# .NET MICROSERVICES IN AKS

**Mike Douglas**
VP Digital Consulting - Engineering

LUNAVI

# ABOUT ME

- Mike Douglas
- Solution Consultant and VP Digital Consulting – Engineering at Lunavi
- Microsoft MVP – Developer Technologies – DevOps
- Organizer of Omaha DevOps Meetup
- Competitive Robotics Club Coordinator for 7th – 8th Graders
- @mikedouglasdev on twitter



LUNAVI

# GOALS

- Pros/Cons of Microservices / Why
- The Microservice (Inward)
- Multiple Microservices and AKS



LUNAVI

# WHY MICROSERVICES

# WHY MICROSERVICES?

- **What problem are we trying to solve?**
  - Less coupling - Independently deploy and release
  - Smaller changes
  - Business Focused, Not technical focused (DDD)
- **What problems do they cause?**
  - Not just architecture
  - Forces changes to engineering practices, people, and culture
  - Must be careful not to create a distributed "monolith"

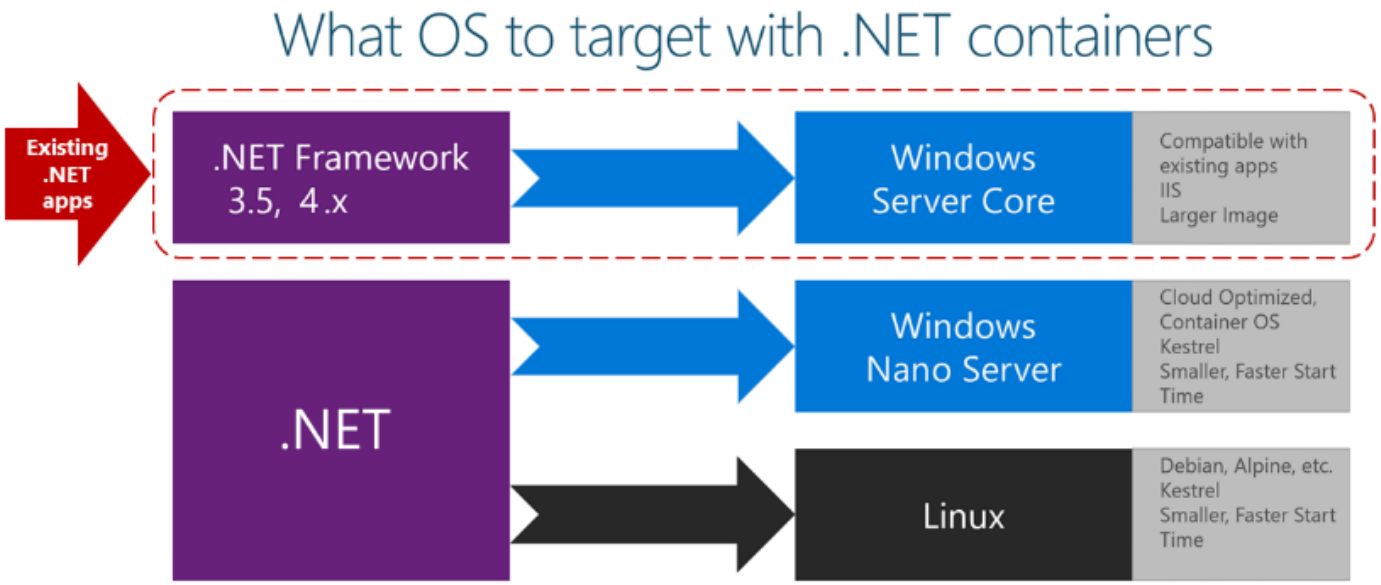




LUNAVI

# THE MICROSERVICE (INWARD)

# .NET ON CONTAINERS




Smaller, Production

Dev and Debugging

| Image | Comments |
|---|---|
| mcr.microsoft.com/dotnet/aspnet:**6.0** | ASP.NET Core, with runtime only and ASP.NET Core optimizations, on Linux and Windows (multi-arch) |
| mcr.microsoft.com/dotnet/sdk:**6.0** | .NET 6, with SDKs included, on Linux and Windows (multi-arch) |



