Integration overview

To integrate 3DS2 authentication with a merchant's or payment gateway's eCommerce site, the checkout process of the eCommerce site needs to implement a **3DS Requestor** as per the EMV 3D Secure 2.0 specifications. The **3DS Requestor** implementation in the checkout process will communicate with **ActiveServer** via it's Authentication API and assist with the browser information collecting and 3DS Method process (if any) to finish a 3DS2 authentication.

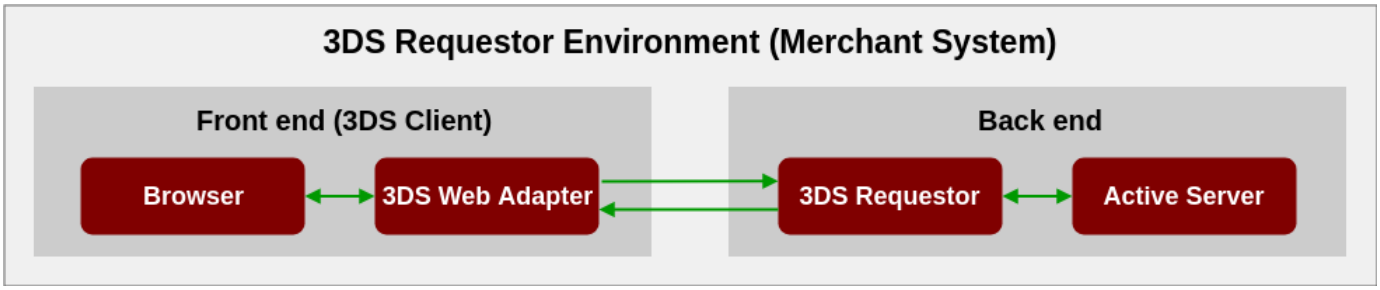
ActiveServer provides a reference implementation of a **3DS Requestor** in the form of source code to help clients implement the **3DS Requestor** process in their existing checkout process.

ActiveServer allows external components such as the **3DS Requestor** to access it's features via Restful APIs. These API calls are operations that an application can invoke at runtime to perform certain tasks. All API requests and responses are in JSON format, which is a lightweight format for transporting data.

> **API documentation**
>
> For details of the API documentation, refer to the API document overview.

The **Integration** section of the documentation provides an introductory guide on how to implement a **3DS Requestor** in your merchant site and integrate with **ActiveServer**, then perform a test transaction. For information regarding merchant **App** integration, refer to the **ActiveSDK documentation**.

To utilise 3DS2, the merchant site needs to implement two parts: a **3DS web adapter** at the front end and a **3DS Requestor** at the back end. The following diagram shows the relationship between the browser, the 3DS web adapter, the 3DS Requestor and **ActiveServer**:



- **3DS web adapter** - The 3DS web adapter is a javascript component provided by the **GPayments 3DS Requestor Demo** and is used to pass 3D Secure data from the consumer device to the 3DS Requestor and assist with the browser information collecting/3DS Method process. This component also processes callback events and page forwarding from **ActiveServer**.
- **3DS Requestor** - The 3DS Requestor is the backend component implemented to act as a bridge between the 3DS web adapter and **ActiveServer**. It receives the 3DS authentication requests from the 3DS web adapter, formulates the requests, and sends the requests to **ActiveServer**. It also receives the authentication results from **ActiveServer** and forwards the results to the 3DS web adapter.

