



# Apple Pay

Integration Guide

Version 6.2.2

As of: 09.02.2017

# Table of Contents

- About Apple Pay ..... 4**
  - What is Apple Pay ..... 4
  - Web payment..... 4
  - In-app payment..... 4
  - Process flow chart..... 5
- Step by Step Set-Up Guide for Apple Pay ..... 7**
- Paygate interface ..... 13**
  - Definitions ..... 13
  - Calling the interface ..... 13
  - Capture / Credit / Reversal ..... 15
  - Batch processing via the interface..... 15

## Document history

Date	Name	Change
09.02.2017	Peter Posse	Parameter Channel added
22.12.2016	Peter Posse	First version

# ABOUT APPLE PAY

---

## What is Apple Pay

Apple Pay is a digital wallet for storing payment details, providing an easy and secure way to pay in iOS applications, websites running on Safari browser and contactless POS terminals. In applications and on websites, users can quickly and securely provide their payment, shipping, and contact information to check out with just one touch using Apple's Touch ID. Apple Pay's simplicity increases conversion rates and new user adoption that come with it.

Computop's Apple Pay solution supports following scenarios:

- In-application payments (also part of the Computop SDK)
- Web payments

## Web payment

Apple Pay on the Web enables purchases while using Safari web browser. For shopping on iPhone or iPad, after selecting checkout with Apple Pay, a payment sheet appears, prompting the customer to confirm payment via Touch ID. For shopping on Mac, customers need to have an iOS device in close range, and they'll be prompted on that device to authorize the payment, which will then synchronize to the browser. The latest Mac Book Pro allows customers to authorize payment directly on the Mac Book.

Apple Pay requirements:

- iOS 10. Apple Pay Web payments are supported on all iOS devices with a Secure Element.
- macOS 10.12. Apple Pay Web payments are supported in Safari.
  - The user must have an iPhone or Apple Watch that can authorize the payment.
  - On the latest Mac Book Pro payments can be authorized directly via Touch ID
- All pages that incorporate Apple Pay must be served over HTTPS.
- Your website must comply with the Apple Pay guidelines. For more information, see [Apple Pay on the Web Acceptable Use Guidelines](#)
- [Server requirements](#)

## In-app payment

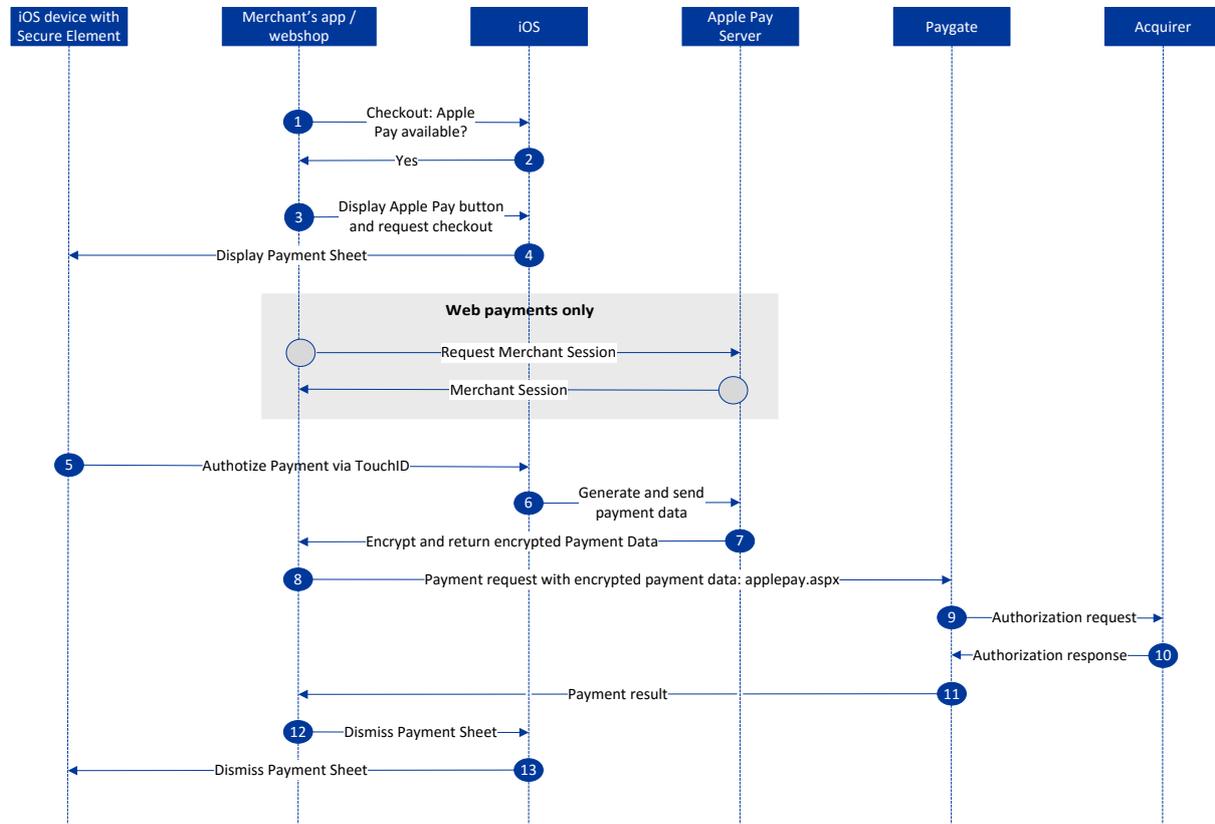
Apple Pay can be used for payments within the iOS applications. The main advantage of Apple Pay is that it is frictionless process with no need to re-type credit card data, shipping or billing address, which can be cumbersome on the smaller iPhone screens. Customer just choose to checkout with Apple Pay and confirms the payment with Touch ID.

