

docker-desktop-monitoring-hello-world

A way to test out end to end Docker Desktop monitoring all in one spot

Installation

Step 1: Install the extension

```
docker extension install -f grafana/docker-desktop-extension:0.0.3
```

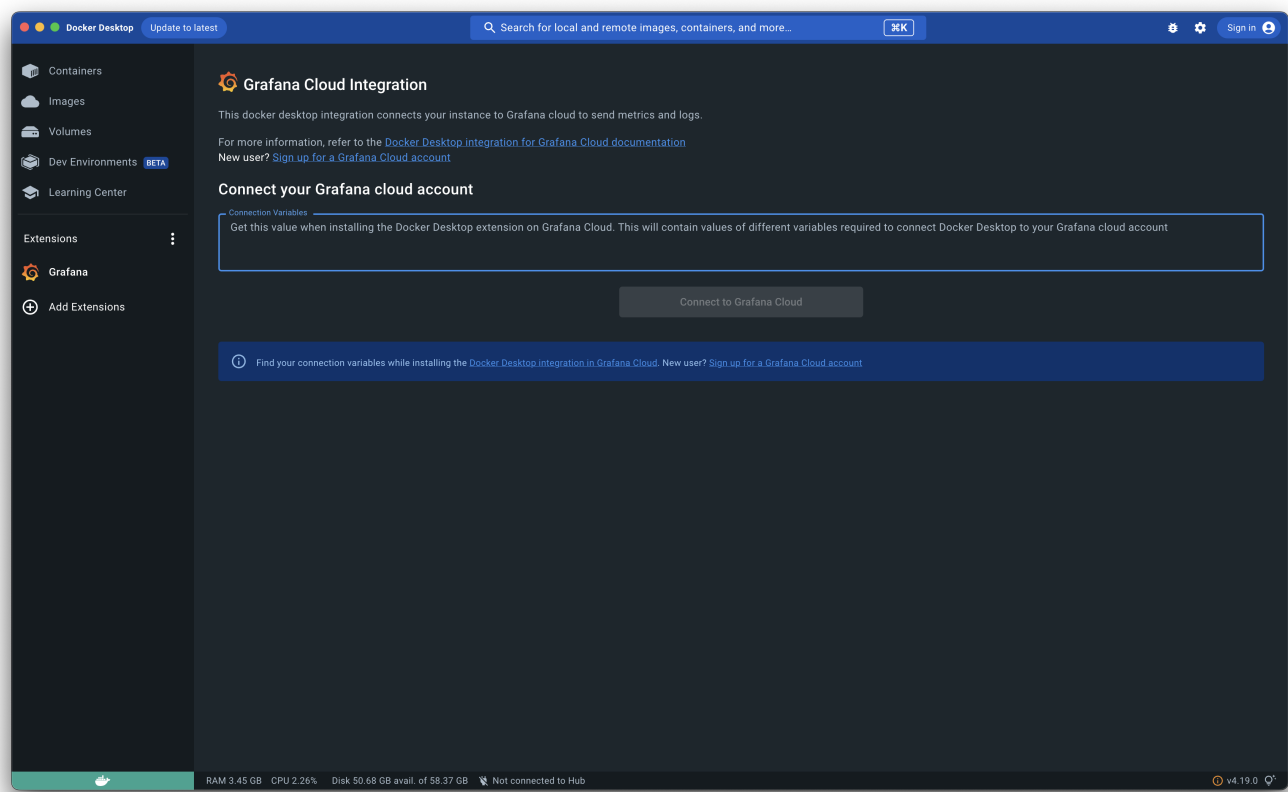
Step 2: Install Grafana, Mimir, and Loki

```
docker compose up
```

