

# Speedtest Tracker

Docker Container um automatisch die Geschwindigkeit der Internetleitung zu testen. Dashboard Admin & öffentlich, Historie, E-Mail Benachrichtigung, Benutzerverwaltung, Test-Serverauswahl.

- [Installation](#)
- [Dashboards](#)
- [Speedtests durchführen](#)
- [Ergebnisse als Tabelle](#)
- [Mail Benachrichtigungen](#)

# Installation

Die Installation ist mit Docker Compose schnell erledigt.

```
version: '3.4'
services:
  speedtest-tracker:
    image: lscr.io/linuxserver/speedtest-tracker:latest
    container_name: speedtest-tracker
    ports:
      - 80:80
      - 443:443
    environment:
      - PUID=1000
      - PGID=1000
      - APP_NAME="Speedtest Tracker"
      - APP_URL=http://localhost
      - DB_CONNECTION=mysql
      - DB_HOST=speedtest-db
      - DB_PORT=3306
      - DB_DATABASE=speedtest_tracker
      - DB_USERNAME=speedy
      - DB_PASSWORD=password
    volumes:
      - /pfad/zur/config:/config
      - /pfad/zum/web:/etc/ssl/web
    restart: unless-stopped
    depends_on:
      - speedtest-db

  speedtest-db:
    image: mariadb:10
    container_name: speedtest-db
    restart: unless-stopped
    environment:
      - MARIADB_DATABASE=speedtest_tracker
      - MARIADB_USER=speedy
```

```
- MARIADB_PASSWORD=password
- MARIADB_RANDOM_ROOT_PASSWORD=true
volumes:
- /pfad/zur/db:/var/lib/mysql
```

Vor dem Ausführen der Konfiguration sollte das Passwort für die MariaDB angepasst werden, dazu das Passwort an den beiden Stellen (`DB_PASSWORD=password` und `MARIADB_PASSWORD=password`) anpassen. Für den Benutzer `root` in der MariaDB wird automatisch ein zufälliges Passwort gesetzt (`MARIADB_RANDOM_ROOT_PASSWORD=true`).

Zusätzlich sind noch die Pfade für die Volumes anzupassen, am besten einen eigenen Ordner erstellen, sodass darin die 3 Ordner für die Konfigurationen und Daten gespeichert werden können:

- `/pfad/zur/config:/config`
- `/pfad/zum/web:/etc/ssl/web`
- `/pfad/zur/db:/var/lib/mysql`

Um die Docker Compose Konfiguration auszuführen, kann am besten in das Verzeichnis der YAML Datei gewechselt werden. Danach wird je nach gewählter Installation `sudo docker-compose up -d` oder `sudo docker compose up -d` (keine Bindestrich zwischen docker und compose) eingegeben, um die Standard Konfiguration `docker-compose.yml` zu starten. Compose erstellt dann die gewünschten Container mit den angegebenen Optionen. Sollten die Container bereits mit dieser Compose Konfiguration erstellt worden sein, so werden die Container in dieser neu erstellt, dessen Konfiguration geändert wurde.

Anschließend ist die Seite des Speedtest unter `http://Server-IP` und `https://Server-IP` erreichbar.

