



Data Transformation & Automation

Lab2

Mau - Emre - Deep

AU^ROCON 3

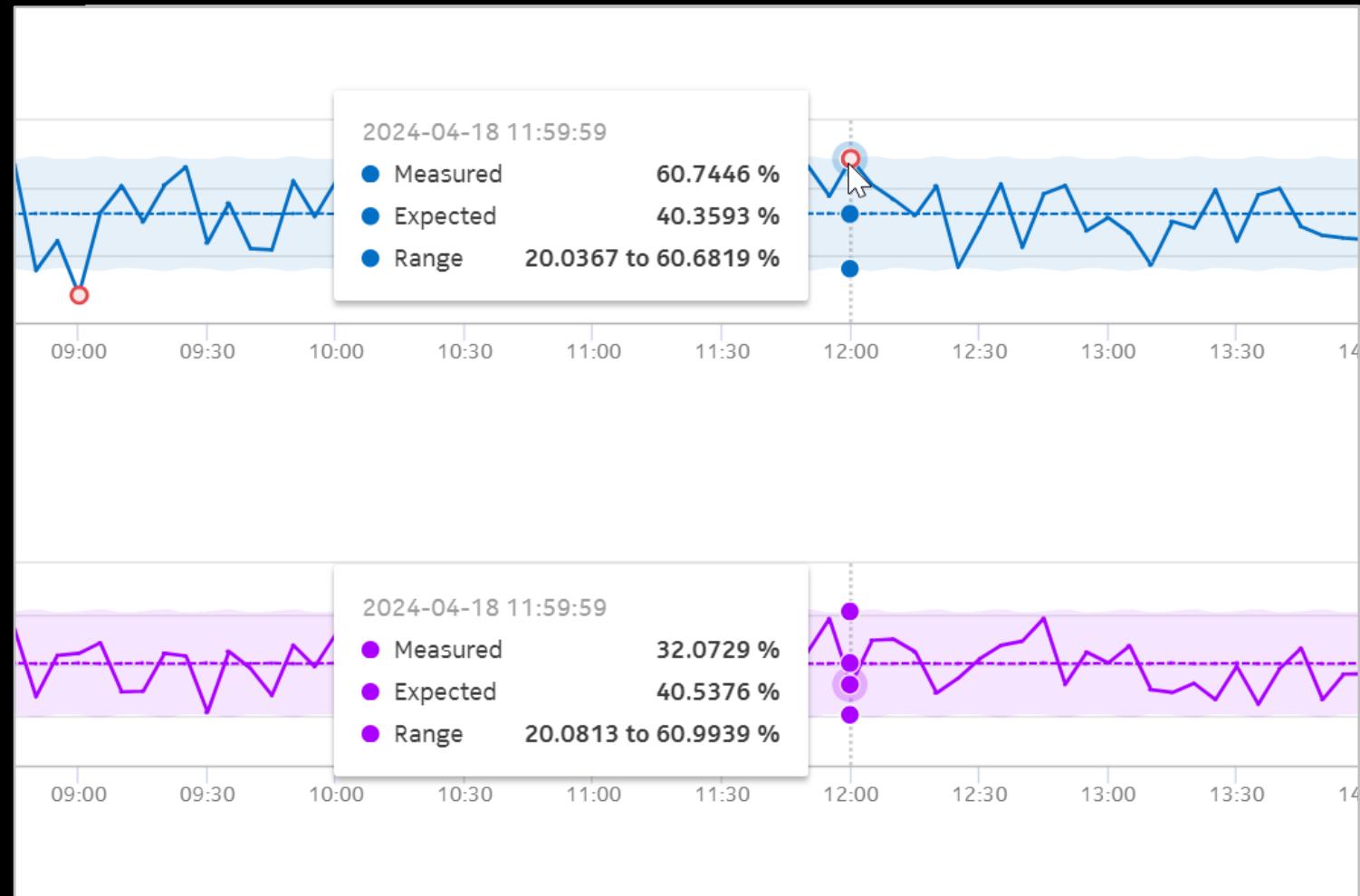
THE NETWORK AUTOMATION CONFERENCE

Lab-2: Data Transformation & Automation

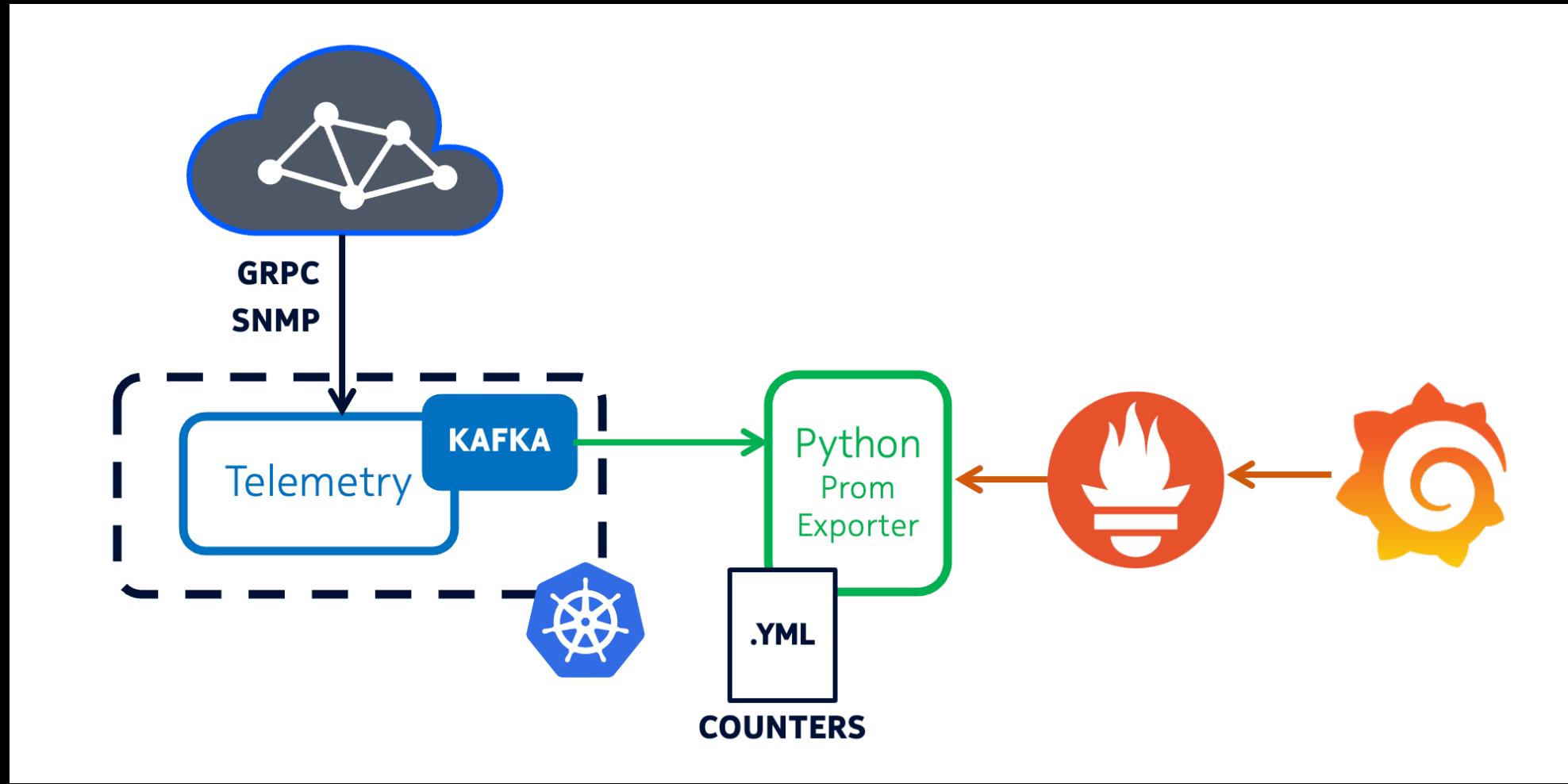
- Stream real-time network events using Kafka
- Visualize data and alerts with Grafana and Prometheus
- Use anomaly-based telemetry from Nokia analytics as input
- Learn to convert any data source into actionable network insights
- GitHub Actions

Nokia NSP: Baseline Analytics

- Identifies the “normal” or expected range of values for key resources with a configurable degree of precision.
- Flag historically anomalous data patterns that may be missed by simple threshold crossing algorithms.
- Might automatically trigger corrective actions by registering to anomaly events



