Creating and Using Security Keys

User Guide

cybersource
A Visa Solution

Restricted Rights Legends

Trademarks

Version: 24.01

Contents

Recent Revisions to This Document

24.01

Added additional support for creating passwords for REST and Simple Order API keys. See Creating a REST API Key (on page 9) and Creating a Simple Order API Key (on page 11).

23.02

Added a paragraph to the REST API keys section (on page 9) stating that when you sign up for a Sandbox account in the Developer Center, your confirmation email contains a key and a shared secret key for HTTP Signature authentication.

23.01

Added Regenerating a Meta Key (on page 28).

22.01

Expanded the Meta Keys (on page 17) section.

21.02

Added the following:

- Meta Keys (on page 17)
- Searching for Keys and Filtering Results (on page 29)
- Deactivating Keys (on page 31)
- Deleting Keys (on page 30)

21.01

Procedures were rewritten to reflect the more streamlined flow of the Cybersource Business Center Key Management page.

About This Guide

This section describes the structure and content of this guide.

Audience and Purpose

This guide is written for application developers who want to use Cybersource services that require a security key, including API requests.

Customer Support

For support information about any service, visit the Support Center: http://support.cybersource.com

VISA Platform Connect: Specifications and Conditions for Resellers/Partners

The following are specifications and conditions that apply to a Reseller/Partner enabling its merchants through Cybersource for Visa Platform Connect ("VPC") processing. Failure to meet any of the specifications and conditions below is subject to the liability provisions and indemnification obligations under Reseller/Partner's contract with Visa/Cybersource.

1. Before boarding merchants for payment processing on a VPC acquirer's connection, Reseller/Partner and the VPC acquirer must have a contract or other legal agreement that permits Reseller/Partner to enable its merchants to process payments with the acquirer through the dedicated VPC connection and/or traditional connection with such VPC acquirer.

2. Reseller/Partner is responsible for boarding and enabling its merchants in accordance with the terms of the contract or other legal agreement with the relevant VPC acquirer.

3. Reseller/Partner acknowledges and agrees that all considerations and fees associated with chargebacks, interchange downgrades, settlement issues, funding delays, and other processing related activities are strictly between Reseller and the relevant VPC acquirer.

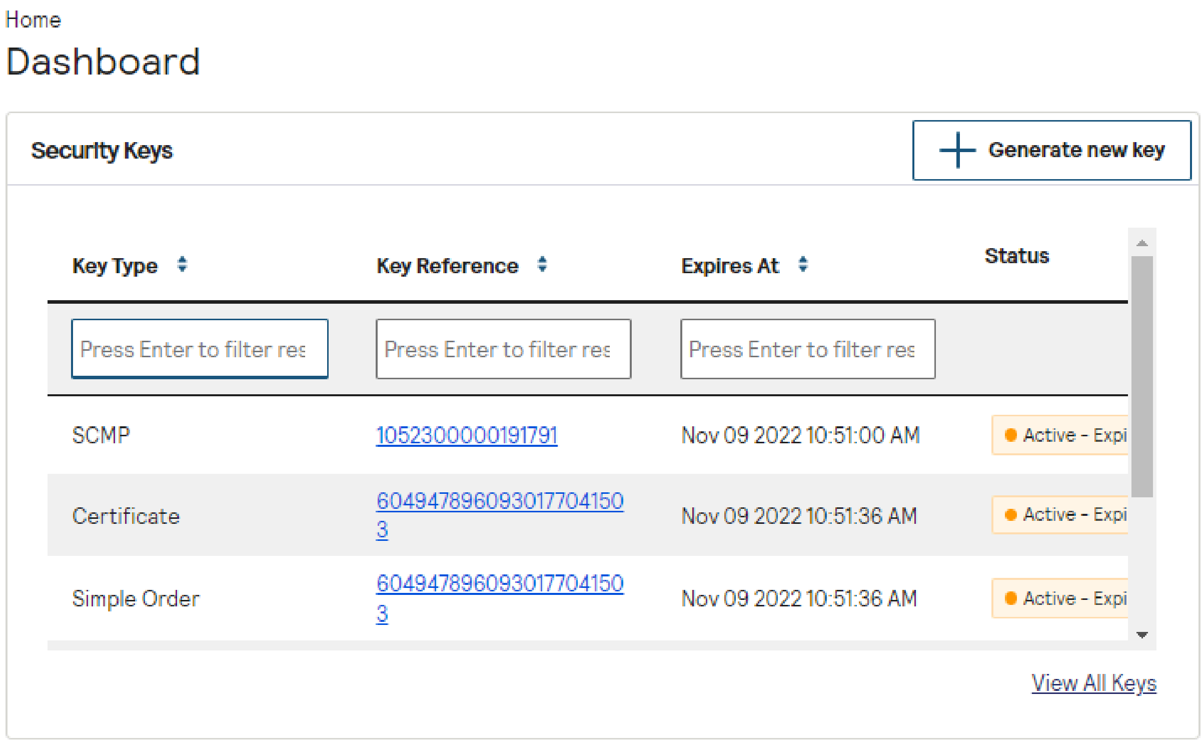
4. Reseller/Partner acknowledges and agrees that the relevant VPC acquirer is responsible for payment processing issues, including but not limited to, transaction declines by network/issuer, decline rates, and interchange qualification, as may be agreed to or outlined in the contract or other legal agreement between Reseller/Partner and such VPC acquirer.

