

MAC calculation

The MAC is calculated always like this: `HmacSHA256("PayId*TransID*MerchantID*Amount*Currency", "YourHmacPassword")` where:

Key	Value	Comments
PayId	Referenced PayId	May be empty, e.g. for creating an initial payment process or risk management request; is used with subsequent requests like capture/refund.
TransId	Your transactionId to reference / identify your request	Your own reference to identify each request / payment process.
MerchantId	Your MerchantId	Your MerchantId assigned to you by Computop identifying this request.
Amount	Amount in smallest unit of currency, e.g. 123=1,23 (EUR)	Amount of this request; may be empty if not used, e.g. for status inquiries.
Currency	Currency of payment process in ISO 4217, e.g. EUR, USD, GBP	Currency of this request; may be empty if not used, e.g. for status inquiries.
YourHmac Password	Your HMAC-password assigned to you by Computop	Your HMAC-password assigned to a specific MID; if you have different MIDs you will have different HMAC passwords, too.

Notes:

- in case that a value is not present just leave it empty, e.g.:
 - with amount/currency, without PayId to initiate a new payment - like in this example: `HmacSHA256("**TID-4453732122167114558*yourMerchantId*1234*EUR", "mySecret")`
 - with amount/currency, without PayId, without TransId: `HmacSHA256("**yourMerchantId*1234*EUR", "mySecret")`
 - with PayId, without amount/currency: `HmacSHA256("fe3f002e19814eea8aa733ec4fdacafe*TID-4453732122167114558*yourMerchantId**", "mySecret")`
- you will find more details for HMAC-calculation
 - for requests: [HMAC Authentication \(Request\)](#)
 - for responses/notify: [HMAC Authentication \(Notify\)](#)
- you can find an application to verify for MAC-calculation here: <https://computop.com/de/developer/paygate-test>

Raw parameters before encryption

The raw parameters define basic settings for this payment call, e.g. your MerchantId, amount, currency, your reference and URLs for success, failure and notify:

Key-Value	Comments
MerchantID=yourMerchantId	Your MerchantId to identify your request at Computop Paygate
MsgVer=2.0	Indicate that 3-D Secure 2.x shall be used; Specially for 3-D Secure 2.x it is useful to provide additional data (like billing- and shipping-address) to improve frictionless processing (i.e.: payment is authenticated without challenge). These additional data are provided in JSON-structure .
TransID=TID-18724420542167170812	Your request identifier
RefNr=RG123-2021	Your payment process reference
Amount=1234	The desired amount in smallest currency unit, e.g. 1234 + EUR 12,34 EUR
Currency=EUR	and currency
URLSuccess, URLFailure, URLBack	URLs for forwarding the customer in case of success, failure, cancel
URLNotify	URL to receive Computop Paygate notifies
Response=encrypt	Computop Paygate shall respond with encrypted data
Language=en	Customer wants english language

Parameters before encryption

MerchantID=yourMerchantId&MsgVer=2.0&TransID=TID-18724420542167170812&RefNr=RG123-2023&Amount=1234&Currency=EUR&URLSuccess=https://www.yourshop.info/success.php&URLFailure=https://www.yourshop.info/failure.php&URLNotify=https://www.yourshop.info/notify.php&Response=encrypt&MAC=ca3c75eaf2120dfd15de77af2398b1561d8473f647b72aa7270fde94df7756d6&Language=en



As "=" and "&" are used for building key-value-pairs these characters **must not** be part of **any value**.

Do not send empty values, but only keys which are required and really having values.

For credit card processing with 3-D Secure 2.x (EMV 3DS) you must add "MsgVers=2.0"

Hosted Payment Page works like a proxy for the other payment forms (i.e. Credit Card Form (PaySSL), Direct Debit Form (PaySDD), paymethod specific forms (e.g. PayPal))

- so you have to add "MsgVers=2.0" to enable 3-D Secure 2.x for Credit Card Form (PaySSL)
- you may supply other key-values for other paymethods (e.g. PayPal)

Encrypt parameters into Data/Len (Blowfish)

The raw parameters are encrypted via Blowfish ECB and then hex-encoded. We provide you predefined functions in our toolkits for a quick start.

