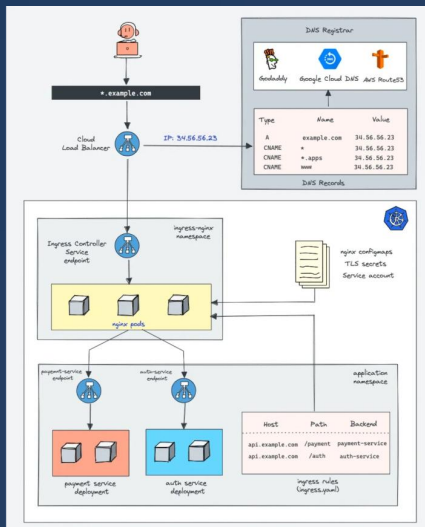


Ingress dans Kubernetes

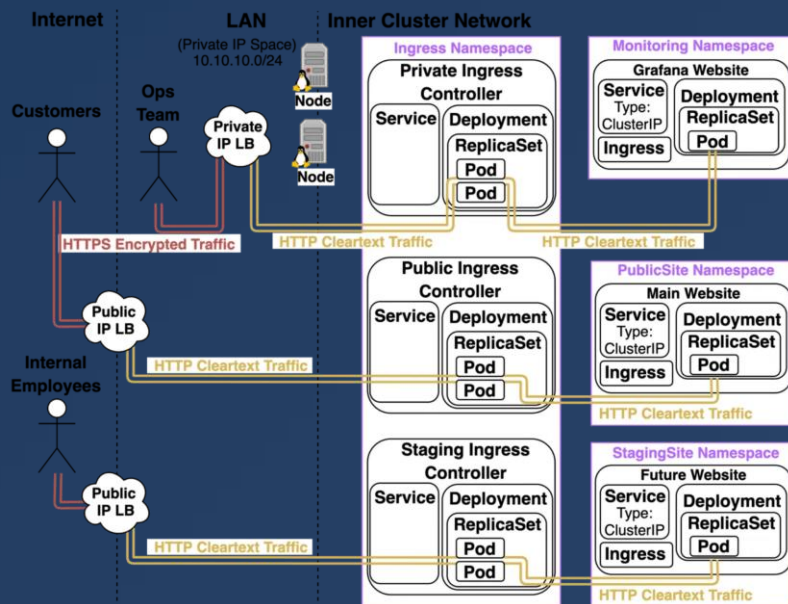
La porte d'entrée intelligente de votre cluster



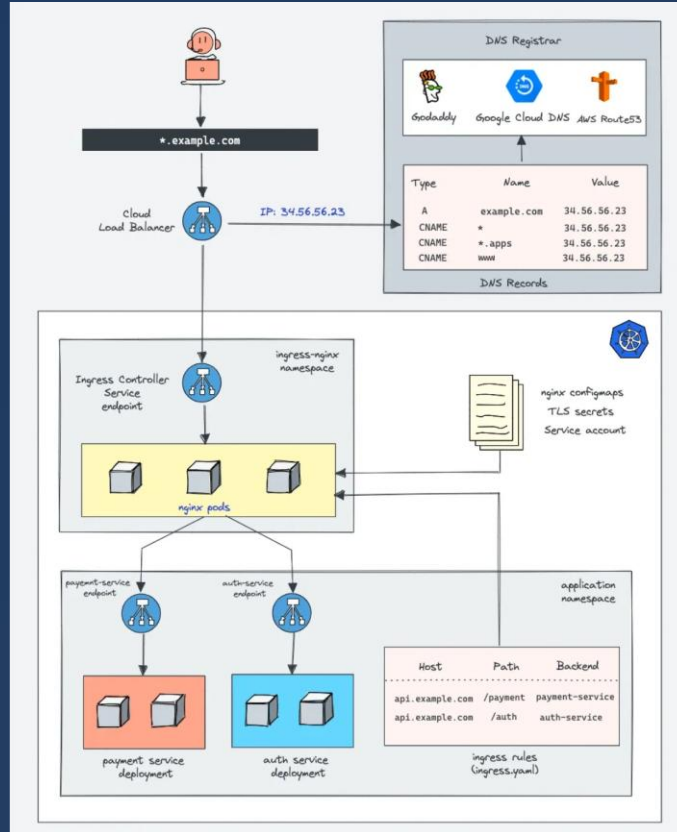
Qu'est-ce que l'Ingress ?

