

# Bist du da?

## Webdienste mit Uptime Kuma überwachen

**Egal, ob Sie Webdienste professionell im Unternehmen hosten oder als Hobby-Admin für Freunde und Familie bereitstellen: Wenn die Webseite offline ist, gibt es lange Gesichter. Uptime Kuma hilft Ihnen zu reagieren, bevor die erste Beschwerde reinflattert. Verschaffen Sie sich mit unserer Anleitung einen Vorsprung.**

Von Niklas Dierking

**U**ptime Kuma ist ein praktisches und leichtgewichtiges Open-Source-Werkzeug, das Ihre Webserver und darauf laufende Webanwendungen in regelmäßigen Abständen auf Erreichbarkeit prüft. Bei Misserfolg benachrichtigt Sie das Programm über einen der zahlreichen Kanäle wie E-Mail, Telegram, Slack, Discord oder ruft eine konfigurierte Adresse per HTTP auf (Webhook). Nach der Installation von Uptime Kuma müssen Sie keine weiteren Ausflüge auf die Kommandozeile mehr unternehmen, um sich durch Konfigurationsdateien zu wühlen. Sie konfigurieren alles über eine intuitiv bedienbare und schicke Weboberfläche.

Damit alle Nutzer Ihrer Dienste deren Funktionsfähigkeit auf einen Blick erfassen

sen können, generiert Uptime Kuma eine öffentliche Statusseite. Dort können Sie als Administrator Hintergrundinformationen zur Störung bereitstellen, beispielsweise, wann mit der Lösung des Problems zu rechnen ist. Mehr Informationen zu Statusseiten und wie Sie diese mit cState und dem Static-Site-Generator Hugo erstellen, haben wir kürzlich in einem c't-Artikel festgehalten [1]. Neben Uptime-Statistiken stellt die Software noch viele weitere nützliche Informationen bereit und protokolliert etwa die Antwortzeiten der Server sowie den Status von Zertifikaten. Uptime Kuma lässt sich schnell in Betrieb nehmen und ist in Deutsch, Englisch und viele weitere Sprachen übersetzt.

### Installation

Weil der Entwickler Uptime Kuma in einen Docker-Container verpackt hat, können

Sie die Software auf einem Windows-, macOS- oder Linux-Host mit Docker installieren. In dieser Anleitung kommt Ubuntu 20.04 LTS zum Einsatz.

Bedenken Sie, dass Ihnen auch das beste Monitoring-Werkzeug nichts nützt, wenn das System ausfällt, auf dem Sie es betreiben. Sie sollten Uptime Kuma deswegen

Frisch aus  
**c't Nerdistan**

nicht mit dem gleichen Host in Betrieb nehmen, auf dem die Dienste laufen, die Sie überwachen wollen. Installieren Sie Uptime Kuma stattdessen lieber auf einem System, das vom Rest Ihrer Infrastruktur möglichst unabhängig ist. Je nach Einsatzumgebung kann diese Aufgabe beispielsweise die Minimalausführung eines angemieteten vServers oder ein Raspberry Pi erfüllen.

Der einfachste und schnellste Weg Uptime Kuma zu installieren, führt über Docker und Docker-Compose. Eine einsteigerfreundliche Einführung in die Container-Software lesen Sie in [2]. Legen Sie zunächst in einem Verzeichnis Ihrer Wahl einen Ordner namens uptime-kuma an und wechseln Sie dann dorthin:

```
mkdir uptime-kuma
cd uptime-kuma
```

Jetzt erstellen Sie mit einem Texteditor eine YAML-Datei namens docker-compose.yml. Darin ist definiert, welches Container-Image zum Einsatz kommt und welcher TCP-Port des Containers dem TCP-Port des Hostsystems entspricht. Über die IP-Adresse des Host und den Port erreichen Sie mit einem Browser nach der Installation die Weboberfläche von Uptime Kuma. Außerdem legen Sie in der Datei ein Verzeichnis fest, das als Volume in den Container gereicht wird. Dort speichert Uptime Kuma seine SQLite-Datenbank. Passen Sie die Werte für ports und volumes an Ihre eigene Umgebung an:

```
version: '3.3'
services:
  uptime-kuma:
    image: louislam/uptime-kuma:1
    container_name: uptime-kuma
    volumes:
      - ./uptime-kuma:/app/data
    ports:
      - 3001:3001
```

