MERCHANT INTEGRATION MANUAL

Integračný manuál obchodníka

Názov	Integračný manuál obchodníka
Verzia	4.75

Obsah

1 Úvod

1.1 Termíny a ustálené výrazy

PSP	payment service provider - poskytovateľ platobnej služby
24pay	automatizovaný systém platobnej inštitúcie - poskytovateľ platobnej služby
Obchodník Obchod Merchant	online obchod poskytujúci tovar/služby, prijímajúci platby
Klient Zákazník Client	osoba nakupujúca tovar/služby, vykonávajúca platby
RURL	Redirection Return URL – návratová URL adresa obchodu, kam je zákazník presmerovaný po platbe
NURL	Notification Return URL – URL adresa, kde sú zasielané notifikácie o zmene stavu platby prostredníctvom protokolu HTTP/HTTPS metódou POST v rámci tela requestu

1.2 Obsah dokumentu

Účelom tohto dokumentu je popísať komunikačný protokol medzi webovým serverom obchodníka a platobným rozhraním systému 24pay. Slúži ako technická príručka pre služby poskytované systémom 24pay a obsahuje popis krokov ako sa korektne pripojiť a komunikovať s jeho platobným rozhraním.

Dokument nie je návodom na vytváranie web stránok. Jeho úlohou je vymenovať a popísať podmienky, ktoré musí web obchodníka spĺňať za účelom úspešnej realizácie platobnej služby.

2 Integrácia s 24pay

2.1 Konfiguračné údaje

Nasledujúca sekcia popisuje množinu údajov, ktoré si navzájom medzi sebou vymenia obchodník a 24pay.

Obchodník uvádza nasledujúce údaje:

- RURL
- NURL

24pay poskytuje obchodu nasledovné údaje:

- Mid
- EshopId
- Key

2.2 Procesný model

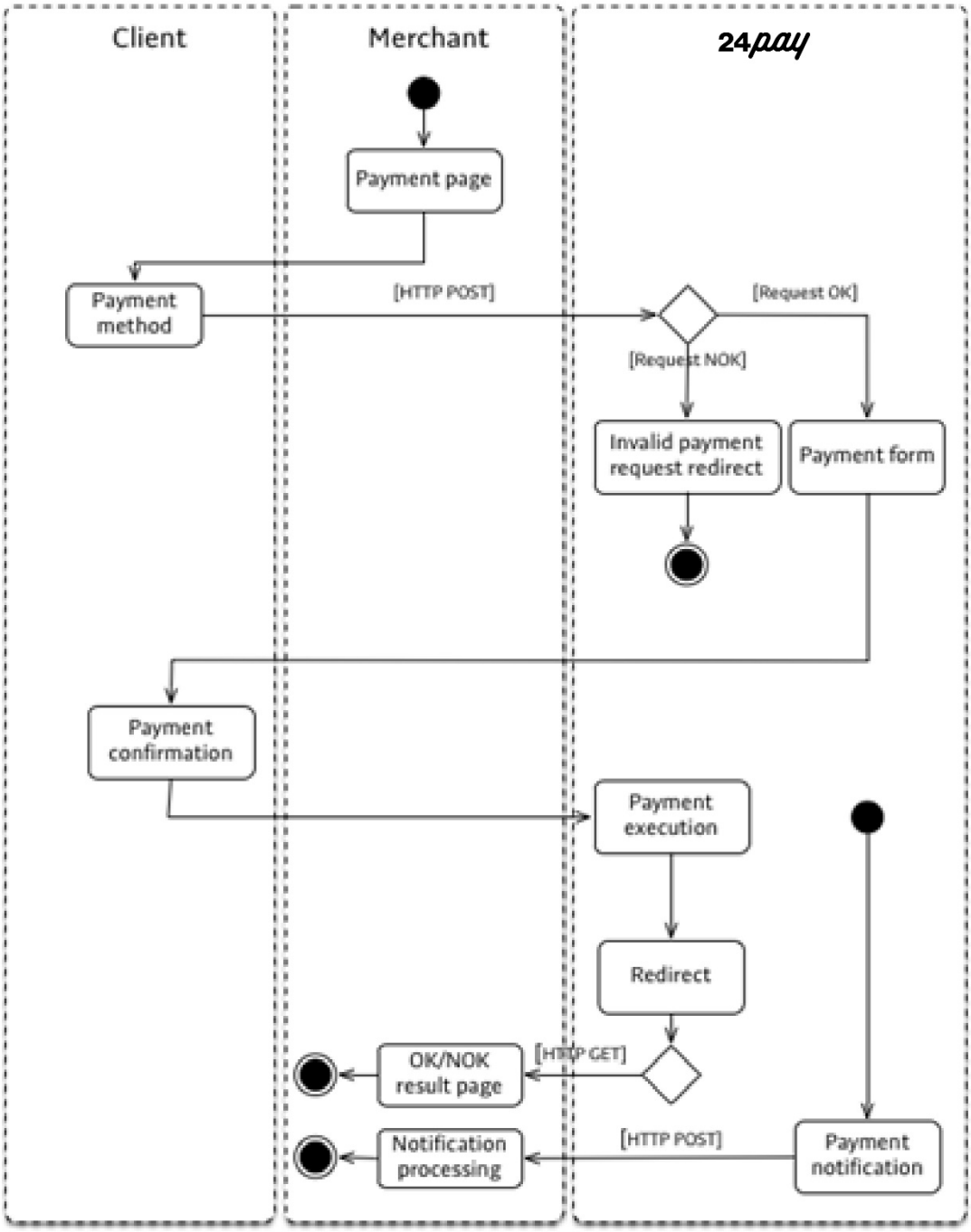
Účelom tejto sekcie je načrtnúť procesný model spracovania a realizácie platobnej relácie zobrazením interakcie medzi aktérmi: klient – obchodník – 24pay.

