

KVERKOM - NOP Lite - Integrovaný manuál

- Verzia dokumentu
- Prehľad a účel
 - Prostredia a DNS
 - Bezpečnosť a certifikáty (mTLS)
 - Certifikačná politika
 - Pre pokladnice/klientsky SW (erp-api)
 - Pre banky (bank-api)
 - Identita klienta
 - Postup získania/predloženia certifikátu
 - REST API (HTTPS)
 - Štandardy a verziovanie
 - Auth (mTLS)
 - Hlavičky cez F5 / proxy
 - Limity a time-outy
 - Chybové kódy
 - MQTT (TLS, mTLS)
 - Protokol a vlastnosti
 - Identita a autorizácia
 - Štruktúra topicov
 - Kvalita služby (QoS) a doručovanie
 - Dátové formáty a konvencie
 - Monitoring, metriky a health-check
 - Prevádzkové zásady
 - Mapovanie chýb
 - Diagnostika
- Príklady volaní a základné testovacie scenáre
 - Predpoklady
 - Testovanie z príkazového riadku
 - Čo budeme potrebovať
 - Overenie platnosti certifikátu
 - Overenie dostupnosti serverov a služieb
 - Scenár - okamžitá platba na pokladni
 - Vygenerovanie (nového) transactionId
 - Čakanie na notifikáciu
 - Zaslanie notifikácie
 - Prevzatie notifikácie
 - Testovanie použitím python skriptu kverkom_test.py
 - Čo budeme potrebovať
 - Inštalácia skriptu kverkom_test.py
 - Popis skriptu
 - Overenie dostupnosti serverov a služieb
 - Scenár - okamžitá platba na pokladni
- Referencie a prílohy

Verzia dokumentu

Verzia	Dátum	Popis zmeny
0.1	14. 8. 2025	Iniciálny dokument
0.2	18. 8. 2025	Upravená formálna štruktúra
0.3	27. 8. 2025	Doplnené vybrané prevádzkové údaje pre integračné prostredie
1.0	16. 10. 2025	Malé textové úpravy
1.1	21. 10. 2025	Úprava textu v súvislosti s certifikátmi (pridané vysvetlenie k príkladom)

1.2	21. 10. 2025	Doplnenie kontaktných údajov na podporu
1.3	3. 11. 2025	Kontakty na podporu počas uzatvoreného pilotu
1.4	28. 11. 2025	Pridaná poznámka ku <i>getTransactionHistory</i> ; aktualizované swagger/openapi prílohy
1.5	8. 12. 2025	Aktualizovaná informácia ku <i>getTransactionHistory</i>
1.6	21. 1. 2026	Doplnená a spresnená informácie o Date hlavičke pre POST /payments; doplnené aj informácie o mTLS a certifikátoch
1.7	29. 1. 2026	Spresnené linky na openapi specification

Prehľad a účel

- Toto API umožňuje bezpečný prístup k službám Notifikátor okamžitých platieb (NOP) v rámci projektu KVERKOM cez **REST (HTTPS)** a **MQTT over TLS**.
- Prístup je chránený **vzájomným TLS (mTLS)** pomocou klientskych certifikátov.
- Cieľ: poskytnúť technické pokyny, testovacie scenáre a pravidlá prevádzky pre partnerov.

Kontakty na podporu:

- Incidendy: počas trvania uzatvoreného pilotu, do 15.1.2026 kverkom.kasoveIS@financnasprava.sk
- Onboarding počas uzatvoreného pilotu: obchodník požiadava cez kverkom.kasoveIS@financnasprava.sk o zaslanie Dohody o spolupráci a po obojstrannom podpise bude obchodníkovi umožnený prístup
- Prevádzková doba: 24x7 a 365 dní v roku, s výnimkou plánovanej doby odstávky IS, oznámenou 7 kalendárnych dní pred jej uskutočnením
- SLA (podpora počas uzatvoreného pilotu): 8x5

