AltaPay & NetSuite Bundle Installation Guide

Through the partnership of AltaPay and NetSuite, we have made it much easier for you as merchant/developer to receive secure payments in your web shop. AltaPay is fully integrated with NetSuite platform via a bundle. All you have to do is to install the bundle, which will only take a few minutes to complete.

We will show you how to complete the installation through a step-by-step guide. Follow this guide to start accepting payments globally.

To make the set-up as smooth as possible for you, this installation guide is divided into following sections:
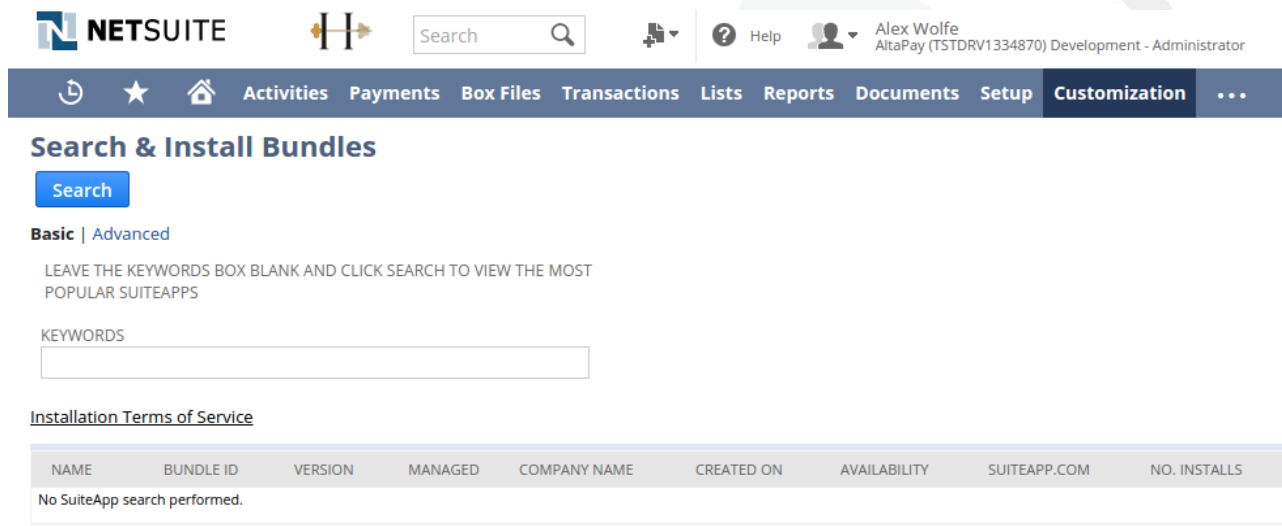
1. Prerequisites

We highly recommend to gather all the below information before starting the installation.

1. Prerequisites for installing the bundle
   a. The bundle 'Payment Gateways' (ID 47196) needs to be installed on the merchant NetSuite account.
   b. Be aware that due to NetSuite limitations consumers do not get informed if the transaction declines/fails. Only the merchant will be notified. This is only the case if NetSuite is used as a frontend.

2. Prerequisites for configuration of AltaPay bundle. AltaPay provides these.
   a. AltaPay credentials
       i. Username
       ii. Password
   b. AltaPay gateway information
       i. Terminal
       ii. Gateway