Platobná stránka obchodníka obsahuje odkaz na 24pay. Zákazník, ktorý si vybral 24pay ako želanú platobnú metódu, zašle zo systému obchodníka do 24pay žiadosť o realizáciu platby. Žiadosť obsahuje predpísanú množinu údajov potrebných pre spracovanie a realizáciu platobnej relácie.

Zákazník je presmerovaný na platobný portál bankovej inštitúcie. Následne potvrdí, alebo zruší platbu. 24pay realizuje potrebné kroky spracovania platobnej relácie, pošle notifikačnú správu o stave transakcie a presmeruje zákazníka späť na stránku obchodníka. V prípade nevyhovujúceho formátu/obsahu parametrov prijatej žiadosti je zákazník presmerovaný na stránku systému 24pay informujúcu o neúspešnej požiadavke na realizáciu platobnej relácie.

24pay zasiela notifikáciu o výsledku realizácie platobnej relácie obchodníkovi na adresu špecifikovanú konfiguračnou položkou NURL. Server side obchodníka má povinnosť reagovať odpoveďou HTTP status 200 OK, potvrdzujúcou prijatie odpovede.

24pay presmeruje zákazníka metódou GET na stránku obchodníka špecifikovanú konfiguračnou položkou RURL. Návratové adresy obsahujú reťazec parametrov informujúcich o výsledku spracovania platobnej relácie, na základe ktorých systém obchodníka oboznámi zákazníka o úspešnom, či neúspešnom spracovaní. Návratové adresy slúžia iba pre informatívne účely, na ich základe nie je možné vykonávať žiadne rozhodnutia.



Obrázok 1. Procesný model

2.3 Grafické prvky

Pre zobrazenie platobného tlačidla a loga na stránke použite logo 24pay, ktoré je uvedené na: https://24-pay.sk/o-spolocnosti/dokumenty-na-stiahnutie/

3 Protokol platieb

3.1 Požiadavka na realizáciu platby od obchodníka

Pre zaslanie novej platobnej žiadosti je nutné na stránku webového sídla obchodu umiestniť príslušný formulár, ktorý presmeruje zákazníka na 24pay platobnú bránu.

