

# Services

## Was verbirgt sich hinter dem Feature DTProxy? Was verbirgt sich hinter dem Feature DTProxy?

### DTProxy

DTProxy ermöglicht es Ihnen Ihre selbstgeschriebenen Server/Daemons in Ihre Domain einzubinden.

Unterstützt werden grundsätzlich alle Sprachen. Einzige Bedingung ist, dass das einzubindende Programm die Umgebungsvariable SERVER\_PORT als seinen Listen-Port übernimmt.

Ist Ihnen Spring Boot ein Begriff?

Nein?!? -> Spring Boot bietet die Möglichkeit sehr einfach eigenständig lauffähige Spring-Anwendungen in Java zu entwickeln. Weitere Informationen hierzu finden Sie unter: <http://projects.spring.io/spring-boot/>

Ich möchte anhand eines Spring Boot Beispiels DTProxy erklären:

#### 1. Das Beispielprojekt

Wie immer geht es mit einem "Hello World" los :-)

Unter nachfolgender Adresse finden Sie alle notwendigen Infos:

<http://docs.spring.io/spring-boot/docs/current-SNAPSHOT/reference/htmlsingle/#getting-started-first-application>

#### 2. Übertragung der jar-Datei auf deltatree.de

Hierzu haben Sie vielerlei Möglichkeiten: scp / rsync / (s)ftp / CPanel-Dateimanager.

Erzeugen Sie einen Ordner apps in Ihrem Homeverzeichnis und kopieren Sie die jar dorthin.

# Services

## 3. Konfiguration

Die Konfiguration erfolgt innerhalb nachfolgender Datei: `./dtproxy/dtproxy.conf`

Falls es das Verzeichnis `.dtproxy` nicht gibt, legen Sie dieses an: `mkdir ~/.dtproxy`

Die Datei hat folgenden Aufbau: Anwendungsname;;;Aufruf der Anwendung;;;Autostart nach Reboot

Hier ein Beispieldateiinhalte:

```
hello;;;usr/local/bin/java8 -Xmx20M -jar  
"home/username/apps/HelloWorld-0.0.1-SNAPSHOT.jar";;;0
```

## 4. Der erste Start der Anwendung

Geben Sie nun auf der Shell Folgendes ein: `dtproxy hello start`

DTPProxy gibt Ihnen aus, wo Sie STDOUT und STDERROR finden, je nachdem wie Ihre Anwendung gebaut wurde, befinden sich die Hauptinformationen in Ihrem eigenen Log. Aber ab und zu können STDOUT und STDERROR bei der Fehlersuche sehr hilfreich sein.

Prüfen Sie anhand der Logfiles, ob alles sauber hochgefahren wurde. Danach können Sie die Anwendung im Browser aufrufen:

`http(s)://$IhreDomain$/dtapp/$Anwendungsname$/`

In unserem Fall ist `$Anwendungsname$` `hello` =>

`http(s)://$IhreDomain$/dtapp/hello/`

## 5. Kontrolle der Anwendung

`dtproxy hello stop`

Stop der Anwendung durch einen SIGKILL

`dtproxy hello start`

Start der Anwendung

`dtproxy hello status`

Status der Anwendung

# Services

## 6. Wichtige Hinweise / Bekannte Probleme

Der Port, der an die Anwendung delegiert wird, wird 2 Minuten lang gecached: Dies bedeutet bei schnellen stop/start - Anweisungen kommt es am Browser zu Fehlermeldungen, da dort noch der alte Port verwendet wird. Dieses Problem ist bekannt und hierzu wird es bald eine Lösung geben.

Eindeutige ID: #1005

Verfasser:

Letzte Änderung: 2015-01-05 16:29