L'Ingress est un **contrôleur** qui fonctionne comme un **routeur HTTP/HTTPS intelligent**, exposant les services Kubernetes à l'extérieur du cluster.

Il agit comme une **porte d'entrée unique** pour le trafic externe, permettant d'accéder aux différents services de votre cluster.



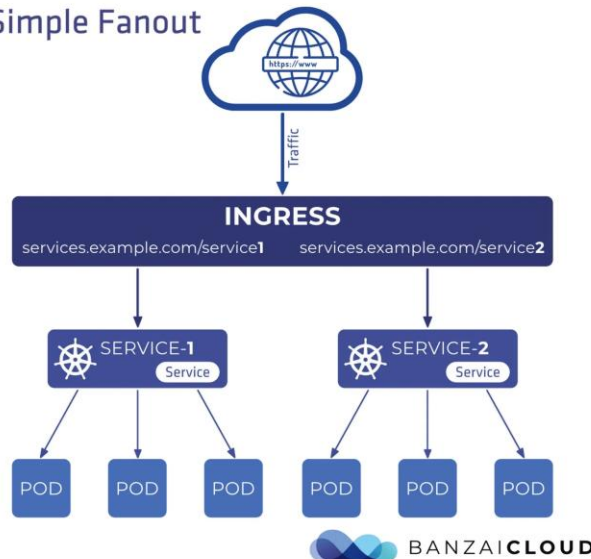
Ingress dans Kubernetes



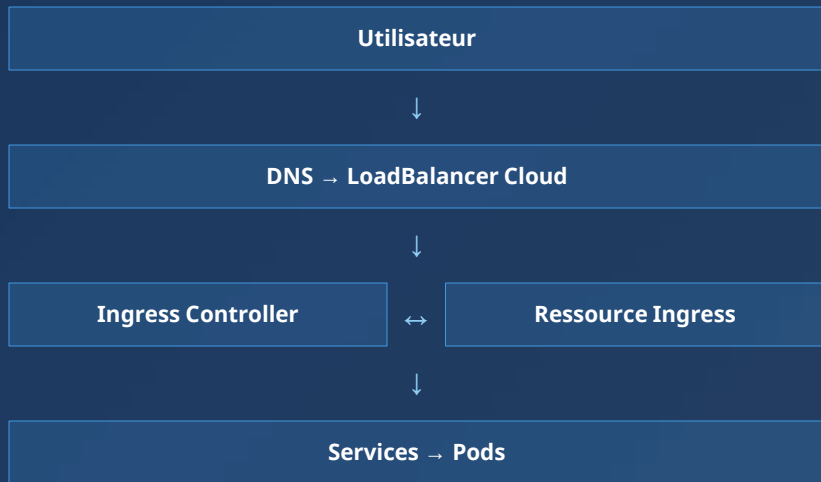
Pourquoi utiliser l'Ingress ?

Sans Ingress	Avec Ingress
Chaque service a son propre LoadBalancer/NodePort	Un point d'entrée unique pour toutes les applications
Gestion complexe des certificats SSL	Terminaison SSL centralisée
Pas de routage basé sur le host/path	Routage intelligent par domaine et chemin

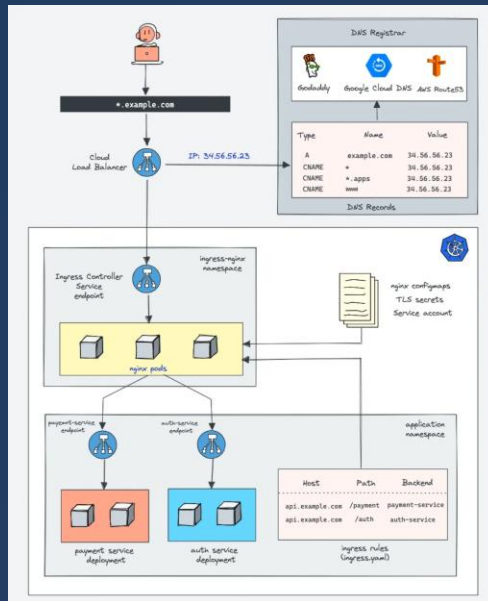
Simple Fanout



Vue d'ensemble des composants



L'architecture Ingress permet de **router intelligemment** le trafic externe vers les services internes de votre cluster.



