OSDPETH

Připojení čtečky OSDP přes RS485 k síti Ethernet

Převodník OSDP na Ethernet



Verze 1

19. října 2026

Obsah

1 Přehled změn 2
2 O zařízení 3
2.1 Bezpečnost síťového provozu zařízení 3
3 Zapojení 4
3.1 Ethernet, napájení a RS485 4
3.2 Rezistory BIAS a zakončení IMP 4
4 Uvedení do provozu 5
5 Webové rozhraní 6
5.1 Síť 6
5.2 Zabezpečení 7
5.3 HTTP / MQTT 7
5.4 OSDP protokol 7
5.5 Ostatní nastavení a ovládací tlačítka 8
6 Datová komunikace 9
6.1 Spinel přes TCP 9
6.2 Čtení poslední události přes HTTP 12
6.3 HTTP a MQTT 13
7 Diagnostika a obnovení nastavení 16
7.1 Hardwarový reset 16
8 Technické parametry 17

1 Přehled změn

Verze	Datum	Popis
1	19. 10. 2026	Popis zařízení, zapojení, nastavení a komunikace

2 O zařízení

OSDPETH připojuje čtečku s protokolem OSDP na lince RS485 do sítě Ethernet. Umožňuje předávat data karet a klávesnice do nadřazené aplikace.

OSDPETH funguje jako řídicí jednotka pro jednu čtečku: pravidelně ji dotazuje a přijímá data karty nebo klávesnice. Aplikace na Ethernetu získává události protokolem Spinel přes TCP, metodou HTTP POST nebo pomocí MQTT. Ovládání LED, bzučáku a výstupů čtečky lze doplnit na přání.

2.1 Bezpečnost síťového provozu zařízení

Zařízení je určeno výhradně pro provoz v důvěryhodné lokální síti (LAN) za síťovým routerem nebo firewallem. Není určeno pro přímé připojení k veřejné síti internet ani pro přímé zpřístupnění z externích sítí (např. pomocí veřejné IP adresy nebo přesměrováním portů).

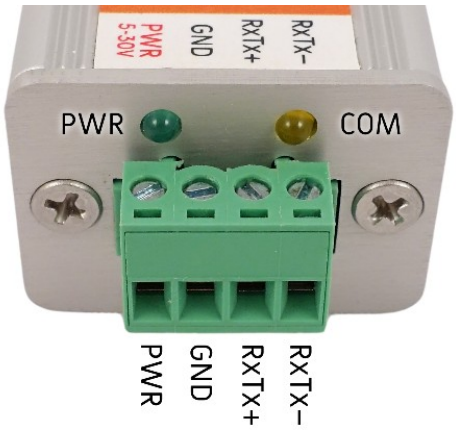
Webová konfigurace používá HTTP bez šifrování. Vestavěná autentizace slouží pouze jako základní ochrana proti neúmyslnému přístupu v lokální síti a nepředstavuje ochranu proti útokům z veřejné sítě. Pro vzdálený přístup použijte VPN nebo srovnatelný bezpečný mechanismus.

Volba OSDP Secure Channel se vztahuje ke komunikaci se čtečkou na RS485. Sama o sobě nezabezpečuje síťové rozhraní Ethernet.

3 Zapojení

3.1 Ethernet, napájení a RS485

Rozhraní Ethernet se připojuje konektorem RJ45 běžným nekříženým kabelem ke switchi. Napájení a linka RS485 jsou vyvedeny na odnímatelnou šroubovací svorku.



obr. 1: svorky pro připojení napájení a RS485

Označení	Význam
PWR	Kladný pól napájení, 5 až 30 V DC
GND	Záporný pól napájení
RxTx+	Kladný signál linky RS485
RxTx-	Záporný signál linky RS485

tab. 1: Připojení napájení a RS485

3.2 Rezistory BIAS a zakončení IMP

Uvnitř převodníku jsou propojky pro připojení rezistorů 680 Ω definujících klidový stav linky (BIAS) a zakončovacího rezistoru 120 Ω (IMP). Trvale jsou zapojeny rezistory 22 kΩ definující klidový stav.



obr. 2: propojky BIAS a IMP uvnitř převodníku

Zakončovací rezistory se připojují na opačných koncích linky RS485, nejvýše dva na jedné lince. Propojky BIAS připojte u jednoho zařízení na lince.