Mit dieser Konfiguration können Sie Uptime Kuma in einem Netzwerk betreiben, das nicht aus dem Internet erreichbar ist. Wenn Sie das Monitoring-Tool ins Netz hängen wollen, dann sollten Sie es im Verbund mit einem Reverse-Proxy betreiben, der sich bestenfalls auch automatisch um die Beschaffung von TLS-Zertifikaten kümmert. Eine Docker-Compose-Vorlage mit dem vielseitigen Edge-Router Traefik stellen wir in einem GitHub-Repository zur Verfügung, das wir unter [ct.de/y96q](https://github.com/ctde/y96q)

verlinkt haben. Starten Sie den Container mit dem Befehl `docker-compose up -d`.

Rufen Sie jetzt mit einem Browser die Weboberfläche von Uptime Kuma auf, indem Sie die IP-Adresse des Host und den Port 3001 ansteuern, beispielsweise 192.168.0.82:3001. Registrieren Sie dann einen Administrator-Account und vergeben Sie ein sicheres Passwort. Um Uptime Kuma besser zu sichern, sollten Sie später in den Einstellungen eine Zwei-Faktor-Authentifizierung mittels TOTP (Time-based One-time Password) konfigurieren.

### Fühler ausstrecken

Standardmäßig präsentiert Uptime Kuma Ihnen das Dashboard als Startseite. Der regelmäßige Check eines Servers heißt in Uptime Kuma „Monitor“. Wenn Sie noch keinen Monitor hinzugefügt haben, herrscht im Dashboard gähnende Leere. Weisen Sie Uptime Kuma also an, einen Webserver oder Webdienst zu beobachten, indem Sie über die Schaltfläche „Neuer Monitor“ in der oberen linken Ecke des Fensters einen neuen Monitor hinzufügen. Uptime Kuma hat diverse Arten von Monitoren im Gepäck, um unterschiedliche Anwendungsfälle abzudecken.

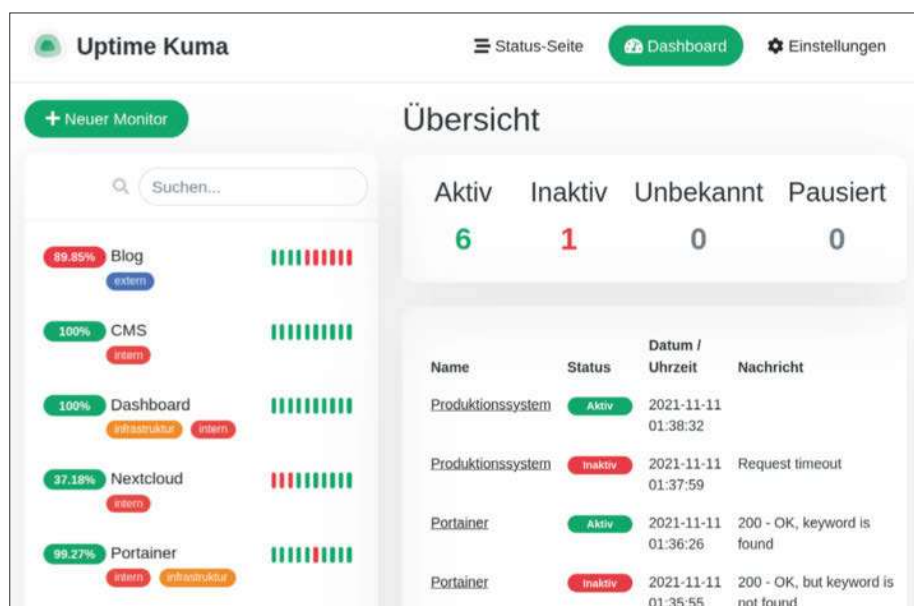
Beim Standard-Monitor handelt es sich um eine gewöhnliche HTTP(S)-Abfrage. Vergeben Sie dafür unter „Anzeigenamen“ einen Namen für den Monitor und tragen Sie die URL des Ziels ein, beispielsweise `https://ct.de`. Im Punkt „Prüfintervall“ bestimmen Sie, in welchen zeitlichen Abständen Uptime Kuma Anfragen sen-

