

ИНСТРУКЦИЯ ПО ПОДКЛЮЧЕНИЮ APPLEPAY WEB

03.09.2018



Тинькофф

Содержание

1. ТРЕБОВАНИЯ К ИСПОЛЬЗОВАНИЮ APPLE PAY НА ВАШЕМ САЙТЕ	3
2. РЕГИСТРАЦИЯ MERCHANT ID	4
3. СОЗДАНИЕ СЕРТИФИКАТА (PAYMENT PROCESSING CERTIFICATE)	7
4. ВАЛИДАЦИЯ ДОМЕНА	14
5. СОЗДАНИЕ MERCHANT IDENTITY CERTIFICATE	17
6. НАСТРОЙКА ПЛАТЕЖЕЙ НА СТОРОНЕ ПРОДАВЦА	22

1. Требования к использованию Apple Pay на вашем сайте

1. Ваш сайт должен соответствовать рекомендациям Apple Pay.
Подробнее:
<https://developer.apple.com/apple-pay/acceptable-use-guidelines-for-websites/>
2. У вас должна быть учетная запись Apple Developer.
Подробнее:
https://developer.apple.com/documentation/apple_pay_on_the_web/configuring_your_environment
3. Все страницы, которые включают Apple Pay, должны обслуживаться через HTTPS.
Подробнее:
https://developer.apple.com/documentation/apple_pay_on_the_web/setting_up_your_server

2. Регистрация Merchant ID

Merchant ID – идентификатор продавца на стороне Apple.

Для регистрации нового Merchant ID необходимо:

1. Зайти в личный кабинет разработчика Apple Developer.
2. Перейти в раздел «Certificates» – «Merchant IDs».

Примечание. Если в верхней части страницы появляется баннер, в котором говорится, что необходимо принять соглашение – нужно нажать кнопку «Обзор соглашения» и следовать инструкциям.

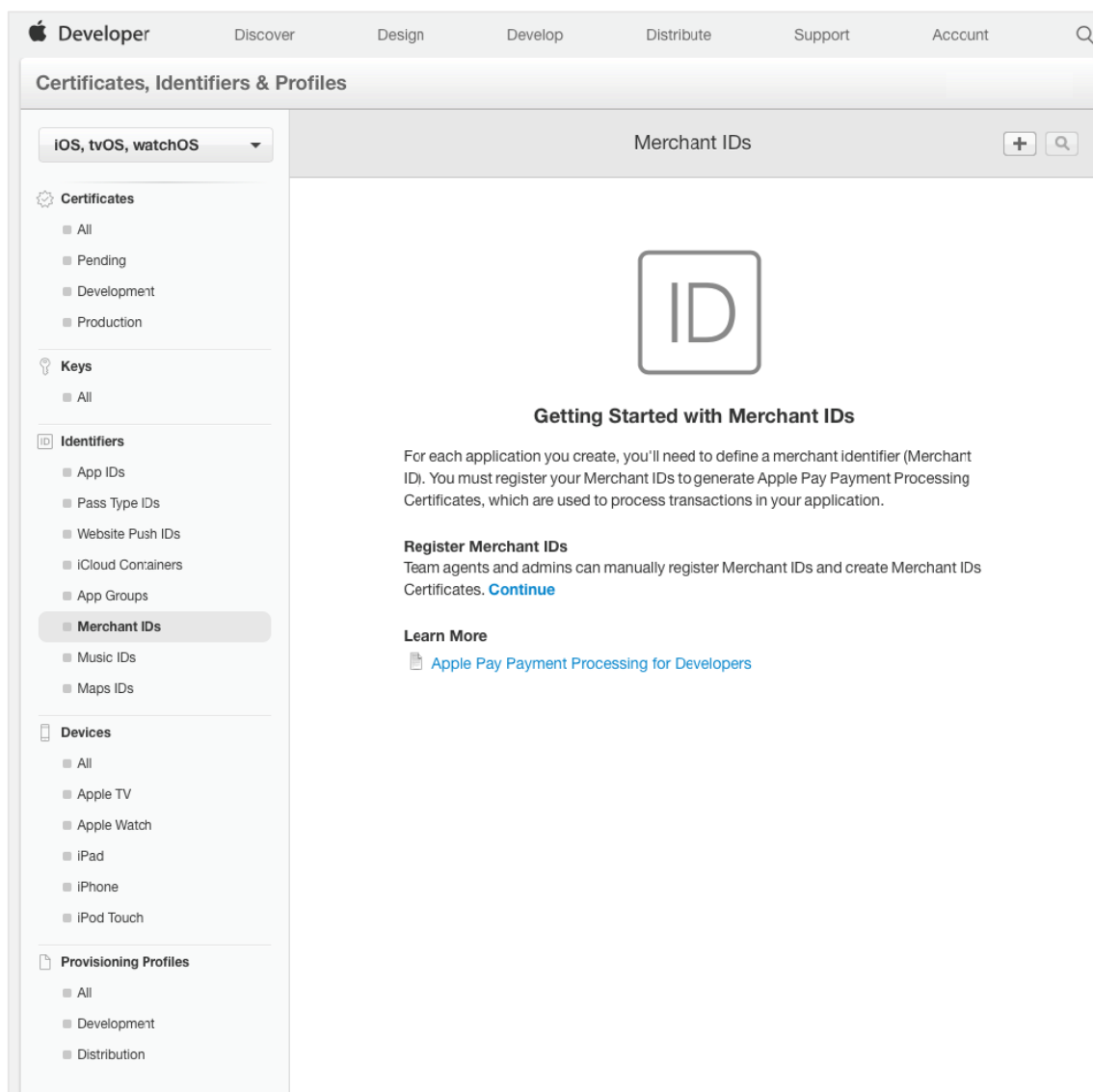


Рисунок 1. Merchant IDS

3. Нажать «+» в правой верхней части окна.
4. Ввести название терминала в поле Merchant ID Description.
5. В поле «ID» сформировать идентификатор для Merchant ID, начинающийся со слова «merchant». Последним доменом добавить значение из пункта 4.

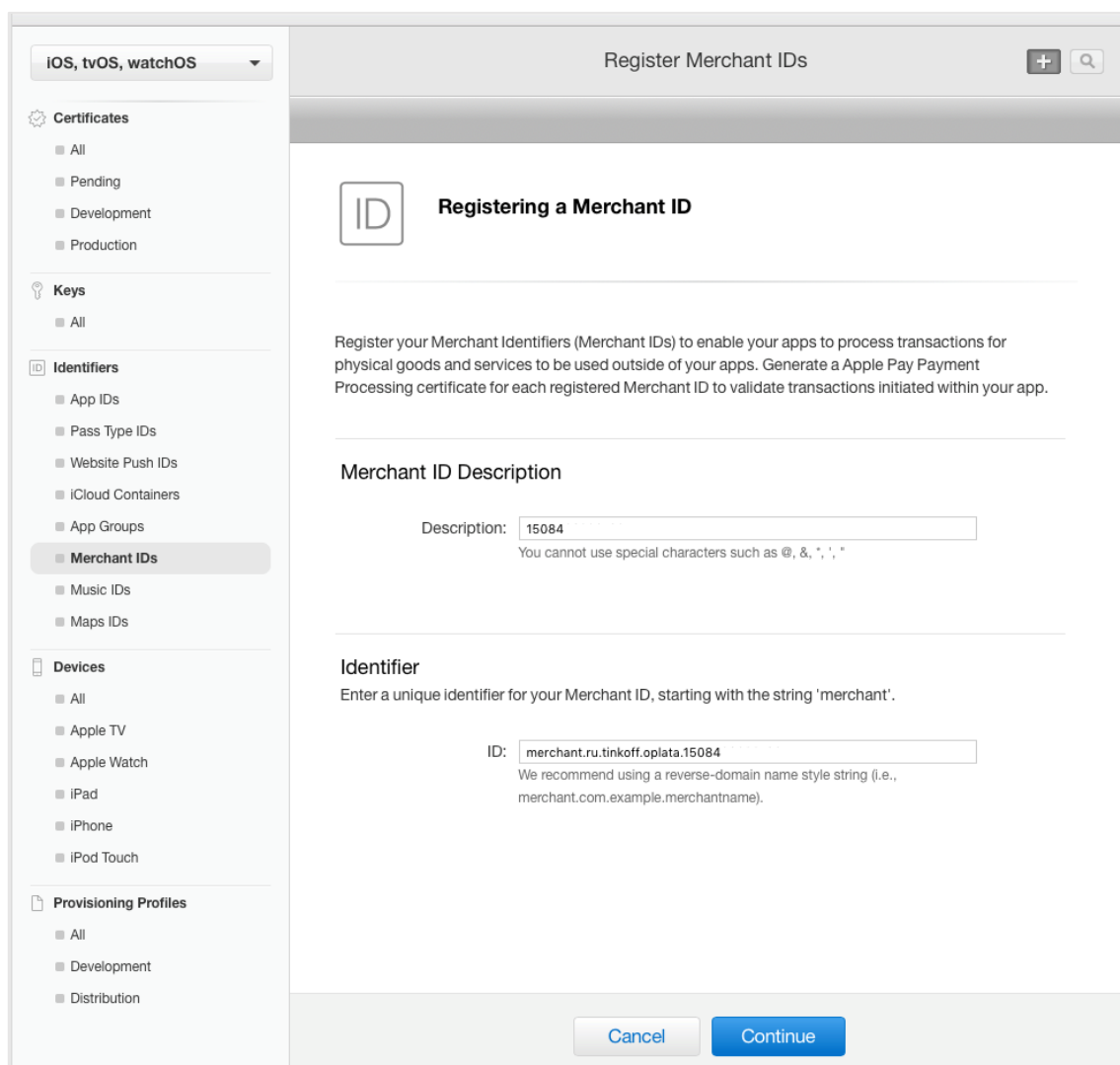


Рисунок 2. Создание идентификатора

6. Нажать кнопку **CONTINUE**.
7. На следующем экране нажать кнопку **REGISTER**.

