

Version: 4.7

Published date: April 2025

Copyright © 2025 by ESW. All rights reserved. Information contained herein is subject to change without notice.

STORED PAYMENT CARDS INTEGRATION GUIDE



**SFCC - ESW Cartridge**

+

Contents

[**1** **Stored Payment Cards Integration** 3](#_Toc93315141)

[1.1 Faster Checkout Experience using Stored Payment Cards 3](#_Toc93315142)

[1.2 How this works? 3](#_Toc93315143)

[1.3 Cookie Details 4](#_Toc93315144)

[1.4 First time Experience- Returning Shoppers prompted to save the cards for later use 5](#_Toc93315145)

[1.5 Repeat Shoppers Experience- Returning Shoppers are presented with the saved cards 6](#_Toc93315146)

[1.6 Testing Sandbox/Staging Environment 6](#_Toc93315147)

# **Stored Payment Cards Integration**

* 1. Benefits

Following are some of the benefits of using stored cards feature.

* This ESW SFCC Plugin feature allows logged-in shoppers to store and re-use stored payment cards for quick checkout experience.
* Stored cards will enable faster checkout which is proven to increase conversion and more internationals sales revenue for retailers.
* It will also encourage the shoppers for repeat purchases.
* Plug-n-Play Standard SFCC plugin feature where No integration efforts required.

In addition to faster checkout, stored cards also results in more satisfying experience for repeat shoppers.

* 1. How this works?

The following diagram explains the workflow between the ESW-SFCC plugin/cartridge, ESW Checkout and ESW Payments.

Timeline

Description automatically generated

The following flow assumes that shopper is already “logged in” , the checkout is executed in a secured context and both retailer and checkout domains use same top level domains.

1. The shopper adds an item to the cart, and clicks Checkout. If not already logged in, shoppers are then presented with two checkout options - a forced login and optional guest checkout.
   1. Once shopper is logged in, ESW SFCC Plugin/cartridge creates order in SFCC with status ‘Created’
   2. Plugin extracts logged in user’s profile ID and assign it to **registeredProfileId** than assembles ESW checkout pre-order service request payload that drives ESW checkout experience when shoppers are redirected to ESW checkout.
2. On Success, ESW Checkout API will return following data in the response
   1. ESW checkout URL where Shopper will be redirected
   2. The created access token is returned as part of the API response in the field *shopperAccessToken*, this value needs to be created as a cookie (esw-shopper-access-token) in the shopper browser so that the ESW Checkout can authenticate shoppers.
3. Upon receiving response, ESW SFCC Plugin will perform following actions
   1. Check if authentication token is present, if yes save the token in the browser cookie and make sure the cookie domain is set with top level domain saved in **ESW Top Level Domain** preference value.
   2. If no token is present, simply redirect shopper to ESW checkout
4. Once shopper lands on ESW checkout and proceed to payment panel, ESW checkout fetch the saved authentication token from cookie, validate the token and then fetch the stored cards by calling ESW Payment API using server-to-server communication.
5. ESW Payment returns shopper’s stored cards which are then displayed as easily selectable drop down options in the payment panel. Saving shopper to locate the card and manual entry.
   1. Cookie Details

**Name:** esw-shopper-access-token

**Max-Age**: Depends on SFCC platform’s native session configuration

**Domain:** {your checkout domain} (e.g. mycheckoutdomain.com).

**ESW Checkout Configuration -> ESW Top Level Domain** Site preference in SFCC BM can be used to configure this value.

**Secure:** True\Enabled

**Sample Http Cookie Header:**

Set-Cookie: esw-shopper-access-token=abcdef; Secure; Domain=mycheckoutdomain.com; Max-Age=3600

**Non Headless Architect – Cookie creation**

Sitegenisis/SFRA storefront stored cards cookie is created through ESW SFCC plugin.

**Headless Architect – Cookie creation**

Headless storefront developers will need to create the stored cards cookie with above mentioned cookie details, as ESW SFCC plugin cannot control headless storefront to create client side cookies.

* 1. First time Experience- Returning Shoppers prompted to save the cards for later use

Graphical user interface

Description automatically generated

* 1. Repeat Shoppers Experience- Returning Shoppers are presented with the saved cards

Graphical user interface, application

Description automatically generated

* 1. Testing Sandbox/Staging Environment

There are two options to test this feature on sandbox/staging environment.

1. Top level domains should be same for both SFCC testing instance and the ESW checkout.
2. If sandbox top level domain is different than the ESW checkout tenant in that case need to add **esw-shopper-access-token** cookie manually for ESW checkout to enable the stored cards feature. The token value can be found in the preorder response in the SFCC Logs. Please note this step is only required for staging/sandbox.
   1. Review that the **esw-shopper-access-token** is created for SFCC top level domain as can be seen in following screenshot for following domain.  
      **zzqy-004.sandbox.us01.dx.commercecloud.salesforce.com**  
      A screenshot of a computer

      Description automatically generated
   2. Once redirected to ESW checkout update  **esw-shopper-access-token** cookie's domain with ESW Checkout top level domain (.[eshopworld.com](http://eshopworld.com/)).A screenshot of a computer

      Description automatically generated with medium confidence
   3. Once the cookie is created for and ESW checkout top level domain, you can see the stored cards payment option gets enabled on ESW checkout as following snapshot.  
      Graphical user interface

      Description automatically generated