Faultnotify □□

> **Faultnotify, Debian (+Derivate)**
> ***Wiki-Stand: 10.04.2026***
> ***Script-Stand: 10.04.2026***
>
> *Sprachdateien-Referenz: 24.01.2026 (mehr...)*

Download

□□ Vorwort

Faultnotify ist ein leichtgewichtiges, modular aufgebautes Bash-Script zur Überwachung von Geräten und Diensten.
Es wurde mit dem Ziel entwickelt, Störungen **zuverlässig und frühzeitig** zu erkennen. Dazu überprüft Faultnotify Geräte und Dienste fortlaufend und informiert im Fehlerfall automatisch per Benachrichtigung.

Alle benötigten Daten werden zentral in einem festen Verzeichnis abgelegt. Dort befinden sich unter anderem die Konfiguration, Statusinformationen sowie Steuer- und optional auch Sprachdateien. Dadurch bleibt alles übersichtlich an einem Ort.

Optional kann Faultnotify um zusätzliche Sprachdateien erweitert werden. In diesem Fall ist das Script in der Lage, Benachrichtigungen und Ausgaben in der jeweiligen Sprache bereitzustellen.

Faultnotify eignet sich sowohl für einzelne Systeme als auch für größere, verteilte Umgebungen. Es kommt bewusst ohne zusätzliche Software oder Datenbanken aus und bleibt dadurch einfach, robust und wartungsarm.

In verteilten Umgebungen können Geräte und Dienste zu Gruppen zusammengefasst werden. So wird sichergestellt, dass im Störungsfall nur gezielte und sinnvolle Benachrichtigungen ausgelöst werden.

Für den Dauerbetrieb empfiehlt sich die Einrichtung als Systemdienst. Dadurch läuft Faultnotify permanent im Hintergrund und bleibt auch nach Neustarts zuverlässig aktiv.

□□ Allgemein und Installation

Allgemein

Faultnotify arbeitet in einem zentralen Arbeitsverzeichnis unter:

/home/$USER/script-data/faultnotify/

In diesem Verzeichnis werden alle relevanten Dateien abgelegt, darunter die Konfiguration, Status- und Steuerdateien sowie Protokolle. Dadurch bleiben alle Informationen übersichtlich an einem Ort.

Optional kann Faultnotify um Sprachdateien erweitert werden. Diese werden im folgenden Verzeichnis abgelegt:

/home/$USER/script-data/faultnotify/lang/faultnotify-xxx.txt

Deutsch ist die im Script integrierte Standardsprache. Weitere Sprachen können über entsprechende Sprachdateien ergänzt werden. Sprachen können im Downloadbereich heruntergeladen werden.

Installation und Ersteinrichtung

Faultnotify wird über den Befehl **install** eingerichtet. Das Script führt dabei interaktiv durch die Ersteinrichtung und legt alle grundlegenden Einstellungen an.

Während der Installation prüft Faultnotify, ob alle benötigten Programme auf dem System vorhanden sind. Fehlende Abhängigkeiten werden klar angezeigt und müssen anschließend manuell nachinstalliert werden.

Im nächsten Schritt wird der gewünschte Benachrichtigungsweg festgelegt. Zur Auswahl stehen **E-Mail** oder **Telegram**. Je nach Auswahl werden die erforderlichen Angaben abgefragt und in der Konfiguration gespeichert.

Zum Abschluss der Einrichtung wird ein Verifizierungscode an die angegebene E-Mail-Adresse oder an den ausgewählten Telegram-Account gesendet. Dieser Code muss eingegeben werden, um sicherzustellen, dass Benachrichtigungen im Störungsfall korrekt zugestellt werden.

Wird die Verifizierung übersprungen oder schlägt sie fehl, wird die Installation dennoch fortgesetzt. Die Benachrichtigungseinstellungen können später manuell angepasst werden, jedoch ohne erneute automatische Verifizierung. In diesem Fall liegt die vollständige Verantwortung für eine funktionierende Zustellung beim Nutzer. Ein entsprechender Workaround ist separat beschrieben.

Gruppenlogik und Abhängigkeiten

Die Gruppenlogik wurde eingeführt, um Benachrichtigungsfluten gezielt zu vermeiden. Ein typisches Anwendungsbeispiel ist die Überwachung einer VPN-Verbindung mit mehreren dahinterliegenden Geräten oder Diensten.