URL 24pay platobná brána:

https://admin.24-pay.eu/pay_gate/paygt

Podmienkou je vytvorenie HTTPS požiadavky metódou POST. Údaje kódované vo forme application/x-www-form-urlencoded. Zoznam parametrov obsahuje nasledujúca tabuľka:

Parameter	Povinný	Formát	Dĺžka	Popis	Príklad
Mid	•	Alpha-numeric	8	Identifikátor obchodníka (case sensitive)	1a2B3c4D
EshopId	•	Numeric	1..10	Identifikátor e-shopu	135
MsTxnId	•	Alpha-numeric	1..32	Jednoznačný/jedinečný identifikátor platby poskytnutý obchodníkom (variabilný symbol)	1234567890
Amount	•	#0.00	1..10,2	Suma platby. Oddeľovačom desatinnej časti je bodka. Desatinná časť je vždy zastúpená dvoma číslicami.	1.00
CurrAlphaCode	•	AAA	3	Mena platby ISO 4217	EUR
ClientId	•	Alpha-numeric	3..10	Identifikátor zákazníka v systéme obchodníka (case sensitive)	12345
FirstName	•	Alphabetic	2..50	Zákazník – krstné meno	Jožko
FamilyName	•	Alphabetic	2..50	Zákazník – priezvisko	Mrkvička
Email	•	email	6..128	Zákazník – emailová adresa	jozko.mrkvicka@demo.com
Country	•	AAA	3	Zákazník – kód krajiny bydliska ISO 3166-1	SVK
Timestamp	•	yyyy-MM-dd HH:mm:ss	19	Časová pečiatka tvorby platobnej požiadavky. Oddeľovačom dátumovej a časovej položky je znak medzera. Timestamp a MsTxnId musia tvoriť unikátnu kombináciu.	2014-12-01 13:00:00
Sign	•	Alpha-numeric	32	Kontrolný súčet zasielaných parametrov	
LangCode		aa	2	Kód jayka ISO 639-1. sk, cs, en, de, hu, es, fr, it, pl. Štandardne sk.	sk

RURL		URL	256	URL adresa, kam je zákazník presmerovaný po zrealizovaní transakcie. V prípade prítomnosti prekryje konfigurovanú položku RURL.	http://mojobchod.sk/rurl
NURL		URL	256	URL adresa obchodu, kam sú zasielané notifikácie o zmene stavu platby prostredníctvom HTTP/HTTPS POST. V prípade prítomnosti prekryje konfigurovanú položku NURL.	http://mojobchod.sk/nurl
NotifyEmail		email	6..128	Emailová adresa, kam sú zasielané dodatočné notifikácie o zmene stavu platby.	platby@mojobchod.sk
RedirectSign		true/false	4/5	Možnosť pridania podpisu pri presmerovaní.	false
PreAuthProvided		true/false	4/5	Možnosť predautorizácie platby (iba pre platobné karty)	false
Phone		Alha-numeric	8..25	Zákazník – telefónny kontakt	0901 000 001
Street		Alpha-numeric	3..50	Zákazník – ulica	Kvetná 123
City		Alphabetic	2..50	Zákazník – mesto bydliska	Bratislava
Zip		Alpha-numeric	1..10	Zákazník – poštové smerovacie číslo bydliska	821 08

3.2 Notifikácia o stave spracovania platby od 24pay

Po ukončení spracovania žiadosti na realizáciu platby zo strany obchodu 24pay notifikuje o stave spracovania platby. Správa je odoslaná v rámci HTTP POST požiadavky adresovanej na **NURL**.