LUNAVI

# DEMO

- .NET API on Docker



LUNAVI

# MULTIPLE MICROSERVICES

# MULTIPLE MICROSERVICES

- Challenges
- Patterns (BFF / API Gateway)
- Cross Cutting Concerns (Dapr)
- Lifecycle and Orchestration
  - AKS
  - AzDO Pipelines





LUNAVI

# CHALLENGES

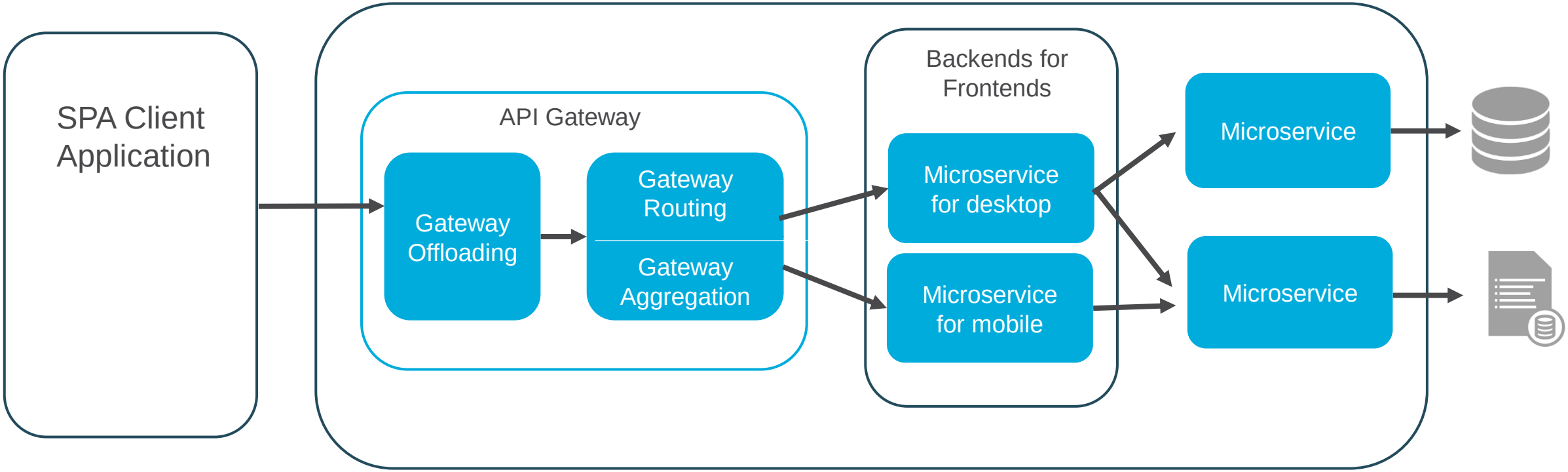
- APIs at Scale
- Distributed Tracing
- Service Discovery and Invocation / Coupling
- Security Access Restrictions
- Cost



LUNAVI

# PATTERNS



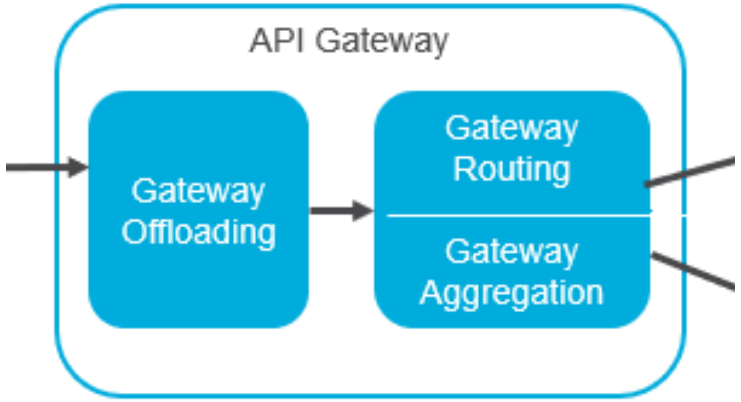





LUNAVI

# PATTERNS

- **API Gateway**
  - Single Entry Point
  - Provide a less chatty client specific response by aggregating multiple backend microservices responses
  - Features
    - Gateway Routing
    - Gateway Aggregation
    - Gateway Offloading
      - SSL termination
      - Authorization
      - Access restrictions



