

PaperCut | PaperCut Payment Gateway Module - Heartland Quick Start Guide

This guide is designed to supplement the *Payment Gateway Module* documentation and provides a guide to installing, setting up and testing the *Payment Gateway Module* for use with Heartland's OneCard system. The main Payment Gateway Module documentation may be downloaded from:

<http://www.papercut.com/files/pcng/ext/payment-gateway/PaymentGatewayModule.pdf>

Contents

1	Introduction.....	2
2	Architecture	3
3	End-User Overview	4
3.1	Option 1: Manual Transfers.....	4
3.2	Option 2: On-Demand Transfers	6
4	Prerequisites	7
4.1	Heartland Settings and Configuration.....	7
4.1.1	Preventing Duplicate Transaction Errors.....	7
4.2	Determine How PaperCut Will Find Heartland Account Ids	8
4.2.1	Populate Heartland Account Ids into PaperCut Card/Identity Numbers	8
4.2.2	Looking Up Heartland Account Ids in a Database	9
5	Installation	9
5.1	Installing the Payment Gateway Module	9
6	Testing	11
6.1	Testing Manual Transfers.....	11
6.2	Testing On-Demand Transfers.....	11
7	Troubleshooting.....	12
7.1	Log files	12
7.2	Improve performance	12

1 Introduction

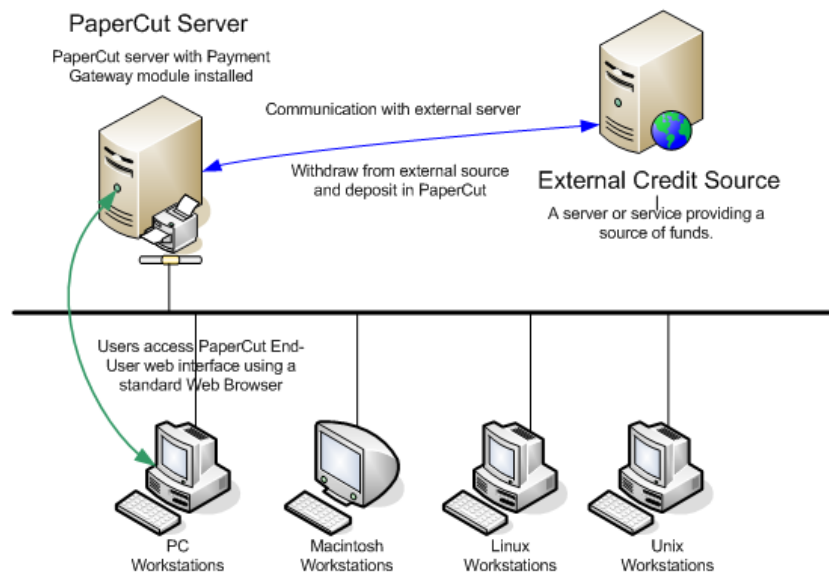
Heartland develops a variety of payment systems. Heartland's OneCard system is Heartland's solution for managing campus payments. The PaperCut payment gateway module for Heartland provides integration that allows students to pay for printing from their Heartland OneCard account.

Key Feature Summary

- Automatic network level integration with Heartland – no mandatory hardware requirement.
- Different operation modes:
 - Automatic transfer on low/zero account balance.
 - End-user instigated transfer via a web interface.
 - Combination thereof as configured by the administrator.
- Full transaction auditing for both end-users and administrators.
- Ability to extract/import Heartland account numbers out of Active Directory (if secure) and/or an external database table such as a student management system.
- Implementation does not compromise/replace standard PaperCut features such as free quotas, overdraft rules or the ability to add alternate payment types in the future (e.g. credit card services such as PayPal).

2 Architecture

The solution is designed with security as the number one objective. All communication with Heartland's OneCard Transaction Gateway (the external credit source) is made via the PaperCut Server. No client software is used. Communication with end-users is encrypted using SSL/TLS browser encryption.