Fällt in einem solchen Szenario die VPN-Verbindung aus, sind alle nachgelagerten Geräte ebenfalls nicht erreichbar. Ohne Gruppenlogik würde dies zu einer Vielzahl gleichzeitiger Störungsmeldungen führen, obwohl die eigentliche Ursache lediglich der VPN-Ausfall ist.

Durch die Gruppenbildung prüft Faultnotify zuerst den übergeordneten Eintrag. Ist dieser nicht erreichbar, werden die Prüfungen der abhängigen Geräte ausgesetzt und es erfolgen keine weiteren Benachrichtigungen für diese Einträge.

Erst wenn der übergeordnete Dienst wieder verfügbar ist, werden die abhängigen Geräte erneut geprüft. Dadurch bleiben Benachrichtigungen übersichtlich, aussagekräftig und auf die tatsächliche Ursache beschränkt.

□□ Geräteprüfung (TEST)

Das Modul test ist die zentrale Prüffunktion von Faultnotify. Es prüft alle in der Konfiguration hinterlegten Geräte und Dienste und erkennt Störungen sowie Wiederherstellungen.

Bei direktem Aufruf führt test genau einen vollständigen Durchlauf aus. Das bedeutet:

- Alle Geräte und Dienste werden einmal geprüft.
- Im Fehlerfall werden Benachrichtigungen ausgelöst.
- Danach endet der Durchlauf automatisch.

□□ Automation (RUN)

Dies ist der Dauerschleifen-Modus. Hier werden permanent mit einem Versatz von 5 Sekunden die in der Konfiguration eingerichteten Geräte und Dienste geprüft.

Hinweis: Wird `run` direkt im Terminal gestartet, läuft Faultnotify im Vordergrund und blockiert das Terminal. Für den Dauerbetrieb wird die Einrichtung als Systemdienst empfohlen. Dadurch läuft

Faultnotify dauerhaft im Hintergrund und der Status ist jederzeit über `systemctl status faultnotify` sichtbar.

Geräte/Dienste hinzufügen (ADD)

Mit dem Modul **add** können neue Geräte und Dienste interaktiv zur Überwachung hinzugefügt werden. Der Vorgang ist dialoggeführt aufgebaut und kann mehrfach hintereinander ausgeführt werden, um mehrere Einträge nacheinander anzulegen.

Zu Beginn ermittelt Faultnotify automatisch die nächste freie Geräte-ID. Diese wird fortlaufend vergeben.

Im Anschluss werden Schritt für Schritt folgende Angaben abgefragt:

- **Name des Geräts oder Dienstes**
  Der Anzeigename dient ausschließlich der Übersicht. Kritische Zeichen werden automatisch bereinigt, um die Konfigurationsdatei konsistent zu halten.
- **IP-Adresse oder Domain**
  Es kann entweder eine IPv4-Adresse oder ein gültiger Domainname angegeben werden.
- **Dienstüberwachung (optional)**
  Falls ein Dienst überwacht werden soll, wird der zu prüfende Port abgefragt.
- **Abhängigkeit von einem Master-Gerät (optional)**
  Geräte und Dienste können einem bestehenden Eintrag untergeordnet werden. Dadurch lassen sich Abhängigkeiten abbilden, zum Beispiel:
  - Dienste hängen von einem Server ab
  - Untergeräte hängen von einem Hauptgerät ab

  Ist kein Master angegeben, wird der Eintrag automatisch der Root-Gruppe (G000) zugeordnet.

Nach Abschluss der Eingaben wird der neue Eintrag:

- in die Konfigurationsdatei geschrieben
- im internen Log protokolliert
- sofort für künftige Prüfungen berücksichtigt

Der Hinzufügen-Modus kann jederzeit durch eine leere Eingabe beim Namen beendet werden. Nach dem ersten erfolgreichen Hinzufügen wird Faultnotify als „eingerichtet“ markiert, sodass die Überwachung genutzt werden kann.

