

- [ePay.bg комуникационен пакет за търговци](#)
 - [I. Заявка за плащане от WEB търговец](#)
 - [II. Заявка за плащане от WEB търговец \(директно плащане с кредитна карта\)](#)
 - [III. Заявка за плащане - ``Свободен превод``](#)
 - [IV. Заявка за плащане - ``Вносна бележка``](#)
 - [EasyPay комуникационен пакет за WEB търговци](#)
 - [I. Заявка за плащане през EasyPay от WEB търговец](#)
 - [Подаване на задължение за плащане към бюджетни организации](#)
-

ePay.bg комуникационен пакет за ТЪРГОВЦИ

ePay.bg <office@epay.bg>, <epay2_demo@epay.bg>

Пакетът се състои от следните файлове:

demo.cgi

- примерна заявка за плащане написана на Perl

demo.php

- примерна заявка за плащане написана на PHP

demo_recv.cgi

- пример за получаване на известие относно регистрирано искане за плащане (Платено или Отказано) написан на Perl

demo_recv.php

- пример за получаване на известие относно регистрирано искане за плащане (Платено или Отказано) написан на PHP

demo.html

- примерна заявка за Свободен превод към регистриран потребител/търговец

demo_vnbel.cgi

- примерна заявка за Вносна бележка

I. Заявка за плащане от WEB търговец

Схема:

1. Клиент на WEB търговец прави поръчка в електронния му магазин
2. След като е готов с поръчката, клиентът формира искане да плати
3. Търговецът изготвя пакет - заявка за плащане и препраща клиента към ePay.bg системата, като подава по описания по-долу начин заявката за плащане
4. След успешен вход на клиента в ePay.bg системата, заявката за плащане (ако е коректно подадена) се **регистрира/записва**, като чакащо задължение към дадения търговец. **Ако клиентът се откаже без да прави вход или не направи успешен такъв в ePay.bg, заявката за плащане НЕ се записва в системата**
5. Клиентът може да плати задължението или да го остави за по-късно
6. ePay.bg системата следи за статуса на **регистрираните/записаните** чакащи задължения и при плащане, отказване или изтичане се изпраща известие на търговеца за тях. Известията от системата се изпращат на портове 80 за HTTP и 443 за HTTPS.
7. При получаване на известие от ePay.bg, търговецът трябва да формира съответен отговор. Ако известието се получи повторно се повтаря първият върнат отговор.

Търговец	---[ЗАЯВКА ЗА ПЛАЩАНЕ]--->	ePay.bg
Клиент	---[ВХОД]--[ЗАПИС НА ЗАЯВКАТА]--->	ePay.bg
ePay.bg	---[ИЗВЕСТИЕ]--->	Търговец
Търговец	---[ОТГОВОР НА ИЗВЕСТИЕ]--->	ePay.bg

Всеки регистриран в системата търговец има генерирани **буквено-цифрена секретна дума с дължина 64 и идентификационен номер (КИН)**. Търговецът може да ги види в личните си данни, без да може да ги променя.

A. Заявката за плащане се изпраща, като HTTP POST заявка към <https://www.epay.bg/>

Пример:

```
<form action="https://www.epay.bg/" method=post>
<input type=hidden name=PAGE value="paylogin">
<input type=hidden name=ENCODED value="[ENCODED]">
<input type=hidden name=CHECKSUM value="[CHECKSUM]">
<input type=hidden name=URL_OK value="http://...">
<input type=hidden name=URL_CANCEL value="http://...">
<input type=submit>
</form>
```

(*) За версия на английски език заявката се изпраща на адрес: <https://www.epay.bg/en/>

(*) Ако се използва ePay.bg DEMO системата на адрес: <https://demo.epay.bg/>, то **ТЕСТОВИТЕ** заявки за плащане се изпращат към **него!!!**

Задължителните полета са: PAGE, ENCODED и CHECKSUM, като стойността на PAGE задължително е "paylogin".

