AZURE DAY ROME 2024

Implementing a Zero Trust model on AKS with the new Istio Service Mesh add-on for Microsoft Azure

Carlo Pio Luigi Ciciriello Luca Passaretta

AZURE DAY

Who We Are

Hi there. Here's Carlo

Network Hybrid Cloud & Multi-Cloud Architect @ Kyndryl
Certified Technical Trainer
Cisco CCIE # 67718 | Tech Blogger & Public Speaker

Microsoft certifications:

- Azure Solutions Architect Expert
- Cybersecurity Architect Expert
- Azure Network Engineer Associate
- Azure Administrator Associate
- Security Operations Analyst Associate
- Azure Fundamentals
- Azure AI Fundamentals

Key Focus Areas:
Cloud Networking | Cloud Security | DevOps | AI | Network Programmability | Network Automation | Network Observability | Network Security | IPv6 | BGP Security

https://it.linkedin.com/in/cplciciriello

Hi there. Here's Luca

Cloud Architect & Platform Engineer @ Kyndryl
Tech Blogger & Public Speaker

Microsoft certifications:

- Azure Solutions Architect Expert
- DevOps Engineer Expert
- Azure Security Engineer Associate
- Azure Administrator Associate
- Azure Fundamentals

The Linux Foundation certifications:

- CKA: Certified Kubernetes Administrator

Key Focus Areas:
Cloud & Cloud Native | DevOps | K8s | Terraform & Scripting | Data & AI | DevSecOps & GitOps | Identity | Cloud Networking | Cloud Security | DB | Event-Driven & Message-Based Solutions | Software & App Solutions

https://it.linkedin.com/in/lucapassaretta

© Copyright Kyndryl, Inc. 2024

kyndryl

AZURE DAY

Monolithic Architecture

- Traditional approach to building applications where the entire application is built as a single, monolithic unit
- Better suited for smaller projects
- Requires less technical knowledge
- Fewer parts to manage
- Simplified codebase (easier development and maintenance)

© Copyright Kyndryl, Inc. 2024

kyndryl

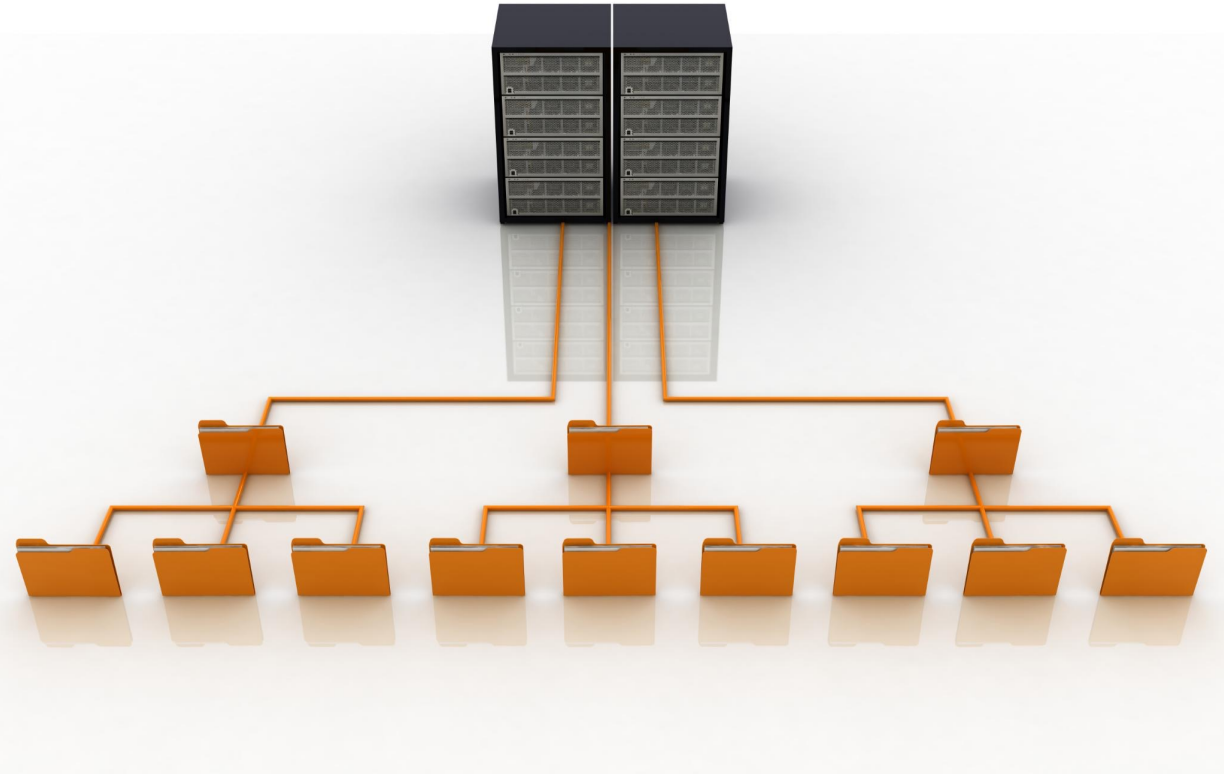
Challenges of Monolithic Architecture

- Quick feature roll-out requires coordinated effort
- Deprecation of subsystems may require complete rewrite of multiple subsystems
- Steep learning curve for new developers due to tight coupling between subsystems
- Scaling can be difficult and costly in a monolith application

© Copyright Kyndryl, Inc. 2024

Microservice-based Architecture

- Modern approach to building applications where individual services are treated as independently deployable components: MICROSERVICES
- Designed to address the challenges of monolithic architecture



© Copyright Kyndryl, Inc. 2024

AZURE DAY

Benefits of Microservice-based Architecture

- Enables greater flexibility and modularity in application development
- Provides better visibility, control, and security over the network traffic
- Allows for better resiliency, improved management, and reduced costs.

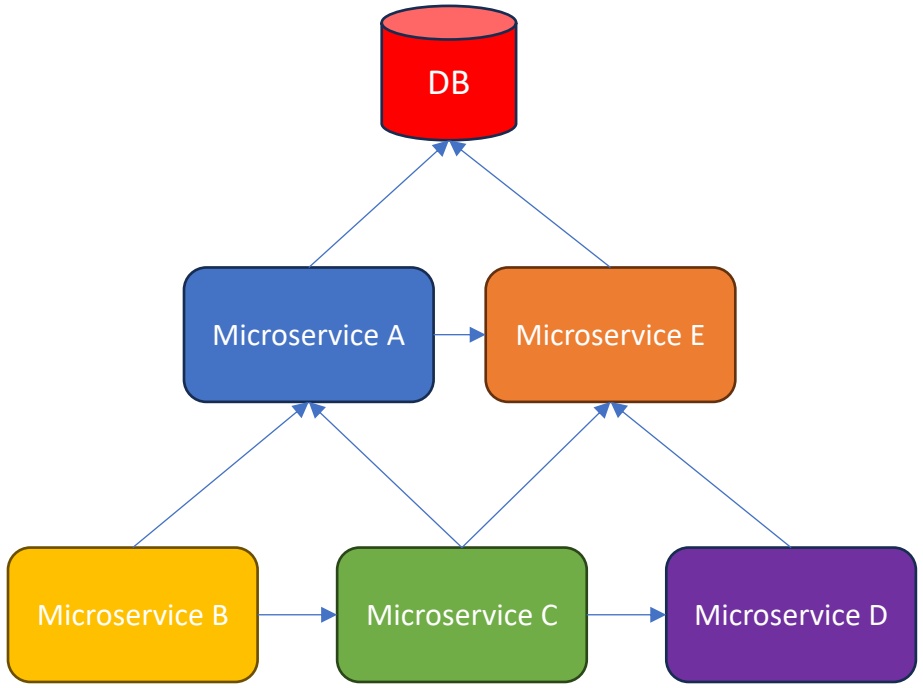
© Copyright Kyndryl, Inc. 2024

kyndryl

AZURE DAY

Challenges of Microservice-based Architecture

- Modern applications are architected as distributed systems, where each microservice has a fixed responsibility
- These microservices are part of a family. And just like members of any family, they need to communicate with each other
- With a growing number of microservices, managing, observing, and securing the communication between them requires as much effort as it takes to keep adding new features in your application
- You start facing problems related to:
  - microservice discovery
  - load balancing the traffic between microservices
  - graceful recovery from failures of microservices
  - monitoring the microservice communication
  - collect metrics
  - risk of data breaches and unauthorized access



Microservices deployed in a Kubernetes cluster

© Copyright Kyndryl, Inc. 2024

kyndryl

AZURE DAY

Introduction to Zero Trust Architecture and Istio Service Mesh

© Copyright Kyndryl, Inc. 2024

kyndryl

AZURE DAY

What is Zero Trust Architecture?