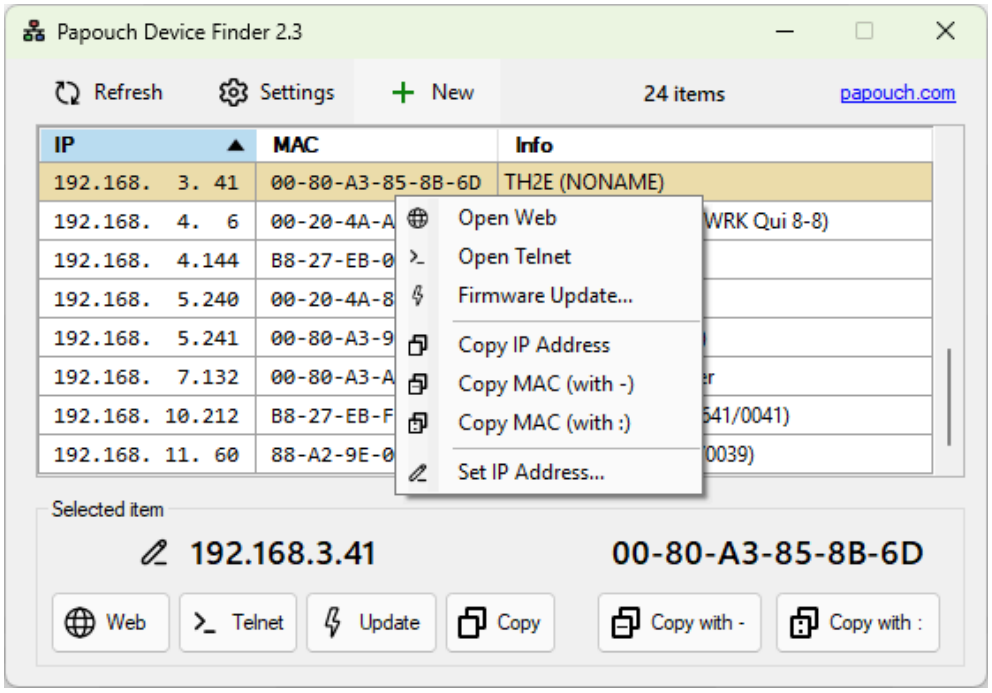
4 Uvedení do provozu

Výchozí síťové nastavení:

- IP adresa: **192.168.1.254**.
- Maska sítě: **255.255.255.0**.
- IP adresa brány: **0.0.0.0**.

IP adresu nastavte programem Papouch Device Finder:

1. Připojte OSDPETH k místní síti a připojte napájení. Počítač musí být připojen ke stejné LAN.
2. Spusťte Papouch Device Finder, vyberte síťové rozhraní připojené k této LAN a obnovte seznam tlačítkem **Refresh**.
3. Vyhledejte OSDPETH podle MAC adresy uvedené na štítku zařízení.
4. V nabídce zařízení vyberte **Set IP Address...** a zadejte IP adresu, masku a bránu podle parametrů vaší sítě. IP adresa musí být v síti unikátní.
5. Otevřete webové rozhraní v prohlížeči na nastavené IP adrese.



obr. 3: Papouch Device Finder – ukázka nabídky zařízení

Obrázek ukazuje nabídku programu na jiném typu zařízení od Papouch.com.

Výchozí port webu je 80; port datové komunikace Spinel je 10001. Z výroby není nastaveno heslo administrátora. Po jeho nastavení se používá přihlašovací jméno `admin`.