Notes

- The value for "Len" is the **string length** of the **unencrypted parameter** string built in the step before.
- Blowfish is standard encryption mode of Computop Paygate

To ease your integration we provide predefined functions to help you with Blowfish ECB:

Your language	Where to find
ASP	txmsCrypto.dll // txmsCrypto.BlowFish
ASP.NET	CompuTop.Core.Crypto.dll // CompuTop.Core.Crypto.BlowFish
Java	Blowfish.java
PHP	function.inc.php ctHMAC ctEncrypt ctDecrypt

Here a sample

Element	Value
MerchantID	yourMerchantId
Password	TestTestTestTest
Unencrypted request	MerchantID=yourMerchantId&MsgVer=2.0&TransID=TID-18724420542167170812&RefNr=RG123-2023&Amount=1234&Currency=EUR&URLSuccess=https://www.yourshop.info/success.php&URLFailure=https://www.yourshop.info/failure.php&URLNotify=https://www.yourshop.info/notify.php&Response=encrypt&MAC=ca3c75eaf2120dfd15de77af2398b1561d8473f647b72aa7270fde94df7756d6&Language=en
Len	354

Data	397fb1b3eadb19c4c4610422e3426cecb9e5f3c83ff04be4d78a63827839703847465e7118da27f8e186b7053923d5189c321d6b b04dbe1f147561184fc3c4c999861c69b79a94a1a44494219adaec1688765fa72282e04eca73b5725996acddfb874a615610df41c 4eb8ae12bc62eece8c44dc50726afcf4d249d8a4d5af7ee93f9ea95839bf6ffcaa94eaa70e46f002b5954c3bbe9ae2a69ee1b451cf8 d96387c09c4c7300ab95e5850df082d778a31f84e7c3722760d5f71927869a1ab3139154673d908a96ab2f5be4493b10112a4fa82 5b257310b79834027703224aa831f84e7c3722760d5f71927869a1ab314f78b3f489b9b6e165163bbdbb086ca49072acdfbb9fa31 7d847056fc38d6e0ec7a19685cfc7eaf733c965596e2a2df611686b5078cbf4f3f73338a8769334d88b674fa0ef1a03ffa518668a6f1 2e6fd0a4ab5dfb093354f551c52de3a75c0a113dd8837ba810ae8926051ed6edbfcc3c1aef6417699fb6
-------------	--

- you can find an application to verify for MAC-calculation and Blowfish-encryption here: <https://computop.com/de/developer/paygate-test>
- just select Blowfish/ECB

Encrypt parameters into Data/Len (AES)

The raw parameters are encrypted via AES/CBC/PKCS7 and then hex-encoded.

Notes

- AES 128 / AES 192 / AES 256 are supported and depend on password-length with 16, 24 or 32 Byte
- AES can be setup by [Computop Helpdesk](#) on request
- Libraries with AES encryption are in progress
- AES does not require LEN-parameter by technology, but it's still required for Computop Paygate-compatibility
- AES is used with Initialization Vector (IV) of 16 Bytes. IV is Bin2Hex-encoded and sent plain as part of Data: Data=concat (Bin2Hex (IV), "-", Bin2Hex(encryptedData))
- The Initialization Vector (IV) is created randomly with each encryption process, is sent in plain text (Bin2Hex-encoded) and required for decryption

To ease your integration here some links to an encryption with AES CBC:

Your language	Links
ASP	https://learn.microsoft.com/en-us/dotnet/api/system.security.cryptography.aesryptoserviceprovider.iv?view=net-6.0
Java	https://docs.oracle.com/javase/7/docs/api/javax/crypto/Cipher.html
PHP	https://www.php.net/manual/en/function.openssl-encrypt.php