3. Prerequisites for fraud detection
   a. Agreement with external fraud partner such as ACI ReD or MaxMind.

2. Install the bundle

Installing this bundle will enable your web site to handle card transactions through AltaPay's gateway.
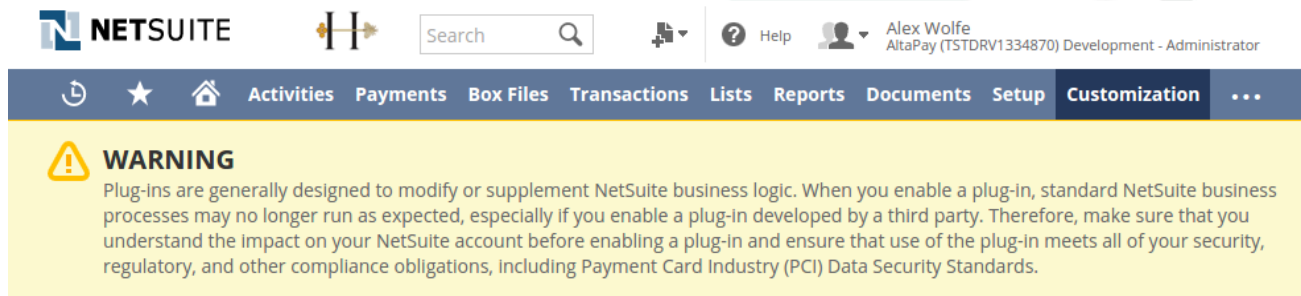
1. Select the account, where you want to install AltaPay
2. Go to: Customization → SuiteBundler → Search & Install bundles
3. In the keywords field enter 'AltaPay'
4. Select 'AltaPay' (Bundle ID: 92258)
5. Click 'Install'
6. Follow the instructions. Note that this is a managed bundle and updates for the bundle is pushed from AltaPay.

3. Enable the bundle

You have now installed the bundle, but before continuing the configuration you need to be sure that the bundle is enabled.

1. Go to: Customization → Plug-ins → Manage Plug-ins
2. Check if the AltaPay bundle is enabled. If not, check the checkbox.

4. Configuration of AltaPay bundle

In this step you can start configuring the AltaPay bundle, so it fits your web shop. This includes adding payment methods and configuring payments.

4.1. Add payment method

The standard configuration connects the bundle with the test gateway. Follow the below steps to connect the bundle with AltaPay's production gateway.

1. Go to Customization → Plug-ins → Plug-in implementations → Click on 'config.js' in the AltaPay bundle → Click 'Edit' → replace 'testgateway.altapaysecure.com' with the gateway information for your production environment (provided by AltaPay). Do not include 'https' → Press 'Save'.

4.1.1. Credit/debit cards

In this step you add card as payment method in the web shop. For each currency you need to go through step 4.1.1.

1. Go to setup → Accounting → Payment Processing Profiles → 'Add AltaPay Plugin Profile'

a) Select which web sites the bundle should cover
b) Add 'Name'. This is for internal use and will not be shown to customers. We recommend to name it 'Payment Method - Currency' (e.g. Credit Card - EUR).
c) Select Subsidiary
d) 'CHARGE CURRENCIES'. This must correspond with the currency of the terminal and only select one currency.
e) Select Settlement Bank account. This is related to accounting and will not affect the settlement agreement with the acquirer.
f) f. 'Authentication Credentials' → Add username and password. You will only be able to see asterisks, since they are password fields.
g) 'AltaPay configuration' → Add terminal. Leave the 'Alternative payment' checkbox unchecked.

h) 'Gateway Request Types' → Add the types that are needed for the terminal.
i) Press 'Save'.

2. Go to Setup → Accounting → Accounting lists → New → Select 'Payment Method'
   a. Select 'PAYMENT METHOD' → Name the payment method. This will be shown in the web shop.

   b. Check 'CREDIT CARD'
   c. 'SUPPORTING MERCHANT ACCOUNTS' → Select the Payment processing profile (ref. 4.1.1. –
   d. step 1.b.) you want to match the payment method with.
   e. 'COUNTRIES' → Check 'DISPLAY IN WEB SITE'
   f. Select the tab 'Payment Visuals'

Flag
web/standard
URL
Place the icon in the File Cabinet and link to the icon or link to an icon on the internet

   g. Press 'Save'

3. Configure payments
This step is mandatory if CVC needs to be enabled for card payments. This is highly recommended and may be required by the acquirer.
1. Go to Setup → Accounting → Accounting preferences → Click the tab 'Items/Transactions'
   a. Check 'USE CARD SECURITY CODE FOR CREDIT CARD TRANSACTIONS' under the 'Payment