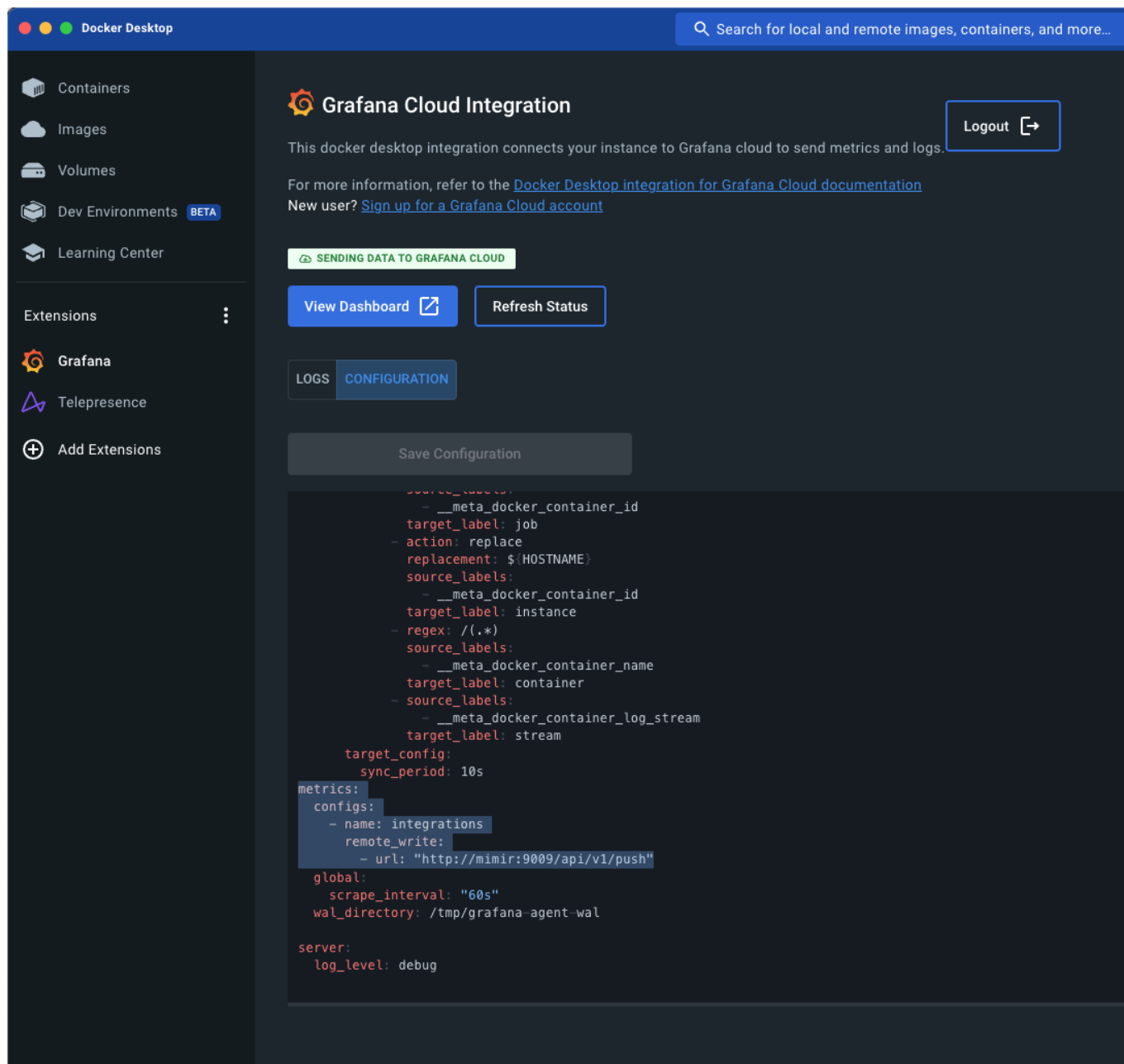
Step 3: Enter connection variables as shown below

Paste these

```
CLOUD_URL="http://localhost:3000"  
CLOUD_HOSTED_METRICS_URL="http://mimir:9009/api/v1/push"  
CLOUD_HOSTED_METRICS_ID="1" CLOUD_SCRAPE_INTERVAL="60s"  
CLOUD_HOSTED_LOGS_URL="http://loki:3100/loki/api/v1/push"  
CLOUD_HOSTED_LOGS_ID="1" CLOUD_RW_API_KEY=""
```