Údaje týkajúce sa danej platby sú prenášané vo forme štruktúry majúcej XML formát ako hodnota parametra **params.**

Príklad notifikácie:

```
<?xml version="1.0" encoding="UTF-8"?>
<Response sign="21f22ef2af21d3819cd0cff06ef55943">
    <Transaction>
        <Identification>
            <MsTxnId>1234567890</MsTxnId>
            <PspTxnId>0987654321</PspTxnId>
        </Identification>
        <Presentation>
            <Amount>1.00</Amount>
            <Currency>EUR</Currency>
        </Presentation>
        <Customer>
            <Contact>
                <Email> jozko.mrkvicka@demo.com</Email>
                <Phone>0901 000 001</Phone>
            </Contact>
            <Address>
                <Street>Kvetná 123</Street>
                <Zip>821 08</Zip>
                <City>Bratislava</City>
                <Country>SVK</Country>
            </Address>
            <Name>
                <Given>Jožko</Given>
                <Family>Mrkvička</Family>
            </Name>
        </Customer>
        <Processing>
            <Timestamp>2014-12-01 13:01:00.548</Timestamp>
            <Result>OK</Result>
            <Reason code="00">Successful Processing</Reason>
             <PSPCategory>2</PSPCategory>
             <CreditCard/>
        </Processing>
    </Transaction>
</Response>
```

<Result> označuje stav platby. Môže nadobúdať nasledujúce hodnoty:

- **OK** – platba úspešná
- **FAIL** – platba neúspešná
- **PENDING** – platba bola odoslaná na spracovanie. Po spracovaní platby je odoslaná nová notifikácia, kde bude <Result> buď OK alebo FAIL.
- **AUTHORIZED** – žiadosť o predautorizáciu bola úspešná. Dokončenie alebo zrušenie platby je možné vykonať do 7 dní.

<PSPCategory> označuje kategóriu platobnej metódy, ktorú klient využil na platbu.

1 – platby kartou

2 – okamžité platby

3 – bankové prevody

4 – ostatné

3.3 Presmerovanie zákazníka do systému obchodníka

Po dokončení platby je klient presmerovaný späť do systému obchodníka na **RURL**, ktoré uvádza obchod. Presmerovanie je vykonané HTTP GET požiadavkou, pričom reťazec dopytu obsahuje parametre nesúce informáciu o úspešnom, či neúspešnom výsledku spracovania platby.

Je nutné si uvedomiť, že **RURL** slúži iba pre informatívne účely. Na základe údajov prijatých v rámci presmerovania späť na systém obchodníka nie je možné vykonávať žiadne rozhodnutia.

Parameter	Formát	Dĺžka	Popis	Príklad
MsTxnId	Numeric	1..256	Jednoznačný/jedinečný identifikátor platby poskytnutý obchodníkom (variabilný symbol)	1234567890
Amount	#0.00	1..10, 2	Suma platby. Oddeľovačom desatinnej časti je bodka. Desatinná časť je vždy zastúpená dvoma číslicami.	1.00
CurrCode	AAA	3	Mena platby ISO 4217	EUR
Result	OK/ FAIL/ PENDING/ AUTHORIZED	2/4/7	OK - platba úspešná. FAIL – platba neúspešná. PENDING – platba odoslaná na spracovanie AUTHORIZED – žiadosť o predautorizáciu úspešná	OK
Sign	Alpha-numeric	32	Kontrolný súčet zasielaných parametrov. Posielaný iba v prípade, že pri požiadavke bol zaslaný parameter ‘RedirectSign=true‘.	21f22ef2af21d3819cd0cff06ef55943

Príklad presmerovania:

http://mojobchod.sk/rurl?MsTxnId=1234567890&Amount=1.00&CurrCode=EUR&Result=OK

3.4 Dokončenie/zrušenie predautorizovanej platby

Dokončenie alebo zrušenie predautorizácie je možné volať iba pri platbách, ktoré sú založené ako predautorizované a sú v stave AUTHORIZED.

Podmienkou je vytvorenie HTTPS požiadavky metódou POST. Údaje kódované vo forme application/x-www-form-urlencoded.

https://admin.24-pay.eu/pay_gate/auth

Zoznam parametrov obsahuje nasledujúca tabuľka:

Parameter	Povinný	Formát	Dĺžka	Popis	Príklad
Mid	•	Alpha-numeric	8	Identifikátor obchodníka (case sensitive)	1a2B3c4D
EshopId	•	Numeric	1..10	Identifikátor e-shopu	135
MsTxnId	•	Alpha-numeric	1..32	Jednoznačný/jedinečný identifikátor platby poskytnutý obchodníkom (variabilný symbol)	1234567890
PspTxnId	•	Alpha-numeric	1..32	Jednoznačný/jedinečný identifikátor platby generovaný °, posielaný v notifikačnej správe po predautorizácii	0987654321
Amount	•	#0.00	1..10, 2	Suma platby. Oddeľovačom desatinnej časti je bodka. Desatinná časť je vždy zastúpená dvoma číslicami. Pri dokončení predautorizácie musí byť hodnota rovnaká alebo nižšia ako suma predautorizácie. V prípade zrušenia musí byť hodnota rovnaká ako suma predautorizácie.	1.00
CurrAlphaCode	•	AAA	3	Mena platby ISO 4217	EUR
Timestamp	•	yyyy-MM-dd HH:mm:ss	19	Časová pečiatka tvorby platobnej požiadavky. Oddeľovačom dátumovej a časovej položky je znak medzera.	2014-12-01 13:00:00
Target	•	OK/FAIL	2/4	OK – dokončenie platby FAIL – zrušenie platby	OK
Sign	•	Alpha-numeric	32	Kontrolný súčet zasielaných parametrov	
NURL		URL	256	URL adresa obchodu, kam sú zasielané notifikácie o zmene stavu platby prostredníctvom HTTP/HTTPS POST. V prípade prítomnosti prekryje konfigurovanú položku NURL.	http://mojobchod.sk/nurl

V odpovedi obdrží obchodník nasledujúce informácie vo formáte json:

```
{ “MsTxnId“:“1234567890“,
  “PspTxnId“:“ 0987654321“
  “Amount“:“1.00“,
 “CurrCode“:“EUR“,
 “Target”:”OK”,
  “Status“:“OK“}
```

Táto odpoveď potvrdzuje prijatie na spracovanie. Potvrdenie o zmene stavu transakcie je zasielané notifikačnou správou na NURL (sekcia 4.2) to však iba v prípade, že je po zaslaní požiadavky Status OK alebo FAIL, v prípade ERROR notifikačná správa nie je zasielaná, nakoľko nedôjde k zmene stavu platby.

3.5 SIGN

Pre každú požiadavku na realizáciu platby zo strany obchodu a notifikáciu o statuse spracovania platby zo strany 24pay, je vytvorený kontrolný súčet. Prostredníctvom kontrolného súčtu možno overiť integritu a autenticitu údajov.

Správnosť vytváraného podpisu je možné dodatočne overiť v rozhraní 24pay https://admin.24-pay.eu/sup_gui/pages/PayReqSimulation.jsf.

3.5.1 Bezpečnostný lúč

Pre každého obchodníka je vygenerovaný bezpečnostný kľúč key. Obchodník získa key v hexadecimálnom zápise - reťazec 64 znakov.

Okrem bezpečnostného kľúča, je pre výpočet kontrolného súčtu potrebný aj inicializačný vektor IV. Inicializačný vektor je vytvorený zreťazením parametra Mid so svojou reverznou podobou. Týmto spôsobom získaná sekvencia 16 znakov reprezentuje inicializačný vektor IV.

3.5.2 Kontrolný účet

Pri komunikácii je vytvorený kontrolný súčet, resp. bezpečnostný podpis nasledovným spôsobom:

a) Zreťazením podpisom chránených parametrov v predpísanom poradí sa vytvorí MESSAGE, ktorého obsah bude predmetom šifrovania.
b) Vytvorený reťazec je transformovaný na HASH/MD (message digest) pevnej dĺžky (20 B = 160 bits) pomocou hashovacej funkcie SHA1.
c) Takto získaný "odtlačok" MD je následne šifrovaný symetrickým algoritmom AES[1] použitím:
   a. inicializačného vektora **IV**
   b. a definovaného bezpečnostného kľúča **key**
d) Výstupom je bezpečnostný podpis dĺžky 32 B = 256 bits. Prvých 16 B podpisu je konvertovaných na reťazec zodpovedajúci hexadecimálnemu zápisu tejto časti podpisu. Pôvodný otvorený text MD je týmto spôsobom transformovaný na šifrovaný text reprezentujúci bezpečnostný podpis o dĺžke 32 znakov.

