

Spring actuator

the Spring actuator was already available and pre-configured.

Prometheus und Alertmanager

I've added a `docker-compose.yml`, which can be run with `docker compose up` to scaffold the entire application, including the Spring Boot App.

It leverages Dockers' internal networking capability, which let the services talk to one another inside their internal network.

This also requires a `.env` file, which can be derived from the included `.env.example` file.

Booting the services and running a little loadtest, which is included in `send_many_requests.sh` triggers the configured alerts on purpose.

Inspecting the Alerts panel inside Prometheus indicates, that the configured alerts are firing as intended.

http://localhost:9090/alerts?search= Prometheus Alerts Graph Status ▾ Help

🔍 Filter by name or labels Show annotations

/etc/prometheus/rules/too_many_requests.yml > AllInstances inactive

> InstanceDown (0 active)

/etc/prometheus/rules/too_many_requests.yml > default firing (1)

▼ RequestRate (1 active)

name: RequestRate
expr: rate(http_server_requests_seconds_count{uri="/actuator"}[1m]) > 0
for: 1m
labels:
 severity: critical
annotations:
 summary: Application receiving too many requests

Labels	State	Active Since	Value
alertname=RequestRate exception=None instance=st-testing-application:8080 job=spring-actuator method=GET outcome=SUCCESS severity=critical status=200 uri=/actuator	FIRING	2024-06-07T14:23:52.74783594Z	38.142680532324924

Prometheus Alerts Graph Status ▾ Help

🔍 Filter by name or labels Show annotations

/etc/prometheus/rules/too_many_requests.yml > AllInstances inactive

> InstanceDown (0 active)

/etc/prometheus/rules/too_many_requests.yml > default pending (1)

▼ RequestRate (1 active)

name: RequestRate
expr: rate(http_server_requests_seconds_count{uri="/actuator"}[1m]) > 0
for: 1m
labels:
 severity: critical
annotations:
 summary: Application receiving too many requests

Labels	State	Active Since	Value
alertname=RequestRate exception=None instance=st-testing-application:8080 job=spring-actuator method=GET outcome=SUCCESS severity=critical status=200 uri=/actuator	PENDING	2024-06-07T14:03:07.74783594Z	42.81217

The alertmanager is triggered by prometheus, and sends webhooks by default.

Filter

Group

Receiver: All ☐ Silenced ☐ Inhibited

+

Silence

Custom matcher, e.g. `env="production"`

+ Expand all groups

webhook alertname="RequestRate" 1 alert

2024-06-07T14:24:52.747Z Info Source Silence Link

exception="None" +

instance="st-testing-application:8080" +

job="spring-actuator" +

method="GET" +

outcome="SUCCESS" +

severity="critical" +

status="200" +

uri="/actuator" +

Trying to use an in-build solution by the alertmanger, PagerDuty, which is able to send phone notifications beside email notifications for alerts required a US Phone number,

which I did not have access to.

Welcome to PagerDuty!

First, let's send a test
notification

We will send you a test incident via
SMS, phone, and email.

☒ Phone & SMS

+49

151



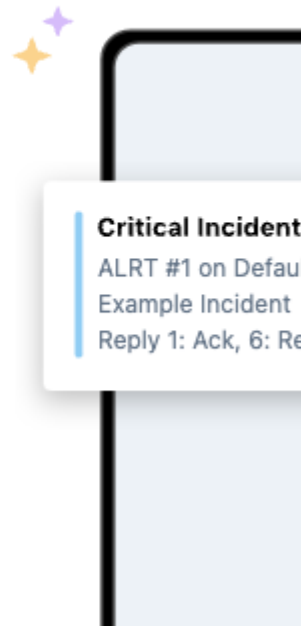
⚠ For your country code,
please [submit a ticket](#) to our
support team to enable SMS.

☒ Email

v.mattfeld@stud.uni-goettingen

[Skip step](#)

[Send test](#)




Therefore, I defaulted to using the a GMail-based notification method, which sent emails from a Google-Mail account. That required to setup an application password, which would be required by the alertmanager.

Create & use app passwords

Important: To create an app password, you need 2-Step Verification on your Google Account.

If you use 2-Step-Verification and get a "password incorrect" error when you sign in, you can try to use an app password.

1. Go to your [Google Account](#) .
2. Select **Security**.
3. Under "How you sign in to Google," select **2-Step Verification**.
4. At the bottom of the page, select **App passwords**.
5. Enter a name that helps you remember where you'll use the app password.
6. Select **Generate**.
7. To enter the app password, follow the instructions on your screen. The app password is the 16-character code that generates on your device.
8. Select **Done**.

