Authorize.Net®
a CyberSource solution

Advanced Integration Method (AIM)

Developer Guide

Card Not Present Transactions

Authorize.Net Developer Support

http://developer.authorize.net

Authorize.Net LLC 082007 Ver.2.0

Authorize.Net LLC (“Authorize.Net”) has made efforts to ensure the accuracy and completeness of the information in this document. However, Authorize.Net disclaims all representations, warranties and conditions, whether express or implied, arising by statute, operation of law, usage of trade, course of dealing or otherwise, with respect to the information contained herein. Authorize.Net assumes no liability to any party for any loss or damage, whether direct, indirect, incidental, consequential, special or exemplary, with respect to (a) the information; and/or (b) the evaluation, application or use of any product or service described herein.

Authorize.Net disclaims any and all representation that its products or services infringe upon any existing or future intellectual property rights. Authorize.Net owns and retains all right, title and interest in and to the Authorize.Net intellectual property, including without limitation, its patents, marks, copyrights and technology associated with the Authorize.Net services. No title or ownership of any of the foregoing is granted or otherwise transferred hereunder. Authorize.Net reserves the right to make changes to any information herein without further notice.

Authorize.Net Trademarks:

Advanced Fraud Detection Suite™

Authorize.Net®

Authorize.Net Your Gateway to IP Transactions™

Authorize.Net Verified Merchant Seal™
Authorize.Net Where the World Transacts®

Automated Recurring Billing™

eCheck.Net®

FraudScreen.Net®

Authorize.Net®
a CyberSource solution

Last revised: 5/27/2011

Table of Contents

Revision History

PUBLISH DATE	UPDATES
August 2007	Release of Ver 2.0 Advanced Integration Method (AIM) Developer Guide
May 2008	Remove SecureSource requirements and various updates
March 2009	Addition of error codes 315-319
July 2009	Clarify use of x_recurring_billing, x_version Added warning for Merchant Defined Fields Additions to the Response Code list
September 2009	Modified test transactions in live mode to allow for zero authorizations for Visa.
November 2009	Updated table of reason-response codes
December 2009	Updated table of reason-response codes
June 2010	Partial authorization changes
January 2011	Special field for CardWorks (x_merchant_descriptor)
May 2011	Corrected example for x_tax format Correct references to AVS/CCV

Section 1
Introduction

Welcome to the Authorize.Net Advanced Integration Method (AIM) Developer Guide. This guide describes the Web development required to connect an e-commerce Web site or other application to the Authorize.Net Payment Gateway in order to submit credit card transactions for authorization and settlement using AIM.

AIM is a customizable payment processing solution that gives the merchant control over all the steps in processing a transaction, including:

- Collection of customer payment information through a custom application
- Generation of a receipt to the customer
- Secure transmission to the payment gateway for transaction processing
- Secure storage of cardholder information
- And more, depending on the merchant's business requirements

The security of an AIM transaction is assured through a 128-bit Secure Sockets Layer (SSL) connection between the merchant's Web server and the Authorize.Net Payment Gateway.

AIM is an ideal integration solution because it allows merchants the highest degree of customization and control over their customers' checkout experience.

Note: For merchants who prefer a payment solution that handles the collection, transmission and storage of cardholder data, Authorize.Net recommends the Server Integration Method (SIM). The *SIM Developer Guide* can be found at http://www.authorize.net/support/SIM_guide.pdf.

With SIM, merchants never have to collect, transmit, or store sensitive cardholder information. Additionally, SIM does not require merchants to purchase and install a Secure Sockets Layer (SSL) digital certificate. This removes the complexity of securely handling and storing cardholder information, simplifying compliance with the Payment Card Industry (PCI) Data Security Standard.

AIM Minimum Requirements

Before you begin, check with the merchant to make sure that the following AIM requirements have already been met. It is strongly recommended that you work closely with the merchant to ensure that any other business and Web site requirements (for example, bank or processor requirements, Web site design preferences) are included in their AIM integration.

- The merchant must have a U.S. based merchant bank account that allows Internet transactions.

- The merchant must have an e-commerce (Card Not Present) Authorize.Net Payment Gateway account.
- The merchant must have a valid Secure Sockets Layer (SSL) certificate and their Web site must be capable of initiating both client and server side SSL connections.
- The merchant's Web site must have server-side scripting or CGI capabilities such as ASP Classic, C#, Cold Fusion, Java, Perl, PHP or VB.Net.
- The merchant must be able to store payment gateway account data securely (for example, API Login ID, or Transaction Key).

Payment Card Industry (PCI) Data Security Standard

IMPORTANT: AIM involves the transmission of sensitive cardholder data by means of the merchant's Web server. Because of this, if the merchant stores cardholder information, it must be stored securely and in accordance with the Payment Card Industry (PCI) Data Security Standard. For more information about PCI and other industry standard processing practices, please see the Developer Security Best Practices White Paper for more information.

If the merchant needs a solution that handles the collection, transmission and storage of cardholder data, they should use the Server Implementation Method (SIM). For more information about SIM, please see the SIM Developer Guide.

Managing Integration Settings

When integrating to the payment gateway, you should be aware that most settings for a merchant's integration can be configured and managed in two ways:

- Included in the transaction request on a per-transaction basis using the application programming interface (API), (as described in this guide), OR
- Configured in the Merchant Interface and applied to all transactions.

IMPORTANT: The Merchant Interface at https://account.authorize.net is a secure Web site where merchants can manage their payment gateway account settings, including their Web site integration settings. It is recommended that you review the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/ for information on managing the merchant's payment gateway integration using the Merchant Interface.

Transaction settings submitted in the transaction request will override transaction settings configured in the Merchant Interface. However, **please be aware that some integration settings must be configured in the Merchant Interface**. To help the merchant maintain a robust integration, it is recommended that you review the integration settings that can be configured in the Merchant Interface with the merchant and determine which integration settings can be posted on a per-transaction basis and which should be configured in the Merchant Interface. See the "Appendix A Fields by Transaction Type" section of this document for a list of fields the payment gateway recommends be submitted on a per-transaction basis.

Features of AIM

In addition to basic transaction processing, AIM provides merchants with several features for configuring transaction security options and further customizing their customers' checkout experience. These features are listed in the AIM Feature Selection Guide provided below. Please take a few moments to discuss these with your merchant and select which features they would like to include in their integration.

FEATURE	**DESCRIPTION**	**REQUIREMENTS**
Address Verification Service (AVS) Filter	This feature allows merchants to compare the billing address submitted by the customer for the transaction with the address on file at the card issuing bank. Filter settings in the Merchant Interface allow the merchant to reject transactions based on the AVS response received.	To implement AVS, the merchant must require the Address and ZIP Code fields on their custom payment form. For more information about AVS, please see the Merchant Integration Guide at http://www.authorize.net/support/merchant/.
Card Code Verification (CCV) Filter	This feature allows merchants to compare the card code submitted by the customer for the transaction with the card code on file at the card issuing bank. Filter settings in the Merchant Interface allow the merchant to reject transactions based on the CCV response received.	To implement CCV, the merchant must require the Card Code on their custom payment form. For more information about CCV, please see the Merchant Integration Guide at http://www.authorize.net/support/merchant/.
Itemized Order Information	This feature allows merchants to submit details for items purchased. This information is included in the customer's confirmation email, in the Transaction Details for the transaction, and in QuickBooks download reports in the Merchant Interface.	To implement Itemized Order Information, the line item field must be submitted on a per-transaction basis. Please see the "Itemized Order Information" section of this document for details.
Email Receipt	This feature allows merchants to have the payment gateway send an automatic email receipt to their customers.	To configure the payment gateway email receipt, the merchant must require the customer email address on their custom payment form, and settings must be configured in the Email Receipts section of the Settings menu in the Merchant Interface or submitted on a per-transaction basis. Please see the "Receipt Options" section of this document for details.

eCheck.Net®

In addition to processing credit card transactions, the payment gateway also supports electronic check transactions with our exclusive eCheck.Net® product. Please contact the merchant to determine whether eCheck.Net is enabled for their payment gateway account or if they would like to sign up. **If eCheck.Net is enabled, you need to ensure that the merchant's Web site integration supports all eCheck.Net field requirements.** Please see the eCheck.Net Developer Guide for more information.

Developer Support

There are several resources available to help you successfully integrate a merchant Web site or other application to the Authorize.Net Payment Gateway.

The Developer Center at http://developer.authorize.net provides test accounts, sample code, SDKs, FAQs, and troubleshooting tools.

If you can't find what you need in the Developer Center, our Integration Team is available to answer your questions by email at developer@authorize.net. (Our Integration Team can only assist with support requests specifically about the Authorize.Net application programming interface (API) and/or services.)

Be sure to read our Developer Security Best Practices White Paper for information on how to maximize the security and reliability of your merchant integration solutions.

If you have any suggestions about how we can improve or correct this guide, please email documentation@authorize.net.

Software Development Kits

Authorize.Net offers software development kits (SDKs) that present an alternate object-oriented model, in several popular languages. To use these SDKs, the merchant's transaction version must be set to 3.1. The SDK performs the core payment activities (such as error handling and parsing, network communication, and data encoding) behind the scenes.

The SDK provides utility methods to help developers build payment flows for each of the integration methods. You can download the SDKs at http://developer.authorize.net/downloads/.

Section 2
Submitting Transactions

The payment gateway supports several credit card transaction types for transactions submitted by AIM.

To implement AIM for a merchant's Web site or proprietary business application, you will need to develop an application that performs the following:

- Securely obtains all of the information required to process a transaction (including data requirements specified by the merchant).
- Initiates a secure SSL connection from the merchant's Web server to the payment gateway transaction post location to pass transaction data in name/value pairs.
- Receives and parses the transaction response from the payment gateway and displays the results to the customer.

There are two options for developing the application:

- You can develop a custom application yourself using the information provided in this document, OR
- You can use Authorize.Net sample code available for free from our Developer Center.

If you choose to use sample code, please be aware that to achieve a successful implementation it **must** be modified with the merchant's specific payment gateway account information. Be sure to carefully review the readme.txt files and comments included in each file of sample code in order to achieve a faster, successful integration.

Developer test accounts with API Login IDs and Transaction Keys are also available for testing your integration methods to the Authorize.Net Payment Gateway at http://developer.authorize.net/testaccount.

Minimum Requirements

The following table represents the minimum fields required for submitting a credit card transaction request to the payment gateway using AIM. The data fields are name/value pairs with the following syntax:

```
x_name_of_field=value of field&.
```

FIELD NAME	VALUE	FORMAT	NOTES
x_login	The merchant's unique API Login ID	Up to 20 characters	The merchant API Login ID is provided in the Merchant Interface

Last revised: 5/27/2011

FIELD NAME	VALUE	FORMAT	NOTES
			and must be stored securely. The API Login ID and Transaction Key together provide the merchant authentication required for access to the payment gateway. See the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/ for more information.
x_tran_key	The merchant's unique Transaction Key	16 characters	The merchant Transaction Key is provided in the Merchant Interface and must be stored securely. The API Login ID and Transaction Key together provide the merchant authentication required for access to the payment gateway. See the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/ for more information.
x_type	The type of credit card transaction	AUTH_CAPTURE (default), AUTH_ONLY, CAPTURE_ONLY, CREDIT, PRIOR_ AUTH_CAPTURE, VOID	If the value submitted does not match a supported value, the transaction is rejected. If this field is not submitted or the value is blank, the payment gateway will process the transaction as an AUTH_CAPTURE.
x_amount	The amount of the transaction	Up to 15 digits with a decimal point (no dollar symbol) Ex. 8.95	This is the total amount and must *include* tax, shipping, and any other charges.
x_card_num	The customer's credit card number	Between 13 and 16 digits without spaces When x_type=CREDIT, only the last four digits are required.	This is sensitive cardholder information and must be stored securely and in accordance with the Payment Card Industry (PCI) Data Security Standard. For more information about PCI, please see the Developer Security Best Practices White Paper.
x_exp_date	The customer's credit card expiration date	MMYY, MM/YY, MM-YY, MMYYYY, MM/YYYY, MM-	This is sensitive cardholder information and must be stored securely and in accordance with the Payment Card Industry (PCI)

FIELD NAME	VALUE	FORMAT	NOTES
		YYYY	Data Security Standard. For more information about PCI, please see the Developer Security Best Practices White Paper.
x_trans_id	The payment gateway assigned transaction ID of an original transaction		Required only for CREDIT, PRIOR_AUTH_CAPTURE, and VOID transactions. Do not include this field if you are providing x_split_tender_id. For more information about transaction types, see the "Credit Card Transaction Types" section of this document.
x_split_tender_id	The payment gateway-assigned ID that links the current authorization request to the original authorization request.	Numeric	This value applies to partial authorization transactions only, and is returned in the reply message from the original authorization request. For more information, see Partial Authorization Transactions on page 15.
x_auth_code	The authorization code of an original transaction *not* authorized on the payment gateway	6 characters	Required only for CAPTURE_ONLY transactions. See the "Credit Card Transaction Types" section below.
x_relay_response	FALSE	FALSE, F, NO, N, 0	SIM applications use relay response. Set this to false if you are using AIM.
x_delim_data	Indicates whether a delimited transaction response is required	TRUE, FALSE, T, F, YES, NO, Y, N, 1, 0	A value of TRUE indicates a request for delimited response from the payment gateway. Since all AIM transactions are direct response, a value of TRUE is required. It is recommended that you submit this field on a per-transaction basis to be sure that transaction responses are returned in the correct format. See Transaction Response below for more about delimited response

Credit Card Transaction Types

This section describes the credit card transaction types supported by the payment gateway and their specific field requirements. It's a good idea to talk to your merchant about how their business plans to submit transaction so that you can properly integrate their payment gateway account to support their business processes.