Полетата URL_OK и URL_CANCEL са опционални:

URL_OK	- URL, на което клиента да бъде препратен в случай, че потвърди плащането (не гарантира, че плащането е извършено)
URL_CANCEL	- URL, на което клиента да бъде препратен в случай, че откаже плащането за момента (може да го плати или откаже)

по-късно, но не след указаната дата от търговеца)

Подписване на заявката за плащане

ENCODED - кодирана с base64 (RFC 3548) заявка за плащане, EOL=''
CHECKSUM - контролна сума върху ENCODED, генерирана като HMAC с
алгоритъм SHA-1 и секретната дума на търговеца.

Perl примерен код: {

```
# Кодиране на заявката
$ENCODED = encode_base64('DATA', ''); # '' за EOL (def. е "\n")

# Генериране на контролна сума
$CHECKSUM = hmac_hex($ENCODED, $secret, \&sha1);

}
```

PHP примерен код: {

```
# Кодиране на заявката
$ENCODED = base64_encode('DATA');

# Генериране на контролна сума
$CHECKSUM = hmac('sha1', $ENCODED, $secret);
# кода на функцията hmac може да видите в demo.php

}
```

Примерна заявка за плащане:

MIN=1000000000 (задължително - Идентификационен номер на търговеца)
EMAIL=a@merch.bg (задължително - E-mail на търговеца в системата)

(*) За идентификация на търговеца се подава задължително MIN или EMAIL

INVOICE=123456 (задължително Номер фактура)
AMOUNT=22.80 (задължително Сума)
CURRENCY=BGN (опционално)
EXP_TIME=01.08.2020 (задължително Крайна дата/час за плащане)
DESCR=Test (опционално Описание до 100 символа)
ENCODING=CP1251 (задължително encoding на DESCR параметъра cp1251 или utf-8)

Допълнителни опции:

а) отстъпки при плащане с определени карти
DISCOUNT=cardbin1,cardbin2,cardbin3:amount1
DISCOUNT=cardbin4,cardbin5,cardbin6:amount2

б) https://www.epay.bg/?page=let_pay --отложени плащания (само за търговци с микросметка)
HOLD=1
HOLD_START_DATE=DD.MM.YYYY
(дата на доставка - ако не е подадена се приема текущата дата)

Не е задължително да са подредени точно в този ред.

MIN - съответства на КИН в личните данни на търговеца
EMAIL - e-mail на търговеца, с който той е регистриран
INVOICE - само цифри
AMOUNT - валидна сума > 0.01 (например: 22, 22.8, 22.80)
CURRENCY - приемана валута е BGN, опционално
DESCR - CP1251 символи, ако не е подаден друг ENCODING

ENCODING - приема се само CP1251 или utf-8

За полето EXP_TIME е валиден следният формат:

```
EXP_TIME=DD.MM.YYYY[ hh:mm[:ss]]
```

```
EXP_TIME=01.08.2020
```

```
EXP_TIME=01.08.2020 23:15 (може да се подаде и с час:мин)
```

```
EXP_TIME=01.08.2020 23:15:30 (може да се подаде и с час:мин:сек)
```

След като заявката за плащане е регистрирана за клиента, то системата ще извести търговеца за състоянието на плащането: "Платено", "Отказано" или "Изтекло". Ако клиентът не потвърди или откаже искането преди подадената крайна дата, то ще се маркира като изтекло.

Заявка с даден INVOICE може да влезе в системата ЕДИН единствен път и чака за "потвърждаване" или "отказване" от клиента.

В. Известието на системата се изпраща на зададено от търговеца URL, като HTTP POST заявка, на която търговецът връща отговор в същата HTTP сесия.

