



Link zu meiner GitHub

[CloudWatch-Praxis/Grafana-IAC at main · AlmohanadIlewi/CloudWatch-Praxis \(github.com\)](https://github.com/AlmohanadIlewi/CloudWatch-Praxis)

Ü131 - Grafana

Martin | 08.11.2023

Diese Übung dient dazu, eine Grafana-Instanz in einer EC2 Instanz aufzusetzen und erste Schritte mit Grafana zu machen. Zusätzlich soll mehr Routine im Umgang mit Terraform und Ansible entstehen und das erlernte Können vertieft werden.

Tagesziel: Eine Grafana Instanz auf einem EC2 Server kann Daten aus dem CloudWatch Feed des eigenen AWS Accounts anzeigen.

Um das Tagesziel zu erreichen, müssen nachfolgende Teilaufgaben erfüllt werden. Bei Aufgaben 1 und 2 ist die Abgabe von Code per git gern gesehen, ansonsten wäre ein Screenshot von der Ausführung wünschenswert (kann man auch in GitHub hochladen 😊).

Aufgabe 1: Deploy with Terraform!

Erstelle eine Terraform-Konfiguration, um die folgende Infrastruktur in AWS zu erstellen:

1. Eine Virtual Private Cloud (VPC) mit einem öffentlichen Subnetz.
2. Eine EC2 Instanz mit einer öffentlichen IP-Adresse im öffentlichen Subnetz.
3. Stelle sicher, dass die Ports 22/TCP und 3333/TCP der EC2 Instanz erreichbar sind.

Hilfreiche Links:

- <https://registry.terraform.io/providers/hashicorp/aws/latest>
- <https://developer.hashicorp.com/terraform/tutorials/aws-get-started/aws-build>

VPC > Your VPCs > vpc-0e42e24b3b9684b8c

vpc-0e42e24b3b9684b8c / grafana_vpc

DetailsInfo

VPC ID
vpc-0e42e24b3b9684b8c

Tenancy
Default

Default VPC
No

Network Address Usage metrics
Disabled

State
Available

DHCP option set
dopt-08b230beb1ddc4a4e

IPv4 CIDR
10.0.0.0/16

Route 53 Resolver DNS Firewall rule groups
Failed to load rule groups

DNS hostnames
Disabled

Main route table
rtb-0d8d88f1bd4bb62b

IPv6 pool
-

Owner ID
622445140057

DNS resolution
Enabled

Main network ACL
acl-04c23f0beb246dfce

IPv6 CIDR (Network border group)
-

Resource mapNew

CIDRs

Flow logs

Tags

Integrations

Resource mapInfo

VPCShow details
Your AWS virtual network

grafana_vpc

Subnets (1)
Subnets within this VPC

eu-central-1c

grafana-sb

Route tables (2)
Route network traffic to resources

grafana-rt

rtb-0d8d88f1bd4bb62b

Network connections (1)
Connections to other networks

grafana-gw

Was the resource map helpful today?

Give us feedback as often as possible. We are improving continually.

VPC > Subnets > subnet-01d505ea1e1df0dd8

subnet-01d505ea1e1df0dd8 / grafana-sb

Actions

Details

Subnet ID
subnet-01d505ea1e1df0dd8

Available IPv4 addresses
250

Network border group
eu-central-1

Default subnet
No

Customer-owned IPv4 pool
-

IPv6-only
No

DNS64
Disabled

Subnet ARN
arn:aws:ec2:eu-central-1:622445140057:subnet/subnet-01d505ea1e1df0dd8

IPv6 CIDR
-

VPC
vpc-0e42e24b3b9684b8c | grafana_vpc

Auto-assign public IPv4 address
Yes

Outpost ID
-

Hostname type
IP name

Owner
622445140057

State
Available

Availability Zone
eu-central-1c

Route table
rtb-0ef6237d3c482593 | grafana-rt

Auto-assign IPv6 address
No

IPv4 CIDR reservations
-

Resource name DNS A record
Enabled

IPv4 CIDR
10.0.1.0/24

Availability Zone ID
eu1-az1

Network ACL
acl-04c23f0beb246dfce

Auto-assign customer-owned IPv4 address
No

IPv6 CIDR reservations
-

Resource name DNS AAAA record
Disabled

EC2 > Instances > i-03477953193db2b37

Instance summary for i-03477953193db2b37 (grafana-ec2)Info

Updated less than a minute ago

RefreshConnectInstance stateActions

Instance ID
i-03477953193db2b37 (grafana-ec2)

