Technical Specification
Mobile Content Billing (MCB)

(WB.001)

Author(s):

Erwin van den Boom (EvdB)
Silvan van der Veen (SvdV)
Michel Westerink (MW)

Version history

V4.0	MW	28/01/15
V3.0	MW	29/04/13
V2.2	SvdV	04/05/11
V2.1	MW	18/06/10
V2.0	MW	08/04/10
V1.9	MW	01/04/10
V1.8	MW	11/03/10
V1.7	MW	11/02/10
V1.6	MW	09/01/10
V1.5	MW	12/09/09
V1.4	SvdV	12/30/09
V1.3	SvdV	09/28/09
V1.2	SvdV	09/09/09
V1.1	SvdV	09/03/09
V1.0	EvdB	12/30/09

Table of Contents

1 Terminology

Mobile Content Billing
The process of charging money on the mobile internet from an end-user, which can be in the form of a subscription or "one off".

WAP push message
An SMS that contains an URL to a mobile website. Instead of a normal text message, the end-user gets a confirmation screen for visiting the website. When the user confirms, a connection to the site will be made automatically.

Payment
One single billing moment where the end-user is charged for the entire amount or a portion (based on operator tariffs) multiple times.

Transaction
A transaction consists of one or more payments:

- In a one-off payment there is one payment per transaction.
- In a subscription, multiple recurring payments make for one transaction. The transaction starts at the moment of signup and ends when the user signs off.

Code of Conduct
The Code of Conduct or "Gedragscode voor Betaalde mobiele diensten" (in Dutch) applies to all WAP billing services and promotional activities. The Code of Conduct can be downloaded from:
https://www.targetpay.com/docs/Gedragscode-voor-Betaalde-Mobdienst.pdf
Each country supported by TargetSMS has its own (self-)regulation with which your service must comply.

"Real" content billing
With "real" content billing we mean charging a specific amount with a single click. This means that we can charge money from end-users in the background without the need to send a Premium SMS. From the end-users perspective this is by far the most convenient way to pay on the mobile internet and also leads to an optimal ROI.

SMS content billing
Not all operators support "real" content billing. However, this doesn't mean that these end-users cannot be billed. Premium SMS can be used instead.

2 Rules, rules and rules

In the Netherlands there is a code of conduct. That document describes the “do's and don't” about Mobile Content Billing required screens, functionalities, layouts, complains, notifications and so on.

For clarity, those rules are not only formed by us. We are represented in the committee that compose those rules, but we are not alone. Also the Operators and the content providers are represented in this committee.

If you serve Mobile Content Billing as a pay-method for your customers, you must comply to this code of conduct. But beware, this code could be changed any time and any day, but don't worry most of those changes had to be done by us.

The changes for the most functionalities are, as noted above, will be done by us. But there are still some exceptions which we couldn't change likewise the look-and-feel of several required pages*, expressions on the main website of your product and so on.

On every update of this code of conduct, we'll send a reminder to all account owners. Most of time with a bullet list what is changed in the newest version of the code of conduct and what kind of changes should be done by you.

You are always responsible to comply the latest version of the code of conduct. Are there issues which are part of us and we didn't changed, please let us know on mcb@targetpay.com

On the page http://www.payinfo.nl/gedragscodes you will find the latest version of this code of conduct.

*If you use altered layouts, see the section 7.1 K

3 MCB In a nutshell...

This document describes how to use the MCB Payment API.

Before you can start using the MCB API, you have to setup a service in TargetPay.com
Here you specify the initial settings of your service which will result in a service ID.
https://www.targetpay.com/easy/wap/wizard

The transaction process in a nutshell:

1. The end-user orders a product on your website and opts for a payment using his smartphone.
2. Your service requests a transaction at TargetPay.com
3. Your service redirects the end-user to a payment screen
4. The end-user confirms the payment screen which completes the payment
5. The end-user is forwarded to the return URL where he receives the ordered product.

In case of a subscription service you can thereafter send requests to TargetPay.com in order to make follow-up payments. Optionally you can request the MCB gateway to inform you about all status changes, like a sign off.

