

ACHTUNG! Diese Seite wird nicht mehr länger gepflegt.
Wechselt auf die englischsprachige Seite für eine aktuelle
Anleitung (Link unten links).

CGI

Für schnellen Zugriff auf die [Kommandoschnittstelle](#) mit einem Browser stellt der Netzer eine CGI-Version zur Verfügung.

Verwendung

Die CGI-Version wird genutzt, indem man die Seite `action.cgi` des Netzers mit einem HTTP GET aufruft. Dabei wird der gewünschte Befehl als Schlüssel und der gewünschte Wert als Schlüssel-Wert-Paar im [Query String](#) des Aufrufs angegeben. Für den Netzer mit der Kennung `mein_netzer`, den Befehl 'a' und den Wert '0' lautete der Aufruf:

```
http://mein-netzer.local/action.cgi?a=0
```

Im einfachsten Fall wird diese URL direkt in die Adressleiste des Browsers eingeben.

Die HTTP-Response hat den Content-Type `application/json` und enthält die Antwort des Netzers als JSON-Objekt.

Pro Aufruf kann nur ein Kommandoschnittstellenbefehl ausgeführt werden. Werden mehrere Kommandoschnittstellenbefehle angegeben, wird nur der erste Befehl ausgeführt, alle anderen werden ignoriert. Zusätzlich zu dem Kommandoschnittstellenbefehl kann noch ein CGI-spezifischer Befehl angegeben werden, der ebenfalls ausgeführt wird. Auch hier gilt, dass pro Aufruf nur ein CGI-spezifischer Befehl ausgeführt wird. Werden mehrere CGI-spezifische Befehle angegeben, wird nur der erste ausgeführt.

Da der Netzer im Query String stets Schlüssel-Wert-Paare erwartet, kann es bei Lesebefehlen zu Problemen kommen, da diese nur aus einem Schlüssel (dem Befehl) ohne Wert bestehen. Dies geschieht immer dann, wenn mehrere Befehle in einem CGI-Aufruf übergeben werden.

Es wird grundsätzlich nur der erste gefundene Befehl ausgeführt, allerdings kann beim Versuch mehrere Befehle in einem Aufruf auszuführen aus einem Lese- ein Schreibbefehl werden:

Die beiden getrennten Befehle a und b in

```
http://mein-netzer.local/action.cgi?a&b
```

werden vom Netzer wie ein Befehl `a=b` interpretiert.

Äquivalent zu:

```
http://mein-netzer.local/action.cgi?a=b
```

Problematisch ist dieser Umstand bei der Kombination von Kommandoschnittstelle und CGI-spezifischen Befehlen. In diesen Fällen können zwei Befehle aus dem Query String ausgeführt werden, jeweils ein Befehl der Kommandoschnittstelle und ein CGI-spezifischer.

Der Lesebefehl `a` und der Redirect `zr=meine_seite`

```
http://mein-netzer.local/action.cgi?a&zr=meine_seite
```

werden als Schreibbefehl `a=zr` und Lesebefehl `meine_seite` interpretiert.

```
http://mein-netzer.local/action.cgi?a=zr&meine_seite
```

`meine_seite` wird nicht ausgeführt, da es der zweite Befehl der Kommandoschnittstelle wäre.

Das kann zu Fehlerbildern führen, die auf den ersten Blick sehr verwirrend sind.

Der Aufruf

```
http://mein-netzer.local/action.cgi?a&zr=zr
```

wird interpretiert wie

```
http://mein-netzer.local/action.cgi?a=zr&zr
```

Also wie der Schreibbefehl `a=zr` und der Redirect `zr`. Dabei wird als Ziel des Redirects ein leerer String angenommen.

Daher gilt: Wenn mehrere Befehle in einem Aufruf übergeben werden sollen, darf nur der letzte die Form eines Lesebefehls haben (nur ein Befehl, kein Wert). Alle anderen Befehle müssen die Form von Schreibbefehlen haben (ein Befehl und ein Wert).

Korrekt:

```
http://mein-netzer.local/action.cgi?zr=meine_seite&a
```

Falsch:

```
http://mein-netzer.local/action.cgi?a&zr=meine_seite
```

Spezifische Befehle

Redirect

Der Befehl `z r` veranlasst einen Redirect auf die als Wert von `z r` angegebene Seite.

Verwendet wird es vor allem in Kombination mit Schreibbefehlen der Kommandoschnittstelle. Durch diese Kombination kann man einen Schreibbefehl ausführen, ohne dass die Antwort des Netzers angezeigt wird.

Beispiel

```
http://mein-netzer.local/action.cgi?a=1&zr=/
```

Dieser Aufruf würde den GPIO-Kanal `a` auf 1 setzen und danach die Startseite des Netzers anzeigen.

From:

<http://mobacon.de/wiki/> - MoBaCon Wiki

Permanent link:

<http://mobacon.de/wiki/doku.php/de/netzer/actioncgi>

Last update: **2014/02/09 13:32**