Zero Trust Architecture

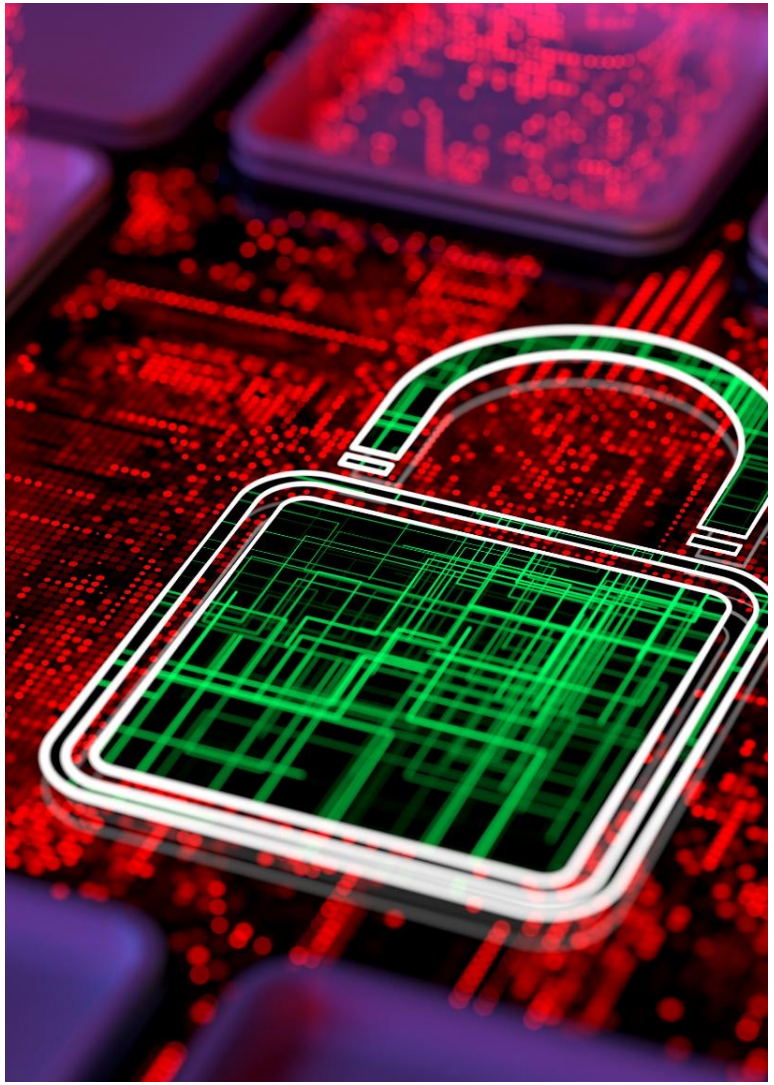
Security model that assumes that all network traffic is untrusted and must be authenticated and authorized.

Principle of Least Privilege

Key principle of Zero Trust Architecture that limits users' access to only the resources they need to do their jobs, reducing the risk of data breaches and unauthorized access.

Continuous Monitoring

Key principle of Zero Trust Architecture that involves continuous monitoring of all network traffic for signs of suspicious activity, enabling organizations to detect and respond to cyber attacks in real-time.



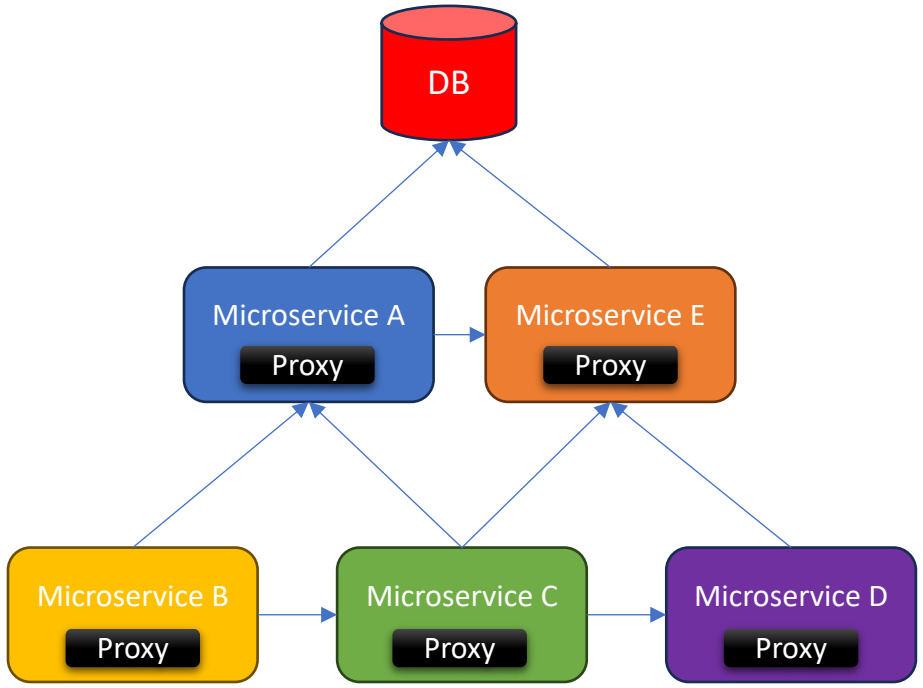
© Copyright Kyndryl, Inc. 2024

kyndryl

AZURE DAY

What is a Service Mesh?

- An **architecture pattern** to **effectively manage** the **communication** between microservices:
  - not limited to Kubernetes deployed services
  - not specific to any cloud
- It is an additional infrastructure layer that you can add to your application, getting capabilities like observability, traffic management, and security, for which you would otherwise have to write code and maintain that on your own
- With the **proxy instance** (aka **sidecar**) injected alongside every microservice, all the incoming and outgoing communications from a microservice go via proxy
- All proxies are connected to each other via a Control plane component of Service Mesh (not shown in the picture)
- Concerns like service discovery, load balancing, monitoring, resiliency and so on, are not tied with business logic and **abstracted** by the proxy
- Business logic stays inside your application container, making it more fit-for-purpose



Service Mesh in a Kubernetes cluster

© Copyright Kyndryl, Inc. 2024

kyndryl

AZURE DAY

What is Istio?

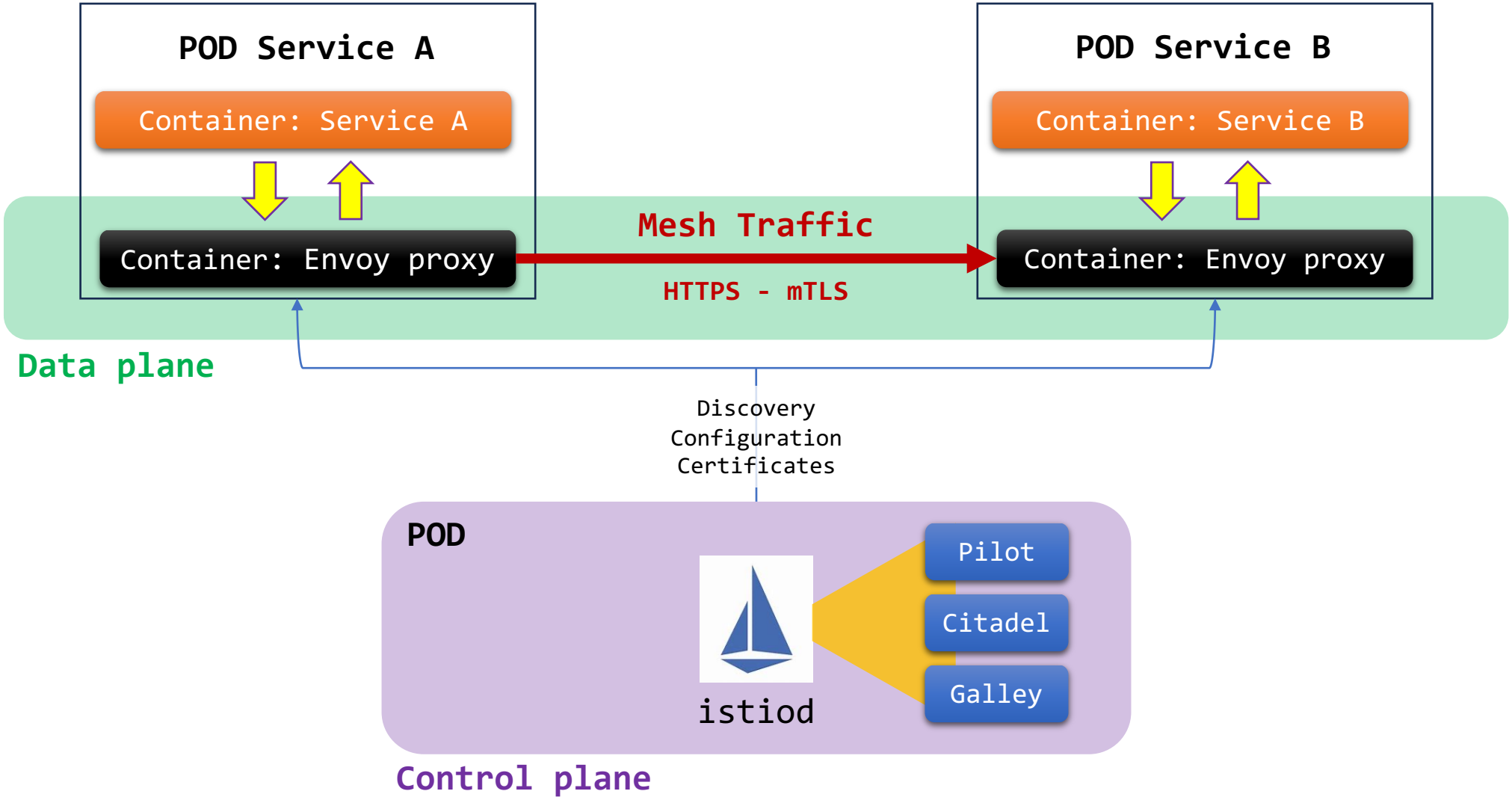
- Open-source, platform-independent service mesh
- Started by teams from IBM and Google in partnership with the team from Lyft that built Envoy proxy
- It is now part of the Cloud Native Computing Foundation
- Over the years, it has become THE Service Mesh that provides:
  - traffic management
  - policy enforcement
  - telemetry collection