4 Start API

In order to start a new transaction (that is an one-off payment or the start of a subscription), you need to call the start API using HTTP GET or HTTP POST:

4.1.1 Start request

https://www.targetpay.com/wap/start

Parameter description:

Parameters	Explanation
service	The ID of your service.
ip	The IP address of the end-user
amount*	Amount to be charged in eurocents, e.g. 100 = € 1,-
returnurl*	The end-user is redirected to this URL after paying or signing up. See section 8 Request payment or sign up status (pull)
notifyurl*	We will update the transaction status to this URL. For more information see paragraph 9 Using notifications (push)
cancelurl*	When the customer doesn't want to pay he/she could click on the link back to your page. When you are creating or updating the content billing service in Targetpay you were able to set a default cancelurl. This url could be overwritten by setting this cancelurl. This cancelurl will also be used to create a link when the customer clicked on "cancel". If there is an contentId given on the transaction this id will be added to the link.
check*	If you want to test your transaction request you can add "check". The script will execute all the checks for a normal transaction, but the transaction will not be created. Just for testing.
autofirstbilling*	If you want to charge the end user directly after sign up, you have to set this parameter to 1. Then you may omit the follow-up api for the first payment. **Note:** one-off payments are always executed directly
pnotifyurl*	If auto first billing is enabled, you will receive the payment id and payment status on this url.

*: These parameters are optional. If omitted they will be copied from the service settings. The bare parameters you need to use the start.php5 page are simply the ID of your service and the IP of the end-user.

Example:
https://www.targetpay.com/wap/start?service=1&ip=12.34.56.78&amount=150&returnurl=http%3A%2F%2Fwww.mysite.nl%2Freturn.php&notifyurl=http%3A%2F%2Fwww.mysite.nl%2Fnotify.php&cancelurl=http%3A%2F%2Fwww.mysite.nl%2Fcancel.php

4.2 Start response

The response will be *000000 transactionID*|*Redirecturl*
Example: 000000 61|http://www.targetpay.mobi/order.php5?trxid=61

If not successful, the response will be a (WBxxx) error code.
Example: WB011 No rtaff found for rtlo

Required Actions:

- Store the transaction ID for later use!
- Redirect the end-user to the payment URL

If you used the "check" parameter the only response will be 000001 *Transaction request is valid.*

5 Web payment API

The Web-payment API is intended for customizing the MSISDN page. Before you show this page to the end user you may want to use the Light weight User Identification as described in paragraph 12. If the user is MCB billable you omit this page.

The web payment API is located in the start API (start.php5) and requires additional parameters. The response is a little bit different pertaining to the normal response of the original transaction-request. Usually a redirect url is provided in the response, but in this case, that url will be send in the free sms to the end-user.

5.1 Web-payment API Parameters

The web payment API is the same as the normal MCB start request with a few extra parameters
http://www.targetpay.com/wap/start.php5

Parameter description:

Parameter	Required	type	Explanation
idbysms	Y	varchar	When Id by sms is enabled, an identification sms with a link to the order screen will be send to the given MSISDN
msisdn	Y	int	The destination of the free sms. Max length: 11
smstext	N	varchar	This text will be shown up in the sms-message which also contains the orderscreen url.

5.2 Web payment API response

If successful the response will be *000000 transactionID*
Example: 000000 61

If not, the response will be a (WBxxx) error code.
Example: *WB011 No rtaff found for rtlo*

Required Actions:

- Store the transaction ID for later use!

The Payment URL will be provided to the user via the free SMS.

If you used the "check" parameter the only response will be 000001 *Transaction request is valid.*

6 Submit follow-up payments

When the end-user confirmed for a subscription, payments will not be executed automatically. The start call only starts the membership. You charge the member for the first time directly after he confirms the membership (except when autofirstbilling is used).

6.1 Followup Request

With the follow-up api you may charge the end user for the amount and frequency you specified in your service.