# PATTERNS

- **API Gateway Options**
  - Reverse Proxy ingress controller
    - NGINX
  - Service Mesh ingress controller
    - Istio, LinkerD
  - Azure Application Gateway
  - Azure API Management



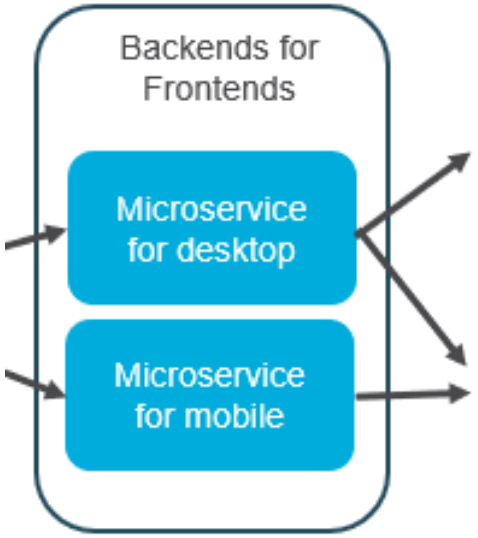




LUNAVI

# PATTERNS

- **Backends for Frontends**
  - Variance of API Gateway
  - Gateway API for each front-end API
  - Different clients need different data