© Copyright Kyndryl, Inc. 2024

kyndryl

AZURE DAY

Istio Service Mesh Architecture



© Copyright Kyndryl, Inc. 2024

kyndryl

Istio Service Mesh Architecture
Data Plane

The Data Plane is composed of a set of Envoy proxies that are deployed as sidecars in front of each microservice.

When Service A wants to talk to Service B, the traffic goes through the proxies. Thus, the proxies get to know and control all network communication between the services.

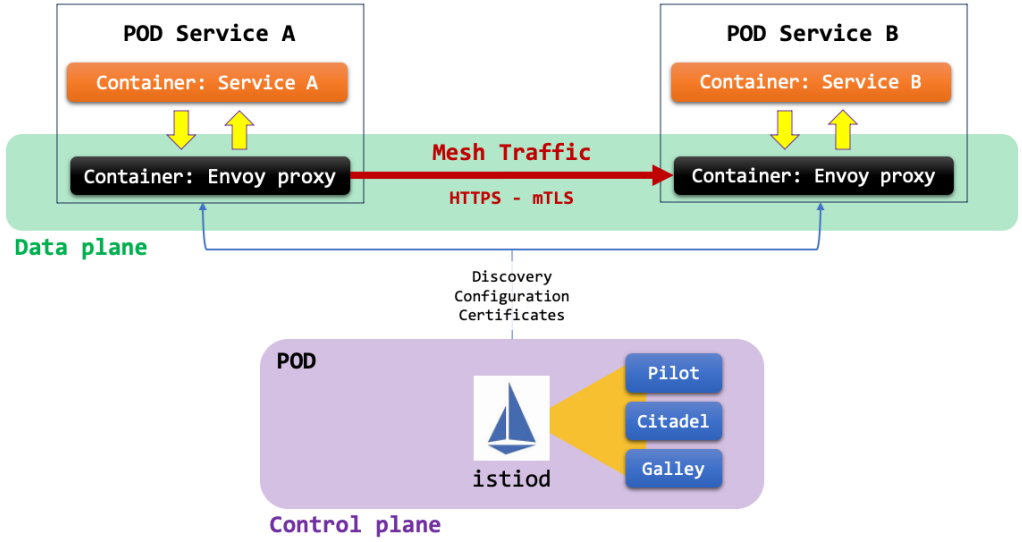
The Envoy proxy has many built-in features, as follows:

- Load balancing
- Dynamic service discovery
- TLS termination
- Circuit breakers
- Health checks
- Staged rollouts with %-based traffic split
- Fault injection
- Rich metrics

For the application development teams, this reduces a lot of rearchitecting and coding, which would have otherwise required to build the istio-like capabilities in the system.

The Data Plane helps build the following features for your application:

1. **Traffic management** using **routing rules** to enforce fine-gained control over the traffic
2. **Resiliency features** to set up retries, failovers, circuit breaking, rate limiting and fault injection
3. **Security features** to enforce security policies and access control
4. **Custom policy enforcement** and **telemetry generation** for mesh traffic



© Copyright Kyndryl, Inc. 2024

Istio Service Mesh Architecture
Control Plane

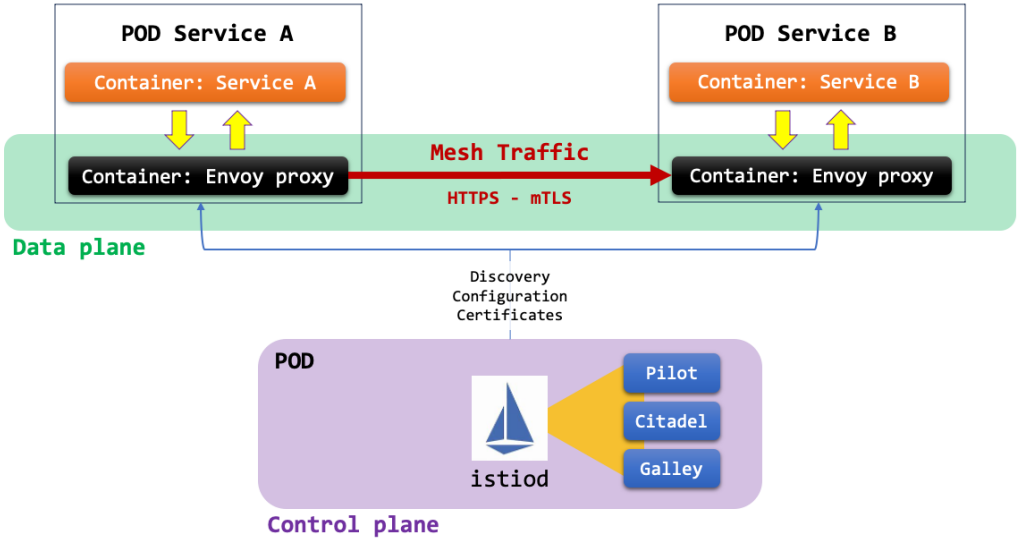
The Control Plane is provided by a single binary called **Istiod**.

The Traffic Management API from Istio can be used to instruct Istiod to refine the Envoy configuration for more granular control over the traffic between microservices.

PILOT: spreads **configuration** to proxies (routing, load balancing, resiliency, service discovery) at runtime.

CITADEL: built-in **certificate authority**, used to generate certificates to allow secure communication with mutual TLS in the Data Plane. It manages **authentication**, **access control** and **encryption**.