3.5.3 Požiadavka na realizáciu platby od obchodníka

Obchodník zasiela bezpečnostný podpis v rámci komunikácie ako hodnotu parametra **SIGN**.

Predmetom zreťazenia sú nasledujúce parametre:

MESSAGE =
Mid ⊕ Amount ⊕ CurrencyAlphaCode ⊕ MsTxnId ⊕ FirstName ⊕ FamilyName ⊕ Timestamp

[1] Blokový symetrický kryptografický algoritmus; key-size 256bits; block-size 128 bits; mód AES/CBC/PKCS7Padding.

3.5.4 Notifikácia o stave spracovania platby od 24pay

Obchodník z parametrov notifikácie vytvorí rovnakým spôsobom kontrolný bezpečnostný podpis a porovná ho s hodnotou prijatého parametra **SIGN**.

Predmetom zreťazenia sú nasledujúce parametre:

MESSAGE =

Mid ⊕ Amount ⊕ Currency ⊕ PspTxnId ⊕ MsTxnId ⊕ Timestamp ⊕ Result

3.5.5 Presmerovanie zákazníka do systému obchodníka

Posielaný iba v prípade, že pri požiadavke bol zaslaný parameter ‘RedirectSign=true‘

Obchodník zo zaslaných parametrov pri presmerovaní vytvorí rovnakým spôsobom kontrolný bezpečnostný podpis a porovná ho s hodnotou prijatého parametra **SIGN**.

Predmetom zreťazenia sú nasledujúce parametre:

MESSAGE =

MsTxnId ⊕ Amount ⊕ CurrCode ⊕ Result

3.5.6 Dokončenie/zrušenie predautorizovanej platby

Obchodník zasiela bezpečnostný podpis v rámci komunikácie ako hodnotu parametra **SIGN**.

Predmetom zreťazenia sú nasledujúce parametre:

MESSAGE =

Mid ⊕ Amount ⊕ CurrencyAlphaCode ⊕ MsTxnId ⊕ PspTxnId⊕Target ⊕ Timestamp

4 Prílohy

4.1 Predloha tvorby podpisu

4.1.1 Prípad požiadavky na realizáciu platby

Key	1234567812345678123456781234567812345678123456781234567812345678
IV	{0x58, 0x32, 0x34, 0x35, 0x6e, 0x53, 0x4f, 0x33, 0x33, 0x4f, 0x53, 0x6e, 0x35, 0x34, 0x32, 0x58}
Mid	DemoOMED
Amount	1.00
CurrencyAlphaCode	EUR
MsTxnId	1234567890
FirstName	Jožko
FamilyName	Mrkvička
Timestamp	2014-12-01 13:00:00
Sign	2b817107edb88129d9aa8316f8758270

hexKey = 1234567812345678123456781234567812345678123456781234567812345678

length 64 characters

byte[] keyBytes = {0x12, 0x34, 0x56, 0x78, 0x12, 0x34, 0x56, 0x78, ... , 0x34, 0x56, 0x78}

length 32B = 256 bits

txtIV = DemoOMEDDEMOomeD

length 16 characters

byte[] IV= {0X44, 0X65, 0X6D, 0X6F, 0X4F, 0X4D, 0X45, 0X44, 0X44, 0X45, 0X4D, 0X4F, 0X6F, 0X6D. 0X65. 0X44}

length 16B = 128 bits

MESSAGE = DemoOMED1.00EUR1234567890JožkoMrkvička2014-12-01 13:00:00

byte[] hash/md = SHA-1(message) = {0X78, 0XF7, 0XDA, 0X5C, 0X9D, 0X06, 0XEB, 0X02, 0X5A, 0X55, 0X7D, 0XBA, 0XB9, 0X41, 0X31, 0X83, 0X32, 0XA7, 0X2F, 0XB1}