Heartland integration architecture

Setup and testing time for basic integration should take around 30 minutes. More detailed integrations are possible and time should be allocated accordingly after reading through this guide. No system level restart is required; however, the PaperCut application server will be restarted during the install process. If other administrators are using the PaperCut administration interface at this time, it may be advisable to warn them of the pending restart.

3 End-User Overview

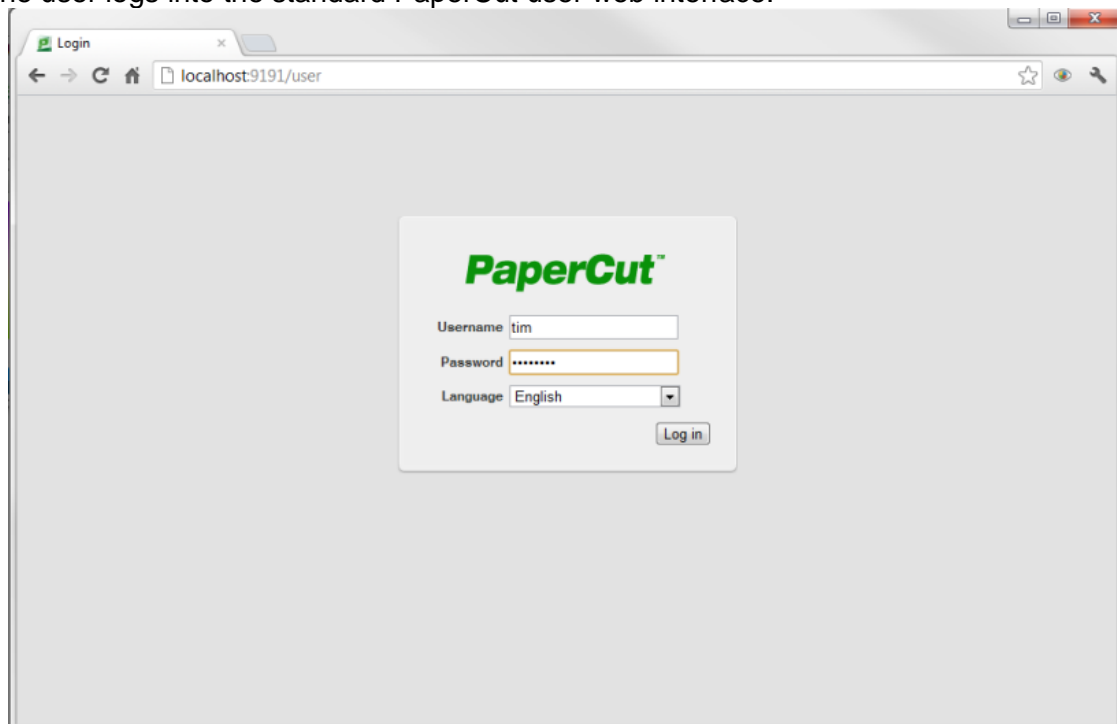
The system may be configured using one or both possible integration options. The integration options are configured by the administrator. The section below provides an overview of the end-user experience for each option. Many organizations may choose to start off with one (e.g. Manual), and move to the other (e.g. On-demand) as their system/project evolves over time.

3.1 Option 1: Manual Transfers

Manual transfers allow users to instigate a transfer of a selected amount from their Heartland account into their PaperCut account.

This option does not require any additional hardware such as card readers.

1 The user logs into the standard PaperCut user web interface.



2 A new “Add Credit” link is available in the navigation menu. (The link will appear after the module is installed. The name of the link is configurable.)

PaperCut

Summary

Username	tim
Balance	\$7.50
Heartland Balance	\$17.60
Total print jobs	228
Total pages	1,567

Activity

Balance history for tim

3 The user clicks “Add Credit” and selects the amount to transfer. (The range of amounts can be defined by the administrator.)

