Merchant Web Services API

Automated Recurring Billing™ (ARB)

XML Guide

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 do not 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:

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®

Fraud Detection Suite™

FraudScreen.Net®

Table of Contents

Revision History

PUBLISH DATE	UPDATES
January 2007	Initial release of the Automated Recurring Billing (ARB) API
August 3, 2007	Updated description for response code E00012
September 24, 2007	No end date/Silent Post URL updates
May 2008	Removal of SecureSource requirements and various updates
	Clarification of subscription start date validation Add ERR00045 Clarify CardCode element usage
October 2009	Correct hyperlinks

Section 1

Developer Introduction

This guide describes the Web development required to submit Automated Recurring Billing™ (ARB), or subscription-based, payments to the Authorize.Net Payment Gateway directly from a Web site or other application using extensible markup language (XML).

Specifically, the Authorize.Net ARB Application Programming Interface (API) provides a mechanism for developers and value added resellers (VARs) to create, update and cancel ARB subscriptions by means of direct integration between client software or applications and the Authorize.Net Payment Gateway.

A subscription is a set of multiple transactions, or payments, created for the purchase of a subscription-based product or service or for an installment-based payment plan. Payments for the subscription are then generated by the payment gateway at later dates based on a specified payment schedule and subscription duration.

The behavior of the ARB API is the same as when a merchant creates, updates, and cancels ARB subscriptions in the Merchant Interface. When a merchant creates a subscription in the Merchant Interface, they enter all required information (customer payment information, subscription interval and duration, etc.) into the Create New ARB Subscription form. When the merchant submits the information, the Subscription Confirmation page returns a message to the merchant regarding whether or not the subscription was created successfully. The subscription ID assigned for a successfully created subscription is also displayed.

The ARB API accomplishes these same functions through an XML call and subsequent XML response. Whether a subscription is created in the Merchant Interface or through the ARB API, the results are the same.

Note:	You may want to log on to the Merchant Interface to step through the manual ARB process. You can request a developer test account from our Integration Center at http://developer.authorize.net/testaccount. Be sure to include in the comments section that you need the ARB feature enabled for your test account. If you already have a test account, please email integration@authorize.net to request that ARB be enabled for your test account.

Minimum Requirements

Before you begin ARB integration for an Authorize.Net Payment Gateway account, please check with the merchant to make sure that the following minimum requirements have already been met.

- **The merchant must have a merchant bank account that allows Internet transactions.**

- **The merchant must have an active Authorize.Net Card Not Present Payment Gateway account.**

- **The merchant must be signed up for the Authorize.Net ARB service.**

- **The merchant must store account authentication data securely (for example, API login ID, transaction key).**

Note:	Merchants should avoid storing any type of sensitive cardholder information. However, in the event that a merchant or third party must store sensitive customer business or payment information, compliance with industry standard storage requirements is required. Please see the *Developer Security Best Practices White Paper* at http://www.authorize.net/files/developerbestpractices.pdf for guidelines.

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

+ If you can't find what you need in the Integration Center, our Integration Team is available to answer your questions by email at integration@authorize.net.

+ Be sure to read our *Developer Security Best Practices White Paper* at http://www.authorize.net/files/developerbestpractices.pdf 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.

Section 2

Executing an API Call

The following sections describe the minimum requirements for executing an API call for an ARB subscription request using XML.

There are two options for developing the subscription request script:

+ You can develop a custom script yourself using the API fields information in this section, OR

+ You can use Authorize.Net sample code in C#, Java, PHP, Ruby, and VBNet available for free from our Integration Center at http://developer.authorize.net/samplecode.

Note: If you choose to use Authorize.Net sample code, please be aware that in order to achieve a successful implementation it **must** be modified with the merchant's specific payment gateway account information.