To initiate a follow-up payment, call:
http://www.targetpay.com/wap/followup

Parameter description:

Parameters	Explanation
trxid	The transaction ID you received during the signup (see 4.2 Start respons)
service	The ID of your service (for security)
rtlo	Layoutcode (for security)
description	Description of the payment (max. 255 chars)
amount	Amount to be charged in eurocents, e.g. 100 = € 1,-. This must be Lower or equal to the amount specified in your service.
pnotifyurl*	The URL we call as soon as we know the status (See 6.2 Followup response)
ok*	This is a test-option. When this is set to 1, the follow-up will be act like a normal request, but just before the script will create the real payment, it will respond with “000001 Follow-up request is valid.” There will be no charge to the end user

* Those values are optional.

Example:
http://www.targetpay.com/wap/followup?trxid=1651556156&service=1&rtlo=1&description=Your+subscription+has+been+extended+for+another+week&amount=10&pnotifyurl=http%3A%2F%2Fwww.mysite.nl%2Fpnotifyurl.php

6.2 Followup response

The response of the followup api is 00000 + <paymentid>

Example:
00000 32094898

Required actions:

- Store the paymentid! This is the payment batch identifier that you can use to ***pull*** payment status information (see 8 Request payment or sign up status (pull)).
- Don't use this paymentid to create records for the status of the payment because when you use our push notifications with the status of the payment you receive the push status earlier than you will receive the paymentid from followup.php5

As soon as the status is known, the *pnotifyurl* URL will be called:

Example:
http://<pnotifyURL>/?trxid=1651556156&pid=25732&paymentid=25732 8&status=fail&errorcode=5&reason=No+Credit

Parameter description:

Parameters	Explanation
trxid	The transaction ID (which also is the memberid)
pid	Unique payment identifier.
paymentid	Non-unique payment batch identifier. While the *pid* represents a portion of the total amount for which you issued a billing request, these portions are grouped together by the *paymentid*, which is always the same as the first *pid* of the payment batch. An example would be: You bill the customer for an *amount* of 150. The WAP Billing API then checks the tariffs for that specific operator and breaks the payment into 100 and 50. If the *pid*'s would be 12345 and 12346 or these respective payments, the *paymentid* would be 12345 for both. In your own system and administration you can opt to log all the *pid*'s or simply the *paymentid*, but do note that every *pid* has a status and it is in theory possible that 1 payment will succeed and 1 payment will fail.
errorcode	This the errorcode. This could be used instead of "reason".
reason	A more detailed reason (WRxxx code), conform the list of resultcodes in section 8.1.1 Uni error codes:
amount	The amount for which this particular payment part was billed.
piece	The number of this particular payment in this "paymentid" batch.
totalpieces	The total number of payments in this "paymentid" batch.

Please respond with the plain-text: *45000*

7 Interaction with end-user

There are two kinds of flows. The difference between them is limited to deciding whether your template should be sent to the TargetPay staging area or to Payinfo for inspection. When the API is requested without the "finalize" option, the API will push the layout to the TargetPay staging area. This area will be accessible by the link which is provided in the API response. When using “kit.html” instead of the API, the system will redirect you to the staging area automatically.

When the "finalize" option is used, the system will show an error (see the last chapter) or a “00000 OK”* which means that the request is successfully processed and is queued to be approved by Payinfo.

Warning:

Payinfo provides an option to delete a style from their system. Unfortunately, TargetPay will not receive any notification from Payinfo when this happens. To keep the administrations in sync, it is advised to perform all layout changes through this procedure.

* A token will be attached when “showpid” is set to “1”, or kit.html is used.

7.1 Kit

When kit.html is used (https://www.targetpay.com/wap/loket/kit.html), the following information is required:

Value	Description
RTAFF	TargetPay customer ID.
RTLO	Layout on which the MCB service is created.
Password	Password of the account.
Service	The MCB service ID to request a layout exception for.
ColorBackGround*	The background color of on the order screen. Example: #FFFFFF
ColorButtons*	The color of the buttons on the order screen. Example: #742B88
ColorButtonFont*	The color of the font inside the buttons. Example: #FFF500
ColorFont*	The color of the font on the rest of the page Example: #000000
ColorLines*	The color of the lines on the order screen. Example: #000000
Logo**	Select an image from your local PC. When no image is selected, the default TargetPay logo will be used.
Finalize	When selected, the request will be pushed to Payinfo for acceptance. Otherwise, it is pushed to the TargetPay staging area.