DISCLAIMER: NEITHER VISA NOR CYBERSOURCE WILL BE RESPONSIBLE OR LIABLE FOR ANY ERRORS OR OMISSIONS BY THE VISA PLATFORM CONNECT ACQUIRER IN PROCESSING TRANSACTIONS. NEITHER VISA NOR CYBERSOURCE WILL BE RESPONSIBLE OR LIABLE FOR RESELLER/PARTNER BOARDING MERCHANTS OR ENABLING MERCHANT PROCESSING IN VIOLATION OF THE TERMS AND CONDITIONS IMPOSED BY THE RELEVANT VISA PLATFORM CONNECT ACQUIRER.

Using the Dashboard

When you log in to the Business Center, the dashboard appears. You can use the Expiring Keys dashboard to view any keys that will expire soon. You can click **View All Keys** to go directly to the Key Management page, or click **Generate new key**. to create a new key.

Dashboard



Home

Dashboard

Key Type	Key Reference	Expires At	Status
Press Enter to filter res	Press Enter to filter res	Press Enter to filter res	
SCMP	1052300000191791	Nov 09 2022 10:51:00 AM	● Active - Expi
Certificate	6049478960930177041503	Nov 09 2022 10:51:36 AM	● Active - Expi
Simple Order	6049478960930177041503	Nov 09 2022 10:51:36 AM	● Active - Expi

REST API Keys

The REST API uses public key cryptography to securely exchange information over the Internet. Before you can send requests for Cybersource services using the REST API, you must create a security key for your Cybersource merchant account.

The REST API supports two types of security key:

- Shared secret key for using HTTP signature authentication
- P12 certificate for using JSON Web Token authentication

REST API keys expire after 3 years.

Security keys can be used to make any call, including payments. Treat your security keys as you would any secure password.

You must use separate keys for the test and production environments.

When you sign up for a Sandbox account, your confirmation email contains a key and shared secret key for HTTP Signature authentication. To create the keys manually, or to use a JSON Web Token instead, follow the instructions in Creating a REST API Key (on page 9).

For more information about REST API authentication, see the Developer Center's Authentication section.

Creating a REST API Key

Follow these steps to create a REST API key:

1. Log in to the Business Center.
2. On the left navigation panel, click the **Payment Configuration** icon.
3. Click **Key Management**. The Key Management page appears.
4. Click **+ Generate Key**. The Create Key page appears.
5. Select the type of REST key that you want, and click **Generate Key**.
6. Choose the option below that corresponds to the key you selected.

- ◦ **REST Shared Secret:** copy the generated key to your clipboard by clicking the clipboard icon, or click **Download key** to download the shared secret. The first value is the key and the second value is the shared secret.
- ◦ **REST Certificate:** click **Download key** to download the certificate. The **Set Password** page appears.

7. If you selected a REST Certificate key, enter your new password and confirm your password.

8. Click **Generate Key**.

Simple Order API Keys

The Simple Order API uses public key cryptography to securely exchange information over the Internet. Before you can send requests for Cybersource services using the Simple Order API, you must go to the Business Center and create a security key for your Cybersource merchant account.

Simple Order API keys expire after 3 years.

Security keys can be used to make any call, including payments. Treat your security keys as you would any secure password.

You must use separate keys for the test and production environments.

Creating a Simple Order API Key

To create a Simple Order API key:

1. Log in to the Business Center.
2. On the left navigation panel, click the **Payment Configuration** icon.
3. Click **Key Management**. The Key Management page appears.
4. Click **+ Generate Key**. The Create Key page appears.
5. Select **Simple Order API** and click **Generate Key**.
6. Click **Download key** to download the .p12 file. The **Set Password** page appears.
7. Enter your new password and confirm your password.
8. Click **Generate Key**.

Secure Acceptance Keys

The Cybersource Secure Acceptance API uses public key cryptography to securely exchange information over the Internet. Before you can send requests for Cybersource services using the Secure Acceptance, you must go to the Business Center and create a security key for your Cybersource merchant account.

Secure Acceptance keys expire after 2 years.

Security keys can be used to make any call, including payments. Treat your security keys as you would any secure password.

You must use separate keys for the test and production environments.

Creating a Secure Acceptance Key

To create a Secure Acceptance key:

1. Log in to the Business Center.
2. On the left navigation panel, click the **Payment Configuration** icon.
3. Click **Key Management**. The Key Management page appears.
4. Click **+ Generate Key**. The Create Key page appears.
5. Select **Secure Acceptance** and click **Generate Key**.
6. Enter the required information:
    - Key Name: enter a name for this key.
    - Signature Version: select **1** from the drop-down menu.
    - Signature Method: select **HMAC-SHA256** from the drop-down menu.
    - Security Profile: select a security profile from the drop-down menu.
7. Click **Generate Key**. You can copy the access key and secret key by clicking the clipboard icons, or click **Download key** to download a text file containing both keys.

SOAP Toolkit Keys

The Cybersource SOAP Toolkit uses public key cryptography to securely exchange information over the Internet. Before you can send requests for Cybersource services using the SOAP Toolkit, you must go to the Business Center and create a security key for your Cybersource merchant account.

SOAP Toolkit keys expire after 3 years.

Security keys can be used to make any call, including payments. Treat your security keys as you would any secure password.

You must use separate keys for the test and production environments.

Creating a SOAP Toolkit Key

To create a SOAP Toolkit key:

1. Log in to the Business Center.
2. On the left navigation panel, click the **Payment Configuration** icon.
3. Click **Key Management**. The Key Management page appears.
4. Click **+ Generate Key**. The Create Key page appears.
5. Select **SOAP Toolkit** and click **Generate Key**.
6. You can copy the generated key to your clipboard by clicking the clipboard icon, or click **Download key** to download the key.

PGP Keys

Cybersource uses PGP encryption for Account Updater response files and Notice of Change (NOC) reports. For information about Account Updater, see the *Account Updater User Guide*. For information about NOC reports, see *Electronic Check Services Using the Simple Order API*.