IPv6 address
-

Hostname type
IP name: ip-10-0-1-15.eu-central-1.compute.internal

Answer private resource DNS name
IPv4 (A)

Auto-assigned IP address
3.69.24.130 [Public IP]

IAM Role
-

IMDSv2
Optional

Public IPv4 address
3.69.24.130 [open address]

Instance state
Running

Private IP DNS name (IPv4 only)
ip-10-0-1-15.eu-central-1.compute.internal

Instance type
t3.micro

VPC ID
vpc-0e42e24b3b9684b8c (grafana_vpc)

Subnet ID
subnet-01d505ea1e1df0dd8 (grafana-sb)

Private IPv4 addresses
10.0.1.15

Public IPv4 DNS
-

Elastic IP addresses
-

AWS Compute Optimizer finding
User: arn:aws:sts::622445140057:assumed-role/AWSReservedSSO_Student_0838bc6a67cc17c5/mohanad.ilewi@docc.techstarter.de is not authorized to perform: compute-optimizer:GetEnrollmentStatus on resource: * with an explicit deny in a service control policy
Retry

Auto Scaling Group name
-

DetailsSecurityNetworkingStorageStatus checksMonitoringTags

Instance detailsInfo

Platform
Ubuntu (Inferred)

Platform details
Linux/UNIX

Stop protection
Disabled

Instance auto-recovery
Default

AMI ID
ami-0aa6457dc2d115893

AMI name
ubuntu/images/hvm-ssd/ubuntu-focal-20.04-amd64-server-20231030

Launch time
Wed Nov 08 2023 22:51:09 GMT+0100 (Mitteleuropäische Normalzeit) (about 22 hours)

Lifecycle
general

Monitoring
disabled

Termination protection
Disabled

AMI location
amazon/ubuntu/images/hvm-ssd/ubuntu-focal-20.04-amd64-server-20231030

Stop-hibernate behavior
Disabled

VPC > Security Groups > sg-094a80a230a34e57f - grafana_ssh_sg

sg-094a80a230a34e57f - grafana_ssh_sg Actions

Details			
Security group name grafana_ssh_sg	Security group ID sg-094a80a230a34e57f	Description grafana_ssh_sg	VPC ID vpc-0e42e24b3b9684b8c
Owner 622445140057	Inbound rules count 4 Permission entries	Outbound rules count 2 Permission entries	

Inbound rules | Outbound rules | Tags

Inbound rules (4) Manage tags Edit inbound rules

Search

<input type="checkbox"/>	Name	Security group rule...	IP version	Type	Protocol	Port range	Source	Description
<input type="checkbox"/>	-	sgr-0fcd5b41ae8ba04a6	IPv6	Custom TCP	TCP	3333	::/0	Grafana
<input type="checkbox"/>	-	sgr-0aa9055a05e71ddee	IPv4	SSH	TCP	22	0.0.0.0/0	SSH
<input type="checkbox"/>	-	sgr-0c0446da5fec89c4	IPv6	SSH	TCP	22	::/0	SSH
<input type="checkbox"/>	-	sgr-0c3ec5aa4772bed6d	IPv4	Custom TCP	TCP	3333	0.0.0.0/0	Grafana

```

✓ TERMINAL
● PS C:\Users\Al Mohanad Ilewi\Desktop\Cloudwatch-Practice\Grafana-IAC\Terraform> terraform init

Initializing the backend...

Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Using previously-installed hashicorp/aws v5.24.0

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
● PS C:\Users\Al Mohanad Ilewi\Desktop\Cloudwatch-Practice\Grafana-IAC\Terraform> terraform fmt
main.tf
● PS C:\Users\Al Mohanad Ilewi\Desktop\Cloudwatch-Practice\Grafana-IAC\Terraform> terraform validate
Success! The configuration is valid.

○ PS C:\Users\Al Mohanad Ilewi\Desktop\Cloudwatch-Practice\Grafana-IAC\Terraform> 

```

Aufgabe 2: Configure with Ansible!

