

Account Updater

Developer Guide

© 2025. Cybersource Corporation. All rights reserved.

Cybersource Corporation (Cybersource) furnishes this document and the software described in this document under the applicable agreement between the reader of this document (You) and Cybersource (Agreement). You may use this document and/or software only in accordance with the terms of the Agreement. Except as expressly set forth in the Agreement, the information contained in this document is subject to change without notice and therefore should not be interpreted in any way as a guarantee or warranty by Cybersource. Cybersource assumes no responsibility or liability for any errors that may appear in this document. The copyrighted software that accompanies this document is licensed to You for use only in strict accordance with the Agreement. You should read the Agreement carefully before using the software. Except as permitted by the Agreement, You may not reproduce any part of this document, store this document in a retrieval system, or transmit this document, in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written consent of Cybersource.

Restricted Rights Legends

For Government or defense agencies: Use, duplication, or disclosure by the Government or defense agencies is subject to restrictions as set forth the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 and in similar clauses in the FAR and NASA FAR Supplement.

For civilian agencies: Use, reproduction, or disclosure is subject to restrictions set forth in subparagraphs (a) through (d) of the Commercial Computer Software Restricted Rights clause at 52.227-19 and the limitations set forth in Cybersource Corporation's standard commercial agreement for this software. Unpublished rights reserved under the copyright laws of the United States.

Trademarks

Authorize.net and The Power of Payment are registered trademarks of Cybersource Corporation. Cybersource and Cybersource Decision Manager are trademarks and/or service marks of Cybersource Corporation. Visa, Visa International, Cybersource, the Visa logo, the Cybersource logo, and 3-D Secure are the registered trademarks of Visa International in the United States and other countries. All other trademarks, service marks, registered marks, or registered service marks are the property of their respective owners.

Version: 25.03.01

Contents

- Account Updater.....5**
 - Recent Revisions to This Document..... 6
 - VISA Platform Connect: Specifications and Conditions for Resellers/Partners..... 7
- Introduction to Account Updater.....8**
 - Tokenization.....8
 - Token Harvest Updates..... 9
 - Token Selective Batch Updates..... 9
 - Storing PANs..... 9
 - Enabling Account Updater.....9
 - Business Center Permissions..... 10
 - Terms of Use..... 10
- Token Updates.....12**
 - Token Harvest.....12
 - Token Batch Updates.....12
 - Submitting Visa and Mastercard One-Time Updates..... 13
 - Registering Tokens for American Express Daily Updates..... 13
 - Batch Creation Request Examples.....13
 - Batch Creation Response Examples.....16
 - Retrieving Batch Status..... 17
 - American Express Daily and Token Harvest Update Reports.....21
 - Retrieving a Batch with a Batch ID.....26
 - American Express Daily Updates.....27
- PAN Updates.....33**
 - Creating Security Keys.....33
 - Authentication Key..... 33
 - PGP Public/Private Key..... 33
 - Formatting a Request File.....34
 - Request Header Record.....34
 - Request Detail Record.....35
 - Request Footer Record..... 36
 - Request File Examples.....37
 - Uploading a Request File.....37

Email Notification.....	39
Viewing the Status of a Batch File	40
Downloading a Response File.....	40
Response File Records.....	41
Response Header Record.....	41
Response Detail Record.....	41
Response Footer Record.....	43
Response File Examples.....	44
Testing.....	45
American Express Test Card Numbers.....	45
Mastercard Test Card Numbers.....	46
Visa Test Card Numbers.....	48
API Fields.....	50
Request Fields.....	50
Response Fields.....	52
PAN Upload Response Codes and Reason Codes.....	61

Account Updater

This section describes how to use this guide and where to find further information.

Audience and Purpose

This guide is written for merchants and partners who want to keep stored card data updated for recurring payments and credentials-on-file payments. It describes tasks that a merchant or partner must complete to submit a batch of tokens using the REST API or to upload request files with new customer primary account numbers (PANs). Account Updater is intended to help you reduce the number of authorization declines to retain revenue and reduce the cost of manually updating payment data. You can store card data within your system or within the tokenization system, the Token Management Service, and the legacy Payment Tokenization.

Conventions

The following special statements are used in this document:



Important: An *Important* statement contains information essential to successfully completing a task or learning a concept.



Warning: A *Warning* contains information or instructions, which, if not heeded, can result in a security risk, irreversible loss of data, or significant cost in time or revenue or both.

Related Documentation

Refer to the Technical Documentation Hub in the Cybersource Developer Center for additional technical documentation:

<https://developer.cybersource.com/docs.html>

Customer Support

For support information about any service, visit the Support Center:

<http://support.cybersource.com>

Recent Revisions to This Document

25.03.01

Added a note that a merchant can only request updates for the tokens that they created. See [Token Harvest \(on page 12\)](#).

Added a note that an Account Updater merchant can request updates for any TMS tokens within the same TMS vault. See [Token Batch Updates \(on page 12\)](#).

Added a note about time taken to generate reports. See [Submitting Visa and Mastercard One-Time Updates \(on page 13\)](#) and [PAN Updates \(on page 33\)](#).

Removed the sample Java code for uploading PANs and added a sample PAN upload request. See [Uploading a Request File \(on page 37\)](#).

Added a note to the ACL response code that the state of the instrument identifier token gets set to closed. See [PAN Upload Response Codes and Reason Codes \(on page 61\)](#).

Updated the steps for viewing the status of a batch file. See [Viewing the Status of a Batch File \(on page 40\)](#).

Updated the steps for downloading a response file. See [Downloading a Response File \(on page 40\)](#).

Removed tokens from the description in the **included** field in [Request Fields \(on page 50\)](#).

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.

Introduction to Account Updater

Account Updater helps you to keep stored card data up to date so that you can improve authorization success rates by cutting down on declined payments related to lost, stolen, or expired cards. Account Updater updates card data stored on your servers or in the Cybersource tokenization system, which includes Recurring Billing, Token Management Service (TMS), and the legacy Payment Tokenization. The updates include expiration dates, card numbers, and brands.

Account Updater provides a single interface to access updates from the Visa Account Updater and Mastercard Automatic Billing Updater services. If you are using TMS, Cybersource also provides updates to you from the American Express Cardrefresher service.

Integration options are available depending on whether you are:

- Using TMS or Recurring Billing for tokenization. See [Tokenization \(on page 8\)](#).
- Storing PANs on your system. See [Storing PANs \(on page 9\)](#).

Tokenization

If you are already using TMS or Recurring Billing, Account Updater is simple to integrate. You will benefit from token updates from Visa, Mastercard, and American Express.



Important: If you are using tokens that preserve the last four digits, the new PAN updates that you receive from card networks result in the token (subscription ID) being changed. When you receive the new tokens in the update report, you should update them in your system immediately to avoid authorization failures.



Important: Account Updater is incompatible with instrument identifier tokens that are 22-digits in length.

To use this service you must generate the REST API keys in the key management section of the [Business Center](#). Contact your Cybersource representative if you do not have this option enabled for your account. Find more details on authenticating API requests on the [Developer Center](#).

Token Harvest Updates

You can configure Account Updater to automatically update all of your tokens with the latest credit card data. For Visa and Mastercard, update reports are generated monthly. For American Express, update reports are generated daily. See [Token Harvest \(on page 12\)](#).

Token Selective Batch Updates

Using the Account Updater REST API, you can add the specific tokens (also known as subscriptions) that you wish to batch update. For Visa and Mastercard, these batches produce one-time update reports.

For tokens containing American Express cards, card numbers are enrolled for automatic updates for which reports are generated daily. Tokens are removed automatically when deleted or updated to a different card type. See [Token Batch Updates \(on page 12\)](#).

Storing PANs

When you directly manage customer card data, you create a file containing PANs that Account Updater updates. Create a request file containing new PANs and POST it to the Account Updater URL. Download the response file using the Business Center or a client application. See [PAN Updates \(on page 33\)](#).

Enabling Account Updater

Contact your account representative who submits enrollment forms on your behalf to Mastercard and Visa. The enrollment process can take up to 10 business days.