Prostredia a DNS

Prostredie	API-BANKA Base URL (REST)	API-ERP Base URL (REST)	MQTT host/port	Poznámka
INT (integračné)	<a href="https://<api-banka-i.kverkom.sk>">https://<api-banka-i.kverkom.sk>	<a href="https://<api-erp-i.kverkom.sk>">https://<api-erp-i.kverkom.sk>	mqtt-i.kverkom.sk :8883	Prostredie určené pre integračné testy klientov (pokladnic) aj bánk.
PROD (produkčné)	<a href="https://<api-banka.kverkom.sk>">https://<api-banka.kverkom.sk>	<a href="https://<api-erp.kverkom.sk>">https://<api-erp.kverkom.sk>	mqtt.kverkom.sk :8883	Produkčné prostredie (v príprave)

- **SNI:** vyžadujeme správny hostname v TLS (SNI) zhodný s uvedenými doménami
- **IP whitelisting:**
 - Pre INT prostredie je rozsah IP obmedzený na Slovensko, Európu a USA
 - Pre PROD prostredie je rozsah IP obmedzený na Slovensko, Európu a USA
- **Identity whitelisting:**
 - Pre INT prostredie neplánujeme whitelisting na základe identity/certifikátu.
 - Pre PROD prostredie plánujeme **po dobu uzavretého testu** nasadiť whitelist podľa **VATSK** (z certifikátu pokladnice /klienta) a **organizationIdentifier** (z certifikátu banky).
- **Časová zóna systému:** Europe/Bratislava

Bezpečnosť a certifikáty (mTLS)

Certifikačná politika

Pre pokladnice/klientsky SW (erp-api)

- Akceptované klientské CA: **DigiCert Global Root G2** (root) a **GeoTrust TLS RSA CA G1** (intermediate)

- Cert formáty: **PEM** (.cert + .key) alebo **PKCS#12** (.p12/.pfx)
- Podporované TLS: **TLS 1.2+** (odporúčané 1.3)
- Certifikáty vydáva FR SR, a zabezpečuje OSCP/CRL, a ich expiráciu/obnovu nastavuje FR SR - viď napr. <https://www.info-rplaty.sk/> alebo <https://www.financnasprava.sk/>

Pre banky (bank-api)

- Na overenie identity banky sa používa QWAC certifikát (viď napr. <https://eidas.ec.europa.eu/efda/discover/qwac>)
- Banka si musí zabezpečiť klientsky QWAC certifikát nezávisle, do FR SR iba zašle informáciu s údajmi nižšie (kvôli white-listingu)
- Čo minimálne NOP kontroluje/overuje/validuje v klientskom certifikáte (pri volaní služby NOP):
 - **Dátumy** platnosti (napr. či certifikát neexpiroval)
 - **Identitu CA** ktorá certifikát vydala (voči zoznamu trusted CA)
 - Či certifikát obsahuje napr. "**Policy: 0.4.0.194112.1.4**" (teda či ide o QWAC certifikát)
 - Či je uvedený **organizationIdentifier** na NOP whiteliste; tento identifikátor na Slovensku ho bankám vydáva/prideľuje NBS, v rámci eIDAS/PSD2 aktivít; technicky je možné akceptovať aj iný identifikátor (po dohode s FR SR)

Príklad certifikátu (extrakt z dump-u klientskeho certifikátu typu QWAC):

```
Certificate:
  Data:
    Version: 3 (0x2)
    Serial Number:
      03:ee:35:27:28:ce:83:19:1a
    Signature Algorithm: sha256WithRSAEncryption
    Issuer: CN = I.CA TLS EV CA/RSA 06/2022, O = "Prvn\C3\AD certifika\C4\8Dn\C3\AD autorita, a.s.",
organizationIdentifier = NTRCZ-26439395, C = CZ
    Validity
      Not Before: Jul 18 13:48:30 2025 GMT
      Not After : Jul 18 13:48:30 2026 GMT
    Subject: C = SK, ST = Bratislavsk\C3\BD, L = Bratislava, O = "Banka ABC, a.s.", jurisdictionC = SK,
serialNumber = 00631630, organizationIdentifier = PSDSK-NBS-00666666, CN = abc.sk
    ...
    X509v3 extensions:
      ...
      X509v3 Certificate Policies:
        ...
        Policy: 0.4.0.194112.1.4
```