A PGP public/private key pair enables you to use encryption to protect payment data. You exchange the public part of this key pair with Cybersource, which uses the public key to encrypt response files or NOC reports. You use the private part of the key pair to decrypt the response files or NOC reports. Only the private key can decrypt files that are encrypted with the public key.

PGP keys expire after 3 years.

Security keys can be used to make any call, including payments. Treat your security keys as you would any secure password.

You must use separate keys for the test and production environments.

Creating PGP Keys

You can use any OpenPGP-compliant software to generate PGP keys. The key you generate must be an RSA key. The following free OpenPGP solutions are available:

- Bouncy Castle
- GPG4WIN

Cybersource recommends that you do the following:

- Make the key at least 2048 bits long.
- Store the private key in an encrypted format to protect it from unauthorized use.
- Back up the private key in case of disaster.

Place the backup of the private key on removable media, and lock it in secure storage.

Cybersource does not receive a copy of your private key and cannot decrypt files that are encrypted with your public key. After you create a public/private key pair, add the public key to the Business Center as described in the next section.

Adding a PGP Key to Your Account

To add a PGP key to your account:

1. Log in to the Business Center.
2. On the left navigation panel, click the **Payment Configuration** icon.
3. Click **Key Management**. The Key Management page appears.
4. Click **+ Generate Key**. The Create Key page appears.
5. Select **PGP** and click **Generate Key**.
6. Enter the ASCII string into the text field, and click **Create Key**.

Granting User Permissions

A user account requires certain permissions to work with PGP keys and the Account Updater request files and reports. To grant user permissions:

1. Log in to the Business Center.
2. In the left navigation pane, choose **Account Management > Roles**.
3. Choose the role assigned to the user account that needs to work with PGP keys and click the **Edit** icon.
4. In the Role Editor, select the following permissions:
    a. Under Credit Card Account Updater Permissions, choose **View Status**. This option enables the user to view the status of uploaded Account Updater request files and NOC reports.

    b. Under Merchant Settings Permissions, choose **PGP Security Settings**. This option gives the user permission to upload, activate, and deactivate encryption keys.

    c. Under Reporting Permissions, choose **Report Download**. This option gives the user permission to download Account Updater response files and NOC reports.
5. At the bottom of the Role Editor, click **Save**.

Message-Level Encryption Keys

Before you can send requests for Cybersource services using message-level encryption, you must go to the Business Center and create a security key for your Cybersource merchant account.

Message-level encryption keys expire after 3 years.

Security keys can be used to make any call, including payments. Treat your security keys as you would any secure password.

You must use separate keys for the test and production environments.

Creating a Message-Level Encryption Key

To create a message-level encryption key:

1. Log in to the Business Center.
2. On the left navigation panel, click the **Payment Configuration** icon.
3. Click **Key Management**. The Key Management page appears.
4. Click **+ Generate Key**. The Create Key page appears.
5. Select **Message-Level Encryption** and click **Generate Key**.
6. Enter the string into the text field, and click **Create Key**.

Meta Keys

A meta key is an API key that a portfolio or account-level user can assign to multiple merchants IDs (MIDs) simultaneously. This can be useful for managing dozens or hundreds of MIDs.

Meta Keys are available for the following APIs:

- REST
- Simple Order API
- SOAP
- SCMP

When you are logged in to a portfolio or account in the Business Center, you can assign a meta key to either a static subset of MIDs or to all current and future MIDs. If you choose to assign a meta key to only a subset of MIDs, you can reassign the key later to all current and future MIDs.

When using a meta key, the portfolio or account submits the transaction on the MID's behalf. However, the transaction belongs to the MID. Searching for or reporting on the transaction is done at the MID-level. However, the portfolio, account, and merchant can all perform follow-on actions to the transaction. For example, capture, void, and refund.

Your portfolio must be configured for meta keys. Accounts can use meta keys only if they are under a portfolio that has meta keys enabled. To enable your portfolio for meta keys, contact your Cybersource representative. Accounts that are not under a portfolio are not eligible for meta keys at this time.

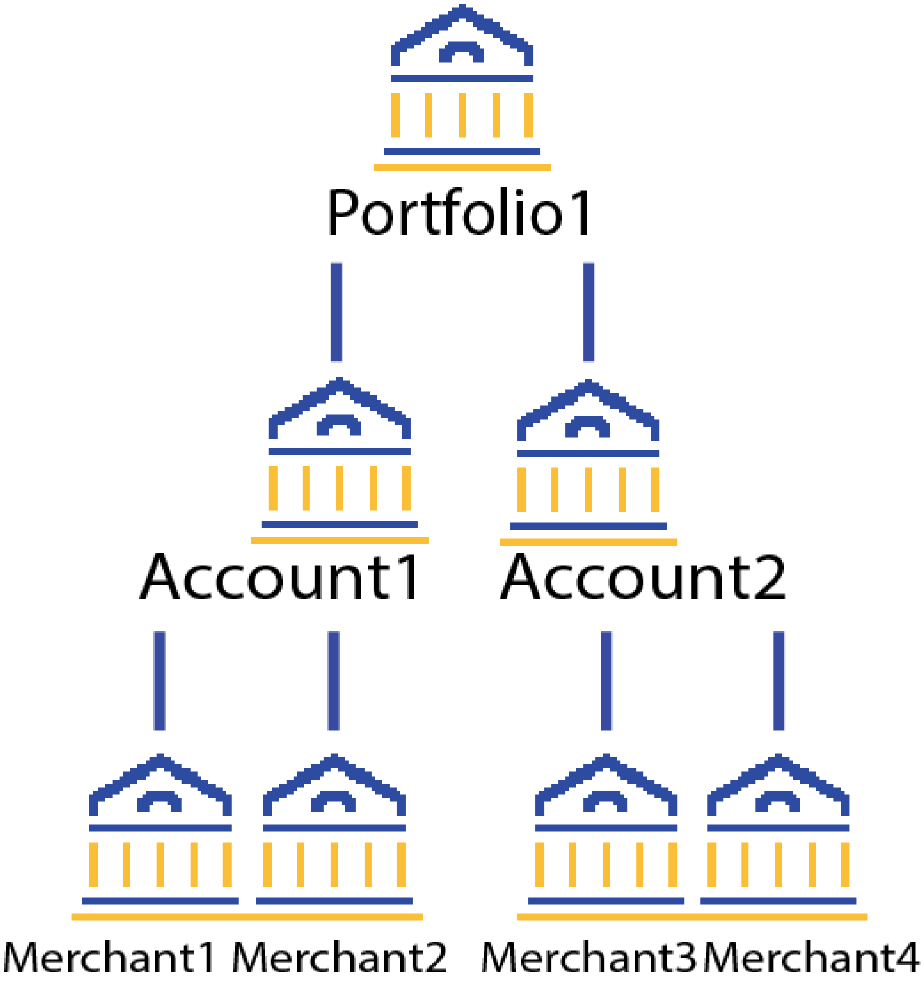
MIDs cannot assign meta keys but can assign regular keys. For security, do not give the meta key to merchants.