length 20B = 160bits

byte[] signBytes = {0X2B, 0X81, 0X71, 0X07, 0XED, 0XB8, 0X81, 0X29, 0XD9, 0XAA, 0X83, 0X16, 0XF8, 0X75, 0X82, 0X70, 0X31, 0X71, 0X5D, 0XAF, 0X1F, 0X70, 0XB6, 0X7A, 0X6F, 0X92, 0X0A, 0XF7, 0XB7, 0X19, 0X13, 0X72}

length 32B = 256 bits

sign = **2b817107edb88129d9aa8316f8758270**

4.1.2 Prípad notifikácie o statuse spracovania platby

Key	1234567812345678123456781234567812345678123456781234567812345678
IV	{0x58, 0x32, 0x34, 0x35, 0x6e, 0x53, 0x4f, 0x33, 0x33, 0x4f, 0x53, 0x6e, 0x35, 0x34, 0x32, 0x58}
Mid	DemoOMED
Amount	1.00
Currency	EUR
PspTxnId	0987654321
MsTxnId	1234567890
Timestamp	2014-12-01 13:01:00
Result	OK
Sign	21f22ef2af21d3819cd0cff06ef55943

hexKey = 1234567812345678123456781234567812345678123456781234567812345678

length 64 characters

byte[] keyBytes = {0x12, 0x34, 0x56, 0x78, 0x12, 0x34, 0x56, 0x78, ... , 0x34, 0x56, 0x78}

length 32B = 256 bits

txtIV = DemoOMEDDEMOomeD

length 16 characters

byte[] IV= {0X44, 0X65, 0X6D, 0X6F, 0X4F, 0X4D, 0X45, 0X44, 0X44, 0X45, 0X4D, 0X4F, 0X6F, 0X6D, 0X65, 0X44}

length 16B = 128 bits

message = DemoOMED1.00EUR0987654321123456789020l4-12-01 13:00:00OK

byte[] hash/md = SHA-1(message) = {0XC4, 0X77, 0X06, 0X33, 0X7F, 0X91, 0XAB, 0X96, 0XEE, 0X20, 0X6A, 0XEA, 0X35, 0XFD, 0X2A, 0X8E, 0X74, 0X57, 0XED, 0XBF}

length 20B = 160bits

byte[] signBytes = {0X21, 0XF2, 0X2E, 0XF2, 0XAF, 0X21, 0XD3, 0X81, 0X9C, 0XD0, 0XCF, 0XF0, 0X6E, 0XF5, 0X59, 0X43, 0X57, 0X67, 0X14, 0XC1, 0XB0, 0XD1, 0X95, 0X67, 0X99, 0X12, 0XF9, 0XDE, 0X38, 0X72, 0X38, 0XCE}

length 32B = 256 bits

sign = **21f22ef2af21d3819cd0cff06ef55943**

4.1.3 Prípad dokončenia predautorizovanej platby

Key	1234567812345678123456781234567812345678123456781234567812345678
IV	{0x58, 0x32, 0x34, 0x35, 0x6e, 0x53, 0x4f, 0x33, 0x33, 0x4f, 0x53, 0x6e, 0x35, 0x34, 0x32, 0x58}
Mid	DemoOMED
Amount	1.00
CurrencyAlphaCode	EUR
MsTxnId	1234567890
PspTxnId	0987654321
Target	OK
Timestamp	2014-12-01 13:00:00
Sign	34087afa7367d29507f2d3561bd63171