Identita klienta

- Identita klienta sa mapuje z certifikátu podľa:
 - pre **API-ERP a MQTT** (teda pre klienta typu "pokladnica"): **Subject**: C = SK, OU = XXXXXXXXXXXXXXXXXXXX, CN = **VATSK-XXXXXXXXXX POKLADNICA XXXXXXXXXXXXXXXXXXXX**
 - z ktorej sa odvodí subjekt VATSK-1234567890 a pokladnica POKLADNICA-88812345678900001
 - a tieto údaje sa používajú na autorizáciu volaní a MQTT topic ACL
 - pre **API-BANKA** (teda pre klienta typu "banka"): "**organizationIdentifier**" a "**O**" z QWAC klientskeho certifikátu
 - pričom "organizationIdentifier" bude použitý na identifikáciu voči whitelistu
 - a "O" bude slúžiť pre zrozumiteľnosť logov/auditných informácií

Postup získania/predloženia certifikátu

- Pre "pokladnice": využívajú sa rovnaké certifikáty ako pre systém eKasa
- Pre "banky": využívajú sa QWAC certifikáty v zmysle kapitoly 3 v štandarde <https://www.sbaonline.sk/projekt/standard-sluzby-notifikacie/>

REST API (HTTPS)

Štandardy a verziovanie

- **OpenAPI**: (viď prílohy)
- **Verziovanie URI**: napr. /api/v1/... (major verzia v ceste).
- **Formát**: JSON (UTF-8).
- **Idempotencia**: (viď jednotlivé služby v [NOP Lite - NOP Services API](#))

Auth (mTLS)

- Ak je klientsky certifikát neplatný/chýba: **HTTP 401**
- Ak identita nemá oprávnenie: **HTTP 403**

Hlavičky cez F5 / proxy

- Forwardované hlavičky: X-Forwarded-For, X-Forwarded-Proto, X-Request-ID
- (do klienta odporúčame poslať vlastný X-Correlation-ID)

Limity a time-outy

Parameter	Hodnota
Max. veľkosť request body	(TBD) <napr. 5 MB>
Read/Idle timeout	30s
Rate limit	(TBD) <napr. 600 req/min/klient>
Burst	(TBD) <napr. 120 req>

Chybové kódy

Kód	Popis	Príklad code	message
400	Neplatná požiadavka	INVALID_PAYLOAD	„Missing field ...“
401	Chýba/nesedí klientsky cert	MTLS_REQUIRED	„Client certificate required“
403	Zakázané	FORBIDDEN	„Not authorized for resource“
404	Nenašlo sa	NOT_FOUND	„Resource not found“
405	Nedovolená operácia (GET, POST...)	Method Not Allowed	„Method Not Allowed“
409	Konflikt	CONFLICT	„Duplicate“
415	Nepodporovaný typ	UNSUPPORTED_MEDIA_TYPE	„Use application/json“
422	Nevalidné dáta	VALIDATION_ERROR	„Field X ...“
429	Príliš veľa požiadaviek	RATE_LIMITED	„Retry later“
5xx	Interná chyba	INTERNAL_ERROR	„Try again“

MQTT (TLS, mTLS)

Protokol a vlastnosti

- **MQTT 3.1.1**
- Transport: **TLS na porte 8883**, mTLS povinné.
- **KeepAlive: 60 s**
- **Max payload:** (TBD, napr. 256 kB)
- **Retained messages: áno, max 2 hod**
- **QoS:** podporujeme **0 a 1**
- **Will message (LWT): N/A**

Identita a autorizácia

- Pri TLS handshaku identifikujeme klienta z certifikátu (vid' *Identita klienta* vyššie)
- ACL: prístup na publish/subscribe je viazaný na identitu (cert) a topic patterny

Štruktúra topicov

V rámci MQTT brokera v NOP existujú nasledovné topics ("témy"):

Pre SUBSCRIBE:

- **VATSK-XXXXXXXX/POKLADNICA-XXXXXXXXXXXXXXXX/QR-XXXXXXXXXXXXXXXXXXXXXXXX**
 - prijíma správy pre dané konkrétne ID transakcie (QR-XXXXXXXXXXXXXXXXXXXXXXXX)
 - príklad: **VATSK-1234567890/POKLADNICA-88812345678900001/QR-ab29e346f1d841c8a95a63d857490818**
- **VATSK-XXXXXXXX/POKLADNICA-XXXXXXXXXXXXXXXX/#**
 - prijíma správy pre konkrétnu pokladnicu (klienta) ale akúkoľvek transakciu
 - príklad: **VATSK-1234567890/POKLADNICA-88812345678900001/#**
- **VATSK-XXXXXXXX/#**
 - prijíma správy pre všetky pokladnice v rámci daného DIČ
 - príklad: **VATSK-1234567890/#**

Pre PUBLISH:

- **TRANSACTIONS/VATSK-XXXXXXXX/POKLADNICA-XXXXXXXXXXXXXXXX**
 - po zápise správy v tvare {"request": "transaction_id"} systém vygeneruje nové transactionId a publikuje do MQTT na topic VATSK-XXXXXXXX/POKLADNICA-XXXXXXXXXXXXXXXX správu formáte popísanom v službe *generateNewTransactionId*.
 - ide o alternatívu ku HTTP/REST službe *generateNewTransactionId*

Kvalita služby (QoS) a doručovanie

- **QoS 1:** garantuje „aspoň raz“, klient musí implementovať de-duplikáciu.
- **Retain:** len na kanáloch **VATSK-XXXXXXXX/#**, max TTL 2h

Dátové formáty a konvencie

- **JSON** kľúče v camelCase
- Čas: **ISO-8601** UTC (2025-08-27T09:00:00Z).
- Číselné formáty: desatinná bodka.
- Identifikátory: **N/A**
- Schémy: Vid' dokument [NOP Lite - NOP Services API](#)

Monitoring, metriky a health-check

- REST: **GET /api/v1/status** (bez citlivých údajov)
- MQTT: publish spravy "test/ping" na topic "test/ping"

Prevádzkové zásady

- **Backoff & retry:** exponenciálny backoff (napr. 1s, 2s, 4s, max 30s), max 5 pokusov.
- **Zmeny kontraktu:** backward-compatible 6 mesiacov; breaking zmeny vo verzii **v{n+1}**.
- **Deprekácia:** oznámenie **min. 90 dní** vopred.
- **Údržbové okná:** (TBD)
- **GDPR/PII:** neposielať senzitivitu bez právneho základu; dáta v transporte vždy cez TLS.
- **Časové synchronizácie:** NTP (drift < **<200 ms>**).

Mapovanie chýb

(viď samostatný dokument [NOP Lite - NOP Services API](#))

Diagnostika


Pre účely diagnostiky priebehu transakcie a trasovaní notifikácie odporúčame využiť (verejne dostupnú) službu `getTransactionHistory`; jej popis viď samostatný dokument [NOP Lite - NOP Services API](#)

Pre rýchlu kontrolu QR-platieb resp. notifikácií môže byť užitočná jednoduchá webová aplikácia (verejne prístupná) využívajúca službu `GET getTransactionHistory` : <https://www.kdejemojaplatba.sk>

Príklady volaní a základné testovacie scenáre

Uvádzame dva spôsoby rýchleho otestovania rozhraní a funkcionality NOP:

- Testovanie použitím command-line príkladov (`curl`, `mosquitto`) - použiteľné minimálne na Windows aj Linux,
- a tiež testovanie použitím pripraveného python skriptu (`kverkom_test.py`) pre tento účel - použiteľné na všetkých platformách s Python3.

 Všetky príklady sú uvádzane pre **integračné prostredie**.

Predpoklady

Predpokladáme, že:

- Všetky hosty a porty sú dostupné z internetu (zo slovenských IP adries, resp. z Európy a USA), konkrétne:
 - api-banka-i.kverkom.sk:443
 - api-erp-i.kverkom.sk:443
 - mqtt-i.kverkom.sk:8883
- Tester disponuje platným certifikátom; konkrétny formát a názov certifikátu závisí od implementácie klienta, ale **v príkladoch používame tieto konkrétne názvy** (a formáty):
 - **"kverkom-int-client.pem"** - klientsky certifikát v PEM formáte; pre ERP-API a MQTT ide o "pokladničný" certifikát vydaný FRSR, pre BANK-API ide o QWAC certifikát (ale pre neprodučné prostredia su akceptované aj všetky certifikaty platné pre ERP-API)
 - **"kverkom-int-client.key"** - kľúč k certifikátu
 - **"kverkom-ca-bundle.pem"** - serverový certifikát (bundle) v PEM formáte
- pričom v certifikáte (`kverkom-int-client`) je uvedený atribút "CN" vo formáte **napr.**
 - "Subject: C = SK, OU = 88812345678900001, CN = VATSK-1234567890 POKLADNICA 88812345678900001"
 - čo oprávňuje testera volať služby pre subjekt **VATSK-1234567890** a pokladnicu **POKLADNICA-88812345678900001**

Testovanie z príkazového riadku

Čo budeme potrebovať

- Windows 10+ alebo Linux (aktuálny)
- Nástroj "curl" (pre volanie HTTP/REST služieb) - v Linuxe je by default, vo Windows sa da instalovat napr. z <https://curl.se/download.html>
- Nástroj "mosquitto client" (pre volanie MQTT subscribe resp. publish) - <https://mosquitto.org/download/>
- (voliteľne) openssl - v Linuxe je by default, vo Windows sa da inštalovať napr. z <https://slproweb.com/products/Win32OpenSSL.html>

Overenie platnosti certifikátu

Je vhodné overiť platnosť certifikátov pred samotným testovaním, napr. takto:

```
openssl s_client -connect api-erp-i.kverkom.sk:443 -cert kverkom-int-client.pem -key kverkom-int-client.key -CAfile kverkom-ca-bundle.pem
```

Openssl by nemal vypísať žiadnu chybu, resp. mal by napísať - okrem iného - aj "Verification: OK".

Overenie dostupnosti serverov a služieb

Príkazy ktoré volajú "status" operácie na príslušných rozhraniach:

```
curl -s -S -i -X GET https://api-erp-i.kverkom.sk/api/v1/status --cert kverkom-int-client.pem --key kverkom-int-client.key --cacert kverkom-ca-bundle.pem
curl -s -S -i -X GET https://api-banka-i.kverkom.sk/api/v1/status --cert kverkom-int-client.pem --key kverkom-int-client.key --cacert kverkom-ca-bundle.pem
```

A ich typický výstup:

```
{ "timestamp": "2025-08-20T12:49:08.808Z", "instance": "kverkom-erp-api", "build": "2025-08-19", "version": "0.2.1" }
{ "timestamp": "2025-08-20T12:49:15.224Z", "instance": "kverkom-bank-api", "build": "2025-08-19", "version": "0.2.1" }
```

Kontrola MQTT servera (vykoná sa zápis na fiktívny/neexistujúci topic, ktorý sa na serveri v skutočnosti nevykoná, ale overí to jeho dostupnosť):

```
mosquitto_pub -h mqtt-i.kverkom.sk -p 8883 -t "test/ping" -m 'ping' --cafile kverkom-ca-bundle.pem --cert kverkom-int-client.pem --key kverkom-int-client.key -d
```

Typický výstup:

```
Client (null) sending CONNECT
Client (null) received CONNACK (0)
Client (null) sending PUBLISH (d0, q0, r0, m1, 'test/ping', ... (9 bytes))
Client (null) sending DISCONNECT
```

Scenár - okamžitá platba na pokladni

Vygenerovanie (nového) transactionId

Pokladňa (ERP) si nechá vygenerovať **transactionId** - napr. príkazom (zobrazujeme príkaz, aj typicky výstup vrátane "debug" informácií, ako dôsledok prepínača "-i"):

```
curl -s -S -i -X POST https://api-erp-i.kverkom.sk/api/v1/generateNewTransactionId --cert kverkom-int-client.pem --key kverkom-int-client.key --cacert kverkom-ca-bundle.pem

HTTP/1.1 200 OK
server: gunicorn
date: Sat, 16 Aug 2025 20:17:30 GMT
content-type: application/json
content-length: 99
x-content-type-options: nosniff
referrer-policy: same-origin
cross-origin-opener-policy: same-origin
set-cookie: c9c66cbdae9d166f0cbecef9f5e248f0=ff0d1077a262d071c5bcebbbc0f645d9; path=/; HttpOnly

{"transaction_id": "QR-01c40ef8bb2541659c2bd4abfb6a9964", "created_at": "2025-08-16T20:17:30.345Z"}
```

Dôležitý je posledný riadok výstupu, a v ňom atribút "transaction_id": "QR-01c40ef8bb2541659c2bd4abfb6a9964".