⚠ **Warning:** When a meta key expires, it expires for all MIDs to which it is assigned. All transactions using that meta key will fail. Careful monitoring is necessary to track meta key expiration dates. You must create and assign a new key before the previous key expires. The length of time after which a key expires depends on the API for which the key was created. Read the instructions for the API key you will use.

Understanding Hierarchy

There is a three-tiered hierarchy - portfolio, accounts, and merchants (MIDs), in descending order. Portfolios and accounts can both create meta keys, and the key can be assigned to a select group in the hierarchy or to all nodes in the hierarchy.

In the diagram below, if the portfolio (portfolio1) assigns a meta key to all merchants, every merchant in the diagram is assigned the key. However if an account (account1) assigns a key to all merchants, only merchants below it are assigned the key (merchant1, merchant2).



Using Meta Keys

Use the following instructions to create and assign meta keys.

Creating a Meta Key (on page 19)

Assigning a Meta Key to All Merchants (on page 21)

Assigning a Meta Key to a Selection of Merchants (on page 22)

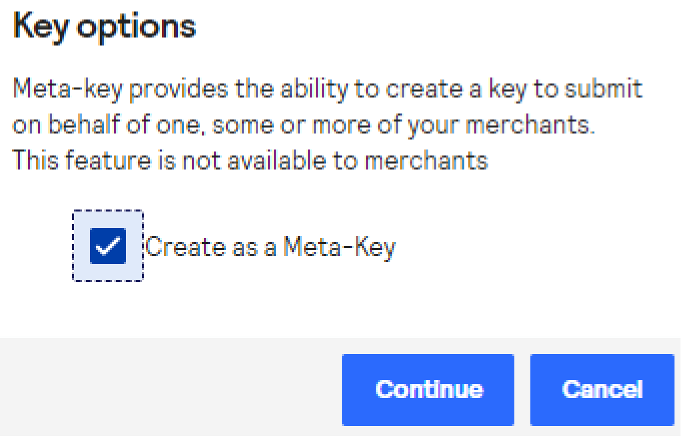
Regenerating a Meta Key (on page 28)

Creating a Meta Key

Portfolios and accounts that are enabled can create meta keys. A meta key is a key that you assign to more than one merchant account.

To create a meta key:

1. Log in to the Business Center.
2. On the left navigation panel, click the **Payment Configuration** icon.
3. Click **Key Management**. The Key Management page appears.
4. Click **+ Generate Key**. The Create Key page appears.
5. Select a key type and click **Generate Key**.
6. If meta key is enabled for your account, you will see a pop-up window asking if you want to create a meta key. To create a meta key, check the **Create as a Meta-Key** box and click **Continue**.



Key options

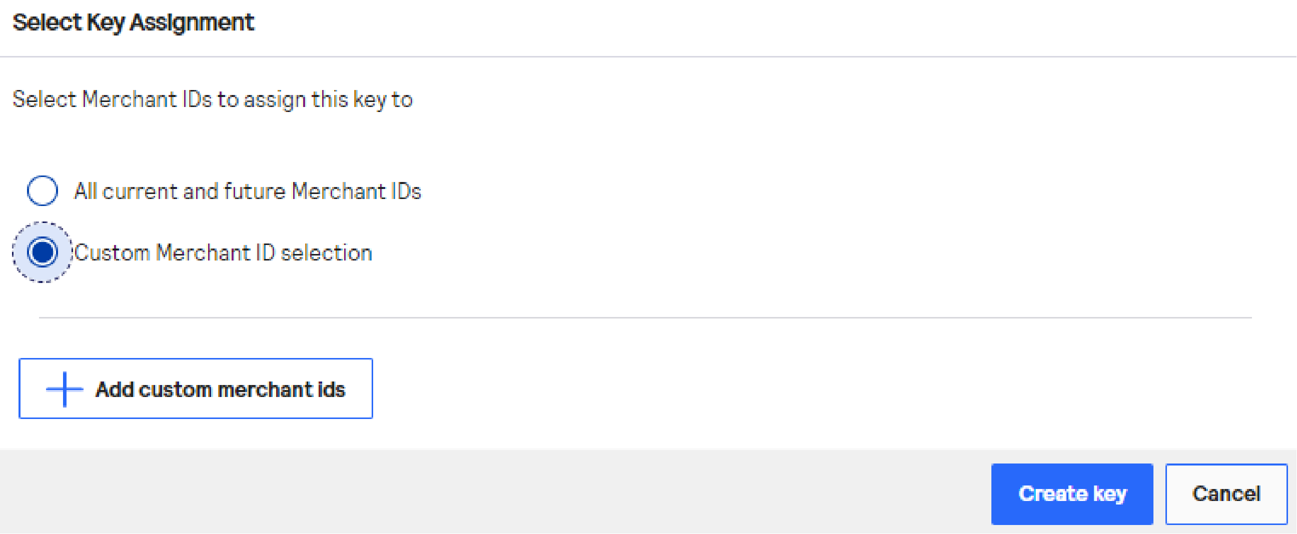
Meta-key provides the ability to create a key to submit on behalf of one, some or more of your merchants. This feature is not available to merchants

