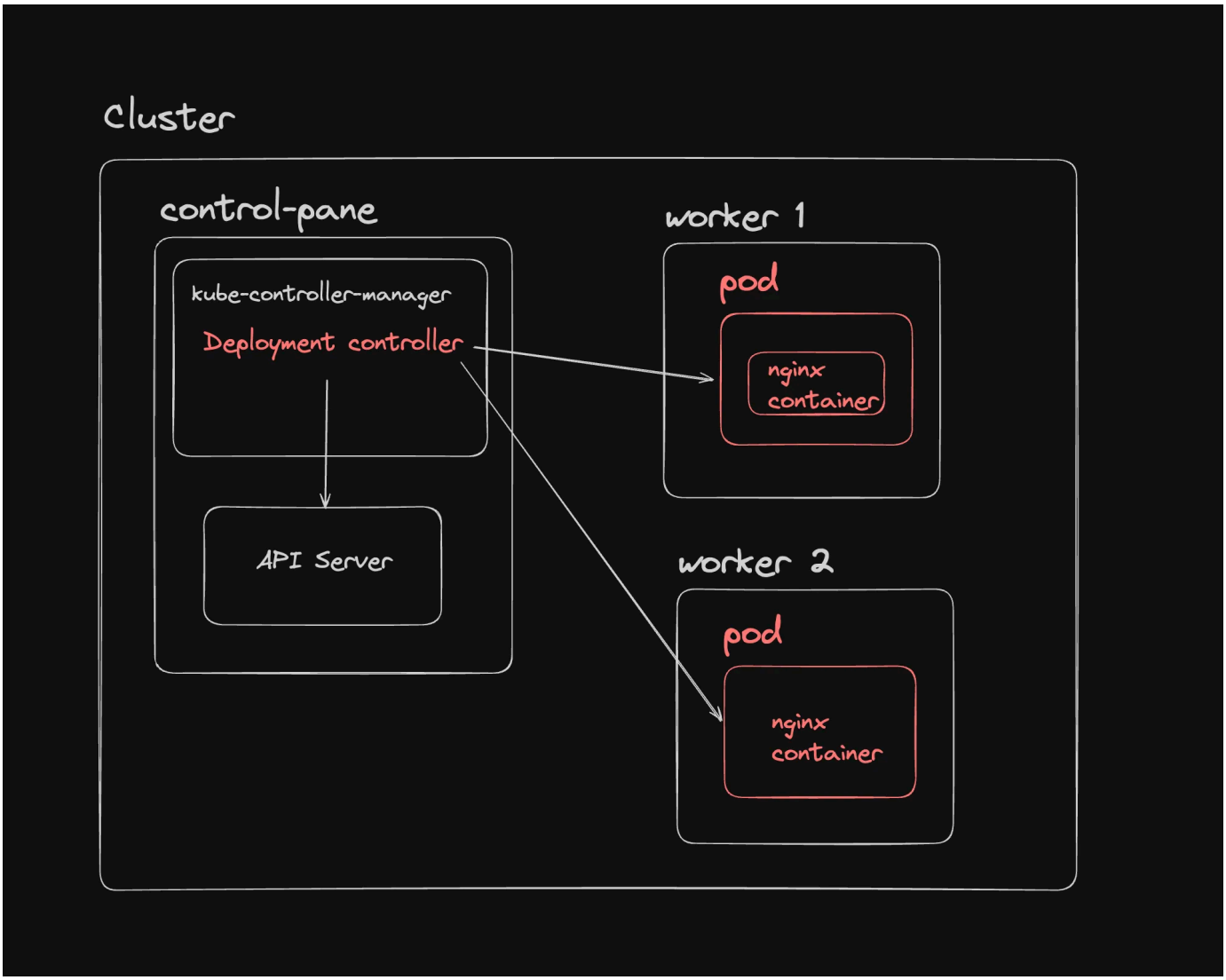
Deployment

A **Deployment** in Kubernetes is a higher-level abstraction that manages a set of Pods and provides declarative updates to them. It offers features like scaling, rolling updates, and rollback capabilities, making it easier to manage the lifecycle of applications.



- **Pod**: A Pod is the smallest and simplest Kubernetes object. It represents a single instance of a running process in your cluster, typically containing

- **Deployment**: A Deployment is a higher-level controller that manages a (sures the desired number of Pods are running and provides declarative updates to the Pods it manages.

Key Differences Between Deployment and Pod:

1. **Abstraction Level**:
    - **Pod**: A Pod is the smallest and simplest Kubernetes object. It represents a single instance of a running process in your cluster, typically containing one or more containers.
    - **Deployment**: A Deployment is a higher-level controller that manages a set of identical Pods. It ensures the desired number of Pods are running and provides declarative updates to the Pods it manages.
2. **Management**:
    - **Pod**: They are ephemeral, meaning they can be created and destroyed frequently.
    - **Deployment**: Deployments manage Pods by ensuring the specified number of replicas are running at any given time. If a Pod fails, the Deployment controller replaces it automatically.
3. **Updates**:
    - **Pod**: Directly updating a Pod requires manual intervention and can lead to downtime.
    - **Deployment**: Supports rolling updates, allowing you to update the Pod template (e.g., new container image) and roll out changes gradually. If something goes wrong, you can roll back to a previous version.
4. **Scaling**:
    - **Pod**: Scaling Pods manually involves creating or deleting individual Pods.

- **Deployment**: Allows easy scaling by specifying the desired . The Deployment controller adjusts the number of Pods automatically.

5. **Self-Healing**:

    - **Pod**: If a Pod crashes, it needs to be restarted manually unless managed by a higher-level controller like a Deployment.
    - **Deployment**: Automatically replaces failed Pods, ensuring the desired state is maintained.