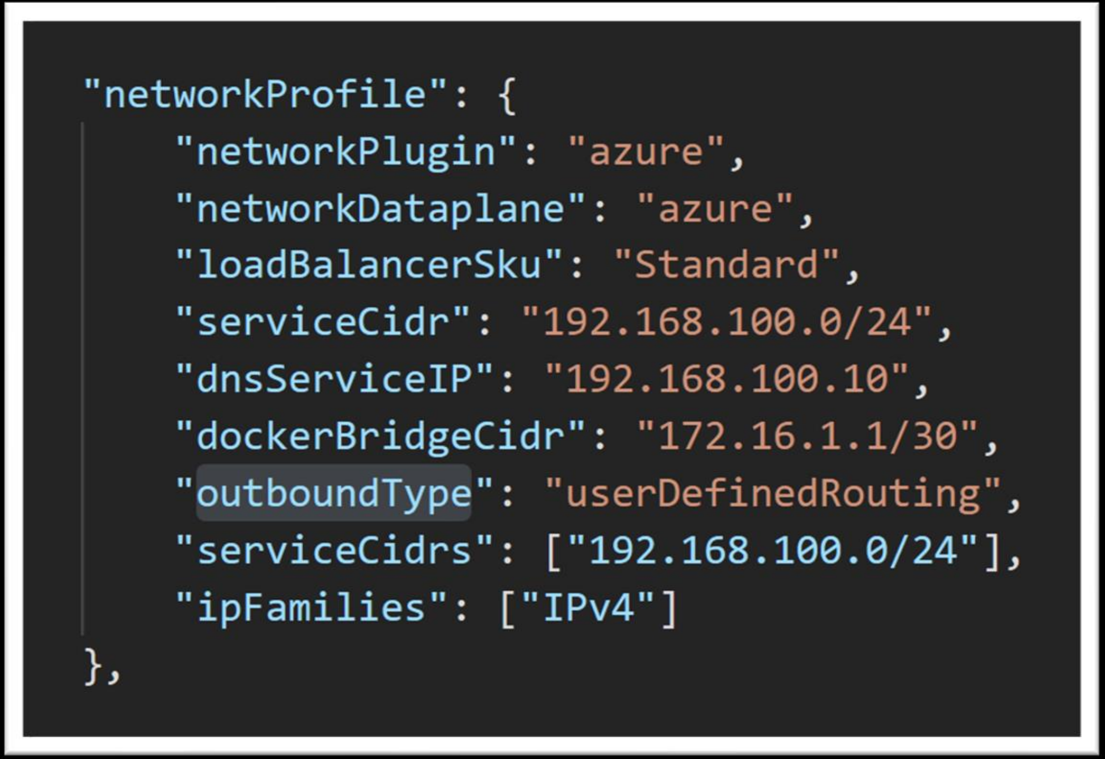
2. Create Azure AD groups
   AKS dev & admin groups

3. Create Network Hub
   VNET, Firewall, Bastion, VM

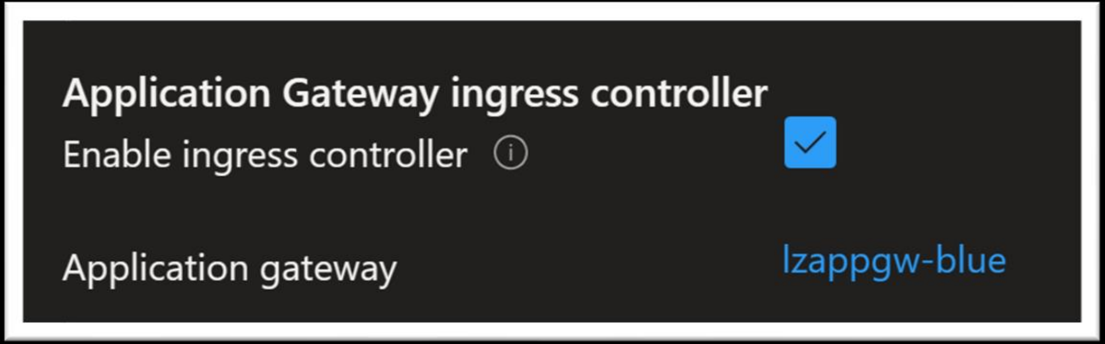
4. Create Network Spoke / LZ
   VNET, Peering, RT, AppGw, Pr. DNS Zones