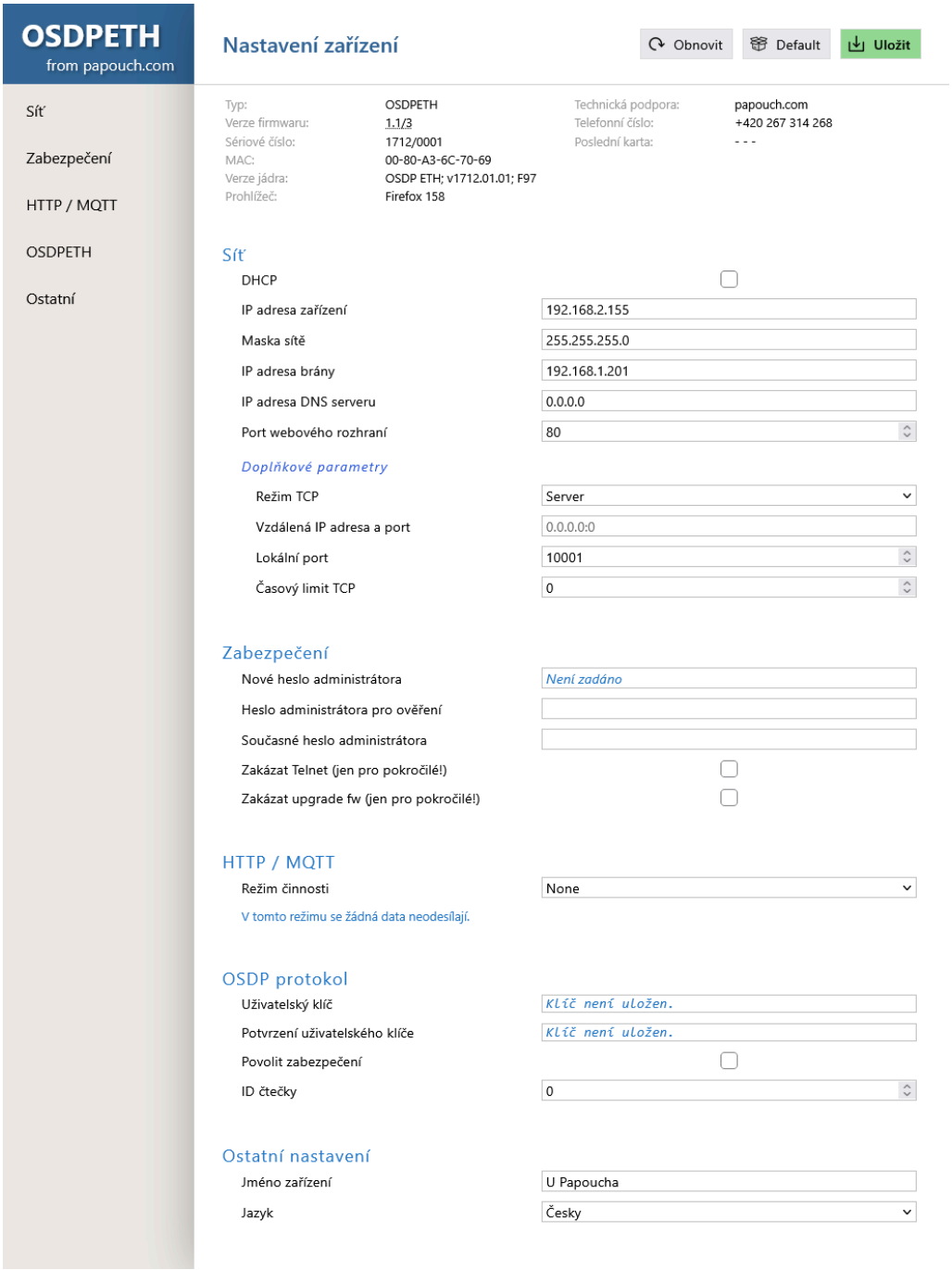
Pro čtečku použijte rychlost **38 400 Bd, 8 datových bitů, bez parity, 1 stopbit**.

Výchozí rychlost závisí na modelu čtečky. Pokud se liší, nastavte čtečku před připojením na uvedenou rychlost podle dokumentace jejího výrobce.

V sekci **OSDPETH** webového rozhraní nastavte **ID čtečky** podle OSDP adresy připojené čtečky a uložte nastavení. Adresa nastavená v převodníku a ve čtečce musí být shodná.

5 Webové rozhraní

Rozhraní má českou a anglickou variantu. Nabízí sekce Síť, Zabezpečení, HTTP / MQTT, OSDPETH a Ostatní. Horní část zobrazuje identifikaci zařízení a poslední událost čtečky.



obr. 4: hlavní stránka českého webového rozhraní OSDPETH

5.1 Síť

Nastavuje se DHCP, IP adresa zařízení, maska sítě, IP adresa brány, IP adresa DNS serveru a port webového rozhraní. Doplňkové parametry zahrnují režim TCP Server/Client, vzdálenou IP adresu a port, lokální port a časový limit TCP.

Časový limit TCP se zadává v sekundách. Hodnota 0 vypíná ukončování spojení podle tohoto limitu.

5.2 Zabezpečení

Rozhraní obsahuje nové heslo administrátora, jeho potvrzení a současné heslo administrátora. Samostatné volby umožňují zakázat Telnet a upgrade firmwaru.

Přihlašovací jméno je `admin`.

5.3 HTTP / MQTT

Vyberte jeden z režimů **None**, **HTTP** nebo **MQTT**. None vypíná odesílání na server nebo broker; HTTP odesílá události metodou POST a MQTT je publikuje do zvoleného topicu. Čtení přes Spinel a `fresh.json` je dostupné nezávisle na této volbě.