For American Express Cardrefresher, contact your American Express representative to ensure that your organization is enabled for Cardrefresher using your existing American Express credentials.



Important: If you are going to process Account Updater requests on behalf of merchants for whom you are not the merchant of record, you must enroll in Account Updater as a billing aggregator.

Billing aggregators can use Account Updater for PAN upload updates. In the Account Updater request files, they must indicate the merchant for whom the request is made. If you are a billing aggregator and fail to include the proper data in a record, Cybersource rejects the record and does not process your requests. For more information about PAN upload updates, see [PAN Updates \(on page 33\)](#).

Business Center Permissions

As part of the enrollment process, an administrator must grant you permission in the Business Center to perform the following actions. If you are an administrator, you already have these permissions:

- View the status of a request file.
- Create REST API credentials for token batch updates.
- For PAN updates, add and activate a PGP Security Key.
- Access downloadable response files.

Terms of Use

By using the Account Updater service, you agree to comply with the Visa U.S.A. Operating Regulations, Visa Account Updater Terms of Use, Mastercard rules and regulations, American Express rules and regulations, and all other applicable rules and regulations issued by any card association.

In addition, you must:

- Request an update for every participating Visa account in your customer database at least:
 - Once every 180 calendar days if you bill daily, weekly, monthly, quarterly, or biannually.
 - Once every 365 calendar days if you bill annually.
- Submit inquiries only for those accounts with which you have an ongoing customer relationship.
- Update your customer account database within 5 business days of receiving an update.
- Ensure that all update information you receive is properly, completely, and accurately incorporated into your data store for use in future transactions.
- Correct erroneous account information within 5 business days of receipt of error notifications.

You may not:

- Request updates on accounts that have returned a response of Contact Card Holder (CCH). You must review your response file for CCH responses and take appropriate action such as removing the customer record from your billing cycle until you have contacted the cardholder.
- Submit update inquiries on behalf of any other entity unless you have enrolled in Account Updater as a billing aggregator.

Token Updates

You can arrange for Account Updater to harvest and update all of your tokens on an agreed-upon date; this is called the *token harvest update* method. You must retrieve a monthly report for Visa and Mastercard updates and a daily update report if you are enrolled for American Express Cardrefresher.

Account Updater requires card number and expiration dates, so the token harvest option is available only when you use the *customer* or *payment instrument* tokens. *Instrument identifier* tokens that are not associated with a *customer* or *payment instrument* token are not updated.

The Account Updater REST API enables you to selectively send a POST request for a batch of tokens (subscription IDs) for American Express cards to be enrolled, processed, and updated. This is called the *token batch update* method.

Both options use the standard REST API HTTP signature authentication method.

Token Harvest

On an agreed-upon monthly date, Account Updater submits your tokenized cards to Visa and Mastercard for updates. American Express tokenized cards are automatically enabled for Cardrefresher daily, and deleted or updated tokens are de-enrolled automatically.



Important: The Account Updater harvest is configured at the Transacting ID level and can only request updates for TMS tokens it has created. If multiple TMS Transacting IDs are creating TMS tokens, each ID must be set up for the Account Updater.

You must retrieve American Express reports daily, or Visa and Mastercard reports monthly. For more information about reports, see [Retrieving Batch Status \(on page 17\)](#).

It is best practice to request updates for your tokens 3 to 5 days before your billing cycle begins. You can choose any calendar day, from the 1st through the 28th.

Token Batch Updates

A Token Batch Update request can contain up to 10,000,000 TMS tokens.



Important: Token Batch Updates allow an Account Updater MID to request updates for any TMS token within the same TMS vault. An Account Updater MID using Token Batch Update can request an update for a TMS token that it did not create.

To access endpoints, use an HTTPS POST request with a valid JSON payload:

- Test: <https://apitest.cybersource.com/accountupdater/v1/batches>
- Production: <https://api.cybersource.com/accountupdater/v1/batches>

Submitting Visa and Mastercard One-Time Updates

You can submit a specific set of tokens or all tokens for a one-time update to Visa and Mastercard. Your update report should be generated in 24 to 48 hours. A report can take up to 5 days to process if there are delays from the card association. A successful response to the batch creation returns a batch ID. You can check the status of the batch, which returns the URI of the batch update report when available.

To perform a one-time update, set the **type** field to `oneOff`.

Registering Tokens for American Express Daily Updates

You can register tokens for American Express cards for daily updates. Account Updater receives updates from American Express daily, applies them to your tokens, and produces a daily report that is available to you through the REST API.

To indicate that the batch contains tokens to be enrolled with American Express Cardrefresher, set the **type** field to `amexRegistration`.

Batch Creation Request Examples

TMS supports these types of tokens:

- Customer
- Payment instrument
- Instrument identifier

Customer tokens and payment instrument tokens store the expiration date in addition to the PAN. Instrument identifier tokens store only the PAN.

Each batch request should contain only one token type: customer, payment instrument, or instrument identifier.

Account Updater requires the existing PAN and expiration date. If you are using instrument identifier tokens, you must also specify the expiration date.

Creating a Batch of Two Customer or Payment Instrument Tokens

```
{
  "type": "oneOff",
  "included": {
    "tokens": [
      {
        "id": "3FA02EB4E49B65FDA194B38994B1F3F3"
      },
      {
        "id": "D1944BD9A7F9052BE431A276EB492C39"
      }
    ]
  },
  "merchantReference": "Merchant reference",
  "notificationEmail": "email@example.com"
}
```

Creating a Batch of Two Instrument Identifier Tokens

```
{
  "type": "amexRegistration",
  "included": {
    "tokens": [
      {
        "id": "7B1F41664F08F6DD3BB1C63892907524",
        "expirationMonth": "12",
        "expirationYear": "2021"
      },
      {
        "id": "E8F44CFA7EBEADDB06A5A9625E7F8696",
        "expirationMonth": "12",
        "expirationYear": "2021"
      }
    ]
  },
  "merchantReference": "Merchant reference",
  "notificationEmail": "email@example.com"
}
```

Batch Creation Response Examples

Example: HTTP 202: Successful batch creation

```
{
  "_links": {
    "self": {
      "href": "https://api.cybersource.com/accountupdater/v1/batches"
    },
    "status": {
      "href":
"https://api.cybersource.com/accountupdater/v1/batches/15269996945240002139594385/status"
    }
  },
  batchId: "15269996945240002139594385",
  batchItemCount: 2
}
```

Example: HTTP 401: Not authorized to access resource

```
{
  "_links": {
    "self": {
      "href": "https://api.cybersource.com/accountupdater/v1/batches"
    }
  },
  "code": "FORBIDDEN_RESPONSE",
  "correlationId": "c7b74452a7314f9ca28197d1084447a5",
  "detail": "You are not authorized to access this resource",
  "fields": null,
  "localizationKey": "cybsapi.forbidden.response",
  "message": "Unauthorized Access"
}
```

Action: Verify that the credentials that you are using are correct for the environment you are accessing. Ensure that your credentials have not expired and that your authentication process is correct.

Example: HTTP 422: Failure to process request

```
{
  "_links": {
    "self": {
      "href": "https://api.cybersource.com/accountupdater/v1/batches"
    }
  },
  "code": "VALIDATION_ERROR",
  "correlationId": "c7b74452a7314f9ca28197d1084447a5",
  "detail": "One or more fields failed validation",
  "fields": [
    {
      "path": "notificationEmail",
      "message": "Email address provided should not be 'null'",
      "localizationKey": "cybsapi.ondemand.batch.email.null"
    }
  ],
  "localizationKey": "cybsapi.validation.error",
  "message": "Field validation error"
}
```

Action: Examine the message to learn what failed validation. Verify that the structure of your JSON format is correct.

Retrieving Batch Status

The update reports contain details of updates that have been applied to the tokens in your batch, and include a masked version of new card numbers and/or expiration dates.

To retrieve the batch, obtain the batch ID. The process for retrieving the batch depends on how the batch was created.

Visa and Mastercard One-Time Updates

To retrieve one-time updates, verify the batch status URL that was returned in the one-time batch creation. For more info about one-time updates, see [Submitting Visa and Mastercard One-Time Updates \(on page 13\)](#).

