

Steffen Probst

Alle hier erstellten Inhalt dieser Seiten und deren Unterseite stelle ich unter der [GPLv3](#). Ihr dürft die Inhalte gerne im Sinne dieser Lizenz weiterverwenden.

Alle auf dieser Seite und Unterseiten veröffentlichten Bilder stelle ich unter der [CC BY-NC-ND^{1\)}](#) Lizenz, wenn dieses nicht anders angegeben wurde. Ihr dürft die Bilder im Sinne der Lizenz verwenden.

Ausgeschlossen von meiner Lizenzierung sind Bilder, Inhalte, Dokumente und Informationen, die ich über extern verlinkt habe. Auch, wenn das nicht direkt ersichtlich ist. Wenn ihr/euch nicht sicher seit, dann kontaktiert mich über meine [LUG-E-Mail-Adresse](#).

Blog

Ubiquiti Unifi Switch - Scripting

Neben dem Standard Unifi Controller kann im Unifi-Umfeld auch gescrriptet werden. Durch meine neue Aufgabe habe ich mich mit Brocade, Ruckus und CISCO befassen dürfen. Dadurch bin ich auf die Idee gekommen auch mal im Ubiquiti und HP Umfeld zu schauen, was dort per CLI(SSH und Telnet) geht und siehe da, die Geräte verstecken eine Menge Features, die man als Laie so nicht kennt. Dabei ist mir dann die Idee gekommen PoE Ports zum Zeitgesteuert an und abzuschalten.

Nach ein paar Test habe ich im Internet zu diesem Thema recherchiert und habe sogar jemand gefunden, der diese Idee schon realisiert hat.

Hier der Link: <https://nerdig.es/usw-poe-per-script-steuern/> Das Script veröffentliche ich als Backup hiermit. Ich hoffe, das ist okay und werden Eigentümer nachträglich dazu befragen. Ich empfinde es als eine sehr smarte Lösung, die man mit einem Unifi Controller kombinieren kann. Ich verwende diesen mit einem Alpine Linux.

```
#!/bin/bash
# https://nerdig.es/usw-poe-per-script-steuern/
# Stephan Klein
# Dipl.-Inform. (FH)

function usage {
    echo
    echo "$Usage: $0 <on|off> "
    echo
    echo "Example: $0 192.168.178.12 on 8"
    echo
}

function is_number {
    [[ "$1" =~ ^[0-9]+$ ]] && return 0 || return 1
}

function is_host {
```

```
    [[ "$1" =~ ^[0-9a-zA-Z.-]+$ ]] && return 0 || return 1
}

function valid_action {
    [[ "${1,,}" =~ ^on|off$ ]] && return 0 || return 1
}

# check if hostname or IP is provided as target switch
if ! is_host "$1"; then
    usage
    echo -e "ERROR: Target switch must be a valid host name or IPv4/v6
address!\n"
    exit 2
fi

# check if on or off is provided as action
if ! valid_action "$2"; then
    usage
    echo -e "ERROR: action must be on or off!\n"
    exit 2
fi

# check if port-parameter is number
if ! is_number "$3" ; then
    usage
    echo -e "ERROR: portnumber contains non-numerical characters!\n"
    exit 3
fi

cmd_pre="enable\n$password\nconfigure\n"

cmd_on="interface 0/$3\npoe opmode auto\nexit"
cmd_off="interface 0/$3\npoe opmode shutdown\nexit"
cmd_status="show poe status 0/$3"

cmd_post="\nexit\nexit\n"

case "${2,,}" in
    on)
        ssh admin@$1 '(echo "enable" ; echo "configure" ; echo
"interface 0/'$3'" ; echo "poe opmode auto" ; echo "exit" ; echo "exit";
echo "exit" ) | telnet localhost 23 ; exit;'
        ;;
    off)
        ssh admin@$1 '(echo "enable" ; echo "configure" ; echo
"interface 0/'$3'" ; echo "poe opmode shutdown" ; echo "exit" ; echo "exit";
echo "exit" ) | telnet localhost 23 ; exit;'
        ;;
    *)
        usage

```

```

                exit 2
                ;;
esac

```

2022/04/15 19:26 · SProbst
[unifi](#), [controller](#), [switch](#), [cronjob](#)

svgo zum bereinigen und reparieren von .svg

Heute habe ich etwas mit inkscape gespielt. Ein Kollege hat aus einem alten Projekt mir Dateien zur Verfügung stellt. Dabei ist mir eine Fehler in den .svg Dateien aufgefallen. [svgo](#) wird im Internet empfohlen, wenn man Dateien reparieren und verkleinern möchte. Ich habe das Tool gleich ausprobiert und bin beeindruckt. Die Datei ist um die Hälfte verkleinert wurden.

