Platobná brána

Technická dokumentácia

verzia 1.0

Payment institution

Obsah

1. Úvod

Tento dokument popisuje implementáciu služby platobnej brány. Účelom dokumentu je poskytnúť návod ako vytvoriť správne fungujúce a bezpečné prepojenie medzi Internet bankingovým serverom platobnej inštitúcie Payment a serverom obchodníka a popísať priebeh komunikácie medzi nimi. Dokument je určený osobám s technickými znalosťami.

2. Slovník pojmov

- *platobná brána* – služba, ktorá umožňuje klientom, ktorí majú oprávnenie pre disponovanie s bežným účtom, realizovať online platby na účet obchodníka.
- *HMAC* – hašovaný autentifikačný kód, ktorý je vypočítaný z reťazca (zostaveného podľa špecifikácie) a bezpečnostného kľúča, ktorý obdrži obchodník od platobnej inštitúcie. Slúži pre overenie integrity správ zasielaných medzi serverom platobnej inštitúcie a serverom obchodníka.
- *ECDSA* – digitálny podpis vytvorený algoritmom ECDSA, ktorý je posielaný v odpovediach zo servera platobnej inštitúcie a slúži pre overenie autenticity. Obchodník podpis skontroluje pomocou verejného kľúča.
- *bezpečnostný kľúč* – 128 znakový kľúč, ktorý je obchodníkovi odovzdaný pri podpise zmluvy.
- *verejný kľúč* – kľúč slúžiaci na kontrolu ECDSA podpisu.

3. Služba platobnej brány

Služba platobnej brány umožňuje klientom, ktorí majú oprávnenie pre disponovanie s bežným účtom, realizovať online platby na účet obchodníka prostredníctvom špeciálneho URL odkazu, ktorý môže byť umiestnený na webovej stránke (internetovom obchode).

3.1 Realizácia platby

Primárnym využitím služby je platba za tovar alebo služby na internetových obchodoch.

3.1.1 Postup

1. zákazník po nákupe tovaru alebo služieb v internetovom obchode klikne na symbol platby prostredníctvom platobnej brány.
2. server obchodníka presmeruje zákazníka prostredníctvom URL odkazu na Internet bankingový server platobnej inštitúcie.
3. platobná inštitúcia overí platnosť a správnosť parametrov zaslaných prostredníctvom URL a zobrazí aplikáciu platobnej brány.
4. zákazník sa prihlási identifikačnými údajmi (email, heslo a kód z autorizačného zariadenia).
5. aplikácia zobrazí predvyplnený formulár platby z bežného účtu zákazníka na účet obchodníka. Klient môže zmeniť účet platiteľa v prípade, že má oprávnenie pre vykonanie platby na viacerých bežných účtoch.
6. zákazník potvrdí alebo zruší platbu.
7. aplikácia zobrazí zákazníkovi informáciu o výsledku spracovania platby.
8. zákazník stlačí tlačidlo **`Pokračovať`** pre návrat na stránku obchodníka.
9. obchodník overí výsledok spracovania platby. K dispozícii má nasledovné možnosti:
    - kontrola parametrov v návratovej URL
    - kontrola notifikačného emailu (pokiaľ bol korektne vyplnený parameter **`remail`** v požiadavke)
    - kontrola pohybov na účte prostredníctvom Internet bankingu alebo mobilnej aplikácie

3.2 Bezpečnosť

Komunikácia medzi obchodníkom a platobnou inštitúciou je:

- prenášaná a šifrovaná protokolom SSL
- server obchodníka aj platobnej inštitúcie zabezpečí integritu zasielaných údajov prostredníctvom hašovaného autentifikačného kódu (HMAC), ktorý je vypočítaný z parametrov a bezpečnostného kľúča, ktorý obdrži obchodník od platobnej inštitúcie
- server platobnej inštitúcie navyše odpovede podpíše digitálnym podpisom (ECDSA), ktorý si obchodník skontroluje na základe verejného kľúča

Obchodník je povinný overiť si pravosť odpovede z banky overením správnosti autentifikačného kódu HMAC a digitálneho podpisu ECDSA.
Ak sa HMAC a ECDSA nezhodujú s vypočítanými hodnotami na strane obchodníka, odpoveď je vyhodnotená ako podozrivá a obchodník je povinný kontaktovať platobnú inštitúciu za účelom preverenia výsledku spracovania platby resp. inej odpovede.