☑ Create as a Meta-Key

7. Choose one of the following:

- To assign this key to all accounts in the current portfolio, choose **All current and future Merchant IDs**, click **Create key**, and continue to the Create Key page. All future merchant IDs will automatically be assigned this key. You are finished, and there is no need to follow the steps below.

- To assign this key to a specific merchant or group merchants, choose **Custom Merchant ID selection**, click **Create key**, and continue to Step 8. This key will not be automatically assigned to any future merchants.



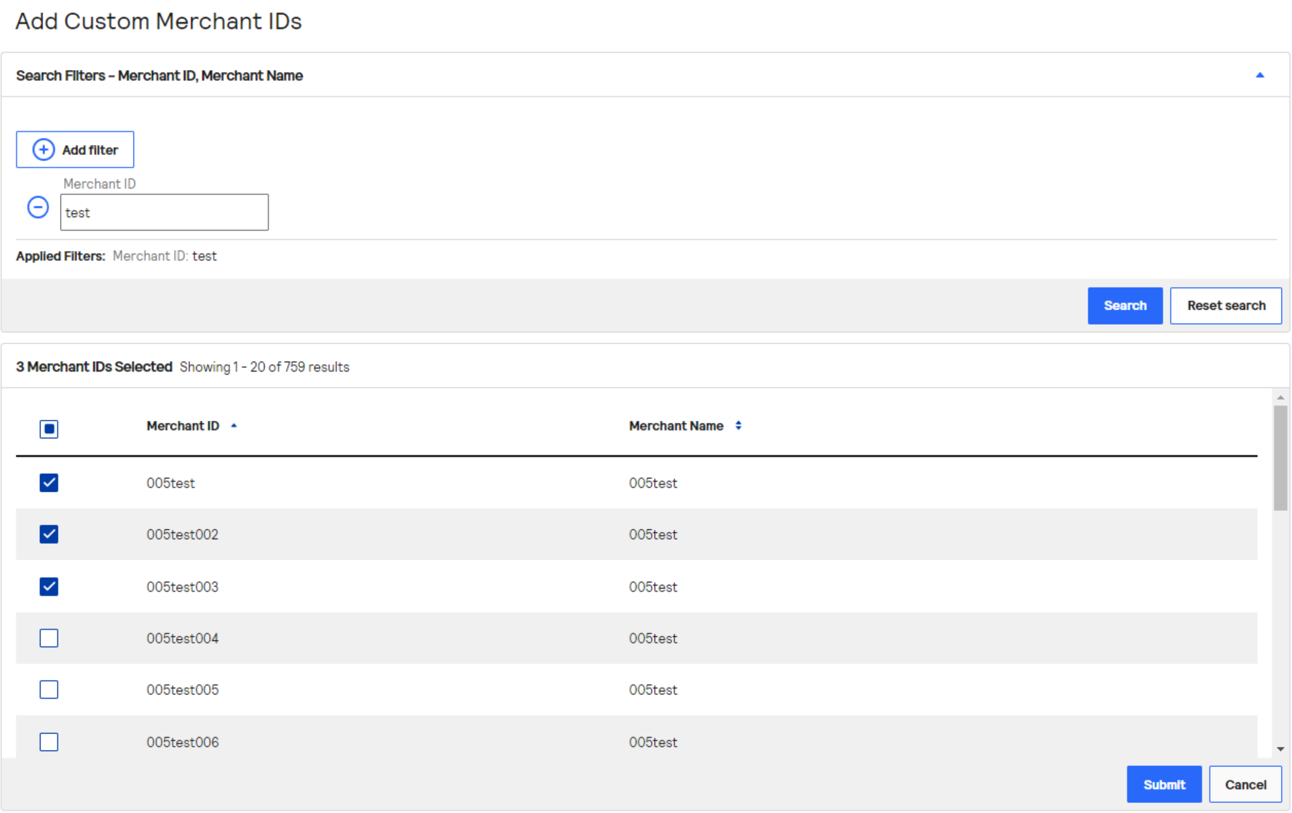
Select Key Assignment

Select Merchant IDs to assign this key to

○ All current and future Merchant IDs

● Custom Merchant ID selection

8. Click **+ Add custom merchant ids**. The Add Custom Merchant IDs page appears. By default, all merchant IDs are shown in the Merchant IDs table. To limit the list to a subset of merchant IDs, click **+ Add filter**, select a search filter from the drop-down menu, and click **Search**.



Add Custom Merchant IDs

Search Filters - Merchant ID, Merchant Name

Merchant ID: test

Applied Filters: Merchant ID: test

3 Merchant IDs Selected Showing 1 - 20 of 759 results

■	Merchant ID	Merchant Name
☑	005test	005test
☑	005test002	005test
☑	005test003	005test
□	005test004	005test
□	005test005	005test
□	005test006	005test

9. Use the check boxes to select one or more merchant accounts, and click **Submit**. You are returned to the Key Generation page.

