3DS1 Integration

> **3DS 1.0.2 only supported for GPayments SaaS product**
>
> 3DS 1.0.2 integration is only supported for SaaS clients, and is not available for in-house deployments.

ActiveServer now supports 3DS1 authentication for SaaS clients. To integrate 3DS1 authentication with a merchant's or payment gateway's eCommerce site, the checkout process of the eCommerce site needs to implement a **3DS1 challenge process** which involves an API call to the **ActiveServer** backend, and a page flow to host the challenge process initialised by the ACS.

Summary of ActiveServer's 3DS1 authentication process

The steps below show how to execute a 3DS1 authentication:

1. The **3DS Requestor** (or the merchant checkout process) determines whether a 3DS1 authentication is required.
2. The **3DS Requestor** (or the merchant checkout process) calls **ActiveServer** 3DS1 Auth API with a `ThreeDS1AuthReq` message
3. The **3DS Requestor** redirects current checkout page to the challenge page that is returned by the response `ThreeDS1AuthResp` in step 2
4. The challenge page executes ACS cardholder authentication process and returns the authentication results via a POST form to the result notification URL that was provided by the **3DS Requestor** in message `ThreeDS1AuthReq` in step 2.
5. The authentication result is available for the **3DS Requestor** via the result notification URL. The **3DS Requestor** then processes the result accordingly.

> **Auth API authentication**
>
> Please note the authentication for the 3DS1 Auth API calls uses the same merchant/masterAuth certificate as the existing 3DS2 process, refer to Auth API Authentication for details of certificate usage.

Step 1: Determine 3DS1 authentication

Before executing a 3DS1 authentication process, users may want to check if 3DS1 authentication is necessary or not. The **3DS Requestor** can implement a static protocol routing process that initiates a 3DS1 authentication process based on cardholder or merchant information, or utilise **ActiveServer**'s Enrol API to check if the PAN is enrolled with 3DS2 or not. **3DS Requestor** can proceed with 3DS1 authentication if the Enrol API call returns `00` (Not Enrolled with 3DS2) if 3DS authentication is required.

> **3DS 1.0.2 authentication determination**
>
> Whether a transaction should be authenticated by 3DS1 or 3DS2 is up to the **3DS Requestor** determination. The use of the Enrol API is not mandatory, it is provided as an option for determining whether a 3DS1 authentication is required.

Step 2: Calling 3DS1 Auth API

There are 2 parts of the code for 3DS1 integration: a backend that initialises the Auth API call and hosts a result notification page, and a frontend Javascript method that checks the Auth API call response and redirects/hosts the ACS challenge page.

ActiveServer's 3DS1 integration supports the same languages and frameworks as the 3DS2 integration. Before proceeding with the 3DS1 integration, refer to Integration Introduction to setup your local test environment with the provided demo requestor code and the language/framework of your choice.

Before calling the 3DS1 Auth API, a proper client certificate is required to setup the mutual authentication TLS communication with **ActiveServer**, refer to Backend Implementation v2 for details.

The backend code snippet below shows how to make a `ThreeDS1AuthReq` call:

```Java tab= //MainController3DS1.java @ResponseBody @PostMapping(value = "/3ds1/auth") public ThreeDS1AuthResp auth(@RequestBody ThreeDS1AuthReq req) { logger.info("3ds1 auth request received: {}", req); return threeDS1Service.handleAuthRequest(req); }

//ThreeDS1Service.java ThreeDS1AuthResp handleAuthRequest(ThreeDS1AuthReq request) {
```

```
//generate the transaction id, this is optional.
request.setThreeDSRequestorTransID(UUID.randomUUID().toString());

logger.info("sending 3ds1 auth reques
```