

Finalni Vodič: Toplik Service + ha-bridge + Alexa Integracija (Verzija 2.0)

Ovaj dokument opisuje kompletan i ispravan proces za postavljanje Toplik Service Python servera i ha-bridge (Philips Hue Emulatora) na Raspberry Pi (toplik-server), te njihovo povezivanje sa Amazon Echo (Alexa) uređajima za glasovnu kontrolu.

Faza 1: Postavljanje Python Servera (Toplik Service)

Ovaj server služi kao glavni mozak sistema koji prima komande (iz Vaše web aplikacije ili od Alexe) i prosljeđuje ih ESP32 uređajima.

1.1. Kreirajte Direktorij Servisa

```
mkdir /home/admin/ToplikService  
cd /home/admin/ToplikService
```

1.2. Instalirajte Zavisnosti

Kreirajte fajl requirements.txt sa nano requirements.txt:

```
Flask  
requests  
PyJWT  
waitress
```

Instalirajte ih:

```
pip install -r requirements.txt
```

1.3. Kreirajte Fajlove Aplikacije

Unutar /home/admin/ToplikService/, kreirajte sve Vaše fajlove:

- config_server.json (Sadrži PIN-ove, mDNS imena, ID-jeve uređaja, itd.)
- server.py (Glavna Python aplikacija)
- login.html, soba.html, admin.html, admin_login.html (i prateći static/ folder)

1.4. KLJUČNA ISPRAVKA: Modifikacija server.py za Ograničavanje Temperature

Ovo je najvažniji korak koji smo otkrili. Moramo osigurati da Vaš server.py može ispravno "ograničiti" (clamping) komande koje stižu od Alexe (koja šalje procenat) na validan raspon temperature (18-30).

1. Otvorite skriptu za editovanje:
nano /home/admin/ToplikService/server.py
2. Pronađite funkciju api_ifttt_control (pri dnu fajla, oko linije 345). Pronađite ovaj **originalni (pogrešni)** blok koda:

```
# Prilagodi parametre za ESP32
if komanda == 'SET_PIN':
    params['VALUE'] = '1' if vrijednost else '0'
elif komanda == 'SET_ROOM_TEMP':
    params['VALUE'] = str(int(round(vrijednost)))
else:
    return jsonify({'success': False, 'message': f'Komanda {komanda} nije podržana preko IFTTT.'}), 500
```

3. **Zamijenite** cijeli taj if/elif/else blok sa ovim **novim, ISPRAVNIM blokom** koji radi **ograničavanje (clamping)**:

```
# Prilagodi parametre za ESP32
if komanda == 'SET_PIN':
    params['VALUE'] = '1' if vrijednost else '0'

elif komanda == 'SET_ROOM_TEMP':
    try:
        # Vrijednost (procenat) stiže od Alexe kao broj. Tretiramo ga kao temperaturu.
        temp_trazena = float(vrijednost)

        # --- OVDJE JE OGRANIČAVANJE (Clamping) ---
        MIN_TEMP = 18 # Minimalna dozvoljena temperatura
        MAX_TEMP = 30 # Maksimalna dozvoljena temperatura

        final_value = int(round(temp_trazena))

        if final_value < MIN_TEMP:
            final_value = MIN_TEMP
            logging.info(f"IFTTT/ALEXA: Vrijednost {temp_trazena} je ISPOD minimuma.
Postavljam na {final_value}°C")
        elif final_value > MAX_TEMP:
            final_value = MAX_TEMP
            logging.info(f"IFTTT/ALEXA: Vrijednost {temp_trazena} je IZNAD maksimuma.
Postavljam na {final_value}°C")
```

```

else:
    logging.info(f"IFTTT/ALEXA: Postavljam temperaturu na {final_value}°C")

    params['VALUE'] = str(final_value)

except ValueError:
    logging.error(f"IFTTT: Nije moguće pretvoriti vrijednost '{vrijednost}' u broj za termostat.")
    params['VALUE'] = str(MIN_TEMP) # Vрати na minimum ako je greška

else:
    return jsonify({'success': False, 'message': f'Komanda {komanda} nije podržana preko IFTTT.'}), 500

```

4. Sačuvajte fajl (Ctrl+O, Enter, Ctrl+X).

Faza 2: Postavljanje ha-bridge (Alexa Emulator)

Ovaj softver će se pretvarati da je Philips Hue bridge, primati komande od Alexe i prosljeđivati ih Vašem server.py (koji smo podesili u Fazi 1).

