

cXML Punchout Integration Plugin - User Guide

Table of Contents

- 1. Introduction
- 2. Initial Setup and Client Configuration
- 3. Invoice Transmission Configuration
- 4. Shipment Transmission Configuration
- 5. Testing and Verification
- 6. Transmission Log
- 7. Punchout Quotes Management
- 8. Logging and Troubleshooting
- 9. Support

1. Introduction

The cXML Punchout Integration Plugin enables seamless integration between your Magento store and client eProcurement systems. This extension supports standard cXML protocol for punchout sessions, order creation, invoice transmission, and shipment notifications.

2. Initial Setup and Client Configuration

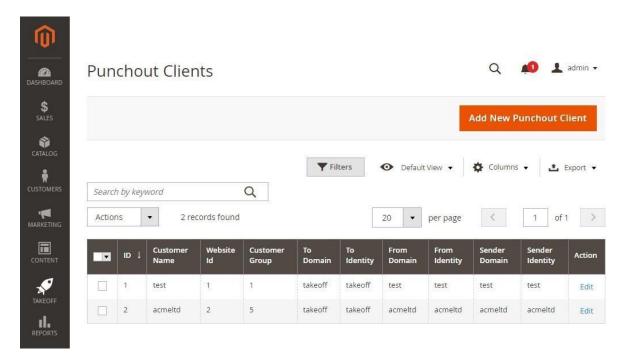
2.1 Accessing Punchout Client Management

- 1. Navigate to your Magento admin panel
- 2. Go to Takeoff → Manage Punchout Clients



3. Click **Add New Client** to begin configuration

Note: A "client" refers to the organization or entity that will connect to your store through their eProcurement system.



2.2 Client Configuration

Complete the following fields in the Client Configuration section:

Customer Name

Enter the client's company name (e.g., "ACME Ltd").

Website ID

Select the Magento website where this client integration will be active.

Customer Group

Choose the customer group to which punchout users will be automatically assigned upon login.

2.3 Punchout Credentials

Configure the authentication credentials for the cXML connection:

From Domain



The client's domain identifier (e.g., "acme").

From Identity

The client's identity credential (e.g., "acme").

To Domain

Your store's domain identifier (e.g., "magentostore").

To Identity

Your store's identity credential (e.g., "magentostore").

Sender Domain

The domain from which punchout requests originate (typically matches From Domain).

Sender Identity

The identity from which punchout requests originate (typically matches From Identity).

Shared Secret

A secure password for authenticating the punchout connection (e.g., "Acme123\$£@DS"). This must be configured identically in both systems.

Client Unique URL Parameter

A unique identifier for this client's URLs (e.g., "acmeltd"). This parameter will be used in all punchout endpoints.

2.4 Punchout URLs

After configuration, provide the following URLs to your client:

Punchout Login URL:

YOUR_STORE_URL/takeoff/punchout/login/client/CLIENT_UNIQUE_URL_PARAM

Order Submission URL:

YOUR_STORE_URL/takeoff/punchout/po/client/CLIENT_UNIQUE_URL_PARAM



2.5 Order Import Configuration

Configure how orders received from the client's eProcurement system will be processed:

Payment Method Code

The payment method to assign to punchout orders.

Delivery Method Code

The shipping method to assign to punchout orders.

Order Status

The initial status for newly created punchout orders.

These settings can be customized to meet specific business requirements.

2.6 cXML Item Configuration

Configure product data formatting for cXML documents:

Unit of Measure Default

Default unit of measure value (e.g., "EA" for "Each") used when no product-specific value is available

UNSPSC Classification Default

Default UNSPSC classification code (e.g., "463261321") used when no product-specific classification exists.

Unit of Measure Product Attribute

The Magento product attribute code that stores unit of measure data (e.g., "product_uom"). Leave empty to always use the default value.

UNSPSC Classification Product Attribute

The Magento product attribute code that stores UNSPSC classification data (e.g., "product_classification"). Leave empty to always use the default value.



Logic Flow: When generating cXML documents, the system first checks if the product has the specified attribute. If the attribute exists and contains a value, it is used. Otherwise, the system falls back to the configured default value.