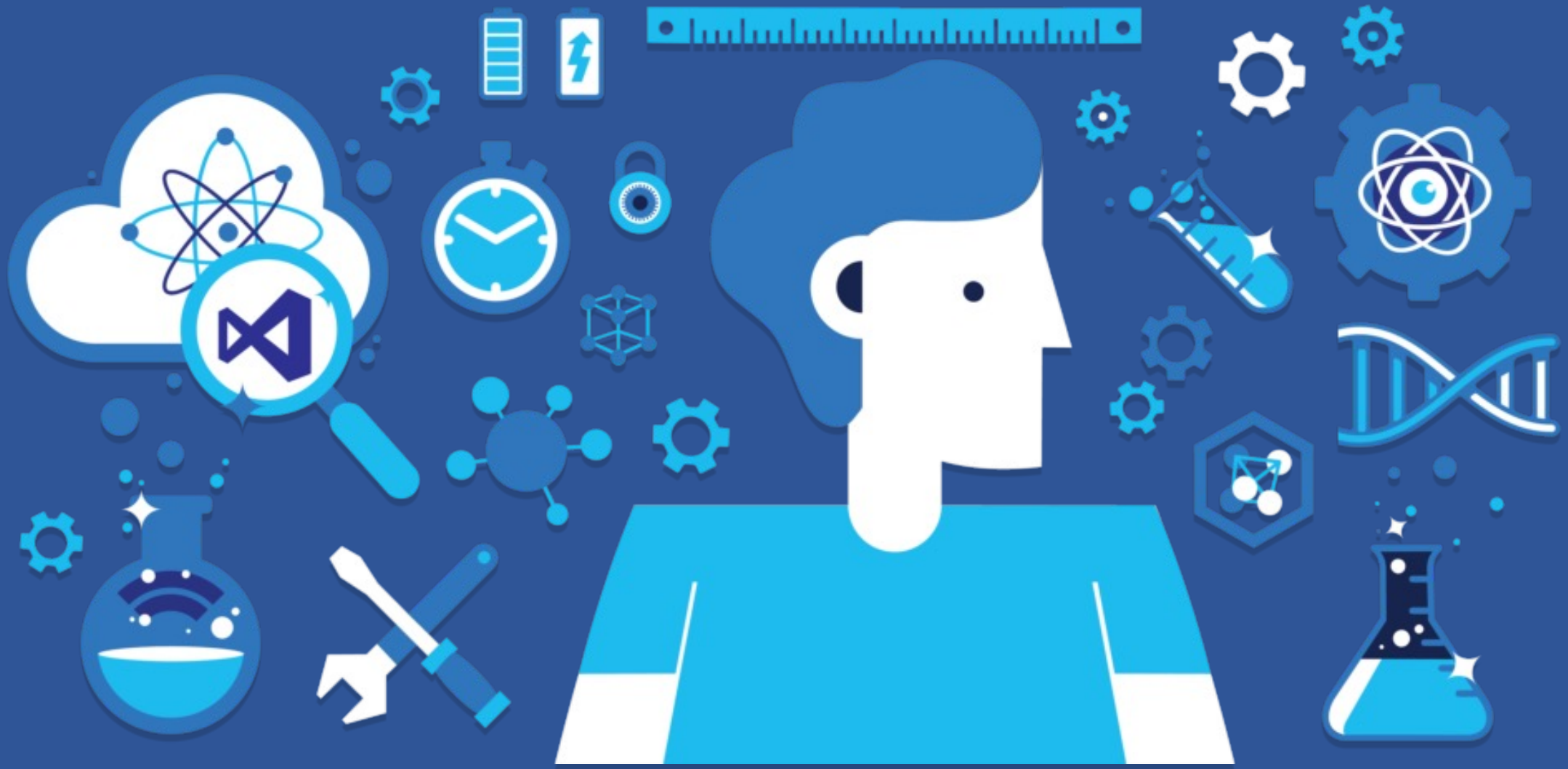
GALLEY: can be considered a **management plane**. It **stores the user configuration**, **validates** it, and then **sends it to Pilot** for distribution towards the proxies.



© Copyright Kyndryl, Inc. 2024

AZURE DAY

Demo



AZURE DAY

Demo - Agenda

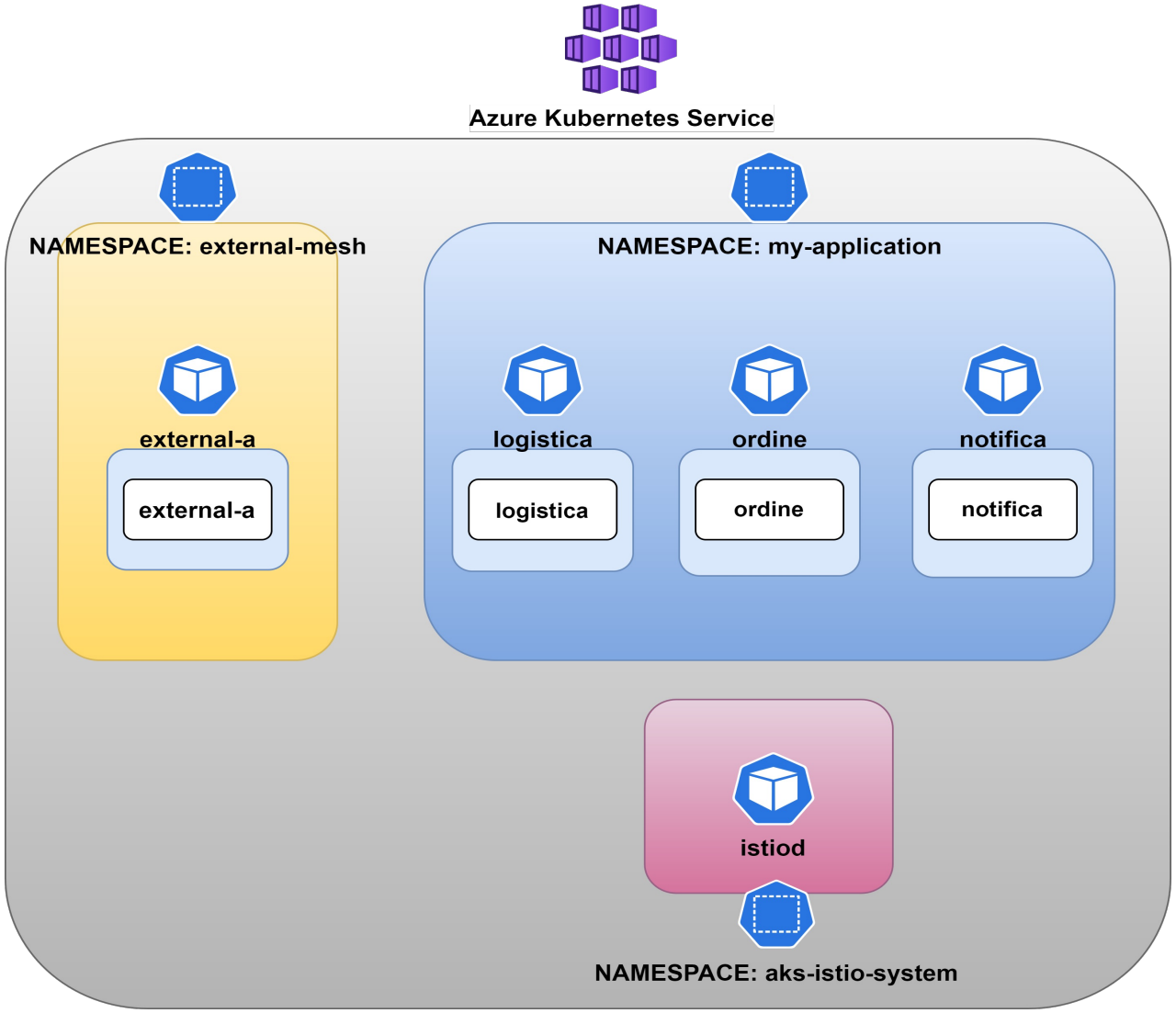
1. Cluster AKS Initial Setup
2. Istio Deploy via AKS Add-On
3. Injection Sidecar
4. mTLS
5. Access Control