For example, are they submitting transactions mainly through an e-commerce Web site? Do they need to integrate a custom application to allow call center representatives to enter mail order/telephone order (MOTO) transactions? Would they like the ability to verify the availability of funds on a customer's credit card account at the time of purchase and then charge the credit card at the time they ship the order?

The payment gateway supports the following credit card transaction types.

Note: Some of the field requirements listed in this section for each credit card transaction type are in addition to the minimum field requirements already set forth above for ALL transactions submitted to the payment gateway. For a list of all fields that are required for each credit card transaction type, please see the "Appendix A Fields by Transaction Type" section of this document.

Authorization and Capture

This is the most common type of credit card transaction and is the default payment gateway transaction type. The amount is sent for authorization, and if approved, is automatically submitted for settlement.

The unique field requirement for an Authorization and Capture is:

```
x_type=AUTH_CAPTURE
```

Authorization Only

This transaction type is sent for authorization only. The transaction will not be sent for settlement until the credit card transaction type Prior Authorization and Capture (see definition below) is submitted, or the transaction is submitted for capture manually in the Merchant Interface. For more information about capturing Authorization Only transactions in the Merchant Interface, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.

If action for the Authorization Only transaction is not taken on the payment gateway within 30 days, the authorization expires and is no longer available for capture. A new Authorization Only transaction would then have to be submitted to obtain a new authorization code.

The unique field requirement for an Authorization Only is:

```
x_type=AUTH_ONLY
```

Merchants can submit Authorization Only transactions if they want to verify the availability of funds on the customer's credit card before finalizing the transaction. This transaction type can also be submitted if the merchant does not currently have an item in stock or wants to review orders before shipping goods.

Prior Authorization and Capture

This transaction type is used to complete an Authorization Only transaction that was successfully authorized through the payment gateway.

Note: An Authorization Only and a Prior Authorization and Capture together are considered one complete transaction. Once the Prior Authorization and Capture is submitted, the transaction will be sent for settlement.

The payment gateway accepts this transaction type and initiates settlement if the following conditions are met:

- The original Authorization Only transaction was submitted within the previous 30 days (Authorization Only transactions expire on the payment gateway after 30 days).
- The transaction is submitted with the valid transaction ID (*x_trans_id*) of an original, successfully authorized, Authorization Only transaction.
- The original transaction is not yet captured, expired or errored.
- The amount being requested for capture is less than or equal to the original authorized amount. Only a single Prior Authorization and Capture transaction can be submitted against an Authorization Only.

The unique field requirements for a Prior Authorization and Capture are:

```
x_type=PRIOR_AUTH_CAPTURE

x_trans_id=Transaction ID here
```

For this transaction type, the amount field (*x_amount*) is only required if a Prior Authorization and Capture is submitted for an amount that is *less* than the amount of the original Authorization Only transaction. If no amount is submitted, the payment gateway will initiate settlement for the amount of the original authorized transaction.

Capture Only

This transaction type is used to complete a previously authorized transaction that was *not* originally submitted through the payment gateway or that requires voice authorization.

The payment gateway accepts this transaction type and initiates settlement if the following conditions are met:

- The transaction is submitted with the valid authorization code issued to the merchant to complete the transaction.

The unique field requirements for a Capture Only are:

```
x_type=CAPTURE_ONLY

x_auth_code=Authorization Code here
```

For instructions on how to perform a Capture Only transaction in the Merchant Interface, please see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.

Credit

This transaction type is used to refund a customer for a transaction that was originally processed and successfully settled through the payment gateway.

The payment gateway accepts Credits if the following conditions are met:

- The transaction is submitted with the valid transaction ID (*x_trans_id*) of an original, successfully *settled* transaction.
- The amount being requested for refund is less than or equal to the original settled amount.
- The sum amount of multiple Credit transactions submitted against the original transaction is less than or equal to the original settled amount.
- At least the last four digits of the credit card number (*x_card_num*) used for the original, successfully settled transaction are submitted. An expiration date is not required.
- The transaction is submitted within 120 days of the settlement date of the original transaction.

The unique field requirements for a Credit are:

```
x_type=CREDIT

x_trans_id=Transaction ID here

x_card_num=Full credit card number or last four digits only here
```

Unlinked Credit

This transaction type is used to issue a refund for a transaction that was *not* originally submitted through the payment gateway. It also allows the merchant to override restrictions for submitting refunds for payment gateway transactions, for example, if the merchant is beyond the 120-day period for submitting a refund or would like to refund an amount that is greater than the original transaction amount.

The ability to submit unlinked credits is not a standard feature of a merchant's payment gateway account. To be enabled for expanded credits capability (ECC), the merchant must submit an application, which can be found at http://www.authorize.net/files/ecc.pdf.

IMPORTANT: A transaction ID must **not** be submitted with an Unlinked Credit. If ECC is enabled for the merchant's account, and a transaction ID is submitted with the Unlinked Credit transaction, then the payment gateway will attempt to apply the credit to an original transaction with the transaction ID submitted.

The unique field requirement for an Unlinked Credit is:

```
x_type=CREDIT
```

Void

This transaction type can be used to cancel either an original transaction that is not yet settled, or an entire order composed of more than one transactions. A void prevents the transaction or the order from being sent for settlement. A Void can be submitted against any other transaction type.

Note: If you are not sure whether a transaction is settled, you can attempt to submit a Void first. If the Void transaction errors, the original transaction is likely settled and you can submit a Credit for the transaction.

The payment gateway accepts Voids if the following conditions are met:

- The transaction is submitted with the valid transaction ID (*x_trans_id*) of an original, successfully authorized transaction. To void an entire order, submit the split tender ID (*x_split_tender_id*).
- The original transaction is not already settled, expired or errored.

The unique field requirements for a Void are:

```
x_type=VOID

x_trans_id=Transaction ID,  or x_split_tender_id=Split Tender ID.
```

Note: Typically, Authorization Only or Authorization and Capture are the primary transaction types submitted by an e-commerce Web site or other application. Though they most likely will not be used for the merchant's Web site integration, all other transaction types listed above can be integrated for automatic submission into an internal or enterprise application, like those used in a call center, or they can also be submitted by the merchant manually using the Virtual Terminal in the Merchant Interface.

Visa Verification Transactions

The following are required for $0 Visa verification calls:

- The transaction type must be AUTH_ONLY. All other transaction types will be rejected.
- Bill To address (`x_address`) and zip code (`x_zip`) are required to perform the AVS check.

Not all processors accept $0 Dollar Amount.

Partial Authorization Transactions

A split tender order is one in which two or more transactions are used to cover the total amount of the order.

The merchant must indicate that they are able to handle the extra processing either by selecting the **Partial Authorization** option in the Account settings of the Merchant Interface, or by sending *x_allow_partial_auth=true* with an individual transaction. Without this flag, the transaction would be handled as any other, and would be either fully authorized or declined due to lack of funds on the card.

When the first transaction is successfully approved for a partial amount of the total order, a split tender ID is generated and returned to the merchant in the response. This ID must be passed back with each of the remaining transactions of the group, using x_split_tender_id=<*value*>. If you include both a split tender ID and a transaction ID on the same request, an error results.

If successfully authorized, all transactions in the group are held until the final transaction of the group is successfully authorized.

If the merchant needs to release the group of transactions before the final one is approved (if the balance is paid by cash, for example), send a prior auth capture request and include the split tender ID instead of a transaction ID.

If the merchant needs to void the group before completion, a void request should be sent, using the split tender ID instead of a transaction ID. This will void all the transactions in the group.

The following rules apply to partial authorization transactions:

- The merchant can choose to accept partial authorization transactions by selecting an option in the Merchant Interface. Alternatively, partial authorization transactions can be submitted by including a new API field (*x_allow_partial_Auth*) in the initial request that enables partial authorization for that specific request.
- When an authorization is granted for an amount less than the purchase amount, a split tender ID is provided, in addition to the Transaction ID. The split tender ID is used on subsequent payments for that purchase.
- The transaction is not submitted for settlement until either the merchant submits payments adding up to the full requested amount, or the merchant indicates the transaction has been completed (in the case when all or part of the remaining balance is paid in cash).
- You can void all transactions in an order using a split tender ID, or you can void individual transactions using a transaction ID.
- The split tender ID cannot be submitted together with a Transaction ID; only one or the other can be submitted.

Unique field requirements for Partial Authorization Transactions are:

- x_allow_partial_auth=TRUE (input, optional) —The default value is set in the Merchant Interface; you can use this parameter to authorize individual transactions if the option is set to False in the Merchant Interface. Including this field in the transaction request overrides the merchant’s account configuration.
- x_prepaid_balance_on_card (output)—this is the authorized amount remaining on the card.
- x_prepaid_requested_amount (output)—this is the amount requested.
- x_split_tender_id (output)—this is the split tender ID provided when the first partial authorization transaction was issued. Use this ID when submitting subsequent transactions related to the same group order.

- `x_split_tender_status (output)`—indicates whether or not the transaction is complete. This parameter is sent to the merchant during relay response processing. It is not included in the delimited response.
- `x_card_type (output)`—the card type.

Using the Merchant Interface

The Merchant Interface allows merchants to manage transactions, capture Authorization Only transactions, void transactions, and issue refunds. These transaction types can also be managed automatically using the API if you are integrating a custom application to the payment gateway. However, for most integrations, these transaction types can be more conveniently and easily managed in the Merchant Interface.

For more information on submitting transactions in the Merchant Interface, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/ or click **Help** in the top right corner of the Merchant Interface.

Section 3
Transaction Data Requirements

The standard payment gateway application programming interface (API) consists of required information fields (introduced in the previous section) and additional optional fields that can be submitted to the payment gateway for real-time transaction processing.

Transaction Post Location

The merchant's Web site should submit transaction requests to the following payment gateway URL:

```
https://secure.authorize.net/gateway/transact.dll
```

Note: If you are developing using an Authorize.Net developer test account, test transactions are posted to a staging environment at **https://test.authorize.net/gateway/transact.dll**. If you do not have a developer test account, you can sign up for one at http://developer.authorize.net.

AIM Transaction Submission API

The following tables list the transaction data fields that can be submitted using the transaction request string. Several of these fields can also be configured in the Merchant Interface. For more information about configuring these settings in the Merchant Interface, please see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.

Fields are name/value pairs with the syntax:

```
x_name_of_field = value of the field&.
```

Merchant Information

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
x_login				
	Required	The merchant's unique API Login ID	Up to 20 characters	The merchant API Login ID is provided in the Merchant Interface and must be stored securely. The API Login ID and Transaction

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
				Key together provide the merchant authentication required for access to the payment gateway. See the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/ for more information.
x_tran_key				
	Required	The merchant's unique Transaction Key	16 characters	The merchant Transaction Key is provided in the Merchant Interface and must be stored securely. The API Login ID and Transaction Key together provide the merchant authentication required for access to the payment gateway. See the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/ for more information.
x_allow_partial_Auth				
	Optional	True, False		Indicates if the transaction is enabled for partial authorization. Including this field in the transaction request overrides your account configuration. For more information, see Partial Authorization Transactions on page 15.