Processing' section
   b. Press 'Save'

4. Enable payments on the website
   a. Go to: Setup → Site Builder or Suite Commerce Advanced → Set Up Website
   b. Click edit on the website you want to enable payments for
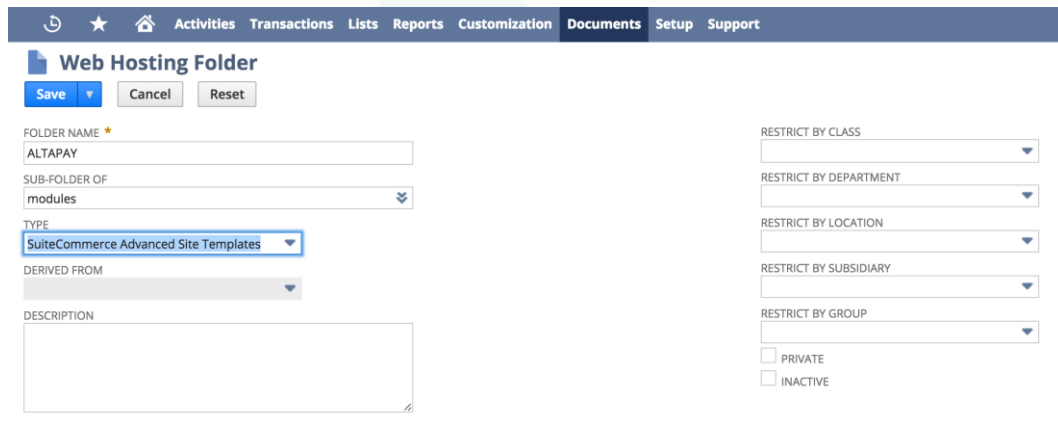
   c. Click the 'Shopping' tab → check 'REQUIRE AUTHORIZATION FOR CREDIT CARD TRANSACTIONS'
   d. Press 'Save'

5. Enable support for Suite Commerce Advanced versions previous to Denali
   If the merchant has Suite Commerce Advanced version previous to Denali these extra steps needs to be followed.
   a. Download the zip file from the GitHub repository
   b. Find the 'Utils.js' in the 'SCA' subfolder → replace 'AltaPay CC',line no: 176, with the name of the payment method created in 4.1.1 - step 2. The name is case sensitive.
   c. Go to: Documents → Files → File Cabinet
   d. Go to: Web Site Hosting Files → Live Hosting Files → SSP Applications → NetSuite Inc. -Checkout 2.05.0 → Custom Checkout → js → src → core

   e. Click on 'Add files' and select the 'Util.js' that you modified.
   f. Find the 'configuration.js' in the 'SCA' subfolder → replace 'AltaPay CC',line no: 176, with the name of the payment method created in 4.1.1 - step 2. The name is case sensitive. → replace the 'img/altapay.jpg' with an image that represent the payment method. The image can be uploaded here: Documents → Files → File Cabinet and navigate to Web Site Hosting Files → Live Hosting Files → SSP Applications → NetSuite Inc. - Checkout 2.05.0 → Custom Checkout -> img
   g. Go to: Documents → Files → File Cabinet → Web Site Hosting Files → Live Hosting Files → SSP Applications → NetSuite Inc. - Checkout 2.05.0 → Custom Checkout → js → src → app → modules → Click on New Folder → name it Altapay
   h. Select SuiteCommerce Site Templates before saving the new folder in the TYPE dropdown

i. Go to: Documents → Files → File Cabinet → Web Site Hosting Files → Live Hosting Files → SSP Applications → NetSuite Inc. - Checkout 2.05.0 → Custom Checkout → js → src → app → modules → Altapay

j. Click 'Add files' and select the 'configuration.js', that you modified

k. Go to: Documents → Files → File Cabinet → Web Site Hosting Files → Live Hosting Files → SSP Applications → NetSuite Inc. - Checkout 2.05.0 → Custom Checkout → js → Click 'DELETE FILES' and remove the following files: 'Application-xxxxx.js' and 'current.js' → Click 'Edit' for the 'combiner.config' → Click on the 'edit' link

l. A popup opens → add the following string 'src/app/modules/Altapay/*.js', as seen in red below

