Wirecard CEE Integration Documentation

() -

Created: 2020-02-25 18:48

Reading Stored Payment Data from Wirecard Data Storage

Prerequisites

- Initialization of Wirecard data storage
- Storing sensitive payment data in Wirecard data storage

Time of reading

After initializing the data storage you are able to read at least the `storageId` for your current checkout of your consumer. When storing sensitive payment data in the data storage you are able to read it with the below described methods.

Please be aware that your data storage session for a specific consumer is only valid for 30 minutes after the latest read or write access from your side. After this time you have to initialize the data storage again and you have also to store the sensitive payment data of your consumer again.

Each store or read operation extends the validity of the data storage for again 30 minutes.

Using the read data storage operation

You can use the read operation for:

- Testing if the session with that `storageId` is still valid or already invalidated,
- Extending the session lifetime for another 30 minutes and
- Reading sensitive payment data you stored to this data storage.

Reading data of the Wirecard data storage

The read operation for the Wirecard data storage is nearly the same as for the initialization of the data storage:

You send a server-to-server request from your web server to the Wirecard Checkout Server to a specific URL containing some specific request parameters.

The URL for the serve-to-server read operation is:

```
https://checkout.wirecard.com/seamless/dataStorage/read
```

Please be aware that it is sometimes necessary to enable server-to-server requests within the configuration of your web server. This issue arises typically on provider managed web servers with PHP.

For a proper request you have to set a correct HTTP header. Therefore you need to set the HTTP header elements within your request as described within the Initialization of Wirecard data storage.

Computing the fingerprint

The fingerprint is computed by concatenating all request parameters without any dividers in between and using the secret as cryptographic key for the hashing function. If you do not use the optional parameter `shopId` you have to omit it in your fingerprint string.

Please be aware that the concatenation of the request parameters has to be done in the following order:

1. customerId
2. shopId
3. storageId

After concatenating all values to a single string create an HMAC-SHA-512 hash with your secret as cryptographic key. The result is the fingerprint which you add as a request parameter to the server-to-server call.

The Wirecard Checkout Server is thus able to check whether the received parameters are manipulated by a 3rd party. Therefore it is essential to keep your secret safe!

Required request parameters

To start the read operation you have to set all required parameters to their corresponding values. If one or more of these parameters are missing you will get an error message.

Parameter	**Data type**	**Short description**
customerId	Alphanumeric with a fixed length of 7.	Unique ID of merchant.
storageId	Alphanumeric with a fixed length of 32.	Unique ID of data storage.
requestFingerprint	Alphanumeric with a fixed length of 128.	Computed fingerprint of the parameter values and the secret.

Optional request parameters

Parameter	Data type	Short description
shopId	Alphanumeric with a variable length of 16.	Unique ID of your online shop.

Format of return values

After you send the data storage read request as a server-to-server request from your web server to the Wirecard Checkout Server you will get the result of the read operation as key-value pairs returned in the content of the response.

Returned response parameters

The following parameters are always returned when querying the data storage.

Parameter	Data type	Description
storageId	Alphanumeric with a fixed length of 32.	Unique ID of data storage.
paymentInformations	Numeric	Number of stored payment methods.
paymentInformation.{n}.paymentType	Alphabetic	Name of payment method.

Returned payment method specific parameters

The following parameters are dependent on the payment method your consumer chose. Please visit Integration of specific payment methods for further information.

Parameter	Data type	Description
paymentInformation.{n}.anonymousPan	Numeric with a fixed length of 4.	Anonymized credit card number containing only the rightmost 4 digits.
paymentInformation.{n}.maskedPan	Numeric with special characters and a variable length of 13 to 19.	masked credit card number: first 6 numbers followed by * and he last 4 numbers of the credit card.
paymentInformation.{n}.financialInstitution	Enumeration	Financial institution.
paymentInformation.{n}.brand	Enumeration	Brand of Credit Card.
paymentInformation.{n}.cardholdername	Alphanumeric with special characters.	Name of card holder.
paymentInformation.{n}.expiry	Numeric with special characters.	Expiry date of credit card in format MM/YYYY.
paymentInformation.{n}.accountOwner	Alphanumeric with special characters.	Name of owner of account.
paymentInformation.{n}.bankName	Alphanumeric with special characters.	Name of bank.

paymentInformation.{n}.bankCountry	Alphabetic with a fixed length of 2.	Country code of bank.
paymentInformation.{n}.bankAccount	Alphanumeric with a variable length of 1 to 11.	Account number.
paymentInformation.{n}.bankNumber	Numeric with a variable length of 1 to 8.	Bank number.
paymentInformation.{n}.bankBic	Alphanumeric with a variable length of 1 to 255.	BIC of bank.
paymentInformation.{n}.bankAccountIban	Alphanumeric with a variable length of 1 to 255.	IBAN of account.
paymentInformation.{n}.payerPayboxNumber	Numeric with minimum length of 8.	Number of paybox account starting with 0.

Returned optional parameters

These optional parameters enhance the functionality and usability of the payment process regarding specific features and functions. To enable one or more of these parameters please contact our support teams.

Parameter	Data type	Description
paymentInformation.{n}.hashedPan	Alphanumeric with a fixed length of 128 (hash mechanism HMAC-SHA-512).	Hashed credit card number. Only if payment was successful. Please visit Credit Card for further information.

Please be aware that due to PCI DSS compliance, `hashedPan` cannot be returned neither with `maskedPan` nor `anonymousPan`.

Error cases

If the read operation did not succeed you will get parameters describing the error:

Parameter	Data type	Short description
errors	Numeric	Number of errors occurred.
error.{n}.errorCode	Numeric with a fixed length of 5.	Numeric error code which you should log for later use.
error.{n}.message	Alphanumeric with special characters.	Error message in English.
error.{n}.consumerMessage	Alphanumeric with special characters.	Error message in localized language for your consumer.

For example a possible error would look like:

```
error.1.errorCode=11500&error.1.message=CUSTOMERID+is+missing.&error.2.error
Code=11506&error.2.message=REQUESTFINGERPRINT+is+missing.&errors=2
```