



Subscription Upgrade System

User's Guide

Contents

- Subscription Upgrade System 1
- Introduction 3
- Overview 3
- Setup Requirements 3
 - Upgrade Form 3
- Implementation..... 4
- Transmission Security Method..... 4
 - Technology 4
 - Option 1: Encryption (Preferred) 4
 - Option 2: Backend HTTPS Posting..... 5
 - Step 1 - Backend HTTPS Post 6
 - Step 2 - Sending consumer to Upgrade Form 7
- Postback 8
- Support Contact Information 8

Introduction

This document is provided as a technical resource to CCBill merchants. It is intended for programmers and technicians with advanced programming skills.

Overview

The Subscription Upgrade System enables CCBill Merchants to upgrade a consumer subscription to a new subscription. This can be done in the same sub-account or a different sub-account. Upgrades must be pre-configured for the account. The system supports both Credit Card and ACH (Automated Clearing House) upgrades.

The Subscription Upgrade System supports the use of Regional Pricing. By default, the Regional Pricing settings of the account to which the subscription is being upgraded will be used. This setting will be ignored if Regional Pricing is not enabled on the account. For more information on Regional Pricing, consult the Regional Pricing User's Guide.

The following URL will be used for all upgrade operations using this system:

<https://bill.ccbill.com/jpost/upgradeSubscription.cgi>

Values will be passed in to this URL as required.

Setup Requirements

Upgrades are set up entirely by CCBill Staff. Setup requirements for this functionality are defined below.

The following requirements are needed to utilize the Subscription Upgrade System when upgrading from one subscription type to another:

- Approval from CCBill management
- If upgrading to a different sub-account, you must have at least two configured subaccounts; otherwise only one is needed
- Ability to create dynamic URLs based on the subscriber
- Subscriber information such as subscription ID or username
- One of the following:
 - Base64-encoded TripleDES encryption string (*preferred*), **OR**
 - Backend HTTPS posting capability

Upgrade Form

Upgrades require an upgrade form, which is used by the consumer to request the upgrade. CCBill provides a default upgrade form which can be customized for Credit Card or ACH Upgrades. The upgrade form can be further modified to meet the merchant's specific needs; however, it must contain all the information on the default upgrade form. It must also be easy to understand and not misleading to the consumer, as well as approved by CCBill management.

The ACH Upgrade form must contain a field where the Bank Account number can be entered and the Credit Card Upgrade form must contain a field where the CVV2 number can be entered.

For more information regarding the customization of forms, please consult the CCBill help file.

Implementation

This section describes the implementation methods for Subscription Upgrades.

To set up an upgrade, CCBill will need the following information for all sub-accounts involved in the upgrade:

- Merchant Account (e.g. 900000)
- Merchant Sub-account (e.g. 0006)
- Pricing (one of the following)
 - Subscription Type ID (found in the Pricing Admin area of the CCBill Admin Portal)
 - Initial Price, Initial Period, Recurring Price, Recurring Period, Rebills (e.g., “5.00, 5, 29.95, 30, 99”). No currency signs may be used.
- Type of Subaccount (Initial or Upgrade)

Provide this information to Merchant Support to set up the upgrade. Once complete, Subscription Upgrade functionality can be implemented on the merchant’s site.

Transmission Security Method

Two transmission security options are available for pre-configured upgrades. Merchants must choose between two different authorization methods when sending consumers to the upgrade form.

Either of the following options can be used:

1. Encryption (Preferred)

- The merchant will need to obtain a TripleDES encryption key from Merchant Support.

2. Backend HTTPS Post

- A list of the CCBill Merchant’s IP addresses and/or IP ranges from which CCBill will receive Backend HTTPS Posts. IP addresses must be as specific as possible; large ranges will not be accepted.

Technology

Parameters that are passed to the above mentioned CGI are explained below. Two transmission methods are available:

Option 1: Encryption (Preferred)

The encryption method is a one-step process utilizing TripleDES encryption. A TripleDES key must be set up and obtained from CCBill Merchant Support. Once data is encrypted, the encryption string must be Base64 Encoded so that it can be used in a URL. The encryption string will be used as the value for the “enc” parameter. Perform the following steps to obtain the enc value:

1. Log in to the CCBill Admin Portal and go to **Account Info → Account Admin → Advanced**.
2. In the Upgrade Security Setup Information section, ensure that **Encryption** is selected as the **Security Type**.
3. Select **TripleDES** as the **Encryption Type**.
4. Enter your **Encryption Key** in the field provided. This key must be obtained from Merchant Support.
5. Click the **Update** button to save the key.
6. Using TripleDES encryption and your preferred Base64 encoding tool, encode the username or Subscription ID of the user using the encryption key. The encoded value will be used as the value for the enc parameter.

The following parameters are **required** and must be passed in to the

<https://bill.ccbill.com/jpost/upgradeSubscription.cgi> script:

Name	Description	Example Value
clientAccnum	CCBill Merchant Account (of original subscription)	900100
clientSubacc	CCBill Merchant Subaccount (of original subscription)	0006
enc	TripleDES encrypted string, Base64 Encoded. Will be inclusive of either the subscriptionId or username.	subscriptionId = 1234567890 OR username = johnDoe (before encryption)

The following variables are required only when more than one upgrade is configured:

Name	Description	Example Value
upgradeClientAccnum	CCBill merchant account to be upgraded.	900100
upgradeClientSubacc	CCBill Merchant subaccount to be upgraded.	0006
upgradeTypeid	Subscription type ID to be used for the upgrade. This value identifies the upgrade and is obtained when setting up the upgrade.	0015

These three variables can be omitted if only one upgrade is configured. In this case, that price point will be the one used for the upgrade.