Input-files: utils/BootUtilities.js utils/json2.js libs/require.js   libs/underscore.js
libs/jquery.js libs/backbone.js   libs/backbone.validation.js libs/bootstrap.js
src/core/Main.js   src/core/Utils.js src/core/ApplicationSkeleton.js
               src/app/modules/BackToTop/*.js src/core/extras/*.js   src/app/Application.js
src/app/Configuration.js   src/app/ItemsKeyMapping.js src/app/modules/Account/*.js
               src/app/modules/Address/*.js src/app/modules/AjaxRequestsKiller/*.js
               src/app/modules/Cart/*.js src/app/modules/Content/*.js
               src/app/modules/CMSadapter/*.js src/app/modules/CreditCard/*.js
               src/app/modules/ErrorManagement/*.js src/app/modules/GoogleAnalytics/*.js
               src/app/modules/GoogleUniversalAnalytics/*.js
               src/app/modules/GoogleAdWords/*.js src/app/modules/Facets/*.js
               src/app/modules/ItemDetails/*.js
               src/app/modules/ItemOptionsHelper/*.js src/app/modules/LoginRegister/*.js
               src/app/modules/Merchandising/*.js
               src/app/modules/MultiCurrencySupport/*.js
               src/app/modules/MultiHostSupport/*.js src/app/modules/NavigationHelper/*.js
src/app/modules/Order/*.js src/app/modules/OrderWizard/*.js src/app/modules/Profile/*.js
               src/app/modules/SiteSearch/*.js src/app/modules/Session/*.js
               src/app/modules/UrlHelper/*.js src/app/modules/User/*.js
               src/app/modules/Altapay/*.js   src/app/modules/Wizard/*.js
src/app/modules/CheckoutSkipLogin/*.js src/app/modules/BrontoIntegration/*.js Starter.js
Combined-file: Application.js
Keep-files: 1
Method: Minify
Referring-files: ../*.ssp
Version: 2

m. Save the file and make sure the 'Application-xxxx.js' and 'current.js' files have been regenerated

4.1.2. Alternative payment methods

If you support alternative payment methods (e.g. wallet, ePayment etc.) you can add the alternative web shop's payment methods by follow the below description. Please note that alternative payment as gift cards and some invoice providers (Klarna and Arvato) are not supported. Until March 2016 alternative payments is only available for web shops using SiteBuilder and NOT Suite Commerce Advanced. Support for Suite Commerce Advanced web shops will be available in March 2016 in NetSuite version 16.1. For each currency and payment method you need to go through step 4.1.2.

1. Go to setup → Accounting → Payment Processing Profiles → 'Add AltaPay Plugin Profile'

a. Select which web sites the bundle should cover
b. Add 'Name'. This is for internal use and will not be shown to customers. We recommend to name it 'Payment Method - Currency' (e.g. Invoice - EUR).

c. Select Subsidiary
d. 'CHARGE CURRENCIES'. This must correspond with the currency of the terminal and only select one currency.
e. Select Settlement Bank account. This is related to accounting and will not affect the settlement agreement with the acquirer.
f. 'Authentication Credentials' → Add username and password. You will only be able to see asterisks, since they are password fields.
g. 'AltaPay configuration' →Add terminal. Check the 'Alternative payments
h. Type the index of the website, that the payment processing profile is related to. Default is 1.
i. 'Gateway Request Types' → Add the types that are needed for the terminal
j. Press 'Save'

