

The background is a dark blue gradient with various technical and abstract graphics. On the left, there is a large, semi-transparent Kubernetes logo (a ship's wheel inside a hexagon). Overlaid on the logo and the background are several circular gauges with numerical scales (e.g., 40, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, 260). To the right of the logo, there are faint, glowing lines and shapes that resemble a network diagram or server racks. The main title is written in a large, white, sans-serif font on the right side of the image.

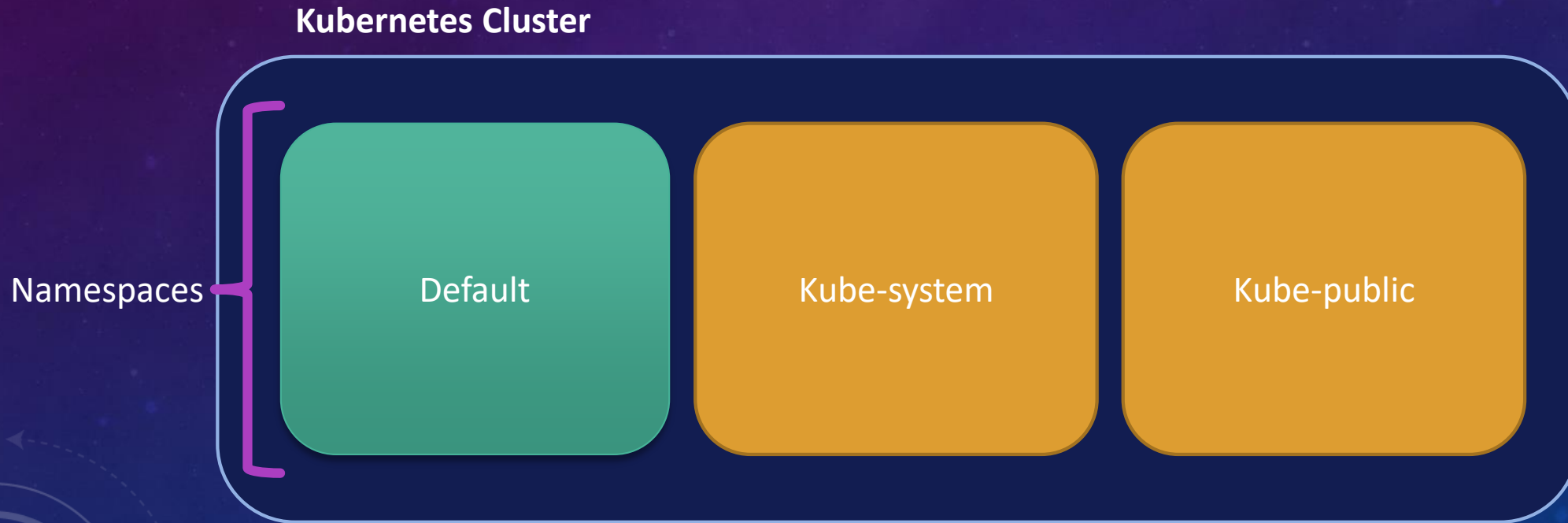
ROLES, USUARIOS Y SERVICE ACCOUNT KUBERNETES

Primeros pasos

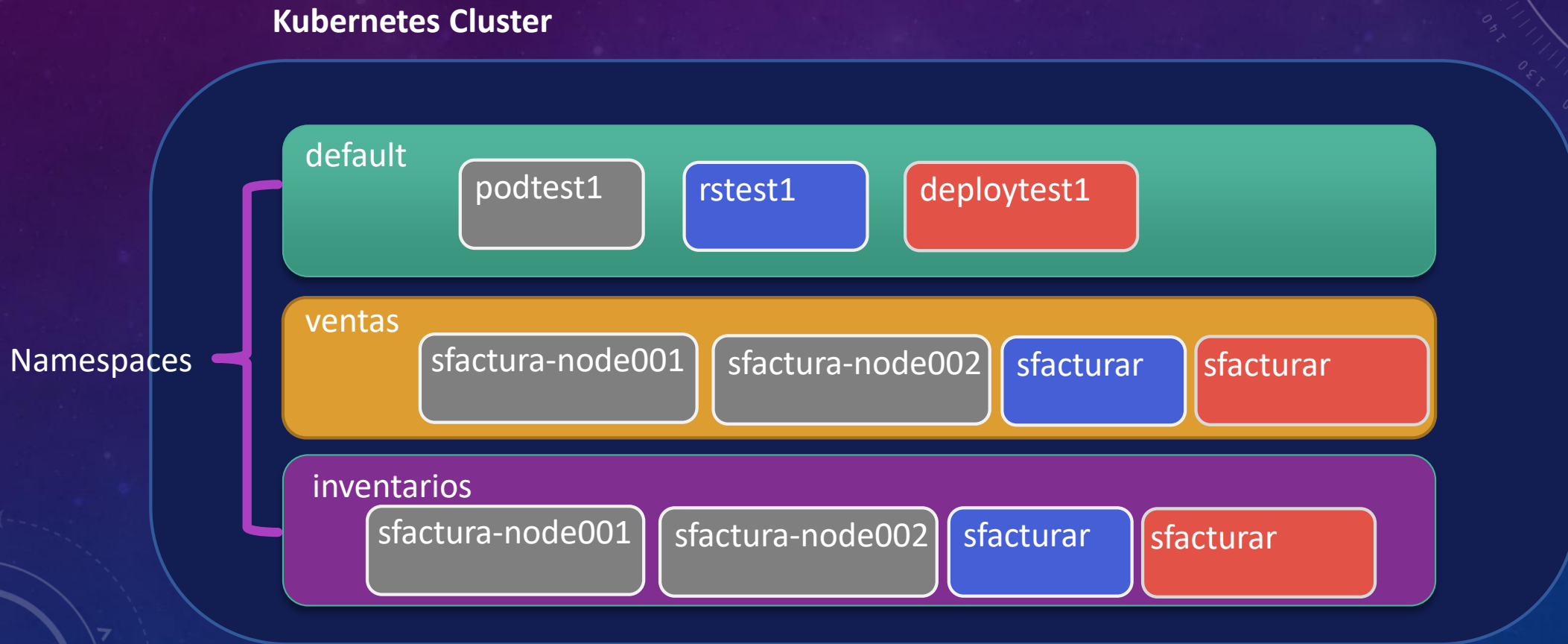
JHON EMMANUEL ZUÑIGA PAREDES
<https://www.linkedin.com/in/jhonezp/>

ENTENDIENDO LOS NAMESPACES

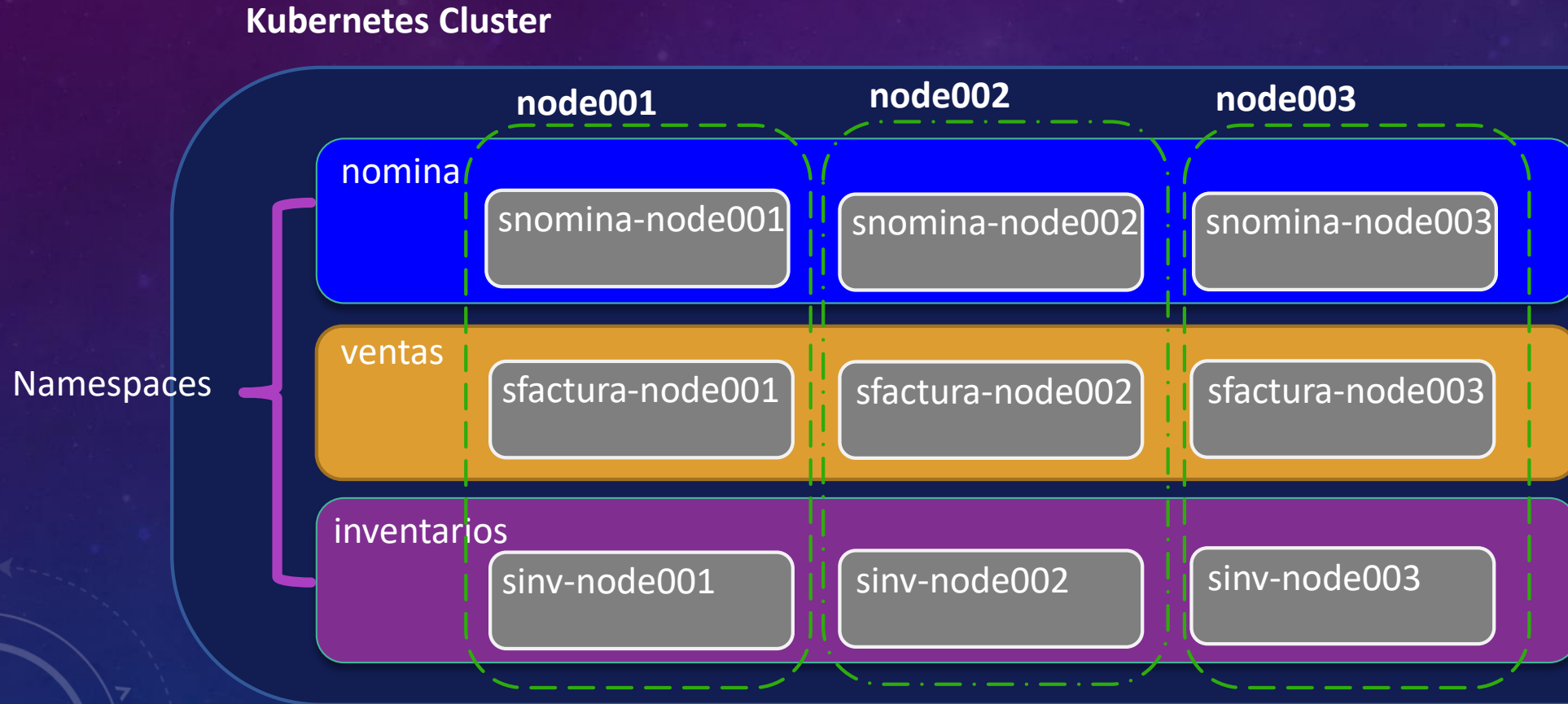
- Los namespaces permiten definir un Scope y hacer una separación lógica dentro de nuestro cluster de K8S.