Streaming Alarms





Lab Guide

Build Kafka Producer Images

```
$ docker build -t kafka-alarm-producer . -f Dockerfile.kafka-alarm-producer
```

```
[+] Building...
```

```
=> [1/8] FROM python:3.9-slim
=> [2/8] WORKDIR /app
=> [3/8] COPY requirements.txt
=> [4/8] RUN pip install -r requirements.txt
=> [5/8] COPY kafka-alarm-producer.py
=> [6/8] COPY kafka-check.py
=> [7/8] COPY kafka-test-producer.py
=> [8/8] COPY kafka-alarm-producer-config.yml

=> Exporting image as kafka-alarm-producer
```

Build Kafka Exporter Images

```
$ docker build -t kafka-alert-exporter . -f Dockerfile.kafka-alert-exporter
```

```
[+] Building...
```

```
=> [1/8] FROM python:3.9-slim
=> [2/8] WORKDIR /app
=> [3/8] COPY kafka-alert-exporter.py
=> [4/8] COPY kafka-msgs.py
=> [5/8] COPY kafka-check.py
=> [6/8] COPY alarms.yml
=> [7/8] COPY requirements.txt
=> [8/8] RUN pip install -r requirements.txt

=> Exporting image as kafka-alert-exporter
```

Docker images

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
kafka-alert-exporter	latest	7d2776407c66	26 hours ago	209MB
kafka-alarm-producer	latest	ffcbf63f9a33	26 hours ago	209MB
grafana/grafana	latest	b8e4a328293d	6 days ago	645MB
prom/prometheus	latest	0d815ee62986	6 days ago	293MB
prom/alertmanager	latest	d1950c730ae9	2 months ago	70.6MB
wurstmeister/kafka	latest	db97697f6e28	2 years ago	457MB
wurstmeister/zookeeper	latest	3f43f72cb283	6 years ago	510MB

Start the Kafka + Exporter Stack

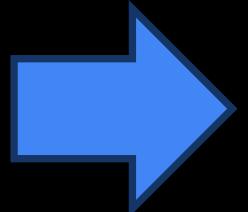
```
$ docker compose up -d
[+] Running 9/9
✓ Network kafka-alarm-sim_kafka-backend           Created 0.0s
✓ Container Prometheus                            Started 0.2s
✓ Container alertmanager                         Started 0.2s
✓ Container zookeeper                           Healthy 5.7s
✓ Container kafka-alert-exporter                Started 0.1s
✓ Container kafka-alert-grafana                 Started 0.2s
✓ Container zookeeper                           Healthy 0.5s
✓ Container kafka                            Started 5.7s
✓ Container kafka-alarm-producer              Started 5.8s
```

Access to Apps

Port	Service	Description
8001	Prometheus Exporter	Exposes raw metrics collected from Kafka messages (e.g., <code>anomaly_detected</code> , <code>thresholdViolation</code>)
9095	Prometheus UI	Interface to query time series, check targets, and validate alert rules
3000	Grafana	Visual dashboards (gauges, graphs) based on Prometheus data
9093	Alertmanager	View firing alerts and check routing logic

Lab2: Port Mapping

codespaces



PROBLEMS	OUTPUT	DEBUG CONSOLE	TERMINAL	PORTS 5
Port	Forwarded Address	Running P		
● 2181	https://laughing-eureka-j...	/usr/b		
● 8001	https://laughing... ↗ ⓘ	/usr/b		
● 9092	https://laughing-eureka-j...	/usr/b		
● 9093	https://laughing-eureka-j...	/usr/b		
● 9095	https://laughing-eureka-j...	/usr/b		

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

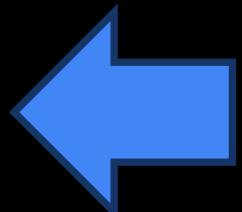
PORTS 13

Port

Forwarded Addr...