Schreibe ein Ansible-Playbook, das Grafana auf der EC2-Instanz installiert und ausführt. Hier sind grob die Schritte, die in der Rolle ausgeführt werden müssen:

1. Stelle sicher, dass das System auf dem neuesten Stand ist. Aktualisiere alle Pakete!
2. Stelle sicher, dass Grafana installiert werden kann. Füge das APT Repository von Grafana hinzu!
3. Installiere das Grafana Paket.
4. Kopiere die Konfigurationsdatei von GitHub auf den Server an den Pfad `/etc/grafana/grafana.ini`.

<https://raw.githubusercontent.com/martindubb/random-stuff/main/grafana/grafana.ini>

5. Stelle sicher, dass die Systemd Unit `grafana-server` läuft.

Hilfreiche Links:

- <https://grafana.com/docs/grafana/latest/setup-grafana/installation/debian/?plcm=learn-nav>
- Ansible Module
 - https://docs.ansible.com/ansible/latest/collections/ansible/builtin/apt_module.html#ansible-collections-ansible-builtin-apt-module
 - https://docs.ansible.com/ansible/latest/collections/ansible/builtin/copy_module.html#ansible-collections-ansible-builtin-copy-module
 - https://docs.ansible.com/ansible/latest/collections/ansible/builtin/systemd_service_module.html#ansible-collections-ansible-builtin-systemd-service-module

The screenshot shows a VS Code editor with a file explorer on the left and a terminal on the right. The file explorer shows a project named 'CLOUDWATCH-PRACTICE' with subfolders 'CloudWatch-Intro-Ue130', 'Grafana-IAC', and 'Ansible'. The 'Ansible' folder contains 'playbooks', 'roles', and 'handlers'. The 'main.yml' file is selected. The terminal shows the output of the command 'sudo ansible-playbook -i hosts playbooks/main.yml'. The output displays the tasks being executed: 'Restart Grafana service', 'Reload Grafana', and 'Start and enable Grafana service'. The terminal output is as follows:

```
2 - name: Restart Grafana
3   service:
4     name: grafana-server
5     state: restarted
6
7 - name: Reload Grafana
8   command: systemctl reload grafana-server
9   ignore_errors: true
10

TASK [grafana : Add Grafana APT repository] *****
ok: [grafana]

TASK [grafana : Install Grafana package] *****
ok: [grafana]

TASK [grafana : Start and enable Grafana service] *****
ok: [grafana]

PLAY RECAP *****
grafana : ok=6 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

mohanad@AWSCLoud:/mnt/c/Users/Al Mohanad Iilewi/Desktop/Cloudwatch-Practice/Grafana-IAC/Ansible$
```