ARB API URLs

ITEM	LOCATION
Production	https://api.authorize.net/xml/v1/request.api
Developer Test	https://apitest.authorize.net/xml/v1/request.api
XML Schema	https://api.authorize.net/xml/v1/schema/AnetApiSchema.xsd

In order to be processed successfully, API requests and responses must conform to the ARB API XML schema.

Note: The Developer Test URL requires the use of a developer test payment gateway account. You can request a test account from our Integration Center at http://developer.authorize.net/testaccount. Developer test accounts cannot be used to test against the Production URL.

ARB Subscription Functions

The ARB API includes the following functions:

+ ARBCreateSubscriptionRequest

+ ARBUpdateSubscriptionRequest

+   ARBCancelSusbscriptionRequest

Each API submission may contain only one ARB request. Including more than one request per submission will result in an error.

Authentication

ALL calls to the ARB API require merchant authentication. The following table represents the required XML elements. **All XML elements are case sensitive and must be submitted in the order listed here. Optional elements should not be submitted unless they contain valid values.**

ELEMENT	VALUE	FORMAT	NOTES
merchantAuthentication	Contains the merchant's payment gateway account authentication information		
name	The merchant's valid API login ID	Up to 25 characters	Submit the API login ID used to submit transactions.
transactionKey	The merchant's valid transaction key	16 characters	Submit the transaction key obtained by the merchant from the Merchant Interface.

Example of Authentication with the API Login ID and Transaction Key

```
<?xml version="1.0" encoding="utf-8"?>

<ARBCreateSubscriptionRequest xmlns=
"AnetApi/xml/v1/schema/AnetApiSchema.xsd">

  <merchantAuthentication>

    <name>mytestacct</name>

    <transactionKey>112223344</transactionKey>

  </merchantAuthentication>
```

Note: The sample code included in this document uses dummy field values. When using or testing sample code, be sure to enter valid field values. Additional sample code is available for download from the Authorize.Net Integration Center at http://developer.authorize.net/samplecode.

Input Elements for ARBCreateSubscriptionRequest

The following table represents the input elements for executing an API call to the ARBCreateSubscriptionRequest function, in addition to the authentication elements. Indentations in

the Element column indicate grouping hierarchy. **Elements are required unless otherwise indicated. All XML elements are case sensitive and must be submitted in the order listed here. Optional elements should not be submitted unless they contain valid values.**

ELEMENT	VALUE	FORMAT	NOTES
refId	Merchant-assigned reference ID for the request *Optional*	Up to 20 characters	If included in the request, this value will be included in the response. This feature might be especially useful for multi-threaded applications.
subscription	Contains information about the subscription		
name	Merchant-assigned name for the subscription *Optional*	Up to 50 characters	
paymentSchedule	Contains information about the payment schedule		
interval	Contains information about the interval of time between payments		
length	The measurement of time, in association with the Interval Unit, that is used to define the frequency of the billing occurrences	Up to 3 digits If the Interval Unit is "months," can be any number between one (1) and 12. If the Interval Unit is "days," can be any number between seven (7) and 365.	
unit	The unit of time, in association with the Interval Length, between each billing occurrence	days, months	

startDate	The date the subscription begins (also the date the initial billing occurs)	YYYY-MM-DD	The date entered must be greater than or equal to the date the subscription was created. The validation checks against local server date, which is Mountain Time. An error might possibly occur if you try to submit a subscription from a time zone where the resulting date is different; for example, if you are in the Pacific time zone and try to submit a subscription between 11:00 PM and midnight, with a start date set for today.
totalOccurrences	Number of billing occurrences or payments for the subscription	Up to 4 digits	To submit a subscription with no end date (an ongoing subscription), this field must be submitted with a value of "9999." If a trial period is specified, this number should include the Trial Occurrences.
trialOccurrences	Number of billing occurrences or payments in the trial period *Optional*	Up to 2 digits	If a trial period is specified, this number must be included in the Total Occurrences.
amount	The amount to be billed to the customer for each payment in the subscription	Up to 15 digits	If a trial period is specified, this is the amount that will be charged after the trial payments are completed.
trialAmount	The amount to be charged for each payment during a trial period *Conditional*	Up to 15 digits	Required when trial occurrences is specified. Once the number of trial occurrences for the subscription is complete, the regular amount will be charged for each remaining payment.
payment	Contains either the customer's credit card or bank account payment information		
creditCard	Contains the		This element should only be