Подписване на известието за плащане

```
ENCODED      - кодирано с base64 (RFC 3548) известие, EOL=''
CHECKSUM     - контролна сума върху ENCODED, генерирана като HMAC с
               алгоритъм SHA-1 и секретната дума на търговеца.
```

Perl примерен код: {

```
$data = decode_base64($ENCODED);

# Калкулиране на контролната сума
# Трябва $CHECKSUM_CALC == $CHECKSUM
$CHECKSUM_CALC = hmac_hex($ENCODED, $secret, \&sha1);

}
```

PHP примерен код: {

```
$data = base64_decode($ENCODED)

# Калкулиране на контролната сума
# Трябва $CHECKSUM_CALC == $CHECKSUM
$CHECKSUM_CALC = hmac ('sha1', $ENCODED, $secret);
# кода на функцията hmac може да видите във файл demo.php

}
```

Примерно известие от ePay.bg:

```
INVOICE=123456:STATUS=PAID:PAY_TIME=YYYYMMDDhhmmss:STAN=[6 числа]:BCODE=[6 числа/букви]
INVOICE=123457:STATUS=DENIED
INVOICE=123457:STATUS=EXPIRED
```

(*) ако плащането е с ползване на отстъпка (примерно с БИН на карта cardbin2 и сума amount1), то извес
INVOICE=123456:STATUS=PAID:PAY_TIME=YYYYMMDDhhmmss:STAN=[6 числа]:BCODE=[6 числа/букви]:AMOUNT=amount

```
STATUS=[PAID | DENIED | EXPIRED] - Платено | Отказано | Изтекло
PAY_TIME                        - Дата/час/сек на плащането
STAN                             - Номер транзакция, присъства само при плащане с карта
BCODE                           - Авторизационен код на БОРИКА, присъства само при плащане с карта
AMOUNT                          - Платена сума, подава се CAMO при отстъпка
```

VIN

- БИН на карта, подава се CAMO при отстъпка

За **всеки номер на фактура в известието**, търговецът трябва да върне статус:

OK - ако всичко е наред

или

ERR - за грешка.

При връщане на OK системата спира да изпраща известие за съответната фактура. Нормално това е отговорът, който търговецът трябва да върне на известие от системата. ERR отговор се връща **по изключение**, при възникване на някаква грешка, поради която търговецът не е успял да запише известието от ePay.

Примерен отговор на търговеца:

```
INVOICE=123456:STATUS=OK  
INVOICE=123457:STATUS=ERR
```

Пример:

ERR=описание за глобалната грешка (пример - грешна CHECKSUM)

Ако търговецът не е заявил URL, на което да получава известия за плащанията, не иска или няма възможност да обработва тези известия, то търговецът може да си вижда статуса на исканията за плащане в ePay.bg системата.

Ако ePay.bg не маркира дадена фактура, като получена от търговеца (примерно върнат статус ERR или пропаднала комуникация) системата ще се опита да изпрати пропадналите известия отново.

Схема за изпращане на известия по дадена фактура:

- 1) 6 опита през < 1 минута
- 2) 6 опита през 5 минути
- 3) 8 опита през 15 минути
- 4) 9 опита през 1 час
- 5) 1 опит на ден

Системата спира да изпраща известие за дадена фактура, ако то не бъде получено от търговеца в продължение на 14 дена.

II. Заявка за плащане от WEB търговец (директно плащане с кредитна карта)

Схемата е същата, като при **(I) Заявка за плащане от WEB търговец**, с тази разлика, че PAGE=credit_paydirect.

Пример:

```
<form action="https://www.epay.bg/" method=post>  
<input type=hidden name=PAGE value="credit_paydirect">  
<input type=hidden name=LANG value="[LANG]">
```

```
<input type=hidden name=ENCODED value="[ENCODED]">
<input type=hidden name=CHECKSUM value="[CHECKSUM]">
<input type=hidden name=URL_OK value="http://...">
<input type=hidden name=URL_CANCEL value="http://...">
<input type=submit>
</form>
```

LANG = bg | en

(*)Ако се използва ePay.bg DEMO системата на адрес:

<https://demo.epay.bg/> то ТЕСТОВИТЕ заявки за плащане се изпращат към него!!!

За тестов номер на карта се въвеждат само 1-ци, за дата на валидност се въвежда 11/11.