## ct kompakt

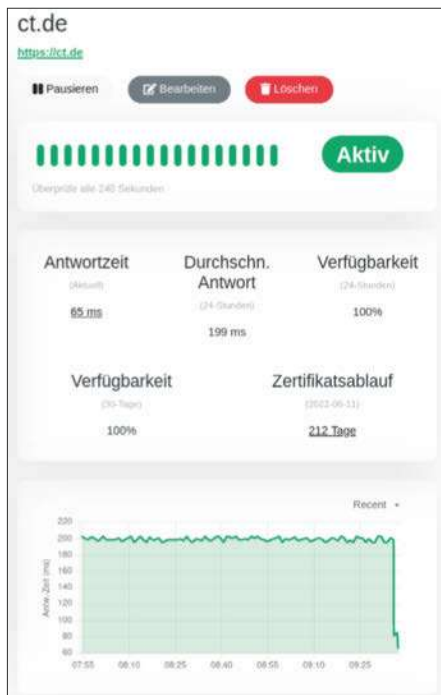
- Uptime Kuma fragt in regelmäßigen Intervallen Server im Netz an und registriert Ausfälle.
- Die Monitoring-Software benachrichtigt Administratoren und Nutzer via E-Mail, Messenger und SMS.
- Eine Statusseite informiert öffentlich, ob alle Server, Webseiten und Web-Apps online sind.

den soll. Voreingestellt ist ein Request alle 60 Sekunden. Jede erfolgreiche Anfrage markiert Uptime Kuma im Dashboard neben den Monitoren durch einen grünen Zapfen. Misserfolge zeigt es als roten Zapfen an. Erfolge und Misserfolge werden gemeinsam in der sogenannten Heartbeat-Leiste visualisiert.

Sie können bei dem Punkt „Wiederholungen“ auch bestimmen, dass Uptime Kuma einen Dienst beispielsweise erst nach drei gescheiterten Anfragen als inaktiv markiert. Die Zeitspanne zwischen diesen Wiederholungsanfragen können Sie unabhängig von den gewöhnlichen Anfragen einstellen. In den erweiterten Monitor-Einstellungen legen Sie fest, ob das Tool einen Check als Erfolg oder Misserfolg wertet, wenn der Server mit Zertifikatsfehlern antwortet. Je nach Umgebung kann es sinnvoll sein, einen Zertifikatsfehler nicht als Miss-



Das Dashboard ist das Herzstück der Weboberfläche von Uptime Kuma. Es zeigt die sogenannte Heartbeat-Leiste aller Monitore und die letzten Ereignismeldungen an.



**In den Monitor-Details stellt Uptime Kuma nützliche Zusatzinformationen zu Zertifikaten und Antwortzeiten des Servers bereit.**

erfolg zu werten, etwa wenn Sie im Heimnetzwerk mit selbst signierten Zertifikaten arbeiten. Für besondere Fälle taugt der „Invertierte Modus“: Uptime Kuma warnt dann, wenn die Anfrage erfolgreich ist. Das hilft Ihnen, wenn Sie sicherstellen wollen, dass sensible Daten nicht plötzlich durch einen Konfigurationsfehler unbemerkt im Netz landen.

Unter „Erlaubte HTTP-Statuscodes“ legen Sie fest, ob es für einen erfolgreichen Check genügt, wenn der Webserver mit bestimmten HTTP-Statuscodes antwortet. Voreingestellt sind die Statuscodes 200 bis 299, die anzeigen, dass eine Operation erfolgreich war. Wenn Sie Ihre eigene Webseite abfragen, dann könnten Sie dort einen Health-Check-Endpoint wie `example.com/health` konfigurieren, der auf Anfragen ausschließlich mit dem HTTP-Statuscode „200: OK“ antwortet. Tragen Sie den Endpoint in Ihrem Monitor als Ziel ein, um Ressourcen und Bandbreite gegenüber einer vollständigen HTTP(S)-Anfrage zu sparen.