The screenshot shows a VS Code editor with a file explorer on the left and a terminal on the right. The file explorer shows a project named 'CLOUDWATCH-PRACTICE' with subfolders 'CloudWatch-Intro-Ue130', 'Grafana-IAC', and 'Ansible'. The 'Ansible' folder contains 'playbooks', 'roles', and 'handlers'. The 'main.yml' file is selected. The terminal shows the output of the command 'sudo ansible-playbook -i hosts playbooks/main.yml'. The output displays the tasks being executed: 'copy grafana config file', 'Start and enable Grafana service', 'Gathering Facts', 'Update system packages', 'Install Grafana package', 'copy grafana config file', 'Start and enable Grafana service', 'Gathering Facts', 'Add Grafana GPG key', 'Add Grafana APT repository', and 'PLAY RECAP'. The terminal output is as follows:

```
13 - name: copy grafana config file
14   ansible.builtin.copy:
15     src: files/grafana.ini
16     dest: /etc/grafana/grafana.ini
17
18
19 - name: Start and enable Grafana service

PLAY [Install and Configure Grafana] *****

TASK [Gathering Facts] *****
ok: [grafana]

TASK [grafana : Update system packages] *****
ok: [grafana]

TASK [grafana : Install Grafana package] *****
ok: [grafana]

TASK [grafana : copy grafana config file] *****
ok: [grafana]

TASK [grafana : Start and enable Grafana service] *****
ok: [grafana]

PLAY [Grafana] *****

TASK [Gathering Facts] *****
ok: [grafana]

TASK [Add Grafana GPG key] *****
ok: [grafana]

TASK [Add Grafana APT repository] *****
ok: [grafana]

PLAY RECAP *****
grafana : ok=8 changed=0 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0

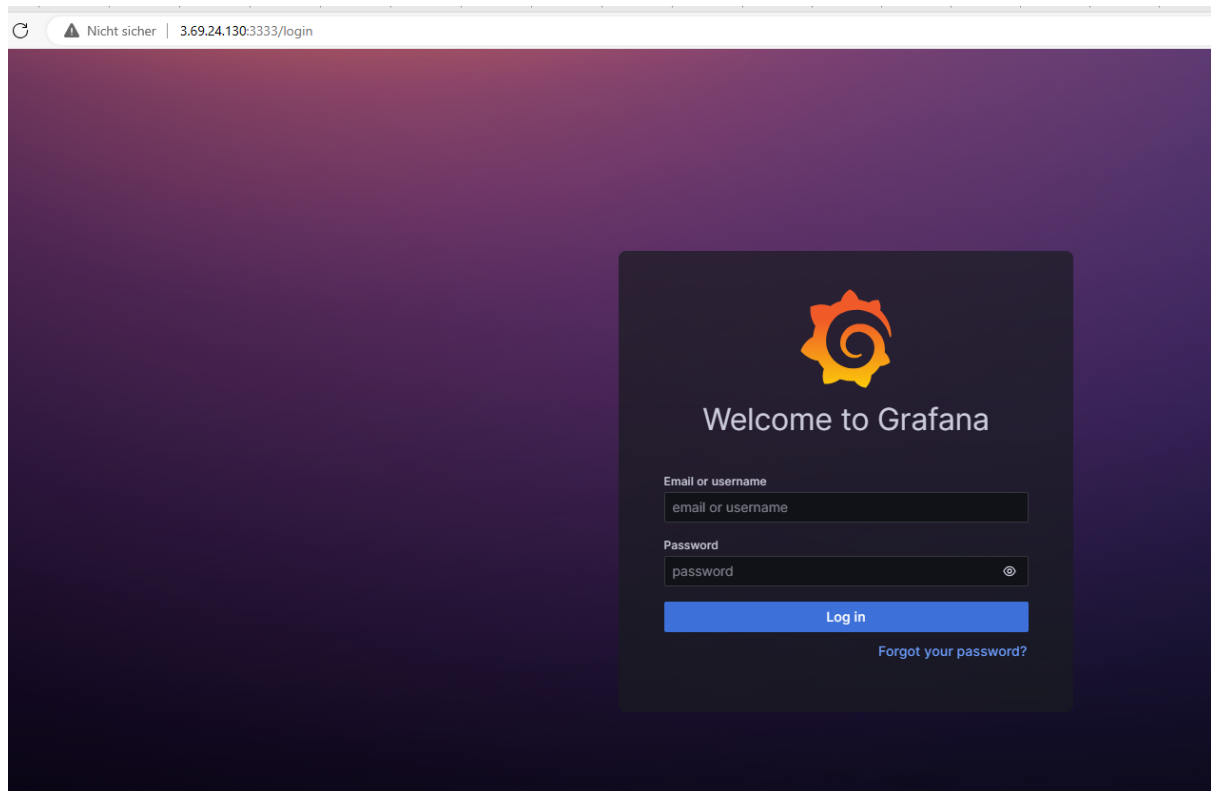
mohanad@AWSCLoud:/mnt/c/Users/Al Mohanad Iilewi/Desktop/Cloudwatch-Practice/Grafana-IAC/Ansible$
```