ENTENDIENDO LOS NAMESPACES



ENTENDIENDO LOS NAMESPACES

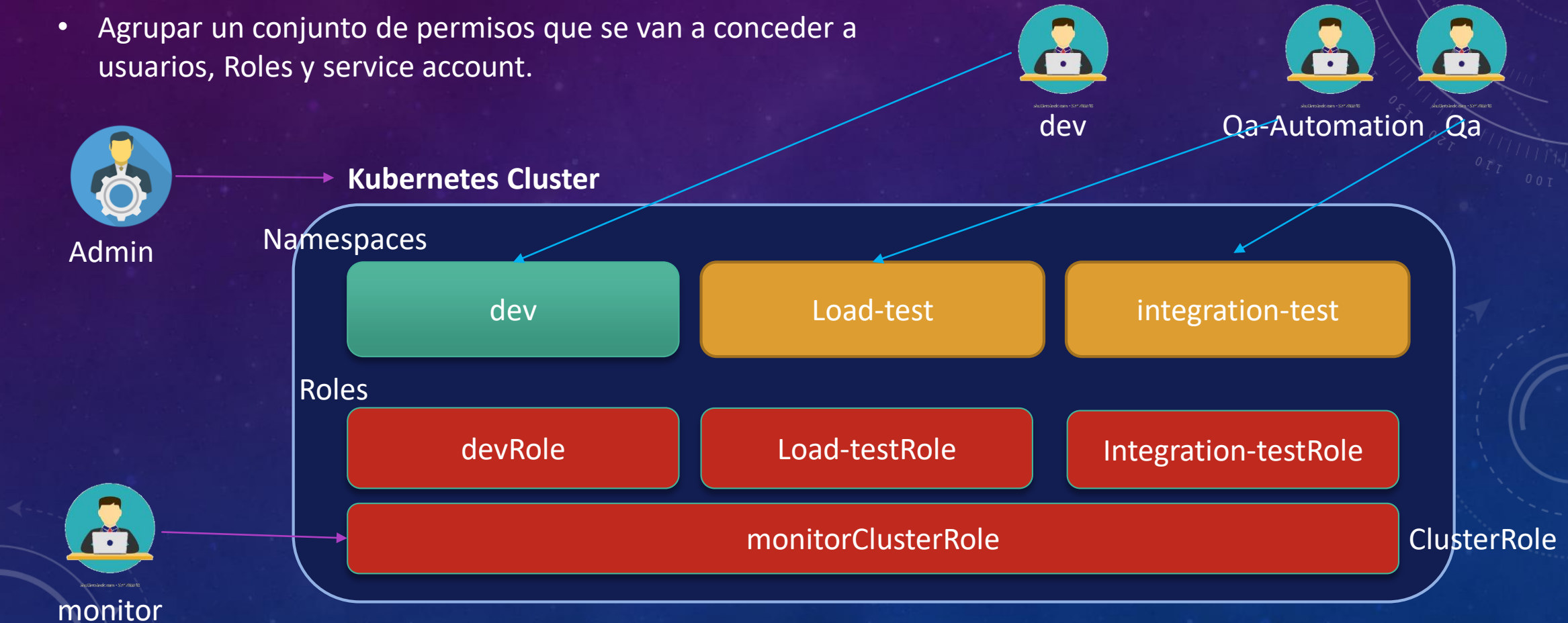




TALLER 1 NAMESPACES

RBAC (ROLE BASED ACCESS CONTROL)

