



Visa Token Service now supports iOS SE use cases

Factsheet



Enable secure in-app contactless payments on iOS

Visa Token Service (VTS) now supports iOS Secure Element (SE) use cases, enabling issuers wallets and third-party wallets providers to use Near Field Communication (NFC) technologies to deliver secure contactless payment experiences directly within their own iOS applications. This capability builds on Apple's NFC & SE Platform, introduced with iOS 18.1, which expands support for NFC-based use cases on supported iOS devices*.

VTS integrates with the Visa Mobile Payment Application (VMPA), the Visa iOS SE SDK, and Apple's NFC and SE Platform to support secure provisioning, transaction, and lifecycle management of payment credentials for contactless payments on supported iOS devices.

Potential benefits

Help streamline iOS Secure Element integration

Support integration with Apple Secure Element and NFC technologies through Visa capabilities designed to help streamline iOS wallet development.

Support credential protection

Help safeguard payment credentials through Secure Element capabilities and secure encrypted channels.

Retain wallet ownership and brand control

Enable secure contactless payments on iOS devices while maintaining the customer relationship, wallet experience, branding, and customer engagement model within your own application.

End-to-end token lifecycle management

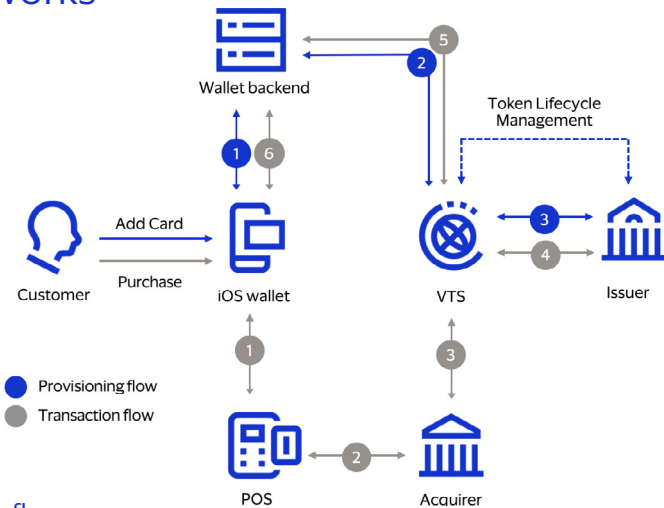
Manage the entire payment credential lifecycle through the Visa Digital Solutions API supporting provisioning, activation, suspension, resumption, deletion, re-personalization, and applet instance state reconciliation.

The total transaction value for contactless card payments is expected to reach **\$15.8 trillion in 2030**, up from just **\$7.7 trillion in 2025²**



OEM Pay userbase will reach **1.1 billion in 2030**, up from **648.5 million in 2025²**.

How it works



Provisioning flow

1. Cardholder adds payment card in the iOS wallet, initiating token provisioning.
2. Wallet backend sends the token request to VTS.
3. VTS validates with issuer and provisions the token and credentials to the device's Secure Element chip.

Transaction flow

1. As a contactless transaction is performed, a token and cryptogram are passed from the user's device to the merchant point of sale.
2. The tokenized transaction is passed from the merchant to the acquirer.
3. The acquirer passes a tokenized authorization request to VTS.
4. VTS detokenizes the card number, sends the authorization request to the issuer, and returns the issuer's response back through the transaction flow to complete payment.
5. When configured, VTS may send a completed transaction event or notification to the wallet backend.
6. Wallet backend then shares the transaction status/history to the iOS wallet.

Features

Device eligibility validation

Validate device, Secure Element, and provisioning readiness before initiating token provisioning, helping improve provisioning success rates.

Secure token provisioning & personalization

Support end-to-end token provisioning, from Secure Element credential setup through token personalization and confirmation.

Secure Element management

Provide applet creation, lifecycle handling, cleanup, and recovery capabilities throughout the payment credential lifecycle, helping to reduce operational complexity and implementation effort.

How can Visa help?

For more information on Visa Token Service, please [visit our website](#) or contact your Visa Account Executive.

Visa for business

For over 65 years, Visa has been an original payments pioneer embracing and shaping technology trends. As transformation across payments unfolds, Visa is enabling businesses, governments, and institutions to embrace new opportunities unlocked by Visa's open platform, backed by our global breadth, local expertise, and commitment to security

*Contact Sales Executive to see availability in your region.

Sources

1. [Apple - NFC & SE Platform for secure contactless transactions](#)
2. [Juniper Research - CONTACTLESS PAYMENTS MARKET DATA:2025-2030](#)