Running P

● 2181	localhost:2181	/usr/bin/do
● 2222	localhost:2222	sshd: /usr/
● 3000	localhost:3000	/usr/bin/do
● 8001	localhost:8001	/usr/bin/do
● 9092	localhost:9092	/usr/bin/do
● 9093	localhost:9093	/usr/bin/do
● 9095	localhost:9095	/usr/bin/do



local

Prometheus Exporter Configuration

- alarms.yml

```
alarms:  
  # - topic: 'rt-anomalies'  
  #   partition: 0  
  #   alertType: 'baseline'  
  #   counters:  
  #     anomaly_detected:  
  #       name: 'anomaly_detected'  
  #       description: 'Anomaly detected in real-time telemetry'  
  #       labels: ['neId', 'neName', 'counter']  
  - topic: 'nsp-act-action-event'  
    partition: 0  
    alertType: 'Threshold-Crossing'  
    counters:  
      thresholdViolation:  
        name: 'thresholdViolation'  
        description: 'Threshold Crossing Alert'  
        labels: ['source.neId', 'source.neName', 'payload.thresholdName']
```

Inspect Messages

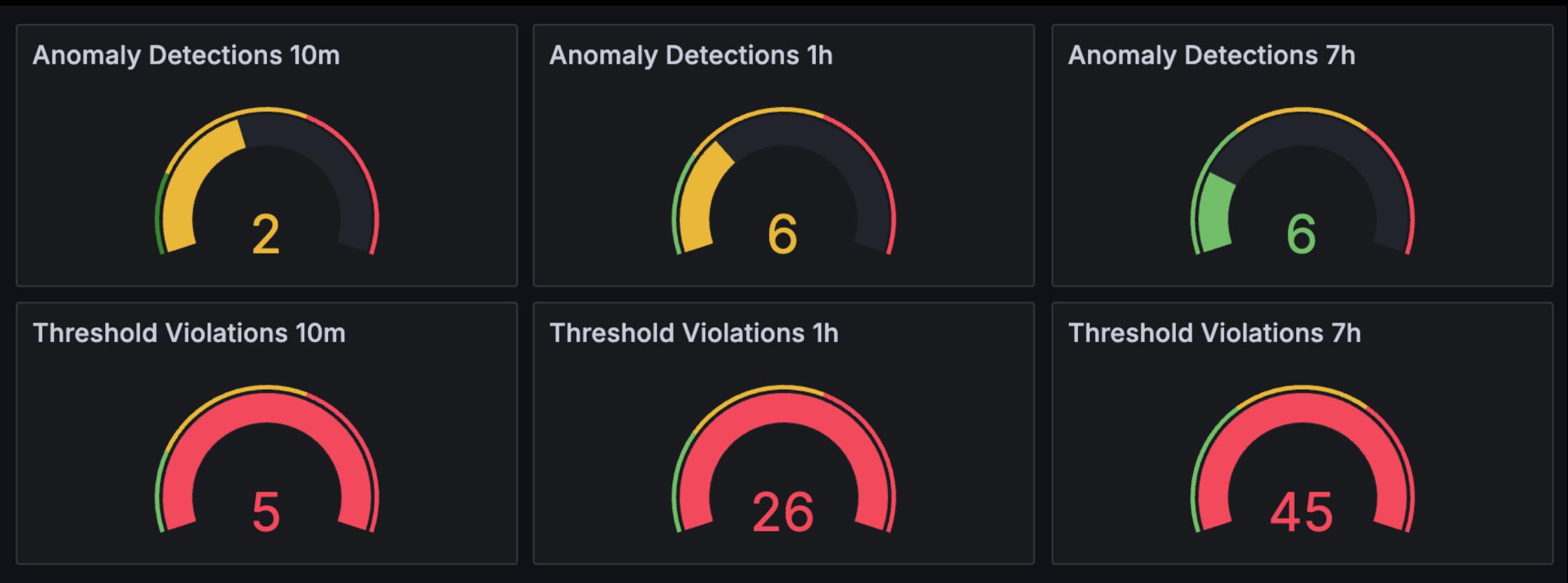
- ✉ Inspect Kafka Messages Being Consumed

```
docker exec -ti kafka-alert-exporter python3 kafka-msgs.py --  
bootstrap-servers kafka:9092
```

- 📜 View Exporter Logs

```
sudo docker logs --tail 10 kafka-alert-exporter
```

Lab2: Grafana Visualization



Lab 2: Prometheus Visualization

The screenshot shows the Prometheus web interface. At the top, there is a navigation bar with icons for back, forward, home, and search, followed by the URL https://silver-dollop-4j64jw4g7g6xc7xw-9095.app.github.dev/query?g0.expr=&g0.show_tree=0&g0.tab=table&g0.r..., and user authentication buttons.