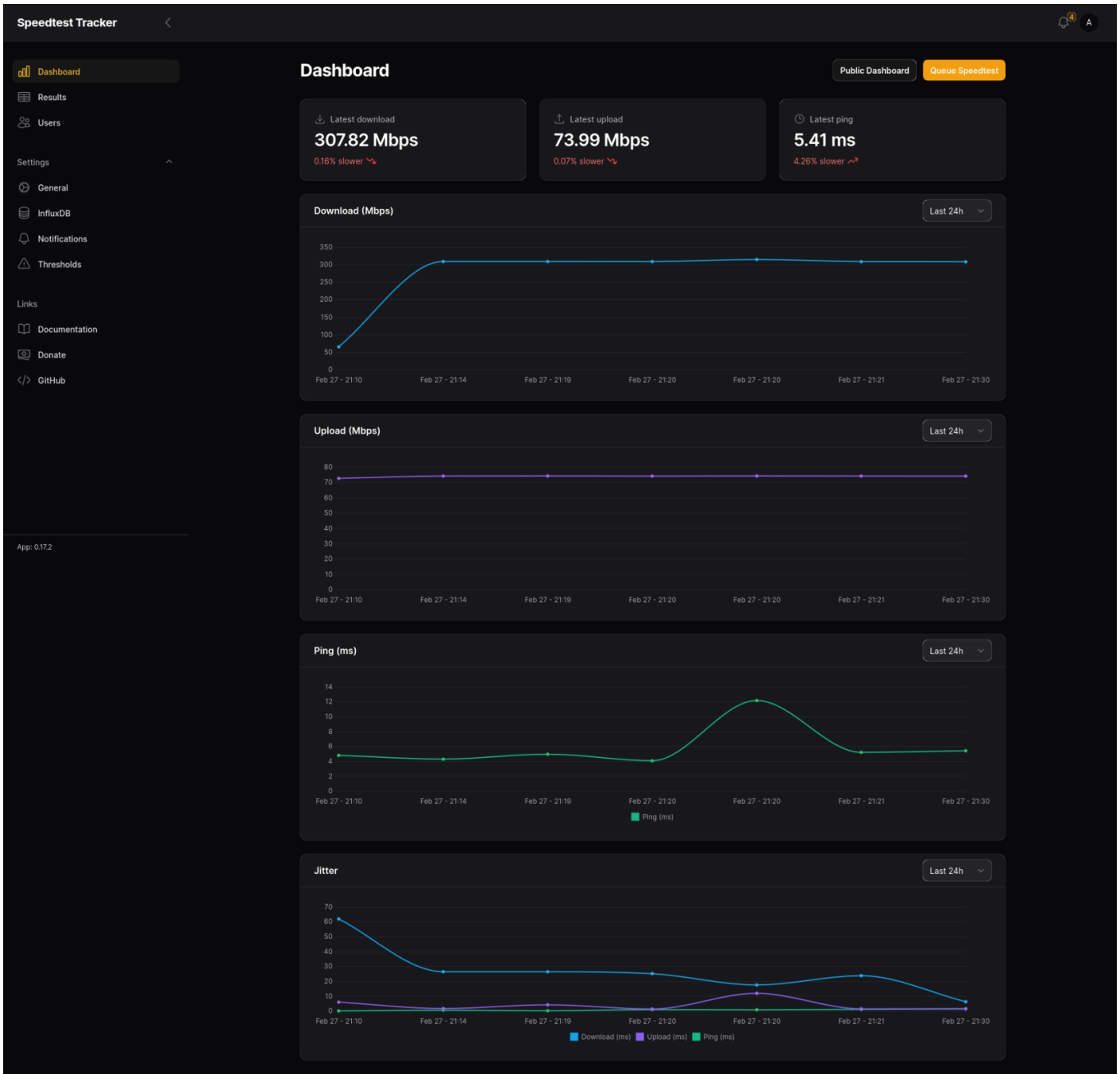
Da Standard Login-Daten sind:

- Email address: `admin@example.com`
- Password: `password`

# Dashboards

## Admin Dashboard

Die Startseite des Admin-Backends ist das Dashboard. Dieses öffnet sich automatisch nach dem Login. Hier werden die letzten Messungen der vergangenen 24h als Grafen angezeigt. Zusätzlich werden ganz oben die letzten Messergebnisse als absolute Werte angezeigt sowie deren prozentuale Veränderung zur vorherigen Messung. Zusätzlich wird links ein Menüband angezeigt, um weitere Menüs und Einstellungen erreichen zu können.