PaperCut

Add credit using Heartland

User	tim
Current Balance	\$7.50
Heartland Balance	\$17.60
Amount to Transfer	<input type="text" value="--- Select the amount ---"/> <ul style="list-style-type: none"> --- Select the amount --- \$2.00 \$5.00 \$10.00

4 After clicking the form Submit button, the selected amount, if available, is placed in the user’s account in realtime.

5 The end-user can view all details of their transaction history via the web interface.

Show Filter (filter active: [remove](#))

« 1 2 »

<u>Transaction Date</u> ▼	<u>Transacted By</u>	<u>Amount</u>	<u>Balance After</u>	<u>Transaction Type</u>	<u>Comment</u>
Aug 10, 2012 4:03:19 PM	admin	\$7.50	\$7.50	Manual adjustment	
Aug 10, 2012 11:18:54 AM	[system] (print)	-\$20.00	\$0.00	Printer Usage	
Aug 10, 2012 11:18:54 AM	[system] (print)	\$20.00	\$20.00	External adjustment	Automatic transfer from Heartland
Aug 10, 2012 11:16:06 AM	[system] (print)	-\$20.00	\$0.00	Printer Usage	
Aug 10, 2012 11:16:06 AM	[system] (print)	\$20.00	\$20.00	External adjustment	Automatic transfer from Heartland

3.2 Option 2: On-Demand Transfers

On-demand transfers occur when a user lacks sufficient balance to print a job. When this happens the system will automatically transfer funds from Heartland into the user's PaperCut account as required, based on rules configured by the administrator.

This option does not require any additional hardware such as card readers.

The amount transferred may be just the amount required to print, or a batch amount. In batch mode \$10 could be transferred whenever the user runs out of balance, for example, reducing the number of transfers that occur.

On-demand transfer example without batching:

1. User Betty has \$0.10 in her PaperCut printing account.
2. Betty prints a job that costs \$0.70.
3. The PaperCut Heartland gateway transfers the required \$0.60 into Betty's PaperCut account and the job is allowed to print.

On-demand transfer example with \$5.00 batching:

1. User Betty has \$0.10 in her PaperCut printing account.
2. Betty prints a job that costs \$0.70.
3. Because the print job costs more than the value currently in her account, the PaperCut Heartland gateway transfers the default batch transfer amount of \$5.00, making a total of \$5.10.
4. The job cost is deducted from the PaperCut account leaving Betty with \$4.40 in her PaperCut account which may be used for future printing.

On-demand transfers are enabled and configured by the administrator. No user intervention is required, although users are provided with a full list of all transfers in the user web interface. If using batch transfers, sites should explain to users how the system works (e.g. default transfer amounts).

Notes:

- On-demand transfers can be enabled for specific groups, e.g. for students but not staff. See the `heartland.manual-transfer.allowed-groups=` configuration option in the `[app-path]\server\lib-ext\ext-payment-gateway-heartland.properties` file.

- Any free quota allocated to users is used first before making transfers from Heartland.
- If the user is out of Heartland funds the “insufficient credit” message will propagate to the user as it normally does in PaperCut (e.g. notified via the client software, or via email).

4 Prerequisites

This section describes the prerequisites required before the payment gateway module can be enabled.

4.1 Heartland Settings and Configuration

Please contact Heartland support for assistance with setting up the system for use with PaperCut.

The PaperCut payment gateway module requires the following information about your Heartland setup:

- The Heartland server hostname or IP address and port number.
- Whether or not an SSL connection will be used to the Heartland server.
- The following site specific information (provided by Heartland):
 - Your Heartland terminal id or address which is typically a number up to 5 digits.
 - The format type of the account. The format of *manual* is used if the account information is the Heartland account id. Whereas the format of *card* is used if the account information is coming directly from the track 2 data of a swipe card. If the format is in *manual* format, then the following extra parameters will be needed:
 - Your Heartland customer number which is typically a number up to 4 digits.
 - Your Heartland account type which is a number between 0 and 3 inclusive.

