Merchant Web Services API

Automated Recurring Billing (ARB)
XML Guide

January 2015

Authorize.Net Trademarks

Advanced Fraud Detection Suite™

Authorize.Net®

Authorize.Net Your Gateway to IP Transactions™

Authorize.Net Verified Merchant Seal™

Automated Recurring Billing™

eCheck.Net®

FraudScreen.Net®

Contents

Revision History

This table lists the changes made in the last six releases of this document:

Release	Changes
January 2015	Increased the maximum characters allowed for the value of the **nameOnAccount** field from 22 to 40. See Table 6 for more information.
July 2014	Added "Input Elements for ARBGetSubscriptionListRequest," page 22. Added "Output Elements for ARBGetSubscriptionListResponse," page 29.
June 2014	This release contains only formatting and editorial updates.
February 2014	Updated the Developer Support section. See "Developer Support," page 10.
October 2013	Updated list of "Payment Processors."
April 2013	Updated list of "Payment Processors."

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).

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.

ARB subscriptions **do not** process transactions in real time. Successful creation of an ARB subscription transaction does not indicate that the subscription payments that process through your account will be successful. ARB subscription transactions process at approximately 2:00 a.m. PST on their scheduled payment dates. Therefore the first scheduled transaction will not be sent to the customer's bank for authorization until approximately 2:00 a.m. PST on the start date that you specified when you created the subscription in your account. If you create a subscription with a start date that equals the creation date, the first scheduled payment will not process until after 2:00 a.m. the following day. If you wish to validate your customer's payment information before creating their subscription in your account, please use one of the real-time transaction processing methods, such as the Advanced Integration Method (AIM).

The ARB API behaves 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 indicating 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 | Log on to the Merchant Interface to step through the manual ARB process. If you do not have a live production account to use for this purpose, you can request a developer test account from our Developer Center. Be sure to include in the comments section that you need the ARB feature enabled for your test account. ARB subscription transactions never process through our test environment, so if you use a test environment account, you never see an ARB subscription transaction process. If you wish to see an ARB subscription transaction process, you MUST use your live production 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 registered for the Authorize.Net ARB service.
- Test Mode must be disabled.
- The merchant must store account authentication data such as API login ID and transaction key securely .

Note | Merchants should avoid storing any type of sensitive cardholder information. However, if a merchant or third party must store sensitive customer business or payment information, they must comply with industry standard storage requirements. See the *Standards, Compliance, and Security* training video.

Payment Processors

The merchant's payment processor determines the card types and currencies that the merchant can support.

North American Payment Processors

Accepted Card Types

All North American payment processors supported by Authorize.Net accept the following card types:

- American Express
- Diners Club
- Discover
- JCB
- Mastercard
- Visa

Accepted Currencies

**Table 1   North American Payment Processors
                and Accepted Currencies**

Payment Processor	Accepted Currencies
Chase Paymentech Tampa	United States Dollar (USD)
	Canadian Dollar (CAD)
Elavon	United States Dollar (USD)
	Canadian Dollar (CAD)
First Data Merchant Services (FDMS) Omaha, Nashville, and EFSNet	United States Dollar (USD)
Global Payments	United States Dollar (USD)
	Canadian Dollar (CAD)
Heartland Payment Systems	United States Dollar (USD)
TSYS Acquiring Solutions	United States Dollar (USD)
WorldPay Atlanta	United States Dollar (USD)

European Payment Processors

Authorize.Net supports the following European payment processors.

Table 2   European Payment Processors, Accepted Card Types, and Accepted Currencies

Payment Processor	Accepted Card Types	Accepted Currencies
AIB Merchant Services	▪ Mastercard ▪ Visa	British Pounds (GBP) Euro (EUR) United States Dollar (USD)
Barclaycard	▪ JCB ▪ Mastercard ▪ Visa	British Pounds (GBP) Euro (EUR)
First Data Merchant Solutions (MSIP platform)	▪ Mastercard ▪ Visa	British Pounds (GBP)
HSBC Merchant Services	▪ Mastercard ▪ Visa	British Pounds (GBP) Euro (EUR) United States Dollar (USD)
Lloyds Bank Cardnet	▪ Mastercard ▪ Visa	British Pounds (GBP)
Streamline	▪ JCB ▪ Mastercard ▪ Visa	British Pounds (GBP) Euro (EUR) United States Dollar (USD)

Asia-Pacific Processors

Authorize.Net supports the following Asia-Pacific payment processors for Card-Not-Present (CNP) transactions.