# Öffentliches Dashboard

Zusätzlich wird ein öffentliches Dashboard angeboten, welches auch ohne Login zugänglich ist. Dieses kann in *Settings - General - Public Dashboard* aktiviert werden.



Anschließend wird im Admin Dashboard eine neue Schaltfläche aktiviert: Public Dashboard

Über diese kann das Dashboard geöffnet werden oder alternativ über den gleichen Link wie für das Admin-Backend. Wenn der aktuelle Benutzer nicht eingeloggt ist, dann öffnet sich automatisch das öffentliche Dashboard und es muss explizit ins Backend gewechselt und eingeloggt werden.

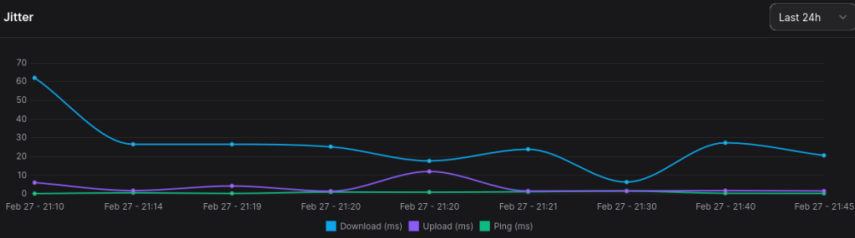
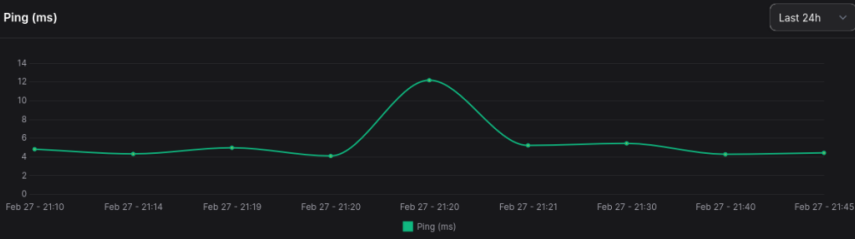
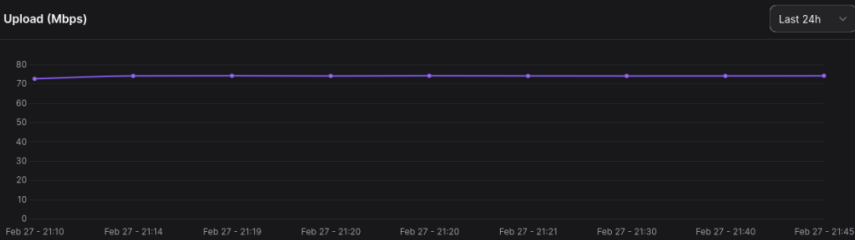
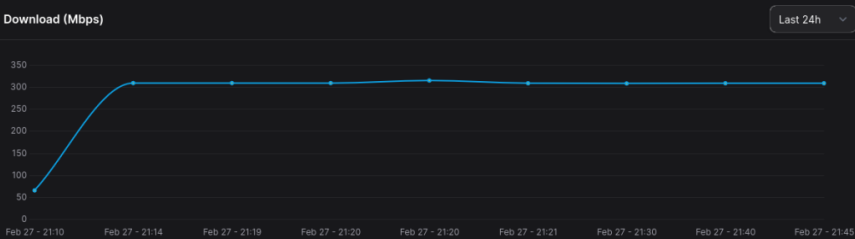
Mehr Einstellungen können nicht vorgenommen werden, das öffentliche Dashboard zeigt die selben grafischen Verläufe und die letzte Messung an, wie das Admin Dashboard, wie folgend zu sehen ist.