Merchants can also use Computop's SDK for facilitating the in-app payment integration, making it easier and faster.

Apple Pay requirements:

- Apple Pay in-app payments are supported on all iOS devices with a Secure Element.

## Process flow chart



Apple Pay process flow

### Presenting the Apple Pay button

Within apps, PassKit provides the API's that your app will use to determine if it is running on a device with a Secure Element and if the device has been provisioned with payment cards that you support. On websites, WebKit provides APIs that allow your website to check if the individual has an Apple Pay capable device and if it is set up.

If the device is Apple Pay enabled you should present the Buy with Apple Pay button using APIs supplied within PassKit within apps and Webkit within Safari.

### Presenting the Payment Sheet

When your user selects goods or services to buy, and selects Apple Pay as the payment method, you create a payment request and ask PassKit in apps or Webkit on your website to present the payment sheet to the user. The payment sheet must immediately follow the user tapping the Apple Pay button, without any interim screens or pop-ups except to prompt for necessary product details, such as size or quantity.

Your app specifies the contents of the payment sheet but it does not control the user's interaction with the sheet. You must decide if it makes sense to present shipping and billing information, shipping method, and other line items to the user. You should only request the information necessary to process the transaction.

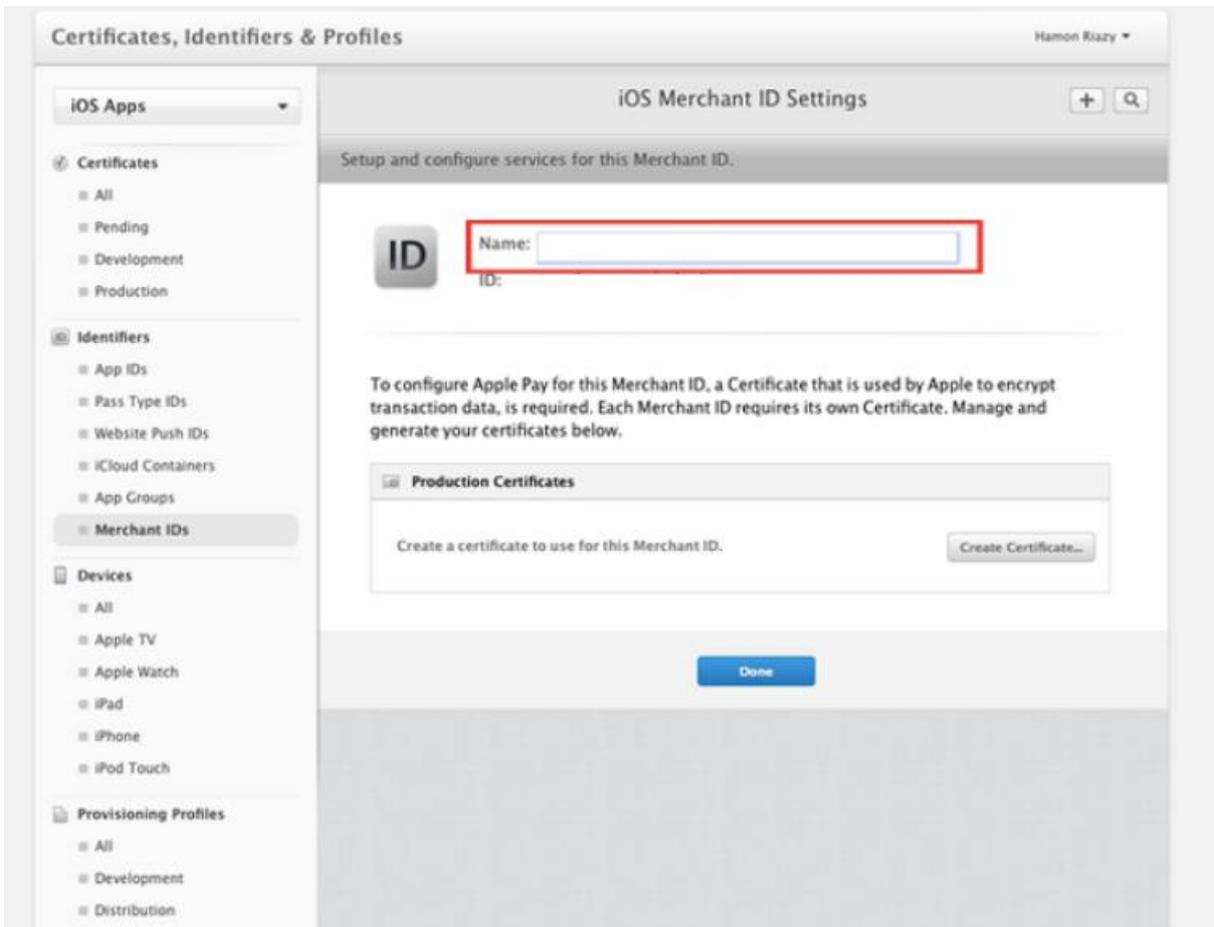
## **Processing Payments**

Once authorized by the user with Touch ID (and bank PIN code in China), your app receives a payment token from PassKit. The payment token encapsulates the information needed to complete a payment transaction, including the device-specific account number, the amount, and a unique, one-time-use cryptogram. The encrypted payment bundle will be decrypted by Computop, which then handles the whole processing of the transaction.

# STEP BY STEP SET-UP GUIDE FOR APPLE PAY

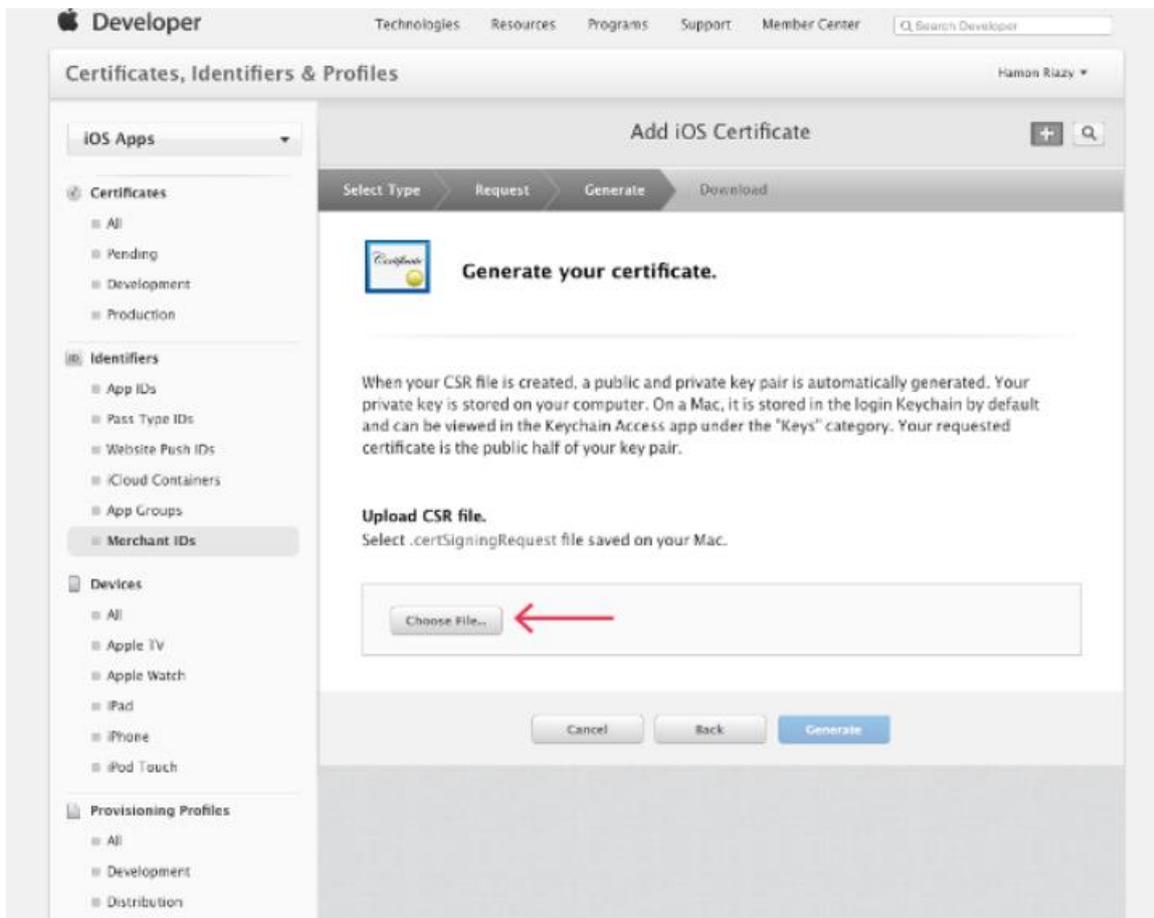
Once you have decided to support Apple Pay, there is a set of steps that need to be completed:

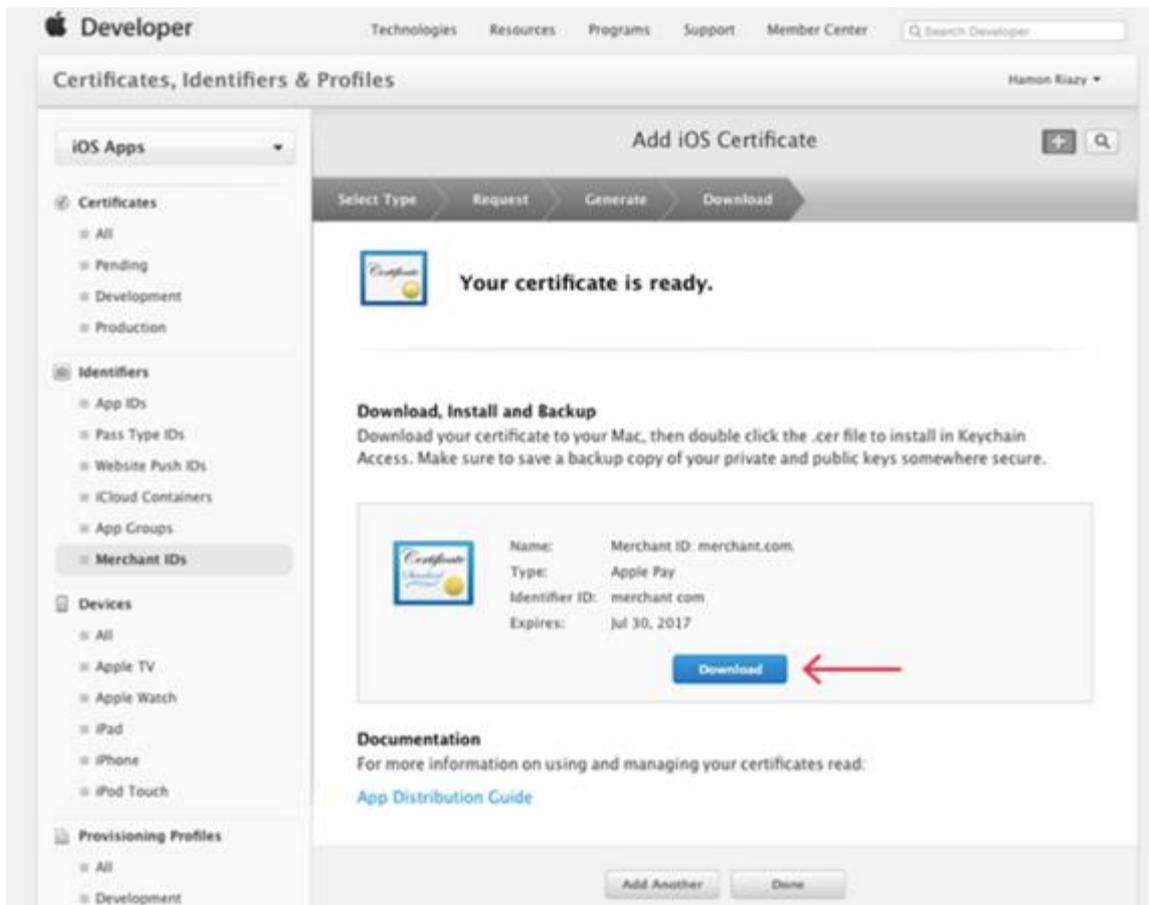
1. Validate if the Acquirer supports Apple Pay
  - a. Not all Acquirers are supporting Apple Pay for in-app and web payments
  - b. Computop can provide information about supporting Acquirers
2. Create **Merchant ID**, unique identifier that represents a merchant for Apple Pay
  - a. Access your Apple Developer Account and visit the *Certificates, Identifiers & Profiles* section of the Apple member center. You can then navigate to *Merchant IDs* and click *add*.
  - b. Set your Merchant ID. The identifier you enter should be a reverse DNS style identifier prefixed with the word



3. Create **Payment Processing Certificate**, used to securely transfer payment data. Apple Pay servers use the payment processing certificate's public key to encrypt the payment data.