Wenn Sie dagegen beispielsweise prüfen wollen, ob Ihre Webseite auch HTML rendert, dann können Sie den HTTP(S)-Monitor von Uptime Kuma um ein Suchwort erweitern. Das Tool durchsucht dann die HTML- oder JSON-Ausgabe nach dem Wort. Um falschen Alarm zu vermeiden,

sollten Sie ein Suchwort wählen, das stets fester Bestandteil der Webseite ist, etwa „Newsticker“ auf `heise.de`.

Schlussendlich können Sie noch Tags vergeben, um Monitore nach bestimmten Kriterien zu markieren, beispielsweise um interne und externe Dienste zu unterscheiden. So bleibt Ihr Dashboard auch mit vielen Monitoren übersichtlich.

## Werkzeugkoffer

Neben der HTTP(S)-Abfrage bringt Uptime Kuma noch weitere Monitore mit, die Sie für spezielle Anwendungszwecke nutzen oder auch kombinieren können, um sich ein umfangreiches Lagebild zu verschaffen. Kontrollieren Sie vom gleichen Ziel neben HTTP(S) auch relevante TCP-Ports und den DNS-Server, beschleunigt das im Falle einer Störung die Fehlersuche.

Mit dem Monitor „TCP-Port“ weisen Sie Uptime Kuma an, einen bestimmten TCP-Port im Auge zu behalten, etwa um zu prüfen, ob Ihr Webserver oder Reverse-Proxy auf Port 80 oder 443 ansprechbar ist, oder weil Ihre Webanwendung auf einem exotischen TCP-Port lauscht. Dazu müssen Sie lediglich Hostname oder IP-Adresse und den gewünschten Port eintragen. Dieser Monitor ist auch nützlich, um sicherzustellen, dass Wartungszugänge zu Servern hinter einer Firewall nicht öffentlich erreichbar sind. Tragen Sie dazu den Hostname oder die IP-Adresse des Servers und den für SSH-Verbindungen üblichen Port 22 ein und setzen Sie das

**Ein Benachrichtigungsdienst ist schnell eingerichtet. Falls es doch hakt, helfen die Links zu Anleitungen der Anbieter.**

Häkchen bei „Invertierter Modus“. Sollte es zu einem Problem mit der Firewall kommen und der Monitor eine Verbindung mit dem TCP-Port 22 herstellen können, schlägt Uptime Kuma direkt Alarm.

Der Monitor „Push“ dreht den Spieß um, indem der zu beobachtende Server sich bei Uptime Kuma meldet und damit seine Anwesenheit bestätigt. Mit diesem Monitor kontrollieren Sie, ob ein Server online ist, der sich etwa in einem geschlossenen Managementsegment eines Netzwerks befindet und von Uptime Kuma nicht kontaktiert werden kann. Damit das Tool die Anfrage des Servers auch dem korrekten Monitor zuordnen kann, generiert es für jeden „Push“-Monitor, den Sie anlegen, eine einzigartige Push-URL, die auf einen Health-Check-Endpoint verweist:

```
https://uptime.example.com/api/push/
🔗DYTY7gQxx?msg=OK&ping=
```

Wenn Sie mit dem Kommandozeilenwerkzeug `curl` die Daten von der Webseite abschöpfen, quittiert Ihnen Uptime Kuma die erfolgreiche Anfrage und registriert einen Erfolg:

```
curl https://kuma.example.com/api/
🔗push/DYTY7gQxx?msg=OK&ping=
{"ok":true}
```