Transaction Information

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
x_version	Required	The merchant's transaction version	3,0, 3.1	Indicates to the system the set of fields that will be included in the response: 3.0 is the default version. 3.1 allows the merchant to utilize the Card Code feature and the Partial Authorization feature, and

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
				is the current standard version. It is highly recommended that you submit this field on a per-transaction basis to be sure that the formats of transaction requests and the responses you receive are consistent. For more information, see the "Appendix A Fields by Transaction Type" sections of this document.
x_type	Optional	The type of credit card transaction	AUTH_CAPTURE (default), AUTH_ONLY, CAPTURE_ONLY, CREDIT, PRIOR_ AUTH_CAPTURE, VOID	If the value submitted does not match a supported value, the transaction is rejected. If this field is not submitted or the value is blank, the payment gateway will process the transaction as an AUTH_CAPTURE.
x_method	Optional	The payment method	CC or ECHECK	The method of payment for the transaction, CC (credit card) or ECHECK (electronic check). If this field is not submitted or is blank, the value will default to CC. For more information about eCheck.Net transaction requirements, see the eCheck.Net Developer Guide.
x_recurring_billing	Optional	The recurring billing status	TRUE, FALSE, T, F, YES, NO, Y, N, 1, 0	Indicating marker used by merchant account providers to identify transactions which originate from merchant hosted recurring billing applications. This value is not affiliated with Automated Recurring Billing.
x_amount	Required	The amount	Up to 15 digits	This is the total amount

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
		of the transaction	with a decimal point (no dollar symbol) Ex. 8.95	and must *include* tax, shipping, and any other charges. The amount can either be hard coded or posted to a script.
x_card_num	Required	The customer's credit card number	Between 13 and 16 digits without spaces When x_type=CREDIT, only the last four digits are required.	This is sensitive cardholder information and must be stored securely and in accordance with the Payment Card Industry (PCI) Data Security Standard. For more information about PCI, please see the Developer Security Best Practices White Paper.
x_exp_date	Required	The customer's credit card expiration date	MMYY, MM/YY, MM-YY, MMYYYY, MM/YYYY, MM-YYYY	This is sensitive cardholder information and must be stored securely and in accordance with the Payment Card Industry (PCI) Data Security Standard. For more information about PCI, please see the Developer Security Best Practices White Paper.
x_card_code	Optional	The customer's card code	Numeric	The three- or four-digit number on the back of a credit card (on the front for American Express). This field is required if the merchant would like to use the Card Code Verification (CCV) security feature. For more information, please see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/. Cardholder information must be stored securely and in accordance with the Payment Card Industry (PCI) Data Security Standard.

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
				Please see the Developer Security Best Practices White Paper for more information.
x_trans_id	Conditional Required only for CREDIT, PRIOR_ AUTH_ CAPTURE, and VOID transactions	The payment gateway assigned transaction ID of an original transaction		For more information about transaction types, see the "Credit Card Transaction Types" section of this document.
x_split_tender_id	Conditional	The payment gateway-assitned ID assigned when the original transaction includes two or more partial payments. This is the identifier that is used to group transactions that are part of a split tender order.		If the first transaction results in a partial authorization, the payment gateway returns this ID to the merchant. The merchant must pass this ID back with each subsequent transaction that will be part of the group of transactions sent to obtain the entire amount of the order. The payment gateway does not calculate new amounts, that is up to the merchant's software. For more information about partial authorization transactions, see the "Partial Authorization Transactions" topic.
x_auth_code	Conditional Required only for CAPTURE_ ONLY transactions	The authorization code of an original transaction *not* authorized on the payment gateway	6 characters	See the "Credit Card Transaction Types" section of this document.
x_test_request	Optional	The request to process test transactions	TRUE, FALSE, T, F, YES, NO, Y, N, 1, 0	Indicates if the transaction should be processed as a test transaction. See the "Test Transactions" section of

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
				this guide for more information.
x_duplicate_window	Optional	The window of time after the submission of a transaction that a duplicate transaction can not be submitted	Any value between 0 and 28800 (no comma)	Indicates in seconds the window of time after a transaction is submitted during which the payment gateway will check for a duplicate transaction. The maximum time allowed is 8 hours (28800 seconds). If a value less than 0 is sent, the payment gateway will default to 0 seconds. If a value greater than 28800 is sent, the payment gateway will default to 28800. If no value is sent, the payment gateway will default to 2 minutes (120 seconds). If this field is present in the request with or without a value, an enhanced duplicate transaction response is sent. See the "Response for Duplicate Transactions" section of this guide for more information.
x_merchant_descriptor				
	Optional	"MDBA= descriptor value"	Up to 255 characters	Only the first 25 characters in the field along with the 10 digit merchant phone number will be sent to the processor. The value must be URL-encoded to be recognized by the Authorize.Net system. **Note**: This field is only available for CardWorks Processing.

Order Information

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES

x_invoice_num	Optional	The merchant assigned invoice number for the transaction	Up to 20 characters (no symbols)	The invoice number must be created dynamically on the merchant server or provided on a per-transaction basis. The payment gateway does not perform this function.
x_description	Optional	The transaction description	Up to 255 characters (no symbols)	The description must be created dynamically on the merchant server or provided on a per-transaction basis. The payment gateway does not perform this function.

Itemized Order Information

Based on their respective business requirements, merchants can choose to submit itemized order information with a transaction. Itemized order information is not submitted to the processor and is not currently returned with the transaction response. This information is displayed on the Transaction Detail page and in QuickBooks download file reports in the Merchant Interface.

Note: The value for the *x_line_item* field is capable of including delimited item information. In this case, line item values must be included in the order listed below.

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
x_line_item				
	Optional	Any string	Line item values must be delimited by a bracketed pipe <|>	Itemized order information.
		Item ID<|>	Up to 31 characters	The ID assigned to an item.
		<|>item name<|>	Up to 31 characters	A short description of an item.
		<|>item description<|>	Up to 255 characters	A detailed description of an item.
		<|>itemX quantity<|>	Up to two decimal places Must be a positive number	The quantity of an item.

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
		<|>item price (unit cost)<|>	Up to two decimal places Must be a positive number	Cost of an item per unit, *excluding* tax, freight, and duty. The dollar sign ($) is not allowed when submitting delimited information.
		<|>itemX taxable	TRUE, FALSE, T, F, YES, NO, Y, N, 1, 0	Indicates whether the item is subject to tax.

The merchant can submit up to 30 distinct line items containing itemized order information per transaction. For example:

Sample 6. Submitting itemized order information

```
x_line_item=item1<|>golf balls<|><|>2<|>18.95<|>Y&x_line_item=
item2<|>golf bag<|>Wilson golf carry bag, red<|>1<|>39.99<|>Y&
x_line_item=item3<|>book<|>Golf for Dummies<|>1<|>21.99<|>Y&
```

Note: For Prior Authorization and Capture transactions, if line item information was submitted with the original transaction, adjusted information can be submitted if the transaction changed. If no adjusted line item information is submitted, the information submitted with the original transaction will apply.

Customer Information

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
x_first_name	Optional	The first name associated with the customer's billing address	Up to 50 characters (no symbols)	
x_last_name	Optional	The last name associated with the customer's billing address	Up to 50 characters (no symbols)	

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
x_company	Optional	The company associated with the customer's billing address	Up to 50 characters (no symbols)	
x_address	Optional	The customer's billing address	Up to 60 characters (no symbols)	Required if the merchant would like to use the Address Verification Service security feature. For more information on AVS, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.
x_city	Optional	The city of the customer's billing address	Up to 40 characters (no symbols)	
x_state	Optional	The state of the customer's billing address	Up to 40 characters (no symbols) or a valid two-character state code	
x_zip	Optional	The ZIP code of the customer's billing address	Up to 20 characters (no symbols)	Required if the merchant would like to use the Address Verification Service security feature. For more information on AVS, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.
x_country	Optional	The country of the customer's billing address	Up to 60 characters (no symbols)	
x_phone	Optional	The phone number associated with the customer's billing address	Up to 25 digits (no letters) Ex. (123)123-1234	

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
x_fax	Optional	The fax number associated with the customer's billing address	Up to 25 digits (no letters) Ex. (123)123-1234	
x_email	Optional	The customer's valid email address	Up to 255 characters Ex. janedoe@customer.com	The email address to which the customer's copy of the email receipt is sent when Email Receipts is configured in the Merchant Interface. The email is sent to the customer only if the email address format is valid. For more information about Email Receipts, please see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.
x_cust_id	Optional	The merchant assigned customer ID	Up to 20 characters (no symbols)	The unique identifier to represent the customer associated with the transaction. The customer ID must be created dynamically on the merchant server or provided on a per-transaction basis. The payment gateway does not perform this function.
x_customer_ip	Optional	The customer's IP address	Up to 15 characters (no letters) Ex. 255.255.255.255	IP address of the customer initiating the transaction. If this value is not passed, it will default to 255.255.255.255. This field is required when using the Fraud Detection Suite™ (FDS) IP Address Blocking tool. For

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
				more information about FDS, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.

Shipping Information

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
x_ship_to_first_name	Optional	The first name associated with the customer's shipping address	Up to 50 characters (no symbols)	
x_ship_to_last_name	Optional	The last name associated with the customer's shipping address	Up to 50 characters (no symbols)	
x_ship_to_company	Optional	The company associated with the customer's shipping address	Up to 50 characters (no symbols)	
x_ship_to_address	Optional	The customer's shipping address	Up to 60 characters (no symbols)	
x_ship_to_city	Optional	The city of the customer's shipping address	Up to 40 characters (no symbols)	
x_ship_to_state	Optional	The state of the customer's shipping address	Up to 40 characters (no symbols) or a valid two-character state code	
x_ship_to_zip	Optional	The ZIP code of the customer's shipping address	Up to 20 characters (no symbols)	
x_ship_to_country	Optional	The country of the customer's shipping address	Up to 60 characters (no symbols)	

Additional Shipping Information (Level 2 Data)

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
x_tax	Optional	The valid tax amount OR delimited tax information	When submitting delimited tax information, values must be delimited by a bracketed pipe <|>	The tax amount charged OR when submitting this information using the transaction request string, delimited tax information including the sales tax name, description, and amount is also allowed. The total amount of the transaction in x_amount must *include* this amount.
		tax item name<|>		The tax item name.
		tax description<|>		The tax item description.
		tax amount	The dollar sign ($) is not allowed when submitting delimited information.	The tax item amount. The total amount of the transaction in x_amount must *include* this amount.
	Example: x_tax=Tax1<|>state tax<|>0.09&			
x_freight	Optional	The valid freight amount OR delimited freight information	When submitting delimited freight information, values must be delimited by a bracketed pipe <|>	The freight amount charged OR when submitting this information using the transaction request string, delimited freight information including the freight name, description, and amount is also allowed. The total amount of the transaction in x_amount must *include* this amount.
		freight item name<|>		The freight item name.
		freight description<|>		The freight item description.
		freight amount	The dollar sign ($) is not allowed when submitting delimited	The freight amount. The total amount of the transaction in x_amount

<table>
<tr><th>FIELD NAME</th><th>REQUIRED</th><th>VALUE</th><th>FORMAT</th><th>NOTES</th></tr>
<tr><td></td><td></td><td></td><td>information.</td><td>must <i>include</i> this amount.</td></tr>
<tr><td></td><td colspan="4">Example: x_freight=Freight<|>ground overnight<|>12.95&</td></tr>
<tr><td>x_duty</td><td>Optional</td><td>The valid duty amount OR delimited duty information</td><td>When submitting delimited duty information, values must be delimited by a pipe <|></td><td>The duty amount charged OR when submitting this information using the transaction request string, delimited duty information including the duty name, description, and amount is also allowed.
The total amount of the transaction in x_amount must <i>include</i> this amount.</td></tr>
<tr><td></td><td></td><td>duty item name<|></td><td></td><td>The duty item name.</td></tr>
<tr><td></td><td></td><td>duty description<|></td><td></td><td>The duty item description.</td></tr>
<tr><td></td><td></td><td>duty amount</td><td>The dollar sign ($) is not allowed when submitting delimited information.</td><td>The duty amount.
The total amount of the transaction in x_amount must <i>include</i> this amount.</td></tr>
<tr><td></td><td colspan="4">Example: x_duty=Duty1<|>export<|>15.00&</td></tr>
<tr><td>x_tax_exempt</td><td>Optional</td><td>The tax exempt status</td><td>TRUE, FALSE,
T, F,
YES, NO,
Y, N,
1, 0</td><td>Indicates whether the transaction is tax exempt.</td></tr>
<tr><td>x_po_num</td><td>Optional</td><td>The merchant assigned purchase order number</td><td>Up to 25 characters (no symbols)</td><td>The purchase order number must be created dynamically on the merchant server or provided on a per-transaction basis. The payment gateway does not perform this function.</td></tr>
</table>

Note: Delimited duty, freight, and tax information are not returned in the transaction response or in the merchant confirmation email. This information is displayed only on the Transaction Detail page in the Merchant Interface.