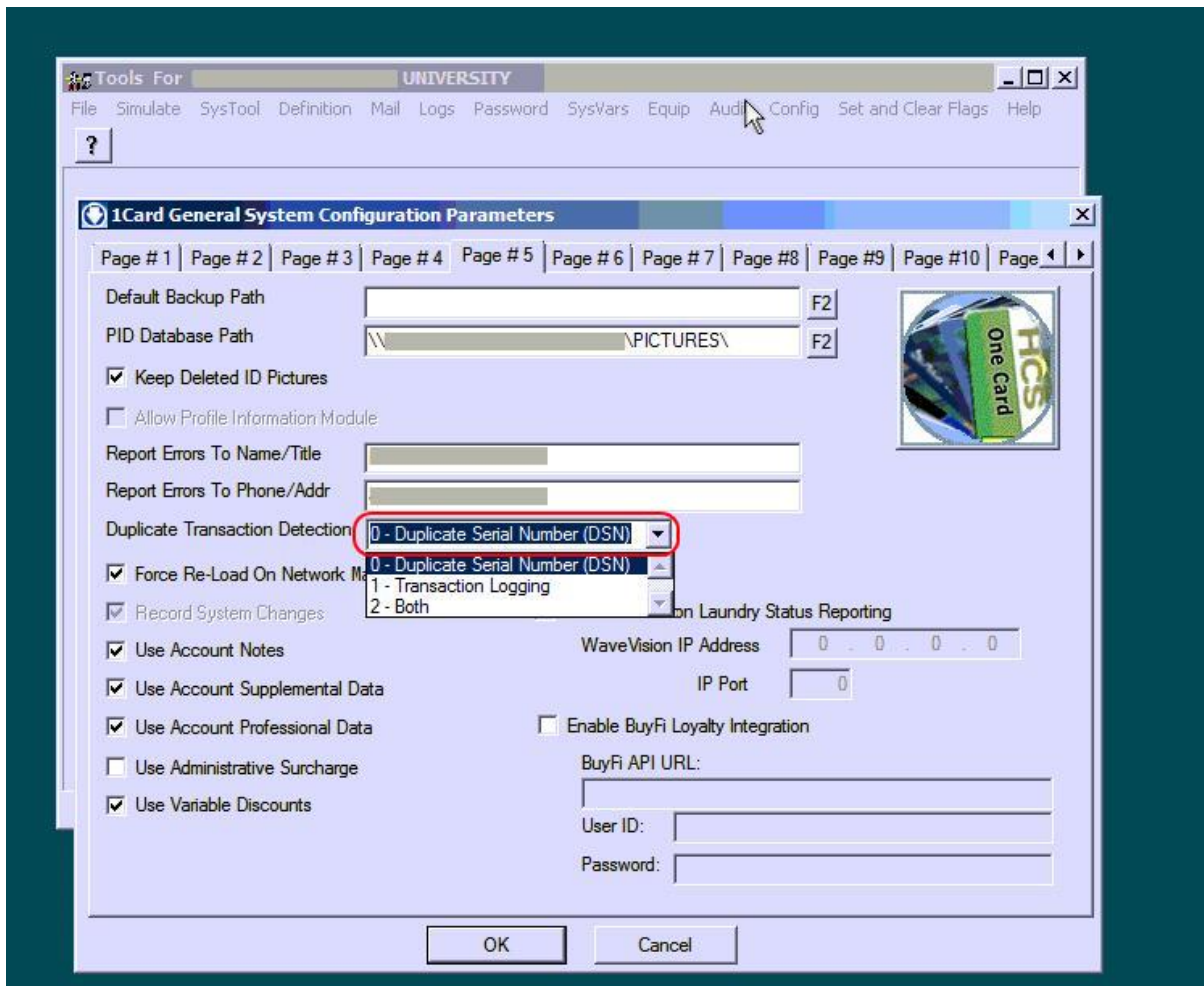
If the format is *card*, the customer number and account type is not explicitly needed as it included in the track 2 data.

These settings will be configured in the payment gateway configuration file.