Here a sample

Element	Value
MerchantID	yourMerchantId
Password	TestTestTestTest
Unencrypted request	MerchantID=yourMerchantId&MsgVer=2.0&TransID=TID-18724420542167170812&RefNr=RG123-2023&Amount=1234&Currency=EUR&URLSuccess= https://www.yourshop.info/success.php &URLFailure= https://www.yourshop.info/failure.php &URLNotify= https://www.yourshop.info/notify.php &Response=encrypt&MAC=ca3c75eaf2120dfd15de77af2398b1561d8473f647b72aa7270fde94df7756d6&Language=en
Len	354
Data	5b447021b775137d2a4249f271200071-634f5b121cf01f3bcc060f00827fb55affe364be050a36a9e97e9571ea1c2c150731300051cb2618f4e6809f72ecdf284e81ef465e0db2b808f5104f79d92cb4055dd60a5227cc82b82d240d502e51a60606c82c69bed317e117cddb110cdd2cc8a7712a18c398990cc47a532cdf8caef6a32f048b51576de4d2575c9cf61ce08c3f6a441ea8e553028b6b7f908ebe0c34e3ed064b17b32f0a2afc7873ec16d10eaa7780ec89c1c6b1a0d68508cf9dbb916e27adcb08aeb774880c61001e66a10c841f0742fca82882260c511ab2a9ee9252ac88c6833b265fb70bb0f99af65b15434d052106dee38d2262ab72973aa28940da8c99c35f57f6fe5c065005b1c1e726b60eaeaba8fd3ed32edfdcbae52af9c92c0ba7c1a7edf88888a1f9af9055a105803beacbf5627d1d4a95b6aceeb4f27b1fab6b47bfc8a2542b527b12514e80c959d038c177473363095b0aef333962c4d5054bc293fea6402fac1fda8fab81b6d3c8127a709e3bbe1dfed19ae
Note	 5b447021b775137d2a4249f271200071 is the Initialization Vector (IV) of 16 Bytes generated with AES encryption converted from Bin to Hex.

- you can find an application to verify for MAC-calculation and AES-encryption here, too: <https://computop.com/de/developer/paygate-test>
- just select AES/CBC (IV part of DATA)

Notice: Please note that if you want to switch to AES encryption as a merchant, all requests for all actions must also be encrypted with AES. Please coordinate the changeover with all parties in advance. Batch submission is excluded here as no API encryption is used. In this case, PGP is used for security.

Now putting all together – the API request

The API request is then built like:

Value	Comments
Basic parameters	
e.g. paymentPage.aspx	Selected desired endpoint of Computop Paygate
MerchantID=yourMerchantId	Your MerchantId to identify your request at Computop Paygate (here additionally as plain value)
Len=<Len>&Data=<Data>	Length of the uncripted parameter string and Data returned by Encryption
Template parameter (plain)	
Template=PaymentPageDropDown_v1	Template for Hosted Payment Page (HPP)
CCTemplate=Cards_v1	Template for Credit Card Form (called by HPP)
SDDTemplate=DirectDebit_v1	Template for Direct Debit Form (called by HPP)
Language=en	Starting language for the customer
CustomField1...16	Some CustomField-Data to display additional information on the HPP depending on the template



Putting API-call together

The API-call consists of:

Category	Description
Computop Paygate endpoint	e.g. https://www.computop-paygate.com/paymentPage.aspx
MerchantID =<>	MerchantID=YourMerchantID
Len & Data	Encrypted data containing request data
additional params	Additional key /values in plain, not encrypted

Building the API-call

```
https://www.computop-paygate.com/paymentPage.aspx?
MerchantID=yourMerchantId&Data=397fb1b3eadb19c4c4610422e3426cecbc9e5f3c83ff
04be4d78a63827839703847465e7118da27f8e186b7053923d5189c321d6bb04dbe1f147561
184fc3c4c999861c69b79a94a1a44494219adaec1688765fa72282e04eca73b5725996acddf
b874a615610df41c4eb8ae12bc62eece8c44dc50726afc4d249d8a4d5af7ee93f9ea95839b
f6ffcaa94eaa70e46f002b5954c3bbe9ae2a69ee1b451cf8d96387c09c4c7300ab95e5850df
082d778a31f84e7c3722760d5f71927869a1ab3139154673d908a96ab2f5be4493b10112a4f
a825b257310b79834027703224aa831f84e7c3722760d5f71927869a1ab314f78b3f489b9b6
e165163bbdbb086ca49072acdffb9fa317d847056fc38d6e0ec7a19685cfc7eaf733c965596
e2a2df611686b5078cbf4f3f73338a8769334d88b674fa0ef1a03ffa518668a6f12e6fd0a4a
b5fdffb093354f551c52de3a75c0a113dd8837ba810ae8926051ed6edbfcc3c1aef6417699fb
6&Len=354&Template=PaymentPageDropDown_v1&Language=en&CCTemplate=Cards_v1&S
DDTemplate=DirectDebit_v1&URLBack=https%3A%2F%2Fwww.yourshop.info%
2F&CustomField1=12%2C34+EUR&CustomField2=Order+Text&CustomField3=https%3A%
2F%2Fwww.paytest.info%2Fphantasy-logo.
png&CustomField4=Shopping+Cart&CustomField5=Company+Name&CustomField6=First
+Name&CustomField7=Strat%3C%
9Fe+4&CustomField8=12345+City&CustomField9=Shipping+Company&CustomField10=S
hipping+Name&CustomField11=Shipping+Street&CustomField12=23456+Shipping+Cit
y&CustomField13=RG-Inv+123%
2F2021&CustomField14=Some+Label&CustomField15=Some+Text&PayType=0
```