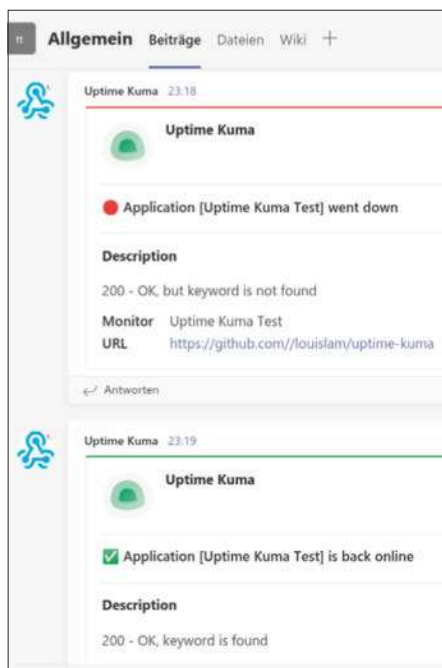
Der Push-Monitor erfüllt natürlich nicht seinen Zweck, wenn Sie mit dem Server händisch Meldung bei Uptime Kuma machen. Legen Sie stattdessen lieber einen Cron-Job an, der innerhalb des Prüfintervalls tätig wird, das Sie für den Push-Monitor konfiguriert haben, beispielsweise alle zehn Minuten. Um auf der Festplatte des Servers keinen Datenmüll zu sammeln, können Sie die den Health-Check-Endpoint auch mit `wget` abrufen und nach `/dev/null` umleiten:

```
wget -O /dev/null -o /dev/null
🔗https://uptime.example.com/api/
🔗push/DYTY7gQxx?msg=OK&ping=
```

Der Monitor „Ping“ sendet genau wie das gleichnamige Kommandozeilenwerkzeug ein kleines Datenpaket und misst die Antwortzeit des Servers. Dafür tragen Sie lediglich den Hostnamen oder die IP-Adresse des Servers ein.

Mit dem Monitor „DNS“ prüft Uptime Kuma die DNS-Einträge von Diensten, die Sie im Auge behalten wollen. Das Tool benötigt dafür den Hostnamen, die IP-





### Das konfigurierte Suchwort wird nicht gefunden: Uptime Kuma schlägt Alarm und informiert via Microsoft Teams.

Adresse des auflösenden DNS-Servers und die Art des DNS-Eintrags, in der Regel einen A-Record, der auf eine IPv4-Adresse zeigt. Als Standard ist der Auflösungs-Server von Cloudflare mit der IP-Adresse 1.1.1.1 eingetragen. Diesen können Sie durch einen DNS-Server Ihrer Wahl ersetzen. Alternativ können Sie auch nicht-öffentliche Hostnamen und Ihren privaten DNS-Server eintragen, etwa wenn ein Pi-hole in Ihrem Heimnetz die Namensauflösung regelt. Einzige Voraussetzung ist, dass Uptime Kuma diesen Server auch erreichen kann.

### Gute schlechte Nachrichten

Beim Ausfall eines kritischen Dienstes ist in der Regel Eile geboten. Konfigurieren Sie deswegen einen der zahlreichen Benachrichtigungsdienste, damit Uptime Kuma Sie informiert, sobald ein Monitor einen Ausfall registriert. Die Schaltfläche „Benachrichtigung einrichten“ finden Sie unter „Einstellungen“ oder im Monitor-Menü. Zur Verfügung stehen E-Mail via SMTP, diverse Chat-Dienste und Instant-Messenger wie Signal, Telegram, Discord, Teams sowie kommerzielle Anbieter von SMS-Benachrichtigungen.

Als Beispiel für Benachrichtigungen via Messenger dient hier die Configuration eines Webhooks für Microsoft Teams. Sie brauchen dazu einen Microsoft-Account. Die übrigen Messenger-Benachrichti-

gungsdienste richten Sie ähnlich ein. In der Regel benötigen Sie einen API-Key, eine Chat-ID oder eine Webhook-URL. Praktisch: Uptime Kuma verlinkt Ihnen im Konfigurationsfenster der Benachrichtigungen Hinweise der Anbieter, die erklären, wie Sie den jeweiligen Chat-Bot oder die API-Schnittstelle einrichten.

Steuern Sie in einem Teams-Kanal, in dem Sie die nötigen Rechte besitzen, die Schaltfläche „Connectors“ in den Kanaleinstellungen an. Weil Sie Informationen an Teams senden wollen, wählen Sie anschließend die Option „Incoming Webhook“. Sie müssen dem Webhook lediglich einen Namen geben, der als Sender angezeigt werden soll und „Erstellen“ klicken. Teams generiert dann eine Webhook-URL. Die dient als öffentlicher Endpunkt, an den Uptime Kuma Daten sendet, beispielsweise:

```
https://example.webhook.office.com/wehookb2/3e3563611-143ad-1b8f-9d8hf2
```