10. Click **Create key**. Continue to the Create Key page.

Assigning a Meta Key to All Merchants

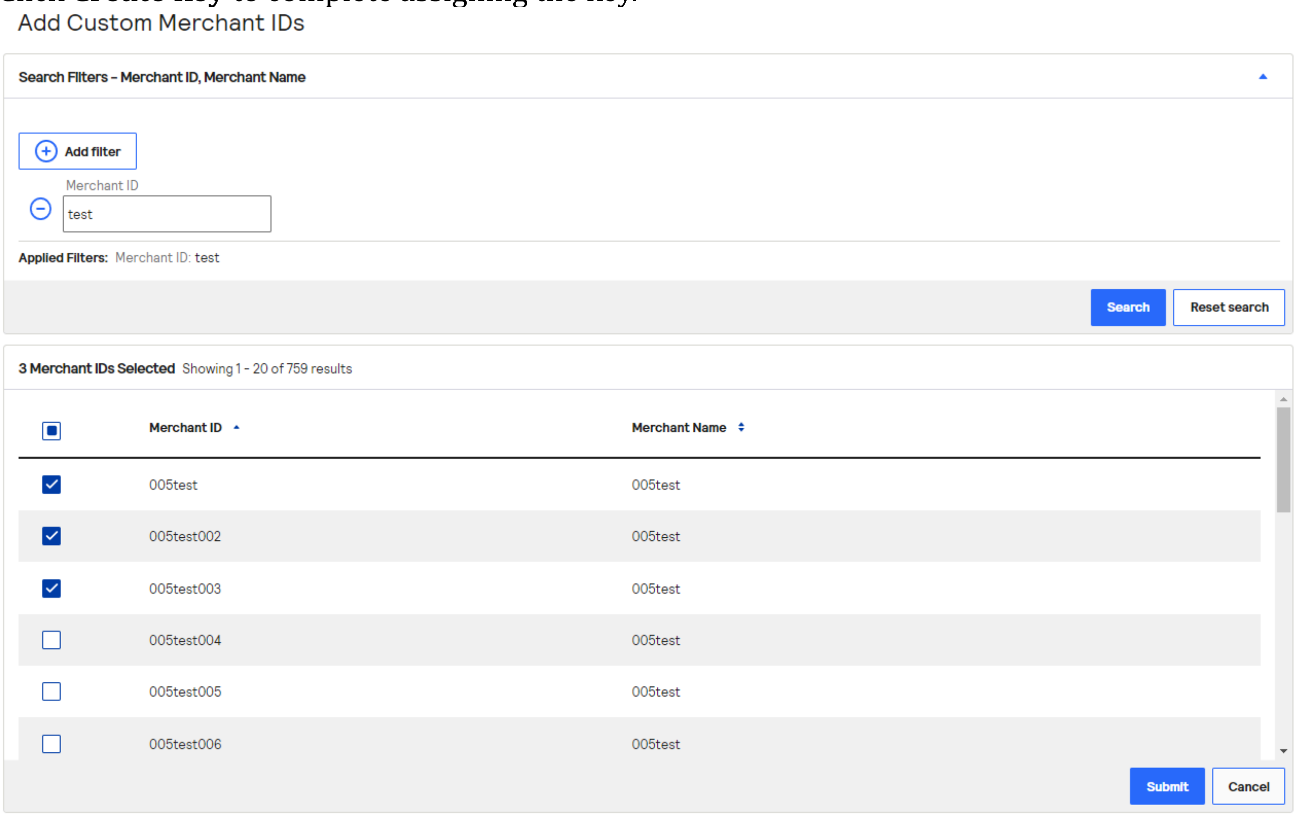
To assign the key to all current MIDs and automatically assign it to all future MIDs:

1. On the left navigation panel of the Business Center, click the **Payment Configuration** icon.
2. Click **Key Management**.
3. Find the key that you want to assign by searching and filtering.
4. In the Edit Key column, click the ✎ icon. The Edit Key page opens.
5. Check the **Meta Key** box and click **Continue**.
6. Select **All current and future MIDs** and click **Create key**.

Assigning a Meta Key to a Selection of Merchants

To assign the meta key to a custom selection of MIDs:

1. On the left navigation panel of the Business Center, click the **Payment Configuration** icon.
2. Click **Key Management**.
3. Find the key that you want to assign by searching and filtering.
4. In the Edit Key column, click the ✎ icon. The Edit Key page opens.
5. Check the **Meta Key** box.
6. Select **Custom MID selection**.
7. Click **+ Add custom merchant ids**.
8. Select the MIDs that you will assign the meta key to. To filter MIDs, click **+ Add filter**, select a filter, and click **Search**. Click **Save**.
9. Click **Submit** to complete the assignment and return to the Key Generation page.
10. Click **Create key** to complete assigning the key.



Merchant ID: test

Applied Filters: Merchant ID: test

3 Merchant IDs Selected Showing 1 - 20 of 759 results

■	Merchant ID ▲	Merchant Name ⇕
☑	005test	005test
☑	005test002	005test
☑	005test003	005test
□	005test004	005test
□	005test005	005test
□	005test006	005test

Submitting API requests using Meta Keys

To submit an API transaction using meta keys, use the following instructions.

REST API Payment Request using a Meta Key

REST API meta keys can use either HTTP signature or JSON web token.

If you use the SDK, see the meta key section of the sample code documentation.

If you do not use the SDK, use the following information.

HTTP Signature

When creating the signature, use the portfolio or account ID as the value of the **v-c_merchant-id** header. However, when sending the API request, use the transacting merchant ID (MID) as the value of the **v-c-merchant-id** header.