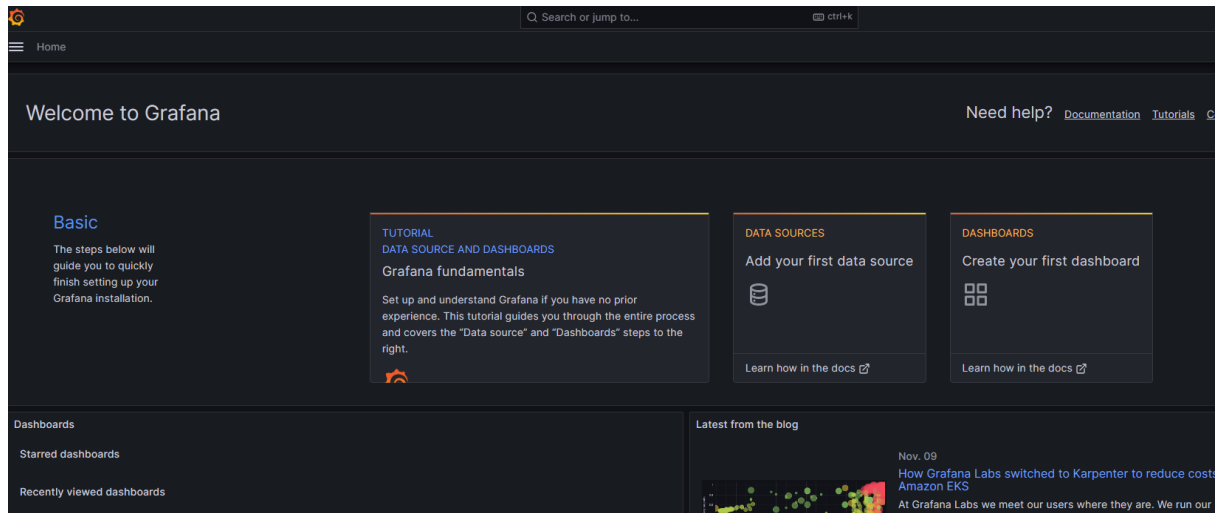
```
PROBLEMS OUTPUT TERMINAL GITLENS
✓ TERMINAL

ubuntu@ip-10-0-1-15:~$ cd /etc/grafana/
ubuntu@ip-10-0-1-15:/etc/grafana$ ls
grafana.ini  ldap.toml  provisioning
ubuntu@ip-10-0-1-15:/etc/grafana$ sudo cat grafana.ini | grep http_port
http_port = 3333
;root_url = %(protocol)s:%%(domain)s:%%(http_port)s/
ubuntu@ip-10-0-1-15:/etc/grafana$
```

```
ubuntu@ip-10-0-1-15:/etc/grafana$ netstat -plnt
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.53:53          0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:22            0.0.0.0:*               LISTEN      -
tcp6       0      0 :::3333                :::*                    LISTEN      -
tcp6       0      0 :::22                  :::*                    LISTEN      -
ubuntu@ip-10-0-1-15:/etc/grafana$
```

```
ubuntu@ip-10-0-1-15:~$ sudo -i
root@ip-10-0-1-15:~# sudo ss -tulpn | grep 3333
tcp        LISTEN      0          4096               *:3333             *:.*               users:((("grafana",pid=41353,fd=10))
root@ip-10-0-1-15:~#
```



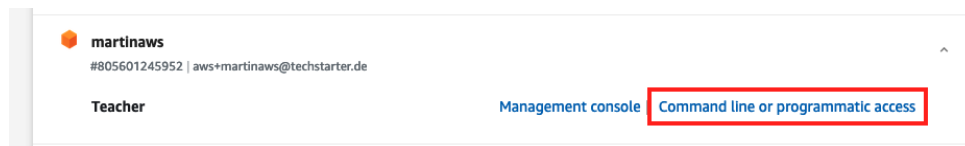


Aufgabe 3: Connect to AWS!