Cardholder Authentication

The payment gateway supports the transmission of authentication fields for the following cardholder authentication programs:

- Verified by Visa
- MasterCard® SecureCode™

Merchants using a third party cardholder authentication solution can submit the following authentication values with Visa and/or MasterCard transactions.

Note: The cardholder authentication fields are currently supported only through the Chase Paymentech, FDMS Nashville, Global Payments and TSYS processors for Visa and MasterCard transactions. If cardholder authentication information is submitted for transactions processed through any other processor, it will be ignored.

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
x_authentication_indicator				
	Optional	The electronic commerce indicator (ECI) value for a Visa transaction; *or* the universal cardholder authentication field indicator (UCAF) for a MasterCard transaction obtained by the merchant after the authentication process.	Special characters included in this value must be URL encoded.	Required only for AUTH_ONLY and AUTH_CAPTURE transactions processed through cardholder authentication programs. When submitted with other transaction types, this value is ignored. This field is currently supported through Chase Paymentech, FDMS Nashville, Global Payments and TSYS.
x_cardholder_authentication_value				
	Optional	The cardholder authentication verification value (CAVV) for a Visa transaction; *or* accountholder authentication value (AVV)/ universal cardholder authentication field (UCAF) for a	Special characters included in this value must be URL encoded.	Required only for AUTH_ONLY and AUTH_CAPTURE transactions processed through cardholder authentication programs. When submitted with other transaction types, this value is ignored.

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
		MasterCard transaction obtained by the merchant after the authentication process.		This field is currently supported through Chase Paymentech, FDMS Nashville, Global Payments and TSYS.

Invalid combinations of the *x_authentication_indicator* and *x_cardholder_authentication_value* fields will cause the transaction to generate an error.

Valid value combinations for these fields are as follows:

Visa

AUTHENTICATION INDICATOR	CARDHOLDER AUTHENTICATION VALUE
5	Not null
6	Not null
6	Null/Blank
7	Null/Blank
7	Not null (some international issuers can provide a CAVV value when ECI is 7)
Null/Blank	Null/Blank

MasterCard

AUTHENTICATION INDICATOR	CARDHOLDER AUTHENTICATION VALUE
0	Blank /Null
2	Not null
1	Null
Null	Null

For example, when the MasterCard value for *x_authentication_indicator* is “1,” the value for *x_cardholder_authentication_value* must be null. In this scenario, if a value is submitted for *x_cardholder_authentication_value*, the transaction fails validation and is rejected.

The authentication verification value returned by Visa or MasterCard is included in the transaction response from the payment gateway and is also included on the Transaction Detail page for the transaction in the Merchant Interface.

Merchant-defined fields

Merchants can also choose to include merchant-defined fields to further customize the information included with a transaction. Merchant-defined fields are any fields that are not recognized by the payment gateway as standard application programming interface (API) fields.

For example, the merchant might want to provide a field in which customers provide specific shipping instructions and product color information. All you need to do is submit a custom field name and any accompanying text with the transaction request string—for example, *shipping_instructions* and *product_color*.

Note: Standard payment gateway fields that are misspelled are treated as merchant-defined fields.

Warning: Merchant-Defined Data fields are not intended to and MUST NOT be used to capture personally identifying information. Accordingly, the merchant is prohibited from capturing, obtaining, and/or transmitting any personally identifying information in or by means of the Merchant-Defined Data fields. Personally identifying information includes, but is not limited to, name, address, credit card number, social security number, driver's license number, state-issued identification number, passport number, and card verification numbers (CVV, CVC2, CVV2, CID, CVN). If Authorize.Net, a CyberSource solution, discovers that Merchant is capturing and/or transmitting personally identifying information by means of the Merchant-Defined Data fields, whether or not intentionally, CyberSource WILL immediately suspend the merchant's account, which will result in a rejection of any and all transaction requests submitted by the merchant after the point of suspension.

Section 4
Transaction Response

The transaction response from the payment gateway is returned as a delimited string and provides information about the status of a transaction—whether it was accepted or declined—as well as information included in the transaction request.

Fields in the response are delimited by a character that is specified in the transaction request string (*x_delim_char*) or configured in the Merchant Interface. The merchant server can parse this data to customize receipt messages to display or email to the customer. Transaction results are also provided in the payment gateway merchant confirmation email, and on the Transaction Detail page for the transaction in the Merchant Interface.

The following fields can be used to customize the format of the payment gateway transaction response. These settings can also be configured in the Merchant Interface. For more information about configuring these settings in the Merchant Interface, please see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.

Fields are name/value pairs with the syntax:

```
    x_name_of_field=value of the field&.
```

FIELD NAME	VALUE	FORMAT	DESCRIPTION
x_delim_data	The request to receive a delimited transaction response	TRUE	In order to receive a delimited response from the payment gateway, this field must be submitted with a value of TRUE or the merchant has to configure a delimited response through the Merchant Interface. You should submit this field on a per-transaction basis to be sure that transaction responses are returned in the correct format.
x_delim_char	The delimiting character	A single symbol Ex. , (comma) | (pipe) “ (double quotes) ‘ (single quote) : (colon) ; (semicolon) / (forward slash) \ (back slash) - (dash)	The character that is used to separate fields in the transaction response. The payment gateway will use the character passed in this field or the value stored in the Merchant Interface if no value is passed. If this field is passed, and the value is null, it will override the value stored in the Merchant Interface and there is no delimiting character in the transaction response.

		* (star)	You should submit this field on a per-transaction basis to be sure that transaction responses are returned in the correct format.
x_encap_char	The encapsulating character	A single symbol Ex. | (pipe) “ (double quotes) ‘ (single quote) : (colon) ; (semicolon) / (forward slash) \ (back slash) - (dash) * (star)	The character that is used to encapsulate the fields in the transaction response. This is only necessary if it is possible that your delimiting character could be included in any field values. The payment gateway will use the character passed in this field or the value stored in the Merchant Interface if no value is passed.

Fields in the Payment Gateway Response

The following table lists the fields returned in the response from the payment gateway.

ORDER	FIELD NAME	VALUE	FORMAT	NOTES
1	Response Code	The overall status of the transaction	• 1 = Approved • 2 = Declined • 3 = Error • 4 = Held for Review	
2	Response Subcode	A code used by the payment gateway for internal transaction tracking		
3	Response Reason Code	A code that represents more details about the result of the transaction	Numeric	See the “Response Code Details” section of this document for a listing of response reason codes.
4	Response Reason Text	A brief description of the result, which corresponds with the	Text	You can generally use this text to display a transaction result or error to the customer. However, please review the “Response Code Details” section of this document to identify any

ORDER	FIELD NAME	VALUE	FORMAT	NOTES
		response reason code		specific texts you do not want to pass to the customer.
5	Authorization Code	The authorization or approval code	6 characters	

6	AVS Response	The Address Verification Service (AVS) response code	A = Address (Street) matches, ZIP does not
			B = Address information not provided for AVS check
			E = AVS error
			G = Non-U.S. Card Issuing Bank
			N = No Match on Address (Street) or ZIP
			P = AVS not applicable for this transaction
			R = Retry – System unavailable or timed out
			S = Service not supported by issuer
			U = Address information is unavailable
			W = Nine digit ZIP matches, Address (Street) does not
			X = Address (Street) and nine digit ZIP match
			Y = Address (Street) and five digit ZIP match
			Z = Five digit ZIP matches, Address (Street) does not
				Indicates the result of the AVS filter.
				For more information about AVS, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.

7	Transaction ID	The payment gateway assigned identification number for the	When `x_test_request` is set to a positive response, or when Test Mode is enabled on the payment gateway, this value will be "0."	This value must be used for any follow on transactions such as a CREDIT, PRIOR_AUTH_CAPTURE or VOID.

ORDER	FIELD NAME	VALUE	FORMAT	NOTES
		transaction		
8	Invoice Number	The merchant assigned invoice number for the transaction	Up to 20 characters (no symbols)	
9	Description	The transaction description	Up to 255 characters (no symbols)	
10	Amount	The amount of the transaction	Up to 15 digits	
11	Method	The payment method	CC or ECHECK	
12	Transaction Type	The type of credit card transaction	AUTH_CAPTURE, AUTH_ONLY, CAPTURE_ONLY, CREDIT, PRIOR_AUTH_CAPTURE VOID	
13	Customer ID	The merchant assigned customer ID	Up to 20 characters (no symbols)	
14	First Name	The first name associated with the customer's billing address	Up to 50 characters (no symbols)	
15	Last Name	The last name associated with the customer's billing address	Up to 50 characters (no symbols)	
16	Company	The company associated with the customer's billing	Up to 50 characters (no symbols)	

ORDER	FIELD NAME	VALUE	FORMAT	NOTES
		address		
17	Address	The customer's billing address	Up to 60 characters (no symbols)	
18	City	The city of the customer's billing address	Up to 40 characters (no symbols)	
19	State	The state of the customer's billing address	Up to 40 characters (no symbols) or a valid two-character state code	
20	ZIP Code	The ZIP code of the customer's billing address	Up to 20 characters (no symbols)	
21	Country	The country of the customer's billing address	Up to 60 characters (no symbols)	
22	Phone	The phone number associated with the customer's billing address	Up to 25 digits (no letters) Ex. (123)123-1234	
23	Fax	The fax number associated with the customer's billing address	Up to 25 digits (no letters) Ex. (123)123-1234	
24	Email Address	The customer's valid email	Up to 255 characters	

ORDER	FIELD NAME	VALUE	FORMAT	NOTES
		address		
25	Ship To First Name	The first name associated with the customer's shipping address	Up to 50 characters (no symbols)	
26	Ship To Last Name	The last name associated with the customer's shipping address	Up to 50 characters (no symbols)	
27	Ship To Company	The company associated with the customer's shipping address	Up to 50 characters (no symbols)	
28	Ship To Address	The customer's shipping address	Up to 60 characters (no symbols)	
29	Ship To City	The city of the customer's shipping address	Up to 40 characters (no symbols)	
30	Ship To State	The state of the customer's shipping address	Up to 40 characters (no symbols) or a valid two-character state code	
31	Ship To ZIP Code	The ZIP code of the customer's shipping address	Up to 20 characters (no symbols)	
32	Ship To Country	The country of the customer's shipping	Up to 60 characters (no symbols)	

ORDER	FIELD NAME	VALUE	FORMAT	NOTES
		address		
33	Tax	The tax amount charged	Numeric	Delimited tax information is not included in the transaction response.
34	Duty	The duty amount charged	Numeric	Delimited duty information is not included in the transaction response.
35	Freight	The freight amount charged	Numeric	Delimited freight information is not included in the transaction response.
36	Tax Exempt	The tax exempt status	TRUE, FALSE, T, F, YES, NO, Y, N, 1, 0	
37	Purchase Order Number	The merchant assigned purchase order number	Up to 25 characters (no symbols)	
38	MD5 Hash	The payment gateway generated MD5 hash value that can be used to authenticate the transaction response.		Because transaction responses are returned using an SSL connection, this feature is not necessary for AIM.

39	Card Code Response	The card code verification (CCV) response code	M = Match
			N = No Match
			P = Not Processed
			S = Should have been present
			U = Issuer unable to process request
				Indicates the result of the CCV filter.
				For more information about CCV, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.

40	Cardholder Authentication Verification	The cardholder authentication verification response	Blank or not present = CAVV not validated 0 = CAVV not validated because erroneous data was submitted	

ORDER	FIELD NAME	VALUE	FORMAT	NOTES
	Response	code	1 = CAVV failed validation 2 = CAVV passed validation 3 = CAVV validation could not be performed; issuer attempt incomplete 4 = CAVV validation could not be performed; issuer system error 5 = Reserved for future use 6 = Reserved for future use 7 = CAVV attempt – failed validation – issuer available (U.S.-issued card/non-U.S acquirer) 8 = CAVV attempt – passed validation – issuer available (U.S.-issued card/non-U.S. acquirer) 9 = CAVV attempt – failed validation – issuer unavailable (U.S.-issued card/non-U.S. acquirer) A = CAVV attempt – passed validation – issuer unavailable (U.S.-issued card/non-U.S. acquirer) B = CAVV passed validation, information only, no liability shift	
51	Account Number	Last 4 digits of the card provided	Alphanumeric (XXXX6835)	This field is returned with all transactions.
52	Card Type	Visa, MasterCard, American Express, Discover, Diners Club, JCB	Text	

ORDER	FIELD NAME	VALUE	FORMAT	NOTES
53	Split Tender ID	The value that links the current authorization request to the original authorization request. This value is returned in the reply message from the original authorization request	Alphanumeric	This is only returned in the reply message for the first transaction that receives a partial authorization.
54	Requested Amount	Amount requested in the original authorization	Numeric	This is present if the current transaction is for a prepaid card or if a splitTenderId was sent in.
55	Balance On Card	Balance on the debit card or prepaid card	Numeric	Can be a positive or negative number. This has a value only if the current transaction is for a prepaid card

Response for Duplicate Transactions

The AIM API allows you to specify the window of time after a transaction is submitted during which the payment gateway checks for a duplicate transaction (based on credit card number, invoice number, amount, billing address information, transaction type, etc.) using the duplicate window field (*x_duplicate_window*). The value for this field can be between 0 and 28800 seconds (maximum of 8 hours).