2.7 Extrinsic Data Configuration

Extend cXML documents with custom data fields:

Extrinsic Data for ItemIn

Add custom fields to line item data. Values can be sourced from:

- Default static values
- Product attributes
- Quote item table columns

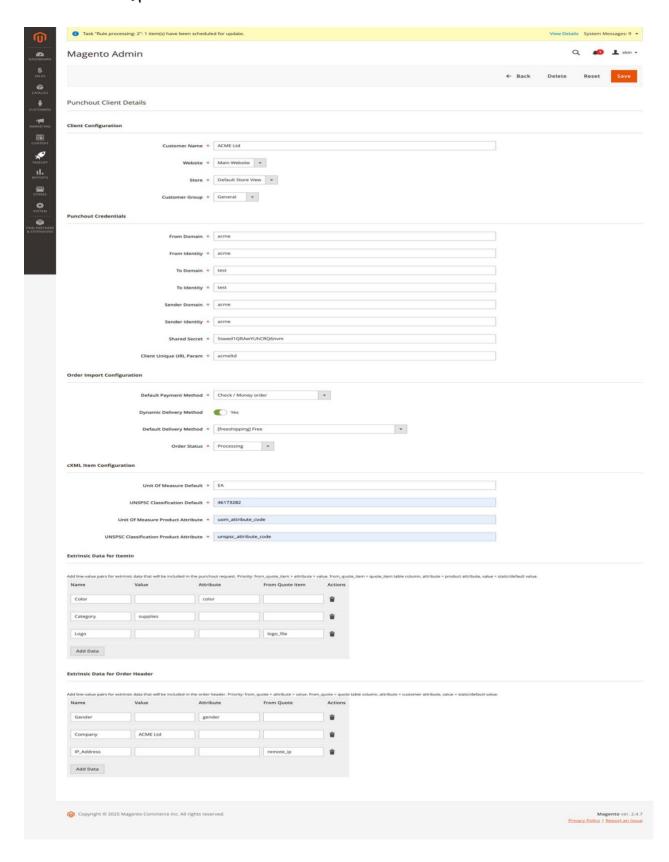
Extrinsic Data for Order Header

Add custom fields to order header data. Values can be sourced from:

- Default static values
- Product attributes
- Quote table columns

These configurations allow you to pass additional business-specific data to client eProcurement systems.







3. Invoice Transmission Configuration

This section configures automatic invoice transmission from your system to the client's eProcurement platform using the cXML invoice format.

Configuration Fields

Enable Invoice Sending

Toggle this setting to enable or disable automatic invoice transmission. When enabled, invoices are automatically sent to the configured endpoint URL upon generation.

Invoice Endpoint URL

The client's system URL that receives cXML invoice documents. This endpoint must be provided by the client's eProcurement administrator and must accept POST requests containing cXML-formatted invoice data.

Invoice Sender Identity

Your system's unique identifier as the invoice sender. This value must match the credentials configured in your sender profile.

Invoice To Identity

The client system's unique identifier as the invoice recipient. This value must be provided by the client.

Invoice From Domain

The domain qualifier for the From credential, typically using standards such as DUNS (Data Universal Numbering System) or NetworkID.

Invoice From Identity

Your company's unique identifier within the client's system, used for matching invoices to purchase orders.

Invoice To Domain



The domain qualifier for the To credential, using standards such as DUNS or NetworkID.

Invoice Shared Secret

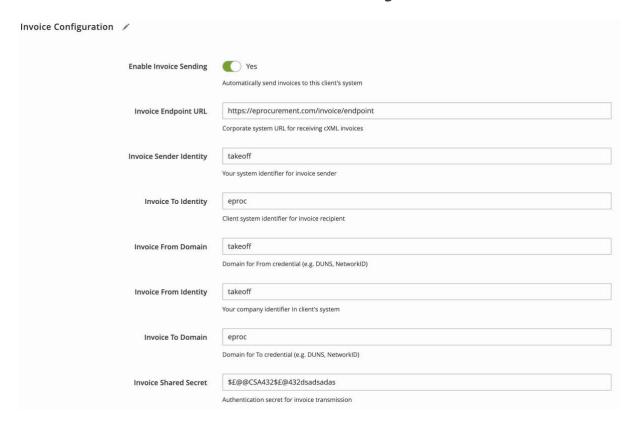
A secure authentication token used to validate invoice transmissions. This must match the value configured in the client's system. Maintain confidentiality and rotate periodically according to security policies.

Transmission Methods

Invoices are transmitted automatically via scheduled cron jobs. Additionally, you can trigger transmissions manually using Magento CLI commands:

Check order and invoice status php bin/magento takeoff:punchout:check-orders # Manually send invoices php bin/magento takeoff:punchout:send-invoices

Monitor transmission status in the **Transmission Log** menu (see Section 6).





4. Shipment Transmission Configuration

This section configures automatic shipment notice transmission from your system to the client's eProcurement platform using the cXML shipment notice format (Advanced Shipping Notice/ASN).