To get the status of the batch, send an authenticated GET request, including the header `ACCEPT=application/json`, to one of these resources:

- Test: <https://apitest.cybersource.com/accountupdater/v1/batches/{batchId}/status>
- Production: <https://api.cybersource.com/accountupdater/v1/batches/{batchId}/status>

One-Time Batch Update Response

Batch processing by Visa and Mastercard can take up to 48 hours; therefore, reports are not available immediately. A successful response returns the status of the batch and additional information relating to the batch as it becomes available.

These batch statuses are possible:

Status Responses

Status	Description
Received	The batch was received and is being checked for errors.
Processing	The batch was sent to the card association(s) to be updated.
Updating	Account Updater received a response from the card association(s) and is updating the tokens.
Completed	Updates have been applied to the tokens. The batch report URL is now available.
Failed	Review specific error message.

Not all data is available immediately. As the batch status progresses from **Received** through **Processing** and **Updating** to **Completed**, additional data becomes available in the batch status. Check the status after submitting the batch to catch early errors that might result in a **Failed** status or incorrect **acceptedRecords** or **rejectedRecords** counts. The URL of the batch report appears when the status is **Completed**.

HTTP 200: Successful Response

```
{
  "_links": {
    "self": {
      "href":
        "https://api.cybersource.com/accountupdater/v1/batches/15269996945240002139594385/status"
    },
    "report": [
      {
        "href":
          "https://api.cybersource.com/accountupdater/v1/batches/15269996945240002139594385/status"
      }
    ]
  }
}
```

```

    }
  ]
  "batchCaEndpoints": "VISA,MASTERCARD",
  "batchCreatedDate": "2018-05-22T14.38.57Z",
  "batchId": "15269996945240002139594385",
  "batchSource": "TOKEN_API",
  "billing": {
    "nan": "0,",
    "ned": "9,",
    "acl": "5,",
    "cch": 0
  },
  "description": "Batch processing complete. Report URL now available.",
  "merchantReference": "Merchant reference",
  "status": "COMPLETED",
  "totals": {
    "acceptedRecords": "8,",
    "rejectedRecords": "7,",
    "updatedRecords": "8,",
    "caResponses": "14,",
    "caResponsesOmitted": 6
  }
}

```

Customer Token Batch Report Responses

The following is an example Token Batch Report. It includes one of each possible billable response

```

{
  "version": "1.0",
  "reportCreatedDate": "2023-11-03T21:19:26Z",
  "batchId": "16990463332850000715218359",
  "batchSource": "TOKEN_API",
  "batchCaEndpoints": "VISA,MASTERCARD",
  "batchCreatedDate": "2023-11-03T21:18:53Z",
  "merchantReference": "4 CU (NAN,NED,ACL,CCH)",
  "totals": {
    "updatedRecords": 4,
    "acceptedRecords": 4,
    "caResponses": 4,
    "rejectedRecords": 0,
    "caResponsesOmitted": 0
  },
  "billing": {
    "nan": 1,
    "ned": 1,

```

```

    "acl": 1,
    "cch": 1
  },
  "records": [
    {
      "id": "48496609484",
      "sourceRecord": {
        "token": "0946E5389B9CC6E8E063AF598E0AB303",
        "cardNumber": "400000XXXXXX7183",
        "cardExpiryMonth": "12",
        "cardExpiryYear": "2020",
        "cardType": "001",
        "customerId": "0946E5389B9CC6E8E063AF598E0AB303",
        "paymentInstrumentId": "0946E514CBD8C699E063AF598E0AE7EE",
        "instrumentIdentifierId": "7030030000048367183"
      },
      "responseRecord": {
        "response": "ACL",
        "reason": "800"
      }
    },
    {
      "id": "48496609494",
      "sourceRecord": {
        "token": "0946E5389C61C6E8E063AF598E0AB303",
        "cardNumber": "400000XXXXXX6236",
        "cardExpiryMonth": "12",
        "cardExpiryYear": "2020",
        "cardType": "001",
        "customerId": "0946E5389C61C6E8E063AF598E0AB303",
        "paymentInstrumentId": "0946E538A054C6E8E063AF598E0AB303",
        "instrumentIdentifierId": "7030390000049956236"
      },
      "responseRecord": {
        "response": "CCH",
        "reason": "800"
      }
    },
    {
      "id": "48496609474",
      "sourceRecord": {
        "token": "0946E514C6BAC699E063AF598E0AE7EE",
        "cardNumber": "400000XXXXXX2087",
        "cardExpiryMonth": "12",
        "cardExpiryYear": "2020",
        "cardType": "001",
        "customerId": "0946E514C6BAC699E063AF598E0AE7EE",
        "paymentInstrumentId": "0946D95F625B9F6DE063AF598E0A36C8",
        "instrumentIdentifierId": "7030070000053862087"
      }
    }
  ]
}

```

```

    },
    "responseRecord": {
      "response": "NED",
      "reason": "800",
      "cardExpiryMonth": "11",
      "cardExpiryYear": "2033"
    }
  },
  {
    "id": "48496609464",
    "sourceRecord": {
      "token": "0946A75C5F0F0E24E063AF598E0AB413",
      "cardNumber": "400000XXXXXX0239",
      "cardExpiryMonth": "12",
      "cardExpiryYear": "2022",
      "cardType": "001",
      "customerId": "0946A75C5F0F0E24E063AF598E0AB413",
      "paymentInstrumentId": "0946E514C386C699E063AF598E0AE7EE",
      "instrumentIdentifierId": "7030120000053460239"
    },
    "responseRecord": {
      "response": "NAN",
      "reason": "800",
      "token": "0946A75C5F0F0E24E063AF598E0AB413",
      "cardNumber": "441224XXXXXX8486",
      "cardType": "001",
      "instrumentIdentifierId": "7030040000052478486",
      "instrumentIdentifierCreated": "true",
      "cardExpiryMonth": "11",
      "cardExpiryYear": "2033"
    }
  }
]
}

```

American Express Daily and Token Harvest Update Reports

American Express update reports are generated daily, so the batch ID is not known in advance.

Similarly, token harvest updates are scheduled by the Account Updater service on a date that you agree upon with your account representative.

For daily and token harvest update reports, the first step is to retrieve the batch ID itself by sending an authenticated GET request, including the header `ACCEPT=application/json`, to one of the following resources:

- Test: <https://apitest.cybersource.com/accountupdater/v1/batches>
- Production: <https://api.cybersource.com/accountupdater/v1/batches>

The response is an array of batches. Paging is supported with offset and limit query parameters. For example, to return the second page of results with 50 per page, send [/v1/batches?offset=1&limit=50](#).

To filter by date, add the **fromDate** and **toDate** fields as a query string using the UTC date format `yyyymmddThhmmssZ`. Example: [v1/batches?fromDate=20200315T000000Z&toDate=20200415T000000Z](#).