The main header includes the Prometheus logo, a "Query" button, "Alerts", "Status", and navigation tabs for "Table", "Graph", and "Explain".

A message box indicates "No data queried yet".

The central area contains a query editor with the placeholder text `>_ {__name__ != ""}`. To the right of the editor are "Execute" and "..." buttons.

Below the editor, there are three tabs: "Table" (selected), "Graph", and "Explain".

An "Evaluation time" selector is present at the bottom left, and performance metrics "Load time: 48ms Result series: 45" are shown at the bottom right.

The results table displays two rows of data:

Series	Value
python_gc_objects_collected_total{generation="0", instance="kafka-alert-exporter:8001", job="nsp-kafka"}	136
python_gc_objects_collected_total{generation="1", instance="kafka-alert-exporter:8001", job="nsp-kafka"}	273



Github actions

Final Step: Publish Your Own Kafka Exporter Image

Step 1: Fork the Repository

Step 2: Set Up GitHub Actions for Docker Image Publishing

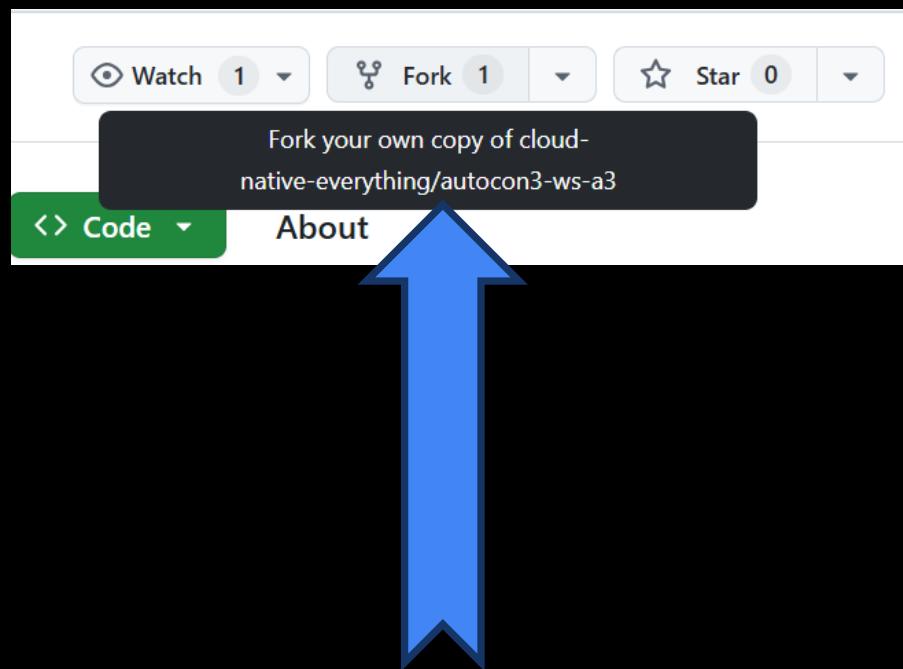
Step 3: Create a GitHub Token for Publishing

Step 4: Trigger the Build

Step 5: Change Package to Public

Step 6: Update docker-compose.yml

Step 1: Fork the Repository



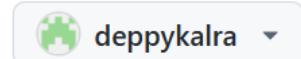
Click to Fork The Repo

Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)

Required fields are marked with an asterisk (*).

Owner *



deppykalra

Repository name *

autocon3-ws-a3

autocon3-ws-a3 is available.

By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description (optional)

Autocon3 event Prague Workshop - WS-A3 - Network App Development: Design, Deploy, & Automate

Copy the `main` branch only

Contribute back to `cloud-native-everything/autocon3-ws-a3` by adding your own branch. [Learn more.](#)

You are creating a fork in your personal account.