Nastavení	Význam
Server	IP adresa nebo doménové jméno a port, například `192.168.1.10:80` pro HTTP nebo `192.168.1.10:1883` pro MQTT. Nezadávejte předponu `http://` ani cestu.
Cesta / téma	V HTTP cesta ke skriptu, například `osdp/events`, nejvýše 100 znaků. V MQTT topic pro publikování, například `papouch/osdpeth/events`, nejvýše 50 znaků.
Šifrovací klíč	Pouze HTTP: volitelný klíč délky přesně 16 znaků z množiny a–z, A–Z a 0–9; zadává se také do potvrzovacího pole. Jde o samostatný klíč, odlišný od klíče OSDP.
QoS	Pouze MQTT: úroveň 0
Jméno uživatele	Pouze MQTT: jméno pro přihlášení k brokeru, nejvýše 15 znaků
Heslo a jeho potvrzení	Pouze MQTT: heslo pro přihlášení k brokeru, nejvýše 15 znaků

tab. 2: Nastavení HTTP a MQTT

Pro doménové jméno nastavte v sekci Síť dostupný DNS server. Pokud je cílový server v jiné síti, nastavte také správnou výchozí bránu.

Po změně parametrů stiskněte **Uložit**. Poté lze tlačítkem **Vyzkoušet odesílání** odeslat zprávu TEST. Příjem zprávy zkontrolujte na cílovém serveru nebo u odběratele MQTT. Datové formáty a příklady nastavení jsou v kapitole HTTP a MQTT.

5.4 OSDP protokol

Uživatelský klíč a **Potvrzení uživatelského klíče**: zadává se stejný 16bajtový klíč do obou polí, přesně jako 32 hexadecimálních znaků. Rozhraní převádí písmena na velká. Uložený klíč se při načtení nastavení nezobrazuje; pole informují, zda je uložen.

Uživatelský klíč slouží jako Master key. OSDPETH z něj a UID čtečky odvozuje klíč SCBK pro Secure Channel. Čtečka musí používat odpovídající klíč a stejný způsob odvození; její nastavení proveďte podle dokumentace výrobce. Do uvedených polí se nezadává již odvozený SCBK.

Povolit zabezpečení: zapíná OSDP Secure Channel. Pokud klíč dosud není uložen, rozhraní vyžaduje při povolení zabezpečení nový platný klíč.

ID čtečky: adresa čtečky v rozsahu 0 až 126.

Podporované události čtečky:

Typ	Význam	Podporovaná délka
0x50 RAW	Neformátovaná data karty	1 až 64 platných bitů; počet bajtů zaokrouhlen nahoru
0x51 FRM	Formátovaná data karty	1 až 16 bajtů
0x53 KEY	Data klávesnice	1 až 16 bajtů

tab. 3: Podporované události čtečky

5.5 Ostatní nastavení a ovládací tlačítka

Nastavuje se jméno zařízení a jazyk rozhraní. Tlačítko **Obnovit** znovu načítá nastavení; **Uložit** odesílá nastavení do zařízení. Tlačítko **Default** vyvolá potvrzovací dialog pro obnovení výchozího nastavení.

Webové obnovení ponechává IP adresu zařízení, ale obnovuje port webu na 80 a maže uživatelská nastavení včetně OSDP klíče. Pro úplné obnovení včetně IP adresy použijte hardwarový reset.

6 Datová komunikace

6.1 Spinel přes TCP

OSDPETH používá binární protokol Spinel, formát 97. Výchozí datový port je TCP **10001**. V režimu TCP Server navazuje spojení nadřazená aplikace k OSDPETH; v režimu TCP Client navazuje spojení OSDPETH k nastavené vzdálené IP adrese a portu. Režim a porty se nastavují v sekci Síť.

Pro sestavení a odesílání dotazů lze použít program Spinel terminál. Hexadecimální zápis v příkladech představuje jednotlivé binární bajty; mezery a oddělovače `|` slouží pro čitelnost a nejsou součástí přenosu.