Table 3   Asia-Pacific Payment Processor, Accepted Card Types, and Accepted Currencies

Payment Processor	Accepted Card Types	Accepted Currencies
FDI Australia	▪ Mastercard ▪ Visa	Australian Dollar (AUD) New Zealand Dollar (NZD) United States Dollar (USD)
Westpac	▪ Mastercard ▪ Visa	Australian Dollar (AUD)

Developer Support

Resources are available to help you successfully integrate a merchant web site or other application to the Authorize.Net Payment Gateway.

- The Developer Center provides sandbox accounts, sample code, FAQs, and troubleshooting tools.
- Developer training videos cover a variety of topics.
- The developer community provides answers to questions from other Authorize.Net developers.
- Ask us a question at our Developer Support page.
- Search our knowledge base for answers to commonly asked questions.

To submit suggestions for improving or correcting this guide, send email to documentation@authorize.net.

Software Development Kits

Authorize.Net offers Software Development Kits (SDKs) that present an alternate object-oriented model, in several popular languages. 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:

http://developer.authorize.net/downloads/.

Executing an API Call

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

You can develop the subscription request script in one of two ways:

■ by yourself, using the API field information in this section

■ using Authorize.Net sample code in C#, Java, PHP, Ruby, and VBNet, available for free from our Developer Center. Unfortunately, we cannot offer all programming languages requested. If you do not wish to use the ARB sample code, use your knowledge of your chosen language, along with this guide, to create your own.

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

Note for .NET programmers

When you use serialization with optional parameters, then the .NET language you are using automatically creates Boolean properties that indicate whether or not non-nullable parameters are specified. For example, if there is a parameter named validationMode that is an Enumeration type, a parameter called validationModeSpecified is automatically created. By default, these properties are set to false. If a request passes a value for an optional parameter, be sure to set these properties to true so that the value is not ignored.

ARB API URLs

Post API requests to the following locations:

Table 4   API Request URLs

Item	Location
Production	https://api.authorize.net/xml/v1/request.api

Table 4   API Request URLs (Continued)

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. Production accounts cannot be used to test against the developer test URL, and vice versa.

ARB Subscription Functions

The ARB API includes the following functions:

- ARBCreateSubscriptionRequest
- ARBUpdateSubscriptionRequest
- ARBCancelSusbscriptionRequest
- ARBGetSubscriptionListRequest

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

The following sections describe the input parameters required for executing the functions listed above. Indentations in the Parameter column indicate grouping hierarchy. **All parameters are case-sensitive and must be submitted in the order listed here. Parameters are required unless otherwise indicated. Optional parameters should not be submitted unless they contain valid values.**

> **Note**
>
> Parameters required for individual API calls are in addition to the authentication parameters required for all API calls.

Authentication

All calls to the ARB API require merchant authentication. The following table shows the required XML elements.

Table 5 **Authentication parameters**

Element	Description
merchantAuthentication	Contains the merchant's payment gateway account authentication information
■ name	**Value**: The merchant's valid API login ID
	Format: Up to 25 characters
	Notes: Submit the API login ID used to submit transactions.
■ transactionKey	**Value**: The merchant's valid transaction key
	Format: 16 characters
	Notes: Submit the transaction key obtained by the merchant from the Merchant Interface.

Example **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 generic 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 Developer Center.

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.

Note For subscriptions with a monthly interval, whose payments begin on the 31st of a month, payments for months with less than 31 days will occur on the last day of the month.

Table 6   Input elements for ARBCreateSubscriptionRequest

Element	Description
refId	**Value**: Merchant-assigned reference ID for the request.
	Required: No.
	Format: Up to 20 characters.
	Notes: If included in the request, this value is included in the response. This feature might be especially useful for multi-threaded applications.
subscription	Contains information about the subscription.
■ name	**Value**: Merchant-assigned name for the subscription.
	Required: No.
	Format: Up to 50 characters.
■ paymentSchedule	Contains information about the payment schedule.
• interval	Contains information about the interval of time between payments.
• • • length	**Value**: The measurement of time, in association with the Interval Unit, that is used to define the frequency of the billing occurrences.
	Format: Up to 3 digits.
	Notes: 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	**Value**: The unit of time, in association with the Interval Length, between each billing occurrence.
	Format: days, months
• startDate	**Value**: The date the subscription begins (also the date the initial billing occurs).
	Format: YYYY-MM-DD
	Notes: 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.
	If the start date is the 31st, and the interval is monthly, the billing date is the last day of each month (even when the month does not have 31 days).
• totalOccurrences	**Value**: Number of billing occurrences or payments for the subscription.
	Format: Up to 4 digits.
	Notes: 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.