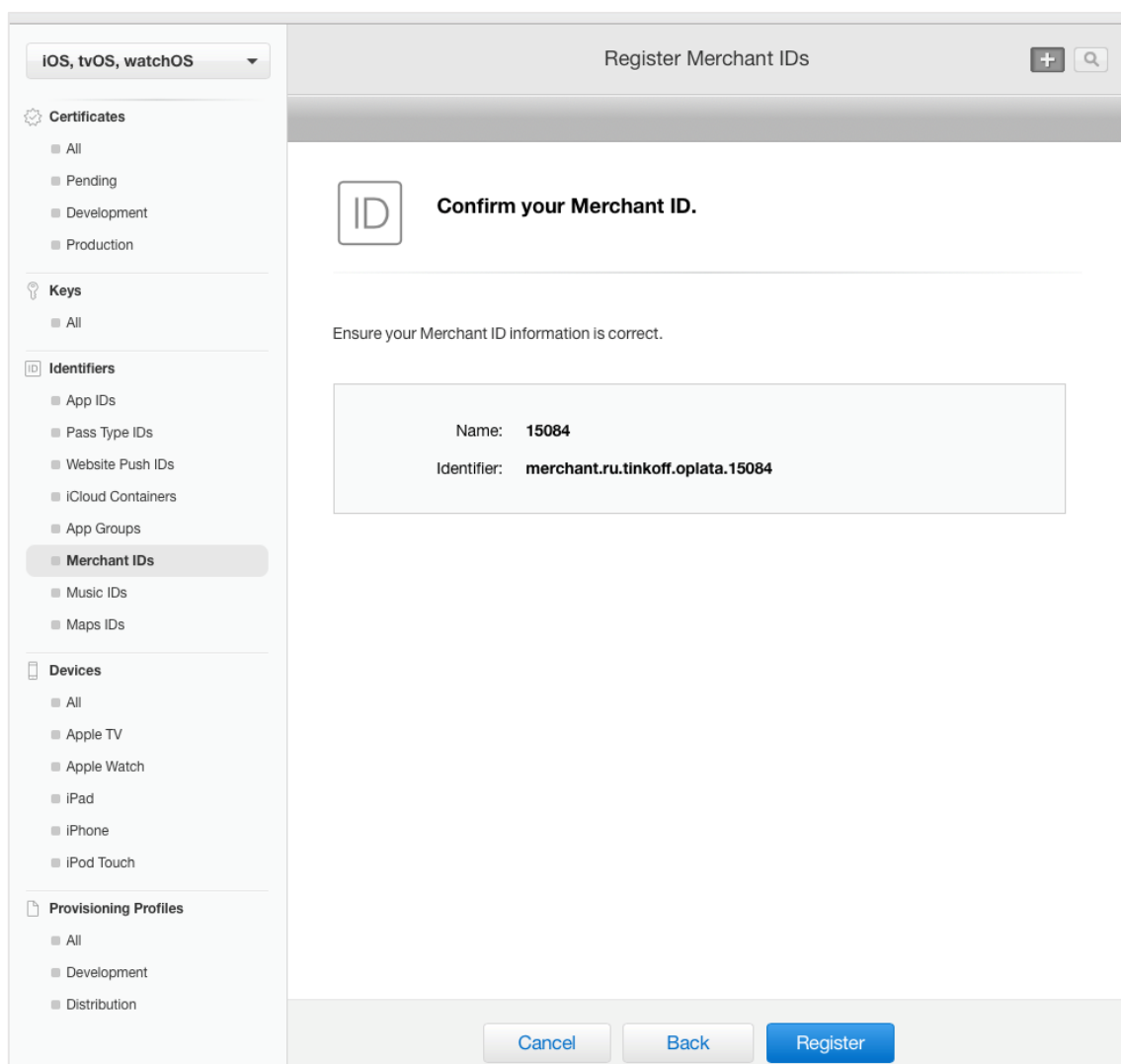


Рисунок 3. Завершение регистрации Merchant ID

8. Нажать кнопку **DONE**.

3. Создание сертификата (payment processing certificate)

Сертификат обработки платежей связан с идентификатором продавца и используется для шифрования информации об оплате. Срок действия сертификата обработки платежей составляет 25 месяцев.

Для создания сертификата необходимо:

1. Перейти в раздел «Merchant IDs».
2. В открывшемся списке справа выбрать Merchant ID и нажать кнопку [EDIT](#).

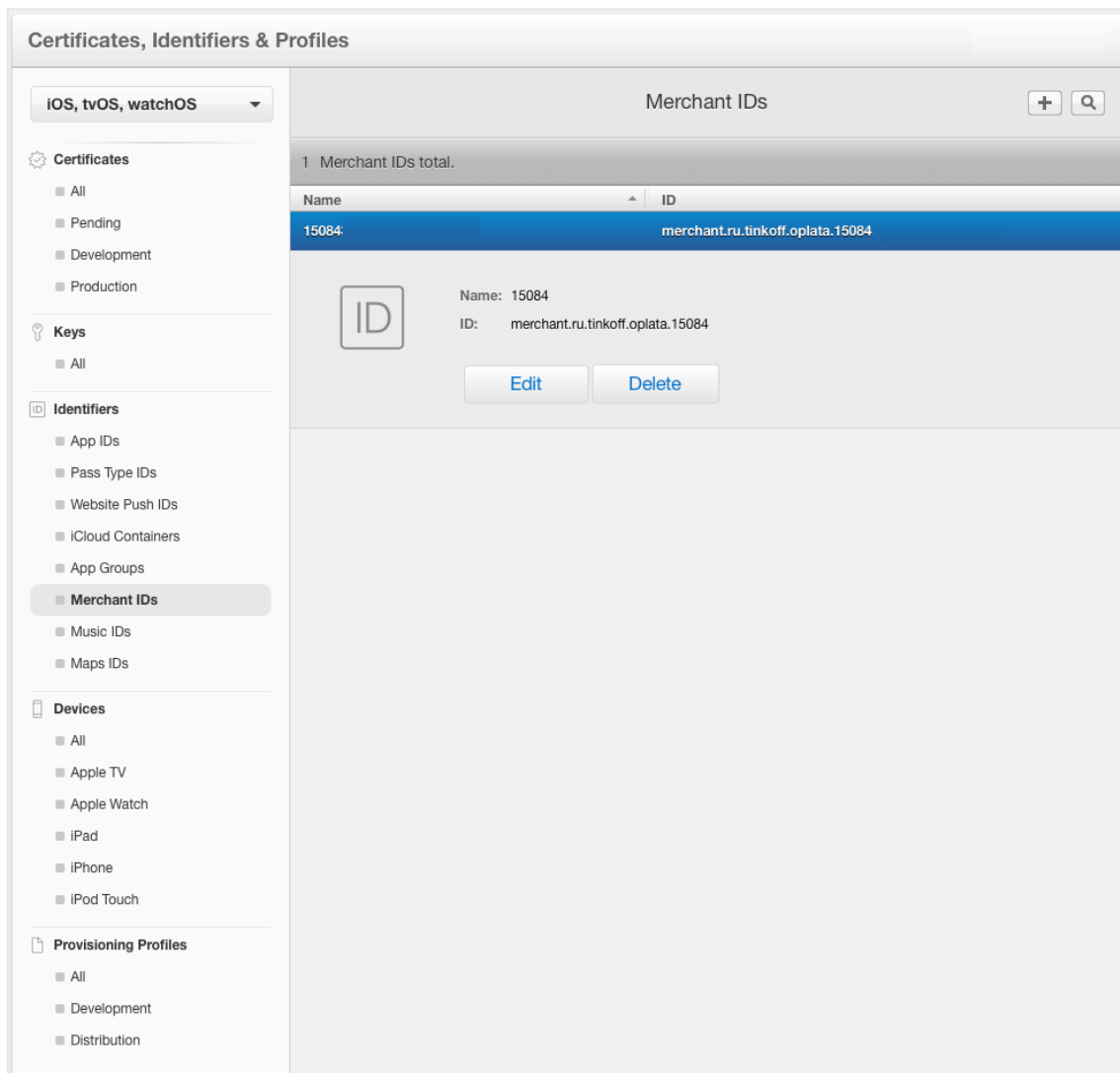


Рисунок 4. Создание сертификата. Шаг 1

3. Нажать кнопку [CREATE CERTIFICATE](#).