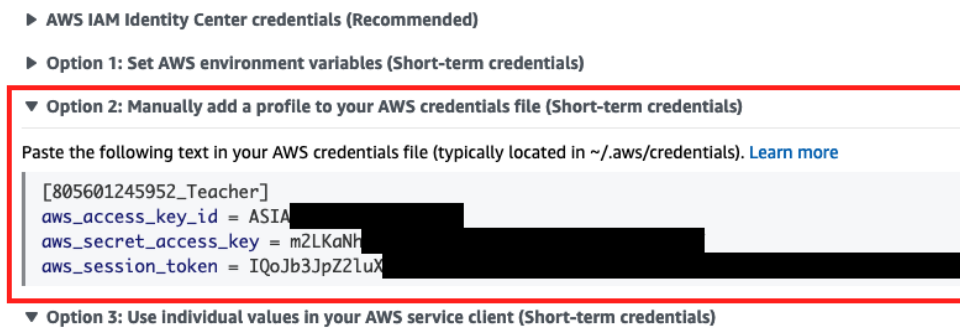
Nun muss noch die Verbindung von Grafana zu CloudWatch eingerichtet werden. Das muss mit den Techstarter Sandboxen in zwei Schritten gemacht werden.

Zuerst muss auf der EC2 Instanz eine Datei mit Zugangsdaten hinterlegt werden:
(Das Anlegen der Datei kann super mit Ansible automatisiert werden 😊)

1. Hole deine Zugangsdaten der Sandbox ab.



- a. Wir benötigen den Dateiinhalte unter "Option 2: Manually add a profile to your AWS credentials file (Short-term credentials)".



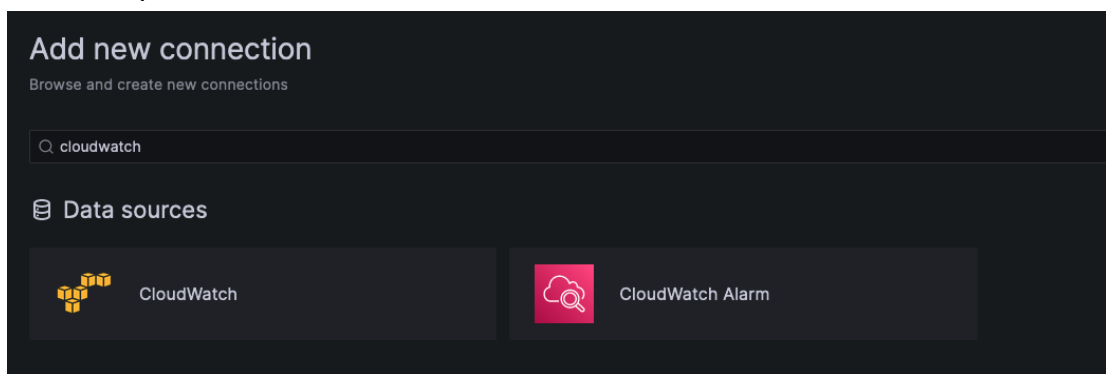
2. Logge dich per SSH auf der EC2 Instanz ein.
3. Erstelle im Ordner `/usr/share/grafana` den Ordner `.aws`.
4. Erstelle im Ordner `.aws` eine Datei mit dem Namen `credentials`.
5. Füge in die erstellte Datei `/usr/share/grafana/.aws/credentials` den Inhalt der Datei aus Schritt 1a ein.
6. Ersetze den Profilnamen mit "default". Die Datei sollte dann so aussehen:

```
Unset
[default]
```

```
aws_access_key_id=ASIA[...]  
aws_secret_access_key=m2LKaNh[...]  
aws_session_token=IQoJb3JpZ2luX[...]
```

Dann kann Grafana so konfiguriert werden, dass AWS CloudWatch als Datenquelle genutzt werden kann. Folge diesen Schritten, um den Zugriff einzurichten:

1. Öffne das Grafana-Dashboard und logge dich ein. Deine Grafana Instanz ist hier erreichbar: [http://\[ÖFFENTLICHE_IP_DEINER_EC2_INSTANZ\]:3333](http://[ÖFFENTLICHE_IP_DEINER_EC2_INSTANZ]:3333)
Die Zugangsdaten als Administrator sind *admin:admin*.
2. Wähle im Menü den Punkt Connections aus. Suche und Installiere CloudWatch als Datenquelle.



