ghp_O0J2IZkoIWDqciiMIWInx8uHTr1QVn3q5q84

9o893Jx25UOyoI26 argo

```
  ssh-keygen -t ed25519 -C "minnu.antony05@gmail.com"
  eval "$(ssh-agent -s)"
  ssh-add ~/.ssh/id_ed25519
  cat ~/.ssh/id_ed25519.pub
  ssh -T git@github.com '
```

```
3.87.182.69
ssh -i "minnunv.pem" ubuntu@ec2-3-87-182-69.compute-1.amazonaws.com
```

```
 sudo usermod -aG docker $USER
 newgrp docker
```

```
FROM python:3.11-slim
WORKDIR /app
COPY requirements.txt .
RUN pip install --no-cache-dir -r requirements.txt \
    && python -m compileall . \
    && addgroup --system appgroup \
    && adduser --system --ingroup appgroup appuser \
    && chown -R appuser:appgroup /app
EXPOSE 5000
COPY . .
USER appuser
CMD ["python", "-m", "app.main"]
```

https://chatgpt.com/share/68b2b4a2-9310-800d-a689-dadfc4328eda
 - reverse proxy thing

.env:

```
MYSQL_ROOT_PASSWORD=rootpassword
MYSQL_DATABASE=micro_db
MYSQL_USER=micro_user
MYSQL_PASSWORD=secret
 cat mysql-secret.yaml
apiVersion: v1
kind: Secret
metadata:
  name: mysql-secret
type: Opaque
stringData:
  MYSQL_ROOT_PASSWORD: rootpassword    # <-- change if needed
  MYSQL_DATABASE: micro_db             # <-- change if needed
  MYSQL_USER: micro_user             # <-- change if needed
  MYSQL_PASSWORD: secret             # <-- change if needed
```

52.87.239.57

```
ssh -i "minnunv.pem" ubuntu@ec2-52-87-239-57.compute-1.amazonaws.com
```

—-------------------------------------------------sep2—---------------------

**Option 1 (Most Basic / Only DB Protection)**
👉 Just apply the MySQL policy. Everything else stays open, but DB is safe.

```
git clone https://github.com/exp-tracker-org/frontend-new

git clone https://github.com/exp-tracker-org/expense-service

git clone https://github.com/exp-tracker-org/user-service

git clone https://github.com/exp-tracker-org/infra
```

```
# Download the latest version
curl -sSL -o argocd
https://github.com/argoproj/argo-cd/releases/latest/download/argocd-linux-amd64

# Make it executable
chmod +x argocd

# Move it to a directory in your PATH, e.g.,
sudo mv argocd /usr/local/bin/
```

```
argocd login 18.208.109.186:32673 \
  --username admin \
  --password $(kubectl -n argocd get secret argocd-initial-admin-secret -o
jsonpath="{.data.password}" | base64 -d) \
  --insecure
```

```
argocd repo add https://github.com/exp-tracker-org/infra.git --username minnuantony --password
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

argocd repo add https://github.com/exp-tracker-org/frontend-new.git --username minnuantony
--password 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

argocd repo add https://github.com/exp-tracker-org/user-service.git --username minnuantony
--password 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

argocd repo add https://github.com/exp-tracker-org/expense-service.git --username minnuantony
--password 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
```

```
kubectl create secret generic github-org-secrets \
  --from-literal=MYSQL_ROOT_PASSWORD=rootpassword \
  --from-literal=MYSQL_DATABASE=micro_db \
  --from-literal=MYSQL_USER=micro_user \
  --from-literal=MYSQL_PASSWORD=secret \
  -n default
```

—-----------------------------------------------

```
rgocd app list
  275  argocd app get root-app
  276  argocd app list
  277  argocd app get root-app --refresh
  278  argocd app get infra-app --refresh
  279  argocd app get frontend-new --refresh
  280  argocd app get user-service --refresh
  281  argocd app get expense-service --refresh
  282  argocd app sync root-app
  283  argocd app sync infra-app
  284  argocd app sync frontend-new
  285  argocd app sync user-service
  286  argocd app sync expense-service
  287  argocd app get root-app
  288  argocd app get infra-app
  289  argocd app delete root-app --cascade=false   # optional, to clean up old invalid state
  290  argocd app create -f argo-apps/root-app.yaml
  291  argocd app get root-app --refresh
  292  argocd app sync root-app
  293  argocd app create root-app   --repo https://github.com/exp-tracker-org/infra.git   --path argo-apps   --dest-server https://kubernetes.default.svc   --dest-namespace argocd   --sync-policy automated   --auto-prune   --self-heal
  294  argocd app get root-app
  295  argocd app list
  296  argocd app get root-app --refresh
  297  argocd app list
  298  argocd app get root-app --refresh
  299  argocd app sync infra-app
  300  argocd app operations list root-app
  301  argocd app get infra-app
  302  kubectl apply -f https://github.com/fluxcd/kustomize-controller/releases/latest/download/kustomization-crd.yaml
  303  argocd app get infra-app
  304  clear
  305  kubectl logs -n default deploy/expense-service
  306  kubectl logs -n default deploy/user-service
  307  kubectl describe pod user-service-559749db96-ckdr8 -n default
  308  kubectl get secret github-org-secrets -n default
  309  kubectl create secret generic github-org-secrets
--from-literal=MYSQL_ROOT_PASSWORD=rootpassword
--from-literal=MYSQL_DATABASE=micro_db   --from-literal=MYSQL_USER=micro_user
--from-literal=MYSQL_PASSWORD=secret   -n default
  310  kubectl get secret github-org-secrets -n default
  311  kubectl describe secret github-org-secrets -n default
  312  argocd app sync infra
  313  argocd login 18.208.109.186:32673   --username admin   --password $(kubectl -n argocd
get secret argocd-initial-admin-secret -o jsonpath="{.data.password}" | base64 -d)   --insecure
  314  argocd app sync infra
  315  argocd app list
  316  argocd app sync infra-app
  317  kubectl get pods -n default
  318  kubectl describe po mysql-db-0
  319  kubectl logs mysql-db-0 -n default
  320  kubectl get secret -n default
  321  kubectl get jobs -n default
  322  argocd app sync infra-app
  323  kubectl get secret mysql-secret -n default
  324  kubectl get pods -n default
  325  kubectl get all
  326  kubectl get svc -n ingress-nginx
  327  argocd app list
  328  kubectl get jobs -n default
  329  argocd app get infra-app
  330  histroy
  331  history
```

—-----------------------------------------------------------

```
apiVersion: v1
kind: Secret
metadata:
  name: mysql-secret
type: Opaque
stringData:
  MYSQL_ROOT_PASSWORD: rootpassword    # <-- change if needed
  MYSQL_DATABASE: micro_db             # <-- change if needed
  MYSQL_USER: micro_user               # <-- change if needed
  MYSQL_PASSWORD: secret               # <-- change if needed
```