* Colors are accepted if they meet the WCAG score of 7.1 or more.
A good test page is available at: http://leaverou.github.io/contrast-ratio/

** The only allowed types are png, jpg and gif.

7.2 Uploading via the API (automatically)

The API is located on:

https://www.targetpay.com/wap/loket/contentproviders

Type of the requests: POST or GET.

Param	Type	Description
rtaff	int	TargetPay customer ID.
rtlo	int	Layout on which the MCB service is created.
password	varchar	Password of the used account.
service	int	The MCB service ID to request a layout exception for.
ColorBackGround	hexadecimal	The background color of on the order screen represented in a hexadecimal. Example: #FFFFFF
ColorButtons	hexadecimal	The color of the buttons on the order screen represented in a hexadecimal. Example: #742B88
ColorButtonsFont	hexadecimal	The color of the font inside the buttons represented in a hexadecimal. Example: #FFF500
ColorFont	hexadecimal	The color of the font on the rest of the page represented in a hexadecimal. Example: #000000
ColorLines	hexadecimal	The color of the lines on the order screen represented in a hexadecimal. Example: #000000
Logo*	File	Logo for the header of the order screen.
Finalize	int	When set with value 1, the request will be pushed to Payinfo. Otherwise, it is pushed to the TargetPay staging area.
showpid	Int	If showpid and finalize are both set to “1”, a reference token is shown after the “OK” Example: 00000 OK|5ad79e4e-f84f-4786-a645-a9694e4d6db1 This is a reference token used as an identifier of `Het loket` which correlates to your MCB service. ¨Het Loket¨ can use this identifier when communicating with you concerning your template or MCB service.

*The only allowed types are png, jpg and gif.

7.3 Responses of the template API

Code	Message	Solutaion
001	Invalid rtaff.	The entered rtaff isn't a valid integer. Allowed numbers 0-9
004	Invalid service ID.	The entered service id isn't a valid integer. Allowed numbers 0-9
009	Logo is not PNG, GIF or JPG. Supplied MIME: <mime type if found>	Please check if you upload a valid png, jpg or gif-file
010	Incorrect RTLO.	The entered rtlo isn't a valid integer. Allowed numbers 0-9
011	Empty password.	Please enter a password.
012	Incorrect ColorBackGround.	Incorrect hexadecimal. Correct format: #FFFFFF Shebang followed by 6 numbers and/or characters.
013	Incorrect ColorButtons.	Incorrect hexadecimal. Correct format: #FFFFFF Shebang followed by 6 numbers and/or characters.
014	Incorrect ColorButtonFont.	Incorrect hexadecimal. Correct format: #FFFFFF Shebang followed by 6 numbers and/or characters.
015	Incorrect ColorFont.	Incorrect hexadecimal. Correct format: #FFFFFF Shebang followed by 6 numbers and/or characters.
016	Incorrect ColorLines.	Incorrect hexadecimal. Correct format: #FFFFFF Shebang followed by 6 numbers and/or characters.
021	Not able to communicate with Payinfo.	The system was not able to communicate with Payinfo. Please try again a few moments later. If this issue still occurs, please contact techsupport.
026	Content of the request response state from Payinfo not found.	This error should not occur.
027	Identification by the request failed. Please try again later.	Payinfo was not able to connect the request to a layout-exception. Please try again later. If this issue still occurs, please contact techsupport.
028	Payinfo is temporarily unavailable. Please try again later	The system was not able to reach Payinfo at all, please try again later. If this issue still occurs, please contact techsupport.
029	Request response state from Payinfo not found.	Payinfo didn't respond as excepted. Missing the status result from the request. Please try again later. If this issue still occurs, please contact techsupport.
030	WCAG-score below norm, see comments. | <comment from Payinfo>	The entered colors are not validated at Payinfo. See the message behind the pipe. Colors are accepted if they meet the wcag score of 7.1 or more. A good test page is available at: http://leaverou.github.io/contrast-ratio/ If you have any questions about "why a combination isn't valid" or have some doubts about the color scheme acceptance, please contact payinfo.
031	Rejected, but colors within norm.	Payinfo rejected the request, but the colors were correct. No further information was given how to solve.
032	Unknown state received from Payinfo.	Payinfo responded with a new / unknow state. Please try again later. If this issue still occurs, please contact techsupport.
034	Template not found at	It's look like you've removed the style on payinfo.mobi