The following example uses the sample values displayed above and a TripleDES key of "8745921457." Optional fields are shown in blue:

```
<form method="POST"
action="https://bill.ccbill.com/jpost/upgradeSubscription.cgi">
<input type="hidden" name="clientAccnum" value="900100">
<input type="hidden" name="clientSubacc" value="0006">
<input type="hidden" name="enc" value="sysl4wpcOfdF80ghzxJ0H/8YVuBGxA+4wqZqXHS48I8=">
<input type="submit" name="submit" value="Upgrade Subscription">
<input type="hidden" name="upgradeClientAccnum" value="900100">
<input type="hidden" name="upgradeClientSubacc" value="0006">
<input type="hidden" name="upgradeTypeid" value="0015">
</form>
```

This code will show a form button. When clicked, the upgrade will be performed and the consumer will be sent to the Approval or Denial Redirect page depending on the outcome of the transaction. For assistance in defining the Approval and Denial Redirect URLs, please consult the CCBill help file.

Option 2: Backend HTTPS Posting

The Backend HTTPS Posting method is a two-step process. The first process is the backend post from the merchant's server to the CCBill upgrade URL. This will return a value indicating whether or not the upgrade can be performed (see **Return Values** table below). The second step is to send the consumer to the

upgrade form. The consumer must be sent to the upgrade form within 30 minutes of the Backend HTTPS Post. If the consumer is not sent within 30 minutes, the Backend HTTPS Post must be sent again.

Note that when passing in the Subscription ID or username, the value is not an encrypted hash as it would be using the Encryption method defined earlier in this document.

Step 1 - Backend HTTPS Post

The required parameters sent are:

Name	Description	Example Value
clientAccnum	CCBill Merchant Account (of original subscription)	900100
clientSubacc	CCBill Merchant Subaccount (of original subscription)	0006
action	The value will always be "allowUpgrade". This signifies that the incoming request will be preparing the system for a consumer to be allowed to upgrade.	allowUpgrade
subscriptionId OR username	The subscriptionId or username that will be allowed to upgrade.	1234567890

The following variables are required only when more than one upgrade is configured:

Name	Description	Example Value
upgradeClientAccnum	CCBill merchant account to be upgraded.	900000
upgradeClientSubacc	CCBill Merchant subaccount to be upgraded.	0006
upgradeTypeid	Subscription type to be used for the upgrade.	0015

These three variables can be omitted if only one upgrade is configured; in this case, that price point will be the one used for the upgrade.

Example using the data above (new lines are used for clarity; optional fields are in blue):

```
https://bill.ccbill.com/jpost/upgradeSubscription.cgi?
clientAccnum=900100&
clientSubacc=0006&
action=allowUpgrade&
subscriptionId=1234567890&
upgradeClientAccnum=900100&
upgradeClientSubacc=0006&
upgradeTypeid=0015
```

IMPORTANT: This post must come from an allowed IP address on the Merchant's server. The consumer should not be directed to this URL.

Return Values:

The possible return values are included in the table below:

Value	Description
1	Success
0	Authorization Denied (Either account is not setup to upgrade, or IP address is not allowed)
-1	Username (username) or Subscription ID (subscriptionId) not provided
-2	No active subscriptions found
-3	Invalid Subscription ID
-4	Internal Error, Contact CCBill

If a value of “1” is returned, the upgrade is approved and step 2 can begin.

Step 2 - Sending consumer to Upgrade Form

The required parameters sent are:

Name	Description	Example Value
clientAccnum	CCBill Merchant Account (of original subscription)	900100
clientSubacc	CCBill Merchant Subaccount (of original subscription)	0006
action	The value will always be “upgradeSubscription”. This signifies that the incoming request will show the consumer the upgrade form.	upgradeSubscription
subscriptionId OR username	The subscriptionId or username that will be allowed to upgrade	1234567890

The following variables are required only when more than one upgrade is configured:

Name	Description	Example Value
upgradeClientAccnum	CCBill merchant account to be upgraded.	900100
upgradeClientSubacc	CCBill Merchant subaccount to be upgraded.	0006
upgradeTypeeld	Subscription type to be used for the upgrade.	0015

These three variables can be omitted if only one upgrade is configured; in this case, that price point will be the one used for the upgrade.

Example using the data above (optional items are in blue):

```

<form method="POST"
action="https://bill.ccbill.com/jpost/upgradeSubscription.cgi">
<input type="hidden" name="clientAccnum" value="900100">
<input type="hidden" name="clientSubacc" value="0006">
<input type="hidden" name="action" value="upgradeSubscription">
<input type="hidden" name="subscriptionId" value="1234567890">
<input type="submit" name="submit" value="UpgradeSubscription">
<input type="hidden" name="upgradeClientAccnum" value="900100">
<input type="hidden" name="upgradeClientSubacc" value="0006">
<input type="hidden" name="upgradeTypeId" value="0015">
</form>

```

Postback

Variable values are posted back using the CCBill Background Post system. The following values will be sent to the Approval URL or Denial URL when the upgrade has been performed or denied, respectively:

- **originalClientAccnum.** The six-digit merchant account number from where the upgrade originated.
- **originalClientSubacc.** The four-digit merchant subaccount number from where the upgrade originated.
- **originalSubscriptionId.** The subscription ID number of the original subscription.

For more information on the use of Background Post, please consult the CCBill [Background Post User's Guide](#).

Support Contact Information

For any additional questions or assistance, feel free to contact CCBill Merchant Support 365 days a year or visit our [Merchant Support Center](#).

- **Phone (TF):** 800.510.2859
- **Hours:** 24/7
- **Fax:** 480.449.8801
- **Email:** merchantsupport@ccbill.com
- **Live Chat:** [Click Here](#)