© Copyright Kyndryl, Inc. 2024

kyndryl

AZURE DAY

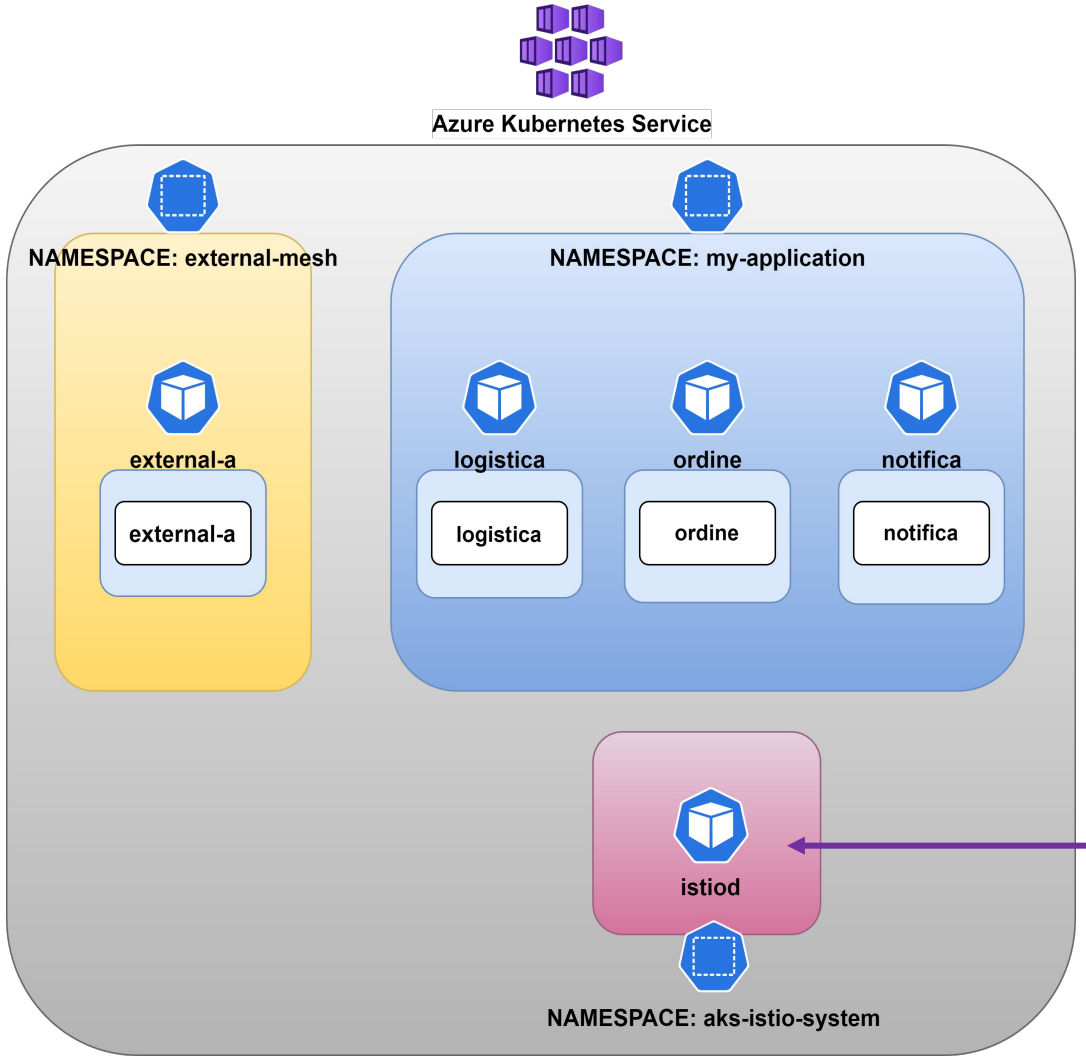
Initial Setup



© Copyright Kyndryl, Inc. 2024

kyndryl

Initial Setup

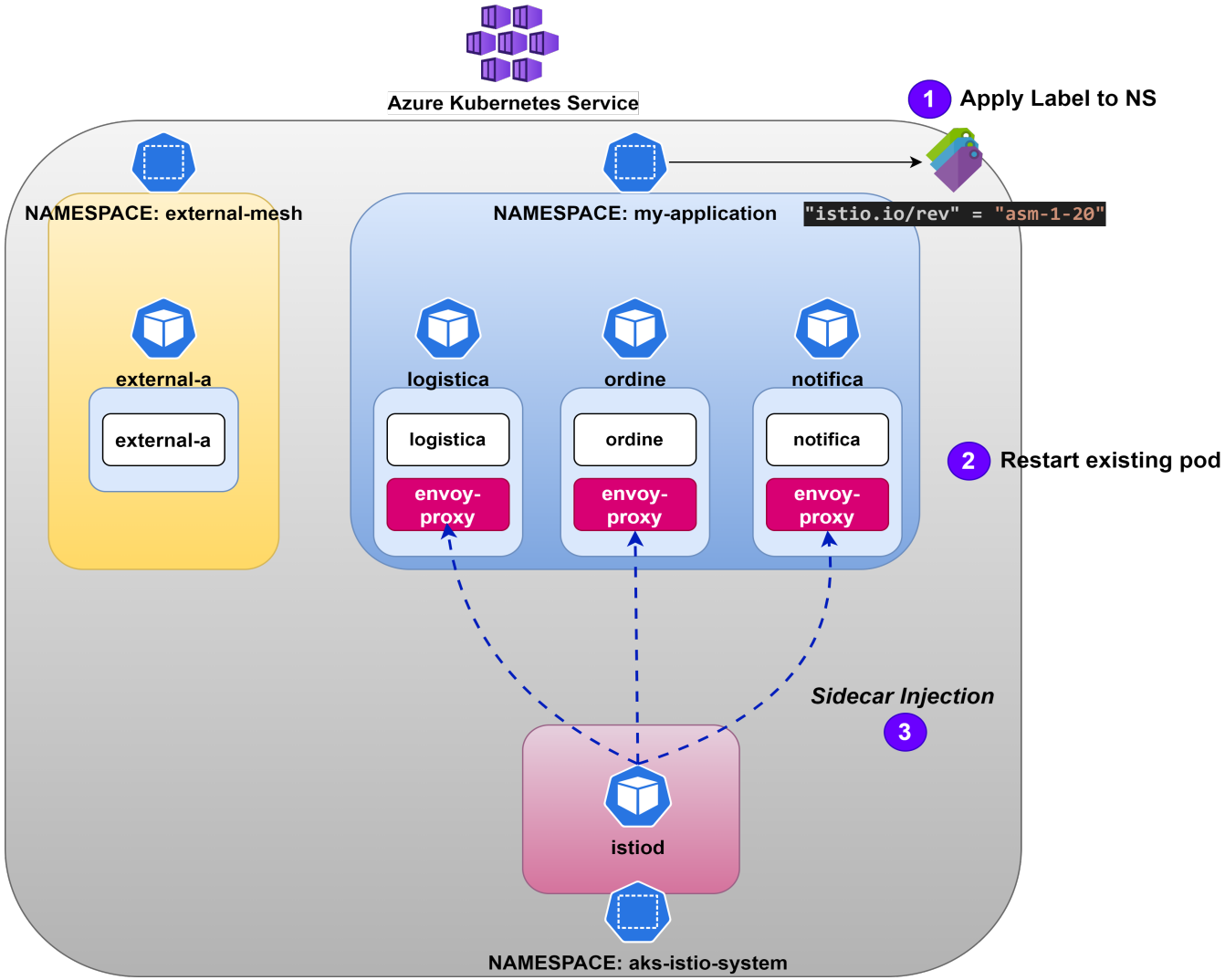



```tf
resource "azurerm_kubernetes_cluster" "azure_day" {
  name                = "aks-ne-azday-istio-${local.count}"
  location            = azurerm_resource_group.azure_day.location
  resource_group_name = azurerm_resource_group.azure_day.name
  dns_prefix          = "aks-ne-azday-${local.count}"

  sku_tier = "Free"

  default_node_pool {
    name           = "main"
    node_count     = 2
    vm_size        = "Standard_D2_v2"
    vnet_subnet_id = azurerm_subnet.aks.id
  }

  identity {
    type = "SystemAssigned"
  }

  network_profile {
    network_plugin = "kubenet"
    network_policy = "calico"
  }

  # ISTIO SERVICE MESH AKS ADD-ON
  service_mesh_profile {
    mode = "Istio"
    #external_ingress_gateway_enabled = true
    #internal_ingress_gateway_enabled = true
  }

  tags = {
    Environment = "Demo"
  }
}
```

© Copyright Kyndryl, Inc. 2024

AZURE DAY

Istio Sidecar Injection

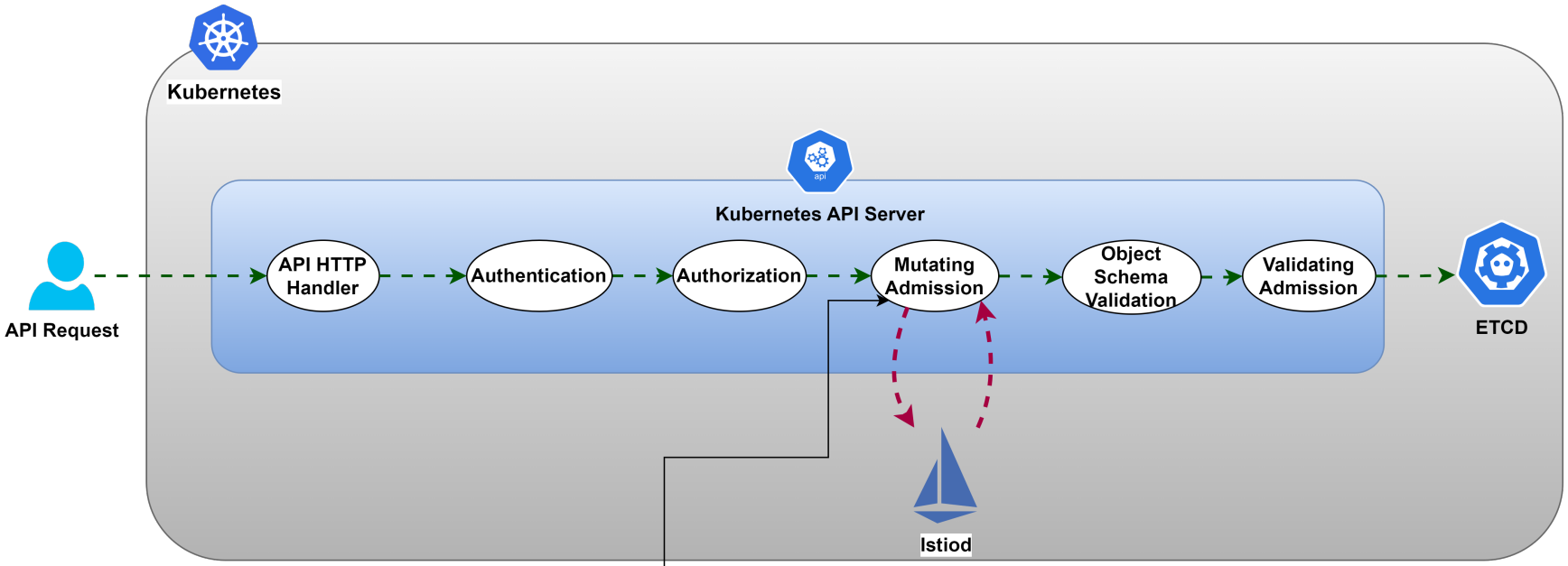


© Copyright Kyndryl, Inc. 2024

kyndryl

AZURE DAY

How Istio Sidecar Injection Happens




```
APIVersion: admissionregistration.k8s.io/v1beta1
kind: MutatingWebhookConfiguration
metadata:
  name: istio-sidecar-injector
webhooks:
  - name:  rev.namespace.sidecar-injector.istio.io
    service:
      name: istiod
      namespace: istio-system
      path: /inject
      port: 443
...
...
```