Composants Clés

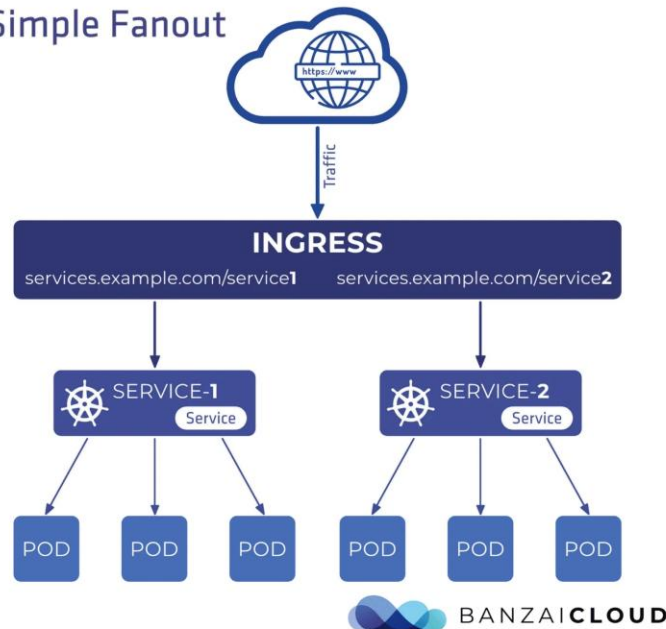
1. Ingress Controller

- **Implémentation** : Nginx, Traefik, HAProxy, Istio, etc.
- **Rôle** : Pod qui fait le vrai travail de routage
- **À déployer** dans votre cluster

2. Ressource Ingress

- **Rôle** : Règles de routage déclaratives (YAML)
- **Définit** : Quel trafic va vers quel service

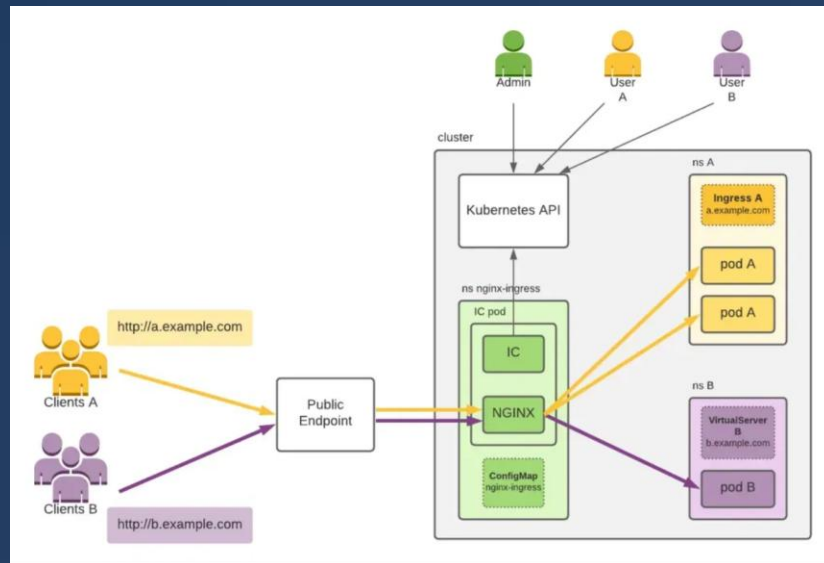
Simple Fanout



Ingress Controller - Le Cerveau

Définition : C'est le **processus actif** qui implémente les règles Ingress. C'est un **Pod** qui tourne dans votre cluster.