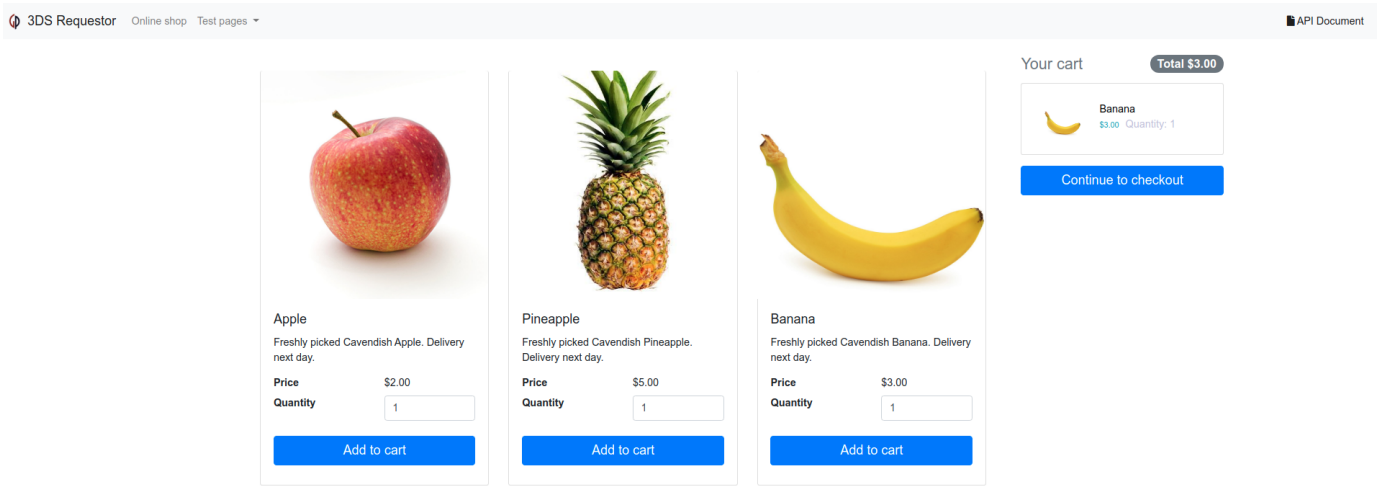
Making a transaction

To simulate a transaction with 3DS2, you can use this demo merchant website to see how the Authentication API works.

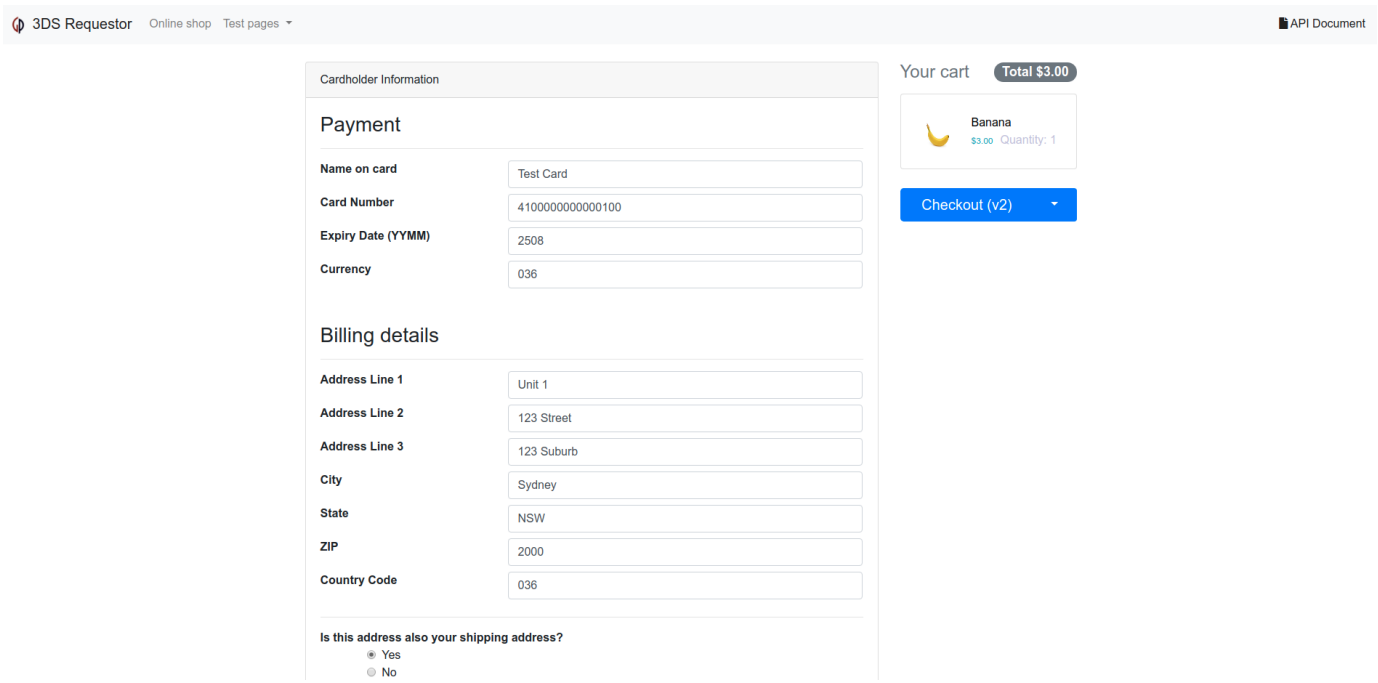
> **Tip**
>
> As this demo merchant website is used as an example throughout this integration guide, please try using it before now before continuing with integration. All the features are explained here for reference.

Frictionless flow

To initiate a frictionless transaction, open the demo merchant website, launch the **Online shop** page, and add an item to the cart.



Select the ***Continue to checkout*** button to move to the checkout page.

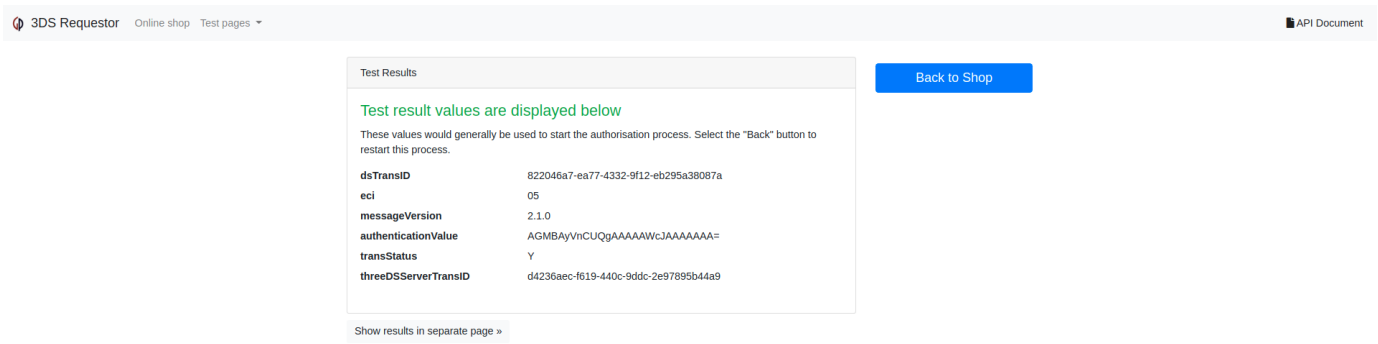


Default payment and billing information has been pre-filled, including a card number, which can be used to complete the transaction. Select the ***Checkout*** button to trigger the 3DS2 authentication process.

> **Note**
>
> You can select the API version to perform the 3DS2 authentication process by clicking the arrow at the right of the ***Checkout*** button.

The **3DS web adapter** will collect the cardholder information and send it to the **3DS Requestor**. The **3DS Requestor** will formulate this into an API request and forward it to **ActiveServer**, which

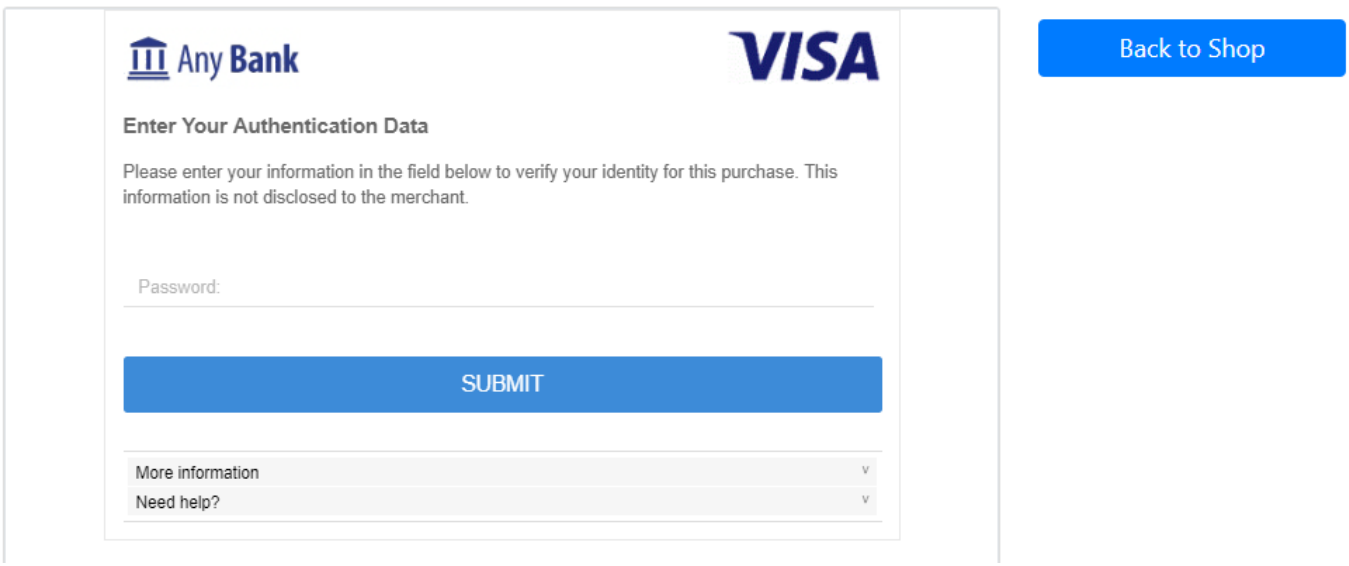
will initiate 3DS2 messaging. The **3DS Requestor** will then wait for the authentication result and forward the result back to the **3DS web adapter**, to be displayed on the following web page.



This completes a transaction using the **frictionless flow**. The simulated transaction was deemed as low risk and hence, no challenge was required.

Challenge flow

To test the challenge flow, select the ***Back to Shop*** button and again add an item to the cart and go to the checkout page. This time, use the card number **4100000000005000** and checkout. In this simulation, the transaction has been deemed as high risk and further cardholder interaction is required, thereby initiating the **challenge flow**. The following challenge screen will be displayed, for this demo the password is **123456**.



Entering the password should result in a successful transaction. In a production scenario, this challenge method could be a variety of different methods, such as **OTP** or **biometrics**, depending on the issuer's ACS and authentication methods registered with the cardholder.

What's next?

Select ***Next*** to learn more about the **Authentication processes**.