	customer's credit card information		included when the payment method is credit card.
cardNumber	The credit card number used for payment of the subscription	13 to 16 digits	
expirationDate	The expiration date of the credit card used for the subscription	YYYY-MM	
cardCode	The three- or four-digit card code on the back of most credit cards, on the front for American Express *Optional*	3 or 4 digits	This element should only be included when the merchant has set the card code value field to **required** in the account settings. The value itself is never validated.
bankAccount	Contains the customer's bank account information		This element should only be included when the payment method is bank account.
accountType	The type of bank account used for payment of the subscription	checking, businessChecking, savings	
routingNumber	The routing number of the customer's bank	9 digits	
accountNumber	The bank account number used for payment of the subscription	5 to 17 digits	
nameOnAccount	The full name of the individual associated with the bank account number	Up to 22 characters	
echeckType	The type of electronic check transaction used for the subscription	For checking or savings accounts, PPD or WEB For business checking accounts, CCD	
bankName	The name of the bank associated with the	Up to 50	

	bank account number *Optional*	characters	
order	Contains optional order information *Optional*		
invoiceNumber	Merchant-assigned invoice number for the subscription *Optional*	Up to 20 characters	The invoice number will be associated with each payment in the subscription.
description	Description of the subscription *Optional*	Up to 255 characters	The description will be associated with each payment in the subscription.
customer	Contains information about the customer		
id	Merchant-assigned identifier for the customer *Optional*	Up to 20 characters	
email	The customer's email address *Optional*	Up to 255 characters	
phoneNumber	The customer's phone number *Optional*	Up to 25 digits	
faxNumber	The customer's fax number *Optional*	Up to 25 digits	
billTo	Contains the customer's billing address information		
firstName	The first name associated with the customer's billing address	Up to 50 characters	
lastName	The last name associated with the	Up to 50	

	customer's billing address	characters	
company	The company associated with the customer's billing address *Optional*	Up to 50 characters	
address	The customer's billing address *Optional*	Up to 60 characters	
city	The city of the customer's billing address *Optional*	Up to 40 characters	
state	The state of the customer's billing address *Optional*	2 characters Must be a valid state code	
zip	The ZIP code of the customer's billing address *Optional*	Up to 20 characters	
country	The country of the customer's billing address *Optional*	Up to 60 characters Must be a valid two-character country code or full country name (spelled in English).	
shipTo	Contains the customer's shipping address information *Optional*		
firstName	The first name associated with the customer's shipping	Up to 50 characters	

	address		
lastName	The last name associated with the customer's shipping address	Up to 50 characters	
company	The company associated with the customer's shipping address	Up to 50 characters	
address	The customer's shipping address	Up to 60 characters	
city	The city of the customer's shipping address	Up to 40 characters	
state	The state of the customer's shipping address	Up to 40 characters	
zip	The ZIP code of the customer's shipping address	Up to 20 characters	
country	The country of the customer's shipping address	Up to 60 characters Must be a valid two-character country code or full country name (spelled in English).	