3. Füge eine neue Datenquelle vom Typ CloudWatch hinzu.
4. Konfiguriere den Zugriff zu aws wie folgt:
 - a. Authentication Provider: Credentials File
 - b. Default Region: eu-central-1
 - c. Bestätige die Funktionalität mit dem Klick auf den Button "Save & test"

Alerting supported	
Name ⓘ	CloudWatch
Default <input checked="" type="checkbox"/>	
Connection Details	
Authentication Provider ⓘ	Credentials file ▾
Credentials Profile Name ⓘ	default
Assume Role ARN ⓘ	arn:aws:iam:*
External ID ⓘ	External ID
Endpoint ⓘ	https://{service}.{region}.amazonaws.com
Default Region ⓘ	eu-central-1 ▾
Namespaces of Custom Metrics ⓘ	Namespace1, Namespace2

Settings
Dashboards

Alerting supported

Name ⓘ CloudWatch
Default
☒

Connection Details

Authentication Provider ⓘ	Credentials file ▾
Credentials Profile Name ⓘ	default
Assume Role ARN ⓘ	arn:aws:iam:*
External ID ⓘ	External ID
Endpoint ⓘ	https://{service}.{region}.amazonaws.com
Default Region ⓘ	eu-central-1 ▾
Namespaces of Custom Metrics ⓘ	Namespace1, Namespace2

CloudWatch Logs

Query Result Timeout ⓘ	30m
Default Log Groups ⓘ	Select log groups

X-ray trace link

Grafana will automatically create a link to a trace in X-ray data source if logs contain @xrayTraceId field

ⓘ There is no X-ray datasource to link to. First add an X-ray data source and then link it to Cloud Watch.

Data source ⓘ Select data source ▾

✓ 1. Successfully queried the CloudWatch metrics API. 2. Successfully queried the CloudWatch logs API.
Next, you can start to visualize data by [building a dashboard](#), or by querying data in the [Explore view](#).