© Copyright Kyndryl, Inc. 2024

kyndryl

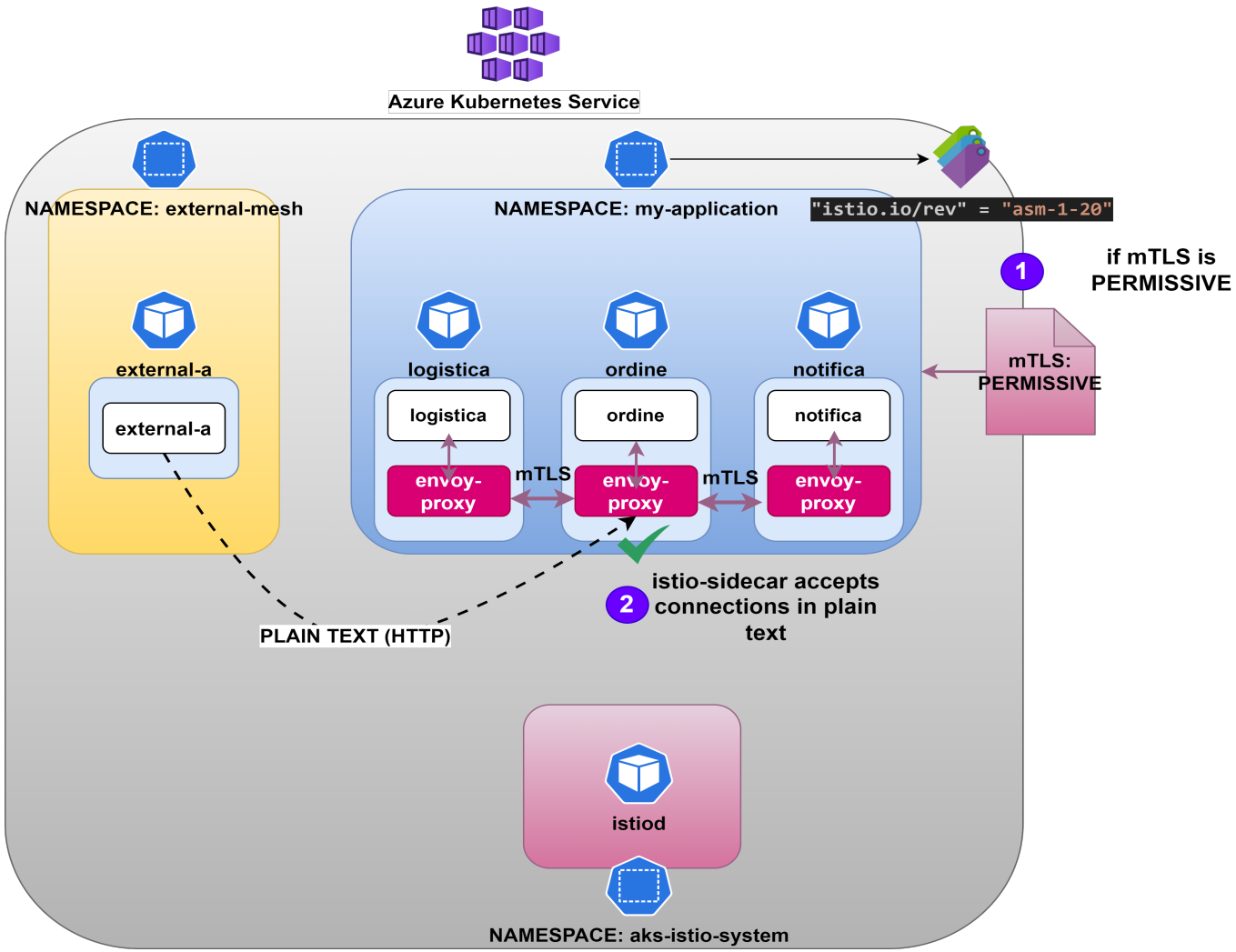
AZURE DAY

Demo Session

© Copyright Kyndryl, Inc. 2024

kyndryl

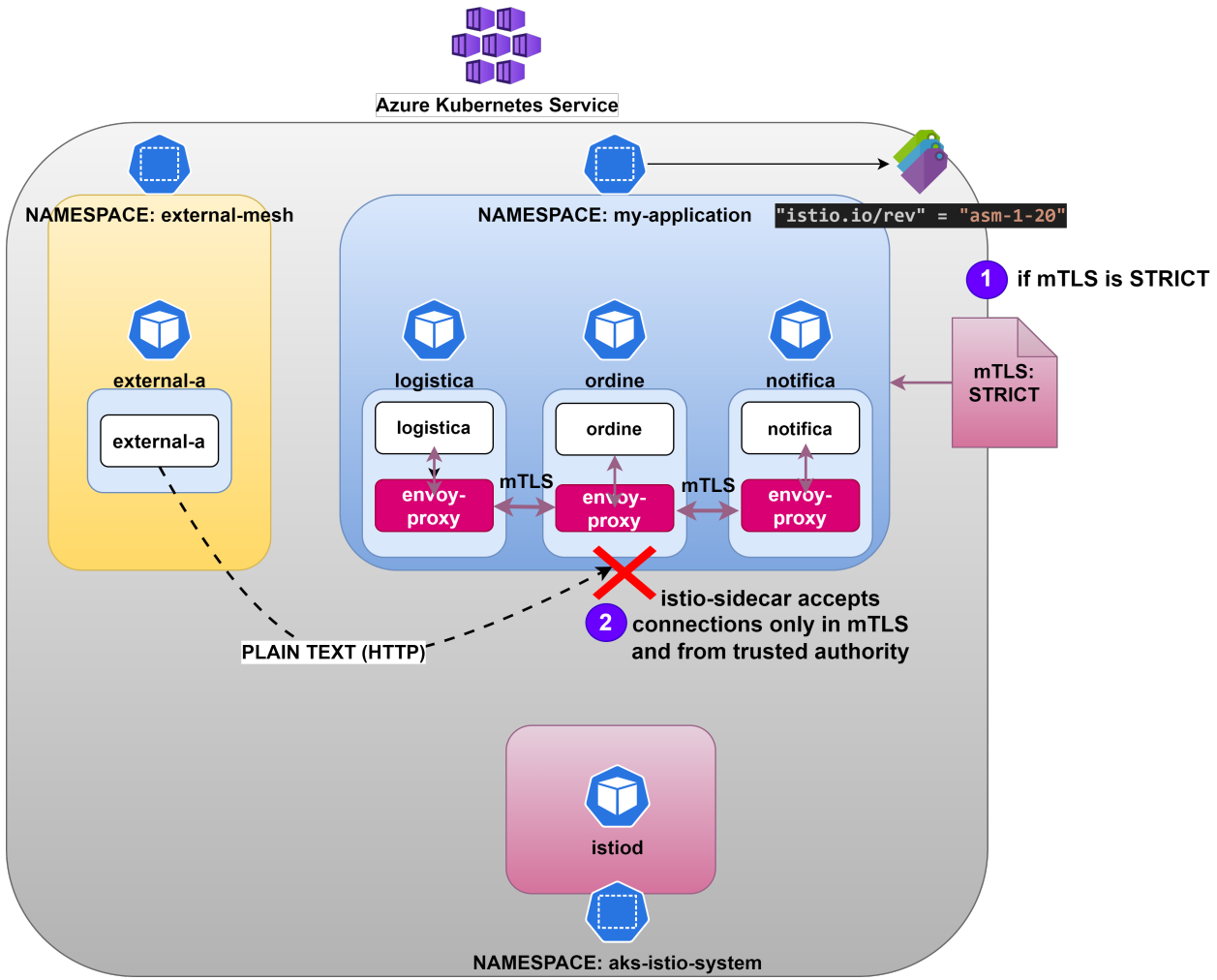
mTLS Permissive



© Copyright Kyndryl, Inc. 2024

AZURE DAY

mTLS Strict



© Copyright Kyndryl, Inc. 2024

kyndryl

AZURE DAY

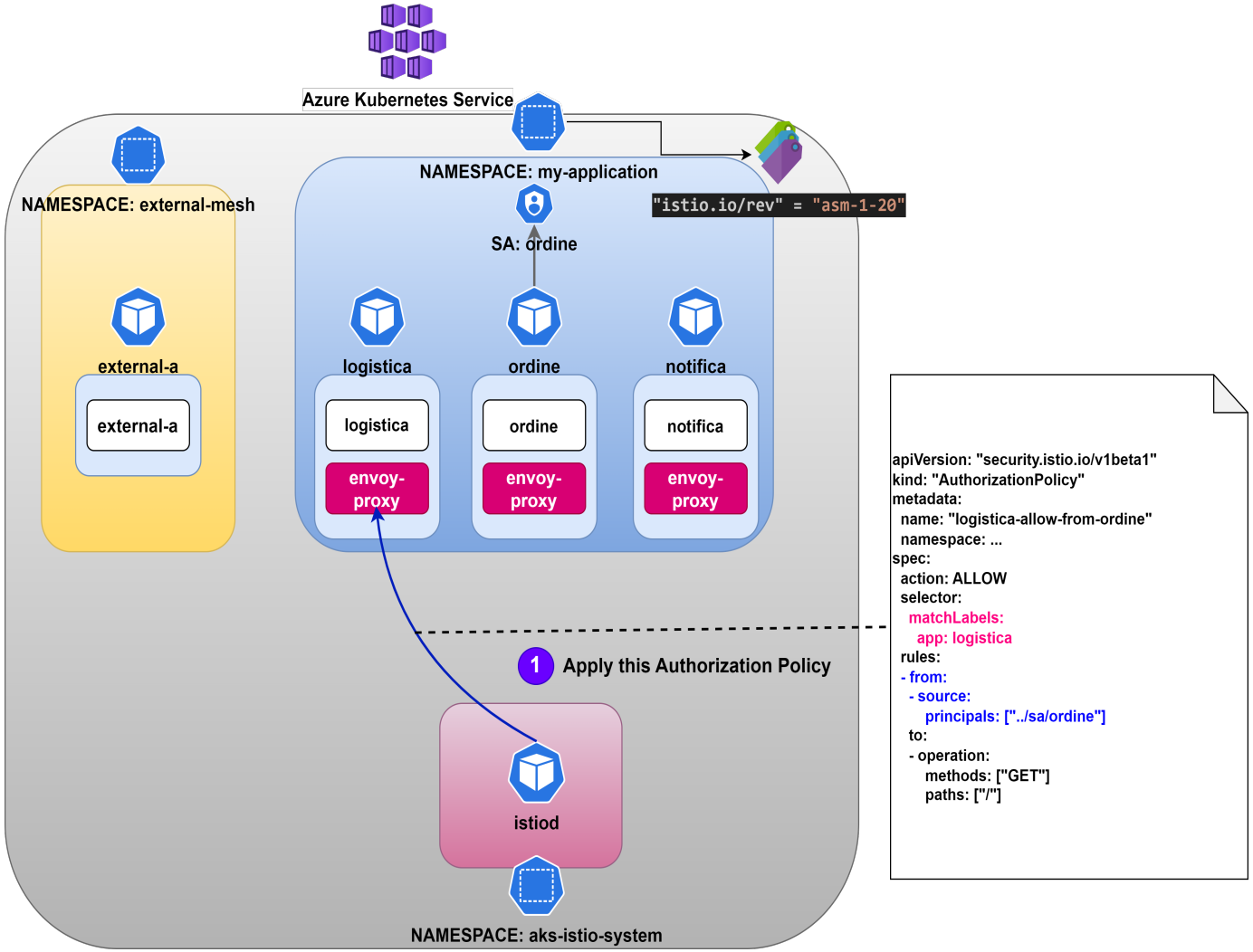
Demo Session

© Copyright Kyndryl, Inc. 2024

kyndryl

