

monfetch

Server-Monitoring im r/unixporn-Style

emily

August 2021

- Server-Monitoring meist sehr komplex & aufwändig

- Server-Monitoring meist sehr komplex & aufwändig
- Erfordert viele Ressourcen:
 - Datenbankcluster
 - Data Sources
 - Datenaufbereiter
 - etc.

- Server-Monitoring meist sehr komplex & aufwändig
- Erfordert viele Ressourcen:
 - Datenbankcluster
 - Data Sources
 - Datenaufbereiter
 - etc.
- Meist Cloud-gebunden

- Server-Monitoring meist sehr komplex & aufwändig
- Erfordert viele Ressourcen:
 - Datenbankcluster
 - Data Sources
 - Datenaufbereiter
 - etc.
- Meist Cloud-gebunden
- Datensammler meist sehr schwergewichtig

- Server-Monitoring meist sehr komplex & aufwändig
- Erfordert viele Ressourcen:
 - Datenbankcluster
 - Data Sources
 - Datenaufbereiter
 - etc.
- Meist Cloud-gebunden
- Datensammler meist sehr schwergewichtig
- Für Enterprise gut, für jedoch Homelab massiver Overkill

Was unterscheidet monfetch davon?

- Sehr simple Architektur
 - Eigener Server/Client: Wochenendsprojekt

Was unterscheidet monfetch davon?

- Sehr simple Architektur
 - Eigener Server/Client: Wochenendsprojekt
- Keine Datenbank benötigt

Was unterscheidet monfetch davon?

- Sehr simple Architektur
 - Eigener Server/Client: Wochenendsprojekt
- Keine Datenbank benötigt
- Keine Cloud-Anbindung

Was unterscheidet monfetch davon?

- Sehr simple Architektur
 - Eigener Server/Client: Wochenendsprojekt
- Keine Datenbank benötigt
- Keine Cloud-Anbindung
- Datensammler (Agent) sehr leichtgewichtig

Was unterscheidet monfetch davon?

- Sehr simple Architektur
 - Eigener Server/Client: Wochenendsprojekt
- Keine Datenbank benötigt
- Keine Cloud-Anbindung
- Datensammler (Agent) sehr leichtgewichtig
- Bonus: Server auch sehr leichtgewichtig

- Generierte Seite sehr klein
 - HTML: 25 Zeilen
 - CSS: 89 Zeilen (nicht minified)
 - kein Javascript

- Generierte Seite sehr klein
 - HTML: 25 Zeilen
 - CSS: 89 Zeilen (nicht minified)
 - kein Javascript
- Sehr übersichtlich
 - Daten im fetch-Stil, daher sehr leicht lesbar für *NIX-Nutzende

- Generierte Seite sehr klein
 - HTML: 25 Zeilen
 - CSS: 89 Zeilen (nicht minified)
 - kein Javascript
- Sehr übersichtlich
 - Daten im fetch-Stil, daher sehr leicht lesbar für *NIX-Nutzende
- Freie Software: AGPL-3.0-or-later

- Server in Python3 (Flask), Agent in Shell

- Server in Python3 (Flask), Agent in Shell
- Agent sendet Daten mit cURL per POST

- Server in Python3 (Flask), Agent in Shell
- Agent sendet Daten mit cURL per POST
- Daten größtenteils aus procfs/sysfs

- Server in Python3 (Flask), Agent in Shell
- Agent sendet Daten mit cURL per POST
- Daten größtenteils aus procfs/sysfs
- Authentifizierung per Shared Secret in Request Path

- WSGI-Server: uWSGI
 - Handlet Anfragen
 - Leichtgewichtig

- WSGI-Server: uWSGI
 - Handlet Anfragen
 - Leichtgewichtig
- systemd
 - Supervision
 - Nicht sonderlich leichtgewichtig, aber einfach ersetzbar

- WSGI-Server: uWSGI
 - Handlet Anfragen
 - Leichtgewichtig
- systemd
 - Supervision
 - Nicht sonderlich leichtgewichtig, aber einfach ersetzbar
- git
 - Versionsverwaltung
 - Macht Aktualisierung einfach
 - Recht leichtgewichtig

- Echtes, laufendes Beispiel

- Echtes, laufendes Beispiel
- Installation Server mit uWSGI

- Echtes, laufendes Beispiel
- Installation Server mit uWSGI
- Installation Client

- Echtes, laufendes Beispiel
- Installation Server mit uWSGI
- Installation Client
- Aufsetzen von systemd-Services

Gibt es noch Fragen?