Step 4: Open the extensions configuration file



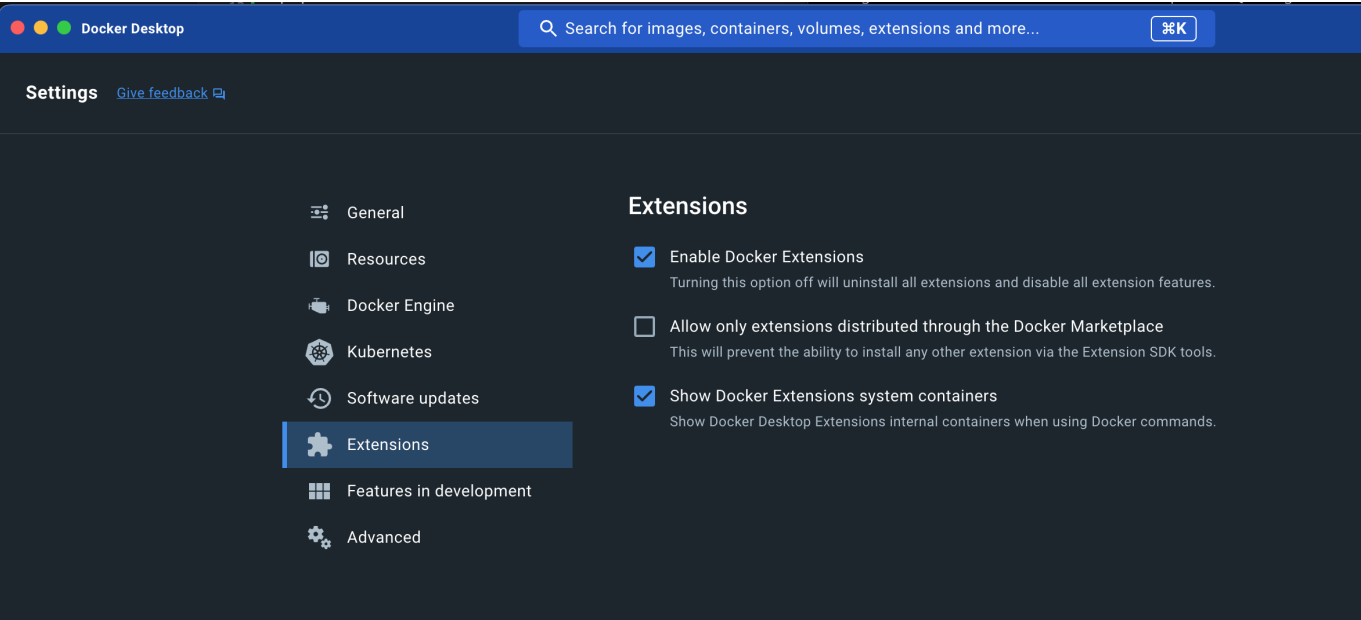
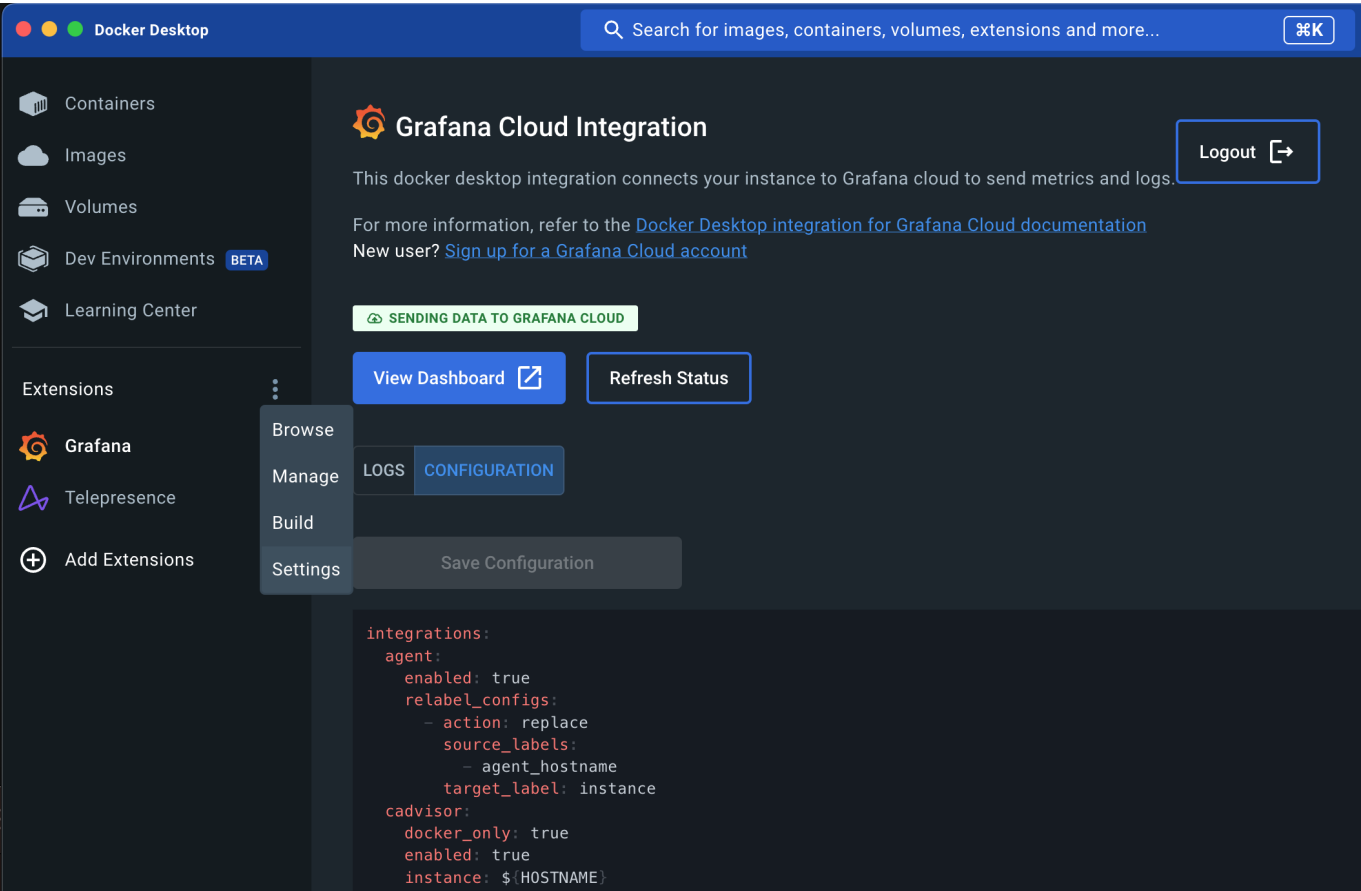
Step 5: Edit the extensions configuration file to look like the following and save

```
integrations:
  agent:
    enabled: true
    relabel_configs:
      - action: replace
        source_labels:
          - agent_hostname
        target_label: instance
  cadvisor:
    docker_only: true
    enabled: true
    instance: ${HOSTNAME}
    relabel_configs:
      - action: replace
        replacement: integrations/docker-desktop
```

```
    target_label: job
node_exporter:
  enabled: true
  relabel_configs:
    - replacement: ${HOSTNAME}
      target_label: instance
prometheus_remote_write:
  - url: "http://mimir:9009/api/v1/push"
logs:
  configs:
    - clients:
        - url: "http://loki:3100/loki/api/v1/push"
      name: integrations
      positions:
        filename: /tmp/positions.yaml
      scrape_configs:
        - docker_sd_configs:
            - host: unix:///var/run/docker.sock
              refresh_interval: 5s
          job_name: integrations/docker-desktop
          relabel_configs:
            - action: replace
              replacement: integrations/docker-desktop
              source_labels:
                - __meta_docker_container_id
              target_label: job
            - action: replace
              replacement: ${HOSTNAME}
              source_labels:
                - __meta_docker_container_id
              target_label: instance
            - regex: /(.* )
              source_labels:
                - __meta_docker_container_name
              target_label: container
            - source_labels:
                - __meta_docker_container_log_stream
              target_label: stream
          target_config:
            sync_period: 10s
metrics:
  configs:
    - name: integrations
      remote_write:
        - url: "http://mimir:9009/api/v1/push"
  global:
    scrape_interval: "60s"
wal_directory: /tmp/grafana-agent-wal

server:
  log_level: debug
```