Table 6   Input elements for ARBCreateSubscriptionRequest (Continued)

Element	Description
• trialOccurrences	**Value**: Number of billing occurrences or payments in the trial period.
	Required: No.
	Format: Up to 2 digits.
	Notes: If a trial period is specified, this number must be included in the Total Occurrences.
▪ amount	**Value**: The amount to be billed to the customer for each payment in the subscription.
	Format: Up to 15 digits.
	Notes: If a trial period is specified, this is the amount that will be charged after the trial payments are completed.
▪ trialAmount	**Value**: The amount to be charged for each payment during a trial period.
	Required: Conditional.
	Format: Up to 15 digits.
	Notes: 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	**Value**: Contains the customer's credit card information.
	Notes: Include this element only when the payment method is credit card.
• • • cardNumber	**Value**: The credit card number used for payment of the subscription.
	Format: 13 to 16 digits.
• • • expirationDate	**Value**: The expiration date of the credit card used for the subscription.
	Format: YYYY-MM
• • • cardCode	**Value**: The three- or four-digit card code on the back of most credit cards, on the front for American Express.
	Required: No.
	Format: 3 or 4 digits.
	Notes: include this element only when the merchant has set the card code value field to **required** in the account settings. The value itself is never validated.
• bankAccount	**Value**: Contains the customer's bank account information.
	Notes: Include this element only when the payment method is bank account.
• • • accountType	**Value**: The type of bank account used for payment of the subscription.
	Format: checking, businessChecking, savings.
• • • routingNumber	**Value**: The routing number of the customer's bank .
	Format: 9 digits.
• • • accountNumber	**Value**: The bank account number used for payment of the subscription.
	Format: 5 to 17 digits.

Table 6   Input elements for ARBCreateSubscriptionRequest (Continued)

Element	Description
• • • nameOnAccount	**Value**: The full name of the individual associated with the bank account number. **Format**: Up to 40 characters.
• • • echeckType	**Value**: The type of electronic check transaction used for the subscription. **Format**: For checking or savings accounts, ARC, BOC, PPD, TEL, or WEB. For business checking accounts, CCD.
• • • bankName	**Value**: The name of the bank associated with the bank account number. **Required**: No. **Format**: Up to 50 characters.
■ order	**Value**: Contains optional order information. **Required**: no.
• invoiceNumber	**Value**: Merchant-assigned invoice number for the subscription. **Required**: No. **Format**: Up to 20 characters. **Notes**: The invoice number will be associated with each payment in the subscription.
• description	**Value**: Description of the subscription. **Required**: No **Format**: Up to 255 characters **Notes**: The description will be associated with each payment in the subscription.
■ customer	**Value**: Contains information about the customer
• id	**Value**: Merchant-assigned identifier for the customer **Required**: No. **Format**: Up to 20 characters
• email	**Value**: The customer's email address. **Format**: Up to 255 characters. **Notes**: Required only when using a European Payment Processor.
• phoneNumber	**Value**: The customer's phone number. **Required**: No. **Format**: Up to 25 digits
• faxNumber	**Value**: The customer's fax number. **Required**: No. **Format**: Up to 25 digits
■ billTo	**Value**: Contains the customer's billing address information
• firstName	**Value**: The first name associated with the customer's billing address. **Format**: Up to 50 characters. **Notes**: Required only when using a European Payment Processor.