Controller	Use Case
Nginx Ingress	Usage général, stable
Traefik	Microservices, configuration automatique
HAProxy	Haute performance, charge élevée
Istio Gateway	Service Mesh avancé, architectures complexes



Fonctionnalités de Routage

Routage basé sur :

🌐 Host (domaine) :

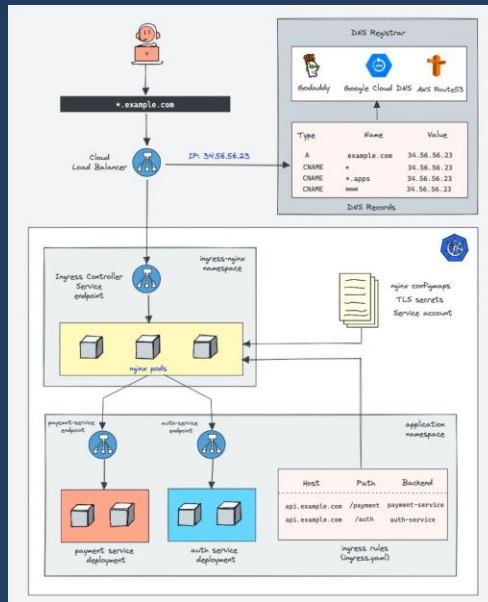
blog.mon-site.com vs api.mon-site.com

📍 Path (chemin) :






/api vs /web vs /static

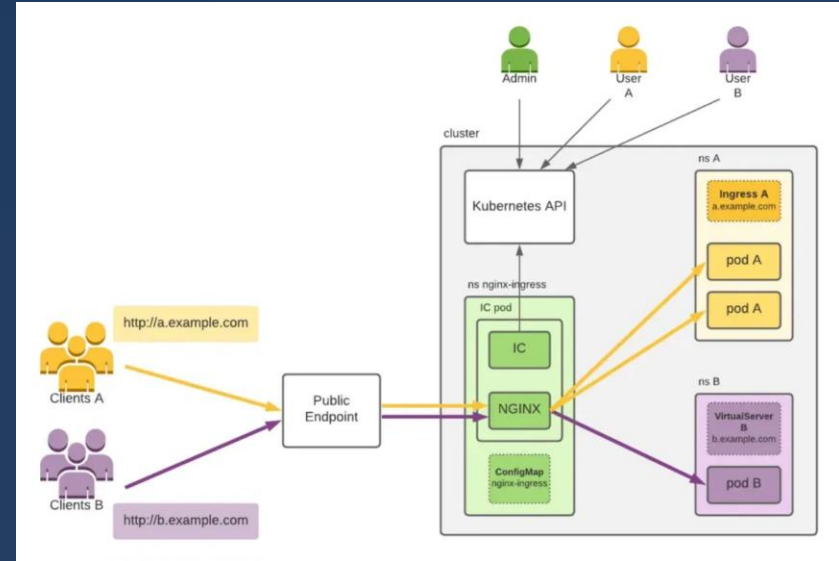
🔌 Port :

HTTP (80) vs HTTPS (443)



Autres Fonctionnalités

-  **Terminaison SSL/TLS:** Gestion centralisée des certificats
-  **Load balancing:** Distribution intelligente du trafic
-  **Réécriture d'URL:** Modification des chemins d'accès
-  **Authentication:** Contrôle d'accès aux services
-  **Rate limiting:** Protection contre les surcharges