Čkanie na notifikáciu

Predpokladáme, že `transaction_id` sa použije na platbu, a platba prebehne mimo systémov FRSR.

Ďalším krokom v rámci testovania FRSR/NOP je až doručenie notifikácie z Banky (ktorá platbu prijala) do ERP, preto ERP sa subscribne na topic v MQTT brokerovi, napr. takto:

```
mosquitto_sub -h mqtt-i.kverkom.sk -p 8883 -v -q 1 -t "VATSK-1234567890/POKLADNICA-88812345678900001/QR-01c40ef8bb2541659c2bd4abfb6a9964" --cafile kverkom-ca-bundle.pem --cert kverkom-int-client.pem --key kverkom-int-client.key -d
```

pričom na výstupe uvidíme

```
Client (null) sending CONNECT
Client (null) received CONNACK (0)
Client (null) sending SUBSCRIBE (Mid: 1, Topic: VATSK-1234567890/POKLADNICA-88812345678900001/QR-01c40ef8bb2541659c2bd4abfb6a9964, QoS: 1, Options: 0x00)
Client (null) received SUBACK
Subscribed (mid: 1): 1
```

Klient (`mosquitto_sub` v roli "ERP") teraz **čaká na notifikáciu** ("z banky") pre konkrétnu transakciu (QR-01c40ef8bb2541659c2bd4abfb6a9964).



Príkaz je možné spustiť aj tak, že klient nebude čakať na konkrétnu transakciu, ale na ľubovoľnú transakciu pre danú pokladnicu, napr.

```
mosquitto_sub -h mqtt-i.kverkom.sk -p 8883 -v -q 1 -t "VATSK-1234567890/POKLADNICA-88812345678900001/#" --cafile kverkom-ca-bundle.pem --cert kverkom-int-client.pem --key kverkom-int-client.key -d
```

alebo dokonca na ľubovoľnú transakciu na akejkoľvek pokladnici v rámci daného subjektu (VATSK):

```
mosquitto_sub -h mqtt-i.kverkom.sk -p 8883 -v -q 1 -t "VATSK-1234567890/#" --cafile kverkom-ca-bundle.pem --cert kverkom-int-client.pem --key kverkom-int-client.key -d
```

Zaslanie notifikácie

V inom okne/konzole (tentoraz v roli "banky") vygenerujeme notifikáciu, na ktorú pokladnica (v roli "ERP") čaká:

```
curl -s -S -i -X POST https://api-banka-i.kverkom.sk/api/v1/payments \
-H "Content-Type: application/json" \
-H "X-Request-ID: 84c457b1-048b-4ca3-8df6-6e965c866333" \
-H "Date: 2026-01-21T09:48:11Z" \
--data '{"transactionStatus": "ACCC", "transactionAmount": {"currency": "EUR", "amount": "667.00"}, "endToEndId": "QR-01c40ef8bb2541659c2bd4abfb6a9964", "dataIntegrityHash": "ABCDEF"}' \
--cert kverkom-int-client.pem --key kverkom-int-client.key --cacert kverkom-ca-bundle.pem
```

Príkaz nevracia žiadnu hodnotu (okrem prípadných debug výpisov).



Do poľa "`dataIntegrityHash`" je potrebné vyplniť hodnotu vypočítanú podľa štandardu <https://www.sbaonline.sk/projekt/standard-sluzby-notifikacie/>



V hlavičke volania je nutné vyplniť "Date" v zmysle štandardu <https://www.sbaonline.sk/projekt/standard-sluzby-notifikacie/> - hodnota je povinná, a formát je **ISO 8601 timestamp v UTC** (teda napr. "2025-05-28T00:20:00Z" a nie RFC 7231 / RFC 9110, napr. "Tue, 13 Jan 2026 10:32:50 GMT"). Rovnaký formát pre Date header je použitý aj v odpovedi (Response Header) z NOP - tiež v zmysle Štandardu.