If the transaction request does not include the duplicate window field, and the payment gateway detects a duplicate transaction within the default window of 2 minutes, the payment gateway response will contain the response code of 3 (processing error) with a response reason code of 11 (duplicate transaction) with no additional details.

If the transaction request *does* include the duplicate window field and value, and the payment gateway detects a duplicate transaction within the window of time specified, the payment gateway response for the duplicate transaction will include the response code and response reason code listed above, as well as information about the original transaction (as outlined below).

If the original transaction was declined, and a value was passed in the duplicate window field, the payment gateway response for the duplicate transaction will include the following information for the original transaction:

- The AVS result
- The CCV result
- The transaction ID

If the original transaction was approved, and a value was passed in the duplicate window field, the payment gateway response will also include the authorization code for the original transaction. All duplicate transactions submitted after the duplicate window, whether specified in the transaction request or after the payment gateway's default 2 minute duplicate window, are processed normally.

AIM Transaction Response Types

There are two versions of the AIM response string.

Version 3.0

The version 3.0 response contains system fields from position 1 to 38 and echoes merchant defined fields from 39 on, in the order received by the system.

Example

The following is a sample transaction query string:

```
https://test.authorize.net/gateway/transact.dll?x_login=YourAPILogi
n&x_tran_key=YourTransactionKey&x_delim_data=true&x_relay_response=
false&x_card_num=4111111111111111&x_exp_date=1010&x_amount=1.00&mer
chant_defined_field1=merchant-defined-
field_1&merchant_defined_field2=merchant_defined_field_2
```

The response would be:

```
1,1,1,This transaction has been
approved.,iKUUAm,Y,2149207036,,,1.00,CC,auth_capture,,,,,,,,,,,,,,,
,,,,,,,,,,,433AF62576BFC33D0B7B8A75FB229220,merchant defined field
1,merchant defined field 2
```

Version 3.1

The version 3.1 response string contains 68 system fields with field number 39 representing the Card Code (CVV2/CVC2/CID) response code. Merchant-defined fields are echoed from field 69 on. Merchants wishing to use the Card Code feature and merchants who accept partial authorization transactions must use transaction version 3.1.

Example

The following is a sample transaction query string:

```
https://test.authorize.net/gateway/transact.dll?x_login=99W58L5veks
j&x_tran_key=2jL4g9447PQJd3uF&x_delim_data=true&x_relay_response=fa
lse&x_card_num=4111111111111111&x_exp_date=1010&x_amount=1.00&merch
ant_defined_field1=merchant defined field
1&merchant_defined_field2=merchant defined field 2
```

The response would be:

```
1,1,1,This transaction has been
approved.,tt9ieF,Y,2149207083,,,1.00,CC,auth_capture,,,,,,,,,,,,,,,
,,,,,,,,,,,DE9F3FAD313D32144FA7EDF4B27CCE02,,2,,,,,,,,,,,,,,,,,,,,,
,,,,,,,,merchant defined field 1,merchant defined field 2
```

Setting the Transaction Version

To set the transaction version, do the following:

1. Log on to the Merchant Interface
2. Select **Settings** from the Main Menu
3. Click **Transaction Version** in the Transaction Response section
4. Change the Transaction Version by using the drop-down box
5. Click **Submit** to save changes

You can also set the Transaction version on a per-transaction basis by using x_version.

Note: You can only upgrade to a higher transaction version. You cannot set your transaction version to a previous version.

Response Code Details

The following tables describe the response codes and response reason texts that are returned for each transaction. In addition to the information in this document, the Authorize.Net Integration Center at http://developer.authorize.net/tools/responsereasoncode provides a valuable tool for troubleshooting errors.

- **Response Code** indicates the overall status of the transaction with possible values of approved, declined, errored, or held for review.
- **Response Reason Code** is a numeric representation of a more specific reason for the transaction status.
- **Response Reason Text** details the specific reason for the transaction status. This information can be returned to the merchant and/or customer to provide more information about the status of the transaction.

Response Codes

RESPONSE	DESCRIPTION

CODE	
1	This transaction has been approved.
2	This transaction has been declined.
3	There has been an error processing this transaction.
4	This transaction is being held for review.

Response Reason Codes and Response Reason Text

RESPONSE CODE	RESPONSE REASON CODE	RESPONSE REASON TEXT	NOTES
1	1	This transaction has been approved.	
2	2	This transaction has been declined.	
2	3	This transaction has been declined.	
2	4	This transaction has been declined.	The code returned from the processor indicating that the card used needs to be picked up.
3	5	A valid amount is required.	The value submitted in the amount field did not pass validation for a number.
3	6	The credit card number is invalid.	
3	7	The credit card expiration date is invalid.	The format of the date submitted was incorrect.
3	8	The credit card has expired.	
3	9	The ABA code is invalid.	The value submitted in the x_bank_aba_code field did not pass validation or was not for a valid financial institution.
3	10	The account number is invalid.	The value submitted in the x_bank_acct_num field did not pass validation.
3	11	A duplicate transaction has been submitted.	A transaction with identical amount and credit card information was submitted two minutes prior.
3	12	An authorization code is required	A transaction that required x_auth_code to be present was

RESPONSE CODE	RESPONSE REASON CODE	RESPONSE REASON TEXT	NOTES
		but not present.	submitted without a value.
3	13	The merchant API Login ID is invalid or the account is inactive.	
3	14	The Referrer or Relay Response URL is invalid.	The Relay Response or Referrer URL does not match the merchant's configured value(s) or is absent. Applicable only to SIM and WebLink APIs.
3	15	The transaction ID is invalid.	The transaction ID value is non-numeric or was not present for a transaction that requires it (i.e., VOID, PRIOR_AUTH_CAPTURE, and CREDIT).
3	16	The transaction was not found.	The transaction ID sent in was properly formatted but the gateway had no record of the transaction.
3	17	The merchant does not accept this type of credit card.	The merchant was not configured to accept the credit card submitted in the transaction.
3	18	ACH transactions are not accepted by this merchant.	The merchant does not accept electronic checks.
3	19 - 23	An error occurred during processing. Please try again in 5 minutes.	
3	24	The Nova Bank Number or Terminal ID is incorrect. Call Merchant Service Provider.	
3	25 - 26	An error occurred during processing. Please try again in 5 minutes.	
2	27	The transaction resulted in an AVS mismatch. The address provided does not match billing address of cardholder.	
2	28	The merchant does not accept this type of credit card.	The Merchant ID at the processor was not configured to accept this card type.
2	29	The Paymentech identification numbers are incorrect. Call	

RESPONSE CODE	RESPONSE REASON CODE	RESPONSE REASON TEXT	NOTES
		Merchant Service Provider.	
2	30	The configuration with the processor is invalid. Call Merchant Service Provider.	
2	31	The FDC Merchant ID or Terminal ID is incorrect. Call Merchant Service Provider.	The merchant was incorrectly set up at the processor.
3	32	This reason code is reserved or not applicable to this API.	
3	33	*FIELD* cannot be left blank.	The word *FIELD* will be replaced by an actual field name. This error indicates that a field the merchant specified as required was not filled in. Please see the Form Fields section of the Merchant Integration Guide for details.
2	34	The VITAL identification numbers are incorrect. Call Merchant Service Provider.	The merchant was incorrectly set up at the processor.
2	35	An error occurred during processing. Call Merchant Service Provider.	The merchant was incorrectly set up at the processor.
3	36	The authorization was approved, but settlement failed.	
2	37	The credit card number is invalid.	
2	38	The Global Payment System identification numbers are incorrect. Call Merchant Service Provider.	The merchant was incorrectly set up at the processor.
3	40	This transaction must be encrypted.	
2	41	This transaction has been declined.	Only merchants set up for the FraudScreen.Net service would receive this decline. This code will be returned if a given transaction’s fraud score is higher than the threshold set by the merchant.
3	43	The merchant was incorrectly set up at the processor. Call your Merchant Service Provider.	The merchant was incorrectly set up at the processor.

RESPONSE CODE	RESPONSE REASON CODE	RESPONSE REASON TEXT	NOTES
2	44	This transaction has been declined.	The card code submitted with the transaction did not match the card code on file at the card issuing bank and the transaction was declined.
2	45	This transaction has been declined.	This error would be returned if the transaction received a code from the processor that matched the rejection criteria set by the merchant for both the AVS and Card Code filters.
3	46	Your session has expired or does not exist. You must log in to continue working.	
3	47	The amount requested for settlement may not be greater than the original amount authorized.	This occurs if the merchant tries to capture funds greater than the amount of the original authorization-only transaction.
3	48	This processor does not accept partial reversals.	The merchant attempted to settle for less than the originally authorized amount.
3	49	A transaction amount greater than $[amount] will not be accepted.	The transaction amount submitted was greater than the maximum amount allowed.
3	50	This transaction is awaiting settlement and cannot be refunded.	Credits or refunds can only be performed against settled transactions. The transaction against which the credit/refund was submitted has not been settled, so a credit cannot be issued.
3	51	The sum of all credits against this transaction is greater than the original transaction amount.	
3	52	The transaction was authorized, but the client could not be notified; the transaction will not be settled.	
3	53	The transaction type was invalid for ACH transactions.	If x_method = ECHECK, x_type cannot be set to CAPTURE_ONLY.
3	54	The referenced transaction does not meet the criteria for issuing a	

RESPONSE CODE	RESPONSE REASON CODE	RESPONSE REASON TEXT	NOTES
		credit.	
3	55	The sum of credits against the referenced transaction would exceed the original debit amount.	The transaction is rejected if the sum of this credit and prior credits exceeds the original debit amount
3	56	This merchant accepts ACH transactions only; no credit card transactions are accepted.	The merchant processes eCheck.Net transactions only and does not accept credit cards.
3	57 - 63	An error occurred in processing. Please try again in 5 minutes.	
2	65	This transaction has been declined.	The transaction was declined because the merchant configured their account through the Merchant Interface to reject transactions with certain values for a Card Code mismatch.
3	66	This transaction cannot be accepted for processing.	The transaction did not meet gateway security guidelines.
3	68	The version parameter is invalid.	The value submitted in x_version was invalid.
3	69	The transaction type is invalid.	The value submitted in x_type was invalid.
3	70	The transaction method is invalid.	The value submitted in x_method was invalid.
3	71	The bank account type is invalid.	The value submitted in x_bank_acct_type was invalid.
3	72	The authorization code is invalid.	The value submitted in x_auth_code was more than six characters in length.
3	73	The driver's license date of birth is invalid.	The format of the value submitted in x_drivers_license_dob was invalid.
3	74	The duty amount is invalid.	The value submitted in x_duty failed format validation.
3	75	The freight amount is invalid.	The value submitted in x_freight failed format validation.
3	76	The tax amount is invalid.	The value submitted in x_tax failed format validation.
3	77	The SSN or tax ID is invalid.	The value submitted in x_customer_tax_id failed

RESPONSE CODE	RESPONSE REASON CODE	RESPONSE REASON TEXT	NOTES
			validation.
3	78	The Card Code (CVV2/CVC2/CID) is invalid.	The value submitted in x_card_code failed format validation.
3	79	The driver's license number is invalid.	The value submitted in x_drivers_license_num failed format validation.
3	80	The driver's license state is invalid.	The value submitted in x_drivers_license_state failed format validation.
3	81	The requested form type is invalid.	The merchant requested an integration method not compatible with the AIM API.
3	82	Scripts are only supported in version 2.5.	The system no longer supports version 2.5; requests cannot be posted to scripts.
3	83	The requested script is either invalid or no longer supported.	The system no longer supports version 2.5; requests cannot be posted to scripts.
3	84	This reason code is reserved or not applicable to this API.	
3	85	This reason code is reserved or not applicable to this API.	
3	86	This reason code is reserved or not applicable to this API.	
3	87	This reason code is reserved or not applicable to this API.	
3	88	This reason code is reserved or not applicable to this API.	
3	89	This reason code is reserved or not applicable to this API.	
3	90	This reason code is reserved or not applicable to this API.	
3	91	Version 2.5 is no longer supported.	
3	92	The gateway no longer supports the requested method of integration.	