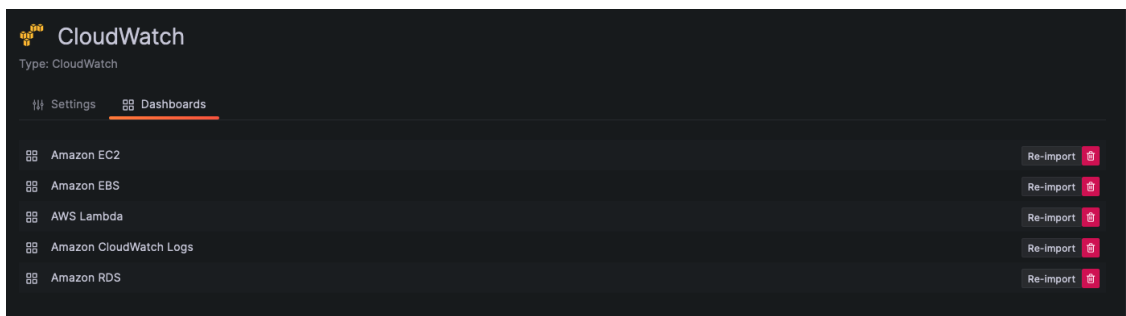
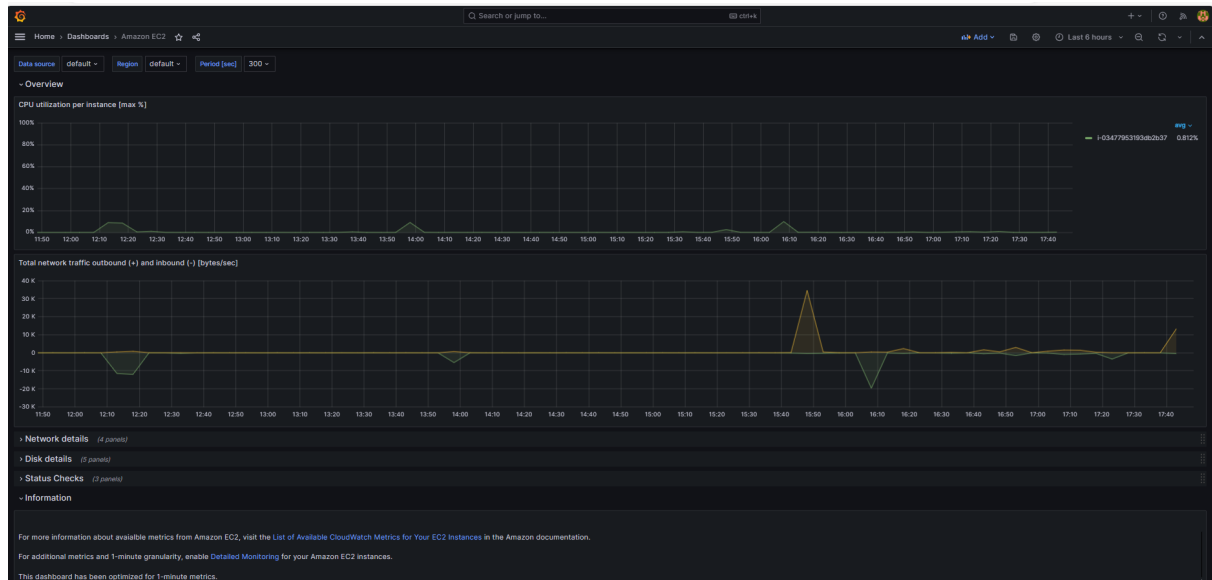
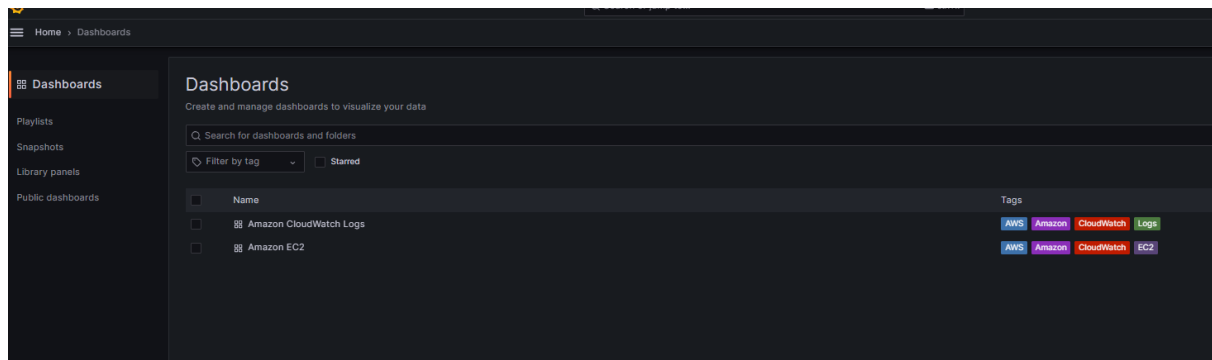
Delete Save & test

Hilfreiche Links:

- <https://grafana.com/docs/grafana/latest/datasources/?plcmt=learn-nav>
- <https://grafana.com/docs/grafana/latest/datasources/aws-cloudwatch/aws-authentication/?plcmt=learn-nav#use-an-aws-credentials-file>

Aufgabe 4: Explore Grafana

Du kannst ein paar default Dashboards für CloudWatch importieren und dir anschauen, was für Metriken angezeigt werden.



Schau dich in Ruhe in Grafana um. Hier gibt es ein weiteres Tutorial für Einsteiger:

<https://grafana.com/tutorials/grafana-fundamentals/?plcmt=learn-nav>