Configuration Fields

Enable Shipment Sending

Toggle this setting to enable or disable automatic shipment notice transmission. When enabled, shipment notifications are automatically sent upon shipment creation or dispatch.

Shipment Endpoint URL

The client's system URL that receives cXML shipment notice documents. This endpoint must be provided by the client's eProcurement administrator and must accept POST requests containing cXML-formatted shipment data.

Shipment Sender Identity

Your system's unique identifier as the shipment sender. This value must match the credentials configured in your sender profile.

Shipment To Identity

The client system's unique identifier as the shipment recipient. This value must be provided by the client.

Shipment From Domain

The domain qualifier for the From credential, typically using standards such as DUNS or NetworkID.

Shipment From Identity

Your company's unique identifier within the client's system, used for matching shipment notices to purchase orders.

Shipment To Domain



The domain qualifier for the To credential, using standards such as DUNS or NetworkID.

Shipment Shared Secret

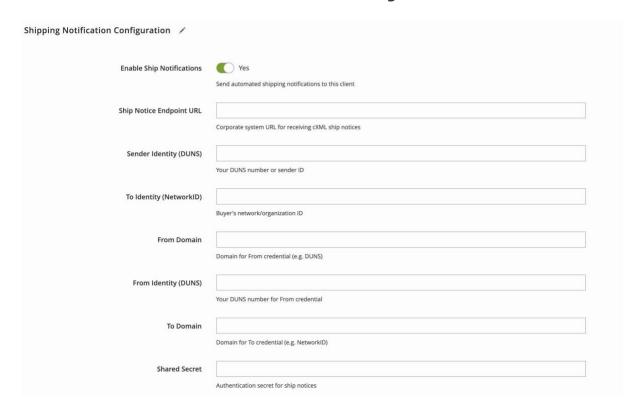
A secure authentication token used to validate shipment notice transmissions. This must match the value configured in the client's system. Maintain confidentiality and rotate periodically according to security policies.

Transmission Methods

Shipment notifications are transmitted automatically via scheduled cron jobs. Additionally, you can trigger transmissions manually using Magento CLI commands:

Check order and shipment status php bin/magento takeoff:punchout:check-orders # Manually send shipment notices php bin/magento takeoff:punchout:send-ship-notices

Monitor transmission status in the **Transmission Log** menu (see Section 6).





5. Testing and Verification

5.1 Testing Punchout Login

Step 1: Access the cXML Testing Tool

Navigate to the cXML Punchout Tester: https://punchoutcommerce.com/tools/cxml-punchout-tester#advanced

Step 2: Configure Test Parameters

Enter the following information in the testing tool:

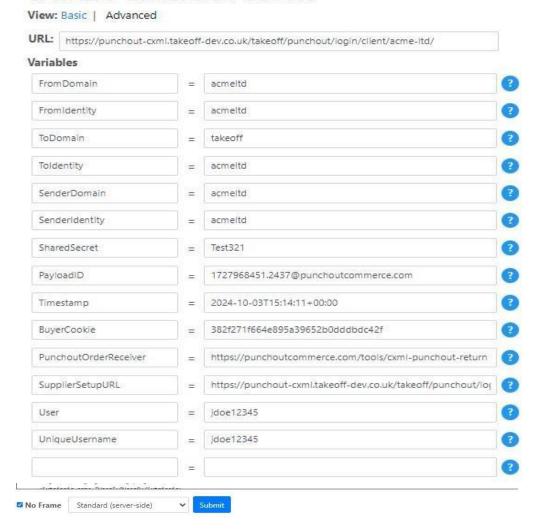
Punchout URL:

YOUR_STORE_URL/takeoff/punchout/login/client/CLIENT_UNIQUE_URL PARAM

- From Domain: Enter the client's domain (as configured in Section 2.3)
- From Identity: Enter the client's identity credential
- **To Domain:** Enter your store's domain
- **To Identity:** Enter your store's identity credential
- Sender Domain: Enter the sender domain
- **Sender Identity:** Enter the sender identity credential
- Shared Secret: Enter the shared secret password
- **Display Option:** Select "No Frame"



cXML PunchOut Tester



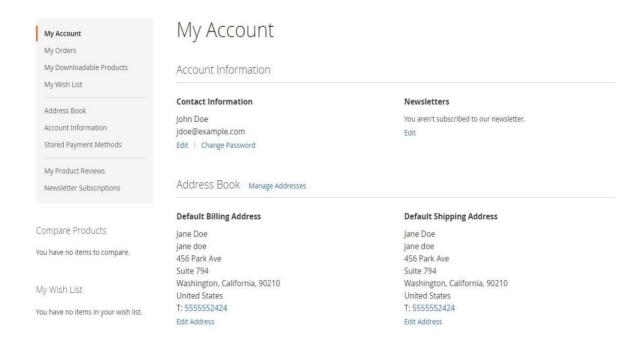
Step 3: Submit Test Configuration

Click the **Submit** button to initiate the punchout session.