HTTP 200: Successful Response

```
{
  "_links": [
    {
      "rel": "self",
      "href":
"https://apitest.cybersource.com/accountupdater/v1/batches?offset=0&limit=1"
    },
    {
      "rel": "first",
      "href":
"https://apitest.cybersource.com/accountupdater/v1/batches?offset=0&limit=1"
    },
    {
      "rel": "next",
      "href":
"https://apitest.cybersource.com/accountupdater/v1/batches?offset=1&limit=1"
    },
    {
      "rel": "last",
      "href":
"https://apitest.cybersource.com/accountupdater/v1/batches?offset=114&limit=1"
    }
  ],
  "object": "collection",
  "offset": 0,
  "limit": 3,
  "count": 1,
  "total": 3,
  "_embedded": {
    "batches": [
      {
        "_links": {
          "reports": [
            {
```

```

        "href":
        "https://apitest.cybersource.com/accountupdater/v1/batches/1541603147941000209921
2314/report"
    }
  ],
  },
  "batchId": "15416031479410002099212314",
  "batchCreatedDate": "2018-11-07T07:05:48Z",
  "batchModifiedDate": "2018-11-07T07:05:50Z",
  "batchSource": "SCHEDULER",
  "tokenSource": "TMS",
  "merchantReference": "Merchant Name",
  "batchCaEndpoints": [
    "VISA",
    "MASTERCARD"
  ],
  "status": "COMPLETE",
  "totals": {
    "acceptedRecords": 1,
    "rejectedRecords": 0,
    "updatedRecords": 1,
    "caResponses": 1,
    "caResponsesOmitted": 0
  }
},
{
  "_links": {
    "reports": [
      {
        "href":
        "https://apitest.cybersource.com/accountupdater/v1/batches/1541602501073000165534
3827/report"
      }
    ]
  },
  },
  "batchId": "15416025010730001655343827",
  "batchCreatedDate": "2018-11-07T06:55:01Z",
  "batchModifiedDate": "2018-11-07T06:56:52Z",
  "batchSource": "AMEX_REGISTRY_API",
  "tokenSource": "TMS",
  "batchCaEndpoints": [
    "AMEX"
  ],
  "status": "COMPLETE"
},
{
  "_links": {
    "reports": [
      {

```

```

        "href":
        "https://apitest.cybersource.com/accountupdater/v1/batches/1541602501073000165534
        3827/report"
    }
  ]
},
"batchId": "15402221273070001683984545",
"batchCreatedDate": "2018-10-22T08:28:47Z",
"batchModifiedDate": "2018-10-22T08:29:19Z",
"batchSource": "AMEX_MAINTENANCE",
"tokenSource": "TMS",
"batchCaEndpoints": [
  "AMEX"
],
"status": "COMPLETE",
"totals": {
  "acceptedRecords": 0
}
}
]
}
}

```

Batches are identified by the **batchCreatedDate** and the **batchSource** field values.

Batch Source Values

batchSource Value	Description
AMEX_REGISTRY_API	Batch for American Express token registration. American Express generates a report only when the registration batch contains errors.
AMEX_MAINTENANCE	Daily updates for tokens enrolled in the American Express Cardrefresher service.
TOKEN_API	Updates relating to a one-time request to Visa or Mastercard.
SCHEDULER	Updates relating to a monthly harvest of all tokens.

After you submit a batch for American Express token registration, you can access the batch status through the authenticated GET request using the URL returned in the response. A successful response returns the status of the batch. For information about registering American Express tokens for daily updates, see [Registering Tokens for American Express Daily Updates \(on page 13\)](#).

American Express Registry Status Response

```
{
  "_links": {
    "self": {
      "href":
        "https://apitest.cybersource.com/accountupdater/v1/batches/15816023535620001646854894/status"
    },
    "report": [
      {
        "href":
          "https://apitest.cybersource.com/accountupdater/v1/batches/15816023535620001646854894/report"
        }
      ]
    },
    "batchCaEndpoints": "AMEX",
    "batchCreatedDate": "2020-02-13T13.59.13Z",
    "batchId": "15816023535620001646854894",
    "batchSource": "AMEX_REGISTRY_API",
    "description": "Updates have been applied to your tokens. A batch report is available.",
    "merchantReference": "Merchant Name",
    "status": "COMPLETE",
    "totals":
      {
```

```

    "acceptedRecords": 999,
    "rejectedRecords": 123,
    "updatedRecords": 0,
    "caResponses": 0,
    "caResponsesOmitted": 0
  }
}

```

Retrieving a Batch with a Batch ID

To access an individual batch report, send an authenticated GET request, including the header `ACCEPT=application/json`, using the URL returned in the batch status or batches resource described in [American Express Daily Harvest Update Reports \(on page 21\)](#).

AMEX_REGISTRY_API Batch Method HTTP 200: Successful Response

```

{
  "version": "1.0",
  "reportCreatedDate": "2018-11-07T15:33:11Z",
  "batchId": "15416047164330001593314231",
  "batchSource": "AMEX_REGISTRY_API",
  "batchCaEndpoints": "AMEX",
  "batchCreatedDate": "2018-11-07T15:31:56Z",
  "merchantReference": "Merchant Name",
  "totals": {
    "acceptedRecords": 0,
    "rejectedRecords": 3
  },
  "records": [
    {
      "sourceRecord": {
        "token": "12345678901234567890",
        "cardExpiryMonth": "01",
        "cardExpiryYear": "2021"
      },
      "responseRecord": {
        "response": "DEC",
        "reason": "852"
      }
    },
    {
      "sourceRecord": {
        "token": "456",
        "cardExpiryMonth": "01",
        "cardExpiryYear": "2021"
      }
    }
  ]
}

```

```

    },
    "responseRecord": {
      "response": "DEC",
      "reason": "851"
    }
  },
  {
    "sourceRecord": {
      "token": "789",
      "cardExpiryMonth": "01",
      "cardExpiryYear": "2021"
    },
    "responseRecord": {
      "response": "DEC",
      "reason": "851"
    }
  }
]
}

```

American Express Daily Updates

Card numbers in TMS are represented by *instrument identifier* tokens. A card number and its associated *instrument identifier* token are set to a CLOSED status under these circumstances:

- The card network sends a direct account closed notification (response code ACL).
- A new card number is issued to replace a cancelled card (response code NAN).

Account Updater updates customer and payment instrument tokens only when you specify them in the request. When you specify a customer token for update or harvest, only the customer's default payment instrument token is updated. When you do not specify the customer and payment instrument tokens, they can become associated with a closed instrument identifier token in the update batch or harvest. These results are detailed in the [additionalUpdates](#) section of the update report. To update customer tokens and payment instrument tokens, include them in a subsequent Account Updater batch API request, or send a direct call to the TMS REST API. See the TMS Developer Guide at <https://developer.cybersource.com/docs/cybs/en-us/tms/developer/ctv/rest/tms/tms-overview.html>.

AMEX_MAINTENANCE Batch Method

```

{
  "version": "1.0",
  "reportCreatedDate": "2020-01-23T11:16:13Z",
  "batchId": "15797780137010000506182090",

```

```

"batchSource": "AMEX_MAINTENANCE",
"batchCaEndpoints": "AMEX",
"batchCreatedDate": "2020-01-23T11:13:33Z",
"totals": {
  "updatedRecords": 3,
  "rejectedRecords": 0,
  "caResponses": 3,
  "caResponsesOmitted": 0
},
"billing": {
  "nan": 1,
  "ned": 1,
  "acl": 1,
  "cch": 0
},
"records": [
  {
    "id": "562239661",
    "sourceRecord": {
      "token": "9CCD3AE24DD9E254E0533F36CF0A356E",
      "cardNumber": "371449XXXXX2009",
      "cardExpiryMonth": "02",
      "cardExpiryYear": "2021",
      "cardType": "003",
      "customerId": "9CCD3AE24DD9E254E0533F36CF0A356E",
      "paymentInstrumentId": "9CCD3AE24DD8E254E0533F36CF0A356E",
      "instrumentIdentifierId": "9CCD3AE24DD7E254E0533F36CF0A356E"
    },
    "responseRecord": {
      "response": "NAN",
      "reason": "800",
      "token": "9CCD3AE24DD9E254E0533F36CF0A356E",
      "cardNumber": "371449XXXXX0102",
      "cardType": "003",
      "instrumentIdentifierId": "9CCDC4D08BE0C16BE0533F36CF0A9916",
      "instrumentIdentifierCreated": "true",
      "cardExpiryMonth": "07",
      "cardExpiryYear": "2021",
      "additionalUpdates": [
        {
          "customerId": "8CCD3AE24DD8E254E0533F36CF0A355E",
          "paymentInstrumentId": "9CCD3AE24DD8E254E0533F36CF0A356D",
          "creator": "aura_regress_tms_report",
          "state": "CLOSED",
          "message": "This Payment Instrument contains the source card number,
which is now closed. If required, you can update manually or through the AU REST
API."
        }
      ]
    }
  ]