4.1.1 Preventing Duplicate Transaction Errors

By default, it is possible for a transaction to produce a duplicate error if the customer does a repeated transaction using the same account, same amount, same transaction type and in the same second. For example, if the customer does a balance query of the same account within the same second then this could produce a duplicate transaction error (though we have added delayed balance query retry logic to help prevent this error from happening).

To ensure these types of duplicate transactions are prevented, we recommend that the Heartland configuration be modified using the tools.exe program from the 1Card package (in the 1Card shared folder). Having run tools.exe, open the Config from the menu and go to page 5 of the tab pages entitled “1Card General System Configuration Parameters”. Change the Duplicate Transaction Detection to “0-Duplicate Serial Number (DSN)” only as can be seen in the screenshot below. PaperCut uses the DSN or effectively a sequence number to make subsequent transactions unique.



4.2 Determine How PaperCut Will Find Heartland Account Ids

The payment gateway requires a user's Heartland account id to perform a transaction from their Heartland account. This requires a mapping between a user in PaperCut and their Heartland account id. There are two ways to achieve this:

1. By importing/storing the Heartland account id in PaperCut. This is generally the easiest method if the Heartland account id is readily available for import into PaperCut.
2. By using details about the user that are available to PaperCut to look up / map to a Heartland account id in a database. This is useful when the Heartland account id cannot easily be imported into PaperCut (or a different card/id is in use), but can be easily "looked up".

4.2.1 Populate Heartland Account Ids into PaperCut Card/Identity Numbers

This method involves importing the Heartland account id into PaperCut. This is the preferred method, where possible, as it allows PaperCut to quickly and easily identify the information that needs to be sent to Heartland.

If your card/id numbers are not already populated in PaperCut, take the time to do this now. There are a number of ways this can be done, including:

- Importing the account numbers from Active Directory or LDAP. This is generally the simplest method and easiest to maintain, particularly if the account numbers are

already stored in AD/LDAP. See the PaperCut user manual section “*Importing Card/Identity numbers from Active Directory or LDAP*” for more details.

- Pointing PaperCut to a database that contains the account numbers and can provide a mapping between the account number and the network username. This can be useful if there is a student management system or similar that contains the account numbers and usernames. See the user manual section “*Looking up account numbers in an external database*” for more details.
- Batch importing the account numbers from a text file. See the user manual section “*Batch User Card/Identity Update*” for more details.

4.2.2 Looking Up Heartland Account Ids in a Database

This method involves using some information about a user that is available to PaperCut to look up a Heartland account in a database. E.g. given a table of Heartland account ids and network usernames, PaperCut knows the network username so it can look up the Heartland account id. E.g. given a table of Heartland accounts and student ids, if the student id is stored in PaperCut then PaperCut can use this to look up a Heartland account.

Configuration for database lookups can be found in the configuration file at [app-path]/server/lib-ext/ext-payment-gateway-heartland.properties, under the section titled “*External Account Id Lookup Settings*”. The configuration file provides further information and examples. Contact your reseller or Authorized Solution Center for assistance. You can find their contact information in your PaperCut Admin interface on the About page.

5 Installation

This section covers the installation of the PaperCut payment gateway module. It is written assuming the reader has good server administration skills and is experienced with general PaperCut administration.

Setup and testing time should take around 30 minutes. No system level restart is required; however, the PaperCut application server will be restarted during the install process. If other administrators are using the PaperCut administration interface at this time, it may be advisable to warn them of the pending restart.

5.1 Installing the Payment Gateway Module

The Payment Gateway Module will function during the PaperCut 40-day trial period. After this, the module must be licensed. If you have been supplied with a new license take the time to install this now. The license install procedure is documented in the PaperCut user manual chapter ‘Licensing and Support’.

1. Download the Payment Gateway Module from the PaperCut website:

<http://www.papercut.com/files/pcng/ext/payment-gateway/pcng-payment-gateway-module.exe>

2. Install the module into the same directory as PaperCut. For example:

C:\Program Files\PaperCut NG\

3. Open the file:

[app-path]\server\lib-ext\ext-payment-gateway-heartland.properties

in a text editor such as Notepad.