Step 4: Verify Successful Login

Upon successful authentication, you will be logged into your Magento store. The user session will be established with the configured customer account.





5.2 Testing Cart Transfer

Step 1: Build Shopping Cart

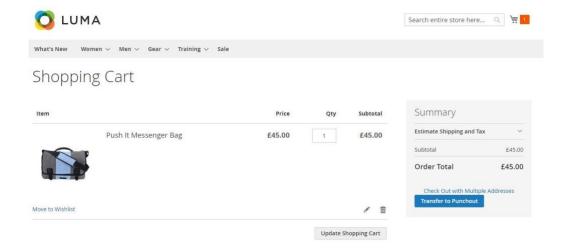
Add products to the shopping cart during the punchout session.

Step 2: Initiate Transfer

Navigate to the shopping cart. Note that the standard "Proceed to Checkout" button is replaced with "Transfer to Punchout."

Step 3: Complete Transfer

Click **Transfer to Punchout** to send the cart contents back to the client's eProcurement system.





Step 4: Verify Transfer

Confirm that the items appear correctly in the client's eProcurement system with accurate pricing, quantities, and product details.



5.3 Testing Order Submission

Step 1: Access the Order Testing Tool

Navigate to the cXML Order Tester: https://punchoutcommerce.com/tools/cxml-order-tester

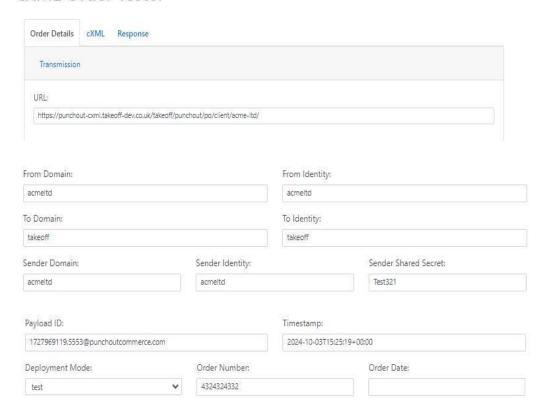
Step 2: Configure Order Details

Complete the form with the following information:

- Order URL:
 - YOUR_STORE_URL/takeoff/punchout/po/client/CLIENT_UNIQUE_URL_PA RAM
- From Domain, Identity, To Domain, To Identity, Sender Domain, Sender Identity, Shared Secret: Enter values matching your configuration
- Bill-To Email: Enter the email address of the user account logged in via punchout
- Item Part Number: Enter a valid Magento SKU
- **Item Description:** Enter the product description



cXML Order Tester



Step 3: Submit Test Order

Click **Submit** to send the test order to your Magento store.