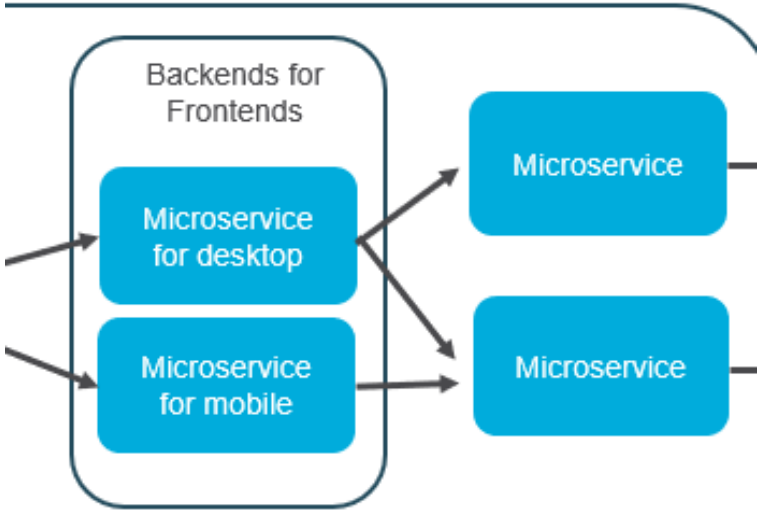


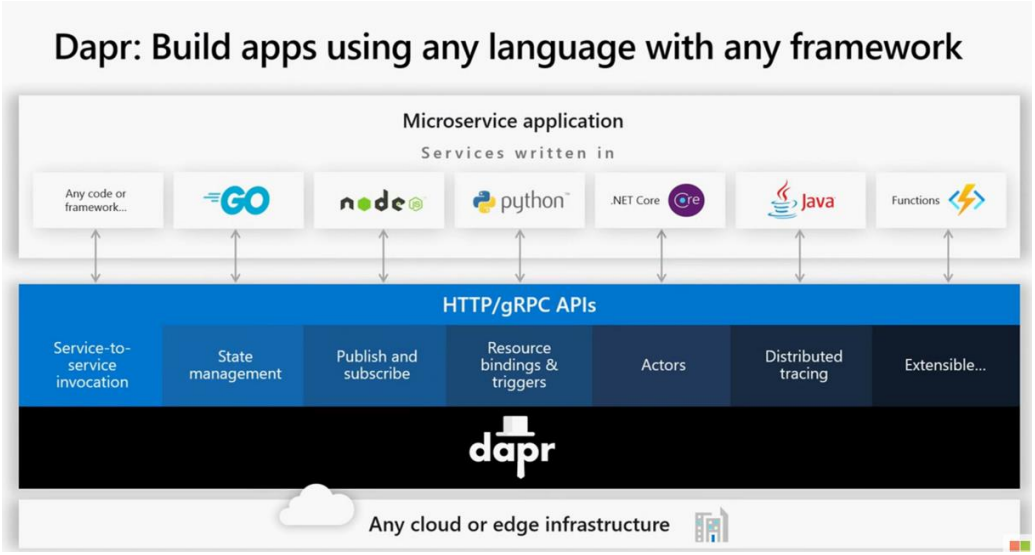



LUNAVI

# PATTERNS

- **Service to Service Communication**
  - Sidecar Container - Dapr
  - Azure Service Bus








LUNAVI

# RESILIENCY

```
var circuitBreakerPolicy = Policy.Handle<TransientException>()
        .CircuitBreaker(exceptionsAllowedBeforeBreaking: 3, durationOfBreak: TimeSpan.FromSeconds(10));

while (true)
{
        try
        {
                circuitBreakerPolicy.Execute(() =>
                {
                        SendRequest();
                        Log("Successfully sent request");
                });
                return;
        }
        catch(BrokenCircuitException)
        {
                Log("The circuit breaker tripped and is temporarily disallowing requests. Will wait before trying again");
                await Task.Delay(TimeSpan.FromSeconds(15));
        }
        catch (TransientException)
        {
                Log("Transient exception while sending request. Will try again.");
        }
}
```





https://makolyte.com/csharp-circuit-breaker-with-polly/

# RESILIENCY

- **AKS**
  - Infrastructure level
  - Load Balancing across multiple Nodes and Replica Sets (Pods)
  - Availability Zones
  - Load Balancing / DR in multiple regions (Region pairs)