Prevzatie notifikácie

Na konzole s bežiacim mosquito_sub klientom (v roli "ERP") za zobrazí takýto výpis:

```
Client (null) received PUBLISH (d0, q0, r0, m0, 'VATSK-1234567890/POKLADNICA-88812345678900001/QR-01c40ef8bb2541659c2bd4abfb6a9964', ... (220 bytes))
VATSK-1234567890/POKLADNICA-88812345678900001/QR-01c40ef8bb2541659c2bd4abfb6a9964 {"transactionStatus": "ACCC", "transactionAmount": {"currency": "EUR", "amount": "667.00"}, "endToEndId": "QR-01c40ef8bb2541659c2bd4abfb6a9964", "dataIntegrityHash": "ABCDEF", "receivedAt": "2025-08-16T20:43:37.562311Z"}
```

ktorý znamená, že klient (v roli "ERP") úspešne prijal notifikáciu o platbe.

Týmto je tento základný (a najdôležitejší) scenár ukončený.

Testovanie použitím python skriptu kverkom_test.py

Čo budeme potrebovať

- Nainštalovaný python (verzia 3.7+)

Inštalácia skriptu kverkom_test.py

- Stiahnuť balík so skriptom - vid príloha [kverkom_test_v1.zip](#)
- Rozbaliť v lokálnom adresári
- Nainštalovať potrebné knižnice
 - pri použití **virtualenv** (odporúčané):

```
cd /path/to/your/project
python -m venv .venv
source .venv/bin/activate
python -m pip install --upgrade pip
python -m pip install -r requirements.txt
```

- pri **globálnej inštalácii** (jednoduchšie):

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

- Do podadresára "cert" je potrebné nakopírovať (platné) certifikáty (vid [Predpoklady](#) vyššie), pričom ich názvy a typy musia korešpondovať s nastavením CLIENT_CERT a CA_BUNDLE v skripte kverkom_test.py:
 - CLIENT_CERT = ("./cert/kverkom-int-client.pem", "./cert/kverkom-int-client.key")
 - CA_BUNDLE = "./cert/kverkom-ca-bundle.pem"
- Overiť spúšťanie skriptu:

```
kverkom_test.py

usage: kverkom_test.py [-h] {erpStatus,bankStatus,newTransaction,getAllTransactions,wait,notify,erp} ...
kverkom_test.py: error: the following arguments are required: command
```

Popis skriptu

Syntax, argumenty a prepínače:

kverkom_test.py - Simulátor ERP a Banky pre projekt KVERKOM

Použitie:

```
kverkom_test.py erpStatus [-d]
kverkom_test.py bankStatus [-d]
kverkom_test.py newTransaction [-d]
kverkom_test.py getAllTransactions POKLADNICA [-d]
kverkom_test.py wait VATSK [POKLADNICA [TRANSACTIONID]] [-d]
kverkom_test.py notify IBAN AMOUNT END_TO_END_ID [-h HOST] [-d]
kverkom_test.py erp VATSK POKLADNICA [-d]
```

Poznámky:

- `kverkom_test.py -h` alebo `kverkom_test.py --help` vypíše krátky popis použitia
- Rovnako je kontextový popis dostupný aj pre jednotlivé príkazy, napr. `kverkom_test.py wait -h`
- Prepínač "-d" spôsobí podrobnejšie výpisy (pre účely diagnostiky)
- Niektoré nastavenia sú priamo súčasťou skriptu (zatiaľ), a v prípade potreby je ich možné zmeniť priamo tam, konkrétne:

```
# defaults (pre integrané prostredie KVERKOM vo FRSR)
ERP_HOST_DEFAULT = "api-erp-i.kverkom.sk"
BANK_HOST_DEFAULT = "api-banka-i.kverkom.sk"
MQTT_HOST_DEFAULT = "mqtt-i.kverkom.sk"
MQTT_PORT_DEFAULT = 8883
TIMEOUT_S = 30

# cesty k lokálnymi súborom s certifikátmi a kúmi
CLIENT_CERT = ("./cert/kverkom-int-client.pem", "./cert/kverkom-int-client.key")
CA_BUNDLE = "./cert/kverkom-ca-bundle.pem"
```