2. Go to Setup → Accounting → Accounting lists → New → Select 'Payment Method'

a. 'PAYMENT METHOD' → Name the payment method. This will be shown in the web shop
b. Check 'EXTERNAL CHECKOUT'
c. 'SUPPORTING MERCHANT ACCOUNTS' → Select the Payment processing profile (ref. 4.1.1. – step 1.b.) you want to match the payment method with
d. 'COUNTRIES' → Check 'DISPLAY IN WEB SITE'. Select the tab 'Payment Visuals'
e. Add flag, and URL with a picture that represents the payment method
f. Press 'Save'

3. Enabled alternative payment in NetSuite
   a. Go to Setup →Side Builder/SCA→Setup web site.
   b. Select the web site to enable alternative payments
   c. Go to the shopping tab
   d. In the Payments Page, enable 'ALLOW NON-CREDIT CARD PAYMENT METHODS DURING CHECKOUT'

e. Click save.
4. Add payment form
    a. Download the zip file from the [GitHub repository](#)
    b. Find the 'ns_form.html' in the 'Alternative payments' subfolder
    c. Go to: Documents → Files → File Cabinet → Web Hosting Files → Live Hosting → site
    d. Click on 'Add file' and select the 'ns_form.html'

4.2. Address Verification System (AVS)

In order to control AVS cases, the checkbox "Enable AVS" must be clicked in the payment processing profile. Note that AVS must also be supported by the acquirer and is only supported by issuers in the US, Canada and UK.

In the "AVS ACCEPT" text box on the payment processing profile enter the AVS cases that should be accepted. Regarding the AVS cases that need to be put on hold, enter those in the text box "AVS HOLD". When entering the AVS cases, use uppercases and separate them by comma (e.g. A,B,D).

If an AVS result isn't in one of those two textboxes (Accept or Hold), the AVS result will be rejected and the order cancelled. If an AVS result is in both textboxes, the hold case has first priority.

Authentication Credentials

USER NAME *

PASSWORD *

Altapay Configuration

TERMINAL

☐ ALTERNATIVE PAYMENTS

WEBSITE

☐ ENABLE AVS

AVS ACCEPT

AVS HOLD

Here is a list of AVS cases that can occur. As a merchant you will need to decide which cases should be accepted or put on hold. Otherwise the order will be rejected.

AVS	Description
A	Address matches, but zip code does not
B	Street address matches, post code not verified due to incompatible formats
C	Street address and postal code not verified for international transaction due to incompatible formats
D	Street address and postal codes match for international transaction
E	AVS not supported for this industry
F	Full Match - Address and Zip/Postal Code match (UK Only)
G	Address information is unavailable; international transaction; non-AVS participant
I	Address information not verified for international transaction
M	Street address and postal codes match for international transaction
N	No Match on Address (Street) or Zip
P	Postal codes match for international transaction. Street address not verified due to incompatible formats
Q	Unknown response from issuer/banknet switch
R	Retry. System unable to process
S	AVS not supported for this industry
U	Address information is unavailable
W	Nine-digit zip code matches, but address does not
X	Exact. Nine-digit zip code and address match
Y	Yes. Five-digit zip code and address match
Z	Five-digit zip code matches, but address does not

4.3. Reconciliation

Go to: Customization -> Scripting -> Scripts

There is several Scripts within NetSuite, so not all of these are related to the reconciliation process. Go to the script that has the name "Recon.js" and click either view or edit. Then there is a tab called "Execution log" and here you can find the log entries related to the fundings files and funding files alone.

These log entries are needed when you have any problems with the reconciliation and need to contact AltaPay support.