1. Kreirajte direktorij za ha-bridge:

```
mkdir /home/admin/habridge
cd /home/admin/habridge
```
2. Preuzmite ha-bridge (verzija 5.4.1 koju Vaši logovi koriste):

```
wget
[https://github.com/bwssystems/ha-bridge/releases/download/v5.4.1/ha-bridge-5.4.1-java11.jar](https://github.com/bwssystems/ha-bridge/releases/download/v5.4.1/ha-bridge-5.4.1-java11.jar)
```
3. Kreirajte data direktorij (neophodan za bazu podataka uređaja):

```
mkdir data
```

Faza 3: Postavljanje Servisa (Automatsko pokretanje)

Kreiramo dva "čuvara" (systemd servisa) da se oba servera pale automatski pri restartu Raspberry Pi-a.

1. **Kreirajte toplik.service (Vaš Python server):**

```
sudo nano /etc/systemd/system/toplik.service
```

Zalijepite ovaj sadržaj:
[Unit]

Description=Toplik Service Python Backend
After=network.target

```
[Service]
User=admin
Group=admin
WorkingDirectory=/home/admin/ToplikService/
ExecStart=/usr/bin/python3 -m waitress --port=5000 server:app
Restart=always
RestartSec=5
```

```
[Install]
WantedBy=multi-user.target
```

Sačuvajte (Ctrl+O, Enter, Ctrl+X).

2. **Kreirajte habridge.service (Alexa Emulator):**

```
sudo nano /etc/systemd/system/habridge.service
```

Zalijepite ovaj sadržaj. **Ovo je ključna ispravka** – dodajemo -Dserver.port=80 (da radi na standardnom portu) i -Ddisable.mDNS=true (da spriječimo "šum" na mreži koji smo vidjeli u logovima).

```
[Unit]
Description=HA Bridge (Alexa Emulator)
# Pobrini se da se pokrene tek nakon što je Vaš Python server aktivan
After=network.target toplik.service
BindsTo=toplik.service
```

```
[Service]
User=admin
Group=admin
WorkingDirectory=/home/admin/habridge/
ExecStart=/usr/bin/java -Dha.config.dir=./data -Dserver.port=80 -Ddisable.mDNS=true
-jar ha-bridge-5.4.1-java11.jar
Restart=always
RestartSec=5
```

```
[Install]
WantedBy=multi-user.target
```

Sačuvajte (Ctrl+O, Enter, Ctrl+X).

3. **Aktivirajte Oba Servisa:**

```
sudo systemctl daemon-reload
sudo systemctl enable toplik.service
```

```
sudo systemctl enable habridge.service
```

```
# Pokreni ih (ili restartuj ako već rade)
```

```
sudo systemctl restart toplik.service
```

```
sudo systemctl restart habridge.service
```

Faza 4: Konfiguracija ha-bridge Uređaja

Sada kada oba servera rade, moramo ha-bridge-u reći koje komande da šalje Vašem server.py (koji radi na portu 5000).