Code	Message	Solutaion
	Payinfo. Did you removed it?	but our system will not receive any notification about that. Please contact techsupport in order to remove your exception from our database.
035	Payinfo FileMaker Error; Payinfo is temporarily unavailable. Please try again later.	Payinfo's FileMaker isn't available. Please try again later. If this issue still occurs, please contact techsupport.
X99	Undefined error from Payinfo	Undefined at all.
	Whoops Internal Database Error.	It seems there was a database issue on our side. Please try again later. If this issue still occurs, please contact techsupport.
00000	OK*	The request is valid and processed. Processed doesn't mean approved, only that the flow was correctly executed.

* A token will be attached when “showpid” is set to “1”, or kit.html is used.

8 Request payment or sign up status (pull)

When the end-user completed the payment or signup, he will be redirected to the *return URL*. The transaction ID and a unique user id will be appended to the return URL.

Example:
http://www.mysite.nl/returnurl.php?trxid=1651556156&unique=23ASEF342

At this return URL and using this trxid you can check the status of the transaction. The unique could be used for own purposes. Maybe you have a system which is already build but this system needs a variable like phone number, than you can use this number.

8.1 checktransaction

Example how to check a status of a payment:

Make a HTTP GET or POST call to:
https://www.targetpay.com/wap/checktransaction

Parameter description:

Parameter	Description
rtlo	The layoutcode/subaccount
trxid	The transaction ID (see section 3)
once	Set this parameter to 1 when you only want to receive an OK status the first time you check, and thereafter a: *WR999 Already checked* status. When you set once to 0, then TargetPay will return OK no matter how many times you asked for the status.

Example:
https://www.targetpay.com/wap/checktransaction?rtlo=12345&trxid=1651556156&once=1

The response will be a 5-digit response code followed by a space and a textual representation of the status. When the transaction was completed successfully, the response code is:

00000 OK

This means that the user has successfully signed up for the subscription service or has completed the one-off payment.

8.1.1 Uni error codes:

Errorcode	Description
1	WR004 Technical fault operator
2	WR007 User unknown
3	WR004 Technical fault TargetPay
4	WR008 User blocked/blacklisted/not active
5	WR005 No credit
6	Operator busy
7	User cannot be charged
9	WR001 Transaction not processed yet, try again later
10	WR002 Transaction cancelled by user
11	WR003 Transaction expired (not paid in 1 hour)
12	WR020 No layoutcode
13	WR021 No transaction ID
14	WR022 No transaction with this ID
15	WR023 Layoutcode doesn't match transaction
16	WR024 Still processing. Partial:
17	WR025 No paymentid
18	WR026 No payments with this paymentid
19	WR027 No transaction associated with this / these payment(s)
20	Unknown or new error from operator
21	WR999 Already checked
22	Subscription closed via TargetMedia helpdesk
23	Subscription closed via TargetPay.mobi
24	Subscription closed via MijnMobieleBetalingen.mobi
25	Subscription closed via affiliate
26	Fallback to SMS module detected, but blocked by service
27	Fallback to SMS module
28	Subscription limit reached
29	Not all the payments were successful
30	Subscription closed by barring
31	Subscription closed because of duplicates
32	Payment blocked due to overcharge
33	Transaction rejected by SMS Gateway
34	MVNO user not billable
35	On blacklist
36	Payment filed by royal decree
37	Subscription closed via SMS procedure

Uni error 16 (WR024 Still processing. Partial) is of particular interest, as it indicates that the user has made a partial payment of the total product cost. This can be due to a variety of causes, like the user having enough money for the first portion of the payment but not the remaining. It can also mean that the payment is delayed, or that the unpaid portion was a failed payment. This status code allows you to decide for yourself what you want to do with the customer and his partial payment.