Hinweis zur Gruppenlogik:

Die Gruppenlogik entstand aus der praktischen Anforderung heraus, Benachrichtigungsfluten zu vermeiden. Ein typisches Beispiel ist die Überwachung einer VPN-Verbindung mit mehreren dahinterliegenden Geräten oder Diensten.

Fällt in einem solchen Szenario die VPN-Verbindung aus, wären die nachgelagerten Geräte zwangsläufig ebenfalls nicht erreichbar. Ohne Gruppierung würde dies zu einer Vielzahl gleichzeitiger Störungsmeldungen führen, obwohl die eigentliche Ursache nur der VPN-Ausfall ist.

Durch die Gruppenbildung wird dieses Problem gezielt verhindert. Wird ein Gerät oder Dienst als abhängig von der VPN-Überwachung angelegt, prüft Faultnotify zuerst den übergeordneten Eintrag. Ist dieser nicht erreichbar, werden die untergeordneten Prüfungen ausgesetzt und es erfolgt keine weitere Benachrichtigung für die abhängigen Geräte.

Erst wenn die VPN-Verbindung wieder verfügbar ist, werden die nachgelagerten Geräte erneut geprüft. Auf diese Weise bleibt die Benachrichtigung aussagekräftig, übersichtlich und auf die tatsächliche Ursache beschränkt.

✏ Modifizierung/Bearbeitung (MOD)

Das Modul **mod** dient zur nachträglichen Pflege und Anpassung von Faultnotify. Es ermöglicht sowohl die Verwaltung einzelner Geräte und Dienste als auch die Änderung zentraler Konfigurationseinstellungen.

Nach dem Start von mod wird zunächst abgefragt, **welcher Bereich geändert werden soll**:

- **Geräte und Dienste**
- **Zentrale Konfiguration**

Geräte und Dienste bearbeiten

Wird der Geräte-/Service-Bereich gewählt, zeigt Faultnotify zunächst **alle aktuell eingerichteten Einträge** in einer übersichtlichen Liste an. Jeder Eintrag enthält unter anderem:

- Geräte- bzw. Service-ID
- Gruppenzugehörigkeit
- Name
- IP-Adresse
- optional Port und Protokoll

Anschließend wird abgefragt, **welcher Eintrag geändert werden soll**. Die Auswahl kann flexibel erfolgen:

- über die Geräte-ID (z. B. `D006` oder `006`)
- über einen Teil des Namens
- über die vollständige IP-Adresse

Nach eindeutiger Zuordnung des Eintrags stehen zwei Optionen zur Verfügung:

1. **Eintrag löschen und neu definieren**
   Der bestehende Eintrag wird entfernt, zugehörige Status- und Jail-Informationen werden bereinigt und anschließend wird automatisch der add-Dialog gestartet, um das Gerät oder den Dienst neu anzulegen.
2. **Eintrag ersatzlos löschen**
   Der Eintrag wird vollständig aus der Konfiguration entfernt. Auch hier werden zugehörige Status- und Jail-Dateien bereinigt.

Damit lassen sich Geräte und Dienste sauber korrigieren, ersetzen oder dauerhaft entfernen.

Konfiguration bearbeiten

Wird die Konfiguration gewählt, zeigt Faultnotify zunächst **alle relevanten aktuellen Einstellungen** an, inklusive erklärender Hinweise. Dazu gehören unter anderem:

- Online-Ping-Prüfung (Referenz für Internet-Erreichbarkeit)
- Art der Benachrichtigung (E-Mail oder Telegram)
- E-Mail-Empfänger
- Telegram-Bot-Token
- Telegram-Chat-ID
- Nächste zu vergebende Geräte-ID

Anschließend kann gezielt ein einzelner Parameter geändert werden. Jede Änderung erfolgt interaktiv:

- Neuer Wert eingeben
- Übersicht anzeigen
- explizite Bestätigung vor dem Übernehmen

Eingaben werden dabei geprüft (z. B. Format von IP-Adressen, Domains, E-Mail-Adressen oder IDs), um fehlerhafte Konfigurationen zu vermeiden. Kommentare in der Konfigurationsdatei bleiben erhalten.

Ziel des MOD-Moduls

Das mod-Modul stellt sicher, dass Faultnotify auch im laufenden Betrieb **einfach, kontrolliert und konsistent gepflegt** werden kann - ohne manuelle Eingriffe in die Konfigurationsdatei und ohne das Risiko inkonsistenter Zustände.

