Obsah

Port 7777 — JSON-RPC server **1**
Parametry spojení 1
Filtrace přístupu 1
Transport 1
Metody (req) 2
Příklady 2
Odchozí protějšek 2
Smoke test 2
Rozšíření o novou metodu 2

Port 7777 — JSON-RPC server

Platí pro verzi NPA R04: 2.13.beta20 · Vytvořeno: 2026-06-02

> Zdroj: `njsrv.cpp / njsrv.h`, třída `NJSrv` (singleton `NJSrv::GetNJSrv()`), potomek `QTcpServer`. Inicializace `NJSrv::Init()`.

Strojové API mezi NPA jednotkami (a pro nástroje). Slouží zejména k vzájemnému párování jednotek přes VPN — vzdálená jednotka si přes něj zjistí parametry Modbus serveru protějšku (viz odchozí klient `NJCli` v `../interfaces.md`).

Parametry spojení

Parametr	Hodnota	Zdroj
Naslouchá na	`QHostAddress::Any` (0.0.0.0)	pevně
TCP port	**7777**	pevně (`Init()`)
Povolená síť	`10.80.0.0` / maska `255.255.0.0` (= `10.80.0.0/16`)	pevně (`mNet/mMask`)

Filtrace přístupu

V `incomingConnection()` se IP klienta porovná: `(ip & mMask) != (mNet & mMask)` → spojení je okamžitě odpojeno (`disconnectFromHost`). Projdou tedy **jen klienti ze sítě 10.80.0.0/16** (VPN/interní rozsah).

Transport

Přestože jde o „JSON-RPC", přenos je obalen do HTTP/1.1:

Požadavek (klient → server): - Server čte do `\r\n\r\n` (konec hlaviček), pak z hlavičky `Content-Length` načte délku těla a počká na kompletní tělo (buffer `mReceiveBuffer`). - Tělo je JSON objekt s povinným polem `req` (název metody).

Odpověď (server → klient):

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: <n>
Connection: close

{ ...JSON... }
```

Spojení se po odpovědi uzavírá (`disconnectFromHost`). Každá odpověď obsahuje: - `time` — UTC čas (`GetUTCTime()`), - `res` — výsledek (`"ok"` nebo výchozí `"command not found"`), - případná další pole specifická pro metodu.

Dispatch probíhá v `Req2Ans(QJsonObject)`.

Metody (`req`)

req	Vstupní parametry	Výstupní pole	Akce
`get_modbus_srv_param`	—	`mb_port` (`ns_netPort`), `mb_id` (`ns_netId`, výchozí 1), `res="ok"`	vrátí parametry lokálního Modbus TCP serveru
`set_time_log_sec`	`mTimeLogSec` (int > 0)	`res="ok"`	nastaví periodu remote logu (`NConfig::SetTimeLogSec`)

Neznámé `req` → `res = "command not found"`.

Příklady

Požadavek:

```
{ "req": "get_modbus_srv_param" }
```

Odpověď:

```
{ "time": "2026-06-02T10:00:00Z", "res": "ok", "mb_port": 502, "mb_id": 1 }
```

Požadavek:

```
{ "req": "set_time_log_sec", "mTimeLogSec": 30 }
```

Odchozí protějšek

Klient `NJCli` (`njcli.cpp`) volá toto API na jiných jednotkách: `POST http://<ip>:7777/`, `Content-Type: application/json`, synchronně přes `QEventLoop`. `NJCli::GetModbusSrvParam(ip)` vrací `mb_port`/`mb_id` vzdálené jednotky — využívá se při navazování Modbus TCP spojení v režimu `dd_id_npaid` (`DeviceModbus::NpaIdConnect`).

Smoke test

`test/test_njsrv.sh` — jediný `curl` proti portu 7777 živé jednotky (není to CI).

Rozšíření o novou metodu

Přidat větev `if (req == "...")` do `NJSrv::Req2Ans()`, naplnit výstupní pole a nastavit `response["res"] = "ok"`.