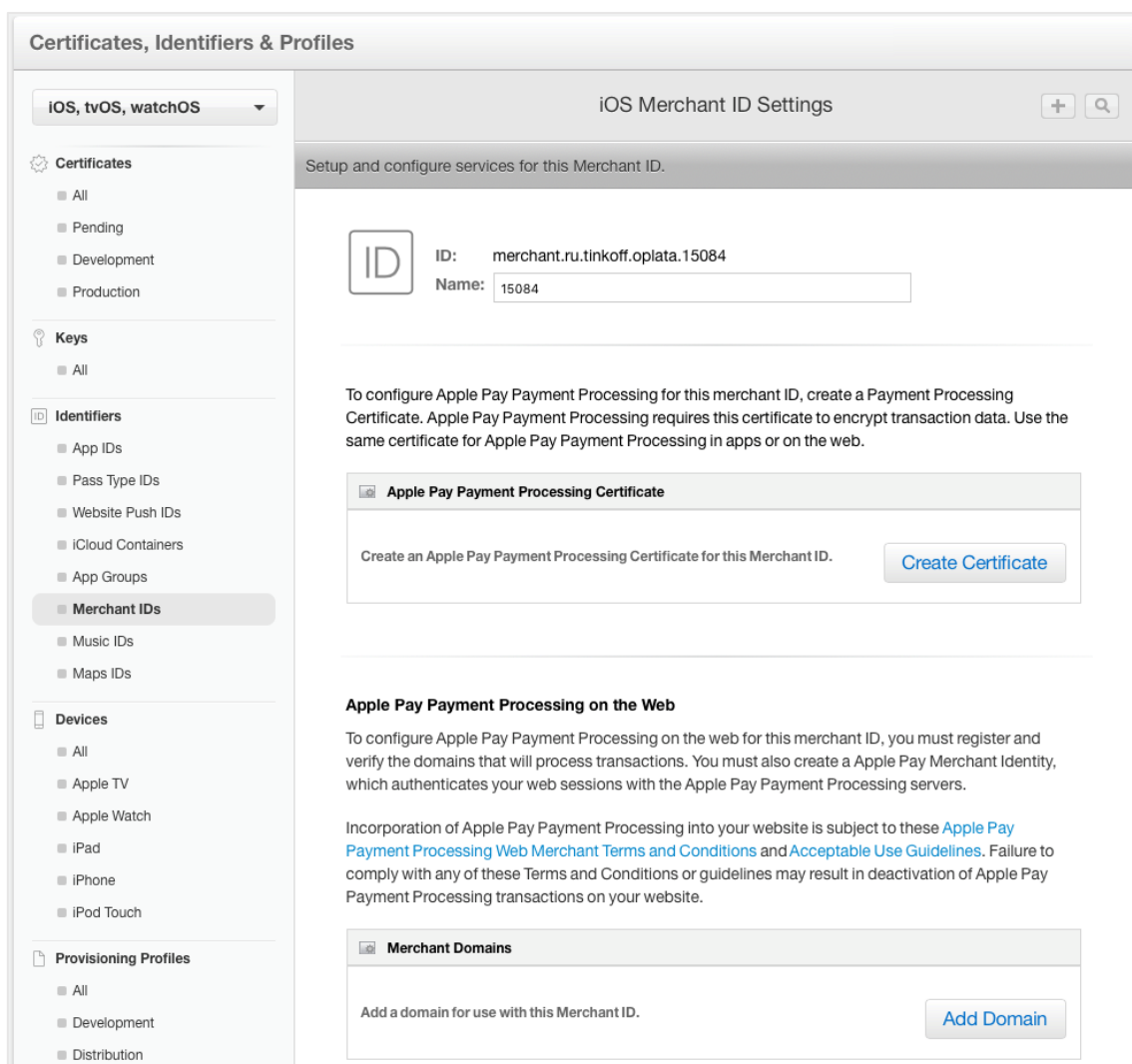
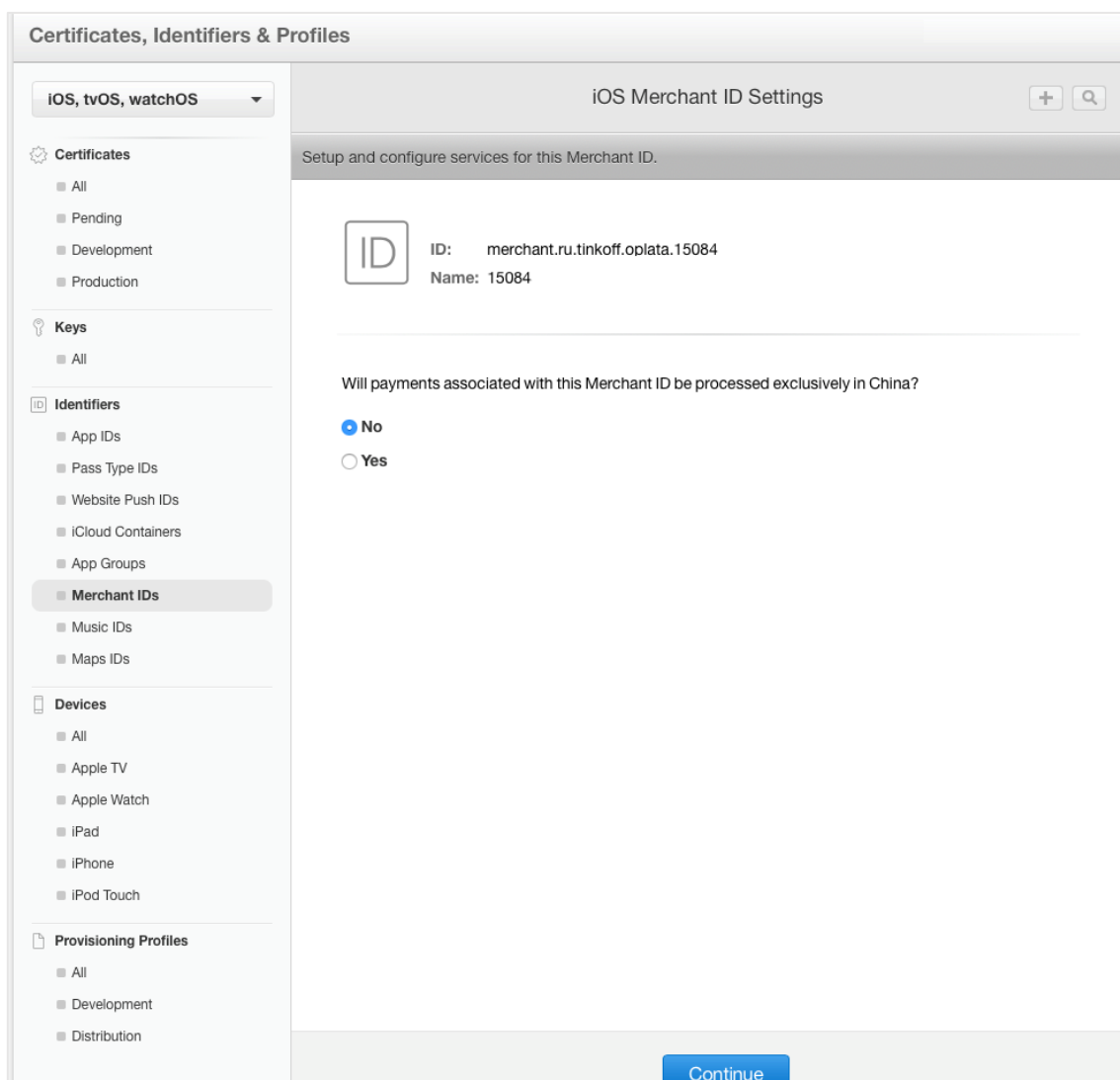
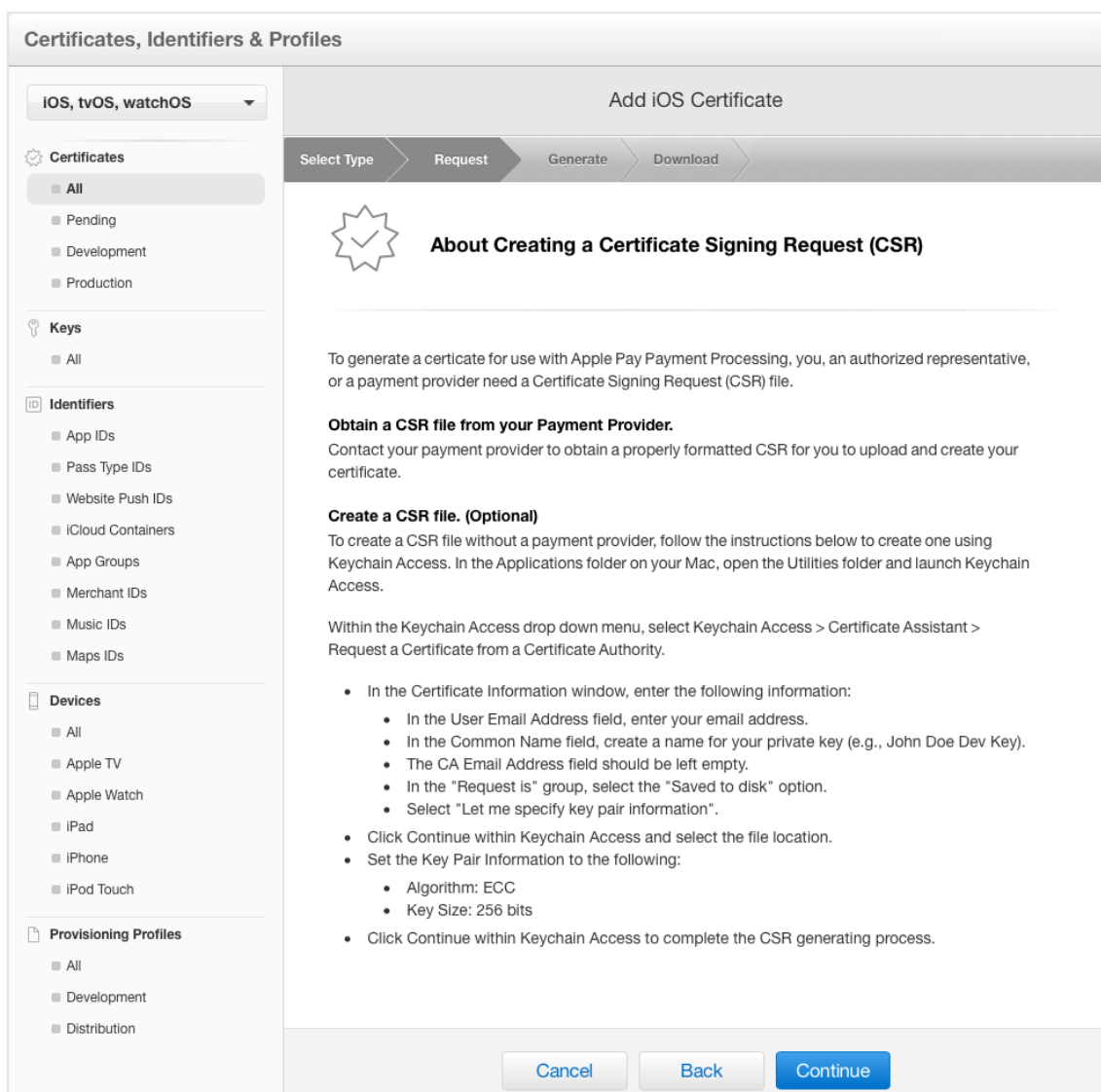