AZURE DAY

Authorization Policy



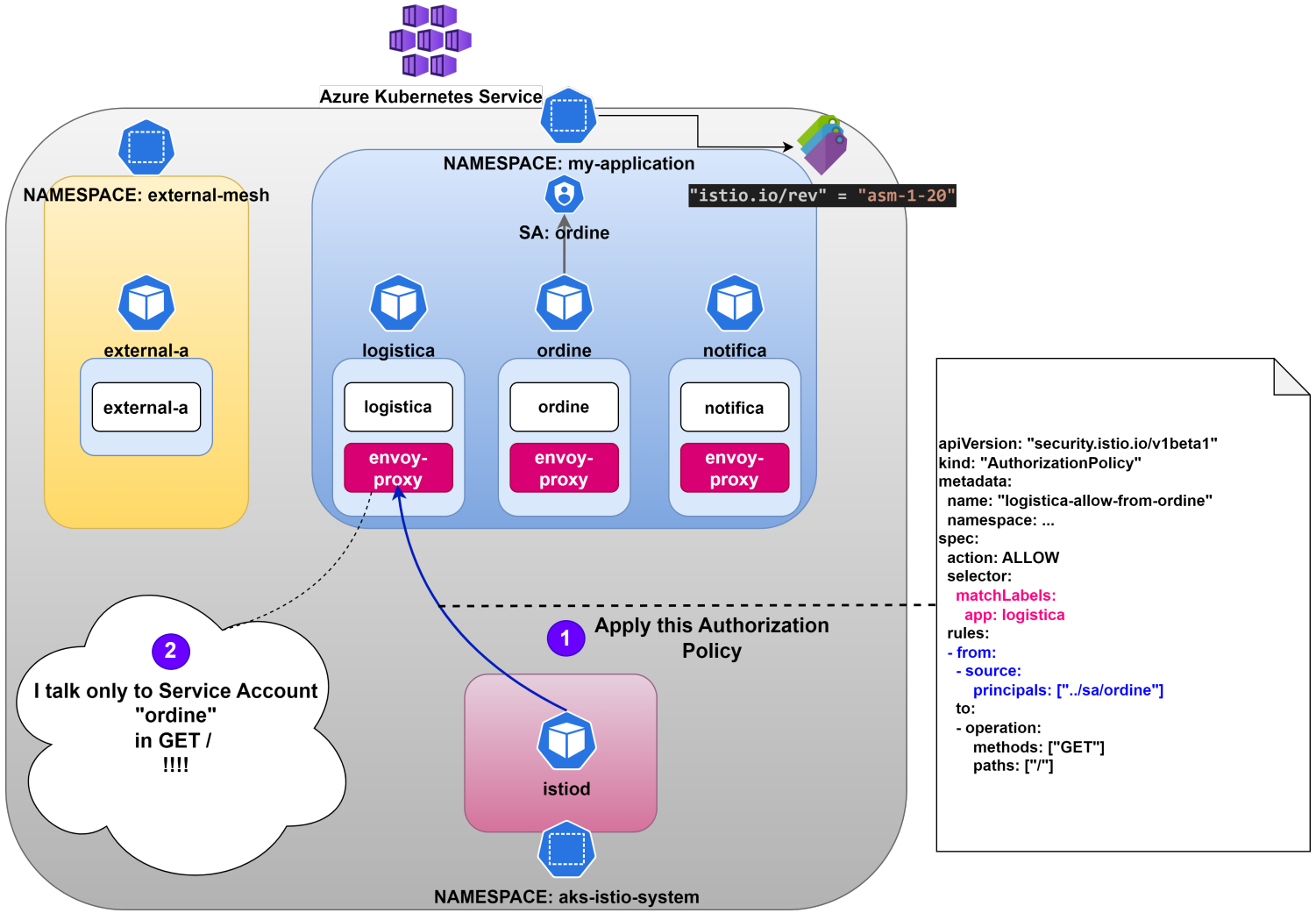

```
apiVersion: "security.istio.io/v1beta1"
kind: "AuthorizationPolicy"
metadata:
  name: "logistica-allow-from-ordine"
  namespace: ...
spec:
  action: ALLOW
  selector:
    matchLabels:
      app: logistica
  rules:
  - from:
    - source:
        principals: ["../sa/ordine"]
  to:
  - operation:
      methods: ["GET"]
      paths: ["/"]
```