```

```

    }
  }
  {
    "id": "562239711",
    "sourceRecord": {
      "token": "9CCD3AE24DF7E254E0533F36CF0A356E",
      "cardNumber": "371449XXXXX1100",
      "cardExpiryMonth": "02",
      "cardExpiryYear": "2021",
      "cardType": "003",
      "customerId": "9CCD3AE24DF7E254E0533F36CF0A356E",
      "paymentInstrumentId": "9CCD3AE24DF6E254E0533F36CF0A356E",
      "instrumentIdentifierId": "9CCD3AE24DF5E254E0533F36CF0A356E"
    },
    "responseRecord": {
      "response": "NED",
      "reason": "800",
      "cardExpiryMonth": "12",
      "cardExpiryYear": "2021"
    }
  },
  {
    "id": "562239751",
    "sourceRecord": {
      "token": "9CCD3AE24E0FE254E0533F36CF0A356E",
      "cardNumber": "371449XXXXX1226",
      "cardExpiryMonth": "02",
      "cardExpiryYear": "2021",
      "cardType": "003",
      "customerId": "9CCD3AE24E0FE254E0533F36CF0A356E",
      "paymentInstrumentId": "9CCD3AE24E0EE254E0533F36CF0A356E",
      "instrumentIdentifierId": "9CCD3AE24E0DE254E0533F36CF0A356E"
    },
    "responseRecord": {
      "response": "ACL",
      "reason": "800",
      "additionalUpdates": [
        {
          "customerId": "7CCD3AE24DD8E254E0533F36CF0A356A",
          "paymentInstrumentId": "9CCD3AE24E0EE254E0533F36CF0A356D",
          "creator": "aura_regress_tms_report",
          "state": "CLOSED",
          "message": "This Payment Instrument contains the source card number,
which is now closed. If required, you can update manually or through the AU REST
API."
        }
      ]
    }
  }
}

```

```
}  
]  
}
```

TOKEN_API Batch Method and Scheduler

```
{
  "version": "1.0",
  "reportCreatedDate": "2018-11-01T14:43:36Z",
  "batchId": "15410833473400000123332450",
  "batchSource": "SCHEDULER",
  "batchCaEndpoints": "VISA,MASTERCARD",
  "batchCreatedDate": "2018-11-01T14:42:27Z",
  "merchantReference": "Merchant Name",
  "totals": {
    "acceptedRecords": 2,
    "caResponses": 3,
    "rejectedRecords": 0,
    "updatedRecords": 2,
    "caResponsesOmitted": 1
  },
  "billing": {
    "nan": 1,
    "ned": 0,
    "acl": 1,
    "cch": 0
  },
  "records": [
    {
      "id": "4451434614",
      "sourceRecord": {
        "token": "4682345889876532701018",
        "cardNumber": "511111XXXXXX3604",
        "cardExpiryMonth": "09",
        "cardExpiryYear": "2021",
        "cardType": "002"
      },
      "responseRecord": {
        "response": "ACL",
        "reason": "800"
      }
    },
    {
      "id": "784311",
      "sourceRecord": {
        "token": "7020000000014008934",
        "cardNumber": "371000XXXXXX8115",
        "cardExpiryMonth": "01",
        "cardExpiryYear": "2021",
        "cardType": "003",
        "instrumentIdentifierId": "7020000000014008115"
      },
      "responseRecord": {
```

```

    "response": "NAN",
    "reason": "800",
    "token": "7020000000012513358",
    "cardNumber": "401000XXXXXX2753",
    "cardType": "001",
    "instrumentIdentifierId": "7020000000012512753",
    "instrumentIdentifierCreated": "true",
    "cardExpiryMonth": "08",
    "cardExpiryYear": "2021"
  }
}
]
}

```

Batch Retrieval Error

```

{
  "_links": {
    "self": {
      "href": "https://api.cybersource.com/accountupdater/v1/batches/154108334734003332450/report"
    }
  },
  "code": "FORBIDDEN_RESPONSE",
  "correlationId": "0386623ab0eb47dfae61d273032f8202",
  "detail": "You are not authorized to access this resource",
  "localizationKey": "cybsapi.forbidden.response",
  "message": "Unauthorized Access"
}

```

PAN Updates



Important: You must enroll in Account Updater and comply with the Terms of Use. See [Terms of Use \(on page 10\)](#).

After the syntax of the request file is validated, Cybersource begins processing the file.

Account Updater files are processed once per day. You can expect your response file to be available 24 to 48 hours after you submit your request file. Cybersource recommends that you send your Account Updater request file 3 to 5 days before your billing cycle starts to ensure that your file completes processing and that you have enough time to update your data store.

Responses from Visa and Mastercard are consolidated and returned in an encrypted response file. See [Response File Records \(on page 41\)](#).

Creating Security Keys

To upload a PAN update request file, you must create two types of security keys: an authentication key and a PGP public/private key pair for encryption.

Authentication Key

You can authenticate using HTTP signature authentication or JSON Web Token (JWT) authentication.

A Base64-encoded shared secret key is passed in the headers you generate for HTTP signature authentication. For instructions on how to create a shared secret key, see [Create a Shared Secret Key for HTTP Signature Authentication](#).

A P12 Certificate is passed in the headers you generate for JWT authentication. For instructions on how to create a P12 Certificate, see [Create a P12 Certificate for JSON Web Token Authentication](#).

PGP Public/Private Key

The PGP public/private key pair is used to protect, by encryption, credit card data contained in the response files. The key pair contains both a public and a private key. You exchange the public part of this key pair with Cybersource, who uses it to encrypt the response files. You maintain the private part of the key pair to decrypt the response file. To create a PGP key pair for encrypting and decrypting credit card data, see [PGP Keys](#).

Formatting a Request File

Account Updater request files must be in CSV format with a maximum file size of 10 MB. The format for a request file consists of these components:

- A header record.
- A detail record with one or more data records, each on a separate line.
- A footer record, which indicates the end of the file.

Request Header Record

The header record consists of comma-separated values and uses the fields listed in the following table:

Header Record Fields

Field Name	Description	Required or Optional	Data Type & Length
Record Identifier	Constant value indicating the record type. Format: H	Required	Alpha (1)
File Classification	Indicates whether this is a request or response file. Format: cybs.au.request.pan	Required	Alpha (30)
merchantID	Your merchant ID. Format: sampleID2	Required	Alphanumeric (30)
batchID	File (batch) identifier that you assign. The batch ID must be unique. If you send a file that contains a previously submitted batch ID, the file is rejected. Format: 12345	Required	Numeric (30)

Header Record Fields (continued)

Field Name	Description	Required or Optional	Data Type & Length
recordCount	The number of detail records in the file. Format: 12345	Required	Numeric
statusEmail	Email address to which status emails for the request are sent. Format: aaa@aaa.aaa	Required	Alphanumeric (100)
creationDate	Optional field that you can pass for reference. If present, it appears in the Business Center Account Updater View Status window. Format: yyyy-MM-DD	Optional	String (10)
Batch Info	Optional field that you can pass for reference. Format: sample12	Optional	Alphanumeric (50)

Request Detail Record

Each file must contain at least one detail record.

Detail Record Fields

Field Name	Description	Required or Optional	Data Type & Length
Record Identifier	Constant value indicating the record type.	Required	Alpha (1)

Detail Record Fields (continued)

Field Name	Description	Required or Optional	Data Type & Length
	Format: D		
Card Number	Card number to process.	Required	Numeric (19)
Card Expiration Month	Expiration month of the card. Format: MM	Required	Alphanumeric (2)
Card Expiration Year	Expiration year of the card. Format: YY	Required	Numeric (2)
Merchant Reference ID	You can use this field to track your Account Updater request records. If this field is populated, the same value is returned in the Account Updater response file. Format: sampleID2	Optional	Alphanumeric (50)
BA Sub Merchant ID	This field is required for billing aggregator merchants only. Format: sampleID2	Optional	Alphanumeric (10)

Request Footer Record

Each file should contain only one footer record.