How to setup reconciliation in NetSuite

For each active AltaPay Payment processing profile in NetSuite, there needs to be a custom record, that is created here:

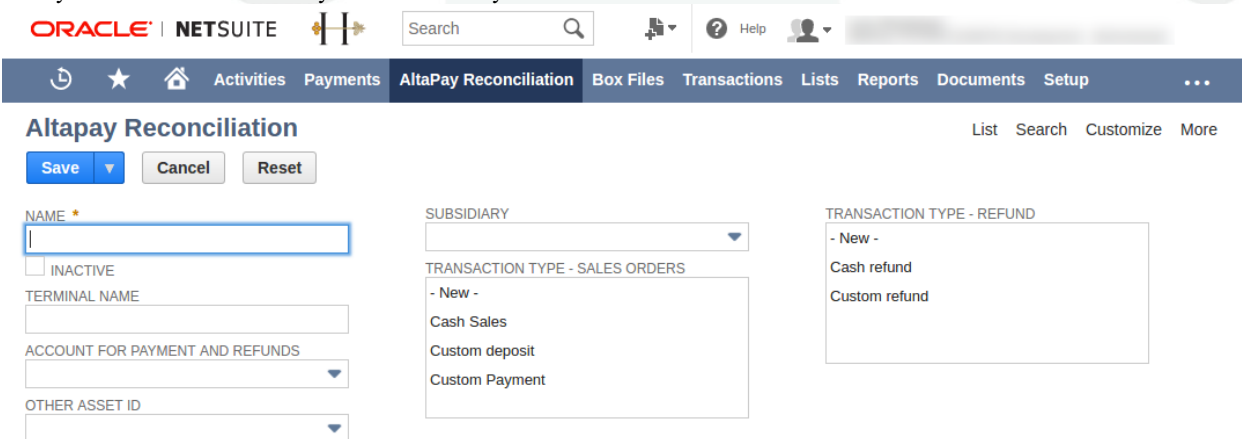AltaPay reconciliation -> Reconciliation Setup -> Setup Accounts.

This will make sure reconciliation entries related to orders will be populated on the correct accounts. Here the terminal name have to match the terminal, which orders that needs to be reconciled.

In the "ACCOUNT FOR PAYMENT AND REFUNDS" field you would need to select the account where entries are related to the order (payment, refund), whereas in the "Other Asset Id" field entries are related to the flow of the transactions (refund/payment rejected, chargebacks (representment, pre arbitration)) that isn't capture/refund.

Transaction type:

- Sales order
  - Select the relevant record types that are related to Sales order.
- Refund
  - Select the relevant record types that are related to refund.

Subsidiary: select the same subsidiary "Account for Payment and refunds" is related

to.

There are also reconciliation entries that are not related to an order and needs to be populated on a different account. Here you would need to add these record types and decide which account they should be populated on. This can be done here:

AltaPay reconciliation -> Reconciliation Setup -> Setup record type.

Then add these (these are case sensitive):

- rolling
- fee
- adjustment
- roundoff
- not_found

Payment method:

Go to: Setup -> Accounting -> Accounting lists

Here the AltaPay payment methods should have the setting: "GROUP WITH UNDEPOSITED FUNDS".

How to upload funding files?

Download the funding files from the gateway. These can be found here: Finances -> Funding files. Download the files that you want to reconcile.

Login into NetSuite and then go to: AltaPay reconciliation -> Import data -> Upload files.

Click on the "Select file" and select a funding file that have been downloaded from AltaPay.

If the funding file was successfully handled, there will be a message "Funding files successfully handled" and there will be an option to upload another funding file. In the case funding data couldn't be populated within NetSuite, log entries will be shown, leading up to the error. For further details, see the log files.

5. FAQ

5.1. Where can I find the log files?
In the case where you experience an issue with the AltaPay bundle we would like you to provide us with the steps leading to the issue and a log file. You can access the log file by following three simple steps.