Exemple de Ressource Ingress

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: my-app-ingress
  annotations:
    nginx.ingress.kubernetes.io/rewrite-target: /
spec:
  tls:
    - hosts:
        - mon-site.com
      secretName: mon-certificat-tls
  rules:
    - host: mon-site.com
      http:
        paths:
          - path: /web
            pathType: Prefix
            backend:
              service:
                name: frontend-service
                port:
                  number: 80
          - path: /api
            pathType: Prefix
            backend:
              service:
                name: backend-service
                port:
                  number: 8080
```

Éléments clés :

- **rules** : Définit les règles de routage basées sur le domaine et le chemin
- **tls** : Configure la terminaison SSL/TLS
- **annotations** : Personnalise le comportement de l'Ingress Controller

Analogie Simple

L'Ingress est comme le réceptionniste d'un grand hôtel :



Point d'entrée unique pour tous les visiteurs, comme l'entrée principale d'un hôtel



Dirige chaque personne vers la bonne chambre (service) selon sa réservation



Vérifie les identités (SSL/auth) avant d'autoriser l'accès



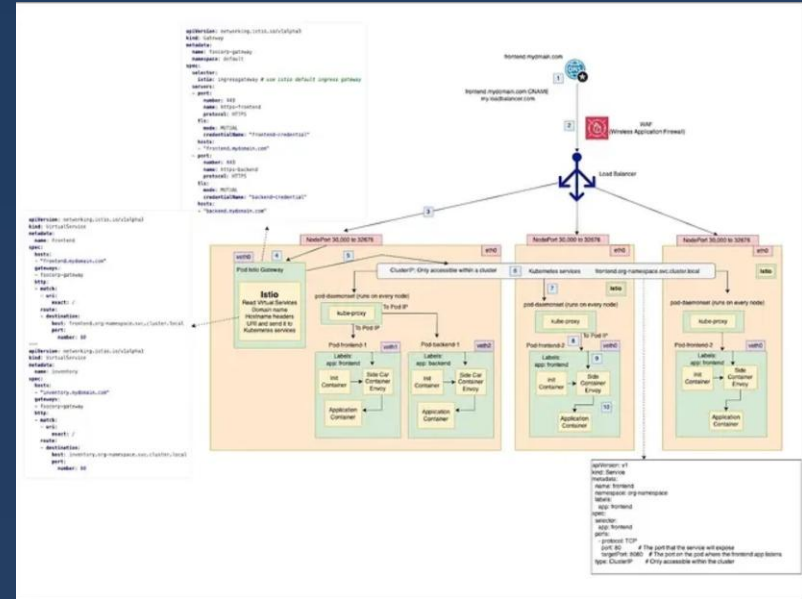
Donne des instructions précises pour trouver les services (réécriture d'URL)



Protège l'accès aux zones réservées (authentification et autorisation)

Workflow Typique

- 1 **Utilisateur** → <https://mon-site.com/api>
- 2 **DNS** → Adresse IP de l'Ingress Controller
- 3 **Ingress Controller** → Consulte les règles Ingress
- 4 **Routage** → Envoie vers le service backend-service:8080
- 5 **Service** → Redirige vers un Pod backend



Avantages Clés

Économie



Un seul LoadBalancer cloud au lieu d'un par service

Centralisation



Gestion unique du trafic entrant

Flexibilité



Routing complexe facilement configurable

Sécurité

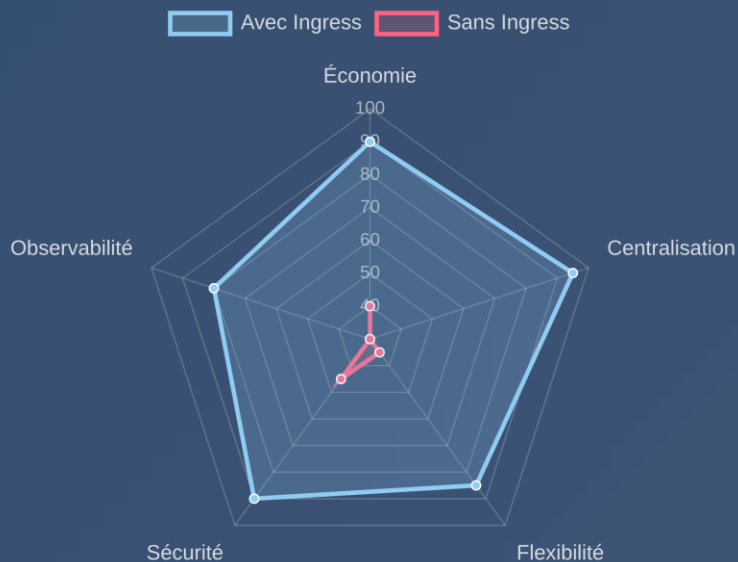


SSL/TLS et authentification centralisés

Observabilité



Métriques et logs centralisés



Conclusion

L'Ingress est la **porte d'entrée intelligente** de votre cluster Kubernetes !

- 🏦 **Économie** : Un seul LoadBalancer cloud
- 📡 **Centralisation** : Gestion unique du trafic
- 🔧 **Flexibilité** : Routage complexe facile
- 🔒 **Sécurité** : SSL/TLS centralisé
- 📊 **Observabilité** : Métriques et logs centralisés

