Infrastruktur-Dokumentation

Die technische Seite der Technik, hauptsächlich intern von der AG Technik genutzt.

Technologieplan

Willkommen in unserem Technologieplan!

Hier sammeln wir zukünftige Pläne und Ideen aus drei Perspektiven in drei Kategorien: aus der Perspektive der **Benutzenden unseres Angebots**, der **Vereinsmitglieder** sowie der **AG Technik** gibt es jeweils, die Softwerke-Infrastruktur betreffend, **Schmerzpunkte** (die einen regelmäßig nerven), **Wünsche** (die ein Problem dieser Perspektive direkt lösen würden) sowie ungenutztes **Potenzial** (also Ideen die das Leben einfacher machen könnten).

Am Anfang jeder Idee steht der Monat, in dem sie dieser Liste hinzugefügt wurde. Die AG Technik wird sich bemühen, sich insbesondere zeitlich an die Reihenfolge der Kategorien und das Alter der Ideen zu halten.

Wenn du einen neuen Punkt für diese Liste hast, schreibe am besten einfach einen Kommentar zu diesem Wiki-Artikel, damit wir die Idee entsprechend ausformulieren und einordnen können.

Perspektive der Benutzenden

Schmerzpunkte

Wünsche

2022/08 Es gibt noch keine Mobilizon-Instanz.

2022/05 Die Webseite ist bisher nur auf Deutsch verfügbar, eine Englische Version oder auch eine Version in leichter Sprache ist sinnvoll.

Potenzial

2022/08 Erstregistrierung bei Vaultwarden erfolgt nicht intuitiv über den Softwerke-Account.
Aktuell braucht man effektiv jemandem der einem zeigt wie es funktioniert.

2021/12 Synchronisation von Nextcloud Circles zu anderen Diensten wäre cool.
Beispielsweise LDAP/Authentik-Gruppen, Vaultwarden-Organisationen, Matrix-Spaces, etc.

Perspektive der Mitglieder

Schmerzpunkte

2022/08 Der Schlüssel für die Vorstands-Mailadresse im WKD ist nicht aktuell. -> Mailserver!

Wünsche

2022/08 Generell sollten verschlüsselte Mails für alle einfach nutzbar sein.

2024/02 Ein Link-Shortener wäre schön für so Dinge wie den Plenums-Links, QR-Codes etc.; es gibt auch schon eine Domain.

2024/02 Eine einfach einzurichtende Spielwiese für Neumitglieder
Das kam aus der Roadmap.

Potenzial

2022/08 Mehr Motivation für Spenden von Nichtmitgliedern.
Über Authentik sollte es möglich sein, einen monatlichen Spendenbetrag zu verwalten. Dieser könnte dann als einfachster Weg genutzt werden, um entsprechend unserer Unkosten automatisch die Limits des Accounts anzupassen.
Zu klären: ist das dann eine geschäftsmäßige Tätigkeit oder dürften wir diesen Betrag auch als zweckungebundene Spende behandeln wenn wir gemeinnützig wären?

Perspektive der AG Technik

Schmerzpunkte

2022/02 Die Dokumentation ist optimierbar.
Das macht es nicht besonders einfach, neue Leute an Bord zu holen

2024/02 Wir haben immer noch keinen ordentlichen Onboarding-Prozess

2024/02 Blackbox nervt, wir sollten auf git-crypt umsteigen.

2024/02 InfluxDB macht immer Probleme mit dem RAM.
Es gibt Alternativen zu Influx, oder wir müssen den Server upgraden. Tunen kann man da wenig.

Wünsche

Potenzial

Infrastruktur der Softwerke

> **Der gesamte Code der für den Betrieb unserer Infrastruktur notwendig ist, ist auf https://codeberg.org/softwerke-magdeburg/infrastructure zu finden.**

Server, DNS & mehr

Unsere Server laufen bei https://netcup.de. Das Passwort zum CCP kann im Passwort-Manager nachgeschlagen werden; das SCP ist nur über das CCP erreichbar.