III. Заявка за плащане - "Свободен превод"

Платената сума от клиента постъпва по Микросметка на получателя. Необходимо е клиента и получателя на плащането да са регистрирани в ePay.bg

Заявката за "Свободен превод" се изпраща, като HTTP POST заявка към <https://www.epay.bg/>

Пример:

```
<form action="https://www.epay.bg/" method=post>
<input type=hidden name=PAGE value="paylogin">
<input type=hidden name=MIN value="[MIN]">
<input type=hidden name=INVOICE value="[INVOICE]">
<input type=hidden name=TOTAL value="[TOTAL]">
<input type=hidden name=DESCR value="[DESCR]">
<input type=hidden name=ENCODING value="[ENCODING]">
<input type=hidden name=URL_OK value="http://...">
<input type=hidden name=URL_CANCEL value="http://...">
<input type=submit>
</form>
```

Задължително стойността на PAGE е "paylogin"

MIN (задължително съответства на КИМ в личните данни на търговеца)
 INVOICE (опционално Номер фактура, само цифри)
 TOTAL (задължително Сума)- валидна сума > 0.01 (например: 22, 22.8, 22.80)
 DESCR (опционално Описание до 100 символа, енкодинг CP1251)
 ENCODING (задължително encoding на DESCR параметъра CP1251 или utf-8)

Еквивалентно на TOTAL се приема и AMOUNT.

Допълнителни опции:

https://www.epay.bg/?page=let_pay --отложени плащания (само за търговци с микросметка)

HOLD=1

HOLD_START_DATE=DD.MM.YYYY

(дата на доставка - ако не е подадена се приема текущата дата)

Схема:

Наредител ---[ЗАЯВКА СВОБОДЕН ПРЕВОД]---> ePay.bg

Клиент ---[ВХОД]--[ЗАПИС НА ЗАЯВКАТА]--> ePay.bg

Полетата URL_OK и URL_CANCEL са опционални:

URL_OK - URL, на което клиента да бъде препратен в случай, че потвърди плащането (не гарантира, че плащането е извършено)
 URL_CANCEL - URL, на което клиента да бъде препратен в случай, че откаже плащането

При тази схема на плащане няма подписване на завките и изпращане на известия към търговеца за статуса на плащането.

IV. Заявка за плащане - "Вносна бележка"

Платените суми от клиенти на ePay.bg постъпват по посочена от търговеца банкова сметка

Заявката за "Вносна бележка" се изпраща, като HTTP POST заявка към <https://www.epay.bg/>

Пример:

```
<form action="https://www.epay.bg/" method=post>
<input type=hidden name=PAGE value="paylogin">
<input type=hidden name=MERCHANT value="[MERCHANT]">
<input type=hidden name=IBAN value="[IBAN]">
<input type=hidden name=BIC value="[BIC]">
<input type=hidden name=AMOUNT value="[AMOUNT]">
<input type=hidden name=STATEMENT value="[STATEMENT]">
<input type=hidden name=PSTATEMENT value="[PSTATEMENT]">
<input type=hidden name=URL_OK value="http://...">
<input type=hidden name=URL_CANCEL value="http://...">
<input type=submit>
</form>
```

Задължително стойността на PAGE е ``paylogin``.

MERCHANT	(задължително Получател на превода (кирилица, латиница, цифри, интервали, "-", " ", "."))
IBAN	(задължително валиден IBAN на получателя)
BIC	(задължително BIC на банката получател)
TOTAL	(задължително Сума (валидна сума > 0.01 - например: 22, 22.8, 22.80))
STATEMENT	(задължително Основание (кирилица, латиница, цифри, интервали, "-", " ", "."))
PSTATEMENT	(ако е нужно Вид плащане - 6 цифри, валиден вид плащане)

Схема:

Наредител --[ЗАЯВКА ВНОСНА БЕЛЕЖКА]--> ePay.bg
 Клиент --[ВХОД]--[ЗАПИС НА ЗАЯВКАТА]--> ePay.bg