Рисунок 5. Создание сертификата. Шаг 2

4. Отметить чекбокс «No» в ответ на вопрос, будут ли платежи проходить исключительно в Китае.

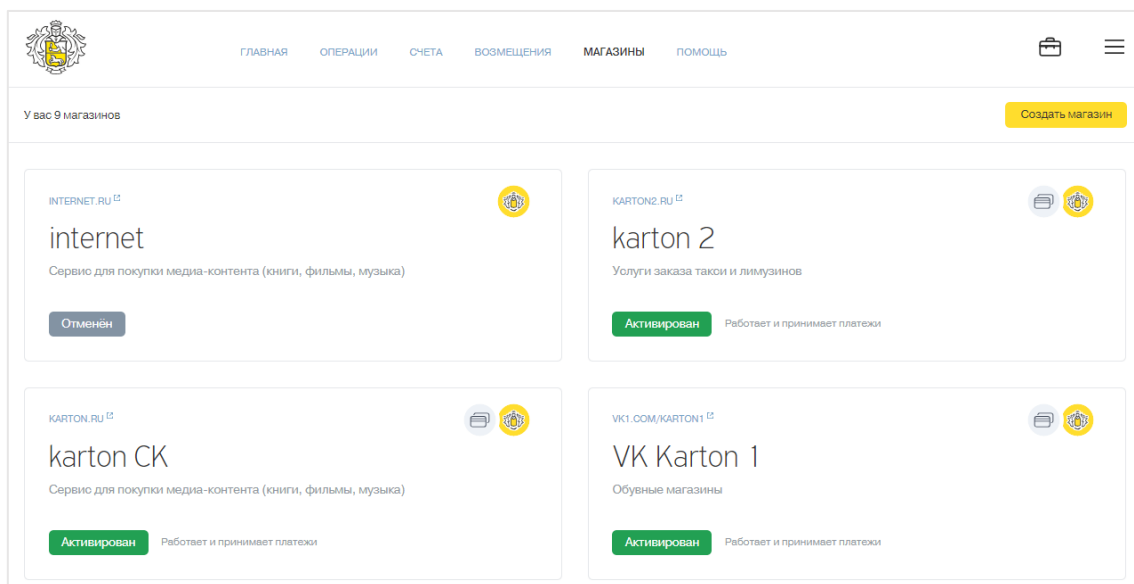


5. Нажать **CONTINUE**.

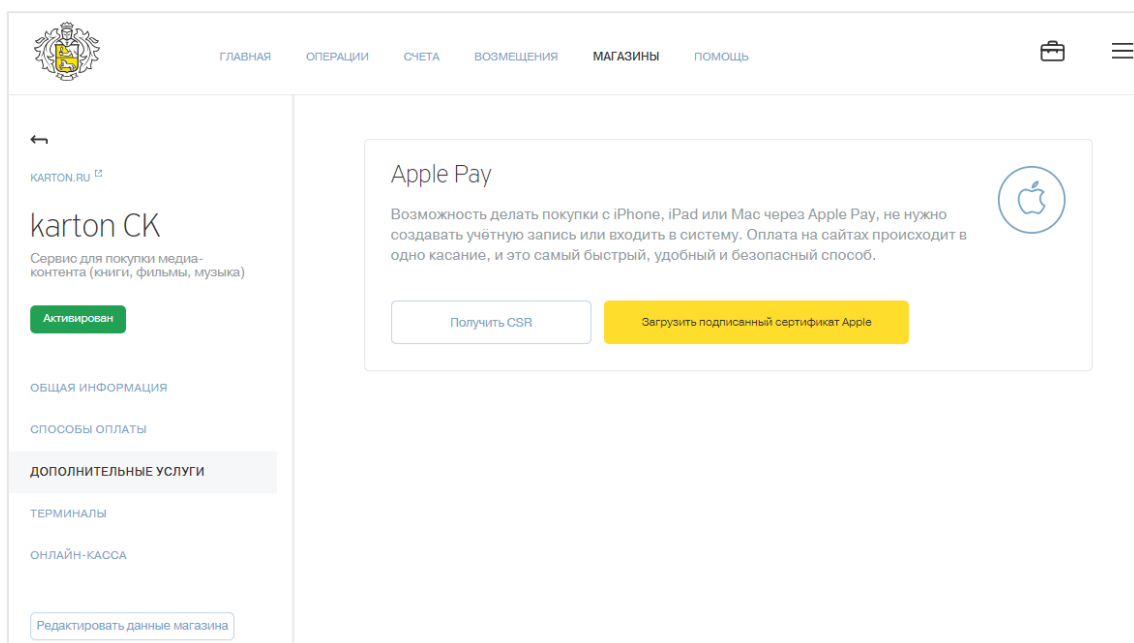


6. В новом окне открыть личный кабинет интернет-эквайринга.

7. Перейти в раздел «Магазины»



8. Выбрать магазин для подключения Apple Pay Web и перейти в раздел «Дополнительные услуги».
9. Нажать кнопку **ПОДКЛЮЧИТЬ**.
10. Нажать кнопку **ПОЛУЧИТЬ CSR**.



11. Сохранить полученный файл (cert.csr)
12. Вернуться в Apple Pay Developer, нажать кнопку **CHOOSE FILE** и выбрать сохранённый сертификат.

IOS, tvOS, watchOS

Certificates

All

Pending

Development

Production

Keys

All

Identifiers

App IDs

Pass Type IDs

Website Push IDs

iCloud Containers

App Groups

Merchant IDs

Music IDs

Maps IDs

Devices

All

Apple TV

Apple Watch

iPad

iPhone

iPod Touch

Provisioning Profiles

All

Development

Distribution


Add iOS Certificate

Select Type

Request

Generate

Download



Generate your certificate.

When your CSR file is created, a public and private key pair is automatically generated. Your private key is stored on your computer. On a Mac, it is stored in the login Keychain by default and can be viewed in the Keychain Access app under the "Keys" category. Your requested certificate is the public half of your key pair.

Upload CSR file.

Select .certSigningRequest file saved on your Mac.

Choose File...