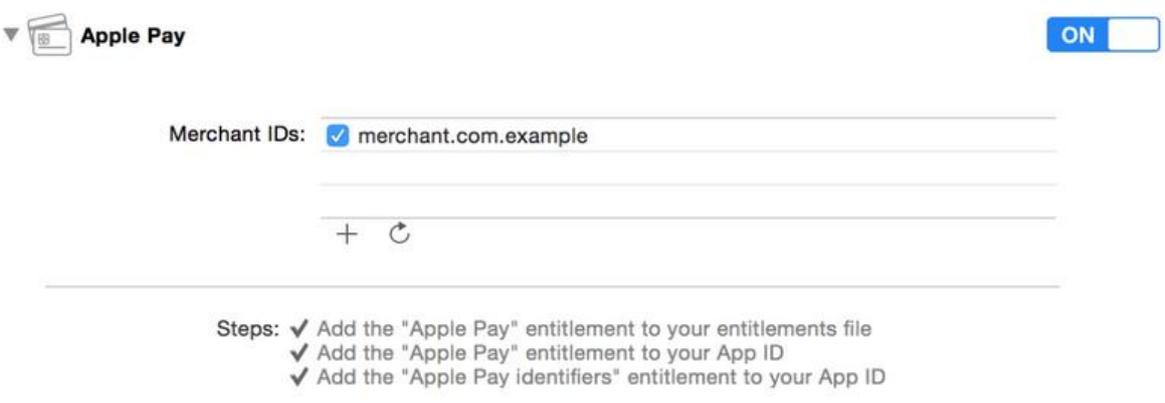
1. Obtaining Certificate Signing Request (CSR) from Computop
  - a. Contact [helpdesk@computop.com](mailto:helpdesk@computop.com) and ask for Apple Pay CSR. Provide us with information about your Apple Merchant ID.
  - b. CSR will be needed by Apple to generate actual certificate for signing Payment Token
2. Upload CSR at Apple
  - a. In Member Center, select *Certificates, Identifiers & Profiles*. Under *Identifiers*, select *Merchant IDs*. Select the merchant ID from the list, and click *Edit*.
  - b. Click *Choose File*, select the CSR received from Computop, and click *Generate*.
  - c. Download the certificate by clicking *Download*, and click *Done*.





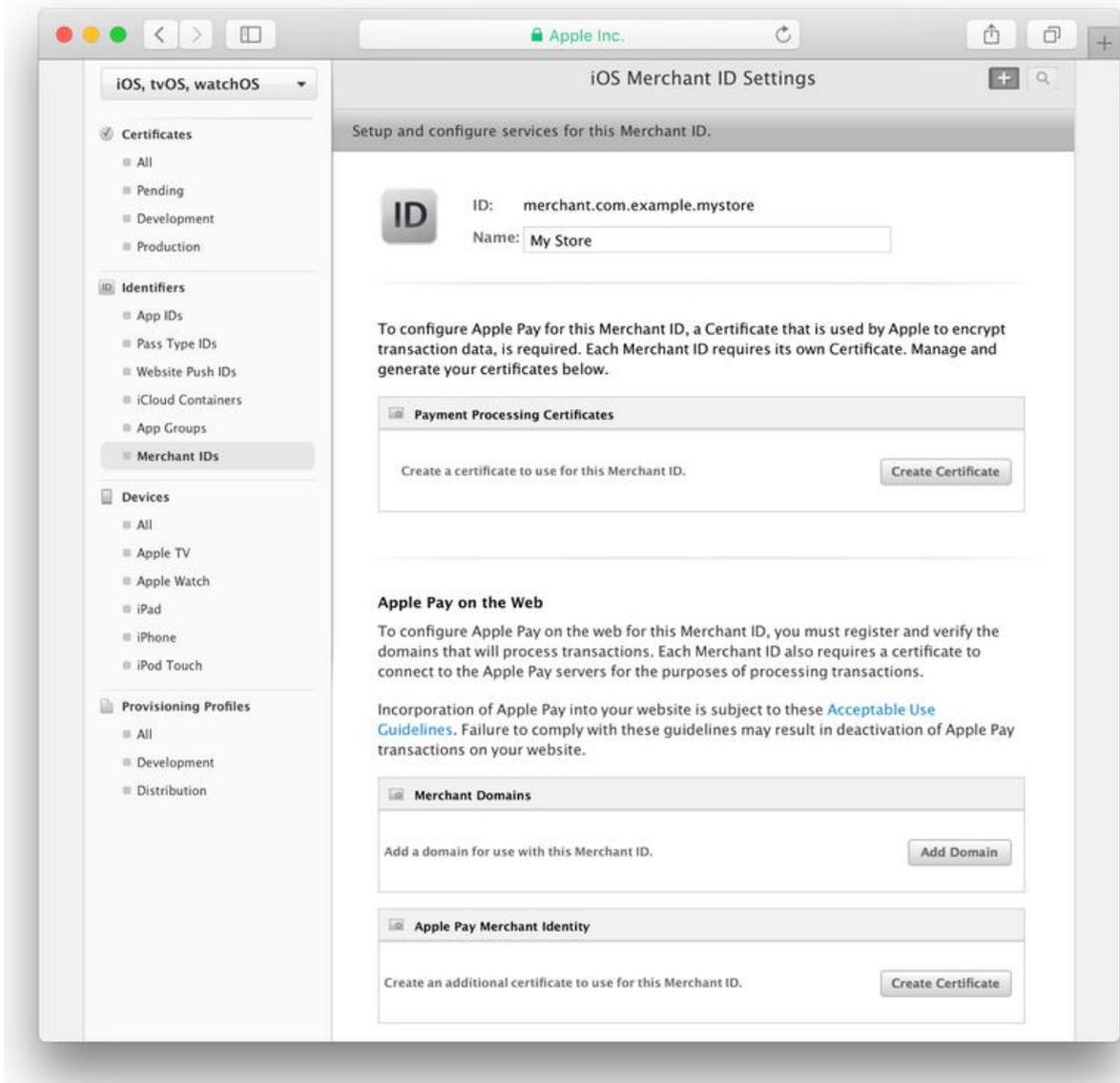
3. Return Certificate to Computop
  - a. Send the downloaded Certificate (CER file) together with your Apple Merchant ID to [helpdesk@computop.com](mailto:helpdesk@computop.com)