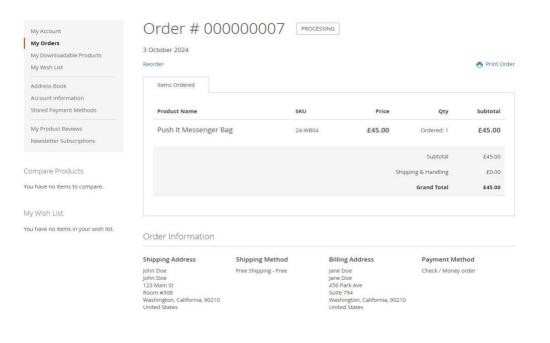
Step 4: Verify Response

A successful submission returns an XML response:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE cXML SYSTEM "http://xml.cxml.org/schemas/cXML/1.2.040/cXML.dtd"> <cXML
payloadID="[PAYLOAD_ID]" timestamp="[TIMESTAMP]">
<Response> <Status code="200" text="OK">Success</Status> </Response>
</cXML>
```

Step 5: Verify Order Creation

Check your Magento order management system to confirm the order has been created with the correct details, payment method, shipping method, and status as configured in Section 2.5.



6. Transmission Log

The Transmission Log provides a comprehensive audit trail of all cXML document transmissions (invoices and shipment notices) sent from your system to client eProcurement platforms.

Overview

Access the Transmission Log via **Takeoff** → **Transmission Log**. The log displays all outbound cXML documents with detailed information about each transmission attempt, including status, timestamps, and response codes.



Features

Search and Filtering

Use the keyword search to locate specific transmissions by document number, order number, or client name. Apply filters to narrow results by date range, transmission type, status, or response code.

Column Descriptions

- ID: Unique internal identifier for each transmission log entry
- **Type:** Document type (invoice or shipment)
- **Document #:** Internal document number (invoice number or shipment number)
- Order #: Associated purchase order number
- **Client Name:** Name of the receiving client
- Status: Current transmission status (sent, pending, failed, retry)
- **Attempts:** Number of transmission attempts made
- **Created:** Timestamp when the transmission record was created
- **Sent:** Timestamp when the document was successfully transmitted
- **Response Code:** HTTP response code from the client's endpoint (200 = success, 401 = authentication failure, 500 = server error, timeout = no response)
- **Error Message:** Detailed error information for failed transmissions
- **Client ID:** Internal identifier for the client configuration

Usage Guidelines

Monitoring Successful Transmissions

Filter for sent status and 200 response codes to verify successful document delivery.

Troubleshooting Failed Transmissions

Filter for non-200 response codes or failed status. Review error messages for specific failure details. Common issues include authentication failures, incorrect endpoint URLs, or network connectivity problems.

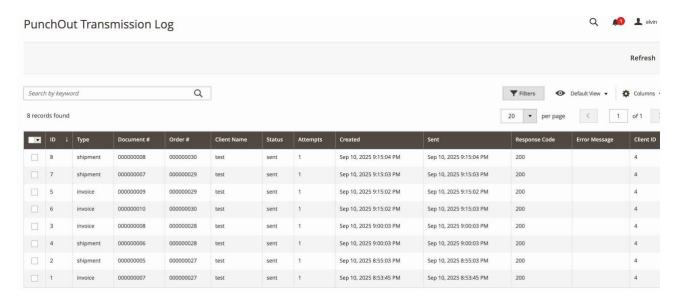


Compliance and Audit Trail

Use the log to maintain records of document transmission times, supporting compliance requirements and resolving delivery disputes with clients.

Performance Monitoring

Identify patterns in failed transmissions that may indicate configuration or connectivity issues requiring attention.



7. Punchout Quotes Management

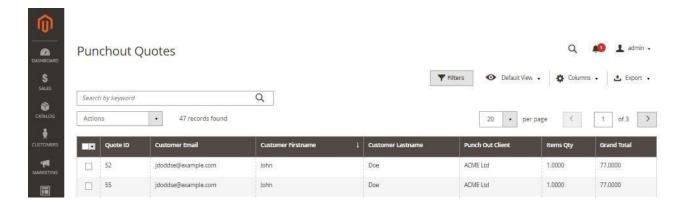
The Punchout Quotes section provides administrators with visibility into items that users have transferred from your store to their eProcurement systems. Access this feature via **Takeoff** → **Punchout Quotes**.

This view displays:

- · Quote ID and creation date
- Customer information
- Items selected during punchout sessions
- Transfer status
- Associated client configuration

Use this information to analyze user behavior, track popular products, and understand purchase patterns across different client organizations.





8. Logging and Troubleshooting

8.1 System Logging

When logging is enabled, all cXML transaction logs are automatically recorded in:

var/log/takeoff/

Log files capture:

- Punchout login requests and responses
- Cart transfer operations
- Order submissions
- Invoice transmissions
- Shipment notice transmissions

8.2 Security and Privacy

For security purposes, sensitive information such as shared secrets and authentication tokens are automatically redacted from log files. This ensures confidentiality while maintaining useful audit trails for troubleshooting.

8.3 CLI Commands

Use the following Magento CLI commands for manual operations and troubleshooting:

```
# Check status of punchout orders and pending transmissions
php bin/magento takeoff:punchout:check-orders # Manually trigger invoice sending
php bin/magento takeoff:punchout:send-invoices # Manually trigger shipment notice sending
php bin/magento takeoff:punchout:send-ship-notices
```



9. Support

For technical assistance, implementation support, or general inquiries, please contact:

Email: studio@takeoffdigital.co.uk **Website:** https://takeoffdigital.co.uk

Document Version: 1.2.0 **Last Updated:** October 2025

Compatible with: Magento 2.4.x, Adobe Commerce (cloud):2.4, Adobe Commerce(on-

prem):2.4

© 2025 Takeoff Digital. All rights reserved.