- Agrupar un conjunto de permisos que se van a conceder a usuarios, Roles y service account.



RBAC (ROLE BASED ACCESS CONTROL)

RoleBinding

Scope: Tiene los permisos aplicados sobre el namespace específico.

Kubernetes Cluster

Namespaces

dev

Roles

devRole
namespace: dev
resouce: ["pod"]
verbs: ["Add, Create, get"]

RoleBinding

devRoleBinding
namespace: dev
-Kind: **User/group/sa**
name: dev
Roleref:
Kind: **Role**
name: **devRole**



dev

RBAC (ROLE BASED ACCESS CONTROL)

ClusterRoleBinding

Scope: Tiene los permisos aplicados sobre todos los Namespaces

Kubernetes Cluster

ClusterRoles

```
monitorClusterRole
namespace: dev
resource: ["pod"]
verbs: ["Add, Create, get"]
```

ClusterRoleBinding

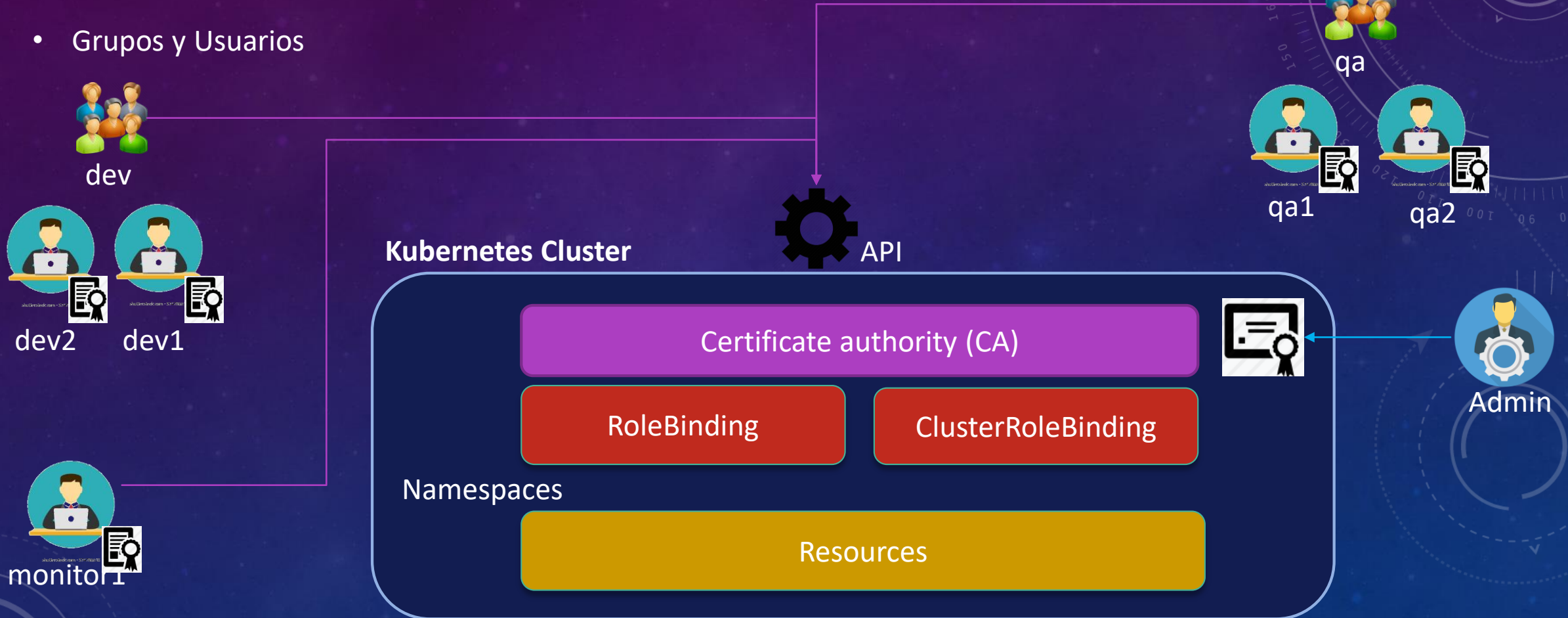
```
monitorClusterRoleBinding
namespace: dev
-Kind: User/group/sa
  name: monitor
Roleref:
  Kind: ClusterRole
  name: monitorClusterRole
```



monitor

RBAC (ROLE BASED ACCESS CONTROL)

