DynDNS Updater (Debian / 10.07.2024)

Der DynDNS Updater aktualisiert eine DynDNS-Domain mit der aktuellen IP-Adresse.
Wenn das Script zum ersten Mal ausgeführt wird, muss man die API zur Aktualisierung eingeben, die dieses Script dann aufruft.
Diese API wird dann im script-data-Ordner gespeichert.

Genaue Zusammenfassung:
data=/home/$USER/script-data/$0

Wenn noch keine IP vorhanden ist ($data/ip.txt), dann wird die IP-Adresse von "icanhazip.com" abgerufen.
API wird aufgerufen, IP wurde aktualisiert.
Jeder Schritt wird mitgeloggt in $data/allgemein.csv (besondere Ereignisse), tagesaktuell in $data/logs/YYYYMMTT.csv (Updates).

□□ Was macht das Script?

□□ Lade bisherigen Status:

- Wenn `ip.txt` vorhanden ist, wird die bisherige öffentliche IP geladen.
- Wenn nicht: → erste Ausführung → IP bleibt leer, Log-Eintrag in `allgemein.csv`.

□□ Prüfe Update-URL:

- Wenn `updateurl.txt` existiert → lese erste Zeile ein.
- Wenn nicht vorhanden:
  - Benutzer wird zur Eingabe einer gültigen `http(s)`-URL aufgefordert.
  - Wenn gültig: speichere in `updateurl.txt` & logge es.
  - Wenn ungültig: Abbruch & Fehlerlog.

□□ Online-Check:

- Ping an `icanhazip.com` (IPv4, 1 Paket, Timeout 3s)
- Wenn erreichbar:
  - Hole neue IP mit `curl`
  - Vergleiche mit vorheriger IP
    - Unverändert: nur Statusmeldung
    - Geändert:
      - speichere neue IP in `ip.txt`
      - rufe `updateurl` auf (z. B. DynDNS-Anbieter)
      - logge Aktualisierung in Tageslog
      - sende E-Mail-Benachrichtigung
- Wenn nicht online:
  - logge, dass keine Aktualisierung möglich ist

□□ Benötigte Tools:

- `curl`
- `ping`
- `mail`
- `sed`, `head`, `tail`
- Bash >= 4

Inhalt der $data/updateurl.txt:

```
https://api.dyndns.tld/update.php?key=q12w3e4r5t6z7u8i9o0p
```

Script:

```
user=$(whoami)
data="/home/$user/script-data/$(basename "$0")"
ippage="icanhazip.com"

□if [ ! -d "$data" ]; then
□□mkdir -p "$data"
□fi

□if [ ! -d "$data/logs" ]; then
□□mkdir "$data/logs"
□fi
```

```
function ipcheck {
	echo "$(date +"[%d.%m.%Y %H:%M:%S]")"

	if [ -f "$data/ip.txt" ]; then
		ip=$(head -n1 "$data/ip.txt")
		echo "Bisherige IP: $ip"
	else
		echo "Erste Ausführung."
		ip=""
		echo "$(date +"%d.%m.%Y;%H:%M:%S");Erste Ausführung;$1" >> "$data/logs/allgemein.csv"
	fi

	if [ -f "$data/updateurl.txt" ]; then
		updateurl=$(sed 's/ //g' "$data/updateurl.txt" | head -n1 | tail -n1)
	else
		echo "Keine Update-URL hinterlegt. URL hier einfügen:"
		echo "$(date +"%d.%m.%Y;%H:%M:%S");Update-URL fehlt" >> "$data/logs/allgemein.csv"
		read -p "Update-URL: " updateurl

		# Überprüfe, ob die URL mit "http://" oder "https://" beginnt
		if [[ "$updateurl" =~ ^(http|https):// ]]; then
			echo "URL OK."
			echo "$updateurl" > "$data/updateurl.txt"
			echo "$(date +"%d.%m.%Y;%H:%M:%S");Update-URL hinterlegt;$1" >> "$data/logs/allgemein.csv"
		else
			echo "Die URL ist ungültig."
			echo "$(date +"%d.%m.%Y;%H:%M:%S");Update-URL ungültig;$1" >> "$data/logs/allgemein.csv"
			exit 1
		fi
	fi

	echo "Prüfe Online-Status"
	if ping -4 -c 1 -W 3 "$ippage" >/dev/null 2>&1; then
		echo "Anschluss Online"
		ipneu=$(curl -s "$ippage")
		echo "IP: $ipneu"

		if [ "$ip" = "$ipneu" ]; then
			echo "IP unverändert."
```

```
	#	echo "$(date +"%d.%m.%Y;%H:%M:%S");IP unverändert;$1" >> "$data/logs/$(date +"%Y%m%d").csv"
	else
		echo "Bisherige IP: $ip"
		echo "Neue IP     : $ipneu"
		echo "Aktualisiere..."
		echo "$ipneu" > "$data/ip.txt"
		curl -sSL "$updateurl" >/dev/null
		echo "OK."
		echo "$(date +"%d.%m.%Y;%H:%M:%S");Aktualisiert auf $ipneu;$1" >> "$data/logs/$(date +"%Y%m%d").csv"
		echo "$(date +"[%d.%m.%Y %H:%M:%S]") IP aktualisiert auf $ipneu" | mail -s "DynDNSupdater"
"mail@mariobeh.de"
	fi
else
	echo "Client ist selbst offline, kein Aktualisieren möglich."
	echo "$(date +"%d.%m.%Y;%H:%M:%S");Client offline;$1" >> "$data/logs/$(date +"%Y%m%d").csv"
fi
}



if [ -z "$1" ]; then
	ipcheck
fi

exit 0
```

Revision #8
Created 6 April 2025 15:00:19 by Admin
Updated 6 April 2025 15:21:50 by Admin