© Copyright Kyndryl, Inc. 2024

kyndryl

AZURE DAY

Authorization Policy

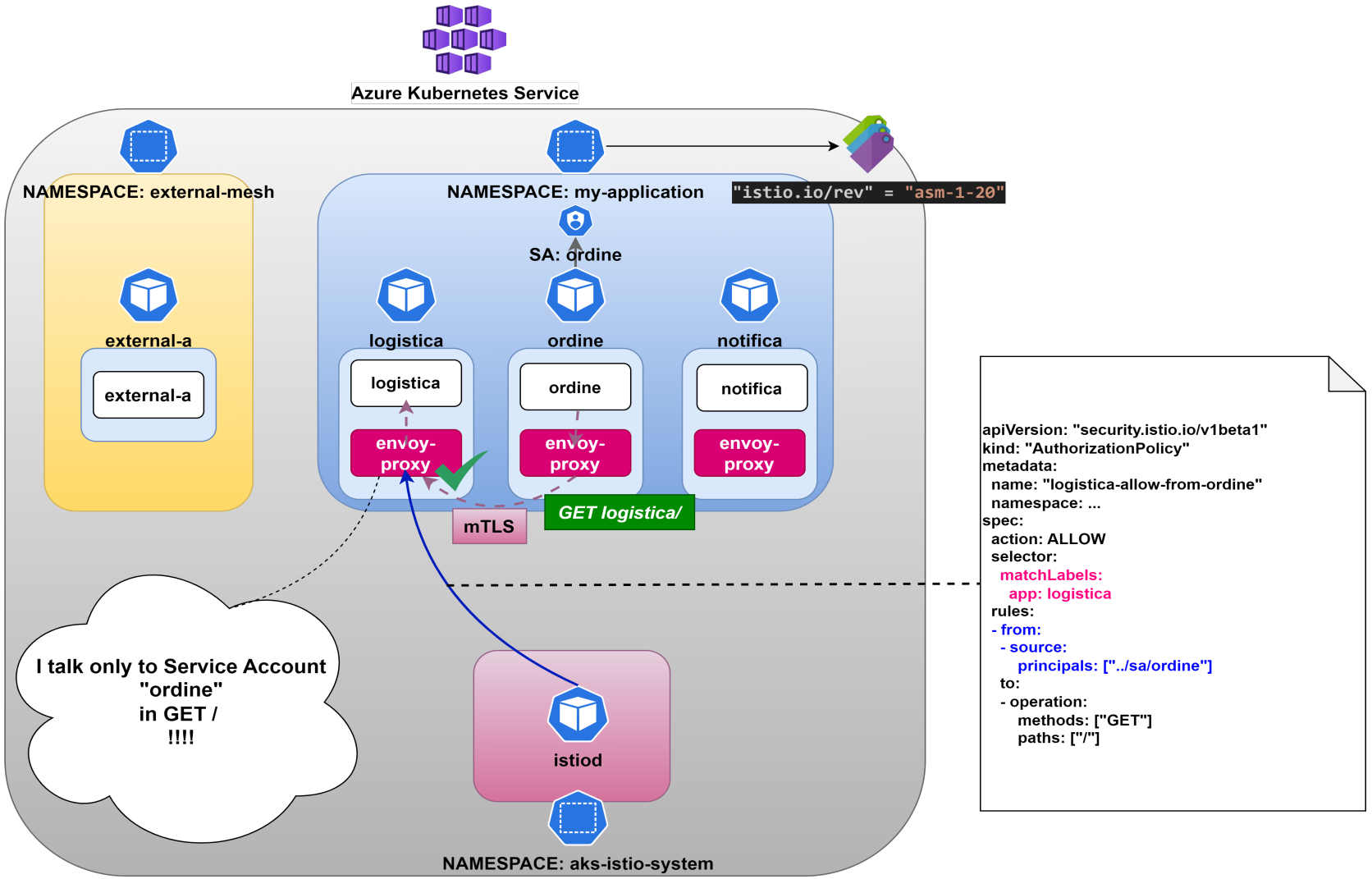



```
apiVersion: "security.istio.io/v1beta1"
kind: "AuthorizationPolicy"
metadata:
  name: "logistica-allow-from-ordine"
  namespace: ...
spec:
  action: ALLOW
  selector:
    matchLabels:
      app: logistica
  rules:
  - from:
    - source:
        principals: ["../sa/ordine"]
    to:
    - operation:
        methods: ["GET"]
        paths: ["/"]
```

© Copyright Kyndryl, Inc. 2024

kyndryl

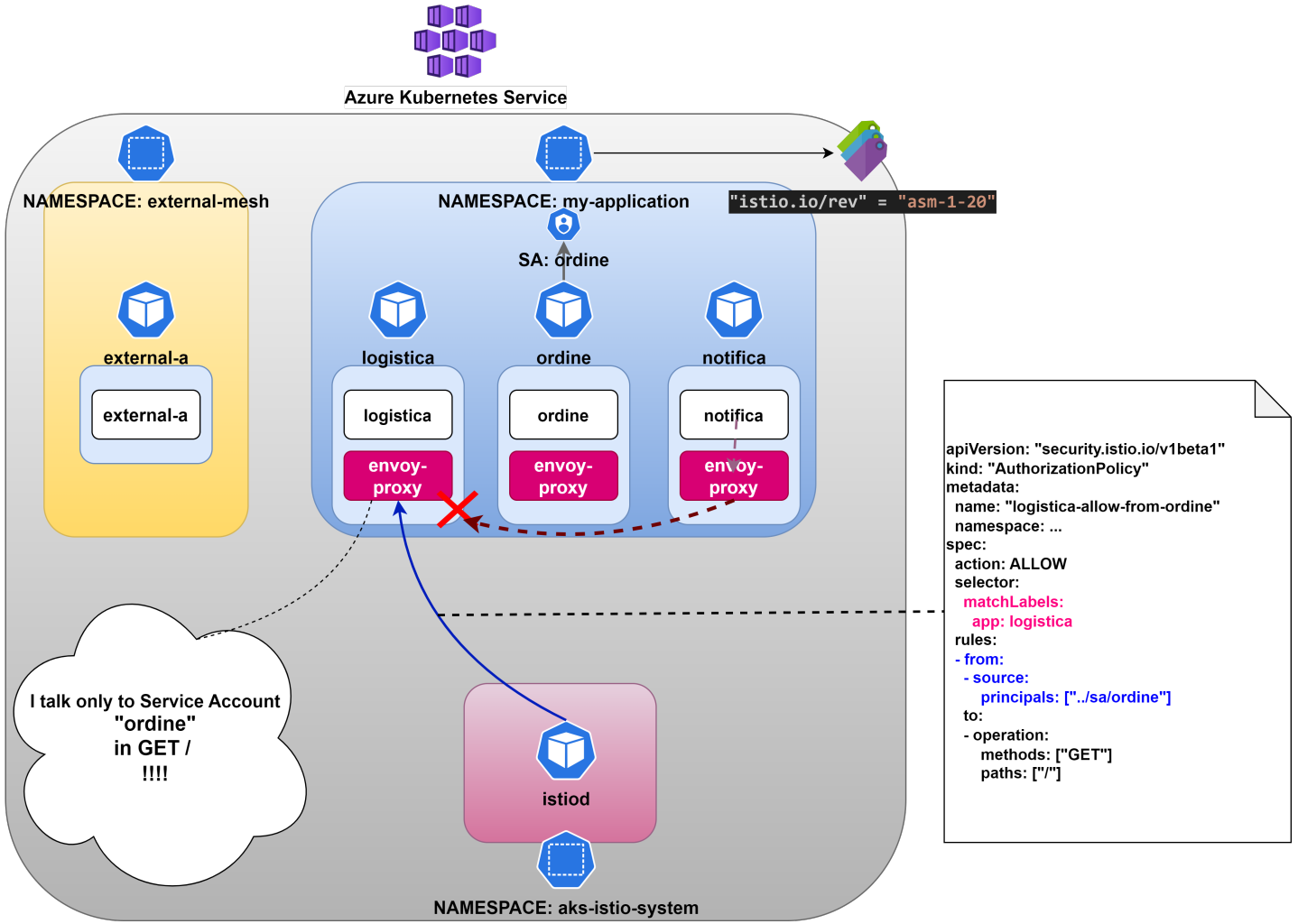
Authorization Policy Test: Allowed flow




```
apiVersion: "security.istio.io/v1beta1"
kind: "AuthorizationPolicy"
metadata:
  name: "logistica-allow-from-ordine"
  namespace: ...
spec:
  action: ALLOW
  selector:
    matchLabels:
      app: logistica
  rules:
  - from:
    - source:
        principals: ["../sa/ordine"]
    to:
    - operation:
        methods: ["GET"]
        paths: ["/"]
```

© Copyright Kyndryl, Inc. 2024

Authorization Policy Test: Denied flow




```
apiVersion: "security.istio.io/v1beta1"
kind: "AuthorizationPolicy"
metadata:
  name: "logistica-allow-from-ordine"
  namespace: ...
spec:
  action: ALLOW
  selector:
    matchLabels:
      app: logistica
  rules:
  - from:
    - source:
        principals: ["../sa/ordine"]
    to:
    - operation:
        methods: ["GET"]
        paths: ["/"]
```

© Copyright Kyndryl, Inc. 2024

AZURE DAY

Demo Session

© Copyright Kyndryl, Inc. 2024

kyndryl

AZURE DAY

Thanks to

Microsoft

avanade

eng

software one

4wardPRO
AN IMPRESOFT COMPANY

BC Soft
consulenza sistemi informativi

LOBRA

UNIKEY
Bringing IT knowledge to the people

Packt

JET BRAINS

NET CODE

Thank You

Vote our session