4. Locate the line `heartland.enabled=N` and change the `N` to `Y`. This will enable the Heartland module.
5. Enable (set to `Y`) one or both of the `heartland.manual-transfer.enabled` and `heartland-demand-transfer.enabled` options (see the End-User Overview section for more detail about these options).
6. Locate the option `heartland.server.host` and enter the hostname or IP address of your Heartland server.
7. Verify that the `heartland.server.port` option is correct for your Heartland server.
8. Locate the option `heartland.terminal-id` and enter the terminal id that PaperCut will use for transactions.
9. Locate the option `heartland.customer-number` and enter the customer number that PaperCut will use for transactions.
10. Locate the option `heartland.account-type` and enter the account type code (0-3).
11. Take some time to review the configurable options available in the file. Options include limits on the amount to transfer, access groups and custom error messages. You may like to enable a group restriction to limit access to administrators until configuration is complete.
12. Save the file and exit the text editor.

6 Testing

The payment gateway module for Heartland is now ready for testing. This test will involve performing a live transaction with the Heartland system, testing end-to-end functionality.

To begin testing you will require a login for a user that has a card/id number that refers to an active account id in Heartland. The simplest way to achieve this is to manually enter a test Heartland account id number in the “Card/Identity Number” field of a test user in the PaperCut administration interface.

6.1 Testing Manual Transfers

(Only if manual transfers are enabled)

1. Log into the PaperCut user web interface at <http://papercut:9191/user> (where papercut is the name of your PaperCut server)
2. A new link called Add Credit should appear on the left. Click this link.
3. Select an amount to add and click Add Value.
4. The requested amount will be transferred from the user’s Heartland account into PaperCut. Check that the balance has been successfully adjusted via the Transaction History page. Check the records in Heartland to ensure that the transfer is correctly accounted for.

6.2 Testing On-Demand Transfers

(Only if on-demand/automatic transfers are enabled)

1. Log into the PaperCut admin interface (<http://papercut:9191/admin>) and select your test user from the **Users** tab.
2. Change the user’s account balance to \$0.00.
3. Login to a workstation as this test user and perform a test print job.
4. The print job should trigger a transfer of credit from Heartland and the print job should be allowed.
5. Login to the PaperCut admin interface and select the test user from the **Users** tab.
6. Select the user’s Transaction History tab, where you should see the funds transferred from Heartland into PaperCut (and then the deducted amount from PaperCut for the cost of the print job).
7. Check that the transferred amount is correctly accounted for in Heartland.

7 Troubleshooting

7.1 Log files

Administrators may find information in the following log files useful when trying to troubleshoot setup/configuration problems or issues reported by end-users.

- **Payment Gateway Event Log:**

[app-path]\server\logs\payment-gateway\event.log

This log contains gateway specific error messages and events.

- **Application Log:**

[app-path]\server\logs\server.log

This log contains general application specific error messages and events.

Transaction Log:

[app-path]\server\logs\payment-gateway\transaction.log

This log contains a list of successful transactions in a tab-delimited form.

Contact your reseller or Authorized Solution Center for assistance. You can find their contact information in your PaperCut Admin interface on the **About** page.

7.2 Improve performance

To improve performance in very large scale environments where Heartland on Demand Transfer is enabled (`heartland.on-demand-transfer.enabled=Y`) and Load Balancers are used, do the following:

1. Upgrade to PaperCut Payment Gateway 18.3.0 or above. For more information, see <https://www.papercut.com/kb/Main/Upgrading>
2. Open the file `ext-payment-gateway-heartland.properties` in a text editor such as Notepad.
3. Add the following line:
`heartland.server.load-balancer=Y`
4. Set the following lines to the Load Balancer's IP address and port, respectively:
`heartland.server.host=Load Balancer's IP address`
`heartland.server.port=Load Balancer's port`