Note that the meaning of each error code (first 5 digits) will remain the same, but the texts after that may be subject to change.

Also note that the expiration of transactions is set to happen after 1 hour of inactivity on the transaction in the case of "real" content-billing. In case of SMS-billing, this expiree is increased to 72 hours to account for delays with status data in the SMS networks.

8.2 Check payment

Aside from using checktransaction.php5 to retrieve the status of a transaction, it is also possible to pull the payment status from another script. This is mainly intended to validate weather a follow-up payment has succeeded or not. Make a HTTP GET or POST call to:

http://www.targetpay.com/wap/checkpayment

Parameter description:

Parameters	Description
rtlo	The layoutcode/subaccount
paymentid	The payment batch identifier that is returned when a followup payment has been created (See paragraph 6 Submit follow-up payments).

Example:
https://www.targetpay.com/wap/checkpayment?rtlo=12345&paymentid=123713

The response will be in the following format:
<#successful>|<#still open>|<#failed>/<#total>

Example:
2|0|0/2

This means that the original payment was broken in 2 (this happens depending on payment tariffs per operator) and both payments were successful.

Result codes:

Result code	Description
WR020	No layoutcode
WR023	Layoutcode doesn't match transaction
WR025	No payment ID
WR026	No payments with this payment ID
WR027	No transaction associated with this / these payment(s)

9 Using notifications (push)

Instead of requesting the status for the transaction you had to specify a notification URL when the transaction starts. When the status of the transaction changes we'll call this URL to notify you of this change. We will call the notify URL, for example:
http://<notifyURL>/?trxid=1651556156&status=fail&reason=WR002+Transaction+cancelled+by+user

http://<notifyURL>/?trxid=1651556156&status=confirmed&reason=00000+OK

Parameter description:

Parameter	Description
trxid	Transaction number which is received at with the request to start.php5
status	Can be one of the following: **confirmed;** Subscription is active. Follow-up payments are allowed. **processing;** One time payment is now being processed. **closed;** The one time payment was successfully completed, or the subscription was closed. **fail;** The transaction failed; for example, there was no credit. **open;** The user has confirmed their payment or subscription via the TargetPay.mobi order screen. **Start;** This value is deprecated since 2.3
errorcode*	Uniform error code. See 8.1.1 Uni error codes:
reason	A more detailed reason (WRxxx code), conform the list of result codes in section 5.
unique	This parameter is deprecated since 2.3
msisdn	The MSISDN of the customer is sent from the open state; **Should be configured in the service.**
asid	Anonymous Subscriber Id (ASID); **Should be configured in the service.**
operator	The operator code; **Should be configured in the service.** For all available operators, see https://www.targetpay.com/operators

* Is not available when the transaction notification is “confirmed”, “processing” or “open”
Please respond with and only with the text: *45000*
If not, 3 more attempts will be made to call the notify URL. 1 attempt every minute.

Take note that the notification URL will never be called with the partial payment status, which is only available by “pull”.

10 Unsubscribe

The end user has several ways to unsubscribe from the service.

- Payinfo.nl
- Unsubscription page in your service
- TargetMedia helpdesk
- Your own customer care center via API call
- operator

10.1 un-subscription page

You may add an un-subscription page to your mobile website.

Example:
http://targetpay.mobi/unsubscribe.php5?trxid=*1651556156*

10.2 via API call

You can setup your own customer care center and build your own unsubscribe screen via this API.

Example:
http://www.targetpay.com/wap/checkout?
trxid=1651556156&service=25732&rtaff=257328&ok=true

Parameter description:

Parameter	Description
trxid	The transaction ID (which also is the memberid).
service	The ID of your service (for security).
rtaff	The account number of rtlo owner.
ok	If you want to test if the checkout should be done.

Possible responses:

Result code	Description
45000	The unsubscribe action was successful.
WB601	Transaction does not exists or trxid missing.
WB602	Transaction already checked out.
WB603	This is not a subscription
WB604	State is closed, but missing close date...
WB605	This transaction does not belong to the given service number
WB606	This transaction does not belong to the given rtaff
WB607	Failure with checkout.