Полетата URL_OK и URL_CANCEL са опционални:

URL_OK - URL, на което клиента да бъде препратен в случай, че потвърди плащането (не гарантира, че плащането е извършено)
 URL_CANCEL - URL, на което клиента да бъде препратен в случай, че откаже плащането

=====

[To top](#)

EasyPay комуникационен пакет за WEB търговци

***Забележка:** При тази схема на плащане от нашата система се връща 10-цифрения код, с който задължението може да се плати, както на гише на EasyPay, така и на банкомат от меню "B-Pay". Моля, уведомете ни, когато преминете към работа в реална среда!

I. Заявка за плащане от WEB търговец

- **Схема:**

1. Клиент на WEB търговец прави поръчка в електронния му магазин
2. След като е готов с поръчката, клиентът формира искане да плати в офис на EasyPay
3. Търговецът изготвя пакет /заявка за плащане/, който изпраща към дадено URL, като HTTP GET заявка скрито от клиента. В същата HTTP сесия, ако всичко със заявката е коректно, ePay.bg ще върне търговеца "Код за плащане в EasyPay". Този код се получава чрез изпълнение на функция за четене от файл, като след прочитането му, търговецът го показва на своя сайт и с него клиентът посещава офис на EasyPay, за да извърши плащането. Кодовете, генерирани за плащане през EasyPay, могат да бъдат заплатени и през ePay на адрес: <https://www.epay.bg/?page=payfly>
4. ePay.bg системата следи за статуса на регистрираните/записаните чакащи задължения и при плащане или изтичане на времето за плащане се изпраща известие на търговеца за тях. Известията от системата се изпращат на портове 80 за HTTP и 443 за HTTPS.
5. При получаване на известие от ePay.bg, търговецът трябва да формира съответен отговор

Търговец	---[ЗАЯВКА ЗА ПЛАЩАНЕ]--->	ePay.bg
ePay.bg	---[КОД ЗА ПЛАЩАНЕ]--->	Търговец
Клиент	---[ПЛАЩАНЕ]--->	EasyPay
ePay.bg	---[ИЗВЕСТИЕ]--->	Търговец
Търговец	---[ОТГОВОР НА ИЗВЕСТИЕ]--->	ePay.bg

Всеки регистриран в системата търговец има генерирани **буквено-цифрена секретна дума с дължина 64 и идентификационен номер (КИН)**.

Търговецът може да ги види в личните си данни без да може да ги променя.

A. Заявката за плащане се изпраща, като HTTP GET заявка към

https://www.epay.bg/ezp/reg_bill.cgi

* за тестове се изпраща на https://demo.epay.bg/ezp/reg_bill.cgi

Подавани параметри:

ENCODED - кодирано с base64 (RFC 3548) известие, EOL=''

CHECKSUM - контролна сума върху ENCODED, генерирана като HMAC с алгоритъм SHA-1 и секретната дума на търговеца.

Perl примерен код: {

```
# Кодиране на заявката
$ENCODED = encode_base64('DATA', ''); # '' за EOL (def. е "\n")
```



```
# Генериране на контролна сума
$CHECKSUM = hmac_hex($ENCODED, $secret, \&sha1);

}
```

PHP примерен код: {

```
# Кодирание на заявката
$ENCODED = base64_encode('DATA');

# Генериране на контролна сума
$CHECKSUM = hmac('sha1', $ENCODED, $secret);
# кода на функцията hmac може да видите в demo.php

}
```

Пример:

```
MIN=10000000000          (MIN - идентификационен номер на търговеца, съответства на КИН в личните да
EMAIL=a@merch.bg        (EMAIL - на търговеца в системата )
```

За идентификация на търговеца се подава задължително MIN или EMAIL

```
INVOICE=123456          (задължително Номер фактура, само цифри )
AMOUNT=22.80           (задължително Сума(валидна сума > 0.01 - например: 22, 22.8, 22.80 ))
EXP_TIME=01.08.2020    (задължително крайна дата/час за плащане, до 30 дни след датата на заявката
DESCR=Test             (опционално Описание до 100 символа, енкодинг CP1251 )
```