Latest download  
**308.15 Mbps**  
0.01% slower ↘

Latest upload  
**74.07 Mbps**  
0.07% faster ↗

Latest ping  
**4.40 ms**  
3.55% slower ↘

Latest result: 2 minutes ago

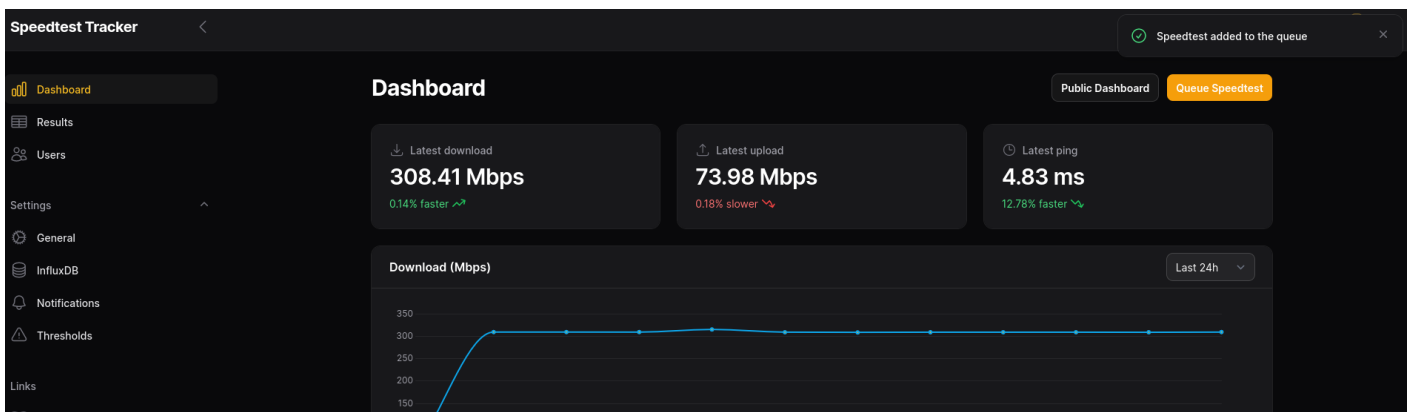


# Speedtests durchführen

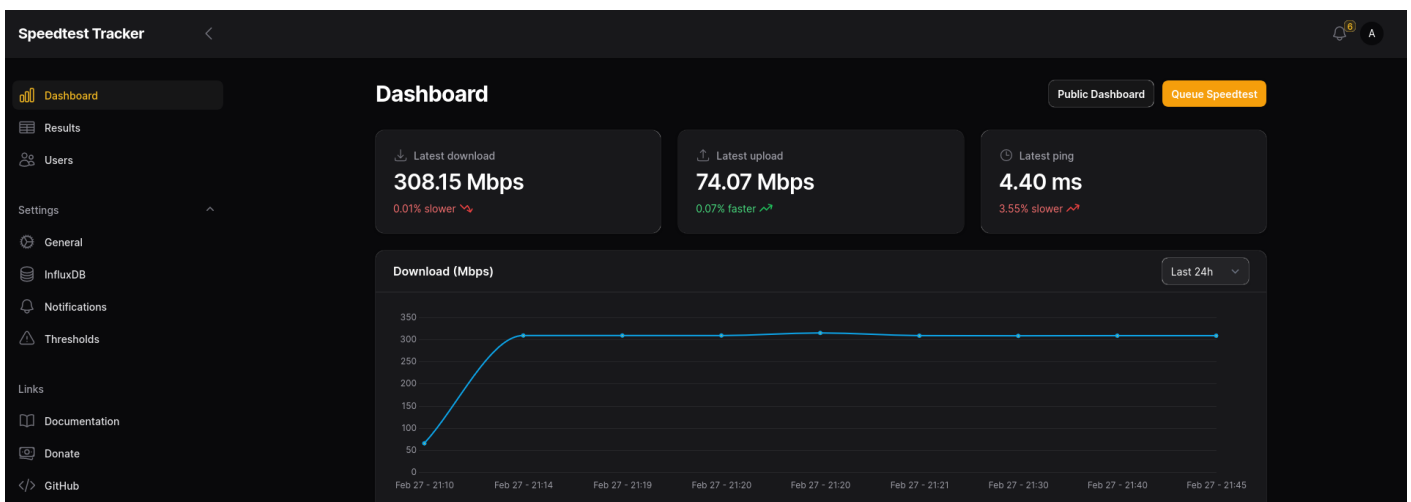
Speedtests können jederzeit manuell im Browser gestartet werden oder zeitgesteuert.