RESPONSE CODE	RESPONSE REASON CODE	RESPONSE REASON TEXT	NOTES
3	97	This transaction cannot be accepted.	Applicable only to SIM API. Fingerprints are only valid for a short period of time. If the fingerprint is more than one hour old or more than 15 minutes into the future, it will be rejected. This code indicates that the transaction fingerprint has expired.
3	98	This transaction cannot be accepted.	Applicable only to SIM API. The transaction fingerprint has already been used.
3	99	This transaction cannot be accepted.	Applicable only to SIM API. The server-generated fingerprint does not match the merchant-specified fingerprint in the x_fp_hash field.
3	100	The eCheck.Net type is invalid.	Applicable only to eCheck.Net. The value specified in the x_echeck_type field is invalid.
3	101	The given name on the account and/or the account type does not match the actual account.	Applicable only to eCheck.Net. The specified name on the account and/or the account type do not match the NOC record for this account.
3	102	This request cannot be accepted.	A password or Transaction Key was submitted with this WebLink request. This is a high security risk.
3	103	This transaction cannot be accepted.	A valid fingerprint, Transaction Key, or password is required for this transaction.
3	104	This transaction is currently under review.	Applicable only to eCheck.Net. The value submitted for country failed validation.
3	105	This transaction is currently under review.	Applicable only to eCheck.Net. The values submitted for city and country failed validation.
3	106	This transaction is currently under review.	Applicable only to eCheck.Net. The value submitted for company failed validation.
3	107	This transaction is currently under review.	Applicable only to eCheck.Net. The value submitted for bank account name failed validation.

RESPONSE CODE	RESPONSE REASON CODE	RESPONSE REASON TEXT	NOTES
3	108	This transaction is currently under review.	Applicable only to eCheck.Net. The values submitted for first name and last name failed validation.
3	109	This transaction is currently under review.	Applicable only to eCheck.Net. The values submitted for first name and last name failed validation.
3	110	This transaction is currently under review.	Applicable only to eCheck.Net. The value submitted for bank account name does not contain valid characters.
3	116	The authentication indicator is invalid.	This error is only applicable to Verified by Visa and MasterCard SecureCode transactions. The ECI value for a Visa transaction; or the UCAF indicator for a MasterCard transaction submitted in the x_authentication_indicator field is invalid.
3	117	The cardholder authentication value is invalid.	This error is only applicable to Verified by Visa and MasterCard SecureCode transactions. The CAVV for a Visa transaction; or the AVV/UCAF for a MasterCard transaction is invalid.
3	118	The combination of authentication indicator and cardholder authentication value is invalid.	This error is only applicable to Verified by Visa and MasterCard SecureCode transactions. The combination of authentication indicator and cardholder authentication value for a Visa or MasterCard transaction is invalid. For more information, see the "Cardholder Authentication" section of this document.
3	119	Transactions having cardholder authentication values cannot be marked as recurring.	This error is only applicable to Verified by Visa and MasterCard SecureCode transactions. Transactions submitted with a value in x_authentication_indicator *and* x_recurring_billing=YES will be rejected.
3	120	An error occurred during	The system-generated void for the original timed-out transaction

RESPONSE CODE	RESPONSE REASON CODE	RESPONSE REASON TEXT	NOTES
		processing. Please try again.	failed. (The original transaction timed out while waiting for a response from the authorizer.)
3	121	An error occurred during processing. Please try again.	The system-generated void for the original errored transaction failed. (The original transaction experienced a database error.)
3	122	An error occurred during processing. Please try again.	The system-generated void for the original errored transaction failed. (The original transaction experienced a processing error.)
3	123	This account has not been given the permission(s) required for this request.	The transaction request must include the API Login ID associated with the payment gateway account.
2	127	The transaction resulted in an AVS mismatch. The address provided does not match billing address of cardholder.	The system-generated void for the original AVS-rejected transaction failed.
3	128	This transaction cannot be processed.	The customer’s financial institution does not currently allow transactions for this account.
3	130	This payment gateway account has been closed.	IFT: The payment gateway account status is Blacklisted.
3	131	This transaction cannot be accepted at this time.	IFT: The payment gateway account status is Suspended-STA.
3	132	This transaction cannot be accepted at this time.	IFT: The payment gateway account status is Suspended-Blacklist.
2	141	This transaction has been declined.	The system-generated void for the original FraudScreen-rejected transaction failed.
2	145	This transaction has been declined.	The system-generated void for the original card code-rejected and AVS-rejected transaction failed.
3	152	The transaction was authorized, but the client could not be notified; the transaction will not be settled.	The system-generated void for the original transaction failed. The response for the original transaction could not be communicated to the client.
2	165	This transaction has been declined.	The system-generated void for the original card code-rejected

RESPONSE CODE	RESPONSE REASON CODE	RESPONSE REASON TEXT	NOTES
			transaction failed.
3	170	An error occurred during processing. Please contact the merchant.	Concord EFS – Provisioning at the processor has not been completed.
2	171	An error occurred during processing. Please contact the merchant.	Concord EFS – This request is invalid.
2	172	An error occurred during processing. Please contact the merchant.	Concord EFS – The store ID is invalid.
3	173	An error occurred during processing. Please contact the merchant.	Concord EFS – The store key is invalid.
2	174	The transaction type is invalid. Please contact the merchant.	Concord EFS – This transaction type is not accepted by the processor.
3	175	The processor does not allow voiding of credits.	Concord EFS – This transaction is not allowed. The Concord EFS processing platform does not support voiding credit transactions. Please debit the credit card instead of voiding the credit.
3	180	An error occurred during processing. Please try again.	The processor response format is invalid.
3	181	An error occurred during processing. Please try again.	The system-generated void for the original invalid transaction failed. (The original transaction included an invalid processor response format.)
3	185	This reason code is reserved or not applicable to this API.	
4	193	The transaction is currently under review.	The transaction was placed under review by the risk management system.
2	200	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The credit card number is invalid.
2	201	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The

RESPONSE CODE	RESPONSE REASON CODE	RESPONSE REASON TEXT	NOTES
			expiration date is invalid.
2	202	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The transaction type is invalid.
2	203	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The value submitted in the amount field is invalid.
2	204	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The department code is invalid.
2	205	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The value submitted in the merchant number field is invalid.
2	206	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The merchant is not on file.
2	207	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The merchant account is closed.
2	208	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The merchant is not on file.
2	209	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. Communication with the processor could not be established.
2	210	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The merchant type is incorrect.
2	211	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The cardholder is not on file.
2	212	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The bank configuration is not on file
2	213	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The merchant assessment code is incorrect.

RESPONSE CODE	RESPONSE REASON CODE	RESPONSE REASON TEXT	NOTES
2	214	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. This function is currently unavailable.
2	215	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The encrypted PIN field format is invalid.
2	216	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The ATM term ID is invalid.
2	217	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. This transaction experienced a general message format problem.
2	218	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The PIN block format or PIN availability value is invalid.
2	219	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The ETC void is unmatched.
2	220	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The primary CPU is not available.
2	221	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. The SE number is invalid.
2	222	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. Duplicate auth request (from INAS).
2	223	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. This transaction experienced an unspecified error.
2	224	This transaction has been declined.	This error code applies only to merchants on FDC Omaha. Please re-enter the transaction.
3	243	Recurring billing is not allowed for this eCheck.Net type.	The combination of values submitted for x_recurring_billing and x_echeck_type is not allowed.

RESPONSE CODE	RESPONSE REASON CODE	RESPONSE REASON TEXT	NOTES
3	244	This eCheck.Net type is not allowed for this Bank Account Type.	The combination of values submitted for x_bank_acct_type and x_echeck_type is not allowed.
3	245	This eCheck.Net type is not allowed when using the payment gateway hosted payment form.	The value submitted for x_echeck_type is not allowed when using the payment gateway hosted payment form.
3	246	This eCheck.Net type is not allowed.	The merchant's payment gateway account is not enabled to submit the eCheck.Net type.
3	247	This eCheck.Net type is not allowed.	The combination of values submitted for x_type and x_echeck_type is not allowed.
3	248	The check number is invalid.	Invalid check number. Check number can only consist of letters and numbers and not more than 15 characters.
2	250	This transaction has been declined.	This transaction was submitted from a blocked IP address.
2	251	This transaction has been declined.	The transaction was declined as a result of triggering a Fraud Detection Suite filter.
4	252	Your order has been received. Thank you for your business!	The transaction was accepted, but is being held for merchant review. The merchant can customize the customer response in the Merchant Interface.
4	253	Your order has been received. Thank you for your business!	The transaction was accepted and was authorized, but is being held for merchant review. The merchant can customize the customer response in the Merchant Interface.
2	254	Your transaction has been declined.	The transaction was declined after manual review.
3	261	An error occurred during processing. Please try again.	The transaction experienced an error during sensitive data encryption and was not processed. Please try again.
3	270	The line item [item number] is	A value submitted in x_line_item

RESPONSE CODE	RESPONSE REASON CODE	RESPONSE REASON TEXT	NOTES
		invalid.	for the item referenced is invalid.
3	271	The number of line items submitted is not allowed. A maximum of 30 line items can be submitted.	The number of line items submitted exceeds the allowed maximum of 30.
3	288	Merchant is not registered as a Cardholder Authentication participant. This transaction cannot be accepted.	The merchant has not indicated participation in any Cardholder Authentication Programs in the Merchant Interface.
3	289	This processor does not accept zero dollar authorization for this card type.	Your credit card processing service does not yet accept zero dollar authorizations for Visa credit cards. You can find your credit card processor listed on your merchant profile.
3	290	One or more required AVS values for zero dollar authorization were not submitted.	When submitting authorization requests for Visa, the address and zip code fields must be entered.
4	295	The amount of this request was only partially approved on the given prepaid card. Additional payments are required to complete the balance of this transaction.	The merchant must have partial authorization enabled in the merchant interface in order to receive this error.
3	296	The specified SplitTenderId is not valid.	
3	297	A Transaction ID and a Split Tender ID cannot both be used in a single transaction request.	
3	300	The device ID is invalid.	The value submitted for x_device_id is invalid.
3	301	The device batch ID is invalid.	The value submitted for x_device_batch_id is invalid.
3	302	The reversal flag is invalid.	The value submitted for x_reversal is invalid.
3	303	The device batch is full. Please close the batch.	The current device batch must be closed manually from the POS device.
3	304	The original transaction is in a closed batch.	The original transaction has been settled and cannot be reversed.
3	305	The merchant is configured for	This merchant is configured for auto-close and cannot manually

RESPONSE CODE	RESPONSE REASON CODE	RESPONSE REASON TEXT	NOTES
		auto-close.	close batches.
3	306	The batch is already closed.	The batch is already closed.
1	307	The reversal was processed successfully.	The reversal was processed successfully.
1	308	Original transaction for reversal not found.	The transaction submitted for reversal was not found.
3	309	The device has been disabled.	The device has been disabled.
1	310	This transaction has already been voided.	This transaction has already been voided.
1	311	This transaction has already been captured	This transaction has already been captured.
2	315	The credit card number is invalid.	This is a processor-issued decline.
2	316	The credit card expiration date is invalid.	This is a processor-issued decline.
2	317	The credit card has expired.	This is a processor-issued decline.
2	318	A duplicate transaction has been submitted.	This is a processor-issued decline.
2	319	The transaction cannot be found.	This is a processor-issued decline.

Note: A very helpful tool for troubleshooting errors is available in our Integration Center at http://developer.authorize.net/tools/responsereasoncode.

Email Receipt

Merchants can choose to send a payment gateway-generated email receipt to customers who provide an email address with their transaction. The email receipt includes a summary and results of the transaction. To the customer, this email appears to be sent from the merchant contact that is configured as the Email Sender in the Merchant Interface.

To send the payment gateway-generated customer email receipt, submit the API fields in the following table, with the transaction request string. These settings can also be configured in the Merchant Interface.

For more information about configuring these settings, please see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.

Fields are name/value pairs with the syntax of:

```
x_name_of_field=value of the field&.
```

FIELD NAME	VALUE	FORMAT	NOTES
x_email	The customer's valid email address	Up to 255 characters Ex. janedoe@customer.com	The email address to which the customer's copy of the email receipt is sent when Email Receipts is configured in the Merchant Interface. The email is sent to the customer only if the email address format is valid. For more information about Email Receipts, please see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.
x_email_customer	The customer email receipt status	TRUE, FALSE, T, F, YES, NO, Y, N, 1, 0	Indicates whether an email receipt should be sent to the customer. If set to TRUE, the payment gateway will send an email to the customer after the transaction is processed using the customer email address submitted with the transaction. If FALSE, no email is sent to the customer. If no value is submitted, the payment gateway will look up the configuration in the Merchant Interface and send an email only if the merchant has enabled the setting. If this field is not submitted and the setting is disabled in the Merchant Interface, no email is sent. For more information about configuring Email Receipts in the Merchant Interface, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.
x_header_email_receipt	The email receipt header	Plain text	This text will appear as the header of the email receipt sent to the customer.