1. Otvorite ha-bridge sučelje u browseru: <http://192.168.88.70>
2. Idite na tab "**Add/Edit**".

UREĐAJ 1: Chandelier 505 (Prekidač)

- **Name:** Chandelier 505
- **Device Type:** Ostavite Switch
- **On Items:**
 - **Type:** httpDevice
 - **Http Verb:** POST
 - **Http Content Type:** application/json
 - **Target Item:** <http://127.0.0.1:5000/api/ifttt/control>
 - **Http Body:**
{
 "api_key": "KLJUC-ZA-HOTELSKI-SOFTVER-98765",
 "room_id": "505",
 "uredjaj":
 "light_luster",
 "vrijednost": "on"
}
- **Off Items:**
 - **Type:** httpDevice
 - **Http Verb:** POST
 - **Http Content Type:** application/json
 - **Target Item:** <http://127.0.0.1:5000/api/ifttt/control>
 - **Http Body:**
{
 "api_key": "KLJUC-ZA-HOTELSKI-SOFTVER-98765",
 "room_id": "505",
 "uredjaj":
 "light_luster",
 "vrijednost": "off"
}
- **Map Type (KLJUČNO):**
 - Označite (čekirajte) **hueDevice**.
- Kliknite "**Add Bridge Device**".

UREĐAJ 2: Termostat 505 (Termostat)

- **Name:** Termostat 505
- **Device Type (KLJUČNO):** Ostavite Switch (Ne "Termostat", jer to zbunjuje Alexu).
- **On Items / Off Items:** Ostavite prazno. Kontrola ide samo preko "Dim".

- **Dim Items (Ovo je za temperaturu):**
 - **Type:** httpDevice
 - **Http Verb:** POST
 - **Http Content Type:** application/json
 - **Target Item:** http://127.0.0.1:5000/api/ifttt/control
 - **Http Body:**

```
{"api_key": "KLJUC-ZA-HOTELSKI-SOFTVER-98765", "room_id": "505", "uredjaj": "termostat_set", "vrijednost": "${intensity.percent}"}
```
- **Map Type (KLJUČNO):**
 - Označite (čekirajte) **hueDevice**.
- Kliknite **"Add Bridge Device"**.

(Ponavljajte Fazu 4 za sve ostale prekidače i termostate koje želite dodati).

Faza 5: Povezivanje sa Alexom (Finalni Proces)

Ovo je najosjetljiviji dio, jer Alexa "pamti" stare greške. Moramo je natjerati na "čisti" start.

1. **Obrišite Stare Uređaje (u Alexa Aplikaciji):**
 - Otvorite Vašu **Alexa Aplikaciju**.
 - Idite na **"Devices"** (Uređaji).
 - Pronađite **SVE** stare uređaje koje ste pokušavali dodati ("Chandelier 505", "Termostat 505", itd.) i **obrišite ih** (Settings -> Delete/Forget).
2. **Restartujte Echo Dot (Ključni Korak za Brisanje Keša):**
 - **Isključite Vaš Echo Dot (192.168.88.81) iz struje.**
 - Sačekajte 30 sekundi.
 - Uključite ga ponovo u struju i sačekajte da se sistem dignu.
3. **Pokrenite Discovery:**
 - Idite na ha-bridge sučelje (<http://192.168.88.70>) -> tab **"Bridge Control"**.
 - Kliknite **"Press Link Button"**.
 - **ODMAH** recite Alexi: **"Alexa, discover devices"**.

Ovaj put, Alexa će pronaći sve uređaje koje ste unijeli u Fazi 4.

Faza 6: Glasovne Komande (Korištenje)

- **Za Prekidače (npr. Chandelier 505):**
 - "Alexa, turn on Chandelier 505."
 - "Alexa, turn off Chandelier 505."
- **Za Termostat (npr. Thermostat 505):**
(Morate koristiti riječ "percent" jer Alexa misli da je ovo "svjetlo" koje se dima)
 - **"Alexa, set Thermostat 505 to 22 percent."**
 - (Skripta u Fazi 1 će ovo primiti kao 22 i poslati VALUE=22)
 - **"Alexa, set Thermostat 505 to 100 percent."**

- *(Skripta će ovo primiti kao 100, ograničiti na 30 i poslati VALUE=30)*
- **"Alexa, set Thermostat 505 to 10 percent."**
 - *(Skripta će ovo primiti kao 10, ograničiti na 18 i poslati VALUE=18)*