Example ARBCreateSubscriptionRequest

```
<?xml version="1.0" encoding="utf-8"?>

<ARBCreateSubscriptionRequest
xmlns="AnetApi/xml/v1/schema/AnetApiSchema.xsd">

  <merchantAuthentication>

    <name>mytestacct</name>

    <transactionKey>112223344</transactionKey>

  </merchantAuthentication>
```

13

```
  <refId>Sample</refId>

  <subscription>

    <name>Sample subscription</name>

    <paymentSchedule>

      <interval>

        <length>1</length>

        <unit>months</unit>

      </interval>

      <startDate>2007-03-15</startDate>

      <totalOccurrences>12</totalOccurrences>

      <trialOccurrences>1</trialOccurrences>

    </paymentSchedule>

    <amount>10.29</amount>

    <trialAmount>0.00</trialAmount>

    <payment>

      <creditCard>

        <cardNumber>4111111111111111</cardNumber>

        <expirationDate>2008-08</expirationDate>

      </creditCard>

    </payment>

    <billTo>

      <firstName>John</firstName>

      <lastName>Smith</lastName>

    </billTo>

  </subscription>

</ARBCreateSubscriptionRequest>
```

Note: The sample code included in this document uses dummy field values. When using or testing sample code, be sure to enter valid field values. Additional sample code is available for download from the Authorize.Net Integration Center at http://developer.authorize.net/samplecode.

Input Elements for ARBUpdateSubscriptionRequest

The input elements for a request to update an ARB subscription are the same as the create an ARB subscription function with the following addition and exceptions. **All XML elements are case sensitive and must be submitted in the order listed here. Optional elements should not be submitted unless they contain valid values.**

- The subscriptionID of the subscription to be updated must be submitted.

ELEMENT	VALUE	FORMAT	NOTES
subscriptionId	The payment gateway assigned identification number for the subscription	Up to 13 digits	

- The subscription start date (subscription.paymentSchedule.startDate) may only be updated in the event that no successful payments have been completed.

- The subscription interval information (subscription.paymentSchedule.interval.length and subscription.paymentSchedule.interval.unit) may not be updated.

- The number of trial occurrences (subscription.paymentSchedule.trialOccurrences) may only be updated if the subscription has not yet begun or is still in the trial period.

- All other fields are optional.

Example ARBUpdateSubscriptionRequest

```
<?xml version="1.0" encoding="utf-8"?>

<ARBUpdateSubscriptionRequest
xmlns="AnetApi/xml/v1/schema/AnetApiSchema.xsd">

  <merchantAuthentication>

    <name>mytestacct</name>

    <transactionKey>112223344</transactionKey>

  </merchantAuthentication>

  <refId>Sample</refId>

  <subscriptionId>100748</subscriptionId>

  <subscription>

    <payment>

      <creditCard>
```

```
        <cardNumber>4111111111111111</cardNumber>

        <expirationDate>2010-08</expirationDate>

      </creditCard>

    </payment>

  </subscription>

</ARBUpdateSubscriptionRequest>
```

Note: The sample code included in this document uses dummy field values. When using or testing sample code, be sure to enter valid field values. Additional sample code is available for download from the Authorize.Net Integration Center at http://developer.authorize.net/samplecode.

Input Elements for ARBCancelSubscriptionRequest

The following table represents the input elements for executing an API call to the ARBCancelSubscriptionRequest function, in addition to the authentication elements. Indentations in the Element column indicate grouping hierarchy. **Elements are required unless otherwise indicated. All XML elements are case sensitive and must be submitted in the order listed here. Optional elements should not be submitted unless they contain valid values.**

ELEMENT	VALUE	FORMAT	NOTES
refID	Merchant-assigned reference ID for the request *Optional*		If included in the request, this value will be included in the response. This feature might be especially useful for multi-threaded applications.
subscriptionId	The payment gateway assigned identification number for the subscription	Up to 13 digits	

Example ARBCancelSubscriptionRequest

```
<ARBCancelSubscriptionRequest
xmlns="AnetApi/xml/v1/schema/AnetApiSchema.xsd">

  <merchantAuthentication>

    <name>mytestacct</name>

    <transactionKey>112223344</transactionKey>
```

```
      </merchantAuthentication>

      <refId>Sample</refId>

      <subscriptionId>100748</subscriptionId>

   </ARBCancelSubscriptionRequest>
```

Note:	The sample code included in this document uses dummy field values. When using or testing sample code, be sure to enter valid field values. Additional sample code is available for download from the Authorize.Net Integration Center at http://developer.authorize.net/samplecode.