FIELD NAME	VALUE	FORMAT	NOTES
x_footer_email_receipt	The email receipt footer	Plain text	This text will appear as the footer on the email receipt sent to the customer.
x_merchant_email	Any valid email address	Up to 255 characters Only one email address per variable is allowed	Email address to which the merchant's copy of the customer confirmation email should be sent. If a value is submitted, an email will be sent to this address as well as the address(es) configured in the Merchant Interface. Warning: If it is included, it can subject the merchant to spam on their business email address, because it announces where the receipt gets returned to, and gives a hint where relay response or silent post information may be sent.

In addition, the merchant may receive a transaction confirmation email from the payment gateway at the completion of each transaction, which includes order information and the results of the transaction. Merchants can sign up for confirmation emails in the Merchant Interface. For more information, please see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.

Section 5
Test Transactions

You need to test the payment gateway integration carefully before going live to ensure successful and smooth transaction processing.

Ideally, an integration is tested in the following phases:

1. First, using an Authorize.Net developer test account.

   In this environment, test transactions are posted to **https://test.authorize.net/gateway/transact.dll**. Although this is a staging environment, its behavior mimics the live payment gateway. Transactions submitted to the test environment using a developer test account are **not** submitted to financial institutions for authorization and are not stored in the Merchant Interface.

   In order to use this environment, you must have an Authorize.Net developer test account with an associated API Login ID and Transaction Key. Test transactions to this environment are accepted with these credentials only. If you do not have a developer test account, you can sign up for one at http://developer.authorize.net/testaccount.

   Note: You do not need to use Test Mode when testing with a developer test account. For more information about Test Mode, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.

2. Once the integration is successfully tested in the developer test environment, the merchant's Authorize.Net Payment Gateway API Login ID and Transaction Key can be plugged into the integration for testing against the live environment. (Developer test account credentials will not be accepted by the live payment gateway.) In this phase, testing can be done in one of two ways:

   a. By including the *x_test_request* field with a value of "TRUE" in the transaction request string to **https://secure.authorize.net/gateway/transact.dll**. See the sample below.

   Sample 1. Submitting the test request field

   ```
   <INPUT TYPE="HIDDEN" NAME="x_test_request"
   VALUE="TRUE">
   ```

   b. By placing the merchant's payment gateway account in Test Mode in the Merchant Interface. New payment gateway accounts are placed in Test Mode by default. For more information about Test Mode, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/. When processing test transactions in Test Mode, the payment gateway will return a transaction ID of "0." This means you cannot test follow-on transactions such as credits, voids, etc., while in Test Mode. To test follow-on transactions, you can either submit x_test_request=TRUE as indicated above, or process a test transaction with any valid credit card number in live mode, as explained below.

Note: Transactions posted against live merchant accounts using either of the above testing methods are not submitted to financial institutions for authorization and are not stored in the Merchant Interface.

3. If testing in the live environment is successful, you are ready to submit live transactions and verify that they are being submitted successfully. Either remove the *x_test_request* field from the transaction request string, or set it to "FALSE;" or, if you are using Test Mode, turn it off in the Merchant Interface. To receive a true response, you must submit a transaction using a real credit card number. You can use any valid credit card number to submit a test transaction. You can void successful transactions immediately to prevent live test transactions from being processed. This can be done quickly on the Unsettled Transactions page of the Merchant Interface. It is recommended that when testing using a live credit card, you use a nominal value, such as $0.01. That way, if you forget to void the transaction, the impact will be minimal. For VISA verification transactions, submit a $0.00 value instead, if the processor accepts it.

Note: VISA verification transactions are being switched from $0.01 to $0.00 for all processors. For Visa transactions using $0.00, the Bill To address (x_address) and zip code (x_zip) fields are required.

Testing to Generate Specific Transaction Results

When testing transaction results in the developer test environment as well as the production environment, you can produce a specific response reason code by submitting a test transaction that uses a test credit card number designed to generate specific transaction results: this is Visa test credit card number "4222222222222." This card number is intended for testing and should only be used for that purpose. Submit the test transaction by either placing the account in Test Mode, or submitting x_test_request=TRUE, with a dollar amount value equal to the response reason code you would like to produce.

For example, to test the AVS response reason code number 27, submit the test transaction with the credit card number "4222222222222" and the amount "27.00."

To test the AVS or CCV responses in the live environment, you need to submit live transactions with correct street address, ZIP Code and Card Code information to generate successful responses, and incorrect street address, ZIP Code and Card Code information to generate other responses. You can void successful transactions immediately to prevent live test transactions from being processed. This can be done quickly on the Unsettled Transactions page of the Merchant Interface. It is not possible to test the AVS or CCV responses in the developer test environment. For more information about AVS, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.

For more information about response reason codes, see the "Transaction Response" section of this guide.

Appendix A
Fields by Transaction Type

This appendix provides a complete listing of all API fields that should be submitted for each transaction type supported for AIM. It is divided into the following sections:

- the minimum fields required to submit a transaction,
- additional fields that are required in order to configure advanced features of AIM
- "best practice" fields, or fields that the payment gateway recommends should be submitted on a per-transaction basis in order to maintain a strong connection to the payment gateway—for example, to prevent possible conflicts if integration settings in the Merchant Interface are inadvertently changed.

Minimum Required Fields

The following table provides a quick reference of all API fields that are required for each transaction type supported for AIM.

	AUTHORIZATION AND CAPTURE	**AUTHORIZATION ONLY**	**PRIOR AUTHORIZATION AND CAPTURE**	**CAPTURE ONLY**	**CREDIT**	**VOID**
Merchant Information	x_login x_tran_key	x_login x_tran_key	x_login x_tran_key	x_login x_tran_key	x_login x_tran_key	x_login x_tran_key
Transaction Information	x_type = AUTH_CAPTURE	x_type = AUTH_ONLY	x_type = PRIOR_AUTH _ CAPTURE x_trans_id or x_split_tender_id	x_type = CAPTURE_ONLY x_auth_code	x_type = CREDIT x_trans_id	x_type = VOID x_trans_id or x_split_tender_id
Payment Information	x_amount x_card_num x_exp_date	x_amount x_card_num x_exp_date	x_amount (required only when less than the original authorization amount)	x_amount x_card_num x_exp_date	x_amount x_card_num x_exp_date**	N/A

* For merchants enabled for expanded credit capabilities (ECC), a Transaction ID should NOT be submitted for Unlinked Credits. For more information, see the "Credit Card Transaction Types" section of this document.

** The expiration date is only required for Unlinked Credits.

Required Fields for Additional AIM Features

The following table provides a quick reference of additional fields that are required for advanced features of AIM and that *cannot* be configured in the Merchant Interface. For example, if the merchant wants to submit itemized order information, you must submit fields in addition to the minimum required fields.

	AUTHORIZATION AND CAPTURE	**AUTHORIZATION ONLY**	**PRIOR AUTHORIZATION AND CAPTURE**	**CAPTURE ONLY**	**CREDIT**	**VOID**
Itemized Order Information	x_line_item x_tax† x_freight† x_duty†	x_line_item x_tax† x_freight† x_duty†	x_line_item x_tax† x_freight† x_duty†	x_line_item x_tax† x_freight† x_duty†	x_line_item x_tax† x_freight† x_duty†	N/A
Cardholder Authentication	x_authentication_ indicator x_cardholder_authentication_value	x_authentication_ indicator x_cardholder_authentication_value	N/A	N/A	N/A	N/A
Fraud Detection Suite™ (FDS)	x_customer_ip (required only when the merchant is using the FDS IP blocking tool)	x_customer_ip (required only when the merchant is using the FDS IP blocking tool)	N/A	N/A	N/A	N/A

† These fields can support either a straight numeric value, or line item details similar to x_line_item.

Note: For Prior Authorization and Capture transactions, if line item information was submitted with the original transaction, adjusted information can be submitted if the transaction changed. If no adjusted line item information is submitted, the information submitted with the original transaction will apply.

Best Practice Fields

The following table provides a quick reference of additional API fields that the payment gateway highly recommends should be submitted on a per-transaction basis in order to maintain a strong connection.

	AUTHORIZATION AND CAPTURE	**AUTHORIZATION ONLY**	**PRIOR AUTHORIZATION AND CAPTURE**	**CAPTURE ONLY**	**CREDIT**	**VOID**
Transaction Information	x_version = 3.1	x_version = 3.1	x_version = 3.1	x_version = 3.1	x_version = 3.1	x_version = 3.1
Transaction Response	x_delim_data = TRUE x_delim_char x_encap_char x_relay_response = FALSE*	x_delim_data = TRUE x_delim_char x_encap_char x_relay_response = FALSE*	x_delim_data = TRUE x_delim_char x_encap_char x_relay_response = FALSE*	x_delim_ data = TRUE x_delim_ char x_encap_ char x_relay_response = FALSE*	x_delim_data = TRUE x_delim_char x_encap_char x_relay_esponse = FALSE*	x_delim_data = TRUE x_delim_char x_encap_char x_relay_response = FALSE*

* The *x_relay_response* field is not technically an AIM feature; however, it is recommended that you submit this field on a per-transaction basis with the value of “FALSE” as a best practice to further define the AIM transaction format.

Appendix B
Alphabetized List of API Fields

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
x_address	Optional	The customer's billing address	Up to 60 characters (no symbols)	Required if the merchant would like to use the Address Verification Service security feature. For more information on AVS, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/. Required for Zero Dollar Authorizations for Visa verification transactions.
x_allow_partial_Auth	Optional	True, False	True, False, T, F	Set this value if the merchant would like to override a setting in the Merchant Interface.
x_amount	Required if x_type = AUTH_ CAPTURE, AUTH_ONLY CAPTURE_ ONLY, CREDIT	The amount of the transaction	Up to 15 digits with a decimal point (no dollar symbol) Ex. 8.95	The total amount to be charged or credited *including* tax, shipping and any other charges. The amount can either be hard coded or posted to a script.
x_auth_code	Required if x_type = CAPTURE_ ONLY	The authorization code for an original transaction not authorized on the payment gateway	6 characters	Required only for CAPTURE_ONLY transactions. See the "Credit Card Transaction Types" section of this document.
x_authentication_ indicator	Optional	The electronic commerce indicator (ECI) value for a Visa transaction; or	Special characters included in this value must be URL	Required only for AUTH_ONLY and AUTH_CAPTURE transactions processed through cardholder authentication programs. When submitted with other

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
		the universal cardholder authentication field indicator (UCAF) for a MasterCard transaction obtained by the merchant after the authentication process.	encoded.	transaction types, this value is ignored. This field is currently supported through Chase Paymentech, FDMS Nashville, Global Payments and TSYS.
x_card_code	Optional	The customer's card code	Must be a valid CVV2, CVC2 or CID value.	The three- or four-digit number on the back of a credit card (on the front for American Express). This field is required if the merchant would like to use the Card Code Verification (CCV) security feature. For more information, please see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.
x_card_num	Required if x_type = AUTH_ CAPTURE, AUTH_ONLYCAPTURE_ ONLY, CREDIT	The customer's credit card number When x_type= CREDIT, only the last four digits are required.	Between 13 and 16 digits without spaces	This is sensitive cardholder information and must be stored securely and in accordance with the Payment Card Industry (PCI) Data Security Standard. For more information about PCI, please see the http://www.authorize.net/files/developerbestpractices.pdf.
x_cardholder_ authentication_value	Optional	The cardholder authentication verification value (CAVV) for a Visa transaction; or accountholder authentication value (AVV)/ universal cardholder authentication	Special characters included in this value must be URL encoded	Required only for AUTH_ONLY and AUTH_CAPTURE transactions processed through cardholder authentication programs. When submitted with other transaction types, this value is ignored. This field is currently supported through Chase Paymentech, FDMS Nashville, Global Payments and TSYS.

