Apple Pay

Overview

Apple Pay is a digital wallet and mobile payment option developed by Apple Inc. that facilitates payments in person, in iOS apps, and on the web using the Safari browser. It can be used on any Apple device: iPhone, iPad, Mac, and Apple Watch.

Availability

Apple Pay is not available on devices that are not made and sold by Apple.

To check in which countries or regions you can use Apple Pay, read more on availability and geographical coverage >. If you need more information on Apple Pay, you can also refer to these resources: and Apple Pay Brand guidelines.

Access

To see the available permission levels for a specific merchant role check the Users and merchants roles documentation.

Benefits

As a merchant, you can:

- Increase your conversion rate in the US, where Apple Pay is a preferred payment method
- Offer your customers localized payment methods making your business more attractive for your shoppers
- Simplify payment acceptance across different sales channels

Workflow

Verifone processes B2C payments with Apple Pay via your acquirer (i.e., Verifone is not involved in the funds' flow) and provides Hosted Checkout, Pay By Link, Portal for Payment Actions (capture, cancel, refund), transactions reporting, 3DS report, and settlements reporting.

If you are already using Verifone's Hosted Checkout, once you activate Apple Pay as a payment method, this will be displayed in your shopping cart. When your shoppers choose to pay with Apple Pay, they are redirected to the Apple Pay payment flow to finalize their purchase. Then, Verifone returns the transaction response/confirmation to you (merchant). After payment confirmation, Verifone redirects the shopper automatically back to your shop.

Activate Apple Pay

To be able to use Apple Pay as a payment method, follow these steps:

> To enable the ApplePay wallet with Mobile SDKs check the Apple Pay | Online payments | Verifone Developer Portal information.

Via Verifone's Hosted Checkout

> **Note**: The wallet will function only if it is created on the same organization as the Payment Provider Contract (PPC).

1. Log in to your Verifone Central account.

---

2. Navigate to **Administration** and click on **Wallets,** under the *Advanced Settings* tab.
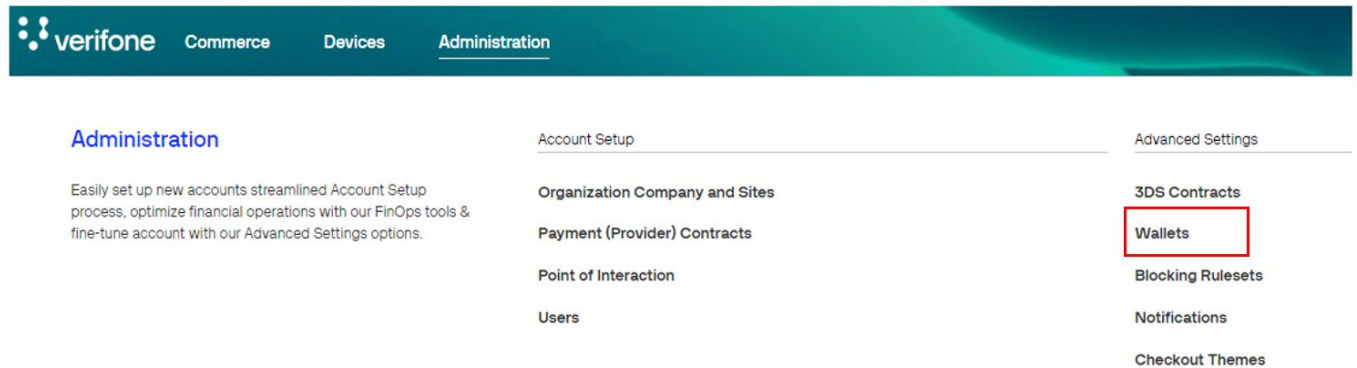

3. To add a new wallet, click on the **Add new Wallet** button.

4. Select your Organization from the drop-down list and select the **Apple Pay for Web** button, from the *Wallet type* section.

> **Note**: The wallet will function only if it is created on the same organization as the Payment Provider Contract (PPC).
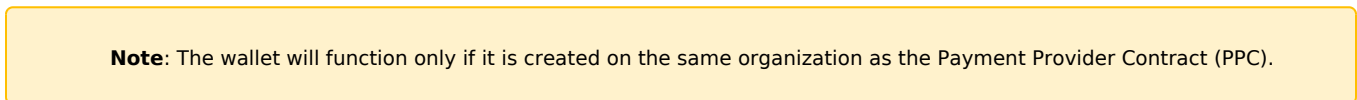
5. Click on **Add wallet** to add the Apple Pay wallet to your payment methods portfolio.
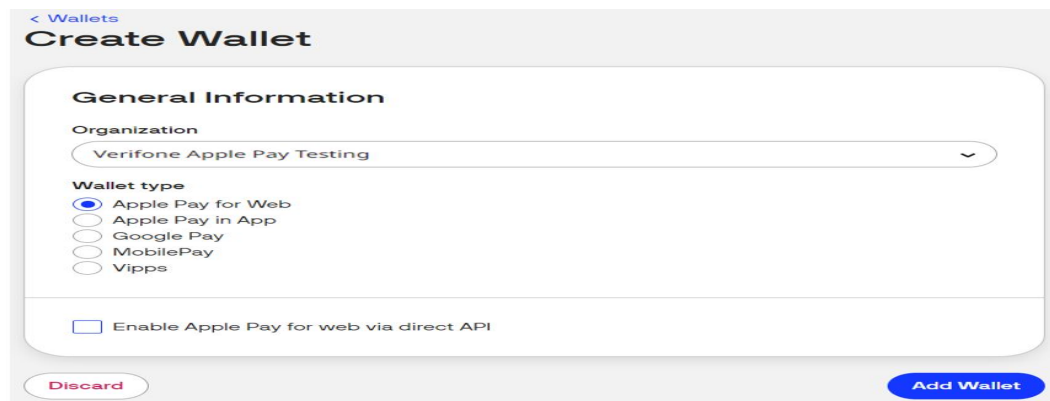
verifone