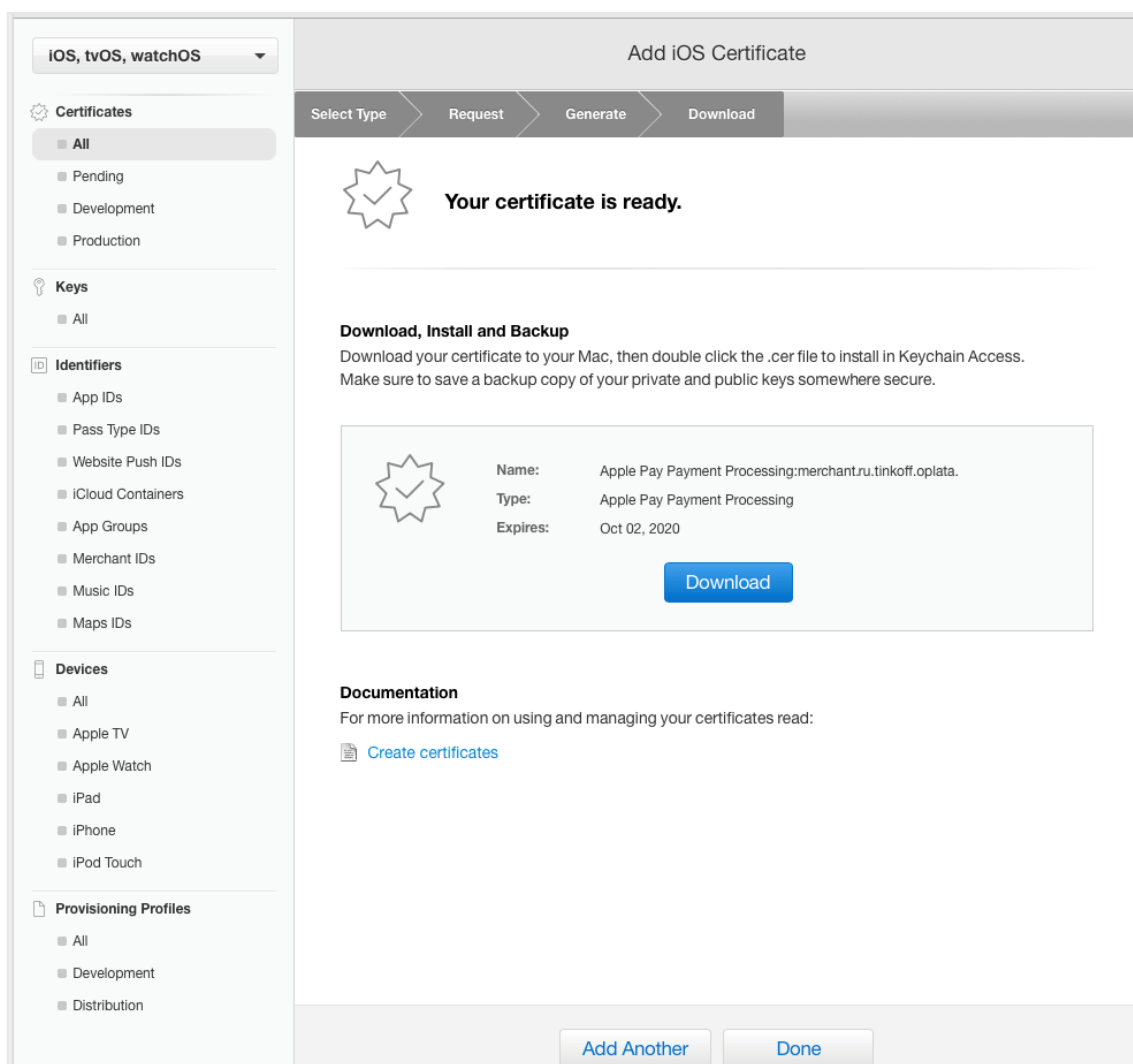
cert (2).csr

Cancel

Back

Continue

13. Нажать кнопку **CONTINUE**.



14. Нажать кнопку **DOWNLOAD** сохранить полученный файл.
15. Вернуться на страницу интернет-эквайринга, нажать кнопку **ЗАГРУЗИТЬ ПОДПИСАННЫЙ СЕРТИФИКАТ APPLE**.
16. В открывшемся диалоге выбрать сохранённый сертификат от Apple и нажать кнопку **ОТПРАВИТЬ**.
17. Вернуться в окно Apple Pay Developer, нажать кнопку **DONE**.

4. Валидация домена

Для валидации домена необходимо:

1. Перейти в раздел «Merchant IDs».
2. В открывшемся списке справа выбрать Merchant ID и нажать кнопку **EDIT**.
3. Нажать кнопку **ADD DOMAIN**.

- Pass Type IDs
- Website Push IDs
- iCloud Containers
- App Groups
- Merchant IDs**
- Music IDs
- Maps IDs

Devices

- All
- Apple TV
- Apple Watch
- iPad
- iPhone
- iPod Touch

Provisioning Profiles

- All
- Development
- Distribution

Apple Pay Payment Processing Certificate

Name: merchant.ru.tinkoff.oplata.15084
Type: Apple Pay Payment Processing
Expires: Oct 2, 2020 (Active Certificate)

Revoke
Download

Create an additional certificate to use for this Merchant ID.

Create Certificate

Apple Pay Payment Processing on the Web

To configure Apple Pay Payment Processing on the web for this merchant ID, you must register and verify the domains that will process transactions. You must also create a Apple Pay Merchant Identity, which authenticates your web sessions with the Apple Pay Payment Processing servers.

Incorporation of Apple Pay Payment Processing into your website is subject to these [Apple Pay Payment Processing Web Merchant Terms and Conditions](#) and [Acceptable Use Guidelines](#). Failure to comply with any of these Terms and Conditions or guidelines may result in deactivation of Apple Pay Payment Processing transactions on your website.

Merchant Domains

Domain: oplata.tinkoff.ru
Status: Pending
Verification Expires:

Remove
Verify

Add a domain for use with this Merchant ID.

Add Domain

Apple Pay Merchant Identity Certificate

Create an Apple Pay Merchant Identity Certificate for this Merchant ID.

Create Certificate

Done

4. Ввести домен для валидации.

Certificates, Identifiers & Profiles

iOS, tvOS, watchOS

Certificates

All

Pending

Development

Production

Keys

All

Identifiers

App IDs

Pass Type IDs

Website Push IDs

iCloud Containers

App Groups

Merchant IDs

Music IDs

Maps IDs

Devices

All

Apple TV

Apple Watch

iPad

iPhone

iPod Touch

Provisioning Profiles

All

Development

Distribution

Register Merchant Domain

Register

Verify

ID

Register

To use Apple Pay Payment Processing on the web you must register and verify your fully qualified domain name. To begin enter your domain below.

Enter the domain you wish to register

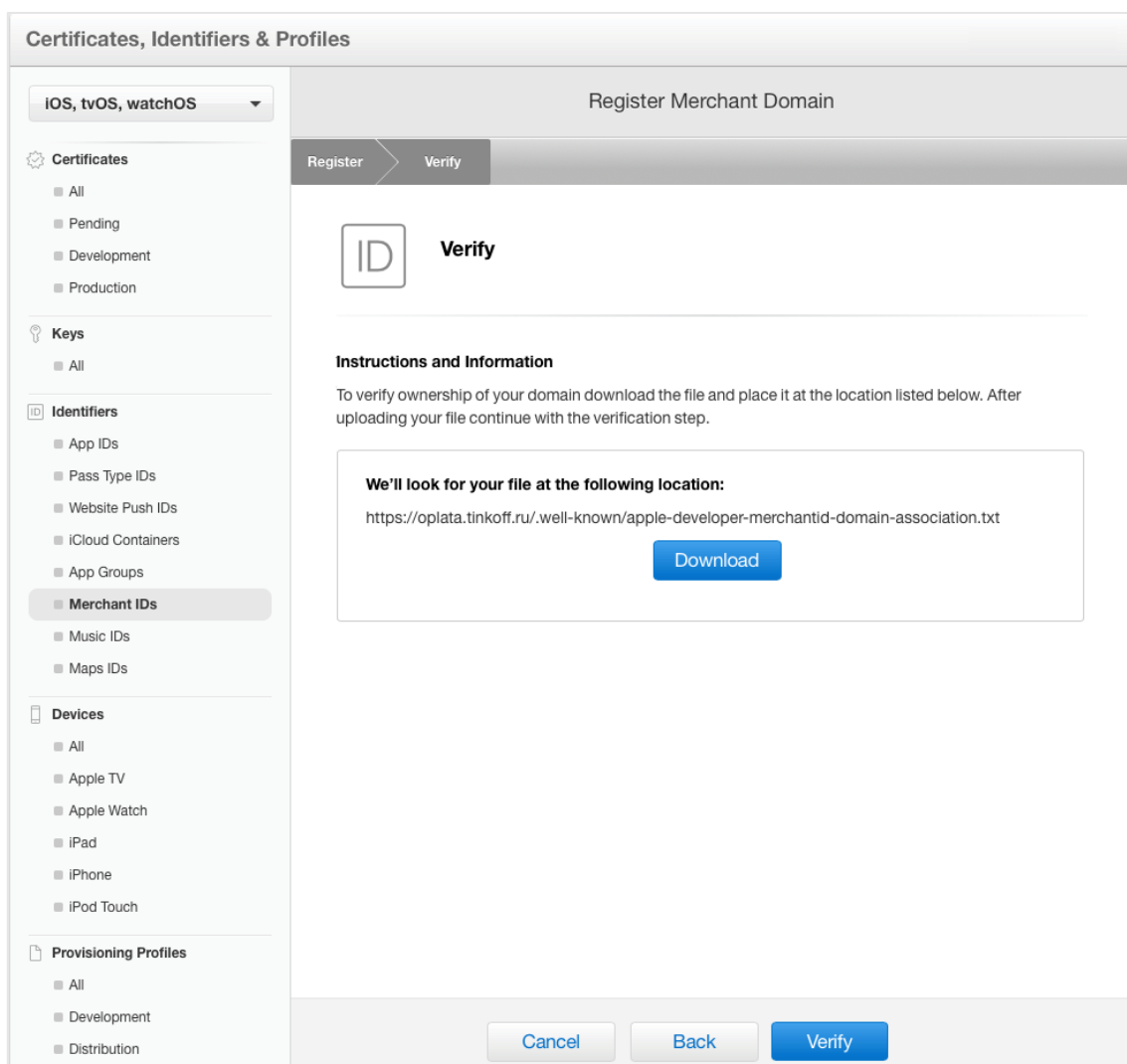
https://

Cancel

Continue

5. Нажать **CONTINUE**.