Domains, DNS und Speicherplatz (als Object Storage) laufen bei https://ovh.de.

CoreOS als Betriebssystem

CoreOS wurde gewählt, weil es sich selbst updated und kaum Wartung benötigt. Für Updates ist ein Neustart notwendig, dafür wurde als Wartungszeitfenster 05:30 bis 06:25 gewählt.

Die Ignition-Konfiguration kann immer nur bei der (Neu)Installation von CoreOS verwendet werden - kleinere Änderungen dürfen manuell erledigt werden, ansonsten ist eine Neuinstallation meist empfehlenswert.

Für das Deployment von CoreOS ist eine `.env`-Datei notwendig (im `coreos`-Verzeichnis dieses Repositories), deren Inhalt im Passwort-Manager zu finden ist.

Mehr Informationen zum Deployment neuer Server ist in der coreos/README.md zu finden.

Docker Swarm + Docker Stack

Docker Swarm kann mehrere Systeme zu einem Cluster zusammenschließen, ist dabei aber deutlich weniger flexibel als Kubernetes.

Docker Stack kann Projekte auf ein Docker-System oder einen Swarm-Cluster deployen, und zwar mit den bekannten Compose-Dateien.

Die Projekte sind im Ordner `services` zu finden und können mit `swarm-make <service>.up clean` gestartet und mit `swarm-make <service>.down clean` gestoppt werden - dafür ist ein erstmaliges Setup erforderlich, wie in der services/README.md beschrieben ist. Ebenfalls ist eine zweite `.env`-Datei erforderlich, die ebenfalls im Passwort-Manager zu finden ist..

Zur Konfiguration der Dienste werden statt Volumes `configs` genutzt, da diese automatisch im Cluster verteilt werden.

Backup-Server

Der Borg Backup-Server ist eine Schnittstelle zwischen dem Internet sowie dem - nur intern erreichbaren - Backup-Speicher bei Scaleway (als Block Storage).

Einrichtung

Erstellung in Scaleway

Der Server wurde mit folgender Konfiguration erstellt:

```
scw instance server create \
  type=STARDUST1-S \
  zone=nl-ams-1 \
  image=rockylinux_8 \
  root-volume=l:10G \
  additional-volumes.0=b:25G \
  name=backup.s.softwerke.md \
  ip=none \
  project-id=869d0909-f443-424a-8a6d-08c21ed750a8
```

Durch diese Konfiguration ist der Server nicht per IPv4 sondern nur per IPv6 erreichbar (das spart etwas über 1 € im Monat).
In der zugehörigen Security Group (namens `Backup`) wird dann ausschließlich SSH-Verkehr aus dem Internet zugelassen und alle anderen eingehenden TCP- und UDP-Verbindungen verworfen.

> Aktuell hängt noch eine IP am Server da IPv6 auf unserem Hauptserver noch nicht wirklich funktioniert.

Einrichtung per Ansible

> Zu beachten ist, dass (Stand Oktober 2021) bei Rocky Linux 8.4 nur OpenSSH 8.0 verfügbar ist, also noch keine Hardware-Schlüssel unterstützt werden.

Der Server kann über Ansible mithilfe des Playbooks im Ordner `backup-server` des Infrastruktur-Repos eingerichtet werden - dies erfolgt mit dem folgenden Befehl:

```
ansible-playbook playbook.yml
```

Vergrößerung des Speicherplatzes

Wenn der Block Storage vergrößert wird, kann die Partitionstabelle sowie das Dateisystem mit folgendem Befehl vergrößert werden:

```
{ printf 'w\nY\nY\n' | gdisk /dev/sda; } && \
{ printf "d\nn\n$(cat /sys/block/sda/sda1/partition)\n$(cat
/sys/block/sda/sda1/start)\n\n8300\nw\nY\n" | gdisk /dev/sda; } && \
partprobe && \
resize2fs /dev/sda1
```