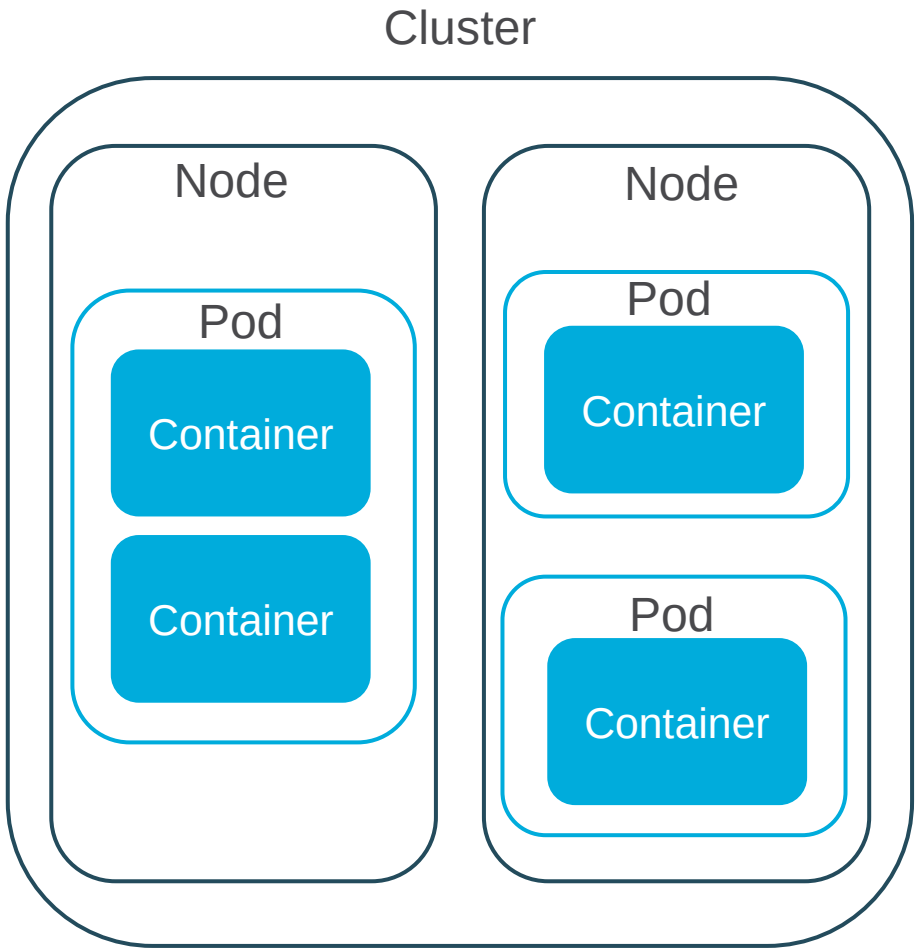


LUNAVI

# AZURE KUBERNETES SERVICE (AKS)

# WHY USE AZURE KUBERNETES SERVICE (AKS)?

- **What is Kubernetes?**
  - Open source orchestration tool for deploying, managing, and scaling container applications
  - Workload Resources
    - Pods
    - Deployments
    - Replica Set
    - Services
    - Namespace