Table 6 **Input elements for ARBCreateSubscriptionRequest (Continued)**

Element	Description
● lastName	**Value**: The last name associated with the customer's billing address.
	Format: Up to 50 characters.
	Notes: Required only when using a European Payment Processor.
● company	**Value**: The company associated with the customer's billing address.
	Format: Up to 50 characters.
● address	**Value**: The customer's billing address.
	Format: Up to 60 characters.
	Notes: Required only when using a European Payment Processor.
● city	**Value**: The city of the customer's billing address.
	Format: Up to 40 characters.
	Notes: Required only when using a European Payment Processor.
● state	**Value**: The state of the customer's billing address
	Format: 2 characters.
	Notes: Must be a valid state code.
	Required only when using a European Payment Processor.
● zip	**Value**: The ZIP code of the customer's billing address.
	Format: Up to 20 characters.
	Notes: Required only when using a European Payment Processor.
● country	**Value**: The country of the customer's billing address.
	Format: Up to 60 characters.
	Notes: Must be a valid two-character country code or full country name (spelled in English).
	Required only when using a European Payment Processor.
■ shipTo	**Value**: Contains the customer's shipping address information.
	Required: No.
● firstName	**Value**: The first name associated with the customer's shipping address.
	Format: Up to 50 characters.
● lastName	**Value**: The last name associated with the customer's shipping address.
	Format: Up to 50 characters.
● company	**Value**: The company associated with the customer's shipping address.
	Format: Up to 50 characters.
● address	**Value**: The customer's shipping address.
	Format: Up to 60 characters.
● city	**Value**: The city of the customer's shipping address.
	Format: Up to 40 characters.

Table 6   Input elements for ARBCreateSubscriptionRequest (Continued)

Element	Description
• state	**Value**: The state of the customer's shipping address.
	Format: Up to 40 characters.
• zip	**Value**: The ZIP code of the customer's shipping address.
	Format: Up to 20 characters.
• country	**Value**: The country of the customer's shipping address.
	Format: Up to 60 characters.
	Notes: Must be a valid two-character country code or full country name (spelled in English).

Example   ARBCreateSubscriptionRequest

```xml
<?xml version="1.0" encoding="utf-8"?>
<ARBCreateSubscriptionRequest xmlns="AnetApi/xml/v1/schema/
AnetApiSchema.xsd">
  <merchantAuthentication>
    <name>mytestacct</name>
    <transactionKey>112223344</transactionKey>
  </merchantAuthentication>
  <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 generic 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 Developer Center.

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.

You must submit the subscriptionID of the subscription to be updated.

Table 7   Input elements for ARBUpdateSubscriptionRequest

Element	Description
subscriptionId	**Value:** The payment gateway-assigned identification number for the subscription.
	Format: Up to 13 digits.

- The subscription start date (subscription.paymentSchedule.startDate) may only be updated if 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.
- If the start date is the 31st, and the interval is monthly, the billing date is the last day of each month (even when the month does not have 31 days).

All other fields are optional.

Note	The sample code included in this document uses generic 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 Developer Center.

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>
```

Input Elements for ARBCancelSubscriptionRequest

The following table represents the input elements for executing an API call to the ARBCancelSubscriptionRequest function.

Table 8   Input elements for ARBCancelSubscriptionRequest

Element	Description
refID	**Value**: Merchant-assigned reference ID for the request.
	Required: No.
	Notes: If included in the request, this value will be included in the response. This feature might be especially useful for multi-threaded applications.
subscriptionId	**Value**: The payment gateway-assigned identification number for the subscription.
	Format: 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 generic 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 Developer Center.

Input Elements for ARBGetSubscriptionStatusRequest

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

Table 9   Input elements for ARBGetSubscriptionStatusRequest

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

Example   ARBGetSubscriptionStatusRequest

```
<?xml version="1.0" encoding="utf-8"?>
<ARBGetSubscriptionStatusRequest xmlns="AnetApi/xml/v1/schema/
AnetApiSchema.xsd">
  <merchantAuthentication>
    <name>mytestacct</name>
    <transactionKey>112223344</transactionKey>
  </merchantAuthentication>
  <refId>Sample</refId>
  <subscriptionId>100748</subscriptionId>
</ARBGetSubscriptionStatusRequest>
```

> **Note** The sample code included in this document uses generic 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 Developer Center.

Input Elements for ARBGetSubscriptionListRequest

This request generates a list of subscriptions. The table below describes the input elements.

Table 10   Input Elements for ARBGetSubscriptionListRequest

Element	Description
searchType	Values include: ■ cardExpiringThisMonth ■ subscriptionActive ■ subscriptionInactive ■ subscriptionExpiringThisMonth
sorting	Contains sorting information.
■ orderBy	Order of transactions in response: ■ id ■ name ■ status ■ createTimeStampUTC ■ lastName ■ firstName ■ accountNumber (ordered by last 4 digits only) ■ amount ■ pastOccurences
■ orderDescending	**Value:** true, false, 1 or 0. **Format:** Boolean
paging	Contains information about list pages.
■ limit	**Value:** 1-1000 **Notes:** The number of subscriptions per page.
■ offset	**Value:** 1-10000 **Notes:** The number of pages.