## Manueller Speedtest

Eben schnell einen Speedtest anzustoßen ist schnell erledigt. Einfach das Admin Dashboard aufrufen und es wird rechts oben ein gelb leuchtende Schaltfläche mit dem Namen *Queue Speedtest* angezeigt. Wird diese angeklickt, wird ein Speedtest angefordert und wenige Sekunden später ausgeführt. Dies wird durch eine Meldung rechts oben bestätigt.

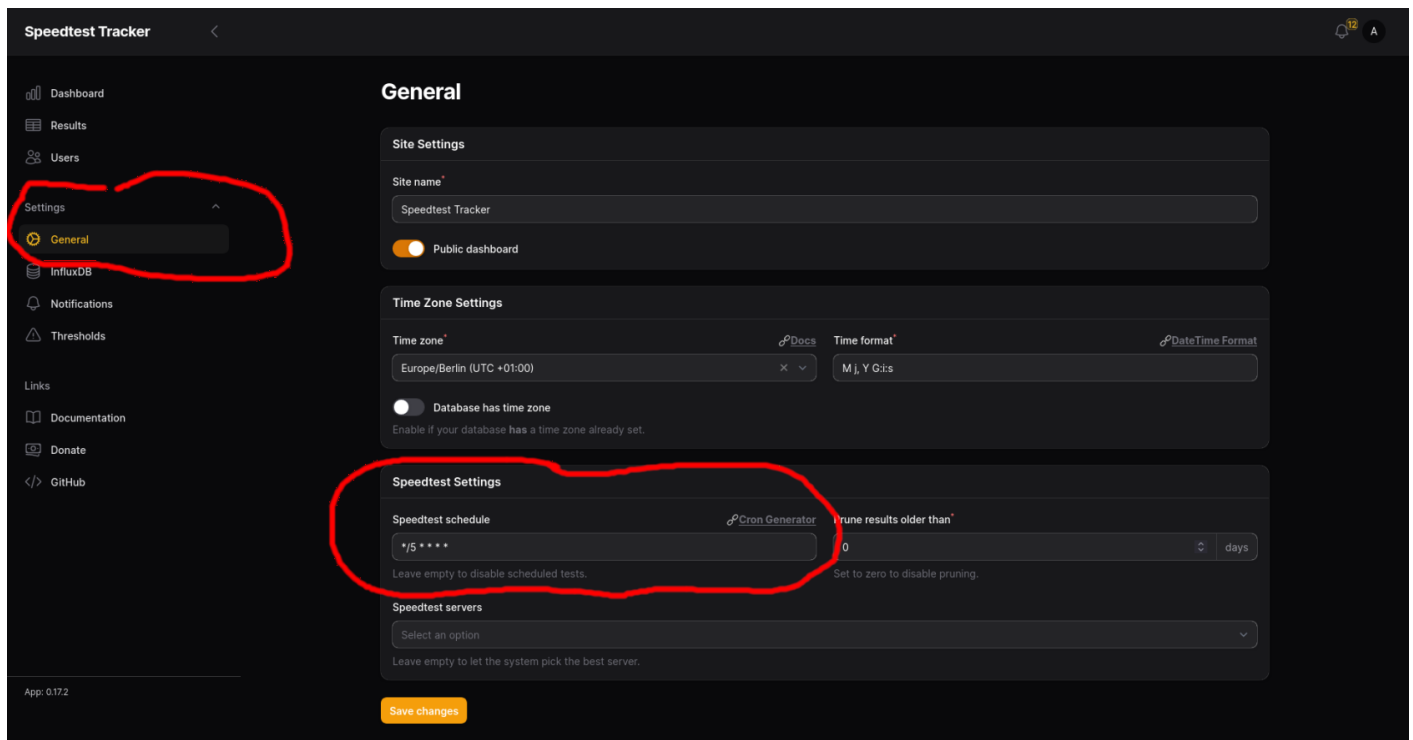


Einige Sekunden gedulden und dann die Seite neu laden. Nun sollten die neuen Messergebnisse vorliegen.