□□ Übersicht (DRAW)

Mit dem Modul draw kann die komplette Geräte- und Dienststruktur übersichtlich als Baum angezeigt werden. Dabei werden alle Abhängigkeiten berücksichtigt, sodass auf einen Blick sichtbar ist, welche Geräte oder Dienste untergeordnet sind und welche als übergeordnete „Master“-Ebene dienen.

Die Darstellung beginnt bei ROOT (G000) und zeigt darunter alle Einträge aus der Konfiguration in einer klaren Hierarchie. Zusätzlich werden die wichtigsten Informationen direkt in der Ausgabe mitgeführt:

- ID des Eintrags
- Name bzw. Beschreibung
- IP-Adresse
- optional Port

Fehlerhinweis bei ungültigen Abhängigkeiten
Falls ein Eintrag auf ein nicht vorhandenes Master-Gerät verweist, gibt draw einen Hinweis aus und listet die betroffenen Einträge auf. Dadurch lassen sich fehlerhafte Abhängigkeiten schnell erkennen und anschließend über mod korrigieren.

```
./faultnotify.sh draw
[Faultnotify Hierarchie]
ROOT
├─ 001  Server 1                                  | 192.168.1.77
├─ 002  VPN Geschäftsstelle 4                     | 10.1.200.10
│   ├─ 003  Server Gechäftsstelle 4               | 10.2.1.1
│   ├─ 004  Plotter                               | 10.2.1.13
│   └─ 005  Terminalserver                        | 10.2.1.100 :3389
├─ 006  VPN Geschäftsstelle 5                     | 10.10.200.10
│   └─ 007  Proxmox 5                             | 10.10.200.20
│       ├─ 008  Webserver                         | 10.10.201.10
│       ├─ 009  Dockerserver                      | 10.10.201.20
│       │   ├─ 011  NPM 80                        | 10.10.201.40 :80
│       │   ├─ 012  NPM 81                        | 10.10.201.40 :81
│       │   └─ 013  Guacamole                     | 10.10.201.40 :10000
│       ├─ 010  Mailserver                        | 10.10.201.30
│       └─ 014  Admin-Terminal                    | 10.10.202.20
└─ 015  Medienserver                              | 192.168.1.78
```

Hier zu sehen: DRAW-Auflistung mit Beispielwerten.

□□ Fehler / Troubleshooting

Wird während der Installation eine Eingabe wie die Benachrichtigung falsch getägigt oder die Verifizierung der Benachrichtigung übersprungen oder schlägt fehl, wird die Installation trotzdem weitergeführt. Die Verifizierung dient dazu, dass sichergestellt werden kann, dass im Störungsfall auch Benachrichtigungen ankommen können.
In diesem Fall kann im Verzeichnis `script-data/faultnotify` die Datei `.installed` gelöscht werden, um die Installation erneut zu starten und die Erstkonfiguration nochmals vollständig durchzuführen.

□□ Sprachdateien

Faultnotify ist in der Lage, in verschiedenen Sprachen zu arbeiten und Benachrichtigungen entsprechend auszugeben. Grundlage dafür sind externe Sprachdateien. Im Script wird die **I18N-Übersetzungstechnologie** eingesetzt. Diese ermöglicht eine klare Trennung zwischen Programmlogik und Textausgaben und bildet die Grundlage für die mehrsprachige Ausgabe von Meldungen und Benachrichtigungen.

Deutsch ist die im Script fest integrierte Standardsprache. Zusätzlich steht eine englische Sprachdatei zum Download bereit.

Weitere Sprachen können jederzeit ergänzt werden. Die Sprachdateien sind bewusst einfach aufgebaut, sodass sie von jedermann erstellt oder erweitert werden können. Dadurch ist Faultnotify nicht auf eine feste Anzahl von Sprachen beschränkt und lässt sich flexibel an unterschiedliche Umgebungen anpassen.

Die Sprachdatei muss dann im zentralen Pfad im Ordner *lang/* eingepflegt werden. Liegen mehrere Sprachdateien im Ordner, wird die erste vom Script automatisch ausgewählt. Es ist daher ratsam, nur die benötigte Sprache herunterzuladen.