Example   ARBGetSubscriptionListRequest

```
<?xml version="1.0" encoding="utf-8"?>
<ARBGetSubscriptionListRequest xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="AnetApi/xml/v1/schema/AnetApiSchema.xsd">
   <merchantAuthentication>
      <name>api_login</name>
      <transactionKey>transaction_key</transactionKey>
   </merchantAuthentication>
```

```
        <searchType>SubscriptionInactive</searchType>
        <sorting>
            <orderBy>Id</orderBy>
            <orderDescending>false</orderDescending>
        </sorting>
        <paging>
            <limit>1000</limit>
            <offset>1</offset>
        </paging>
</ARBGetSubscriptionListRequest>
```

XML Responses

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

Table 11   Output elements for ARBCreateSubscriptionResponse

Element	Description
refID	**Value:** Merchant-assigned reference ID for the request
	Format: Up to 20 characters
	Notes: This element is included in the response only if it was included in the request.
messages	**Value:** Contains information about the results of the request
■ resultCode	**Value:** Contains additional information about the results of the request
	Format: Ok
	Notes: An "Ok" result code indicates that the request was processed and accepted without error.
■ message	**Value:** Contains the result code and text
	Notes: Any messages present are informational only.
● code	**Value:** I00001
● text	**Value:** Successful
subscriptionId	**Value**: The payment gateway assigned identification number for the subscription
	Format: 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>
```

After 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 after 2 a.m. PST 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:

Step 1   Log on to the Merchant Interface at https://secure.authorize.net

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

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

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

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

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

Note	Test environment accounts do not process ARB subscription transactions. If you are using a test environment account, you will not receive these email notifications in any form.

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.

Output Elements for ARBUpdateSubscriptionResponse

and ARBCancelSubscriptionResponse

The output elements in for ARBUpdateSubscriptionResponse and ARBCancelSubscriptionResponse are the same as "Output Elements for ARBCreateSubscriptionResponse" with the following exception: the subscriptionID of the updated subscription is not included in the response.

Output Elements for ARBGetSubscriptionStatusResponse

Table 12   Output elements for ARBGetSubscriptionStatusResponse

Element	Description
refID	**Value:** Merchant-assigned reference ID for the request. **Format:** Up to 20 characters. **Notes:** This element is included in the response only if it was included in the request.
messages	**Value:** Contains information about the results of the request.
■ resultCode	**Value:** Contains additional information about the results of the request. **Format:** Ok **Notes:** An "Ok" result code indicates that the request was processed and accepted without error.
■ message	**Value:** Contains the result code and text. **Notes:** Any messages present are informational only.
● code	**Value:** The response code that represents the status.
● text	**Value:** The text description of the status.
status	**Value**: Contains information about the subscription status. **Possible Values:** ■ active ■ expired ■ suspended ■ cancelled ■ terminated

Example   ARBGetSubscriptionStatusResponse

```
<ARBGetSubscriptionStatusResponse xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="AnetApi/xml/v1/schema/AnetApiSchema.xsd">
   <refId>Sample</refId>
   <messages>
      <resultCode>Ok</resultCode>
      <message>
         <code>I00001</code>
         <text>Successful</text>
      </message>
   </messages>
   <Status>active</Status>
</ARBGetSubscriptionStatusResponse>
```

Output Elements for ARBGetSubscriptionListResponse

Table 13   Output Elements for ARBGetSubscriptionListResponse

Element	Description
messages	**Value:** Contains message information.
totalNumInresultSet	**Value:** The total number of subscriptions that met the search criteria.
subscriptionDetails	**Value:** Contains one or more instances of subscriptionDetail.
■ subscriptionDetail	**Value:** Contains the results of one query.
● id	**Value:** Subscription ID.
● name	**Value:** The name specified when the subscription was created.
● status	**Value**: Contains information about the subscription status. **Possible Values:** ■ active ■ expired ■ suspended ■ cancelled ■ terminated
● createTimeStampUTC	**Value:** When the subscription was created.
● firstName	**Value:** Customer first name.
● lastName	**Value:** Customer last name.

Table 13   Output Elements for ARBGetSubscriptionListResponse (Continued)

Element	Description
• totalOccurences	**Value:** How many total payments will make up the completed subscription. This includes both past and future scheduled payments.
	Format: Numeric
• pastOccurences	**Value:** How many payments have been attempted, whether they were completed or not.
	Format: Numeric
• paymentMethod	**Value:** creditCard, eCheck, or payPal
• accountNumber	**Value:** Last 4 digits of card or bank account number.
	Format: Numeric
• Invoice	**Value:** The invoice specified when the subscription was created.
• amount	**Value:** The amount set to be charged by the subscription.
• currencyId	**Value:** Reserved for future use—subject to change.

Example   ARBGetSubscriptionListResponse