Section 3
XML Responses

The transaction response from the payment gateway is a set of fields that provides information about the status of a request.

Responses for Successful Requests

The following sections describe the output elements that are returned for successful API calls.

Output Elements for ARBCreateSubscriptionResponse

The following table represents the output elements for a successful API call to the ARBCreateSubscriptionRequest function. Indentations in the Element column indicate grouping hierarchy.

ELEMENT	VALUE	FORMAT	NOTES
refID	Merchant-assigned reference ID for the request	Up to 20 characters	This element is included in the response only if it was included in the request.
messages	Contains information about the results of the request		
resultCode	Contains additional information about the results of the request	Ok	An "Ok" result code indicates that the request was processed and accepted without error.
message	Contains the result code and text		Any messages present are informational only.
code	I00001		
text	Successful		
subscriptionId	The payment gateway assigned identification number for the subscription	Up to 13 digits	

Example ARBCreateSubscriptionResponse

```xml
<?xml version="1.0" encoding="utf-8"?>

<ARBCreateSubscriptionResponse
xmlns="AnetApi/xml/v1/schema/AnetApiSchema.xsd">

  <refId>Sample</refId>

  <messages>

    <resultCode>Ok</resultCode>

    <message>

      <code>I00001</code>

      <text>Successful.</text>

    </message>

  </messages>

  <subscriptionId>100748</subscriptionId>

</ARBCreateSubscriptionResponse>
```

Once you receive a response from the payment gateway with an "Ok" result code, your subscription has been successfully created. The response will include the subscription ID assigned to that particular subscription. Individual transactions, or payments, for a subscription are generated automatically by the payment gateway according to the designated payment schedule and subscription duration. Each payment will only be viewable in the merchant's payment gateway account when it is actually generated.

For example, if a new subscription is created with a start date of June 6, with a monthly payment interval, the first payment for the subscription will not be viewable in the merchant's payment gateway account until June 6. All subsequent payments will be visible on their scheduled date (July 6 payment will be visible on July 6, August 6 on August 6, etc.).

Note: If you create a new subscription with the first payment scheduled for that same day, the initial payment for the subscription will actually be submitted the next business day.

Once each scheduled transaction in a subscription has been submitted, which is usually at 2 AM PST for ARB transactions, the merchant will receive an email from the payment gateway indicating the transaction status.

The merchant can also configure their account in the Merchant Interface to receive the following ARB emails:

- Daily Transaction Summary.

- Failed Transaction Notice – sent when a payment in a subscription declines or receives an error response from the processor.

- Subscription Due for Expiration – sent after the second to last payment in a subscription is submitted, to notify the merchant that the next payment is the final one in the subscription.

- Credit Card Expiration – sent immediately after the last possible successful payment in a subscription, to notify the merchant that the credit card expiration date will expire before the next scheduled payment in the subscription.

- Subscription Suspension – sent to notify the merchant that a subscription has been suspended. A subscription will be suspended if the first payment in the subscription is declined, rejected or receives an error response. Additionally, if a subscription is edited, for example payment or shipping information is changed, the subscription will be suspended if the first payment after the edits is declined, rejected or receives an error response.

- Subscription Termination – sent when a subscription is terminated. If a suspended subscription is not edited to fix the problem that caused the suspension, it will terminate on the next scheduled payment.

- Subscription Expiration – sent after a subscription has expired. Once expired, a subscription cannot be reactivated. Instead, a new subscription would have to be created.

Note: The Daily Transaction Summary email returns an Excel file in comma separated value (.csv) format. The merchant will receive Successful.csv, Failed.csv or both files.