3.3 Implementácia na strane obchodníka

Požiadavky na implementáciu:

- presmerovanie na Internet bankingový server nie je možné cez iframe.
- obchodník môže požiadavky zasielať na nižšie uvedené URL služby platobnej brány metódou GET alebo POST cez protokol HTTPS
- parametre budú kódované vo forme application/x-www-form-urlencoded
- každá požiadavka musí obsahovať autentifikačný kód HMAC
- odpovede servera platobnej inštitúcie obsahujú autentifikačný kód HMAC a digitálny podpis ECDSA, ktoré je povinný obchodník overiť
- obchodník od platobnej inštitúcie obdrži pri podpise zmluvy svoj identifikátor **`vt`** a **`bezpečnostný kľúč`**.

3.3.1 Zaslanie požiadavky prostredníctvom HTML formulára

Server obchodníka vygeneruje stránku so skrytým formulárom, ktorý obsahuje **`input`** polia typu **`hidden`** pre každý vstupný parameter.
Pre formulár sa odporúča nastaviť parameter **`method`** na hodnotu POST. V prípade, že ju daný web server nepodporuje, môže sa použiť hodnota GET.

Príklad:

```
<form action="[URL pre zvolené rozhranie]" method="POST">
     <input type=" hidden " name="vt" value="-1" />
     < input type =" hidden " name="amount" value="123.45" />
...
</form>
```

3.3.2 Zaslanie požiadavky zostavením URL

Server obchodníka vygeneruje URL odkaz, ktorý sa skladá z URL pre zvolené rozhranie a vstupných parametrov:

`[URL pre zvolené rozhranie]?[reťazec vstupných parametrov]`

Všeobecná URL pre zvolené rozhranie je v tvare:

`https://subdomena.pay-institution.eu/gateway/`

napr.: `https://ib.pay-institution.eu/gateway/`

Pre reťazec vstupných parametrov platí:

- hodnoty parametrov sú kódované štandardnou metódou URLEncode
- názvy parametrov sú od hodnôt oddelené znakom **`=`** (napr. vt=1)
- parametre sú oddelené znakom **`&`** (napr. vt =1&amount=123.45)

Príklad zaslanej požiadavky:

`https://ib.pay-institution.eu/gateway/?vt=1&amount=123.45&currency=EUR&vs=123456789&ss=987654321&cs=308&remail=vysledok_platby@obchod.sk&rurl=https://www.obchod.sk/vysledok_platby&hmac=8b6033ac6a619a61decf2b679d2ba5daccb1f684865b0c07dd7d55de0aea4336&timestamp=06032018124910`

3.3.3 Výpočet autentifikačného kódu HMAC

Server obchodníka musí:

- vypočítať autentifikačný kód HMAC a pridať ho k parametrom požiadavky zasielanej na server platobnej inštitúcie
- vypočítať autentifikačný kód HMAC a overiť vypočítanú hodnotu voči parametru HMAC v odpovedi zo servera platobnej inštitúcie.

V prípade, že sa hodnoty nezhodujú, musí odpoveď vyhodnotiť ako neplatnú a kontaktuje platobnú inštitúciu za účelom preverenia platby.

Postup výpočtu:

1. server obchodníka pripraví reťazec, ktorý je vstupom pre výpočet autentifikačného kódu HMAC (podľa popisu v podkapitolách nižšie)
2. z tohto reťazca vygeneruje hašovaný autentifikačný kód (HMAC) použitím:
    - kryptografickej funkcie SHA-256
    - 64 bajtového bezpečnostného kľúča, ktorý je zapísaný v hexadecimálnom tvare (128 znakov)

3.3.4 Overenie digitálneho podpisu ECDSA

Server obchodníka overí digitálny podpis ECDSA, ktorý sa nachádza v odpovediach zo servera platobnej inštitúcie. V prípade, že je overenie neúspešné, vyhodnotí odpoveď ako neplatnú.

Server obchodníka overí tento digitálny podpis nasledovne:

1. obchodník si stiahne verejné kľúče zo servera platobnej inštitúcie vo forme súboru a uloží ho na server.

   Poznámka:

   Platobná inštitúcia môže v prípade potreby zmeniť verejný kľúč. Všeobecná URL adresa, na ktorej je dostupný verejný kľúč:

   `https://subdomena.pay-institution.eu/gateway/ecdsa`

   napr. `https://ib.pay-institution.eu/gateway/ecdsa`

2. server obchodníka pripraví rovnaký reťazec ako pri overení HMAC a pripojí k nemu hodnotu HMAC (prijatú alebo vypočítanú hodnotu – musia byť zhodné)
3. overí digitálny podpis volaním OpenSSL funkcie (OpenSSL knižnice 1.0.0 a vyššie) pre overenie digitálneho podpisu ECDSA, ktorej vstupom je:
   - reťazec pre overenie digitálneho podpisu
   - voľba kryptografickej funkcie SHA-256
   - digitálny podpis zaslaný v odpovedi v parametri ECDSA