5. Create resources for AKS
   ACR, Key vault, PE, Public DNS Zone

6. Create AKS cluster
   AKS, LA, MI, RBAC



```
"networkProfile": {
    "networkPlugin": "azure",
    "networkDataplane": "azure",
    "loadBalancerSku": "Standard",
    "serviceCidr": "192.168.100.0/24",
    "dnsServiceIP": "192.168.100.10",
    "dockerBridgeCidr": "172.16.1.1/30",
    "outboundType": "userDefinedRouting",
    "serviceCidrs": ["192.168.100.0/24"],
    "ipFamilies": ["IPv4"]
},
```

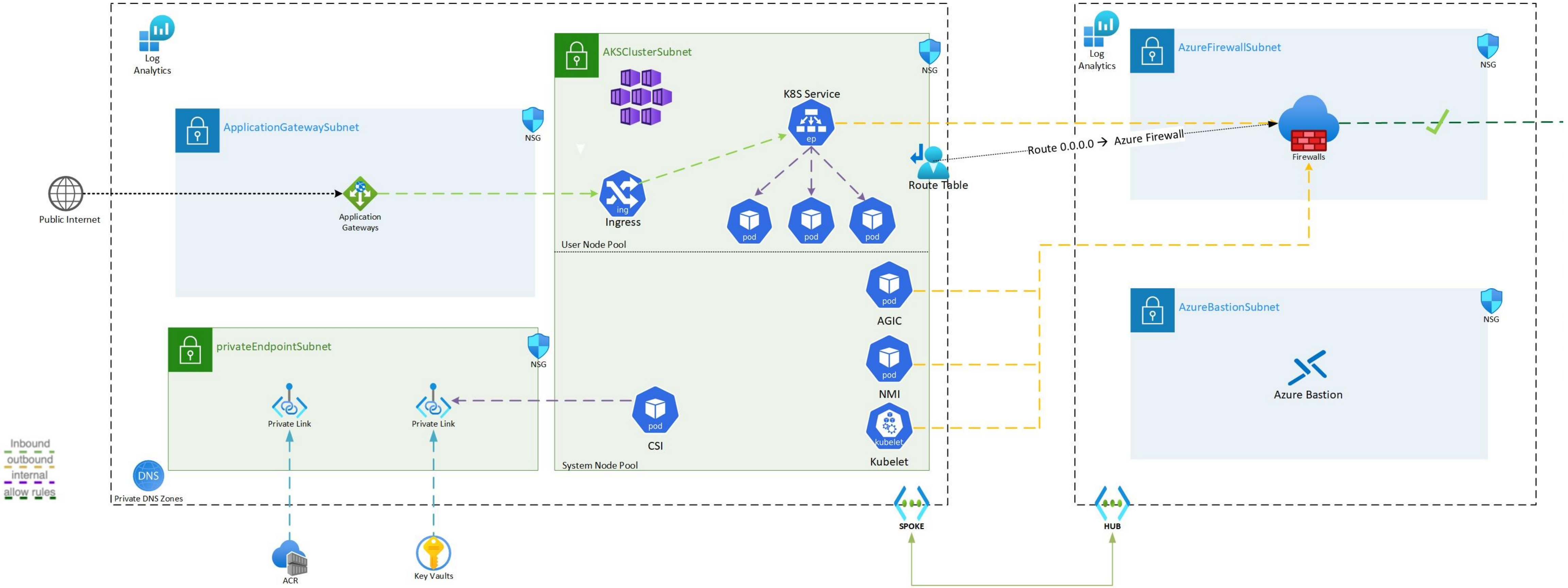


Application Gateway ingress controller

Enable ingress controller ⓘ ☑

Application gateway lzappgw-blue

AKS Landing Zone



Important notes

The AKS Landing Zone is a reference/standard implementation.
It is not a one size fits all.
Feel free to introduce changes.

Some key discussions and decisions:

- Application Gateway (AGIC) in the Hub or Spoke ?
- DNS centralized resolution in the Hub or in the Spoke ?
- Log Analytics for each Spoke/App or one for all Spokes ?

More resources

https://github.com/Azure/AKS-Landing-Zone-Accelerator