This document will help developer to understand keys and certificates required in VISA digitization.

Note that this document is not shared by VISA but prepared for developer internal use only.

Table of Contents

[Diagrams 1](#_Toc64988584)

[Extracts from VTS API sepcs. 3](#_Toc64988585)

# Diagrams

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

# Extracts from VTS API specs.

## API Key Separation for Signing and Encryption H-1

**API Key Separation for Signing and Encryption**

Visa supports the separation and classification of the signing (authentication) and encryption keys to comply with the PCI-DSS specifications. This is applicable to VTS clients.

An API key can be either of the following:

1. Default key - Support existing clients for backward compatibility. To be deprecated in the future.
2. Inbound authentication key - A client can use this key to create x-pay-token:xv2 (sign) for placing API calls to Visa (Inbound to Visa)
3. Outbound Authentication Key - Visa systems can use this key to sign the outbound post- back messages to the client’s systems.
4. Data encryption Key: This key will be used to encrypt the data that is transmitted between the client’s system and Visa systems.

Clients would have three pool of keys/secrets. Each pool can have one to multiple API key/ shared secret.

1. Inbound authentication key – API key/shared-secret for x-pay-token (version 2: xv2). Clients will use this API-key/secret to create HMAC-SHA256 digests by following x-pay-token (version 2: xv2) protocol.
2. Data encryption key - For data encryption in the payload. There will be no separation of API keys for inbound or outbound APIs.
   1. Inbound APIs to VTS - A client can select any encryption key for the encryption of sensitive data passed to VTS.
   2. Outbound APIs from VTS-Clients should have an encryption key defined in the client profile during onboarding with VTS. If encryption key has not been defined in the profile, VTS will select an encryption key from the pool of encryption key.
3. Outbound authentication key - The API-key/shared-secret for x-pay-token (v2) for Notification APIs from VTS.

A key that is not a default key can have only one restriction. Restrictions can be enabled by contacting Visa customer support. A key cannot be used for both inbound and outbound, for example. Once a key is restricted to a specific usage, that usage cannot be changed or modified. For example, an inbound authentication key can never become an outbound authentication key or an encryption key. If the client desires a new key, they must create new one.

**Versioning of x-pay-token.** X-pay-token is supported in 2 versions: x and xv2 supporting the SHA2 and HMAC-SHA2 algorithms respectively. X version of the x-pay-token is supported only for backward compatibility and it will be deprecated.

**Inbound authentication.** Clients can indicate their preference for x-pay-token version for the inbound APIs, by contacting visa customer support. VTS will validate the indicated version of x- pay-token for inbound authentication.

* If the indicated preference is xv2, only xv2 version will be supported.
* If there is no indication, both x and xv2 version will be supported until x version is deprecated.

**Outbound authentication.** Clients can indicate their preference for x-pay-token version for the post-back APIs, by contacting visa customer support during onboarding in the portal. VTS will create the indicated version of x-pay-token for post message back to clients.

* If the indicated preference is xv2, only xv2 version will be supported.
* If there is no indication, both x and xv2 version will be supported until x version is deprecated.

**Key usage error scenarios:**

If an API key is used improperly in terms of API key classification, the API request would be rejected with an error response by VTS.

1. If an encryption key is used for authentication, “Unauthorized” response will be sent to client.
2. If an authentication key is used for encryption, the API request would be rejected as “bad request” by VTS.
3. Client should reject if the API-key in query parameter is not for the outbound purpose in their key management systems.
4. Visa would reject x-pay-token with version x, if the client is configured for xv2.
5. For the post-back APIs, client should reject if the indicated version of the x-pay-token is xv2 and if it receives x-pay-token with version x.

## x-pay-token

**Enroll device/PAN –**

**Required Headers**

|  |  |
| --- | --- |
| **Header** | **Description** |
| x-request-id | *(Required)* Unique ID for the API request.  **Format:** Alphabetic, numeric, and hyphens ( - ), e.g. spaces are not allowed; maximum 36 characters. |
| x-pay-token | *(Required)* A token identifying the transaction and its contents. The token expires in 480 seconds (8 minutes) for all clients.  **Format:** Alphanumeric; 256 characters in the form of x-pay-token: *xv2:UTC\_Timestamp:HMAC- SHA256\_hash*, where:  • *UTC\_Timestamp*isaUNIXEpochtimestamp, in seconds  • *HMAC-SHA256\_hash*isanHMAC-SHA256 hash using the shared secret associated with the API key and the following unseparated items:   1. Timestamp from the transaction; exactly the same as:*UTC\_Timestamp* 2. Resource path (API name). 3. This HTTPS request's query string, if it exists:   **Note**  To create the query string, concatenate all query string components (names and values) as UTF-8 characters, which are URL-encoded per RFC 3986. Hex characters **must** be uppercase. Multiple parameters must be sorted using lexicographic byte ordering and separated from each other by an ampersand (&) character (ASCII code 38). Parameter names are separated from their values by the = character (ASCII character 61), which must be present even if the value is empty. “Unreserved" characters specified in Section 2.3 of RFC 3986, including dash (-), dot (.), underscore (\_), and tilde (~), should not be URL-encoded.   1. Complete request body, when a request body exists.   **Example:**  x-pay-token:  xv2:1440199445:4a3e6ea1d2...  which is concatenated as follows:  x-pay-token: xv2: + *UTC\_Timestamp* + : + HMAC-SHA256\_hash(*shared\_secret*, (*UTC\_Timestamp + resource\_path + query\_string + request\_body))*  *where*  *shared\_secret is the private key value associated with your API key, which is available in your Visa Developer Center account, resource\_path is the API name / URI, query\_string contains query parameters, such as apiKey=key, and request\_ body is a JSON structure representing the request without extra whitespace.* |

**Query Parameters**

|  |  |
| --- | --- |
| **Parameter** | **Description** |
| apiKey | Client-specific API key issued during onboarding. This will be passed as a query parameter.  **Format:** String. Alphabetic, numeric; maximum 64 characters. |

This **apiKey** is different than above mentioned outbound/inbound API keys.