Footer Record Field

Field Name	Description	Required or Optional	Data Type & Length
Record Identifier	Constant value indicating the record type. Format: F	Required	Alpha (1)

Request File Examples

Non-Billing Aggregator Merchants

```
H,cybs.au.request.pan,merchant1,001,2,notify@yourcompany.com,2019-03-23,My January  
Batch  
D,1111222233334444,11,09,0001  
D,2222333344445555,11,09,0002  
F
```

Billing Aggregator Merchants

```
H,cybs.au.request.pan,merchant1,001,2,notify@yourcompany.com,2019-03-23,My January  
Batch  
D,1111222233334444,11,09,0001,subId01  
D,2222333344445555,11,09,0002,subId02  
F
```

Uploading a Request File



Important: For each PAN you upload, you can receive multiple responses. For example, if you upload one Visa card for an update, you can receive both a Mastercard and Visa response, or two Visa responses.

To upload the request file, use HTTPS. Your client application must support HTTP/1.0 or HTTP/1.1 and TLS 1.2 or later.

To access the Account Updater URL, you must authenticate using either HTTP signature authentication or JWT authentication.

Before you submit files to the production server, test your request files. Follow the instructions in [Testing \(on page 45\)](#).

Use the following URLs for submitting test and live Account Updater request files:

- **Test:**<https://apitest.cybersource.com/accountupdater/v1/batches/upload-csv-card-numbers>
- **Production:**<https://api.cybersource.com/accountupdater/v1/batches/upload-csv-card-numbers>

The request is a POST form data request with a CSV file keyed as a data file.

A successful request results in an HTTP 200 response code. The following error codes are possible:

HTTP Error Codes

Code	Description	Failure Scenarios
400	Bad Request	Malformed request. Payload too large.
401	Unauthorized	Invalid request credentials. Public key matching public key fingerprint not found.
403	Forbidden	Invalid merchant header in uploaded file.
500	Internal Server Error	Message fails to be put on queue for processing.
503	Server Unavailable	Internal database unavailable to retrieve public key.

For more information on submitting a PAN upload request see [Sample AU Application for Uploading a CSV of PANs](#).

Email Notification

After you upload the request file, Account Updater validates the syntax and sends you a confirmation email indicating whether the file passed this stage of validation. You must specify an email address in the **statusEmail** header field in order to receive this confirmation email. If this field is left blank, you will not receive an email confirmation, and you must go to the Business Center to view the status. Account Updater sends the email notification within 30 minutes of receiving the request file. However, actual timing depends on the system load when the file is submitted.

The table below lists possible subject lines of the email notifications.

Email Notifications

Subject Line	Reason	Status Viewable in the Business Center
Received	The Account Updater request file was received. Account Updater processes the requests in the file. No action is required.	Yes
Rejected	The file was rejected. Read the contents of the email and follow the suggested remedy.	No
Validated	The file passed validation.	Yes
Declined	The file did not pass validation checks. All records are declined. Read the contents of the email and follow the suggested remedy.	Yes
Processing	The request file is being processed by Account Updater.	Yes
Completed	The response file has been generated and is ready for download.	Yes

Related Information

- [Viewing the Status of a Batch File \(on page 40\)](#)

Viewing the Status of a Batch File

1. Log in to the Business Center:
 - Live: <https://businesscenter.cybersource.com>
 - Test: <https://businesscentertest.cybersource.com>
2. On the left navigation pane, click the **Tools** icon.
3. Click **Account Updater**. The Account Updater page appears.
4. Click **Add filter**. The New Filter selection box appears.
5. Choose **Batch ID** or **Date Range Options**.
 - For Batch ID, enter the specific batch ID.
 - For Date Range Options, choose a time range. If you choose Custom, you can specify the start date, start time, end date, and end time.
6. The Search Results list displays matching results.

Downloading a Response File

You can download response files with a status of *Complete* from the Business Center or with a client application. To download it programmatically, see [Secure File Share API](#) in the Developer Center.

1. Log in to the Business Center:
 - Live <https://businesscenter.cybersource.com>
 - Test: <https://businesscentertest.cybersource.com>
2. On the left navigation pane, click the **Reports** icon.
3. Under Downloadable Reports, click **Available Reports**. The Available Reports page appears.
4. Click the Third-Party Reports tab to view Account Updater CSV response files.
5. In the Download column, click the file format link.

Response File Records

The response file is encrypted with the public part of the PGP key that you generated and uploaded to Account Updater. To read a response file, you must decrypt it using the private part of the PGP key pair. You can do so with the same third-party software that you used to create the keys.

The format for a response file consists of these components:

- A header record.
- A detail record with one or more data records, each on a separate line.
- A footer record, which indicates the end of the file.

Response Header Record

The header record consists of comma-separated values and uses the fields listed in the following table:

Header Record Fields

Field Name	Description	Data Type & Length
Record Identifier	Constant value indicating the record type. Format: H	Alpha (1)
File Classification	Indicates whether this is a request or response file, and the type of service. Format: cybs.au.response.pan	Alphanumeric (30)
MerchantID	Your merchant ID.	Alphanumeric (30)
BatchID	File (batch) identifier sent in the request file.	Numeric (30)

Response Detail Record

Each file contains at least one detail record.

Response Detail Record Fields

Field Name	Description	Data Type & Length
Record Identifier	Constant value indicating the record type. Format: D	Alpha (1)
Request ID	Unique identifier for the record.	Numeric (30)
Old Card Number	Old card number.	Numeric (19)
Old Card Expiration Month	Old expiration month. Format: MM	Numeric (2)
Old Card Expiration Year	Old expiration year. Format: YY	Numeric (2)
New Card Number	New card number.	Numeric (19)
New Card Expiration Month	New expiration month. Format: MM	Numeric (2)
New Card Expiration Year	New expiration year. Format: YY	Numeric (2)
Merchant Reference ID	This field is optional and is returned in the response if present in the request file.	Alphanumeric (50)
BA Sub Merchant ID	This field is returned in the response if sent in the request file.	Alphanumeric (10)

Response Detail Record Fields (continued)

Field Name	Description	Data Type & Length
Response Code	Response code for the record.	Alpha (3)
Reason Code	Reason code for the record.	Numeric (3)

Related Information

- [PAN Upload Response Codes and Reason Codes \(on page 61\)](#)

Response Footer Record

Each file contains only one footer record.

Footer Record Fields

Field Name	Description	Data Type & Length
Record Identifier	Constant value indicating the record type. Format: F	Alpha (1)
Record Count	The number of detail records in the file.	Numeric (10)
Response Code	Response code for the file.	Alpha (3)
Reason Code	Reason code for the file.	Numeric (3)

Related Information

- [PAN Upload Response Codes and Reason Codes \(on page 61\)](#)

Response File Examples

Example: Non-Billing Aggregator Response File

```
H,cybs.au.response.pan,merchant1,001
D,10000000000000000001,1111222233334444,11,09,,,,,0001,,NUP,800
D,10000000000000000002,2222333344445555,11,09,6666777788889999,11,11,0002,,NAN,800
F,2,COM,800
```

Example: Billing Aggregator Response File

```
H,cybs.au.response.pan,merchant1,001
D,10000000000000000001,1111222233334444,11,09,,,,,0001,subId01,NUP,800
D,10000000000000000002,2222333344445555,11,09,6666777788889999,11,11,0002,subId02,NAN,800
F,2,COM,800
```

Testing

The Account Updater test environment provides a simulator in which the response from the card association can be triggered using test card numbers.

This simulator ensures that you can handle the possible response combinations when connecting to multiple card associations.

The test environment typically completes the process in a matter of minutes rather than the 24-hour (or longer) duration of the live environment when updates are sent to the actual card associations.

American Express Test Card Numbers

American Express card updates through the Cardrefresher service are available only when you are using TMS. Use the numbers listed in the following tables to simulate various scenarios. Replace the BIN with [371449](#) and remove spaces when sending to Account Updater.

For response code descriptions, see [PAN Upload Response Codes and Reason Codes \(on page 61\)](#).