4. Enable Apple Pay for your app in **Xcode**
  1. Open *Capabilities* pane
  2. Select the switch in the *Apple Pay* row, and then select the merchant IDs you want the app to use.

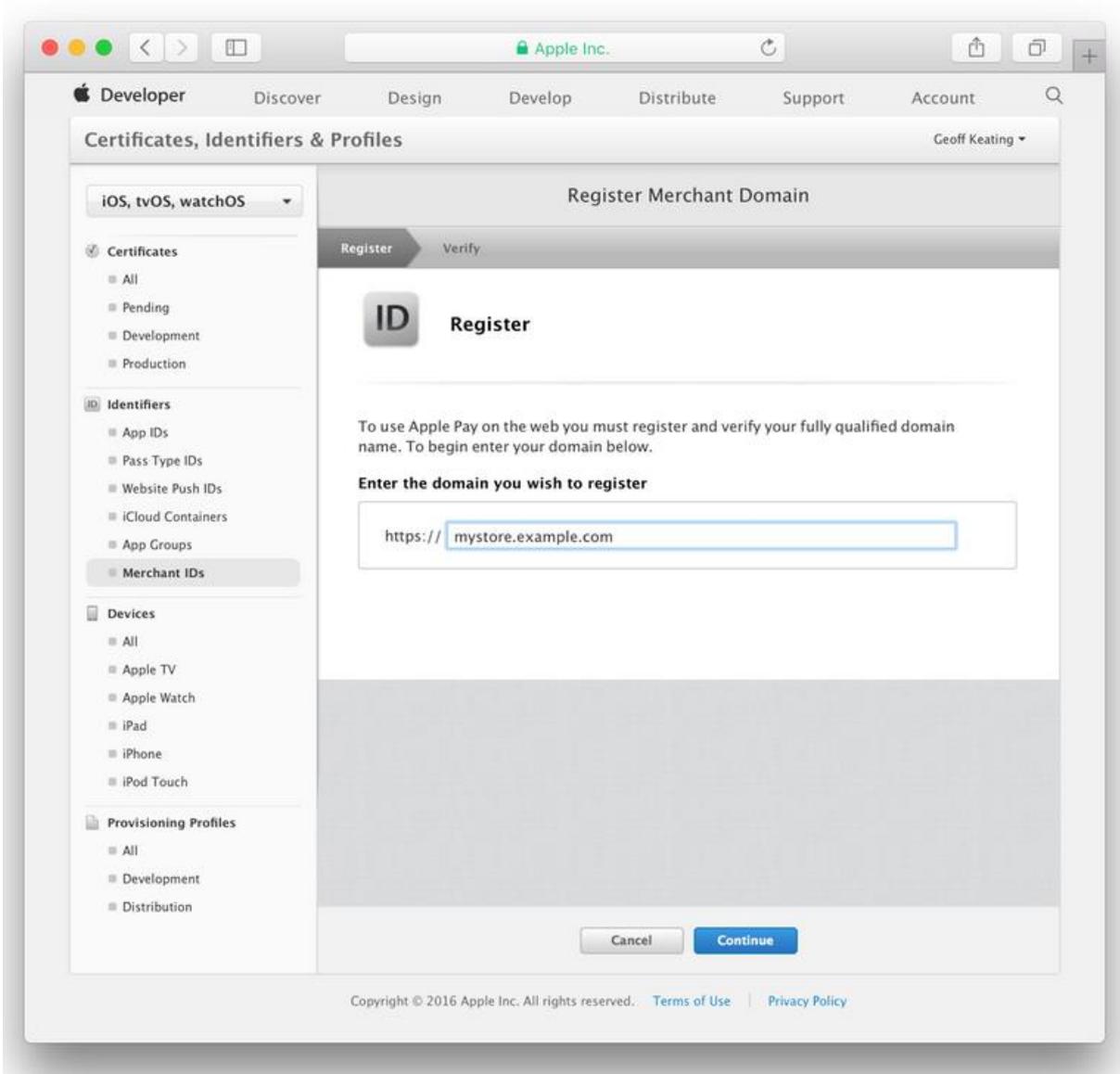


5. **(WEB ONLY)** Create **Merchant Identity Certificate**. A Transport Layer Security (TLS) certificate used to authenticate your merchant sessions with the Apple Pay servers.  
 To register and verify your domain, and create your Merchant Identity Certificate:

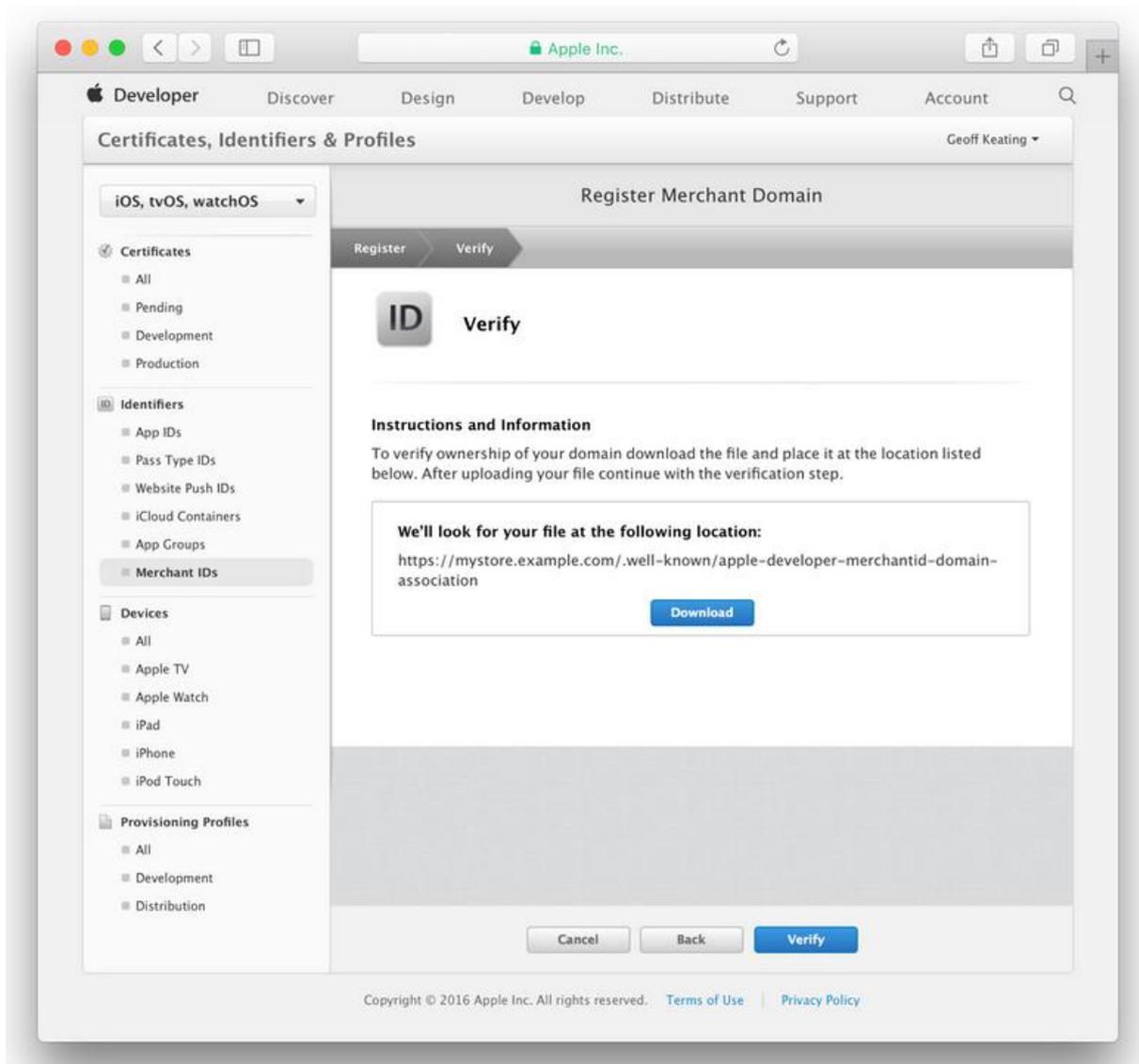
1. In Member Center, select *Certificates, Identifiers and Profiles*.
2. Under *Identifiers*, select *Merchant IDs*.
3. Select the merchant ID from the list, and click *Edit*.
4. In the *Apple Pay on the Web* section, click the *Add Domain* button.



5. Enter your fully qualified domain name, and click *Continue*.



6. The site creates a file and makes it available to download. Download this file and host it at the provided location.



7. As soon as the file is available on your server, click the Verify button. If the verification succeeds, the site returns to the iOS Merchant ID Settings page and shows a green *Verified* status label.
  8. In the *Apple Pay on the Web* section, click the *Create Certificate* button. Follow the instructions to create and download your Merchant Identity Certificate. Use this certificate when requesting a merchant session during payment validation. For more information, see [Merchant Validation](#) in [ApplePaySession](#).
6. Integrate Apple Pay at your application or web shop
- a. Follow guidelines: <https://developer.apple.com/apple-pay>
  - b. **Important:** When creating PKPaymentRequest, request.merchantCapabilities = PKMerchantCapability3DS. Computop only supports 3DS types.