Не е задължително да са подредени точно в този ред

За полето EXP_TIME е валиден следният формат:

```
EXP_TIME=DD.MM.YYYY [hh:mm [:ss]]
EXP_TIME=01.08.2020
EXP_TIME=01.08.2020 23:15 ( може да се подаде и с час:мин )
EXP_TIME=01.08.2020 23:15:30 ( може да се подаде и с час:мин:сек )
```

Примерен отговор на заявка:

```
IDN=1234567890 (Код за плащане - десет цифри)
или
ERR=Описание на грешка
При подаване на един и същи номер на фактура INVOICE системата връща един и същ десет цифрен код.
```

* За имитиране на извършване на плащане в офис на EasyPay се прави следната GET заявка:

https://demo.epay.bg/ezp/pay_bill.cgi?ACTION=PAY&IDN=получения_код_за_плащане

В. Известието на системата се изпраща на зададено от търговеца URL, като HTTP POST заявка, на която търговеца връща отговор в същата HTTP сесия

Подписване на известието за плащане

```
ENCODED - кодирано с base64 (RFC 3548) известие, EOL= ''
CHECKSUM - контролна сума върху ENCODED, генерирана като HMAC
           с алгоритъм SHA-1 и секретната дума на търговеца.
```

```
$data = decode_base64($ENCODED);

# Калкулиране на контролната сума
# Трябва $CHECKSUM_CALC == $CHECKSUM
$CHECKSUM_CALC = hmac_hex($ENCODED, $secret, \&sha1);

}
```

PHP примерен код: {

```
$data = base64_decode($ENCODED);  
  
# Калкулиране на контролната сума  
# Трябва $CHECKSUM_CALC == $CHECKSUM  
$CHECKSUM_CALC = hmac('sha1', $ENCODED, $secret);  
# кода на функцията hmac може да видите в demo.php  
  
}
```

Примерно известие от ePay.bg:

```
INVOICE=123456:STATUS=PAID:PAY_TIME=YYYYMMDDhhmmss:STAN=[6 числа]:BCODE=[6 числа/букви]  
INVOICE=123457:STATUS=EXPIRED
```

STAN - номер транзакция - само при плащане с карта
BCODE - авторизационен код на БОРИКА, само при плащане с карта
STAN и BCODE са 000000, когато плащането не е извършено от карта, възможно е и да липсват

За всеки номер на фактура в известието, търговецът трябва да върне статус: **OK** (ако всичко е наред), или **ERR**(за грешка). При връщане на OK системата спира да изпраща известие за съответната фактура. Нормално се връща отговор OK, отговор ERR се връща **по изключение**, при възникване на някаква грешка, поради която търговецът не е успял да запише известието за конкретната фактура.

Пример:

```
INVOICE=123456:STATUS=OK  
INVOICE=123457:STATUS=ERR
```

ERR=описание за глобалната грешка (пример - грешна CHECKSUM) Ако ePay.bg не маркира дадена фактура като получена от търговеца (примерно върнат статус ERR или пропаднала комуникация) системата ще се опита да изпрати пропадналите известия отново.

Схема за изпращане на известия по дадена фактура:

- 1) 6 опита през < 1 минута
- 2) 6 опита през 5 минути
- 3) 8 опита през 15 минути
- 4) 9 опита през 1 час
- 5) 1 опит на ден

Системата спира да изпраща известие за дадена фактура, ако то не бъде получено от търговеца в продължение на 14 дена.