If you've set up 2-Step Verification but can't find the option to add an app password, it might be because:

- Your Google Account has 2-Step Verification [set up only for security keys](#).
- You're logged into a work, school, or another organization account.
- Your Google Account has [Advanced Protection](#).

Tip: Usually, you'll need to enter an app password once per app or device.

← 2-Faktor-Authentifizierung

Die 2-Faktor-Authentifizierung ist für Ihr Konto aktiviert

Fügen Sie Ihrem Konto eine zusätzliche Sicherheitsebene hinzu, um es besser vor Hackern zu schützen.











Wenn Sie sich nicht mit einem Passkey anmelden, werden Sie aufgefordert, den sichersten zweiten Schritt, den Sie für Ihr Konto eingerichtet haben, auszuführen. Sie können die zweiten Schritte und die Anmeldeoptionen jederzeit in den Einstellungen ändern. [Zu den Sicherheitseinstellungen](#) ↻



[2-Faktor-Authentifizierung deaktivieren](#)

Zweite Schritte

Achten Sie darauf, dass diese Informationen auf dem neuesten Stand sind, und fügen Sie weitere Anmeldeoptionen hinzu, damit Sie jederzeit auf Ihr Google-Konto zugreifen können

	Passkeys und Sicherheitsschlüssel	 2 Sicherheitsschlüssel	>
	Aufforderung von Google	 3 Geräte	>
	Authentifizierungs-App	 Authentifizierungs-App hinzufügen	>
	Telefonnummer	 Telefonnummer hinzufügen	>
	Back-up-Codes	 10 Codes verfügbar	>

← App-Passwörter

Mit App-Passwörtern können Sie sich in älteren Apps und Diensten, die keine modernen Sicherheitsstandards unterstützen, in Ihrem Google-Konto anmelden.

App-Passwörter sind nicht so sicher wie die Verwendung aktueller Apps und Dienste, die moderne Sicherheitsstandards erfüllen. Bevor Sie ein App-Passwort erstellen, sollten Sie prüfen, ob dies für die Anmeldung in Ihrer App erforderlich ist.

[Weitere Informationen](#)

Sie haben keine App-Passwörter.

Um ein neues App-spezifisches Passwort zu erstellen, geben Sie unten einen Namen dafür ein...

App-Name

software-testing-ss2024

Erstellen

After that, I had alertmanager and prometheus configured to use GMail for notifications.

Triggering the alerts with a load test yielded the desired results.

http://localhost:9090/alerts?search= Prometheus Alerts Graph Status ▾ Help

☒ Inactive (2) ☒ Pending (0) ☒ Firing (2) ☐ Show annotations

/etc/prometheus/rules/too_many_requests.yml > AllInstances inactive

> InstanceDown (0 active)

/etc/prometheus/rules/too_many_requests.yml > default inactive firing (2)

> RequestRate (0 active)

> Wrong Method (1 active)

> Not found (1 active)

Expand all groups

— gmail-notifications

Not grouped 2 alerts

2024-06-07T14:56:37.747Z

[+ Info](#)

[Source](#)

[Silence](#)

[Link](#)

alertname="Not found" +

exception="None" +

instance="st-testing-application:8080" +

job="spring-actuator" +

method="GET" +

outcome="CLIENT_ERROR" +

severity="critical" +

status="404" +

uri="/**" +

2024-06-07T14:56:37.747Z

[+ Info](#)

[Source](#)

[Silence](#)

[Link](#)

alertname="Wrong Method" +

exception="HttpRequestMethodNotSupportedException" +

instance="st-testing-application:8080" +

job="spring-actuator" +

method="GET" +

outcome="CLIENT_ERROR" +

severity="critical" +

status="405" +

uri="root" +

Email Notifications were received.

Posteingang

Filter ▾

valledalle@googlemail.com

[FIRING:1] (st-testing-application:8080 sprin

17:07

2 alerts for View in Alertmanager [1] Firing Labels ale...



Heute, 17:07

Mattfeld, Valerius Albert Gongjus ✕

2 alerts for

[View In Alertmanager](#)

[1] Firing

Labels

alertname = Not found
exception = None
instance = st-testing-application:8080
job = spring-actuator
method = GET
outcome = CLIENT_ERROR
severity = critical
status = 404
uri = /**

Annotations

summary = Requested endpoint did not exist
[Source](#)

[1] Resolved

Labels

alertname = Wrong Method
exception = HttpRequestMethodNotSupportedException
instance = st-testing-application:8080
job = spring-actuator
method = GET
outcome = CLIENT_ERROR
severity = critical
status = 405
uri = root

Annotations

summary = Application received wrong request type
[Source](#)

[Sent by Alertmanager](#)

Using Graphana and the Metric Explorer, it was easy to setup a panel for each relevant metric, including system load and the alert frequencies in sum and specific.