Wichtig: Dabei gehen aber alle Ebenen verloren, die man in Inkscape angelegt hat.

Ich kann euch das Tool empfehlen. Es macht was soll.

Link

- <https://github.com/svg/svgo>

2021/07/30 19:58 · SProbst
[svgo](#), [2021](#), [tool](#), [svg](#), [inkscape](#), [werkzeug](#)

kleine Tastenkürzelübersicht zu vi(m)

| Tastenkürzel | Beschreibung |
|--------------|--|
| i | Zeichen einfügen in derselben Zeile wo der Cursor steht. |
| o | Zeichen einfügen eine neue Zeile. |
| x | Zeichen löschen, wo der Cursor steht. |
| dd | Zeile löschen wo der Cursor steht. |
| :q! | vi(m) ohne zu speichern verlassen |
| :wq | vi(m) mit speichern verlassen |

- [kleine Anleitung als PDF von fehcom.de](#)

2021/06/04 13:04 · SProbst
[vim](#), [vi](#), [kleine](#), [uebersicht](#)

openwrt auf der Hetzner Cloud

- Eine kurze Einrichtungsanleitung eines Hetzner Cloud KVM mit openwrt.

erste Einstellung nach reboot von openwrt

- Netzwerkkarte eth0 für WAN konfigurieren und LAN auf eth1, dummy0 oder VLAN binden.

[/etc/config/network](#)

```
config interface 'wan'
  option type 'bridge'
  option ifname 'eth0'
  option proto 'dhcp'
  option peerdns '0'

config interface 'wan6'
  option ifname '@wan'
  option _orig_ifname '@wan'
  option _orig_bridge 'false'
  option proto 'static'
  option ip6ifaceid '::2'
  option ip6gw 'fe80::1'
  option ip6addr '2a01:xxxx:xxxx:xxxx::2' #IPv6 von Hetzner anpassen
  option ip6prefix '2a01:xxxx:xxxx:xxxx::2/64' #IPv6 Prefix von
Hetzner anpassen

config interface 'lan'
  option type 'bridge'
  option ifname 'eth1'
  option _orig_ifname 'eth1'
  option _orig_bridge 'true'
  option proto 'dhcp'
  option delegate '0'
  option defaultroute '0'
  option metric '100'
  option peerdns '0'
```

- Firewall für Zugriff auf SSH einrichten

[/etc/config/firewall](#)

```
config rule
  option src 'wan'
  option target 'ACCEPT'
  option proto 'tcp'
  option dest_port '22'
  option name 'OpenSSH'
```

- Dienste neustarten

```
/etc/init.d/network restart && /etc/init.d/firewall restart
```

2021/06/04 12:46 · [SProbst](#)

[rclone - Sync mit nextcloud](#)



Gestern habe ich ein neues Programm unter Linux getestet. Im Rahmen einer Anforderung sollte automatisch Mediendateien in die Nextcloud kopiert werden. Bei meiner Internetrecherche habe ich das Programm [rclone](#) gefunden. Nach der ersten Einrichtung, was recht benutzerfreundlich ist, für ein Terminalprogramm, lief auch der Test gleich erfolgreich.

Das Programm ist sehr eingängig und ich es kann es jedem empfehlen, der Dateien in die Cloud laden möchte. rclone unterstützt nach eigenen Angaben 40 Clouds. Das habe ich nicht getestet. Aber der Upload mit Nextcloud lief einwandfrei. Es lässt sich auch wunderbar in diverse Scripte integrieren. Eine Anleitung für die Nextcloud ist unter folgendem Link zu finden. →

<https://rclone.org/webdav/#nextcloud>

2021/02/26 07:22 · [SProbst](#)

[sync](#), [cloud](#), [rclone](#), [program](#), [test](#), [empfehlung](#)

[Ältere Einträge >>](#)

Index

A

- [About me ...](#)

B

- [Blog](#)

N

- [Notizen](#)

T

- [Test](#)

L

- [Linksammlung](#)

2019/11/13 20:08 · [SProbst](#)

¹⁾

Namensnennung - Nicht-kommerziell - Keine Bearbeitung

²⁾

Logo von der Homepage von [rclone.org](#)

From:

<https://wiki.lug-wr.de/wiki/> - **Wiki der Linux User Group Wernigerode**

Permanent link:

<https://wiki.lug-wr.de/wiki/doku.php?id=user:sprobst>



Last update: **2019/11/13 20:12**