To select which ARB emails to receive:

1. Log on to the Merchant Interface at https://secure.authorize.net or if you have a developer test account, log in at https://test.authorize.net

2. Click **User Administration** under Account in the main menu on the left

3. Select the user you would like to edit and click **Edit User**

4. Click **Edit Profile Information** under Profile and Security Settings

5. Under the Automated Recurring Billing (ARB) Emails section, click to select or deselect which emails the user should receive

6. Click **Submit** to save the changes

For more information on viewing subscriptions in the Merchant Interface or on the types of ARB emails the merchant can opt to receive, please see the Merchant Interface Online Help Files.

Transaction Response for Individual Payments in a Subscription

The payment gateway sends an email to the merchant for each transaction submitted in a subscription indicating the transaction's status. If you would like to receive a transaction response for each payment in a subscription, you must enable the Silent Post feature in the Merchant Interface. When this feature is enabled, the payment gateway will post a transaction response in name/value pair format to the URL specified in the Silent Post field of the Merchant Interface. The name/value pair response uses the syntax of *x_name_of_field=value of field&.*

IMPORTANT: When the Silent Post URL feature is enabled, name/value pair responses for both ARB transactions *and* all other non-ARB transactions will post to the specified URL. To determine which transaction responses are for ARB payments, you can parse the response for the *x_subscription_id* (Subscription ID) and the *x_subscription_paynum* (Subscription Payment Number) fields. These fields are only returned in ARB subscription transaction responses.

Note:	The Silent Post feature only returns responses for scheduled ARB transactions that are approved or declined. The payment gateway will not return a response to the specified Silent Post URL if the scheduled transaction results in a general error due to expired or invalid payment information. For more information see the "General Errors for Individual Payments in a Subscription" section of this guide.

For information on how to configure the Silent Post URL in the Merchant Interface, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.

The Silent Post feature does not happen in real time. The post will occur when the ARB transactions are submitted, which is usually around 2 AM PST. Because the post does not occur while an SSL connection is established, ARB transactions use an MD5 Hash calculation to validate each transaction response. If you do not have the Silent Post feature enabled, the MD5 Hash information below is not applicable.

Using the MD5 Hash Feature for ARB Transactions

The MD5 Hash feature enables you to authenticate that a transaction response is securely received from Authorize.Net. The payment gateway creates the MD5 Hash using the following pieces of account and transaction information as input:

- MD5 Hash value

- Transaction ID (*x_trans_id*)

- Amount (*x_amount*)

The MD5 Hash value is a random value configured by the merchant in the Merchant Interface. It should be stored securely separately from the merchant's Web server. For more information on how to configure this value, see the *Merchant Integration Guide* athttp://www.authorize.net/support/merchant/.

For example, if the MD5 Hash value configured by the merchant in the Merchant Interface is "wilson," and the transaction ID is "9876543210" with an amount of $1.00, then the field order used by the payment gateway to generate the MD5 Hash would be as follows:

```
wilson98765432101.00
```

Note:	The value passed back for *x_amount* is formatted with the correct number of decimal places used in the transaction. For transaction types that do not include a transaction amount, mainly Voids, the amount used by the payment gateway to calculate the MD5 Hash is "0.00."

To authenticate the MD5 Hash returned by the payment gateway in the transaction response, you will need to create a script that can receive and parse the transaction response, call the merchant's MD5 Hash value, and run the MD5 algorithm on the same fields listed above. If the result matches the MD5 Hash returned by the payment gateway, the transaction response is successfully authenticated.

The fields listed above are only applicable when using the MD5 Hash for ARB transactions. If you are using the MD5 Hash to authenticate non-ARB transactions, the fields used are different. For information on using the MD5Hash for non-ARB transactions, see the *Merchant Integration Guide* at http://www.authorize.net/support/merchant/.