```xml
<?xml version="1.0" encoding="utf-8"?>
<ARBGetSubscriptionListResponse xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="AnetApi/xml/v1/schema/AnetApiSchema.xsd">
  <messages>
    <resultCode>Ok</resultCode>
    <message>
      <code>I00001</code>
      <text>Successful.</text>
    </message>
  </messages>
  <totalNumInResultSet>1</totalNumInResultSet>
  <subscriptionDetails>
    <subscriptionDetail>
      <id>155551<id>
      <name>ARB Subscription Test</name>
      <status>active</status>
      <createTimeStampUTC>2014-07-11T02:52:42.96</createTimeStampUTC>
      <firstName>John</firstName>
      <lastName>doe</lastName>
      <totalOccurrences>8</totalOccurrences>
      <pastOccurrences>5</pastOccurrences>
      <paymentMethod>creditCard</paymentMethod>
      <accountNumber>XXXX1234</accountNumber>
      <invoice />
      <amount>19.99</amount>
      <currencyId>0</currencyId>
    </subscriptionDetail>
  </subscriptionDetails>
```

```
</ARBGetSubscriptionListResponse>
```

Error Response

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

Table 14   Output elements for an error response

Element	Description
refID	**Value:** Merchant-assigned reference ID for the request
	Format: Up to 20 characters
	Notes: This element is included in the response only if it was included in the request.
messages	**Value:** Contains information about the results of the request
■ **resultCode**	**Value:** Contains additional information about the results of the request
	Format: Error
	Notes: The request resulted in one or more errors.
■ **message**	**Value:** Contains the result code and text
	Notes: Messages provide more details about the error(s).
● **code**	**Value:** The code that represents the reason for the error
● **text**	**Value:** 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.

Table 15   **Error codes**

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.

Table 15   Error codes (Continued)

Code	Text	Description
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 12 months.	The interval length must be 7 to 365 days or 1 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.

Table 15 Error codes (Continued)

Code	Text	Description
E00038	The subscription cannot be canceled.	Subscriptions that are expired or terminated cannot be canceled.
E00045	The root node does not reference a valid XML namespace.	An error exists in the XML namespace. This error is similar to E00003.

Duplicate Subscription Verification

A duplicate check occurs against every ARB subscription created in an account in order to prevent duplicate subscriptions from inadvertently being created. The following is a list of the fields that are verified. If ALL of the verified fields are the same, an E00012 will occur and the subscription is not successfully created in the account. The duplicate check verifies for an indefinite amount of time.

- subscription.Article.MerchantID

- subscription.Article.CustomerInfo.Payment.CreditCard.CardNumber

- subscription.Article.CustomerInfo.Payment.eCheck.RoutingNumber

- subscription.Article.CustomerInfo.Payment.eCheck.AccountNumber

- subscription.Article.CustomerInfo.CustomerID

- subscription.Article.CustomerInfo.BillingInfo.BillToAddress.FirstName

- subscription.Article.CustomerInfo.BillingInfo.BillToAddress.LastName

- subscription.Article.CustomerInfo.BillingInfo.BillToAddress.Company

- subscription.Article.CustomerInfo.BillingInfo.BillToAddress.StreetAddress

- subscription.Article.CustomerInfo.BillingInfo.BillToAddress.City

- subscription.Article.CustomerInfo.BillingInfo.BillToAddress.StateProv

- subscription.Article.CustomerInfo.BillingInfo.BillToAddress.Zip

- subscription.OrderInfo.Amount

- subscription.OrderInfo.Invoice

- subscription.Recurrence.StartDate

- subscription.Recurrence.Interval

- subscription.Recurrence.Unit

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.

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.

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.