Faultnotify verwendet zur Einbindung von Sprachdateien einen SHA-256-Referenzcode, der direkt aus dem Script abgeleitet wird. Dieser Referenzcode stellt sicher, dass eine Sprachdatei eindeutig zu genau dieser Script-Version gehört. Diese Referenz ist in der ersten Zeile zu finden. Wenn eine neue Sprache integriert wird, ist dieser Referenzcode einzusetzen.

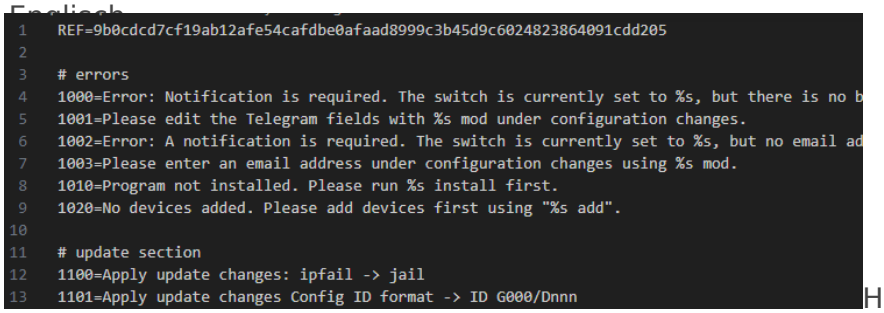
Dadurch wird verhindert, dass versehentlich eine unpassende oder fremde Sprachdatei verwendet wird, zum Beispiel aus einem anderen Script oder einer älteren Version. Stimmen Script und Sprachdatei nicht überein, wird die Sprachdatei nicht geladen.

Der zugehörige Referenzcode lautet:

```
9b0cdcd7cf19ab12afe54cafdbe0afaad8999c3b45d9c6024823864091cdd205
```

Bitte beachten, dass sich dieser Referenzcode mit einer neuen Version ändern kann. Hintergrund ist die Integrität zur neuen Version, falls neue Textabschnitte hinzukommen oder welche abgeändert werden.

Im Falle eines Updates wird darauf hingewiesen mit Link in der Script-Standardsprache (Deutsch) und Englisch.



Hier zu sehen: erste Zeilen der englischen Sprachdatei.

□□ Download

Das **Programm** kann hier in immer der neuesten Version heruntergeladen werden:

Faultnotify Script (via mariobeh.de)

```
curl -s https://public.mariobeh.de/scripts/faultnotify.sh | bash
```

Sprachdateien:

Englisch (via mariobeh.de)

```
USER=$(whoami) && mkdir -p /home/$USER/script-data/faultnotify/lang/ && curl -o /home/$USER/script-data/faultnotify/lang/faultnotify-eng.txt https://public.mariobeh.de/scripts/faultnotify-eng.txt && curl -s https://public.mariobeh.de/scripts/faultnotify.sh | bash
```

(um gleich in Englisch zu starten, ist dieser Befehl erfoderlich. Er erstellt die Verzeichnisstruktur und legt die Sprachdatei an passender Stelle ab. Dieser Befehl lädt nicht nur die Sprachdatei, sondern startet direkt das Script.)

Hinweis: Deutsch ist die im Script fest integrierte Standardsprache und benötigt keine separate Sprachdatei. Daher wird Deutsch hier nicht separat angeboten. Weitere Sprachen werden folgen.

□□ Update

Über ein externes Updater-Script kann Faultnotify immer geupdated werden. Link hier: Updater-Script (COMING SOON!)

Für die Sprachdatei gilt ein direktes Verfahren, einfach den Link einfügen, Sprachdatei wird automatisch an Ort und Stelle geladen. Alternativ kann oben die Datei manuell ins System eingepflegt werden.

```
USER=$(whoami) && mkdir -p /home/$USER/script-data/faultnotify/lang/ && curl -o /home/$USER/script-data/faultnotify/lang/faultnotify-eng.txt https://public.mariobeh.de/scripts/faultnotify-eng.txt
```

Revision #30
Created 20 January 2026 15:18:03 by Admin
Updated 11 April 2026 18:13:32 by Admin