Example Silent Post Response

```
x_response_code=1&x_response_subcode=1&x_response_reason_code=1&
x_response_reason_text=This+transaction+has+been+approved%2E&
x_auth_code=QbJHm4&x_avs_code=Y&x_trans_id=2147490176&
x_invoice_num=INV12345&x_description=My+test+description&
x_amount=0%2E44&x_method=CC&x_type=auth%5Fcapture&
x_cust_id=CustId&x_first_name=Firstname&
x_last_name=LastNamenardkkwhczdp&x_company=&x_address=&x_city=&
x_state=&x_zip=&x_country=&x_phone=&x_fax=&x_email=&
x_ship_to_first_name=&x_ship_to_last_name=&x_ship_to_company=&
x_ship_to_address=&x_ship_to_city=&x_ship_to_state=&x_ship_to_zip=&
x_ship_to_country=&x_tax=0%2E0000&
x_duty=0%2E0000&x_freight=0%2E0000&x_tax_exempt=FALSE&x_po_num=&
x_MD5_Hash=B9B3D19AEFD7BECC86C5FB3DB717D565&
x_cavv_response=2&x_test_request=false&x_subscription_id=101635&
x_subscription_paynum=1
```

Output Elements for ARBUpdateSubscriptionResponse and ARBCancelSubscriptionResponse

The output elements in for ARBUpdateSubscriptionResponse and ARBCancelSubscriptionResponse are the same as ARBCreateSubscriptionResponse with the following exception:

> The subscriptionID of the updated subscription is not included in the response.

Error Response

The following table describes the output elements for an error response to any of the requested API methods.

ELEMENT	VALUE	FORMAT	NOTES
refID	Merchant-assigned reference ID for the request	Up to 20 characters	This element is included in the response only if it was included in the request.
messages	Contains information about the results of the request		
resultCode	Contains additional information about the results of the request	Error	The request resulted in one or more errors.

message	Contains the result code and text		Messages provide more details about the error(s).
code	The code that represents the reason for the error		
text	A text description of the error		

Example Error Response

```
<?xml version="1.0" encoding="utf-8"?>
<ErrorResponse
xmlns="AnetApi/xml/v1/schema/AnetApiSchema.xsd">
  <messages>
    <resultCode>Error</resultCode>
    <message>
      <code>E00003</code>
      <text> An error occurred while parsing the XML
request.
      </text>
    </message>
  </messages>
</ErrorResponse>
```

Error Codes

The following table lists the common error codes and texts.

CODE	TEXT	DESCRIPTION
E00001	An error occurred during processing. Please try again.	An unexpected system error occurred while processing this request.
E00002	The content-type specified is not supported.	The only supported content-types are text/xml and application/xml.
E00003	An error occurred while parsing the XML request.	This is the result of an XML parser error.
E00004	The name of the requested API method is invalid.	The name of the root node of the XML request is the API method being called. It is not valid.
E00005	The merchantAuthentication.transactionKey is invalid or not present.	Merchant authentication requires a valid value for transaction key.
E00006	The merchantAuthentication.name is invalid or not present.	Merchant authentication requires a valid value for name.