## Automatischer Speedtest

Der manuelle Test ist zwar nett für mal eben zwischendurch, aber deutlich sinnvoller ist natürlich der automatische Test, der z. B. alle 5 Minuten ausgeführt wird. Ein solcher Test lässt sich in den *Settings - General - Speedtest Settings* konfigurieren.



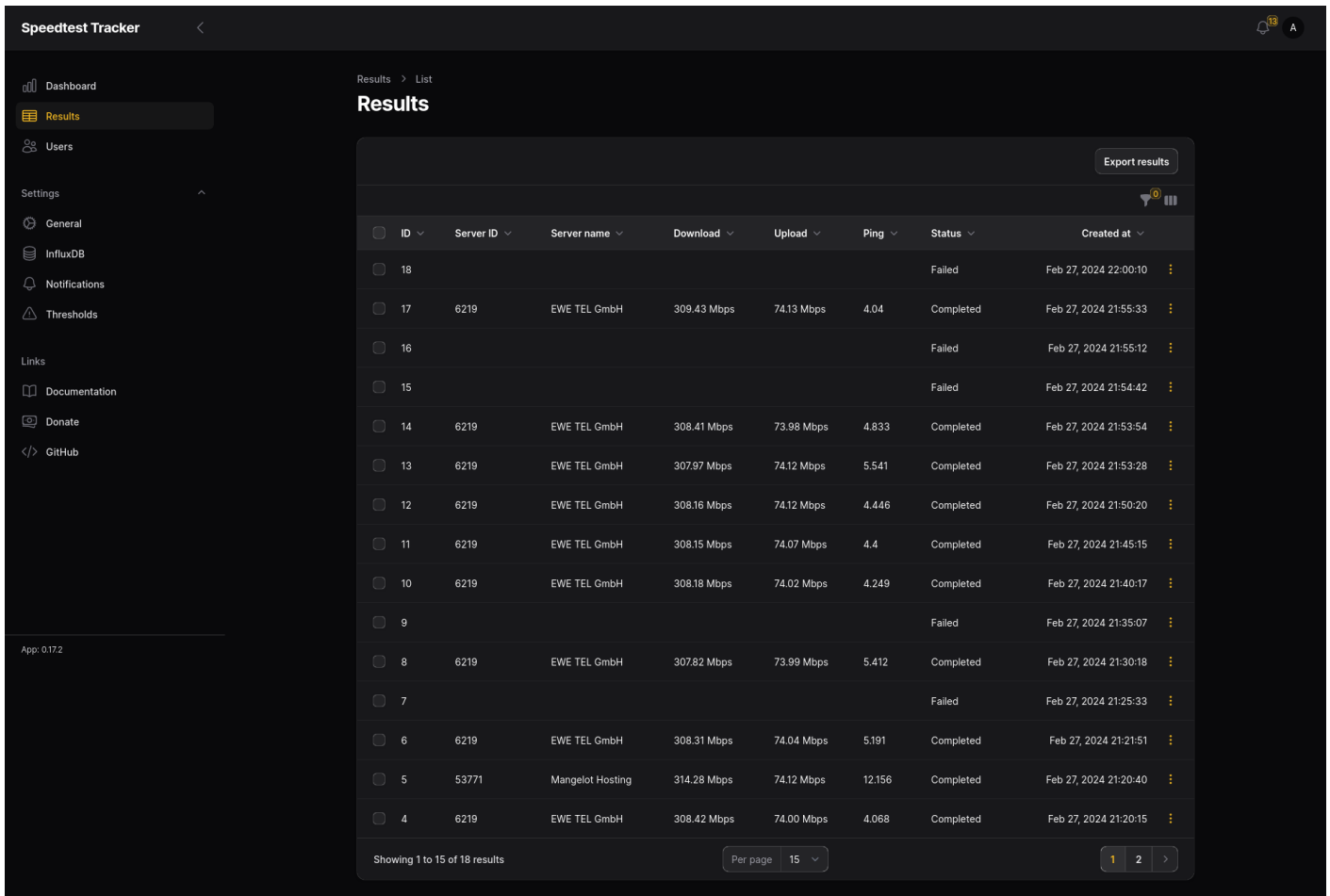
Der *Speedtest Schedule* wird in der Cronjob Notation angegeben. Dies ist ein Linux Standard und lässt sich z. B. mit folgender Webseite erzeugen: <https://crontab.cronhub.io/>