15



6. Загрузить файл и разместить его по указанному адресу.

7. Нажать кнопку **VERIFY**.

Примечание. Процесс верификации может занять около суток.

5. Создание Merchant Identity Certificate

Merchant Identity Certificate – сертификат протокола защиты транспортного уровня (TLS), связанный с идентификатором продавца и используемый для аутентификации с серверами Apple Pay.

1. Перейти в раздел «Merchant IDs».
2. В открывшемся списке справа выбрать Merchant ID и нажать кнопку **EDIT**.
3. Нажать кнопку **CREATE CERTIFICATE**.

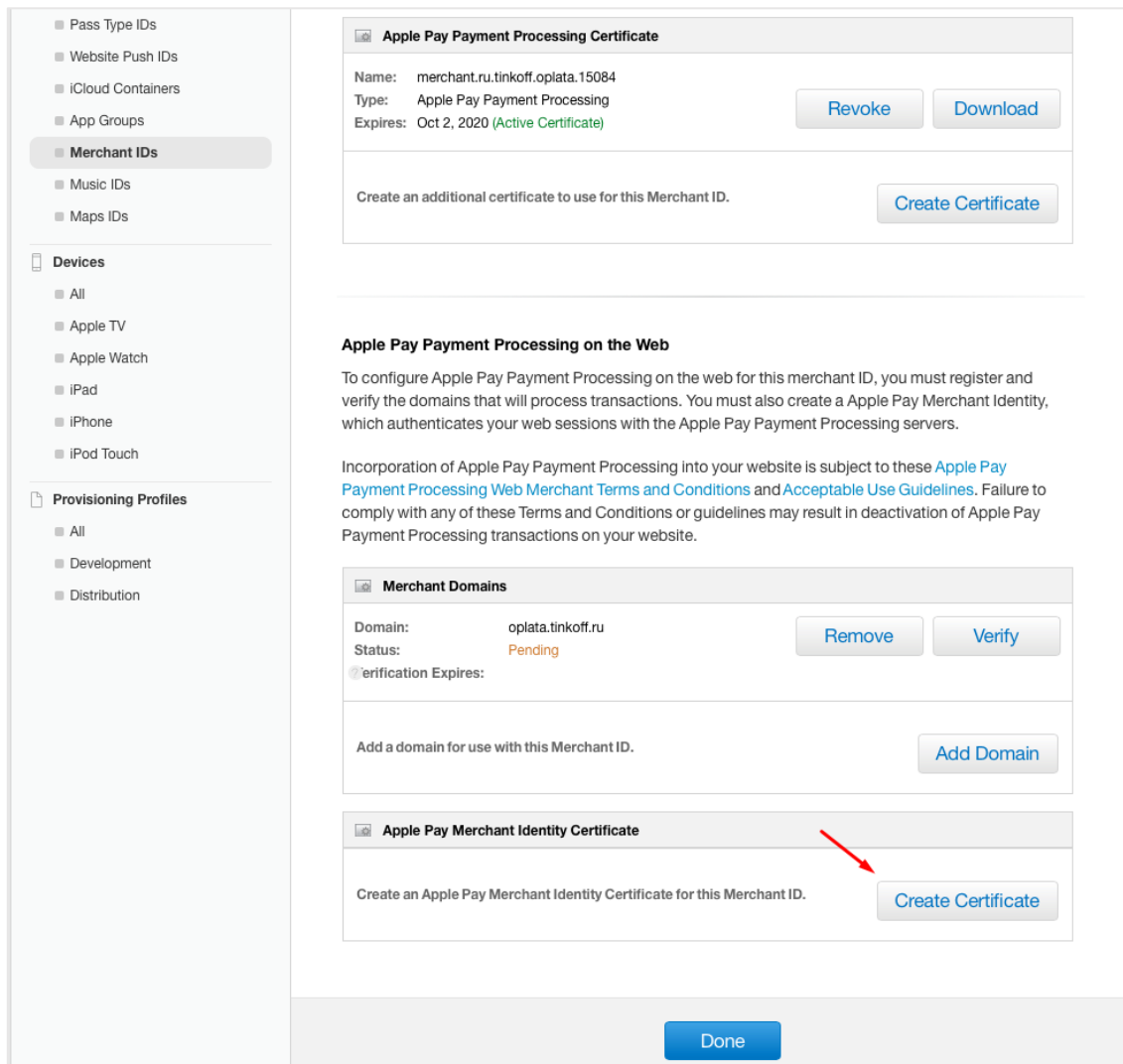
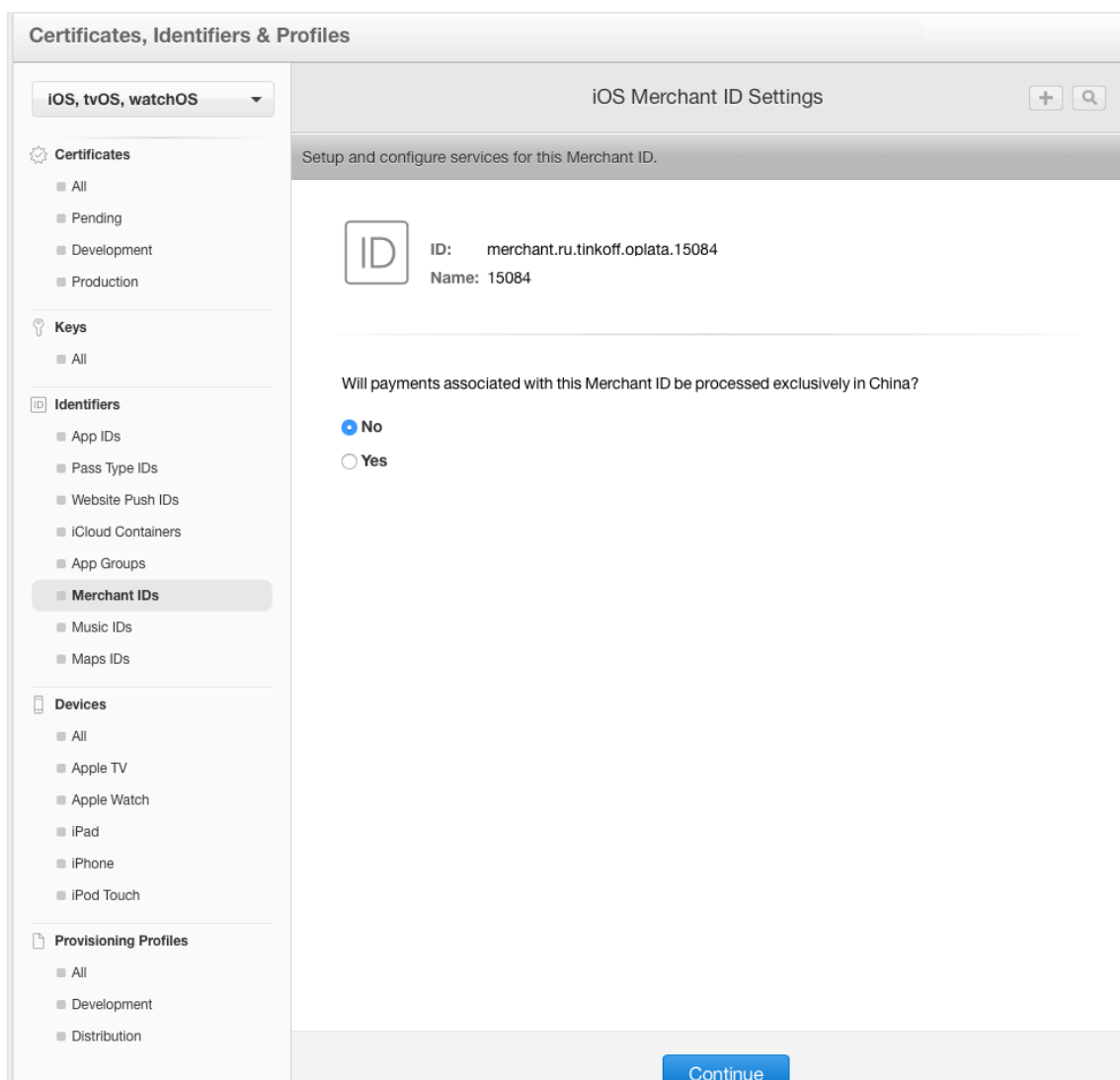
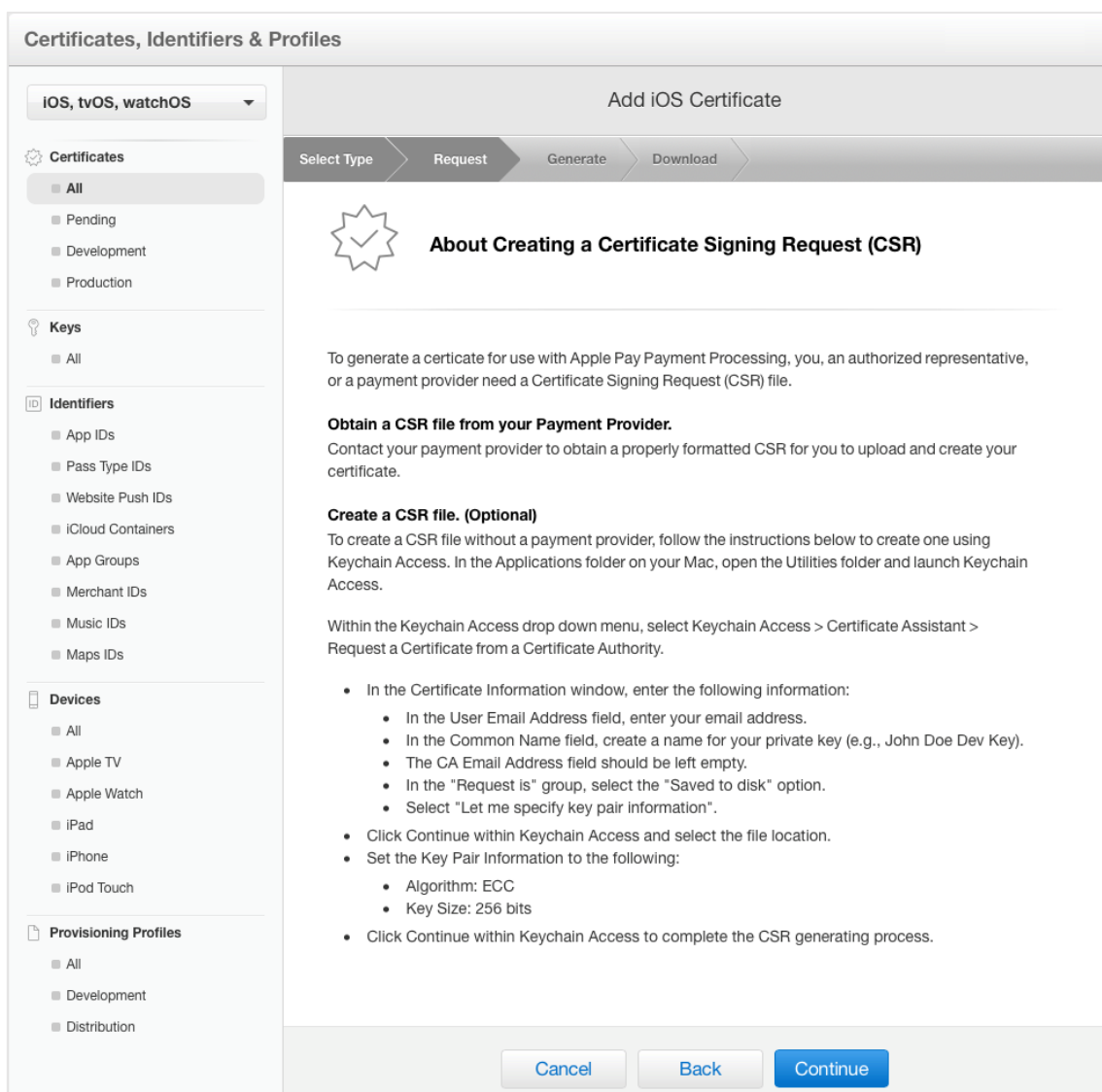