- Grupos y Usuarios



RBAC (ROLE BASED ACCESS CONTROL)

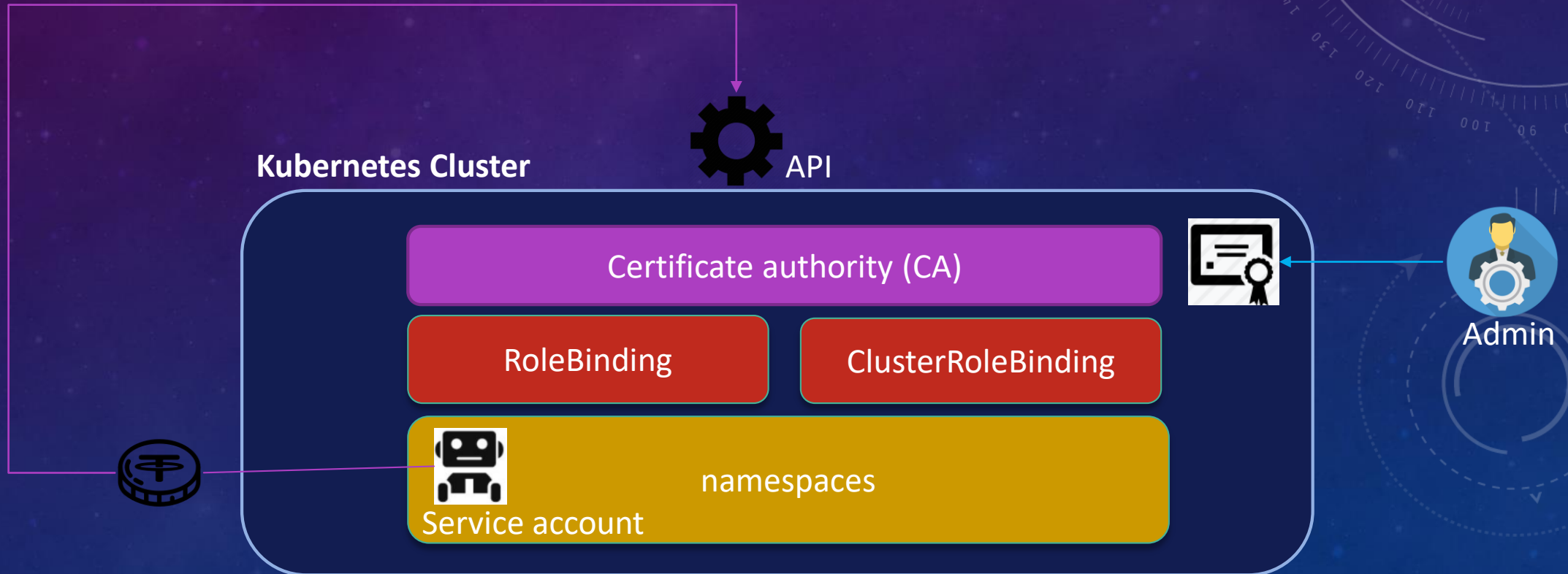
- Generando el certificado
- Del lado del Usuario (dev1)
 - Generar el key con Openssl
 - Generamos el csr (certificate signing request)
 - “/CN=usuario/O=dev” ej. “/CN=dev1/O=dev” ej. “/CN=qa/O=qa”
- Del lado del ADMIN del Cluster
 - Generar el crt del usuario dev1, utilizando el ca.crt del cluster.
- Del lado del Usuario (dev1)
 - Configurar el certificado en su ambiente de K8S en sus context.



TALLER 2
RBAC
USUARIOS
GRUPOS

RBAC (ROLE BASED ACCESS CONTROL)

- Service Account





TALLER 2
RBAC
SERVICE ACCOUNT