Zum Schluss nur noch auf Speichern klicken und fertig.

Wenn gewünscht kann noch *Prune results older than* mit der Anzahl der Tage eingestellt werden, wie lange die Messergebnisse aufbewahrt werden sollen. Eine 0 heißt für immer.

# Ergebnisse als Tabelle

Die gesammelten Daten können auch als Rohdaten in einer Tabelle angezeigt werden. Dazu einfach im Admin Backen den Menüpunkt *Results* öffnen. Die Daten in der Tabelle können gefiltert und exportiert werden. Auch die Spalten lassen sich anpassen.



The screenshot shows the 'Speedtest Tracker' admin interface. The main content area displays a table of test results under the 'Results' menu. The table has columns for ID, Server ID, Server name, Download, Upload, Ping, Status, and Created at. The data is sorted by ID in descending order. A 'Export results' button is visible in the top right corner of the table area. The interface also includes a sidebar with navigation options like Dashboard, Results, Users, Settings, and Links.

ID	Server ID	Server name	Download	Upload	Ping	Status	Created at
18						Failed	Feb 27, 2024 22:00:10
17	6219	EWE TEL GmbH	309.43 Mbps	74.13 Mbps	4.04	Completed	Feb 27, 2024 21:55:33
16						Failed	Feb 27, 2024 21:55:12
15						Failed	Feb 27, 2024 21:54:42
14	6219	EWE TEL GmbH	308.41 Mbps	73.98 Mbps	4.833	Completed	Feb 27, 2024 21:53:54
13	6219	EWE TEL GmbH	307.97 Mbps	74.12 Mbps	5.541	Completed	Feb 27, 2024 21:53:28
12	6219	EWE TEL GmbH	308.16 Mbps	74.12 Mbps	4.446	Completed	Feb 27, 2024 21:50:20
11	6219	EWE TEL GmbH	308.15 Mbps	74.07 Mbps	4.4	Completed	Feb 27, 2024 21:45:15
10	6219	EWE TEL GmbH	308.18 Mbps	74.02 Mbps	4.249	Completed	Feb 27, 2024 21:40:17
9						Failed	Feb 27, 2024 21:35:07
8	6219	EWE TEL GmbH	307.82 Mbps	73.99 Mbps	5.412	Completed	Feb 27, 2024 21:30:18
7						Failed	Feb 27, 2024 21:25:33
6	6219	EWE TEL GmbH	308.31 Mbps	74.04 Mbps	5.191	Completed	Feb 27, 2024 21:21:51
5	53771	Mangelot Hosting	314.28 Mbps	74.12 Mbps	12.156	Completed	Feb 27, 2024 21:20:40
4	6219	EWE TEL GmbH	308.42 Mbps	74.00 Mbps	4.068	Completed	Feb 27, 2024 21:20:15

Wenn auf *Export results* geklickt wird, können auch die Spalten angepasst werden, die in den Export sollen. Der Export wird im Hintergrund erzeugt und als Benachrichtigung rechts oben bei der Glocke angezeigt, sobald die Generierung abgeschlossen ist.