3.4 Platba cez platobnú bránu

3.4.1 Identifikácia platby

Obchodník musí platbu identifikovať jednou z týchto možností:

- variabilným symbolom a prípadne špecifickým a konštantným symbolom
- referenciou platiteľa (tento identifikátor platby bol zavedený v rámci SEPA platieb)

Zvolený identifikátor / identifikátory:

- zašle obchodník v požiadavke na platbu cez platobnú bránu
- zašle platobná inštitúcia obchodníkovi v návratovej URL a notifikačnom emaile

3.4.2 Požiadavka

Server obchodníka pošle požiadavku metódou GET alebo POST na URL adresu:

všeobecne: **`https://subdomena.pay-institution.eu/gateway`**

napríklad: `https://ib.pay-institution.eu/gateway`

Poznámka:

Pridaním jazyka za koniec URL je možné volať platobnú bránu v danom jazyku. Napríklad pre český jazyk bude URL odkaz v tvare *https://subdomena.pay-institution.eu/gateway/cz*. Povolené sú jazyky cz, pl a en, pričom predvolený jazyk je sk.

Vstupné parametre

Názov	Povinný	Popis	Dĺžka	Pravidlá	Príklad
vt	áno	Identifikátor obchodu *Jedinečné identifikačné číslo obchodu.*	1-11	Udeľuje platobná inštitúcia.	1
amount	áno	Suma platby *Suma, ktorú má zákazník previesť na účet obchodníka.*	9+2	- desatinné číslo - max. 9 miest pred oddeľovačom desatín - max. 2 desatinné miesta oddelené bodkou	123.45
currency	áno	Mena platby	3	- musí obsahovať kód meny podľa normy ISO 4217	EUR
vs	áno	Variabilný symbol	<10	- povolené znaky: 0-9	123456789
ss	nie	Špecifický symbol	<10	- povolené znaky: 0-9	987654321
cs	nie	Konštantný symbol	<4	- povolené znaky: 0-9	308
remail	nie	Emailová adresa pre zaslanie notifikácie o výsledku platby	<50	- môže obsahovať iba jednu emailovú adresu, platnú v súlade s RFC 2822 - v prípade, že hodnota prekročí 50 znakov, notifikačný email nebude odoslaný	vysledok@obchod.sk
rurl	áno	Návratová URL *URL adresa, na ktorú banka presmeruje zákazníka po vykonaní platby.*		- URL musí byť vytvorená v súlade s RFC 1738 a musí byť funkčná	https://www.obchod.sk/vysledok_platby
hmac	áno	Autentifikačný kód HMAC z parametrov: vt + amount + currency + vs + ss + cs + rurl + remail + timestamp	64	- platné znaky: 0-9 a-f	95d62075f3ce9c539b52e1c8fad45c4c15fd38701f1a12ade6122bb3ca40ed16

timestamp	áno	Timestamp (časová pečiatka) v UTC Server platobnej inštitúcie spracuje iba požiadavky, ktoré budú mať TIMESTAMP v intervale +/- 1 hodina voči UTC (GMT)	14	- vo formáte DDMMYYYYHHMISS *(DD-deň, MM-mesiac, YYYY-rok, HH-hodina, MI-minúta, SS-sekunda)*	21022018072746

3.4.3 Odpoveď na požiadavku

V prípade, že je požiadavka platná a služba platobnej brány je dostupná, zákazníkovi sa zobrazí aplikácia platobnej brány. Prostredníctvom aplikácie môže potvrdiť platbu na účet obchodníka. Po potvrdení alebo zrušení platby sa zákazníkovi zobrazí jedno z hlásení:

Hlásenie	**Hodnota „result“**	**Popis**
Vaša platba prebehla úspešne.	OK	
Vaša platba nebola spracovaná.	FAIL	Nastala chyba pri spracovaní. Zákazníkovi sa zobrazí aj dôvod chyby napr. nedostatok prostriedkov na účte.
Platba bola zrušená	CANCEL	Zobrazí sa v prípade, že zákazník platbu zrušil.

Obchodník si môže overiť stav platby týmito spôsobmi:

- kontrolou parametrov v návratovej URL
- kontrolou notifikačného emailu (pokiaľ bol korektne vyplnený parameter `remail` v požiadavke)
- online dopytom na server platobnej inštitúcie
- kontrolou pohybov na účte prostredníctvom Internet bankingu alebo mobilnej aplikácie

3.4.5 Návratová URL a parametre

Aplikácia platobnej brány zobrazí zákazníkovi výsledok platby. V prípade, že zákazník nezatvorí okno prehliadača, ale stlačí tlačidlo **`Pokračovať`**, bude presmerovaný na URL stránky obchodníka (zaslanú vo vstupnom parametri `rurl`). Návratová URL obsahuje parametre, vďaka ktorým môže server obchodníka overiť stav platby.