Create fork

Step 2: Set Up GitHub Actions for Docker Image Publishing

```
$ code .github/workflows/docker-publish.yml
```

//Paste the Workflow from README.md

Step 3: Create a GitHub Token for Publishing

Tokens (classic)

Note
GHCR Token

Expiration
30 days (Jun 23, 2025) ▾
The token will expire on the selected date

Select scopes
Scopes define the access for personal tokens. [Read more about OAuth scopes.](#)

<input checked="" type="checkbox"/> repo	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo_deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input checked="" type="checkbox"/> repo:invite	Access repository invitations
<input checked="" type="checkbox"/> security_events	Read and write security events
<input type="checkbox"/> workflow	Update GitHub Action workflows
<input checked="" type="checkbox"/> write:packages	Upload packages to GitHub Package Registry
<input checked="" type="checkbox"/> read:packages	Download packages from GitHub Package Registry

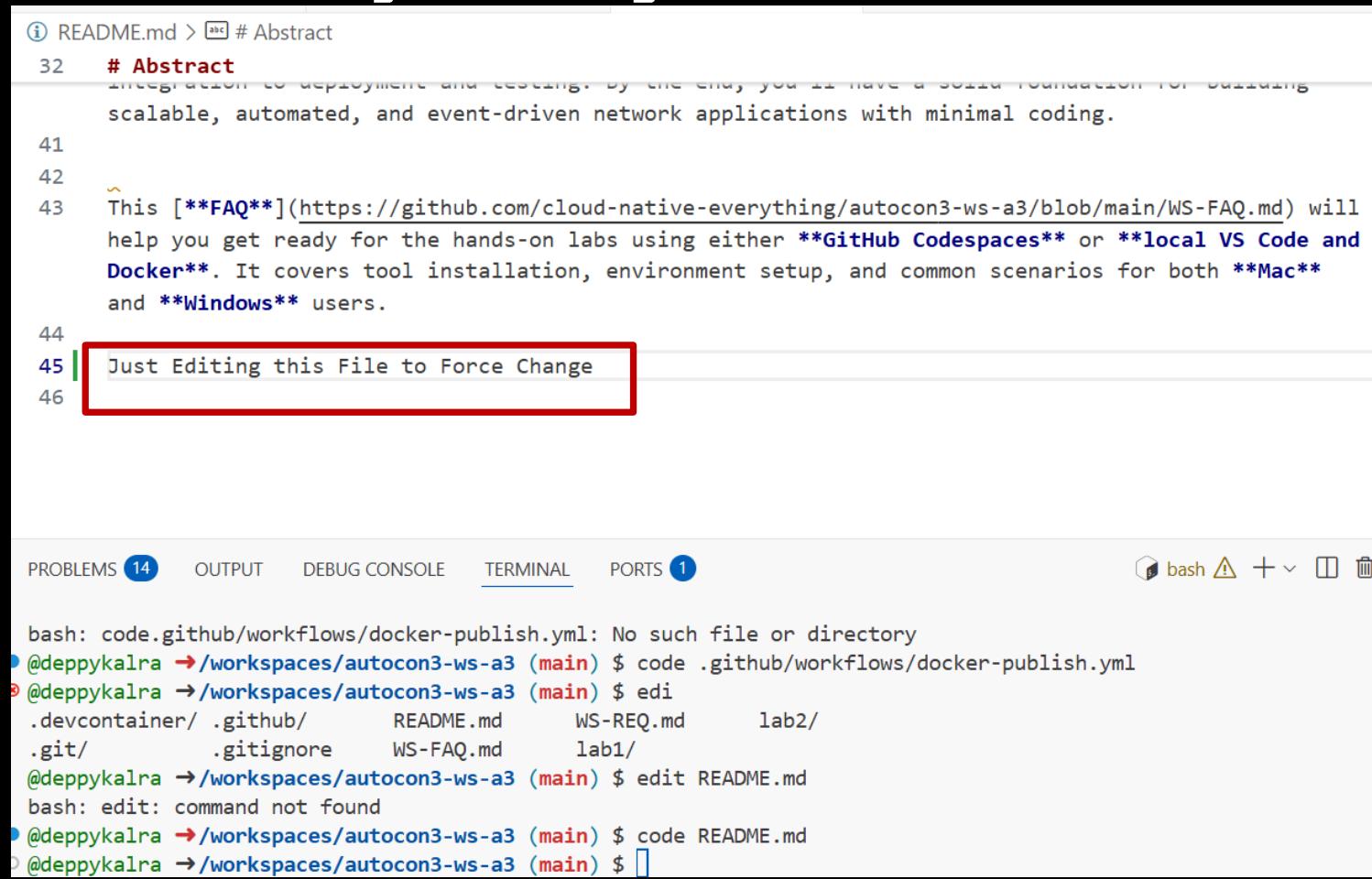
Attention/Pozor:
Copy the token once generated.
Used in next step

Step 3: Contd..