Formát 97

Dotaz:

PRE	FRM	NUM HI	NUM LO	ADR	SIG	INST	DATA	SUM	CR

Odpověď:

PRE	FRM	NUM HI	NUM LO	ADR	SIG	ACK	DATA	SUM	CR

Pole	Délka [B]	Význam
PRE	1	Prefix `0x2A`, znak `*`
FRM	1	Formát 97, `0x61`
NUM	2	Počet bajtů od ADR do CR včetně; nejprve horní, potom dolní bajt
ADR	1	Adresa Spinel; používejte `0x31`, alternativně univerzální adresu `0xFE`. V odpovědích je `0x31`. Nejde o OSDP adresu čtečky.
SIG	1	Podpis dotazu `0x00` až `0xFF`; stejná hodnota se vrací v odpovědi
INST / ACK	1	Kód instrukce v dotazu nebo potvrzovací kód v odpovědi
DATA	N	Data podle konkrétní instrukce; pole může být prázdné
SUM	1	Kontrolní součet
CR	1	Zakončovací znak `0x0D`

tab. 4: Pole rámce Spinel

Je-li datová část dlouhá N bajtů, NUM = N + 5 a celý rámec má N + 9 bajtů. Dvoubajtová čísla v instrukcích i odpovědích se přenášejí s horním bajtem jako prvním, není-li u pole uvedeno jinak.

Kontrolní součet: sečtěte všechny bajty od PRE po poslední bajt DATA, ponechte dolních osm bitů součtu a výsledek odečtěte od `0xFF`. SUM ani závěrečné CR do tohoto součtu nevstupují.

```
SUM = 0xFF - ((PRE + FRM + NUM_HI + NUM_LO + ADR + SIG + INST/ACK + DATA) & 0xFF)
```

Na rámec s chybným kontrolním součtem nebo zakončením zařízení neodpovídá. TCP je proud bajtů: jeden rámec může dorazit v několika částech a při jednom čtení může být přijato více rámců. Při zpracování používejte pole NUM; samotný výskyt `0x0D` neurčuje konec rámce, protože tento bajt může být i v datech.

Příklad: dotaz na identifikaci, NUM = 5, adresa `0x31`, podpis `0x02`:

```
→  2A 61 00 05 31 02 F3 49 0D
```

Potvrzovací kódy

ACK	Význam
`0x00`	Instrukce provedena; následují případná data odpovědi
`0x02`	Neznámá instrukce
`0x03`	Neplatná délka nebo hodnota dat dotazu
`0x0C`	Automatická zpráva s novou událostí čtečky

tab. 5: Potvrzovací kódy Spinel

ACK popisuje zpracování instrukce. Příznak `status` uvnitř dat události má samostatný význam a není ukazatelem dostupnosti čtečky.

Přehled instrukcí

Instrukce	Kód	Kategorie	Strana
Čtení identifikace	`0xF3`	Doplňkové	10
Čtení poslední události	`0xA2`	Základní	10
Čtení nastavení automatických zpráv	`0xA4`	Konfigurační	11
Nastavení automatických zpráv	`0xB4`	Konfigurační	12

tab. 6: Instrukce Spinel

V následujícím popisu jsou odděleně uvedeny INST, ACK a datová pole. Úplné příklady rámců používají adresu `0x31` a podpis `0x02`, není-li uvedeno jinak.

Čtení identifikace

Vrací typ zařízení, verzi jádra a označení podporovaného formátu komunikace.

Dotaz: `0xF3`
Odpověď: `ACK 0x00` `řetězec`

Legenda:

Pole	Délka [B]	Popis
řetězec	Proměnná	ASCII řetězec, například `OSDP ETH; v1712.01.01; F97`; bez ukončovacího nulového bajtu

Příklad:

```
→  2A 61 00 05 31 02 F3 49 0D
```

```
←  2A 61 00 1F 31 02 00 4F 53 44 50 20 45 54 48 3B 20 76 31 37 31 32 2E 30 31 2E 30 31 3B 20
46 39 37 20 0D
```

Čtení poslední události

Přečte poslední přijatou událost karty nebo klávesnice. Po odeslání odpovědi nastaví příznak události na „již přečtena“. Nová událost přepíše předchozí uloženou hodnotu.

Dotaz: `0xA2` `slot`
Odpověď: `ACK 0x00` `slot` `status` `type` `representation` `bits` `bytes` `data`

Legenda:

Pole	Délka [B]	Popis
slot	1	Hodnota `0x01` v dotazu i odpovědi; interní slot; nejde o OSDP adresu čtečky
status	1	`0x00`: nová platná událost; `0x01`: dosud bez platné události nebo již přečteno přes A2
type	1	`0x50`: RAW, `0x51`: FRM, `0x53`: KEY; před první událostí `0x00`
representation	1	`0x01`: RAW data s počtem platných bitů; `0x00`: bajtová data FRM nebo KEY
bits	2	Počet platných bitů, horní bajt první
bytes	1	Počet následujících datových bajtů N
data	N	Data události; RAW nejvýše 8 bajtů, FRM nebo KEY nejvýše 16 bajtů

tab. 7: Data události Spinel

RAW data se přenášejí v pořadí bajtů přijatém ze čtečky; jejich význam a případné převedení na číslo karty závisí na jejím formátu. Počet bajtů RAW je počet bitů dělený osmi a zaokrouhlený nahoru. U FRM a KEY je počet bitů roven osminásobku počtu bajtů. Řídicí znaky odeslané čtečkou jsou součástí dat.