Рисунок 6. Создание сертификата. Шаг 1

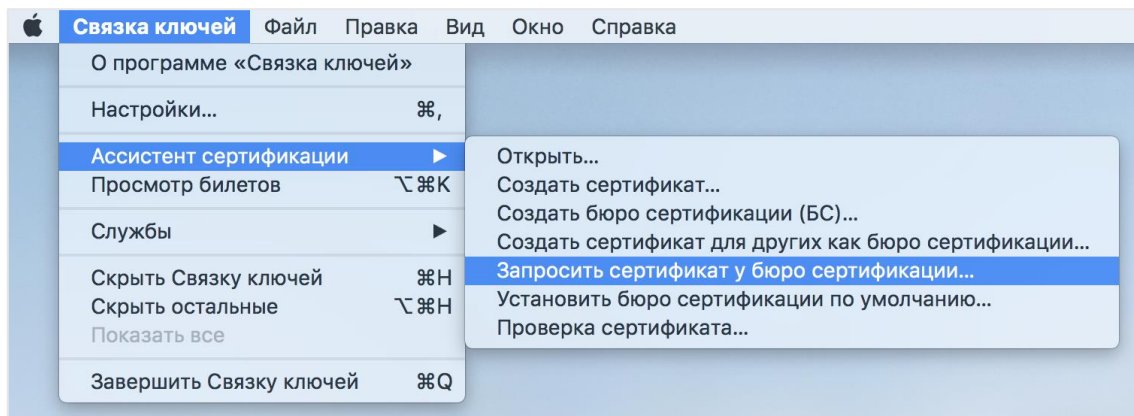
4. Отметить чекбокс «No» в ответ на вопрос, будут ли платежи проходить исключительно в Китае и нажать **CONTINUE**.



5. Нажать **CONTINUE**.

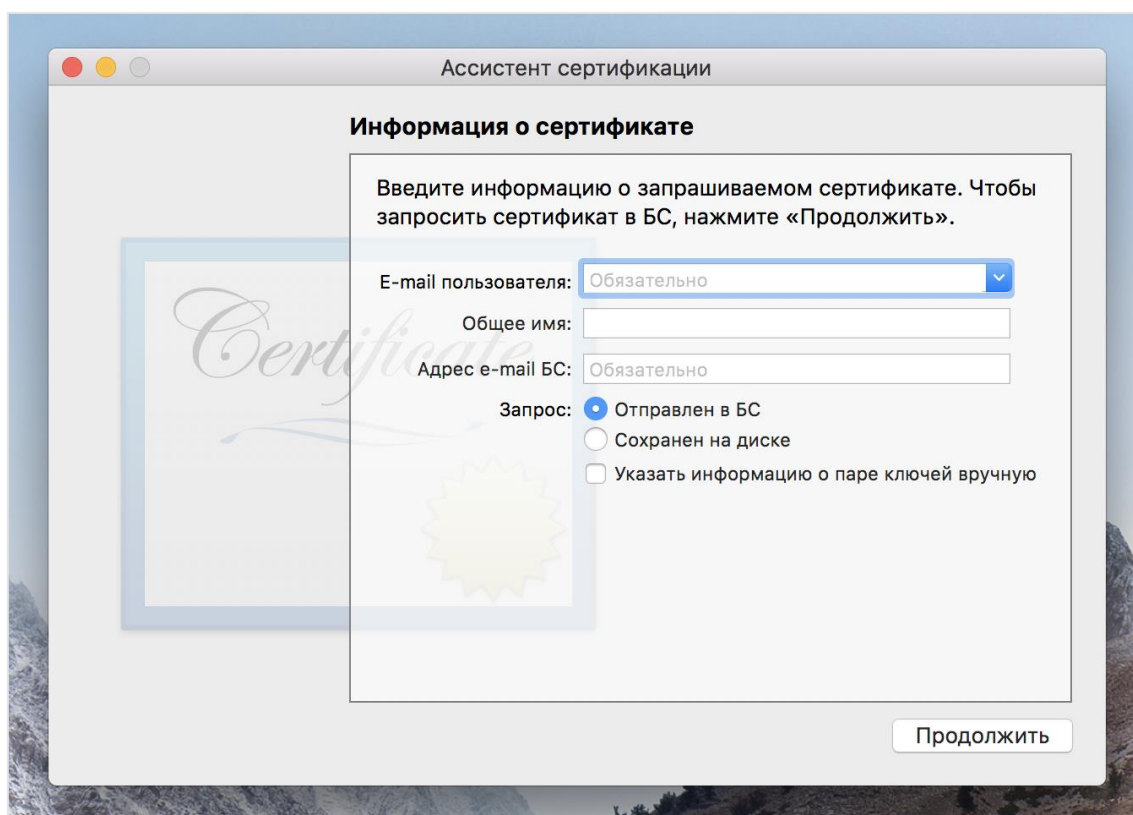


6. Запустить «Связка ключей», расположенную в /Программы/Утилиты.
7. Выбрать «Запросить сертификат у бюро сертификации» в разделе «Ассистент сертификации».