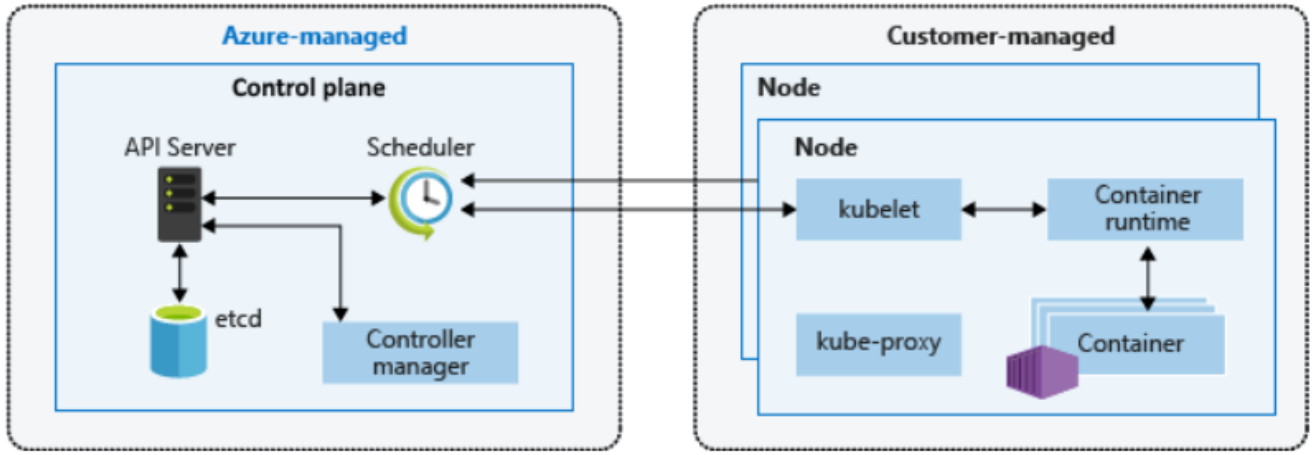




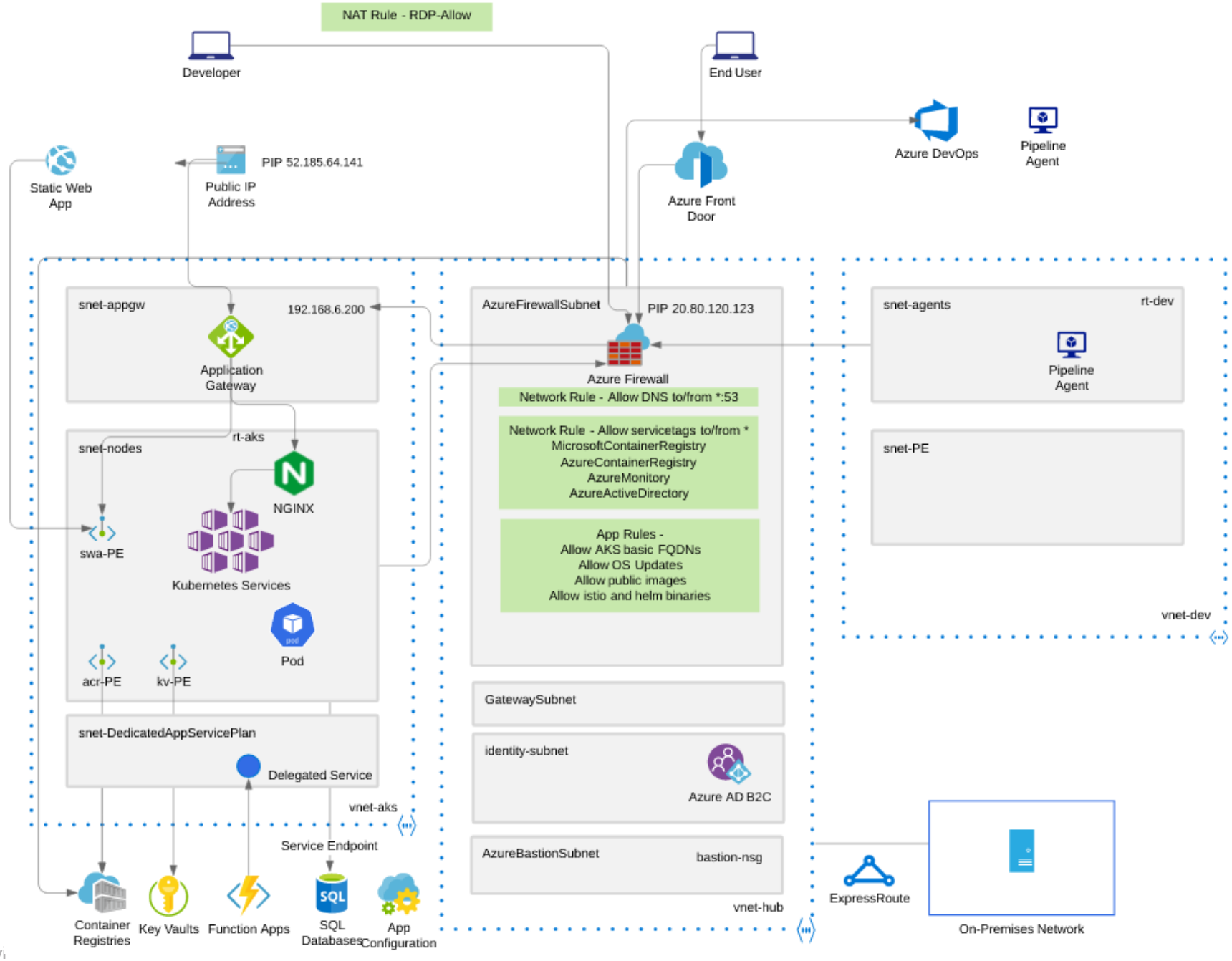

LUNAVI

# WHY USE AZURE KUBERNETES SERVICE (AKS)?

- **What is AKS?**
  - AKS (Azure Kubernetes Service) – Managed Kubernetes Service
    - You don’t worry about underlying servers
    - You also don’t pay for the management servers
  - Control Plane - nodes that provide the core Kubernetes services and orchestration of application workloads
  - Nodes  - Run your workloads