Příklad: nová 26bitová událost RAW, čtyři datové bajty `01 02 03 40`:

```
→ 2A 61 00 06 31 02 A2 01 98 0D
```

```
← 2A 61 00 10 31 02 00 01 00 50 01 00 1A 04 01 02 03 40 7B 0D
```

Příklad: odpověď se vstupem klávesnice `1234` zakončeným CR (`0x0D`), celkem pět bajtů / 40 bitů:

```
← 2A 61 00 11 31 02 00 01 00 53 00 00 28 05 31 32 33 34 0D D8 0D
```

Opakované čtení bez nové události vrací zachovaná data se statusem `0x01`. Před první událostí jsou type, bits a bytes nulové a datové pole je prázdné. Stav čtečky na RS485 nelze z příznaku status určit.

Čtení nastavení automatických zpráv

Dotaz: `0xA4`
Odpověď: `ACK 0x00` `enabled`

Legenda:

Pole	Délka [B]	Popis
enabled	1	`0x00`: automatické zprávy vypnuté; `0x01`: zapnuté

Příklad: automatické zprávy jsou zapnuté:

```
→ 2A 61 00 05 31 02 A4 98 0D
```

```
← 2A 61 00 06 31 02 00 01 3A 0D
```

Nastavení automatických zpráv

Zapne nebo vypne odesílání jednotlivých událostí bez opakovaných dotazů A2. Nastavení se ukládá do zařízení a zůstává zachováno po restartu. Z výroby jsou automatické zprávy vypnuté.

Dotaz: `0xB4` `enabled`
Odpověď: `ACK 0x00`

Legenda:

Pole	Délka [B]	Popis
enabled	1	`0x00`: vypnout; `0x01`: zapnout

Příklad: zapnutí automatických zpráv:

```
→ 2A 61 00 06 31 02 B4 01 86 0D
```

```
← 2A 61 00 05 31 02 00 3C 0D
```

Příklad: vypnutí automatických zpráv:

```
→ 2A 61 00 06 31 02 B4 00 87 0D
```

```
← 2A 61 00 05 31 02 00 3C 0D
```

Automatické zprávy

Nová událost čtečky při povoleném automatickém vysílání vyvolá zprávu s ACK `0x0C`. Zpráva přichází přes navázané TCP spojení bez předchozího dotazu.

Zpráva: `ACK 0x0C` `slot` `status` `type` `representation` `bits` `bytes` `data`

Datová část má stejnou strukturu jako odpověď A2. Podpis SIG se postupně mění; po `0xFF` následuje `0x00`. Není převzatý z dotazu a nepředstavuje trvalý identifikátor události přes restarty.

Příklad: nová 26bitová RAW data, podpis `0x00`:

```
← 2A 61 00 10 31 00 0C 01 00 50 01 00 1A 04 01 02 03 40 71 0D
```

Při novém načtení stejné karty přijde nová automatická zpráva se stejnými daty karty. Automatické zprávy lze používat současně s dotazy A2 a webovým rozhraním. Odeslání automatické zprávy samo nemění příznak přečtení poslední události; tento příznak mění čtení A2. Aplikace musí rozlišovat ACK `0x0C` od odpovědí na vlastní dotazy.

6.2 Čtení poslední události přes HTTP

Požadavek `GET /fresh.json` vrací poslední událost. Příklad odpovědi s 26bitovými RAW daty karty:

```
{
  "sn": "01712/00001",
  "slot": "1",
  "status": "0",
  "type": "50",
  "bits": "26",
  "bytes": "4",
  "data": "01020340"
}
```

Pole	Význam
sn	Sériové číslo převodníku; zachovejte řetězcový typ včetně nul na začátku
slot	Interní slot čtečky, v současné implementaci „1“; nejde o adresu OSDP
status	„0“: nová platná data; „1“: zatím bez platné události nebo již přečteno přes A2
type	Hexadecimální kód události: „50“ RAW, „51“ FRM, „53“ KEY
bits	Počet platných bitů, jako desetinný řetězec
bytes	Počet datových bajtů, jako desetinný řetězec
data	Data jako hexadecimální řetězec, dva znaky na bajt

tab. 8: Pole odpovědi fresh.json

Čtení `fresh.json` samo nemění stav události. Po přečtení instrukcí A2 zůstávají data v JSON odpovědi zachována a status se změní na „1“. Při tomto stavu data nepovažujte za novou událost. Webové rozhraní používá stejný endpoint.