	hexKey = 1234567812345678123456781234567812345678123456781234567812345678
	length 64 characters
	byte[] keyBytes = {0x12, 0x34, 0x56, 0x78, 0x12, 0x34, 0x56, 0x78, ... , 0x34, 0x56, 0x78}
	length 32B = 256 bits
	txtIV = DemoOMEDDEMOomeD
	length 16 characters
	byte[] IV= {0X44, 0X65, 0X6D, 0X6F, 0X4F, 0X4D, 0X45, 0X44, 0X44, 0X45, 0X4D, 0X4F, 0X6F, 0X6D, 0X65, 0X44}
	length 16B = 128 bits
	MESSAGE = DemoOMED1.00EUR123456789009876543210K2014-12-01 13:00:00
	byte[] hash/md = SHA-1(message) = {0XDF, 0XBE, 0X53, 0X2A, 0X00, 0XA8, 0XA9, 0X44, 0XAF, 0X9F, 0XA4, 0X49, 0XE1, 0X7D, 0X25, 0X4B, 0X39, 0X9D, 0X05, 0X7C}
	length 20B = 160bits
	byte[] signBytes = {0X34, 0X08, 0X7A, 0XFA, 0X73, 0X67, 0XD2, 0X95, 0X07, 0XF2, 0XD3, 0X56, 0X1B, 0XD6, 0X31, 0X71, 0X19, 0X20, 0X8A, 0X93, 0XB7, 0XE0, 0X09, 0X89, 0X5D, 0X87, 0XE8, 0XCB, 0XDE, 0X28, 0XE6, 0X86}
	length 32B = 256 bits
	sign = **34087afa7367d29507f2d3561bd63171**

4.1.4 Príklady zdrojového kódu

a) *PHP*

```
public function computeSIGN($mid, $key, $message){
        $hash = hash("sha1", $message, true);
        $iv = $mid . strrev($mid);
        $key = pack('H*', $key);
        $crypted = openssl_encrypt( $hash, 'AES-256-CBC', $key, 1, $iv );
        $sign = strtoupper(bin2hex(substr($crypted, 0, 16)));
        return $sign;
}
```

b) *Java*

```
public String generateSign(String message, String key, String iv) {
        try {
                Security.addProvider(new BouncyCastleProvider());
                byte[] keyBytes = Hex.decodeHex(key.toCharArray());
                byte[] ivBytes = iv.getBytes();

                SecretKeySpec secretKeySpec = new SecretKeySpec(keyBytes, "AES");
                IvParameterSpec ivSpec = new IvParameterSpec(ivBytes);
                Cipher encryptCipher = Cipher.getInstance("AES/CBC/PKCS7Padding");
                encryptCipher.init(Cipher.ENCRYPT_MODE, secretKeySpec, ivSpec);

                byte[] sha1Hash = DigestUtils.sha1(message);
                byte[] encryptedData = encryptCipher.doFinal(sha1Hash);
                return Hex.encodeHexString(encryptedData).substring(0,32);
        } catch (Exception e) {
                logger.error("ERROR!", e);
                return null;
        }
}
```

c) *.NET framework 3.5 (C#)*

```
public static string AesEncrypt( string message, byte[] Key, byte[] IV, PaddingMode
paddingMode , CipherMode  cipherMode)
{
	byte[] hash = GetSha1(message);
	AesManaged aes= new AesManaged();
	aes.Key = Key;
	aes.IV = IV;
	aes.Mode = cipherMode;
	aes.Padding = paddingMode;
	ICryptoTransform encryptor = aes.CreateEncryptor(aes.Key, aes.IV);

	byte[] encrypted = null;

	using (MemoryStream ms = new MemoryStream()) {
		using (var cs = new CryptoStream(ms, encryptor, CryptoStreamMode.Write))
{
			cs.Write(hash, 0, hash.Length);
		}
		encrypted = ms.ToArray();
	}

	return ConvertByteArrayToHexString(encrypted);
}
```

d) *.NET framework 3.5 (VB)*

```
Public Shared Function AesEncrypt(message As String, Key As Byte(), IV As Byte(),
paddingMode As PaddingMode, cipherMode As CipherMode) As String
	Dim hash As Byte() = GetSha1(message)

	Dim aes As New AesManaged()
	aes.Key = Key
	aes.IV = IV
	aes.Mode = cipherMode
	aes.Padding = paddingMode

	Dim encryptor As ICryptoTransform = aes.CreateEncryptor(aes.Key, aes.IV)
	Dim encrypted As Byte() = Nothing

	Using ms As New MemoryStream()
		Using cs = New CryptoStream(ms, encryptor, CryptoStreamMode.Write)
			cs.Write(hash, 0, hash.Length)
		End Using
		encrypted = ms.ToArray()
	End Using

	Return ConvertByteArrayToHexString(encrypted)
End Function
```

4.2 Platobný formulár