Схемата за получаване на десет цифрения код за плащане на банкомат е същата. Стъпките, които трябва да извърши клиентът след получаване на кода, са:

1. Поставя картата в банкомата
2. Избира меню "Други Услуги "
3. Избира меню "B-Pay"
4. Въвежда код на търговец 60000

5. Въвежда десет цифрения код за плащане

=====

[To top](#)

Подаване на задължение за плащане към бюджетни организации

• Схема:

1. Заявка за плащане към бюджетна организация (търговец)
2. След като е готов с поръчката, клиентът формира искане за плащане
3. Търговецът изготвя пакет /заявка за плащане/, който изпраща към дадено URL, като HTTP GET заявка скрито от клиента. В същата HTTP сесия, ако всичко със заявката е коректно, ePay.bg ще върне търговеца "Код за плащане". Търговецът показва на своя сайт този код и с него клиентът извършва плащането
4. ePay.bg системата следи за статуса на регистрираните/записаните чакащи задължения и при плащане, отказване или изтичане на времето за плащане се изпраща известие на търговеца за тях. Известията от системата се изпращат на портове 80 за HTTP и 443 за HTTPS.
5. При получаване на известие от ePay.bg, търговецът трябва да формира съответен отговор

Търговец	---[ЗАЯВКА ЗА ПЛАЩАНЕ]--->	ePay.bg
ePay.bg	---[КОД ЗА ПЛАЩАНЕ]--->	Търговец
Клиент	---[ПЛАЩАНЕ]--->	EasyPay
ePay.bg	---[ИЗВЕСТИЕ]--->	Търговец
Търговец	---[ОТГОВОР НА ИЗВЕСТИЕ]--->	ePay.bg

Всеки регистриран в системата търговец има генерирани **буквено-цифрена секретна дума с дължина 64 и идентификационен номер (КИН)**.

Търговецът може да ги види в личните си данни без да може да ги променя.

A. Заявката за плащане се изпраща, като HTTP GET заявка към https://www.epay.bg/ezp/reg_vnbel.cgi

* за тестове се изпраща на https://demo.epay.bg/ezp/reg_bill.cgi

Подавани параметри:

ENCODED - кодирано с base64 (RFC 3548) известие, EOL=''
CHECKSUM - контролна сума върху ENCODED, генерирана като HMAC с алгоритъм SHA-1 и секретната дума на търговеца.

Perl примерен код: {

```
# Кодиране на заявката
$ENCODED = encode_base64('DATA', ''); # '' за EOL (def. е "\n")

# Генериране на контролна сума
$CHECKSUM = hmac_hex($ENCODED, $secret, \&sha1);

}
```

PHP примерен код: {

```
# Кодиране на заявката
$ENCODED = base64_encode('DATA');

# Генериране на контролна сума
$CHECKSUM = hmac('sha1', $ENCODED, $secret);
# кода на функцията hmac може да видите в demo.php
```

```
}
```

Примерна заявка:

```
MIN = 1000000000 (MIN - само цифри, идентификационен номер на търговеца )
EMAIL = a@merch.bg (EMAIL - на търговеца в системата )
```

За идентификация на търговеца се подава задължително MIN или EMAIL

```
INVOICE = 123456 (задължително Номер фактура )
AMOUNT = 22.80 (задължително Сума) -валидна сума > 0.01 (например: 22, 22.8, 22.80)
EXP_TIME = 01.08.2020 (задължително крайна дата/час за плащане, до 30 дни след датата на заявка
```

За полето EXP_TIME е валиден следният формат:

```
EXP_TIME=DD.MM.YYYY [hh:mm [:ss]]
EXP_TIME=01.08.2020
EXP_TIME=01.08.2020 23:15 ( може да се подаде и с час:мин )
EXP_TIME=01.08.2020 23:15:30 ( може да се подаде и с час:мин:сек )
DESCR = Test (опционално Описание до 100 CP1251 символа )
MERCHANT = (задължително Получател на нареждането)
IBAN = (задължително IBAN на получателя(
BIC = (задължително BIC на банката получател(
PSTATEMENT = (задължително Код на плащането(
STATEMENT = (задължително Описание на плащането(
OBLIG_PERSON = (задължително Име на задълженото лице до 26 символа(
BULSTAT = (само едно от трите ЕГН, БУЛСТАТ или ЛНЧ(
EGN или LNC или BULSTAT = (само едно от трите ЕГН, ЛНЧ или БУЛСТАТ(
DOC_NO = (задължително Вид и номер документ 1 - слепени(
DOC_DATE = (Дата на док.: задължително при Вид на док. - 2|3|6(
DATE_BEGIN = (Начало на период: задължително при Вид на док. - 1|2|4|5(
DATE_END = (Край на период: задължително при Вид на док. - 1|2|4|5(
```

Не е задължително да са подредени точно в този ред.