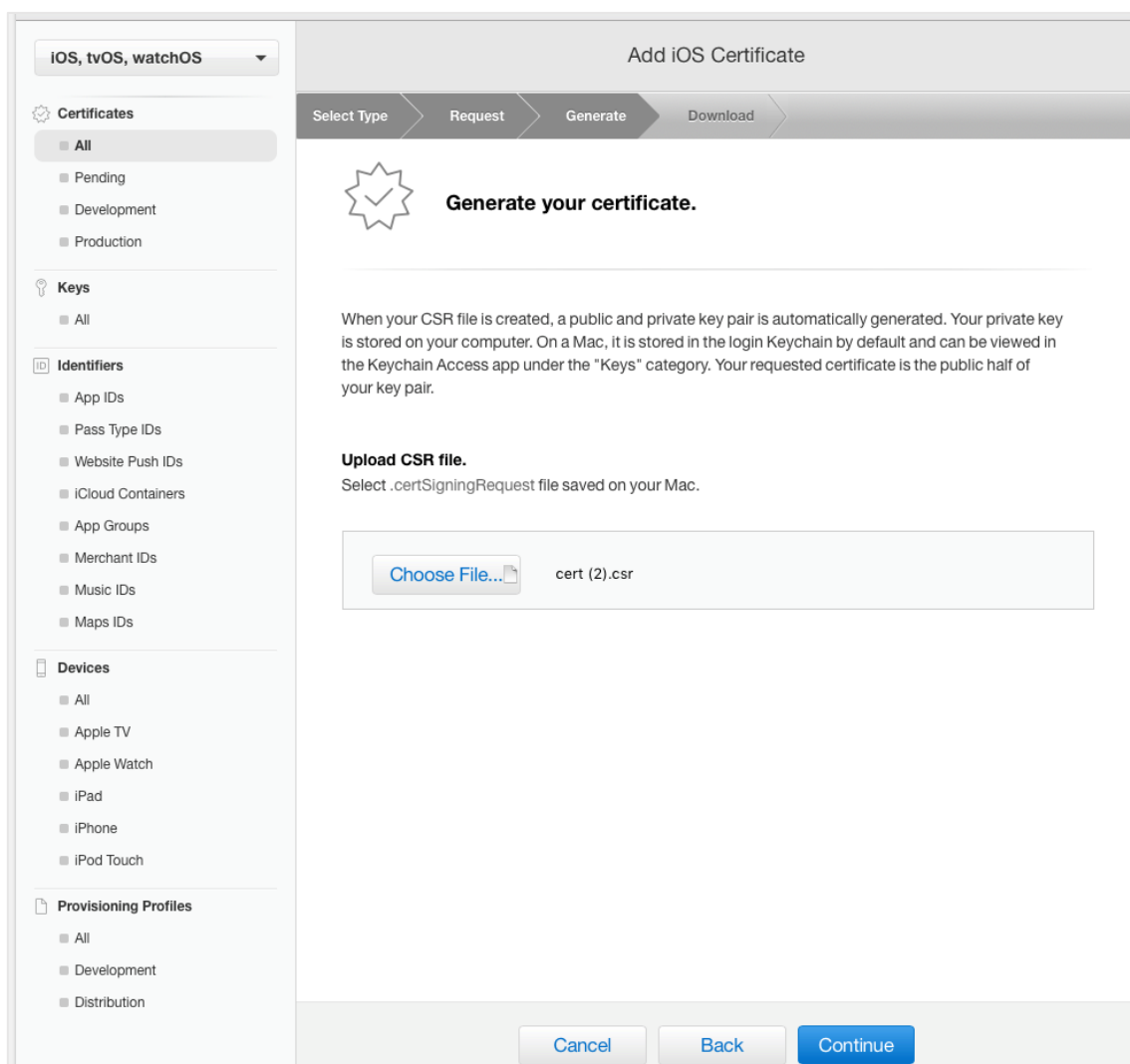
8. Заполнить поля «Е-mail пользователя» и «Общее имя» (Будет использоваться как название ключа) в появившемся диалоге и нажать кнопку **ПРОДОЛЖИТЬ**.

Внимание! Поле «Адрес e-mail адрес БС» должно остаться пустым!



9. Сохранить файл.


10. Нажимаем кнопку «[CHOOSE FILE...](#)» в личном кабинете и загрузить сохраненный файл.



iOS, tvOS, watchOS

Add iOS Certificate

Select Type | Request | **Generate** | Download

 **Generate your certificate.**

When your CSR file is created, a public and private key pair is automatically generated. Your private key is stored on your computer. On a Mac, it is stored in the login Keychain by default and can be viewed in the Keychain Access app under the "Keys" category. Your requested certificate is the public half of your key pair.

Upload CSR file.
Select .certSigningRequest file saved on your Mac.

[Choose File...](#) cert (2).csr

Cancel Back **Continue**

1 1. Нажать кнопку **CONTINUE**.

1 2. Нажать кнопку **DONE**. Загружать файл не требуется, сертификат сохранится у Apple.

6. Настройка платежей на стороне продавца

Примечание. Для упрощения кода примеров использована библиотека jQuery (\$).

Для подключения необходимо:

1. При загрузке страницы проверить поддержку устройством Apple Pay.

Пример:

```
if (window.ApplePaySession) {  
    console.log('Устройство поддерживает ApplePay (Web)');  
} else {  
    console.log('Устройство не поддерживает ApplePay (Web)');  
}
```

2. Убедиться, что устройство пользователя (покупателя) может проводить платежи, используя ApplePay (Web).

Для этого существуют два способа: `canMakePayments` и `canMakePaymentsWithActiveCard`.

- a) Метод `canMakePayments` – проверяет только саму возможность создания платежей и не осуществляет больше никаких обработок.

Пример:

```
if (window.ApplePaySession) {  
    console.log('Устройство поддерживает ApplePay (Web)');  
} else {  
    console.log('Устройство не поддерживает ApplePay (Web)');  
}
```

Подробнее:

https://developer.apple.com/documentation/apple_pay_on_the_web/applepay/session/1778027-canmakepayments)

- b) Метод `canMakePaymentsWithActiveCard` - проверяет не только возможность создавать платежи, но и запрашивает у устройства пользователя через сервера Apple наличие активных (сохраненных) карты на устройстве.

Пример:

```
if (window.ApplePaySession) {  
    console.log('Устройство поддерживает ApplePay (Web)');  
} else {  
    console.log('Устройство не поддерживает ApplePay (Web)');  
}
```

Подробнее:

https://developer.apple.com/documentation/apple_pay_on_the_web/applepay/session/1778000-canmakepaymentswithactivecard)

3. Если проверка устройства завершилась успехом, необходимо отобразить кнопку оплаты на сайт и создать платёж на стороне Тинькофф, вызвав метод `Init`.
(<https://oplata.tinkoff.ru/landing/develop/documentation/Init>).

Требования Apple к виду кнопки:

https://developer.apple.com/documentation/apple_pay_on_the_web/displaying_apple_pay_buttons

Локализация кнопок:

https://developer.apple.com/documentation/apple_pay_on_the_web/localizing_apple_pay_buttons

4. После нажатия на кнопку оплаты необходимо создать платёжную сессию.

Пример:

```
var request = {
  countryCode: 'US',
  currencyCode: 'USD',
  supportedNetworks: ['visa', 'masterCard'],
  merchantCapabilities: ['supports3DS'],
  total: { label: 'Your Merchant Name', amount: '10.00' },
}
var session = new ApplePaySession(3, request);
```

Подробнее:

https://developer.apple.com/documentation/apple_pay_on_the_web/apple_pay_js_api/creating_an_apple_pay_session

5. После созданной сессии необходимо обязательно определить два обработчика событий: `onvalidatemerchant` и `onpaymentauthorized`.

- а) С помощью обработчика события `onvalidatemerchant` Apple проводит проверку, что платёж создаётся зарегистрированным продавцом (магазином). Для этого в качестве параметра события присылается адрес одного из серверов Apple, на который необходимо отправить пакет с данными продавца. После получения успешного ответа необходимо вызвать метод `completeMerchantValidation`.

Пример:

```
session.onvalidatemerchant = function (event) {
  var merchantValidateURL = 'Адрес метода продавца для отправки пакета',
      data = {
        appleValidationUrl: event.validationURL
      };

  $.post(merchantValidateURL, data).then(function (result) {
    session.completeMerchantValidation(result.ValidationResult);
  });
};
```

Подробнее:

- https://developer.apple.com/documentation/apple_pay_on_the_web/apple_pay_js_api/providing_merchant_validation
- https://developer.apple.com/documentation/apple_pay_on_the_web/apple_pay_js_api/requesting_an_apple_pay_payment_session
- https://developer.apple.com/documentation/apple_pay_on_the_web/applepaysession/1778015-completemerchantvalidation

- б) С помощью обработчика `onpaymentauthorized` производится авторизация платежа. Результат авторизации необходимо отправить Apple, вызвав метод `completePayment`.

Пример:

```
session.onpaymentauthorized = function (event) {
  var merchantAuthorizeURL =
    'https://securepay.tinkoff.ru/v2/FinishAuthorize',
      data = {
```

```
        EncryptedPaymentData:
        btoa(JSON.stringify(event.payment.token.paymentData))
        TerminalKey: 'Ключ терминала магазина',
        PaymentId: `Номер платежа, полученный на шаге 3`,
        DATA: { ApplePayWeb: true }
    };

    $.post(merchantAuthorizeURL, data).then(function (result) {
        var status;
        if (result.Success) {
            status = ApplePaySession.STATUS_SUCCESS;
        } else {
            status = ApplePaySession.STATUS_FAILURE;
        };
        session.completePayment(status);
    });
};
```

Подробнее:

- https://developer.apple.com/documentation/apple_pay_on_the_web/applepaysession/1778020-onpaymentauthorized
- https://developer.apple.com/documentation/apple_pay_on_the_web/applepaysession/1778012-completepayment
- <https://oplata.tinkoff.ru/landing/develop/documentation/FinishAuthorize>

6. После настройки сессии необходимо вызвать метод begin.

Пример:

```
session.onvalidatemerchant = function (event) {
    //Шаг 5.1
};
session.onpaymentauthorized = function (event) {
    //Шаг 5.2
};
session.begin();
```

Подробнее:

https://developer.apple.com/documentation/apple_pay_on_the_web/applepaysession/1778001-begin