E00007	User authentication failed due to invalid authentication values.	The name/and or transaction key is invalid.
E00008	User authentication failed. The payment gateway account or user is inactive.	The payment gateway or user account is not currently active.
E00009	The payment gateway account is in Test Mode. The request cannot be processed.	The requested API method cannot be executed while the payment gateway account is in Test Mode.
E00010	User authentication failed. You do not have the appropriate permissions.	The user does not have permission to call the API.
E00011	Access denied. You do not have the appropriate permissions.	The user does not have permission to call the API method.
E00012	A duplicate subscription already exists.	A duplicate of the subscription was already submitted. The duplicate check looks at several fields including payment information, billing information and, specifically for subscriptions, Start Date, Interval and Unit.
E00013	The field is invalid.	One of the field values is not valid.
E00014	A required field is not present.	One of the required fields was not present.
E00015	The field length is invalid.	One of the fields has an invalid length.
E00016	The field type is invalid.	The field type is not valid.
E00017	The startDate cannot occur in the past.	The subscription start date cannot occur before the subscription submission date. (Note: validation is performed against local server date, which is Mountain Time.)
E00018	The credit card expires before the subscription startDate.	The credit card is not valid as of the start date of the subscription.
E00019	The customer taxId or driversLicense information is required.	The customer tax ID or driver's license information (driver's license number, driver's license state, driver's license DOB) is required for the subscription.
E00020	The payment gateway account is not enabled for eCheck.Net subscriptions.	This payment gateway account is not set up to process eCheck.Net subscriptions.
E00021	The payment gateway account is not enabled for credit card subscriptions.	This payment gateway account is not set up to process credit card subscriptions.
E00022	The interval length cannot exceed 365 days or	The interval length must be 7 to 365 days or 1

	12 months.	to 12 months.
E00024	The trialOccurrences is required when trialAmount is specified.	The number of trial occurrences cannot be zero if a valid trial amount is submitted.
E00025	Automated Recurring Billing is not enabled.	The payment gateway account is not enabled for Automated Recurring Billing.
E00026	Both trialAmount and trialOccurrences are required.	If either a trial amount or number of trial occurrences is specified then values for both must be submitted.
E00027	The test transaction was unsuccessful.	An approval was not returned for the test transaction.
E00028	The trialOccurrences must be less than totalOccurrences.	The number of trial occurrences specified must be less than the number of total occurrences specified.
E00029	Payment information is required.	Payment information is required when creating a subscription.
E00030	A paymentSchedule is required.	A payment schedule is required when creating a subscription.
E00031	The amount is required.	The subscription amount is required when creating a subscription.
E00032	The startDate is required.	The subscription start date is required to create a subscription.
E00033	The subscription Start Date cannot be changed.	Once a subscription is created the Start Date cannot be changed.
E00034	The interval information cannot be changed.	Once a subscription is created the subscription interval cannot be changed.
E00035	The subscription cannot be found.	The subscription ID for this request is not valid for this merchant.
E00036	The payment type cannot be changed.	Changing the subscription payment type between credit card and eCheck.Net is not currently supported.
E00037	The subscription cannot be updated.	Subscriptions that are expired, canceled or terminated cannot be updated.
E00038	The subscription cannot be canceled.	Subscriptions that are expired or terminated cannot be canceled.

Last revised: 10/13/2009

E00045	The root node does not reference a valid XML namespace.	An error exists in the XML namespace. This error is similar to E00003.

General Errors for Individual Payments in a Subscription

Anytime an error occurs that prevents the payment gateway from processing a scheduled payment in a subscription, the payment will result in a general error. For example, if the credit card expiration date on file for a subscription is not updated before it expires, the next scheduled payment will not be processed and the transaction will result in a general error. These subscriptions will not be suspended or be automatically terminated unless the general error occurs on the first scheduled payment in the subscription.

Note: If the Silent Post feature of the Merchant Interface is enabled, transactions that receive a general error will NOT post a response to the specified Silent Post URL. See the "Transaction Response for Individual Payments in a Subscription" section of this guide for more information on using the Silent Post feature.

Some of the most common reasons for a payment to receive a general error are:

- The credit card number or expiration date on file has expired.

- The payment gateway account was in test mode at the time of the scheduled payment.

- eCheck.Net has been disabled for the payment gateway account or the specific eCheck.Net type has been disabled.

- A notice of change (NOC) has been received for the eCheck.Net subscription.

Payments with general errors can be identified on the completed transactions page of the Merchant Interface. They will display "N/A" in the Transaction ID field and "General Error" in the Transaction Status field.

Note: Transactions that result in general errors can also be found in the Failed.csv Excel file that comes when you are enabled to receive the Daily Transaction Summary email.