Odpověď představuje poslední stav. Neobsahuje seznam událostí. Opakované načtení stejné karty může vrátit zcela shodný JSON včetně statusu „0“. Pro zpracování jednotlivých událostí včetně opakovaných přiložení použijte automatické zprávy Spinel.

Odpověď neobsahuje stav komunikace se čtečkou. Úspěšné HTTP čtení potvrzuje dostupnost převodníku. Připojení čtečky na RS485 tím neověříte.

6.3 HTTP a MQTT

OSDPETH odesílá události na nastavený server nebo MQTT broker. Název události `CARD` je společný pro kartu i klávesnici; konkrétní druh dat určuje pole `type`.

Formát události

```
{
  "sn": "01712/00001",
  "event": "CARD",
  "slot": "1",
  "status": "0",
  "type": "53",
  "bits": "40",
  "bytes": "5",
  "data": "313233340D"
}
```

V příkladu jde o klávesnici, pět bajtů `31 32 33 34 0D`, tedy text `1234` zakončený CR. Pole sn, slot, status, type, bits, bytes a data mají stejný význam jako v odpovědi fresh.json. Číselné údaje v JSON jsou řetězce. Událost

RAW obsahuje nejvýše 8 datových bajtů, FRM a KEY nejvýše 16; délka HEX řetězce data je vždy dvojnásobek hodnoty bytes.

Tlačítko **Vyzkoušet odesílání** odešle zprávu bez dat čtečky:

```
{
  "sn": "01712/00001",
  "event": "TEST"
}
```

Příjemce musí zprávu TEST odlišit od události CARD. TEST slouží ke kontrole cesty k serveru nebo brokeru a nepotvrzuje dostupnost čtečky na RS485.

Odesílání HTTP POST

Příklad nastavení pro server, který přijímá data na adrese `http://192.168.1.10/osdp/events`:

1. Vyberte režim **HTTP**.
2. Do pole **Server** zadejte `192.168.1.10:80`.
3. Do pole **Cesta / téma** zadejte `osdp/events`, bez úvodního lomítka. Zařízení lomítko doplňuje při tvorbě požadavku.
4. Pro přenos běžného JSON ponechte šifrování těla vypnuté. Případné šifrování nastavte podle následujícího oddílu.
5. Uložte nastavení a tlačítkem **Vyzkoušet odesílání** odešlete zprávu TEST.

Zařízení používá HTTP/1.1, metodu POST a hlavičku `Content-Type: application/json`. Příjemce čte JSON z těla požadavku. Po zpracování vraťte **HTTP 200 OK**; zařízení tuto odpověď vyhodnocuje jako úspěšné odeslání. Odpověď HTTP neslouží k zadávání příkazů pro LED, bzučák nebo výstupy čtečky.

Šifrování těla HTTP

Pro zapnutí zadejte v nastavení HTTP stejný 16znakový klíč do obou šifrovacích polí a uložte nastavení. Každý ze 16 zadaných znaků tvoří jeden bajt klíče; klíč se nezadává v hexadecimálním zápisu. Uložený klíč se nezobrazuje. Pro vypnutí uloženého šifrování vymažte obsah obou polí a změnu uložte.

Při zapnutém šifrování je běžný objekt CARD nebo TEST zašifrován pomocí AES-128 v režimu CFB bez paddingu (`aes-128-cfb`). Tělo požadavku obsahuje JSON objekt s jediným polem `crypt`. Jeho řetězec tvoří hexadecimálně zapsaný 16bajtový inicializační vektor následovaný hexadecimálně zapsaným šifrovaným obsahem.

Příjemce zpracuje pole `crypt` v tomto pořadí:

1. Převede HEX řetězec na bajty.
2. Oddělí prvních 16 bajtů jako inicializační vektor.
3. Zbývající bajty dešifruje AES-CFB s nastaveným 16bajtovým klíčem a odděleným vektorem.
4. Dešifrovaný obsah zpracuje jako běžný JSON události CARD nebo TEST.