Ако платежното е многоредово, то параметъра AMOUNT се заменя с TOTAL, а за всяко отделно плащане сумат съответно SUM1, SUM2, SUM3 и т.н

Примерен отговор на заявка:

```
IDN = 1234567890 (десет цифри)
или
ERR = Описание на грешка
```

След регистриране на заявката за плащане системата ще извести търговеца за състоянието на плащането - платено/отказано или изтекло. Ако клиентът не потвърди или откаже искането преди зададената крайна дата, то ще се маркира, като изтекло. Заявка с даден INVOICE може да влезе в системата един единствен път.

В. Известието на системата се изпраща на зададено от търговеца URL, като HTTP POST заявка, на която търговеца връща отговор в същата HTTP сесия

Подписване на известието за плащане

ENCODED - кодирано с base64 (RFC 3548) известие, EOL= ''
 CHECKSUM - контролна сума върху ENCODED, генерирана като HMAC
 с алгоритъм SHA-1 и секретната дума на търговеца.

```
$data = decode_base64($ENCODED);

# Калкулиране на контролната сума
# Трябва $CHECKSUM_CALC == $CHECKSUM
$CHECKSUM_CALC = hmac_hex($ENCODED, $secret, \&sha1);

}
```

PHP примерен код: {

```
$data = base64_decode($ENCODED);

# Калкулиране на контролната сума
# Трябва $CHECKSUM_CALC == $CHECKSUM
$CHECKSUM_CALC = hmac('sha1', $ENCODED, $secret);
# кода на функцията hmac може да видите в demo.php

}
```

Примерно известие на системата

INVOICE=123456:STATUS=PAID:PAY_TIME=YYYYMMDDhhmmss:STAN=[6 числа]:BCODE=[6 числа/букви] ---> Платено
 INVOICE=123457:STATUS=EXPIRED ---> Изтекло
 INVOICE=123457:STATUS=DENIED ---> Отказано

STAN - номер транзакция, само при плащане с карта
 BCODE - авторизационен код на БОРИКА, само при плащане с карта
 STAN и BCODE са 000000, когато плащането не е извършено от карта.

За всеки номер на фактура в известието, търговецът трябва да върне статус: **OK** (ако всичко е наред), или **ERR**(за грешка). При връщане на OK системата спира да изпраща известие за съответната фактура. Нормално се връща отговор OK, отговор ERR се връща **по изключение**, при възникване на някаква грешка, поради която търговецът не е успял да запише известието за конкретната фактура.

Примерен отговор на търговеца

Ако нещо не е коректно в известието изпратено от ePay.bg системата търговецът връща ERR=описание.

Пример:

INVOICE=123456:STATUS=OK
 INVOICE=123457:STATUS=ERR

ERR=описание за глобалната грешка (пример - грешна CHECKSUM) Ако ePay.bg не маркира дадена фактура като получена от търговеца (примерно върнат статус ERR или пропаднала комуникация) системата ще се опита да изпрати пропадналите известия отново.

Схема на изпращане на известия:

Схема за изпращане на известия по дадена фактура:

- 1) 6 опита през < 1 минута
- 2) 6 опита през 5 минути
- 3) 8 опита през 15 минути

- 4) 9 опита през 1 час
- 5) 1 опит на ден

Системата спира да изпраща известие за дадена фактура, ако то не бъде получено от търговеца в продължение на 14 дена.

КРАЙ

[To top](#)