Signature Headers

```
v-c-merchant-id : merchantId
Key id : 266438gb-2120-4q36-8da7-fbb9a196d452
Shared Key : mgWWJVV2aGQyEPwufdhhe/GiFUhsNIwYvWMih4FMCN9E=
Request Target : post /pts/v2/payments
Host : api.cybersource.com
```

JSON Web Token

The portfolio or account ID is not required in the header or the body. Pass the P12 certificate along with the **v-c-merchant-id** header, using the transacting merchant account ID (MID) as the value.

JSON Web Token

```
// JWT Header
{
"v-c-merchant-id":"MerchantID",
"alg":"RS256",
"x5c":["MIIB2jCCAUOgAwlBAgIWNDg...=="]
}
// JWT Claimset
```

```
{
"digest":"0qjow45/L/m6DIHd8K90rL+tBKufR1RuyE4QG7whZQ=",
"digestAlgorithm":"SHA-256",
"iat":"1594249865"
}
// JWT Signature
{
data=base64urlEncode(JWT header)+"."+base64urlEncode(Claimset)
 signature=RS256Hash(data,private_key);
```

Simple Order API Payment Request using a Meta Key

In this Simple Order API payload, the value of the **merchantID** field is the merchant ID (MID) on behalf of whom this transaction being sent by the portfolio or account. The portfolio or account will use a Simple Order API meta-key certificate to digitally sign the request message before sending it to Cybersource. There is no need to declare the portfolio ID or account ID.

Simple Order API Payment Request

```
<requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.135">
  <merchantID>merchant12378</merchantID>
  <merchantReferenceCode>NGTS1500</merchantReferenceCode>
  <clientLibrary>Java XML</clientLibrary>
  <clientLibraryVersion>5.0.2</clientLibraryVersion>
  <clientEnvironment>Mac OS X/10.14.5/Oracle Corporation/1.8.0_161</clientEnvironment>
  <invoiceHeader>
    <merchantDescriptor>NGMerchants*MyProduct</merchantDescriptor>
    <merchantDescriptorContact>444-444-4444</merchantDescriptorContact>
  </invoiceHeader>
  <billTo>
    <firstName>TSTester</firstName>
    <lastName>NextGen</lastName>
    <street1>201 S. Division St.</street1>
    <street2>Suite 500</street2>
    <city>Ann Arbor</city>
    <state>MI</state>
    <postalCode>48104-2201</postalCode>
    <country>US</country>
    <phoneNumber>999-999-9999</phoneNumber>
    <email>rm@cybersource.com</email>
    <ipAddress>66.185.179.2</ipAddress>
  </billTo>
  <shipTo>
    <firstName>Olivia</firstName>
    <lastName>White</lastName>
    <street1>1295 Charleston Rd</street1>
    <street2>Cube 2386</street2>
    <city>Mountain View</city>
    <state>CA</state>
    <postalCode>94043</postalCode>
    <country>US</country>
    <phoneNumber>650-965-6000</phoneNumber>
  </shipTo>
  <purchaseTotals>
    <currency>usd</currency>
    <grandTotalAmount>2202</grandTotalAmount>
  </purchaseTotals>
  <card>
    <accountNumber>4111111111111111</accountNumber>
    <expirationMonth>12</expirationMonth>
    <expirationYear>2021</expirationYear>
    <cvNumber>111</cvNumber>
    <cardType>001</cardType>
  </card>
  <ccAuthService run="true">
    <commerceIndicator>internet</commerceIndicator>
    <billPayment>true</billPayment>
  </ccAuthService>
  <ccCaptureService run="true"/>
  <businessRules>
    <ignoreAVSResult>true</ignoreAVSResult>
    <ignoreCVResult>true</ignoreCVResult>
  </businessRules>
</requestMessage>
```

SOAP Payment Request using a Meta Key

The request envelope uses a SOAP API password generated for the meta key. The value of the **wsse. Username** field is the portfolio or account ID. The value of the **merchantID** field is the MID on behalf of whom this transaction being sent by the portfolio or account.

In this example, the request is being sent by a portfolio. The portfolio ID is `portfolioabc` and the merchant ID is `merchant12378`.

SOAP API Payment Request

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="h
ttp://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <SOAP-ENV:Header>

  <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssec
urity-secext-1.0.xsd" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-w
ssecurity-utility-1.0.xsd">
      <wsse:UsernameToken>
        <wsse:Username>portfolioabc</wsse:Username>

  <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-to
ken-profile-1.0#PasswordText">8SbCuVZ4FLYakM7Mm+g4jlXgV5kN/uPNfRmpTj8yKNrmvmZU25tFiTyA6Qbx
4jakhKYGRDqnma/52WrOu4GQm9WbYp5xyjlE16+YQFJRXY9jQHAmikc18Na3YugZzuBbu1aRcr597pwmdxkoWb87l+
6gkqJU04eHayfiMNWSkq8piBcK5fIKIah9eSQdH31DaaqAQHvJJKLL8Ki+7TYJHKc24fBLKY4QPKr0pdGNubqjJxl8
YyJXozVv3F4BcmgaklqCVAiORTr/IKTczU6Y56BrPsixsoehBetzqwxnyUjRkS1172fsOFPqPwZSGhMoATyM+EYXTE
Zoni58q5zvvw==</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <requestMessage xmlns="urn:schemas-cybersource-com:transaction-data-1.151">
      <merchantID>merchant12378</merchantID>
      <merchantReferenceCode>BATSNTA1003</merchantReferenceCode>
      <billTo>
        <firstName>James</firstName>
        <lastName>Dough</lastName>
        <street1>600 Morgan Falls Road</street1>
        <street2>Room 2-2123</street2>
        <city>Atlanta</city>
        <state>GA</state>
        <postalCode>30350</postalCode>
        <country>US</country>
        <phoneNumber>650-965-6111</phoneNumber>
        <email>jdough@cybersource.com</email>
      </billTo>
      <item id="0">
        <unitPrice>1.00</unitPrice>
      </item>
      <item id="1">
        <unitPrice>1.00</unitPrice>
      </item>
      <purchaseTotals>
        <currency>USD</currency>
      </purchaseTotals>
      <card>
        <accountNumber>4111111111111111</accountNumber>
        <expirationMonth>04</expirationMonth>
        <expirationYear>2025</expirationYear>
        <cvNumber>111</cvNumber>
        <cardType>001</cardType>
      </card>
      <ccAuthService run="true"/>
      <ccCaptureService run="true"/>
    </requestMessage>
    <urn:requestMessage xmlns:urn="urn:schemas-cybersource-com:transaction-data-1.151"/>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SCMP API Payment Request using a Meta Key