American Express Test Numbers

Card Number	Response Code
BIN 0 0002 0115	NAN (No New Expiry Date)
BIN 1 0211 2216	NAN (No New Expiry Date)
BIN 2 0121 2206	NAN (No New Expiry Date)
BIN 1 0021 1119	NAN (No New Expiry Date)
BIN 1 0101 0023	NAN (No New Expiry Date)
BIN 0 0100 2112	NAN (New Expiry Date)
BIN 1 2101 2009	NAN (New Expiry Date)
BIN 0 2201 2009	NAN (New Expiry Date)
BIN 2 1000 0113	NAN (New Expiry Date)
BIN 0 2100 0229	NAN (New Expiry Date)
BIN 2 2210 0224	NED
BIN 0 0112 0203	NED
BIN 0 2102 1100	NED
BIN 2 0121 2107	NED

American Express Test Numbers (continued)

Card Number	Response Code
BIN 0 1121 0119	NED
BIN 0 1022 1109	ACL
BIN 1 0112 1226	ACL
BIN 2 0201 0005	ACL
BIN 1 2121 0207	ACL
BIN 0 1012 2109	ACL
BIN 1 2120 0224	DEC
BIN 2 1010 0020	861 (Attempt to enroll. Customer already enrolled.)
BIN 1 1202 1118	862 (Registry rejected due to card member opt out.)
BIN 1 0122 0218	ERR
BIN 0 0010 2004	ERR
BIN 1 1111 2108	861
BIN 1 4254 6639	NAN (No New Account Number, No New Expiry Date)
BIN 6 7595 0950	NAN (No New Account Number, New Expiry Date)

Related Information

- [PAN Upload Response Codes and Reason Codes \(on page 61\)](#)

Mastercard Test Card Numbers

The bold fields represent the token updates for TMS, Recurring Billing, and Payment Tokenization merchants using the REST API batch update and harvest update. Replace the BIN with **511111** and remove spaces when sending to Account Updater.

Mastercard Card Test Numbers

Card Number	Response Codes
BIN 10 4714 3086	Visa Response: NAN Mastercard Response: NAN

Mastercard Card Test Numbers (continued)

Card Number	Response Codes
BIN 10 2999 7178	Visa Response: ACL Mastercard Response: NAN
BIN 10 1548 6814	Visa Response: CUR Mastercard Response: NAN
BIN 10 5459 2548	Visa Response: NUP Mastercard Response: NAN
BIN 10 4871 8571	Visa Response: CCH Mastercard Response: NAN
BIN 10 5798 7356	Visa Response: NAN Mastercard Response: NED
BIN 10 7450 2964	Visa Response: ACL Mastercard Response: NED
BIN 10 6971 3154	Visa Response: CUR Mastercard Response: NED
BIN 10 2030 4416	Visa Response: NUP Mastercard Response: NED
BIN 10 4733 5823	Visa Response: CCH Mastercard Response: NED
BIN 10 3135 3600	Visa Response: NAN Mastercard Response: ACL
BIN 10 4816 3604	Visa Response: ACL Mastercard Response: ACL
BIN 10 1867 3020	Visa Response: CUR Mastercard Response: ACL
BIN 10 3056 0627	Visa Response: NUP Mastercard Response: ACL
BIN 10 0270 8865	Visa Response: CCH Mastercard Response: ACL

Mastercard Card Test Numbers (continued)

Card Number	Response Codes
BIN 10 6646 9396	Visa Response: NAN Mastercard Response: CUR
BIN 10 5787 1816	Visa Response: ACL Mastercard Response: CUR
BIN 10 7350 8855	Visa Response: CCH Mastercard Response: CUR

Related Information

- [PAN Upload Response Codes and Reason Codes \(on page 61\)](#)

Visa Test Card Numbers

The bold response codes represent the token updates for TMS, Recurring Billing, and Payment Tokenization merchants using the REST API batch update and harvest update. Replace the BIN with [400000](#) and remove spaces when sending to Account Updater.

Visa Card Test Numbers

Card Number	Response Codes
BIN 71 0951 9220	Visa Response: NAN Mastercard Response: NAN
BIN 15 3919 2096	Visa Response: NAN Mastercard Response: ACL
BIN 18 6481 0239	Visa Response: NAN Mastercard Response: CUR
BIN 91 9582 8465	Visa Response: NED Mastercard Response: NAN
BIN 27 5765 7455	Visa Response: NED Mastercard Response: ACL

Visa Card Test Numbers (continued)

Card Number	Response Codes
BIN 71 1311 2087	Visa Response: NED Mastercard Response: CUR
BIN 21 1752 4874	Visa Response: ACL Mastercard Response: NAN
BIN 71 1629 4650	Visa Response: ACL Mastercard Response: ACL
BIN 20 5548 7183	Visa Response: ACL Mastercard Response: CUR
BIN 52 8063 4792	Visa Response: CUR Mastercard Response: NAN
BIN 24 0631 2635	Visa Response: CUR Mastercard Response: ACL
BIN 89 2339 9344	Visa Response: CUR Mastercard Response: CUR
BIN 55 7908 8940	Visa Response: NUP Mastercard Response: NAN
BIN 57 9875 5634	Visa Response: NUP Mastercard Response: ACL
BIN 80 9110 0706	Visa Response: CCH Mastercard Response: NAN
BIN 26 9567 5155	Visa Response: CCH Mastercard Response: ACL
BIN 35 8627 6236	Visa Response: CCH Mastercard Response: CUR

Related Information

- [PAN Upload Response Codes and Reason Codes \(on page 61\)](#)

API Fields

The following fields can be used with Account Updater.

Request Fields

Request Fields

Field Name	Description	Used By & Required (R) /Optional (O)	Validation
notificationEmail	Email address to which batch status updates are sent.	POSTs to /batches (R)	Valid email address
merchantReference	Your reference to identify the batch.	POSTs to /batches (O)	0 to 255 characters
type	Indicates whether batch is a one-time update for Visa and/or Mastercard, or an enrollment in American Express Cardrefresher. Possible values: • <code>oneOff</code> (default): Visa or Mastercard • <code>amexRegistration</code>: American Express	POSTs to /batches (O)	
included	Elements to be included. Must include one of the following: • <code>tokens</code> • <code>instrument_identifier</code>	POSTs to /batches (R)	

Request Fields (continued)

Field Name	Description	Used By & Required (R) /Optional (O)	Validation
tokens	Comma-separated list of subscription IDs, Token Management Service customer or payment instrument tokens.	POSTs to /batches (O)	If the array is present, then it should not be empty (min length = 1) or contain null values. Maximum number of tokens is 10 million.
instrumentIdentifiers	Token Management Service instrument identifier token assigned to the tokenized PAN and its associated expiration dates.	POSTs to /batches (O)	
id	ID for the instrument identifier token.	POSTs to /batches (R)	String (32)
expirationMonth	Two-digit month in which the card expires.	POSTs to /batches (R)	String (2)
expirationYear	Four-digit year in which the card expires.	POSTs to /batches (R)	String (4)

Response Fields

Response Fields

Field Name	Description	Returned By	Data Type & Length
batchId	When the request is successful, a batch ID is returned to the user.	/batches /.../status /.../report	Alphanumeric (26)
batchItemCount	When the request is successful, this value is the number of items that were included in the request. When the request is unsuccessful, the value of this field is 0.	/batches /.../status /.../report	Numeric (9)
_links	JSON object containing link elements relating to the request. Successful requests return the URI of the batch status.	/batches /.../status /.../report	
self	The resource address that was requested. Element within _links .	/batches /.../status /.../report	URL
first	First page in the result set.	/batches	URL
next	Next page in the result set.	/batches	URL
last	Last page in the result set.	/batches	URL
status	URI of the batch status resource.	/batches	URL

Response Fields (continued)

Field Name	Description	Returned By	Data Type & Length
	 Important: Do not hard-code the link to the batch status resource. Use the returned value to avoid errors if the URI structure changes.		
reports	URI of the batch associated with the batchId .	/batches /status	URL
correlationId	Returned when an error occurs. Provide this ID to Customer Support to help identify your transaction.	/batches /.../status /.../report	String (36)
code	HTTP Response code. Returned when an error occurs.	/batches /.../status /.../report	String (3)
detail	Returned when an error occurs. Detailed description of the error.	/batches /.../status /.../report	String (1024)
fields	Returned when an error occurs. The array contains elements that describe the erroneous fields.	/batches	

Response Fields (continued)