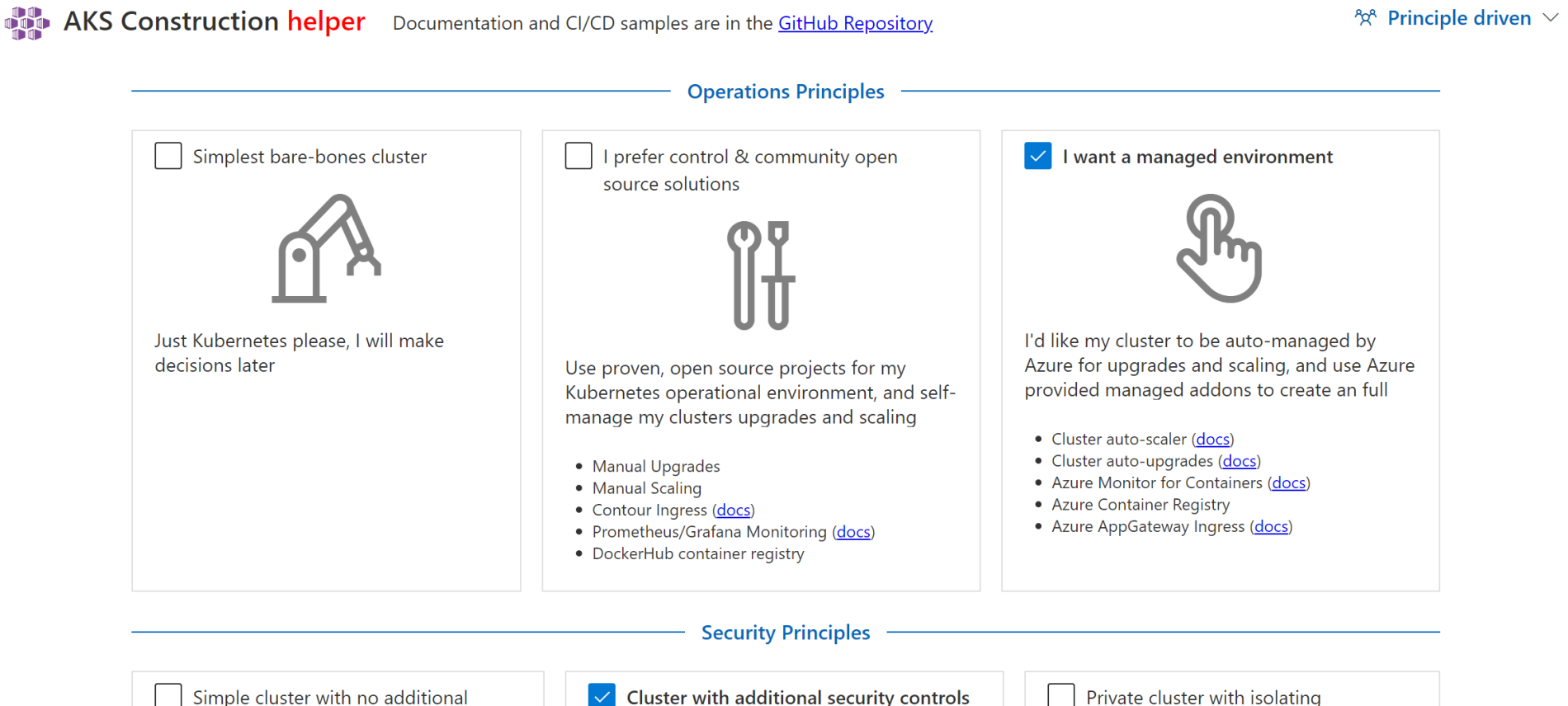


# AKS

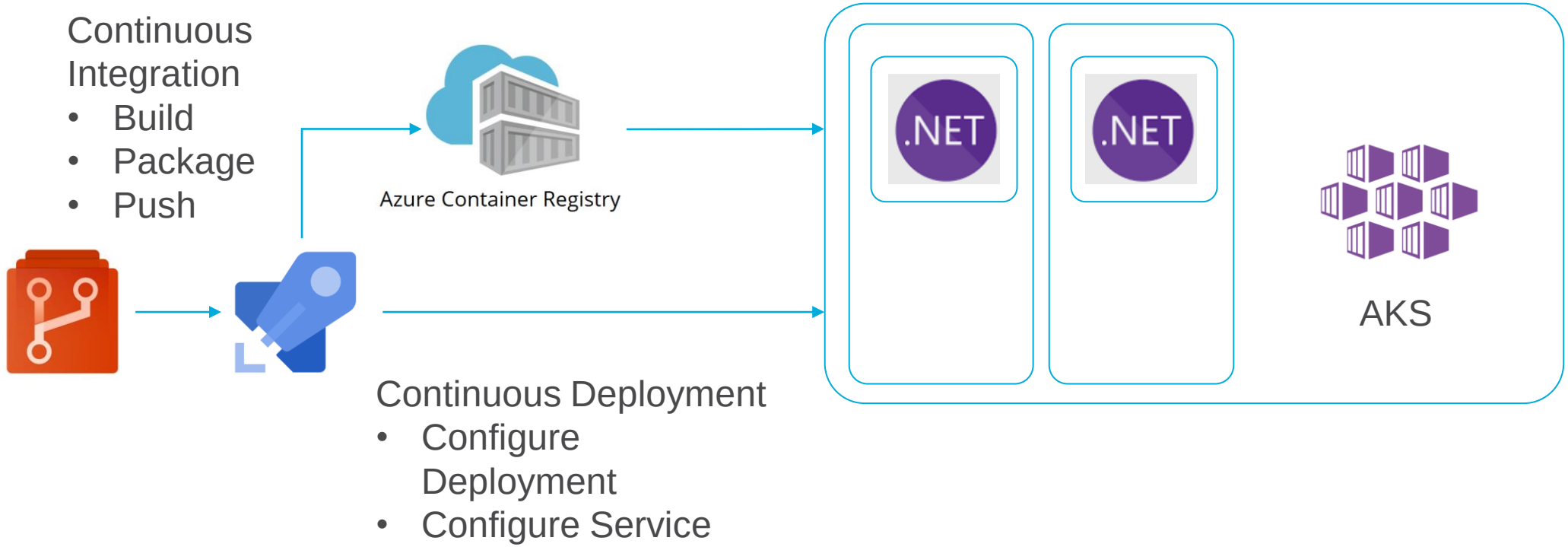

NAT Rule - RDP-Allow
Developer
End User
Static Web App
Public IP Address
PIP 52.185.64.141
Azure Front Door
Azure DevOps
Pipeline Agent
snet-appgw
192.168.6.200
Application Gateway
AzureFirewallSubnet
PIP 20.80.120.123
Azure Firewall
Network Rule - Allow DNS to/from *:53
Network Rule - Allow servicetags to/from *
MicrosoftContainerRegistry
AzureContainerRegistry
AzureMonitory
AzureActiveDirectory
App Rules -
Allow AKS basic FQDNs
Allow OS Updates
Allow public images
Allow istio and helm binaries
snet-agents
rt-dev
Pipeline Agent
snet-PE
vnet-dev
rt-aks
snet-nodes
NGINX
swa-PE
Kubernetes Services
Pod
acr-PE
kv-PE
snet-DedicatedAppServicePlan
Delegated Service
vnet-aks
Service Endpoint
GatewaySubnet
identity-subnet
Azure AD B2C
AzureBastionSubnet
bastion-nsg
vnet-hub
ExpressRoute
On-Premises Network
Container Registries
Key Vaults
Function Apps
SQL Databases
App Configuration

# DEMO

- **AKS Deployment**
- **AKS Construction helper (azure.github.io)**



# AKS APPLICATION PIPELINES



# DEMO

- **weather-api AKS pipeline**





LUNAVI

# LINKS

- **Kubernetes Cheat Sheet**
  - https://linuxacademy.com/site-content/uploads/2019/04/Kubernetes-Cheat-Sheet_07182019.pdf
- **Hands On Labs**
  - https://aksworkshop.io/
  - https://kubesec.aksworkshop.io/ (some items in lab are outdated)
- **Azure Pipelines HOL**
  - https://docs.microsoft.com/en-us/learn/modules/deploy-kubernetes/
- **Dapr**
  - https://dapr.io/

# QUESTIONS?