11 Register Service API

The Register Service API is intended for TargetPay resellers and high volume users to create, copy and change Mobile Content Billing services on the TargetPay platform.

The Register Service API provides a similar functionality as the TargetPay user interface to create new services, with extended customization to accommodate resellers that maintain services for their clients.

As the Code of Conduct for Mobile Internet Services has significantly changed per March, 1, 2012 we have updated the API significantly.

Most noticeable, the procedure for applying for T-Mobile approval has been removed, since T-Mobile approves services by default as of march 1, 2012.

11.1 API specification

This document details all the available parameters of the MCB Register Service API. Access to the API is restricted by default, and must be obtained via mcb@targetpay.com.

The API operates in two modes: **create** and **change**, to enable creating a new service or change an existing service respectively.

Some parameters can only be set upon creation and cannot be changed once the service is created. These parameters are marked as **immutable**. Trying to change these parameters will result in an error. Although these parameters cannot be changed they might still be required for verification purposes.

Parameters marked required are mandatory, the service will not complete if the parameter is not included or incorrectly specified.

Parameters marked **optional** are not required and can be left out, generally a default value will be assumed.

Please take note that any usage of these restricted parameters will be reviewed by TargetMedia for consistency with policies and guidelines.

The Register Service API is located here:
http://www.targetpay.com/wap/register_service

Parameters can be sent via either GET or POST (POST is always advised). Please ensure the correct URL encoding of your parameters (note that some libraries take care of URL encoding, particularly when using POST), as they will be decoded by the API upon acceptance of your request before validation of said parameters.

If the service completes without error, the response will be:
(ServiceId)|000000 OK

For example:
3399|000000 OK

Parameter description:

Parameter	Required	Type	Description
rtaff	y	int	Your account number is required for login and verification.
passwd	y	varchar	Your password is required for login and verification.
Rtlo	y	int	Your subaccount number is required to verify or set the account under which the service will be administered. For new services it must be an existing subaccount, for existing services it needs to be the original subaccount. The sub account cannot be changed by changing this parameter.
Serviceid	n	int	The presence of this parameter determines the mode in which the API operates. • If this parameter is not present you will be **creating** a new service. • If this parameter is present you will be able to **change** parameters for the specified existing service Note that all required parameters will have to be specified, even if they do not change.
Amount	y	int	Specifies the amount in eurocents that will be charged to end users using this service. This can be either the recurring amount in case of subscription services or the amount in case of a single payment service.
Frequency	y	Day,week, month or nonsubscription	This indicates the recurring frequency of the payment. In case of **nonsubscription** only one single payment will be made.
Category	y	Integer;	This indicates the category of items that will be sold. One of the following integers for the corresponding text: 1. Item(s) 2. Credit(s) 3. Onbeperkte toegang 4. Onbeperkte downloads 5. Quiz 6. Onbeperkte toegang tot alle escort profielen
Quantity	n	int	This indicates the number of items offered for a single payment. Defaults to 1
Description	y	varchar	The description of the service, generally the name of the service. Please refer to the code of conduct for valid values.
contentprovider	n	varchar	Default, the name of the content provider is

Parameter	Required	Type	Description
			copied from the affiliate company name as set in your TargetPay account. Note that using a different content provider name implies that the content provider must comply with the Code of Conduct.
Tarms	N	Varchar (only via POST-requests)	This will enable you to supply your own terms and agreement text. If this parameter is not present the TargetPay default text will be shown to the end user.
Homeurl	y	varchar	Link to the home page where the service is offered.
Cancelurl	y	varchar	Link to the home page where the user will navigate to when the user clicks a cancel button.
Notifyurl	n	varchar	Web address of the script where notifications will be sent to. Make sure that notifications are properly processed at this web address. If omitted, no notifications will be sent.
Returnurl	y	varchar	Web address of the page where the user will navigate to if the transaction has been completed.
Smsmethod	y	Cross or return	**Retour** will disable Premium SMS fallback all together and will direct the user to the returnurl. In this case a parameter **noContentBilling=1** will be added to the query string. **Cross**
unknownbysms	n		If set to true, users that cannot be identified by the operator (ie. If surfing by wifi) will be identified by passing their MSISDN. A confirmation SMS message will be sent to the end user, containing a link to the order screen. If the user activates the link, the user identification is verified, thus enabling the user to be charged with Mobile Content Billing. The SMS message will be charged to the affiliate. Note that this mechanism will not work for KPN due to their technical restrictions.
Helpdeskemail	n	varchar	This value specifies the helpdesk e-mail address. The default will be the helpdesk e-mail address that is specified in your TargetPay account