1. Go to: Customization → Plug-ins → Plug-in implementations
2. Click 'Edit' or 'View' beside 'AltaPay'
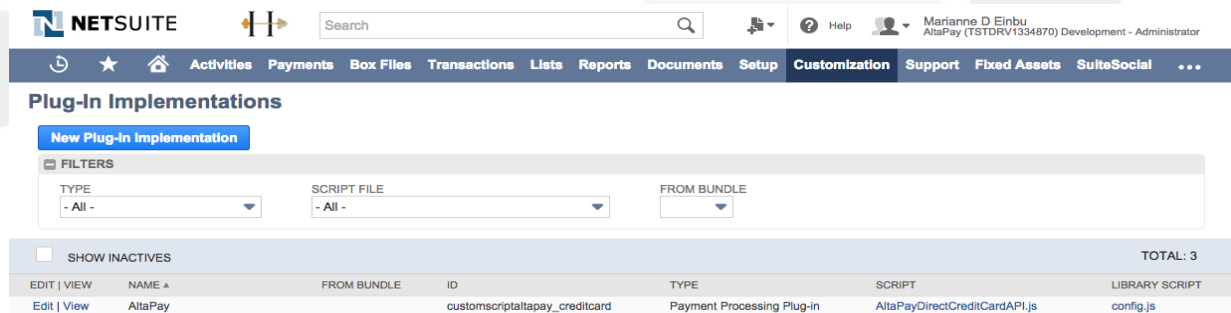
3. Click on the tab 'Execution Log'

5.2. How can I remove the gateway field in the payment processing profile?
In the case, that there is a gateway filed in the payment processing profile, this can be ignored. But you can choose to remove the field by going to: Customization → Lists, Records & Fields → Record types. Then click on 'Altapay_Configuration' and then 'Gateway', you can now check the check box 'INACTIVE' and click 'Save'.

5.3. How does fraud detection works?
If fraud detection is enabled on the terminal, the AltaPay NetSuite app will follow the fraud detection recommendations. The following four recommendations are available:

1. 'Deny'
   The transaction is denied and the order is cancelled as well. You can see the reason here:
   a. Find the order in question
   b. Go to the 'Payment' tab
   c. 'Payment events' → click 'View' and a new window appear
   d. 'Payment Status' → 'DETAILS' → here you can see the message for why the order was denied.
      If you utilize a bespoke version of ReD, the reason for the deny recommendation can be located in the ReD CSI interface.

   Good practice will be to release the payment in our backend. Debit cards do not contain reservations.

2. 'Challenge'
   The transaction is on hold and the merchant will have to make a decision if they want to handle the order. If nothing is done the order will be hold in NetSuite. You can see the reason here:
   a. Find the order in question
   b. Go to the 'Payment' tab
   c. 'Payment events' → click 'View' and a new window appear
   d. 'Payment Status' → 'DETAILS' → here you can see the message for why the order is on hold.

If you utilize a bespoke version of ReD, the reason for the deny recommendation can be located in the ReD CSI interface.

In the case, that the merchant doesn't want to fulfill the order, they have to cancel the order. It will be good practice to release the orders in our backend, so there are no reservations on debit cards. If the merchant will handle this order, the merchant will have to 'Override the payment hold' and fulfill the order. The merchant can find the 'Challenge' orders here:

    a.   Go to Transactions → Sales → Manage Payment Hold

    b.   Dropdown menu 'HOLD REASON' → select 'External Fraud Review' → a list of orders on hold is shown and from here you can accept or decline the order.

3.   'Accept'

    a.   This order will be accepted without any notice

4.   'Unknown'

    a.   This can be due to technical problems at either AltaPay or the fraud company. The order will be handled as accepted.

5.4 How to access log files for the reconciliation process?

Go to: Customization -> Scripting -> Scripts

There is a several Scripts within NetSuite, so not all of these are related to the reconciliation process. Goto the script, that has the name "Recon.js" and click either view or edit. Then there is a tab called "Execution log" and here you can find the log entries related to the fundings files and funding files alone.

These log entries are needed, when you have any problems with the reconciliation and need to contact AltaPay support.

6. Change log

Version	Note
1.0.0	First release. Support for credit/debit cards only
1.1.0	Support for Fraud detection
1.1.1	Fix bug for live production
1.2.0	Added support for alternative payments. NB. only supported for SiteBuilder. No support for Klarna, PaybyBill and gift cards. Support for Suite Commerce Advanced version: 1.0.5 and 1.0.6.
1.4.7	Support for AVS, reconciliation and bug fixes.