Jako základ přijímací aplikace lze použít příklady dešifrování AES pro PAPAGO v PHP, Node.js a Node-RED. Pro OSDPETH upravte načtení vstupu: místo parametru GET `encrypted_data` načtěte JSON z těla HTTP POSTu a z něj pole `crypt`. Toto pole už obsahuje HEX řetězec bez znaků `%`; po převodu na bajty použijte stejné oddělení vektoru a dešifrování. Výsledkem je JSON objekt, který zpracujte JSON parserem.

Šifruje se pouze tělo odchozího HTTP požadavku. Webová konfigurace, HTTP hlavičky, čtení fresh.json a datové spojení Spinel tím zabezpečeny nejsou. Tato funkce nenahrazuje HTTPS ani ověřování protistrany.

Publikování MQTT

Příklad nastavení pro broker na `192.168.1.10`, port 1883, topic `papouch/osdpeth/events`:

1. Vyberte režim **MQTT**.
2. Do pole **Server** zadejte `192.168.1.10:1883`.
3. Do pole **Cesta / téma** zadejte `papouch/osdpeth/events`. Pro publikování nepoužívejte zástupné znaky `+` ani `#`.
4. Pokud broker vyžaduje přihlášení, vyplňte jméno uživatele, heslo a jeho potvrzení. Pro broker bez přihlašování ponechte údaje prázdné.
5. Uložte nastavení. V odběratelské aplikaci se přihlaste ke stejnému brokeru a topicu a vyzkoušejte odeslání zprávy TEST.

Payload je běžný JSON objekt CARD nebo TEST. Zařízení publikuje s **QoS 0** a **retain = 0**. Broker tedy nepotvrzuje jednotlivé publikace protokolem PUBACK a zpráva se neukládá jako retained hodnota pro nové odběratele. Odběratele připojte před vyvoláním události. Tento režim neposkytuje potvrzení doručení každé události až do cílové aplikace.

OSDPETH v tomto režimu publikuje data; nepřijímá z topicu příkazy k ovládání čtečky. Připojení MQTT není šifrované pomocí TLS a klíč pro šifrování HTTP se na něj nevztahuje. Pro přenos přes nedůvěryhodné sítě použijte zabezpečenou síťovou cestu podle kapitoly Bezpečnost síťového provozu.

7 Diagnostika a obnovení nastavení

7.1 Hardwarový reset

Hardwarový reset obnoví výchozí nastavení včetně IP adresy 192.168.1.254:

1. Odpojte napájení zařízení.
2. Stiskněte tlačítko RST v otvoru vlevo pod konektorem Ethernetu a držte jej stisknuté.
3. Připojte napájení a vyčkejte 10 sekund.
4. Uvolněte tlačítko.
5. Po restartu se připojte na výchozí IP adresu.

Reset vymaže také uživatelský OSDP klíč a nastavení zabezpečení. Před obnovením provozu znovu nastavte síť, čtečku a požadované zabezpečení.

8 Technické parametry

Ethernet

Parametr	Hodnota
Připojení	RJ45 Ethernet 10/100BASE-T
Kompatibilita rozhraní	Ethernet verze 2.0 / IEEE 802.3

Napájení a RS485

Parametr	Hodnota
Napájecí napětí	5 až 30 V DC
Proudový odběr při 12 V	Typicky 80 mA
Proudový odběr při 5 V	Typicky 190 mA
Konektor	Odnímatelná šroubovací svorka
Signály RS485	RxTx+, RxTx-
Parametry RS485	38 400 Bd, 8 datových bitů, bez parity, 1 stopbit
Galvanické oddělení portu RS485	Ano
Galvanické oddělení napájení	Ne
Zakončení	120 Ω, připojitelné propojkou IMP
Definice klidového stavu	22 kΩ trvale; 680 Ω připojitelné propojkami BIAS

Mechanické a okolní podmínky

Parametr	Hodnota
Hmotnost	60 g
Stupeň krytí	IP30
Pracovní teplota	−25 až +75 °C
Pracovní vlhkost	0 až 90 %RH, nekondenzující
Rozměry bez připojené svorky	54 mm × 24 mm × 33 mm
Rozměry s připojenou svorkou	63 mm × 24 mm × 33 mm

Poznámky:

Poznámky:

Papouch s.r.o.

Přenosy dat v průmyslu, převodníky linek a protokolů, RS232, RS485, RS422, USB, Ethernet, LTE, WiFi, měřicí moduly, inteligentní teplotní čidla, I/O moduly, zakázkový vývoj a výroba.

Adresa: Strašnická 3164/1a, 102 00 Praha 10
Telefon: +420 267 314 268
Internet: www.papouch.com
E-mail: papouch@papouch.com

© 2026 Papouch s.r.o.