Parameter	Required	Type	Description
Helpdeskurl	n	varchar	This value specifies the helpdesk url. The default will be the helpdesk url that is specified in your TargetPay account
helpdeskphone	n	varchar	This parameter specifies a your helpdesk phone number. The default will be the helpdesk phone number that is specified in your TargetPay account
callbackmsisdn	n	Boolean (default **false**)	If set to **true**, the msisdn of the user will be included in the calls to notifyurl as a query string parameter named msisdn. I set to **false** the user msisdn will not be included in the callback.
callbackoperatorhex	n	Boolean (default **false**)	If set to **true**, the operator of the user will be included in the calls to notifyurl as a query string parameter named operator. The hex code of the operator will be sent as the value. I set to **false** the user operator will not be included in the callback.
Callbackasid	n	Boolean (default **false**)	If set to **true**, the ASID of the user will be included in the calls to notifyurl as a query string parameter named asid. The ASID will be only sent in case that the operator is KPN. I set to **false** the user ASID will not be included in the callback.

11.2 Changes in the parameters

Deprecated parameters are:

- Duplicate
  Since the T-Mobile approval is no longer a limitation, the need to create duplicated immutable services with T-Mobile approval is obsolete.The duplication parameter -though still functional- will be no longer supported.
  We strongly recommend affiliates to administer their service parameters and copy from their own parameters when needed.
  In the rare case of exceptions we are fully able to support exceptions for all services for a particular rtaff or rtlo
- Ordertext (obsolete)
- Altordertext (obsolete)
- Paymentprovider (obsolete)
- Paymentproviderlink (obsolete)
- Maxsubscriptions (now default 1)
- Paymentfailtreshold (now set to default 8)
  (please request this by sending a mail to mcb@targetpay.com if required)
- Interfacedomain (requires manual configuration)
  (please request this by sending a mail to mcb@targetpay.com if required)

12 Light weight User Identification

The Lightweight User Identification API is intended for MCB customers that want to customize the flow of their consumers. It provides functionality to determine on what Telecom operator's mobile network a consumer is browsing and gives an indication of whether the consumer is billable by Mobile Content Billing or not.

Lightweight User Identification is done by sending the consumer to an API on TargetMedia's domain. We will then identify the consumer and send him or her back to a supplied URL, accompanied by two parameters that describe the operator code and the content-billing method.

This process can be initiated whenever a consumer is about to purchase a product, before the regular MCB transaction is made. The results of identification can be used by the customer to decide on a course of action, such as initiating a transaction or attempting to identify the consumer via the Web payment flow.

Please note that the Lightweight User Identification API does not return the MSISDN of the consumer. It is intended to help customers initiate an efficient Wifi-identification flow, should you wish to host the Wifi-identification page yourself. If you would like to know more about hosting your own Wifi-identification page, please consult the "Web payment" documentation.

12.1 API Parameters

The Lightweight User Identification API is located here:
http://www.targetpay.com/wap/detect_method_by_user

You may redirect a consumer to this API with the following (URL encoded) GET parameter:

Parameter	Required	type	Description
returnURL	y	varchar	The URL where the consumer should be returned to after identification has been done.
version	y	Default 1	

The end-user is redirected to this URL with two new GET parameters attached:

Parameter	type	Description
operator	varchar	The representation code of the mobile network that the consumer is browsing on. For a full list of these codes, please check https://www.targetpay.com/operators
method	varchar	Whether the consumer would be billable by content-billing or not. When this is 'cb', a customer should initiate a regular MCB transaction as no further identification is needed. When this is 'sms', a customer can decide if they want to initiate a Wifi-identification Web-payment flow.