In an SCMP API payment request, the value of the **merchant_id** field is the MID on behalf of whom this transaction being sent by the portfolio or account. The value of the **sender_id** field is the ID of the portfolio or account. The portfolio or account uses the SCMP API meta-key certificate to sign and encrypt the request before sending it to Cybersource. The SCMP API payment request below is sent by a portfolio.

SCMP API Request

```
request_id=5580301042523113616883
sender_id=portfolioabc
merchant_id=merchant123
merchant_ref_number=MERCH_SCMP_123
ics_applications=ics_auth
currency=usd
return_auth_record=true
client_lib_version=Oracle Corporation/1.8.0_192/Windows Server 2008
 R2/6.1/-/Java/5.2.1/Oracle Corporation/1.8.0_201/Mac OS X/10.14.3/-/Java/5.2.0
offer0=amount:2^offer_id:0^product_name:PName1^merchant_product_sku:testdl^quantit
y:1^product_code:clothing
ignore_avs=yes
tax_indicator=Y
user_po=LII Test
customer_email=jdoe@example.com
customer_cc_expmo=10
customer_firstname=Bob
customer_cc_expyr=2020
customer_cc_number=4111111111111111
customer_hostname=bob.bob.com
customer_ipaddress=120.1.1.1
customer_lastname=Dough
customer_phone=555-555-5555
```

```
bill_country=US
bill_city=Atlanta
bill_zip=30350
bill_address2=Room 2-2123
bill_address1=123 Test Road
bill_state=GA
ship_to_email=bob@example.com
ship_to_lastname=Jones
ship_to_country=US
ship_to_county=Monroe
ship_from_city=San Jose
ship_to_city=bloomington
ship_to_co_name=Bob's Excursion Emporium
ship_from_zip=94538
ship_from_state=CA
ship_to_zip=47404
ship_from_country=US
ship_from_county=Santa Clara
ship_to_state=indiana
ship_to_firstname=Cat
ship_to_address2=suite 2-5A
ship_to_address1=37 se main street
```

Regenerating a Meta Key

When a security key expires, it must be updated. If you update the meta key manually, you have to reassign merchants to it, which can be time-consuming. Meta key regeneration enables you to update the meta key with all its assignments intact, streamlining the process.

1. On the left navigation panel, click the **Payment Configuration** icon.
2. Click **Key Management**. The Key Management page opens.
3. Use the Search tool to find the key you want to regenerate. Results appear in the Search Results table.
4. Click the **Regenerate meta key** button for the key you want to regenerate. The Key Generation page opens. The new key appears on the screen. The original key remains active until its original expiration date.
5. Provide the new key details to the merchants associated with the affected MIDs, and instruct them to update the information wherever it is used.

Searching for Keys and Filtering Results

When your account contains a large number of keys, it might be necessary to sort them using the search function and then filter the search results.

Search (on page 29)

Filter (on page 29)

Searching for Keys

You can search for keys by using the Search options.

1. Click the **Key Type** drop-down menu and select a key type (required).
2. Click the **Created At** drop-down menu and select a date range (required). To choose a custom date, select **Custom Date**. Choose the start date, end date, start time, and end time.
3. For partner level accounts: Click the **Merchant** drop-down menu and select a merchant (required).
4. To add a custom search option, click **Add filter** and choose a filter from the drop-down menu. A new drop-down menu appears based on your choice, from which you can choose additional options.
5. Click **Search**.
   Results are displayed in the table below the search options. To reset the search, click **Reset Search**.

Filtering and Sorting Key Search Results

Large sets of search results can be sorted and filtered in the table of keys.

To sort the results of a column, click the column heading. Columns with alphanumeric data sort alphabetically from A-Z or Z-A. Columns with numbers will filter from greatest to smallest or smallest to greatest. The Delete column is not sortable.

Use the filters at the top of the page to create a subset of results.Only keys that match the filter text are returned. To reset the search, click **Reset Search**.

Deleting Keys

You can delete a key when you no longer need to use it for payment processing.

Keys become inactive after reaching the expiration date.

To delete security keys:

1. On the left navigation panel, click the **Payment Configuration** icon.
2. Click **Key Management**. The Key Management page appears.
3. In the table of keys, find the key that you want to delete, and click the **Delete** icon in the row for that key.
4. Click **Yes**.

Deactivating Keys

You can deactivate a key if you no longer need to use it. You can only deactivate MLE, PGP, or Secure Acceptance keys.

Keys become inactive after reaching the expiration date.

To deactivate security keys:

1. On the left navigation panel, click the **Payment Configuration**icon.
2. Click **Key Management**. The Key Management page appears.
3. Find the key in the table of keys, or search for one using the search filters, and click the link for that key in the Keys column. The Key Information page appears.
4. Click the **Deactivate** (⊖) icon.
5. Click **Yes**.