- Dashboard
- Results
- Users
- Settings
  - General
  - InfluxDB
  - Notifications
  - Thresholds
- Links
  - Documentation
  - Donate
  - GitHub

Results > List

### Results

ID	Server ID	Server name	Download	Upload	Ping	Status
18						Failed
17	6219	EWE TEL GmbH	309.43 Mbps	74.13 Mbps	4.04	Completed
16						Failed
15						Failed
14	6219	EWE TEL GmbH	308.41 Mbps	73.98 Mbps	4.833	Completed
13	6219	EWE TEL GmbH	307.97 Mbps	74.12 Mbps	5.541	Completed
12	6219	EWE TEL GmbH	308.16 Mbps	74.12 Mbps	4.446	Completed
11	6219	EWE TEL GmbH	308.15 Mbps	74.07 Mbps	4.4	Completed
10	6219	EWE TEL GmbH	308.18 Mbps	74.02 Mbps	4.249	Completed
9						Failed
8	6219	EWE TEL GmbH	307.61 Mbps	73.99 Mbps	5.419	Completed

Mark all as read Clear

- Export completed** 5 seconds ago

Your result export has completed and 18 rows exported.

[Download .csv](#)
- Speedtest completed** 5 minutes ago
- Speedtest completed** 9 minutes ago
- Speedtest completed** 9 minutes ago
- Speedtest completed** 10 minutes ago
- Speedtest completed** 11 minutes ago
- Speedtest completed** 12 minutes ago
- Speedtest completed** 15 minutes ago
- Speedtest completed** 20 minutes ago
- Speedtest completed** 25 minutes ago

# Mail Benachrichtigungen

Es können bei jedem Speedtest oder bei Überschreiten eines Limits Mails verschickt werden.

Dazu ist zuerst die Docker Compose Datei anzupassen, folgende Environment Variablen sind hinzuzufügen.

```
environment:
  - MAIL_MAILER=smtp
  - MAIL_HOST=smtp.beispiel.de
  - MAIL_PORT=25/587/465
  - MAIL_USERNAME=speedtest@beispiel.de
  - MAIL_PASSWORD=Tolles-Beispiel-Kennwort
  - MAIL_ENCRYPTION=NULL/SSL/TLS
  - MAIL_FROM_ADDRESS="speedtest@beispiel.de"
  - MAIL_FROM_NAME="Speedtest Tracker"
```

Hier müssen natürlich die Daten vom Mailserver und die Logindaten angepasst werden.

Anschließend den Container mittels `sudo docker-compose up -d` oder `sudo docker compose up -d` neu erstellen lassen und starten.

Danach kann im Admin die Einstellung unter *Settings - Notifications - Mail* konfiguriert werden. Empfehlenswert ist der Trigger *Notify on threshold failures*, damit nicht bei jedem Test eine Mail verschickt wird, sondern nur dann, wenn diese zu schlechte Werte haben.

Die *Thresholds* müssen natürlich auch eingestellt werden, dazu einfach in *Settings - Thresholds* die Option aktivieren und die Grenzwerte eintragen.



Dashboard

Results

Users

Settings



General

InfluxDB

Notifications

Thresholds

Links

Documentation

Donate

GitHub

### Discord

Enable Discord webhook notifications

### Mail

Enable mail notifications

#### Triggers

Notify on every speedtest run

Notify on threshold failures

### Recipients



Email address\*

benutzer1@beispiel.de



Email address\*

benutzer2@beispiel.de

Add to recipients

Test mail channel