Overenie dostupnosti serverov a služieb

Príkazy

```
kverkom_test.py erpStatus
kverkom_test.py bankStatus
```

typicky vrátia stav a základné údaje rozhraní:

```
{ "timestamp": "2025-08-20T12:49:08.808Z", "instance": "kverkom-erp-api", "build": "2025-08-19", "version": "0.2.1" }
{ "timestamp": "2025-08-20T12:49:15.224Z", "instance": "kverkom-bank-api", "build": "2025-08-19", "version": "0.2.1" }
```

Scenár - okamžitá platba na pokladni

Nasledovný príkaz simuluje interakcie pokladnice (ERP) a Banky (ako klienti) s NOP (ako server) v súvislosti s okamžitou platbou:

```
kverkom_test.py erp VATSK-1234567890 POKLADNICA-88812345678900002 -d
```

Tento príkaz

- vyžiada nový identifikátor transakcie (transactionId, resp. endToEndId) - a vypíše ho,
- subscribne sa na špecifický topic v MQTT brokeri (podľa transactionId),
- a čaká na konkrétnu notifikáciu (s daným transactionId).

Typický výstup je:

```
[newTransaction] 200 {"transaction_id": "QR-e22732a02dca40ed8d850b0e38ed130e", "created_at": "2025-08-17T12:58:40.792Z"}
[MQTT.CONNECT]
[MQTT.SUBSCRIBE] topic=VATSK-1234567890/POKLADNICA-88812345678900001/QR-e22732a02dca40ed8d850b0e38ed130e mid=1
granted_qos=[ReasonCode(Suback, 'Granted QoS 1')]
```

V ďalšom (inom) okne (termináli) nasimulujeme registráciu (poslanie) notifikácie "z banky" (pričom použijeme transaction_id z výstupu vyššie):

```
kverkom_test.py notify SK4811000000002944116480 123.45 QR-e22732a02dca40ed8d850b0e38ed130e
```

V pôvodnom okne - z bežiacim "erp" príkazom - sa príchod notifikácie prejaví nasledovným výpisom a ukončením programu:

```
[MQTT.SUBSCRIBE] topic=VATSK-1234567890/POKLADNICA-88812345678900001/QR-e22732a02dca40ed8d850b0e38ed130e mid=1
granted_qos=[ReasonCode(Suback, 'Granted QoS 1')]
[MQTT.MESSAGE] VATSK-1234567890/POKLADNICA-88812345678900001/QR-e22732a02dca40ed8d850b0e38ed130e
{"transactionStatus": "ACCC", "transactionAmount": {"currency": "EUR", "amount": "123.45"}, "endToEndId": "QR-
e22732a02dca40ed8d850b0e38ed130e", "dataIntegrityHash":
"84a88b0b5e1d86c9b942099d8cc467442801147b407dd3d4b0f1783d5653bc65", "creditorAccount": {"iban":
"SK4811000000002944116480"}, "receivedAt": "2025-08-17T12:58:54.840689Z"}
[MQTT.STOP] Prijatá oakávaná správa - koniec
[MQTT.DISCONNECT]
```

Týmto je tento základný (a najdôležitejší) scenár ukončený.

Referencie a prílohy

- Všeobecné informácie o QR platbách, vrátane aktuálnych technických informácií: <https://www.info-qrplatby.sk/>
- Štandard platobnej linky: <https://www.sbaonline.sk/projekt/standard-platobnej-linky/>
- Štandard služby notifikácie o pripísaných prostriedkoch na účet: <https://www.sbaonline.sk/projekt/standard-sluzby-notifikacie/>
- Popis NOP API: [NOP Lite](#) - [NOP Services API](#)
- SWAGGER (OpenAPI 3) definície:
 - [erp_openapi-0.5.8.yaml](#) (príloha)
 - bank-api - vid' openapi specification (swagger) v <https://www.sbaonline.sk/projekt/standard-sluzby-notifikacie/>
- Jednoduchý klient (python) pre ad-hoc testovanie vybraných rozhraní NOP: [py-kverkom-1.0.2.zip](#)