Step 6: Enable "Show Docker Extensions systems containers"



Step 7: Get the extension containers IDs

```
docker ps
```

Step 8: Add the extension containers to the grafanet network

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
029032ad48a0	grafana/mimir:latest	"/bin/mimir --config..."	6 hours ago	Up 6 hours	8080/tcp, 0.0.0.0:9009->9009/tcp	mimir
a5012b0261e8	grafana/grafana	"/run.sh"	6 hours ago	Up 6 hours	0.0.0.0:3000->3000/tcp	grafana
d187d2c06b63	grafana/promtail:2.8.0	"/usr/bin/promtail -..."	4 days ago	Up 6 hours		promtail
0bce1dbfe0f7	grafana/loki:2.8.0	"/usr/bin/loki -conf..."	4 days ago	Up 6 hours	0.0.0.0:3100->3100/tcp	loki
01779b274404	grafana/agent:v0.32.1	"/bin/agent -config..."	6 days ago	Up 7 hours		grafana-docker-desktop-extension-agent
7b0d7d7d3e49	grafana/docker-desktop-extension:0.0.3	"/bin/sh -c '/servic..."	6 days ago	Up 7 hours		grafana_docker-desktop-extension-desktop

```
docker network connect docker-desktop-monitoring-hello-world_grafanet
<grafana agent ID>
```

```
docker network connect docker-desktop-monitoring-hello-world_grafanet
<grafana docker desktop extension ID>
```

Step 9: Navigate to the overview dashboard

<http://localhost:3000/d/docker-desktop-integration-docker/docker-desktop-docker-overview?orgId=1&refresh=30s>

