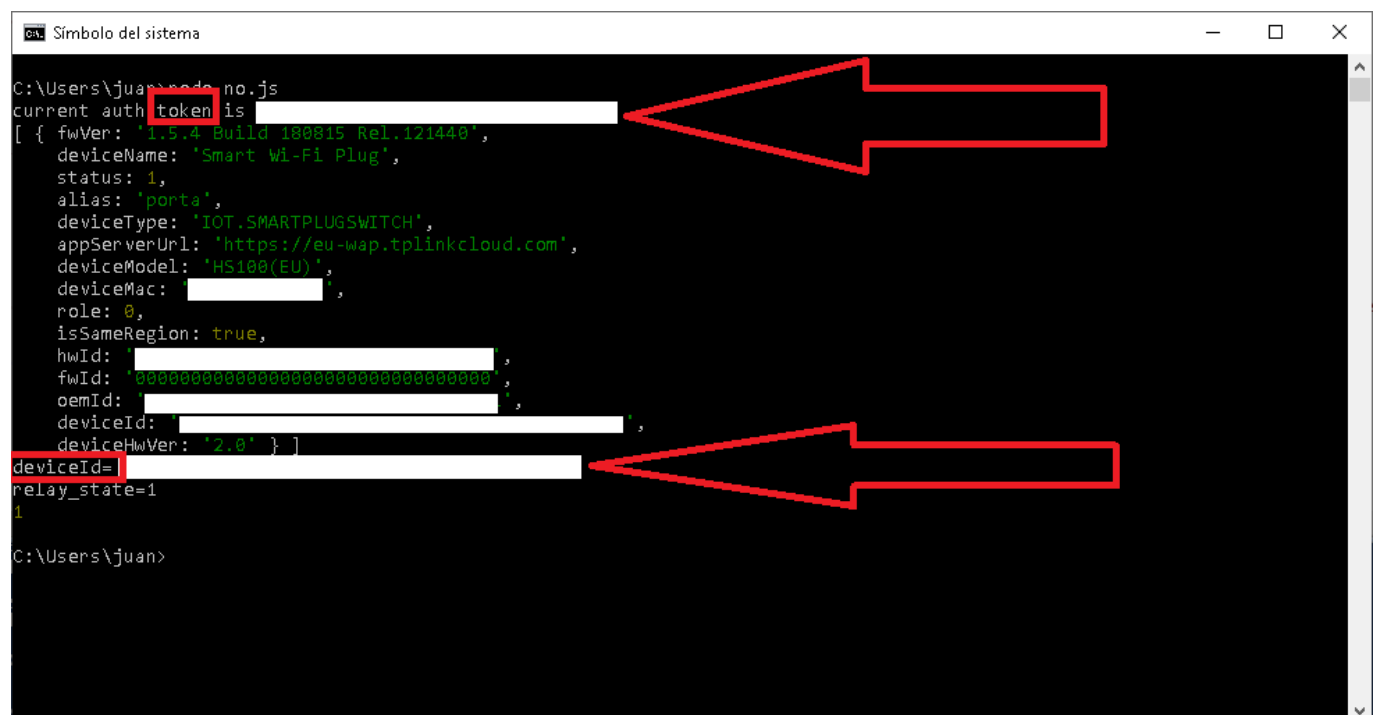


Apagar y encender el enchufe Inteligente TP-Link Wi-Fi HS100 desde PowerShell

El primer paso es obtener el token de autenticación por ejemplo desde el siguiente enlace Token de autenticación, se puede obtener con NodeJS <https://www.jesusnino.com/11/27/obtener-el-token-y-el-deviceid-del-enchufe-inteligente-tp-link-wi-fi-hs100-desde-node-js-y-mostrarlos-en-una-web-con-express>

[crayon-6a9d68bd1586e353236799/]

Obtener token del enchufe Inteligente TP-Link Wi-Fi HS100 con NodeJS



```
Símbolo del sistema
C:\Users\juan>node no.js
current auth token is [redacted]
[ { fwVer: '1.5.4 Build 180815 Rel.121440',
  deviceName: 'Smart Wi-Fi Plug',
  status: 1,
  alias: 'porta',
  deviceType: 'IOT.SMARTPLUGSWITCH',
  appServerUrl: 'https://eu-wap.tplinkcloud.com',
  deviceModel: 'HS100(EU)',
  deviceMac: '[redacted]',
  role: 0,
  isSameRegion: true,
  hwId: '[redacted]',
  fwId: '00000000000000000000000000000000',
  oemId: '[redacted]',
  deviceId: '[redacted]',
  deviceHwVer: '2.0' } ]
deviceId=[redacted]
relay_state=1
1
C:\Users\juan>
```

```
Windows PowerShell ISE
Archivo Editar Ver Herramientas Depurar Complementos Ayuda

Sin título1.ps1 X
1 # Token de autenticación, se puede obtener con NodeJS https://www.npmjs.com/package/tplink-cloud-api
2 $Token = "c0wErMwE-FSADFS0-ASASDF-ASDFSDf"
3
4 # URI del servicio appServerUrl: 'https://eu-wap.tplinkcloud.com'
5 $URI = 'https://eu-wap.tplinkcloud.com/?token='+$Token
6
7 # Cabeceras
8 $Cabeceras = @{
9     'Content-type' = 'application/json'
10 }
11
12 # Parámetros para encender y apagar
13 # Encender
14 ## '{"method":"passthrough", "params": {"deviceId": "YOUR_DEVICEID_HERE", "requestData": "{\\"system\\":{\\"set_relay_state\\":{\\"state\\":1}}}" }'
15 # Apagar
16 ## '{"method":"passthrough", "params": {"deviceId": "YOUR_DEVICEID_HERE", "requestData": "{\\"system\\":{\\"set_relay_state\\":{\\"state\\":0}}}" }'
17 $Body = '{"method":"passthrough", "params": {"deviceId": "24234241442353525435342", "requestData": "{\\"system\\":{\\"set_relay_state\\":{\\"state\\":1}}}" }'
18
19 # Preparar la petición y obtener respuesta
20 $Respuesta = Invoke-RestMethod -Method Post -Uri $URI -Headers $Cabeceras -Body $Body
21
22 # Convertir respuesta en formato JSON y obtener la edad de la persona de la imagen
23 ConvertTo-Json $Respuesta.result

# Preparar la petición y obtener respuesta
$Respuesta = Invoke-RestMethod -Method Post -Uri $URI -Headers $Cabeceras -Body $Body

# Convertir respuesta en formato JSON y obtener la edad de la persona de la imagen
ConvertTo-Json $Respuesta.result
{
    "responseData": "{\\"system\\":{\\"set_relay_state\\":{\\"err_code\\":0}}}"
}

PS C:\Users\juan>
```

Lín. 2 Col. 21 105 %