The screenshot shows a GitHub repository interface for the user 'deppykalra' with the repository name 'autocon3-ws-a3'. The top navigation bar includes links for Code, Pull requests, Actions, Projects, Security, Insights, and Settings. The Settings link is highlighted with a red box. The main content area is titled 'Security' and contains sections for Advanced Security, Deploy keys, Secrets and variables, Actions, Codespaces, and Dependabot. The 'Actions' section is currently selected, as indicated by a red box around its title. To the right, a message states 'This repository has no secrets.' and features a prominent green 'New repository secret' button, which is also highlighted with a red box.

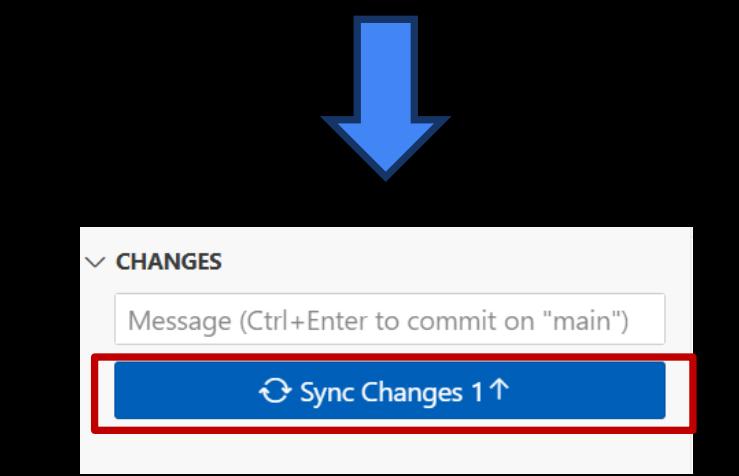
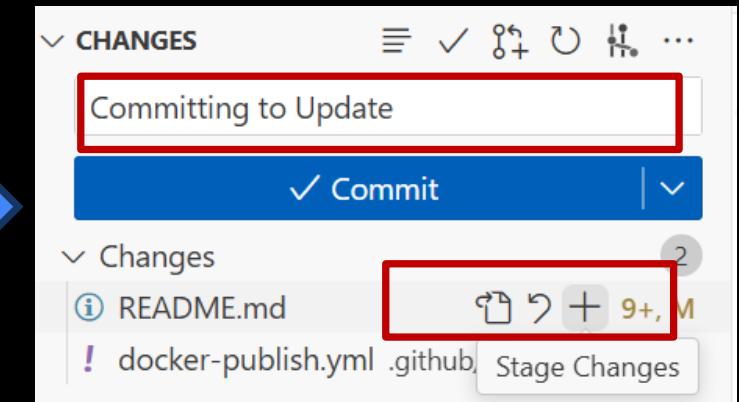
Step 4: Trigger the Build