Via Direct API

If you want to enable Apple Pay via the merchant domain and use it through Verifone's eCommerce API, you need to follow the below steps:

- Tick the **Enable ApplePay for web via direct API** checkbox.
- Add the .txt file into your domain host at MERCHANT_DOMAIN/.well-known/apple-developer-merchantid-domain-association.txt
- Click on **Add wallet**.

< Wallets

Create Wallet

General Information

Organization

RTS FirstData ⌄

Wallet type

◉ Apple Pay for Web
◯ Apple Pay in App
◯ Google Pay
◯ MobilePay
◯ Vipps

☑ Enable Apple Pay for web via direct API

Wallet information

Merchant URL

https://cobalt.io

The merchant URL can be updated from the
Organization details.
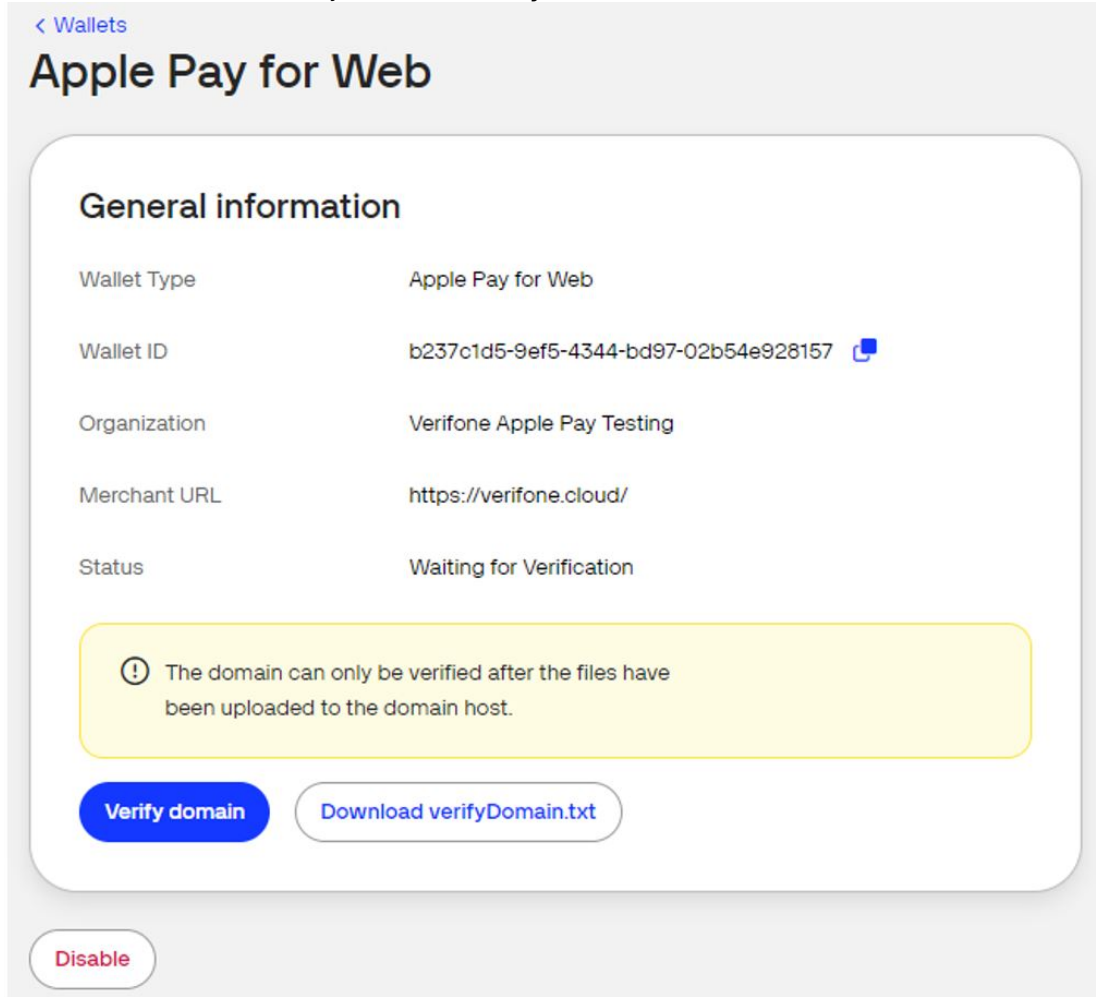
Domain verification

For merchants to enable Apple Pay via their own domain, we should be able to verify domain with Apple. The domain can be verified once the below file has been uploaded to its domain host.