Field Name	Description	Returned By	Data Type & Length
path	Returned when an error occurs. Element within the fields. Path of field name.	/batches	String (36)
message	Returned when an error occurs. This is a plain text error message and can be an element within the fields. This field can also appear with the fields JSON object.	/batches	String (256)
localizationKey	Returned when an error occurs. A unique key that represents the error message and can be an element within fields. This field can also appear with the JSON object.	/batches	String (128)
version	Version of the report. For example, v1.4-1 is the major version of the API used to create the batch and 4 is the minor version of the report. You always receive the latest minor version of the report for the API you used to create the batch.	/report	
batchSource	Method used to create the batch. For example, TOKEN_API .	/batches /status /report	TOKEN_API SCHEDULER AMEX_REGISTRY AMEX_MAINTENANCE
batchCaEndpoints	Card associations to which the card numbers were sent.	/batches /status	Array containing one or more of the following:

Response Fields (continued)

Field Name	Description	Returned By	Data Type & Length
		/report	- VISA - MASTERCARD - batchCaEndpoints AMEX
batchCreatedDate	Date on which the batch was created.	/batches /status /report	ISO_8601 UTC date
reportCreatedDate	Date on which the report was created.	/batches /status /report	ISO_8601 UTC date
merchantReference	Your reference, if present in the request.	/batches /status /report	0 to 255 characters
totals	JSON object containing the high-level summary of the batch.	/batches /status /report	
acceptedRecords	Number of tokens that were identified and retrieved for the merchant ID.	/batches /status /report	String (9)

Response Fields (continued)

Field Name	Description	Returned By	Data Type & Length
rejectedRecords	Number of tokens that were not identified and retrieved.	/batches /status /report	String (9)
updatedRecords	Number of updates that were applied to a token.	/batches /status /report	String (9)
caResponses	Number of updates that were received from the card associations. This value represents updates that may have or have not been applied to a token.	/batches /status /report	String (9)
caResponsesOmitted	Number of updates that were not applied to a token. For example, a response is returned by more than one card association.	/batches /status /report	String (9)
billing	JSON object containing the billing summary information.	/status /report	
nan/ned/act/cch	Number of each billed response type.	/status /report	String (3)

Response Fields (continued)

Field Name	Description	Returned By	Data Type & Length
records	JSON object containing additional objects that relate to the original tokens and the updates or errors that occurred.	/report	
id	ID for the record.	/report	
sourceRecord	JSON object containing details from the source token.	/report	
token	Subscription ID included in the request.	/report	
cardNumber	Masked card number before an update. First six digits and the last four digits are not masked.	/report	
cardExpiryMonth	Two-digit month in which the card expires.	/report	
cardExpiryYear	Four-digit year in which the card expires.	/report	
cardType	Type of card. Possible values: • 001: Visa • 002: Mastercard • 003: American Express	/report	String (3)
customerId	Value of the customer token assigned to the tokenized shipping information and merchant defined data.	/report	

Response Fields (continued)

Field Name	Description	Returned By	Data Type & Length
	This field is for TMS merchants only.		
paymentInstrumentId	The value of the payment instrument token assigned to the tokenized billing information and card expiration dates. This field is for TMS merchants only.	/report	
instrumentIdentifierId	Value of the instrument identifier token assigned to the tokenized PAN. This field is for TMS merchants only.	/report	String (32)
responseRecord	JSON object containing the details that were made to the token.	/report	
response	Type of response.	/report	
reason	Reason code for the response.	/report	
token	If last-four-digit format-preserving tokens are used, a new token (subscription ID) can be returned that replaces the source record token.	/report	
cardNumber	Masked card number. First six and last four digits are not masked.	/report	
cardExpiryMonth	Two-digit month in which the card expires.	/report	
cardExpiryYear	Four-digit year in which the card expires.	/report	

Response Fields (continued)

Field Name	Description	Returned By	Data Type & Length
cardType	Type of card. Possible values: • 001: Visa • 002: Mastercard • 003: American Express	/report	String (3)
instrumentIdentifierId	Value of the instrument identifier token assigned to the updated tokenized PAN. This field is for TMS merchants only.	/report	String (32)
instrumentIdentifierIdCreated	Indicates whether this is the first time the PAN has been tokenized for you. Possible values: • true • false	/report	String (5)
additionalUpdates	Details associated with a closed instrument identifier token.	/report	
customerId	Value of the customer token not present in the batch and associated with a closed instrument identifier token.	/report	String
paymentInstrumentId	Value of the payment instrument token not present in the batch and associated with a closed instrument identifier token.	/report	String
creator	Merchant that created the payment instrument token	/report	String

Response Fields (continued)

Field Name	Description	Returned By	Data Type & Length
state	State of the token.	/report	"CLOSED"
message	Information about the tokens.	/report	String

Related Information

- [PAN Upload Response Codes and Reason Codes \(on page 61\)](#)

PAN Upload Response Codes and Reason Codes

Record Level

The response code and the reason code for the record appear in the details record of the request file.

Example: Details Record

```
D,10000000000000000002,2222333344445555,11,09,6666777788889999,11,11,0002,,NAN,800
```

Record Response Codes and Reason Codes

Response Code	Response Code Description	Reason Code	Reason Code Description	Billable or Non-Billable Code
ACL	Match: account closed. The state of the Instrument Identifier token gets set to Closed.	800	Success.	Billable.
CCH	Contact card holder.	800	Success.	Billable.
CUR	Card data current.	800	Success.	Non-billable.
DEC	—	801	Invalid card number.	Non-billable.
DEC	—	802	Invalid check digit.	Non-billable.
DEC	—	803	Invalid expiration date.	Non-billable.

Record Response Codes and Reason Codes (continued)

Response Code	Response Code Description	Reason Code	Reason Code Description	Billable or Non-Billable Code
DEC	—	804	Unsupported card type.	Non-billable.
DEC	—	805	Invalid card type length.	Non-billable.
DEC	—	806	Unknown card type.	Non-billable.
DEC	—	810	Invalid BA sub merchant ID.	Non-billable.
DEC	—	850	Invalid token format.	Non-billable.
DEC	—	851	Invalid token length.	Non-billable.
DEC	—	852	Unknown token. This token does not exist, is not associated with your account, or might be superseded.	Non-billable.
DEC	—	853	Invalid token status.	Non-billable.

Record Response Codes and Reason Codes (continued)

Response Code	Response Code Description	Reason Code	Reason Code Description	Billable or Non-Billable Code
			This token has a status of CLOSED from a previous Account Updater batch.	
DEC	—	861	Cardholder is already enrolled or cannot cancel cardholder that is not enrolled.	Non-billable.
DEC	—	862	Rejected because cardholder opted out.	Non-billable.
ERR	—	801	Invalid card number.	Non-billable.
ERR	—	802	Invalid check digit.	Non-billable.
ERR	—	803	Invalid expiration date.	Non-billable.
ERR	—	804	Unsupported card type or cancelled card.	Non-billable.

Record Response Codes and Reason Codes (continued)

Response Code	Response Code Description	Reason Code	Reason Code Description	Billable or Non-Billable Code
ERR	—	807	Merchant not enrolled properly in Account Updater.	Non-billable.
ERR	—	808	Incorrect record indicator.	Non-billable.
ERR	—	809	Unknown error code received during processing.	Non-billable.
ERR	—	811	New account number failed MOD-10 check.	Non-billable.
NAN	New account number. It might also include a new expiration date.	800	Success.	Billable.
NED	New expiration date.	800	Success.	Billable.
NUP	No match, no update.	800	Success.	Non-billable.
UNA	Inconsistent update received, not applicable.	800	Inconsistent update received, not applicable.	Non-billable.

Request File Level

The response code and the reason code for the request file appear in the footer record of the request file.

Example: Footer Record

F, 2, COM, 800

Request File Response Codes and Reason Codes

Response Code	Response Code Description	Reason Code	Reason Code Description
COM	The merchant request file has been validated, processed, and the response received.	800	Success.
DEC	The merchant request file was not processed because each record failed record-level validation.	801	All records within the request file failed record-level validation.