- Edit Readme.md to force change
- Commit/Sync Changes

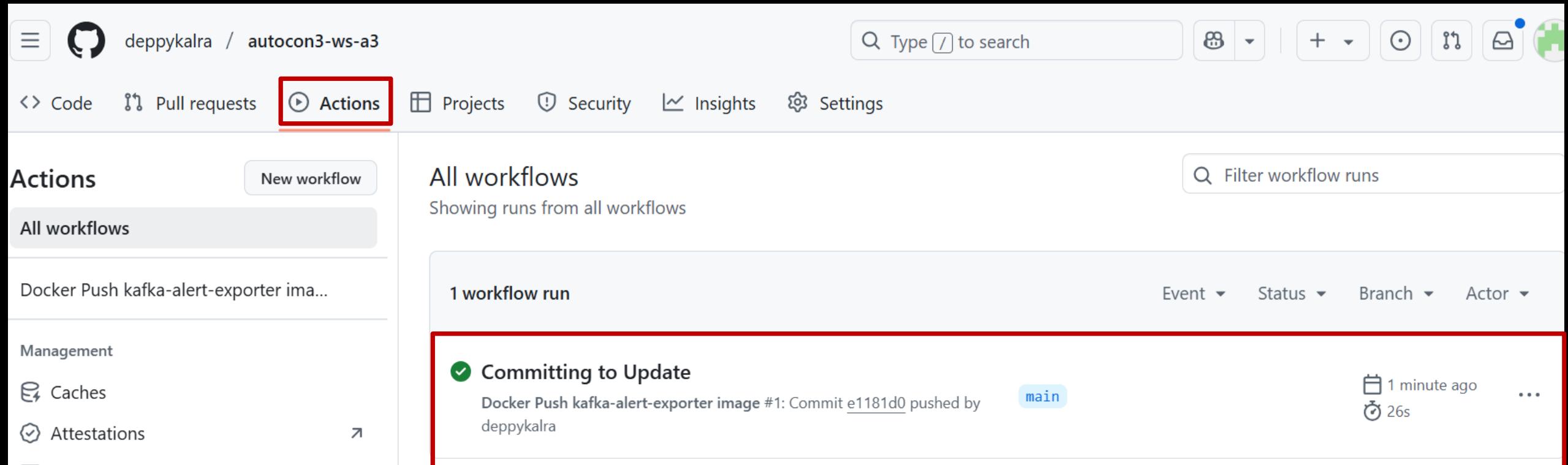


The screenshot shows the VS Code interface with the README.md file open. A red box highlights the line "Just Editing this File to Force Change" at the bottom of the file. The terminal below shows a series of commands being run in a bash shell, indicating a workflow or CI process.

```
bash: code.github/workflows/docker-publish.yml: No such file or directory
@deppykalra → /workspaces/autocon3-ws-a3 (main) $ code .github/workflows/docker-publish.yml
@deppykalra → /workspaces/autocon3-ws-a3 (main) $ edi
.devcontainer/.github/ README.md WS-REQ.md lab2/
.git/.gitignore WS-FAQ.md lab1/
@deppykalra → /workspaces/autocon3-ws-a3 (main) $ edit README.md
bash: edit: command not found
@deppykalra → /workspaces/autocon3-ws-a3 (main) $ code README.md
@deppykalra → /workspaces/autocon3-ws-a3 (main) $
```



Check The Workflow Status



The screenshot shows the GitHub Actions interface for the repository `deppykalra / autocon3-ws-a3`. The `Actions` tab is selected, highlighted with a red border. On the left sidebar, under the `Actions` section, there is a button labeled `New workflow`. The main area displays the `All workflows` section, showing one workflow run. The workflow run is titled `Committing to Update` and details the action: `Docker Push kafka-alert-exporter image #1: Commit e1181d0 pushed by deppykalra`. This specific run is highlighted with a red border. At the top right of the main area, there is a search bar with the placeholder `Type / to search` and a filter button labeled `Filter workflow runs`. The top navigation bar includes links for `Code`, `Pull requests`, `Projects`, `Security`, `Insights`, and `Settings`.

Step 5: Change Package to Public

The screenshot shows the GitHub Packages interface for the repository `deppykalra`. The user is on the `Packages` tab, viewing the package `autocon3-ws-a3` version `v1`. The package is currently listed as `Private`.

A large blue arrow points down to the `Change package visibility` button in a modal window. Another blue arrow points right from the `Change visibility` button to the `Public` radio button in the modal.

Danger Zone

- Change package visibility**
This package is currently private.
- Delete this package**
Once you delete a package, there is no going back. Please be certain.

Change visibility

Public
Make this package visible to anyone.

Private
Make this package visible privately, to organization members that have access.

Please type `autocon3-ws-a3` to confirm: *

autocon3-ws-a3

I understand the consequences, change package visibility

Step 6: Update docker-compose.yml

```
$ cd lab2/kafka-alarm-sim/  
$ code docker-compose.yml
```

```
▶ Run Service  
kafka-alert-exporter:  
  image: ghcr.io/deppykalra/autocon3-ws-a3:v1  
  container_name: kafka-alert-exporter  
  depends_on:  
    - kafka
```

Restart the Stack

```
$ docker compose up -d --force-recreate kafka-alert-exporter
```

Takeaways

You now:

- Own the Kafka exporter image in GHCR
- Use CI/CD with GitHub Actions
- Understand how to make Docker services portable and public



Questions?