Download verifyDomain.txt

Discard   Add Wallet

- If the .txt file was added correctly, click on the **Verify domain** button.

- Once the domain is verified, you should see the status of the wallet as *Active*.

Integrate Apple Pay as a payment method

You can accept payments through Apple Pay via Checkout (Hosted Payments Page - HPP) or via eComm API.

Once you set up an Apple Pay wallet for your Organization, the next step is to create a checkout. To do this, you need to include the code snippet below:

- HPP
- Iframe

Here is an example body for creating a Checkout to be used as a Hosted Payments Page (default interaction_type):

```
curl --location --request POST 'https://checkout_dev.dimebox.com/v2/checkout' \
--header 'Content-Type: application/json' \
--header 'Authorization: Bearer \
--data-raw '{
"currency_code": "EUR",
"amount": 1000,
"entity_id": "a6b45428-0296-4394-8ddd-0b50b5b20aa9",
"customer": "e4cfeb8d-0e36-4478-8a01-88950d17d1b0",
"configurations": {
"apple_pay": {
"dynamic_descriptor": "SNEAKERS DELUXE",
"card": {
"payment_contract_id": "e4ef1f24-4e95-4cdc-be6a-4f18ed2056ad",
"authorization_type": "FINAL_AUTH",
"sca_compliance_level": "FORCE_3DS",
"threed_secure":

Unknown macro: { "enabled"}
}
}
},
"merchant_reference": "ORDER-1234",
"return_url": "https://enzxiezwyvbcusi.m.pipedream.net",
"i18n":

Unknown macro: { "default_language"}
}'
```

The response for creating the Checkout will look like this:

```
{
"id": "e09d31b5-231a-4b26-a3aa-0bbfa901412a",
"url": "https://checkout_dev.dimebox.com/v2/checkout/e09d31b5-231a-4b26-a3aa-0bbfa901412a/view"
}
```

Here is an example body for creating a Checkout to be used as Iframe:

```
{
"amount": 74.55,
"currency_code": "EUR",
"entity_id": "{{entity_id}}",
"configurations": {
"{{payment_method}}": {
// varies per payment_method
}
},
"merchant_reference": "SNKRS-7001",
"return_url": "{{merchant_return_url}}",
"interaction_type": "IFRAME"
}
```

The response for creating the Checkout will look like this:

```
{
"id": "38615263-ed67-4774-bddd-7407edc0b700",
"url": "{{host}}/v2/loader.js?checkoutId=38615263-ed67-4774-bddd-7407edc0b700"
}
```

Payment actions

Here is an example body for creating a Checkout to be used as Iframe:

```
{
"amount": 74.55,
"currency_code": "EUR",
"entity_id": "{{entity_id}}",
"configurations": {
"{{payment_method}}": {
// varies per payment_method
}
},
"merchant_reference": "SNKRS-7001",
"return_url": "{{merchant_return_url}}",
"interaction_type": "IFRAME"
}
```

The response for creating the Checkout will look like this:

```
{
"id": "38615263-ed67-4774-bddd-7407edc0b700",
"url": "{{host}}/v2/loader.js?checkoutId=38615263-ed67-4774-bddd-7407edc0b700"
}
```

You can perform the following payment actions using Apple Pay.

- Payment actions via Verifone Central

    You can process transactions in Verifone Central with Apple Pay only via Pay by Link.

- Payment actions via eComm API - use https://verifone.cloud/api-catalog/verifone-ecommerce-api#operation/readTransaction.

Stored credentials for Apple Pay

Stored Credentials for Apple Pay are supported as long as the acquirer supports both Stored Credentials and Apple Pay.

To perform a "Sign Up" Apple Pay transaction, the "Initiate a wallet payment using Google Pay or Apple Pay" API endpoint should be used.

To perform a "Charge" Apple Pay transaction, an "Initiate a card payment" API call should be submitted and use a re-use token as well.

For more details related to Stored Credentials, access our Stored Credentials page.

Reporting

You can see and export transaction reports, 3DS transaction reports, and settlement reports for the Apple Pay wallet right from your Verifone Central account.

Apple Pay Test User Setup

To create a testing card for Apple Pay, you should first set up a Sandbox account in the Apple App Store.

Once the Sandbox account is created, you can add an Apple-provided test card. This test card can then be used to test Apple Pay transactions in the Verifone Sandbox environment.

We recommend that you always use the Apple-provided test cards as other cards that are utilized in the payloads outside of this list will fail. Read more on the Apple Pay Sandbox Testing here.