This URL doesn't work and will result in "Unexpected exception", because MerchantId "yourMerchantId" doesn't exist. You will find some working URLs below.

Sending the API request

A request can be sent either via GET or POST. We recommend to use POST for two reasons:

- with GET the parameter length is restricted to 2048 bytes depending on the browser, while with POST the request length is limited to 5120 bytes; If you require longer strings please contact [Computop Helpdesk](#)
- via GET the parameters are attached to the URL which can be easily manipulated by a customer - therefore Analytics prevents manipulation using Blowfish/AES encryption

Some tips on Paygate responses

Checking the Paygate response

Server-2-server response

With server-2-server requests a request will respond with a direct response containing

- a status indicating success or failure of transaction
- a code (response code) explaining details of transaction along with a description
- other data like PayId for each payment process
- and other data depending on the type transaction

Payment page / asynchronous notification

In case of a redirect payment an asynchronous notification is sent to your system indicated by a URLnotification.

The response can be either encrypted or in plain text - we recommend an encrypted response.

Common

Please check:

- just checking whether "URLFailed" or "URLSuccess" has been called is not sufficient and can easily misused
 - [Response code](#): only "code=00000000" indicates a successful and completed action
- [HMAC](#) in Analytics response is valid - to ensure that the message is not manipulated



Some tips on responses

Result	Description
Unexpected exception	Paygate response not containing "code" and "status" a very likely reason is that you simply used • a wrong template name or wrong MerchantId. See more details in our documentation for Payment Pages
• PayId=0000....0000 • code=8 digit	Paygate detected some error in request and a payment process has not been created. <i>These payment processes can not be found in Analytics.</i> An overview of response codes can be found here: Response codes
• valid PayId • code=8 digit	A payment process with PayId has been created but the subsequent systems detected an error. An overview of response codes can be found here: Response codes
• code=xxxx0005	The amount is the first parameter which is checked. The amount must be given in numbers without decimals. But you may also simply have mixed up your MIDs and Encryption-keys - just doublecheck.
• valid PayId • code=00000000	Payment / process successful and completed
• valid PayId • code=0	Payment / process pending

A few URL calls to play with

Please find some test data to play here: [Test Guide](#). However, the payments may result in error to prevent abuse.

Click and try	Comments	Notes
Click and try	Link to Hosted Payment Page without specific template data to initiate a payment process for 12,34 EUR	no template data specified

Click and try	The same data (Len + Data) can be used with Hosted Payment Page using a different template for Hosted Payment Page itself and with specific templates for selected credit card payments and direct debit payments	As we start "Hosted Payment Page" the template refers to a HPP-template and we add template names for subsequent payment forms for credit card and direct debit payments: Template=PaymentPageDropDown_v1&Language=en&CCTemplate=Cards_v1&SDDTemplate=DirectDebit_v1 We also add some CustomFields to display some additional customer information, just by changing "CustomField3" you can refer to your own logo.  Only available if the "CustomFields" are supported by the templates)
Click and try	The same data (Len + Data) can be used with Credit Card Form (PaySSL)	As we start "PaySSL" the initial template name refers to a specific credit card template named "Cards_v1": Template=Cards_v1&Language=en Just by changing "Template=ct_responsive" you can use a different payment form design