Názov	Popis	Príklad
amount	Suma platby zaslaná v požiadavke	123.45
currency	Mena platby zaslaná v požiadavke	EUR
vs	Variabilný symbol (Identifikátor)	123456789
ss	Špecifický symbol	987654321
cs	Konštantný symbol	308
result	Kód výsledku platby: *OK - platba prebehla úspešne* *FAIL - platba nebola úspešná* *CANCEL - zákazník zrušil platbu*	CANCEL
pid	Jednoznačný identifikátor platby na strane platobnej inštitúcii *Pomocou tohto identifikátora je možné jednoducho opakovane overiť stav platby prostredníctvom rozhrania. Parameter sa v odpovedi nachádza, pokiaľ je výsledok platby OK.*	54784
timestamp	Timestamp zaslaný v požiadavke	14032018125005
hmac	Reťazcom pre výpočet HMAC je reťazec hodnôt parametrov: amount + currency + vs + ss + cs + result + pid + timestamp	9b559bb38b7471f7f84dec827a8ad17700806294422cb370a39921e2ec313178
ecdsa	Reťazcom pre výpočet ECDSA je reťazec hodnôt parametrov: amount + currency + vs + ss + cs + result + pid + timestamp + hmac	304502201fb6e376a6b7bb8fe34d931e5e409721c80fb481710dac947cf913a6a3f98f5e022100f1f3066ce4a87cd139742edcd15bdb0c100ccbd7b524e6a1a866d81c273472f7

3.4.6 Notifikačný email

Server platobnej inštitúcie odošle notifikačný email na adresu uvedenú v parametri **`remail`**.

Telo emailu obsahuje reťazec parametrov, rovnakých ako návratová URL:

- názvy parametrov sú od hodnôt oddelené znakom **=**
- parametre navzájom sú oddelené medzerou
- pokiaľ parameter nie je vyplnený, nebude sa v reťazci nachádzať (ani názov ani hodnota)
- parametre budú zoradené v tomto poradí:
    1. amount
    2. currency
    3. vs
    4. ss
    5. cs
    6. result
    7. pid
    8. timestamp
    9. hmac
    10. ecdsa

4. Príklady

Pre kontrolu generovania HMAC platobná inštitúcia poskytuje možnosť overiť si správne generovanie HMAC cez URL adresu:

(všeobecne) **`https://subdomena.pay-institution.eu/gateway/example`**

Pre konkrétnu subdoménu to je URL adresa:

`https://ib.pay-institution.eu/gateway/example`

4.1 PHP

Výpočet HMAC

```
$keyBytes = pack("H*" , $key); // konverzia do binárneho formátu
$signature = hash_hmac("sha256", $stringToSign, $keyBytes);
```

Výpočet ECDSA

```
$verified = openssl_verify($stringToVerify, pack("H*", $ECDSA),
$publicKey, "sha256");

if ($verified === 1) {
    // odpoveď verifikovaná
}
```

4.2 Java

Výpočet HMAC

```
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;

byte[] keyBytes = hex2bytes(key); // konverzia do binárneho formátu
SecretKeySpec keySpec = new SecretKeySpec(keyBytes, "HmacSHA256");
Mac mac = Mac.getInstance("HmacSHA256");
mac.init(keySpec);
byte[] hmacBin = mac.doFinal(stringToSign.getBytes());
String signature = bytes2hex(hmacBin); // konverzia do hexadecimálneho
reťazca
```

Výpočet ECDSA

```
import java.math.BigInteger;
import java.security.KeyFactory;
import java.security.PublicKey;
import java.security.Signature;
import java.security.spec.X509EncodedKeySpec;
import javax.xml.bind.DatatypeConverter;

publicKey = publicKey.replaceAll("-----(BEGIN|END).*", "").trim();
X509EncodedKeySpec spec = new X509EncodedKeySpec(
DatatypeConverter.parseBase64Binary(publicKey));
KeyFactory keyFactory = KeyFactory.getInstance("EC");
PublicKey pKey = keyFactory.generatePublic(spec);
Signature ecdsaSign = Signature.getInstance("SHA256withECDSA");
ecdsaSign.initVerify(pKey);
ecdsaSign.update(stringToVerify.getBytes("UTF-8"));

if (ecdsaSign.verify(new BigInteger(ECDSA, 16).toByteArray())) {
    // odpoveď verifikovaná
}
```