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
		field (UCAF) for a MasterCard transaction obtained by the merchant after the authentication process.		
x_city	Optional	The city of the customer's billing address	Up to 40 characters (no symbols)	
x_company	Optional	The company associated with the customer's billing address	Up to 50 characters (no symbols)	
x_country	Optional	The country of the customer's billing address	Up to 60 characters (no symbols)	
x_cust_id	Optional	The merchant assigned customer ID	Up to 20 characters (no symbols)	The unique identifier to represent the customer associated with the transaction. The customer ID must be created dynamically on the merchant server or provided on a per-transaction basis. The payment gateway does not perform this function.
x_customer_ip	Optional	The customer's IP address	Up to 15 characters (no letters) Ex. 255.255.255.255	The IP address of the customer initiating the transaction. If this value is not passed, it will default to 255.255.255.255. This field is required when using the Fraud Detection Suite™ (FDS) IP Address Blocking tool. For more information about FDS, see the *Merchant Integration Guide* at http://www.authorize.net/support/merc

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
				hant/.
x_delim_char	Optional	The delimiting character	Ex. , (comma) | (pipe) " (double quotes) ' (single quote) : (colon) ; (semicolon) / (forward slash) \ (back slash) - (dash) * (star)	The character that is used to separate fields in the transaction response. The payment gateway will use the character passed in this field or the value stored in the Merchant Interface if no value is passed. If this field is passed, and the value is null, it will override the value stored in the Merchant Interface and there is no delimiting character in the transaction response. It is recommended that you submit this field on a per-transaction basis to be sure that transaction responses are returned in the correct format.
x_delim_data	Required for AIM transactions	The request to receive a delimited transaction response	TRUE, FALSE, T, F, YES, NO, Y, N, 1, 0	In order to receive a delimited response from the payment gateway, this field must be submitted with a value of TRUE or the merchant has to configure a delimited response through the Merchant Interface. It is recommended that you submit this field on a per-transaction basis to be sure that transaction responses are returned in the correct format. This field is paired with x_relay_response. If one is set to True, the other must be set to False.
x_description	Optional	The transaction description	Up to 255 (no symbols)	The description must be created dynamically on the merchant server or provided on a per-transaction basis. The payment gateway does not perform this function.

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
x_duplicate_window	Optional	The window of time after the submission of a transaction that a duplicate transaction can not be submitted	Any value between 0 and 28800 (no commas)	Indicates in seconds the window of time after a transaction is submitted during which the payment gateway will check for a duplicate transaction. The maximum time allowed is 8 hours (28800 seconds). If a value less than 0 is sent, the payment gateway will default to 0 seconds. If a value greater than 28800 is sent, the payment gateway will default to 28800. If no value is sent, the payment gateway will default to 2 minutes (120 seconds). If this field is present in the request with or without a value, an enhanced duplicate transaction response is sent. See the "Response for Duplicate Transactions" section of this guide for more information.
x_duty	Optional	The valid duty amount OR delimited duty information	When submitting delimited duty information, values must be delimited by a bracketed pipe <|>	The duty amount charged OR when submitting this information using the transaction request, delimited duty information including the duty name, description, and amount is also allowed. The total amount of the transaction in x_amount must *include* this amount.
		duty item name<|>		The duty item name.
		duty description<|>		The duty item description.
		duty amount	The dollar sign ($) is not allowed when submitting delimited	The duty amount. The total amount of the transaction in x_amount must *include* this amount.

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
			information .	
	Example: x_duty=Duty1<|>export<|>15.00&			
x_email	Optional	The customer's valid email address	Up to 255 characters Ex. janedoe@customer.com	The email address to which the customer's copy of the email receipt is sent when Email Receipts is configured in the Merchant Interface. The email is sent to the customer only if the email address format is valid. For more information about Email Receipts, please see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.
x_email_customer	Optional	The customer email receipt status	TRUE, FALSE, T, F, YES, NO, Y, N, 1, 0	Indicates whether an email receipt should be sent to the customer. If set to TRUE, the payment gateway will send an email to the customer after the transaction is processed using the customer email address submitted with the transaction. If FALSE, no email is sent to the customer. If no value is submitted, the payment gateway will look up the configuration in the Merchant Interface and send an email only if the merchant has enabled the setting. If this field is not submitted and the setting is disabled in the Merchant Interface, no email is sent. For more information about configuring Email Receipts in the Merchant Interface, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.
x_encap_char	Optional	The encapsulat	Ex. | (pipe) " (double	The character that is used to encapsulate the fields in the

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
		ing character	quotes) ‘ (single quote) : (colon) ; (semicolon) / (forward slash) \ (back slash) - (dash) * (star)	transaction response. This is only necessary if it is possible that your delimiting character could be included in any field values. The payment gateway will use the character passed in this field or the value stored in the Merchant Interface if no value is passed.
x_exp_date	Required	The customer’s credit card expiration date	MMYY, MM/YY, MM-YY, MMYYYY, MM/YYYY, MM-YYYY	This is sensitive cardholder information and must be stored securely and in accordance with the Payment Card Industry (PCI) Data Security Standard. For more information about PCI, please see the *Developer Security Best Practices White Paper* at http://www.authorize.net/files/developerbestpractices.pdf.
x_fax	Optional	The fax number associated with the customer’s billing address	Up to 25 digits (no letters) Ex. (123)123-1234.	
x_first_name	Optional	The first name associated with the customer’s billing address	Up to 50 characters (no symbols)	
x_footer_email_ receipt	Optional	The email receipt footer	Plain text	This text will appear as the footer on the email receipt sent to the customer.
x_freight	Optional	The valid freight amount OR delimited freight	When submitting delimited freight information, values must be delimited	The freight amount charged OR when submitting this information using the transaction request string, delimited freight information including the freight name, description, and amount is also

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
		information	by a bracketed pipe <|>	allowed. The total amount of the transaction in x_amount must *include* this amount.
		freight item name<|>		The freight item name.
		freight description <|>		The freight item description.
		freight amount	The dollar sign ($) is not allowed when submitting delimited information.	The freight item amount. The total amount of the transaction in x_amount must *include* this amount.
	Example: x_freight=Freight<|>ground overnight<|>12.95&			
x_header_email_ receipt	Optional	The email receipt header	Plain text	This text will appear as the header of the email receipt sent to the customer.
x_invoice_num	Optional	The merchant assigned invoice number for the transaction	Up to 20 characters (no symbols)	The invoice number must be created dynamically on the merchant server or provided on a per-transaction basis. The payment gateway does not perform this function.
x_last_name	Optional	The last name associated with the customer's billing address	Up to 50 characters (no symbols)	
x_line_item	Optional All line item values are required when this field is submitted	Any string	Line item values must be delimited by a bracketed pipe <|>	Itemized order information.
		Item ID<|>	Up to 31 characters	The ID assigned to an item.

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
		<|>item name<|>	Up to 31 characters	A short description of an item.
		<|>item description<|>	Up to 255 characters	A detailed description of an item.
		<|>itemX quantity<|>	Up to two decimal places Must be a positive number	The quantity of an item.
		<|>item price (unit cost)<|>	Up to two decimal places Must be a positive number	Cost of an item per unit, *excluding* tax, freight, and duty. The dollar sign ($) is not allowed when submitting delimited information.
		<|>itemX taxable<|>	TRUE, FALSE, T, F, YES, NO, Y, N, 1, 0	Indicates whether the item is subject to tax.
x_login	Required	The merchant's unique API Login ID	Up to 20 characters	The merchant API Login ID is provided in the Merchant Interface. The API Login ID and Transaction Key together provide the merchant authentication required for access to the payment gateway. See the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/ for more information.
x_merchant_email	Optional	Any valid email address	Up to 255 characters Only one email address per	Email address to which the merchant's copy of the customer confirmation email should be sent. If a value is submitted, an email will be sent to this address as well as the address(es) configured

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
			variable is allowed	in the Merchant Interface. **Warning**: If included, it can subject the merchant to spam on their business email address, because it announces where the receipt gets returned to, and gives a hint where relay response or silent post information may be sent.
x_method	Optional	The payment method	CC or ECHECK	The method of payment for the transaction, CC (credit card) or ECHECK (electronic check). If left blank, this value will default to CC. For more information about eCheck.Net transaction requirements, see the eCheck.Net Developer Guide.
x_phone	Optional	The phone number associated with the customer's billing address	Up to 25 digits (no letters) Ex. (123)123-1234	
x_po_num	Optional	The merchant assigned purchase order number	Up to 25 characters (no symbols)	The purchase order number must be created dynamically on the merchant server or provided on a per-transaction basis. The payment gateway does not perform this function.
x_recurring_billing	Optional	The recurring billing status	TRUE, FALSE, T, F, YES, NO, Y, N, 1, 0	Indicating marker used by merchant account providers to identify transactions which originate from merchant hosted recurring billing applications. This value is not affiliated with Automated Recurring Billing.
x_relay_response	Optional	The request for a relay response	TRUE, FALSE, T, F, YES, NO, Y, N, 1, 0	This field, when set to TRUE, instructs the payment gateway to return transaction results to the merchant by means of an HTML form POST to the merchant's Web server for a

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
				relay response. Relay response is used for SIM applications. Set this to false if you are using AIM.
x_ship_to_address	Optional	The customer's shipping address	Up to 60 characters (no symbols)	
x_ship_to_company	Optional	The company associated with the customer's shipping address	Up to 50 characters (no symbols)	
x_ship_to_country	Optional	The country of the customer's shipping address	Up to 60 characters (no symbols)	
x_ship_to_city	Optional	The city of the customer's shipping address	Up to 40 characters (no symbols)	
x_ship_to_first_name	Optional	The first name associated with the customer's shipping address	Up to 50 characters (no symbols)	
x_ship_to_last_name	Optional	The last name associated with the customer's shipping address	Up to 50 characters (no symbols)	
x_ship_to_state	Optional	The state of the customer's shipping	Up to 40 characters (no symbols) or a valid two-	

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
		address	character state code	
x_ship_to_zip	Optional	The ZIP code of the customer's shipping address	Up to 20 characters (no symbols)	
x_split_tender_id	Optional	The payment gateway-assigned ID that links the current authorization request to the original authorization request.	Numeric	This value is returned in the reply message from the original authorization request. Transmit this value in subsequent transactions that are related to the same order.
x_state	Optional	The state of the customer's billing address	Up to 40 characters (no symbols) or a valid two-character state code	
x_tax	Optional	The valid tax amount OR the delimited tax information	When submitting delimited tax information, values must be delimited by a bracketed pipe <|>	The tax amount charged OR when submitting this information using the transaction request string, delimited tax information including the sales tax name, description, and amount is also allowed. The total amount of the transaction in x_amount must *include* this amount.
		tax item name<|>		The tax item name.
		tax description <|>		The tax item description.

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
		tax amount	The dollar sign ($) is not allowed when submitting delimited information.	The tax item amount. The total amount of the transaction in x_amount must *include* this amount.
	Example: x_tax=Tax1<|>state tax<|>0.09&			
x_tax_exempt	Optional	The tax exempt status	TRUE, FALSE, T, F, YES, NO, Y, N, 1, 0	Indicates whether the transaction is tax exempt.
x_test_request	Optional	The request to process test transactions	TRUE, FALSE, T, F, YES, NO, Y, N, 1, 0	Indicates if the transaction should be processed as a test transaction. See the “Test Transactions” section of this guide for more information.
x_tran_key	Required for merchant authentication	The merchant's unique Transaction Key	16 characters	The merchant Transaction Key is provided in the Merchant Interface and must be stored securely. The API Login ID and Transaction Key together provide the merchant authentication required for access to the payment gateway. See the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.
x_trans_id	Required when x_type = CREDIT, PRIOR_AUTH_CAPTURE, VOID	The payment gateway assigned transaction ID of the original transaction	Numeric	Required only for CREDIT, PRIOR_AUTH_CAPTURE, and VOID transactions. For more information about transaction types, see the “Credit Card Transaction Types” section of this guide. Do not include this field if you include the x_split_tender_id field.

FIELD NAME	REQUIRED	VALUE	FORMAT	NOTES
x_type	Optional	The type of credit card transaction	AUTH_CAPTURE (default), AUTH_ONLY, CAPTURE_ ONLY, CREDIT, PRIOR_AUTH_ CAPTURE, VOID.	If the value submitted does not match a supported value, the transaction is rejected. If this field is not submitted or the value is blank, the payment gateway will process the transaction as an AUTH_CAPTURE.
x_version	Optional, but highly recommended	The merchant's transaction version	3.0, 3.1	Indicates to the system the set of fields that will be included in the response: 3.0 is the default version. 3.1 allows the merchant to utilize partial authorizations and the Card Code feature, and is the current standard version. It is highly recommended that you submit this field on a per-transaction basis to be sure that the formats of transaction requests and the responses you receive are consistent.
x_zip	Optional	The ZIP code of the customer's billing address	Up to 20 characters (no symbols)	Required if the merchant would like to use the Address Verification Service security feature. For more information on AVS, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/. Required for Zero Dollar Authorizations for Visa verification transactions.