Die URL enthält einen langen, zufälligen String, der gleichzeitig als Geheimnis fungiert. Weil theoretisch jeder im Besitz der URL Daten an den Webhook senden kann, sollten Sie die Adresse wie ein Passwort behandeln. Wechseln Sie schlussendlich zu Uptime Kuma und tragen Sie die Webhook-URL für die Benachrichtigung ein. Sie können den Benachrichtigungsdienst jetzt einzelnen Monitoren zuordnen oder standardmäßig aktivieren. Uptime Kuma wird Sie ab sofort via Teams benachrichtigen, wenn der Check eines konfigurierten Monitors misslingt. Ebenso informiert es Sie, sobald der Dienst wieder erreichbar ist. Alternativ können Sie die Benachrichti-


tigungsdienste auch nutzen, um nicht nur sich selbst als Administrator, sondern auch die Nutzer Ihrer Dienste auf dem Laufenden zu halten.

### Alles auf einen Blick

Um Informationen über die Erreichbarkeit Ihrer Dienste einfach und übersichtlich mit Nutzern zu teilen, stellt Uptime Kuma zusätzlich eine öffentliche Statusseite unter `example.com/status` bereit. Die Statusseite richten Sie über die Weboberfläche ein. Legen Sie nach Bedarf Gruppen von Webanwendungen an und sortieren Sie Gruppen und Monitore mittels Drag & Drop arrangieren, ein eigenes Icon hochladen und die Überschrift der Statusseite anpassen. Die Interna der Monitore wie abgefragte URLs, IP-Adressen und TCP-Ports sind auf der Statusseite nicht sichtbar. Wenn Sie Ihren Monitoren Tags zugewiesen haben, sind diese standardmäßig ausgeblendet. Sie können sie optional für Besucher anzeigen lassen.

Registriert Uptime Kuma eine gescheiterte Anfrage, generiert es automatisch eine Benachrichtigungstafel, etwa mit der Botschaft „Teilweise beeinträchtigter Dienst“. Über die Schaltfläche „Create Incident“ pinnen Sie eine eigene Statusmeldung an, um Nutzer über ein aktuelles Problem zu informieren. Abhängig vom Charakter der Meldung können Sie diese mit farblichen Stilen wie „info“, „warning“ und „danger“ unterstreichen.

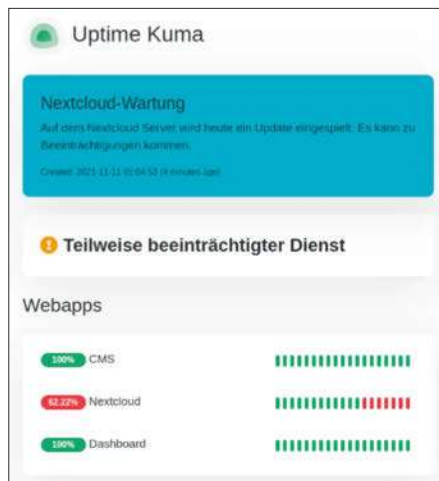
### Fazit

Uptime Kuma ist schnell installiert und hat jede Menge Bordmittel, um Webserver und Webanwendungen im Auge zu behalten. Durch die unterschiedlichen Monitor-Typen passen Sie das Monitoring-Werkzeug auch an spezielle Anwendungsfälle an. Dank intuitiv bedienbarer Weboberfläche geht die Konfiguration leicht von der Hand. Sobald ein Monitor einen Ausfall registriert, benachrichtigt die Software Sie zuverlässig, damit Sie Maßnahmen ergreifen können, um Ihre Dienste flott wieder ans Netz zu bekommen. (ndi@ct.de) 

### Literatur

- [1] Jan Mahn, Schöner Scheitern, Statusseiten für IT-Infrastruktur mit cState generieren, c't 12/2021, S. 156
- [2] Jan Mahn, Zu neuen Ufern, Nach dem Hype: Docker verstehen und loslegen, c't 24/2021, S. 146

**GitHub-Repository mit Docker-Compose-Vorlage:** [ct.de/y96g](https://github.com/y96g)



**Mit einem Blick auf die Statusseite erfahren die Nutzer den Status wichtiger Webdienste.**