Backups & Borg-Repositories

Repository erstellen

Ein verschlüsseltes Backup-Repository wird (auf dem jeweiligen Quellserver) wiefolgt erstellt:

```
borg init \
  --encryption authenticated-blake2 \
  --append-only \
  backup@backup.s.softwerke.md:/mnt/REPONAME
```

Backup manuell erstellen

Ein Backup kann dann folgendermaßen durchgeführt werden:

```
BORG_PASSPHRASE='...' \
  borg create \
  --progress \
  "backup@backup.s.softwerke.md:/mnt/REPONAME::{now:%Y-%m-%d--%H-%M-%S}" \
  /var/lib/docker/volumes
```

Automatische Backups per systemd-Timer

`/etc/systemd/system/backup.service`

```
[Unit]
Description=Backup
After=syslog.target network.target

[Service]
Type=oneshot
ExecStart=/usr/bin/borg create --progress
"backup@backup.s.softwerke.md:/mnt/REPONAME::{now:%%Y-%%m-%%d--%%H-%%M-%%S}"
/var/lib/docker/volumes
Environment=BORG_PASSPHRASE=...
```

`/etc/systemd/system/backup.timer`

```
[Unit]
Description=Run backup every 3 hours

[Timer]
OnCalendar=*-*-* 00,03,06,09,12,15,18,21:00:00

[Install]
WantedBy=timers.target
```

Monitoring

TODO: wir sollten (jeweils mit Benachrichtigung) prüfen ob einerseits im Repo die Backups ankommen, und andererseits der "borg create"-Befehl mit erfolgreichem Exit-Code abschließt.

Speicher-Server

Der Speicher-Server stellt die NextCloud zur Verfügung - dafür würde sich zwar theoretisch auch ein Docker-Container mit der NextCloud-S3-Anbindung eignen, letztere ist jedoch unglaublich langsam. Unsere Zwischenlösung ist darum ein RAID-0 aus Block Storage Volumes, wodurch ausschließlich vertikale Skalierung möglich ist. Langfristig ist die NextCloud wohl der Dienst wo sich am ehesten die Anschaffung eigener Hardware in Kombination mit einem GlusterFS-Dateisystem lohnt, hoffentlich ist bis dahin aber OwnCloud Infinite Scale an einem Punkt angekommen, wo das obsolet ist.

Einrichtung

Erstellung in Scaleway

Der Server wurde mit folgender Konfiguration erstellt:

```
scw instance server create \
  type=DEV1-S \
  zone=fr-par-2 \
  image=rockylinux_8 \
  root-volume=l:20G \
  name=speicher.s.softwerke.md \
  ip=new \
  project-id=473e8c4a-83a4-41f6-a7de-9dd6aa2a3247
```

Einrichtung per Ansible

> Zu beachten ist, dass (Stand Oktober 2021) bei Rocky Linux 8.4 nur OpenSSH 8.0 verfügbar ist, also noch keine Hardware-Schlüssel unterstützt werden.

Der Server kann über Ansible mithilfe des Playbooks im Ordner `speicher-server` des Infrastruktur-Repos eingerichtet werden - dies erfolgt mit dem folgenden Befehl:

```
ansible-playbook playbook.yml
```

Mail-Infrastruktur

// TODO

Postfix

- MTA
- Senden und empfangen von Mails
  - routing, rewriting, restrictions und bedingungen

Rspamd

- spamfilter

Dovecot

- MDA
- Mailboxen

LDAP

DNS

Momentan haben wir folgende DNS-Zonen

magdeburg.jetzt, softwerke.md

für die Erreichbarkeit von Diensten

s.softwerke.md

für Server

swarm.softwerke.md

> Docker Swarm ist bei uns legacy und wird mit dem Server-Umzug abgelöst

für Docker Swarm