# PAYGATE INTERFACE

---

## Definitions

Data formats:

Format	Description
a	alphabetical
as	alphabetical with special characters
n	numeric
an	alphanumeric
ans	alphanumeric with special characters
ns	numeric with special characters
bool	boolean expression (true or false)
3	fixed length with 3 digits/characters
..3	variable length with maximum 3 digits/characters
enum	enumeration of allowed values
dtm	ISODateTime (YYYY-MM-DDDhh:mm:ss)

Abbreviations:

Abbreviation	Description
CND	condition
M	mandatory
O	optional
C	conditional

## Calling the interface

Two transactions are created when making a credit card payment via Apple Pay. In the Apple Pay transaction the required credit card data are determined first and the actual credit card transaction is then carried out automatically. This takes place via a server-to-server connection and supports all usual credit card transaction options. The difference here, however, is that you don't transmit the credit card information, which you don't know. Instead, you transmit the PKPaymentToken generated by Apple Pay, which contains the required credit card data in encrypted form.

In order to make a credit card payment with Apple Pay, go to the following URL:

<https://www.computop-paygate.com/applepay.aspx>

**Notice:** For security reasons, Paygate rejects all payment requests with formatting errors. Therefore please use the correct data type for each parameter.

The following table describes the encrypted transfer parameters for Apple Pay. In addition the specific parameters are required for a credit card payment (without credit card data). More information about this you can find within „Manual\_Computop\_Paygate\_CardProcessing.pdf“:

Parameter	Format	CND	Description
<b>MerchantID</b>	ans..30	M	MerchantID, assigned by Computop
<b>TransID</b>	ans..64	M	TransactionID which should be unique for each payment
<b>RefNr</b>	ns..30	O	Unique reference number
<b>Amount</b>	n..12	M	Amount in the smallest currency unit (e.g. EUR Cent) Please contact the helpdesk, if you want to capture amounts < 100 (smallest currency unit).
<b>Currency</b>	a3	M	Currency, three digits DIN / ISO 4217
<b>MAC</b>	an64	M	Hash Message Authentication Code (HMAC) with SHA-256 algorithm
<b>UserData</b>	ans..1024	O	If specified at request, Paygate forwards the parameter with the payment result to the shop.
<b>URLNotify</b>	ans..256	O	Complete URL which Paygate calls up in order to notify the shop about the payment result. The URL may be called up only via port 443 It may not contain parameters: Use the <b>UserData</b> parameter instead.
<b>MerchantIDExt</b>	ans..128	C	Merchant Identifier of Public Key Mandatory if more then one Merchant Identifiers are stored with Computop.
<b>TokenExt</b>	ans..768	M	Apple Pay PKPaymentToken as JSON string in the Base64 format
<b>Channel</b>	a..10	O	Channel over which the order is processed. Allowed values are WEBSITE and MOBILE_APP

Parameters for Apple Pay

The following table describes the result parameters with which the Paygate responds to your system:

Parameter	Format	CND	Description
<b>MID</b>	ans..30	M	MerchantID, assigned by Computop
<b>PayID</b>	an32	M	ID assigned by Paygate for the payment, e.g. for referencing in batch files This ID refers to the credit card transaction unless an error occurs already at the Apple Pay transaction.
<b>XID</b>	an32	M	ID for all single transactions (authorisation, capture, credit note) for one payment assigned by Paygate
<b>TransID</b>	ans..64	M	Merchant's transaction number
<b>RefNr</b>	ns..30	O	Unique reference number
<b>MAC</b>	an64	M	Hash Message Authentication Code (HMAC) with SHA-256 algorithm
<b>UserData</b>	ans..1024	O	If specified at request, Paygate forwards the parameter with the payment result to the shop.
<b>Status</b>	a..50	M	OK or FAILED
<b>Description</b>	ans..1024	M	Further details in the event that payment is rejected. Please do not use the <b>Description</b> but the <b>Code</b> parameter for the transaction status analysis!
<b>Code</b>	n8	M	Error code according to Paygate Response Codes Excel file (part of the software client from <a href="http://www.computop.de/manual">www.computop.de/manual</a> )

Result parameters for Apple Pay

## Capture / Credit / Reversal

Captures, credits and reversals do not refer to the Apple Pay transaction but directly to the credit card transaction. More information about this you can find within „Manual\_Computop\_Paygate\_CardProcessing.pdf“:

## Batch processing via the interface

Captures, credits and reversals via batch do not refer to the Apple Pay transaction but directly to the credit card transaction. More information about this you can find within the document „Manual\_Computop\_Paygate\_CardProcessing.pdf“:

