# asgard

Der Host **`asgard`** ist der interne Haupserver im deckeigenen Rechenzentrum.

**IP:** 192.168.0.107

## Hostnamen:

- `asgard.maschinendeck.org`
- `tuer.maschinendeck.org`
- `docs.maschinendeck.org`
- `nextcloud.maschinendeck.org`
- `api.maschinendeck.org`
- `nextcloud.maschinendeck.org`
- `social.maschinendeck.org`

# Spezifikation

| | |
|---|---|
| **CPU** | Intel Celeron J4105 CPU, *4 x 1.50GHz* |
| **GPU** | Intel UHD Graphics 600 |
| **RAM** | 8 Gb, DDR4, *2400 MT/s* |
| **Speicher** | 1 Tb, *Intenso SSD SATAIII* |
| **Netzwerk** | 1000BASE-T |
| **OS** | Ubuntu 24.04.1 LTS |

# Aufbau

Der Server läuft vollständig im Dockermodus. Die Konfiguration der Container liegt unter `/config`, deren Daten unter `/data` und etwaige Backups der Datenbanken und der in den Containern laufenden Dienste unter `/backup`.

Die Umgebungsvariablen für einen Container, welche unter Umständen auch Passwörter oder Zugriffstoken enthalten, befinden sich in einer Datei mit dem Namen `.env` in dem jeweiligen Verzeichnis unter `/config`.

# Stack

Ankommende `HTTP`-Anfragen werden von einem traefik Reverseproxy entgegengenommen, welcher diese an die unterliegenden Dockercontainer weiterreicht. Der `traefik`-Container kümmert sich vollautomatisch um die SSL-Zertifikate via *Letsencrypt*.

Zur Konfiguration von `traefik` für einen einzelnen Container, muss die `docker-compose.yml` des Stacks lediglich mit entsprechenden Labels ausgestattet werden. Das Routing auf den exponierten Port geschieht dann automatisch. Hier eine beispielhafte Konfiguration für ein Standardsetup:

```
services:
  example:
    image: example/example
    container_name: example
    env_file: .env
    expose:
      - 80
    networks:
      - public
      - mysql
    labels:
      - traefik.enable=true
      - traefik.http.routers.example.rule=Host(`example.maschinendeck.org`)
      - traefik.http.routers.example.entrypoints=web
      - traefik.http.routers.example_ssl.rule=Host(`example.maschinendeck.org`)
      - traefik.http.routers.example_ssl.tls=true
      - traefik.http.routers.example_ssl.tls.certresolver=letsencrypt
      - traefik.http.routers.example_ssl.entrypoints=websecure
    restart: unless-stopped

networks:
  public:
    external: true
```

Der entsprechende Container muss mindestens dieses Set an Labels aufweisen um funktionieren zu können *(die Strings `example` und `example_ssl` sind für jeden Stack einzigartig zu wählen, müssen also geändert werden)*. Zusätzlich muss er Zugriff auf das Netzwerk `public` erhalten und selbstverständlich muss die entsprechende Domain auf den *DynDNS* des Maschinendeck aufgeschaltet sein.

## Externe Dienste

Auch alle Anfragen an interne Dienste, welche nicht auf dem Server, sondern auf anderen Hosts im internen Netzwerk betrieben werden, landen bei diesem `traefik`-Container. Die Konfiguration für die Handhabung solcher Dienste befindet sich in der Datei `/config/traefik/config/dynamic.yaml`, wo entsprechende Services und Router hinterlegt sind.

Da dort Anfragen in der Regel stupide an einen anderen Host weitergeleitet werden, bestehen diese Konfigurationen aus simplen `loadBalancer`-Direktiven:

```
http:
  services:
    tuer_service:
      loadBalancer:
        servers:
          - url: http://192.168.0.101
  routers:
    tuer:
      service: tuer_service
      entryPoints:
        - web
        - websecure
      rule: Host(`tuer.maschinendeck.org`)
      tls:
        certResolver: letsencrypt
```

Jeweils ein Service und ein Router müssen pro externem Dienst angelegt werden.

# Aktuelle Dienste

Zur Zeit stellt der Host `asgard` folgende Dienste zur Verfügung:

## bookstack

Die Software bookstack stellt die Dokumentation zur Verfügung, die hier gerade zu sehen ist. Jeder der eine @maschinendeck.org E-Mail-Adresse besitzt kann sich hier ein Konto erstellen und an der Dokumentation mitwirken.

## Mastodon

Diese Mastodon-Instanz ist unser Standbein im Fediverse. Alle Maschinisten können einen Account auf Einladung erhalten.

## Nextcloud

Bei Nextcloud handelt es sich um eine Open Source "private cloud"-Lösung, auf welcher Maschinisten Dateien ablegen und Kalender pflegen können.

## Bitwarden

Mit der Open Source Impementierung "*vaultwarden*" unterhält das Maschinendeck hier seine eigene Bitwarden-Instanz. Hierbei handelt es sich um einen Passwortmanager, mit dessen Hilfe Passwörter sicher mit ganzen Personengruppen geteilt werden können.

---

Revision #12
Created 2025-01-31 16:54:48 UTC by MikO
Updated 2025-04-08 11:03:51 UTC by Admin