Overview

This section provides general information about the solution, terminology description and a high-level description of the business and technical aspects.

Abberations and Acronyms

Abbreviations and Acronyms used in the document:

Abbreviations	Description
ACQ	Acquiring Institution/Acquirer
AP	Admin Panel
ACS	Access Control Server
C2C	Card to card
DC	Data Core API
P2P	Peer to peer API
MDC	Mobile Data Core API
SDK	Software Development Kit
THC	Transaction History Core
OS	Operative System
URI	Uniform Resource Identifier
Mid	Merchant Identifier

Terminology

This section explains a number of key terms and concepts used in this document:

Name	Description
Customer	Institution which is using Verestro products. This institution decides which SDK should be used and how transaction should be processed. Basicly Customer can be called Verestro client.
User	User which is using Money Transfer Hub Application. It is root of entity tree. User is identified in Wallet Server by some unique identifier which is provided after registration. User can have access to his data and operations based on session. User's session is created after device pairing is performed. When session expires then user authentication have to be performed. Session is valid 10 minutes, however it is configurable parameter.
Card	Card belongs to the user. User can have many cards. Card is identified via internal id given after storing card on Wallet Server. Whole PAN is stored on Wallet Server which has PCI DSS certificate.
Device	Device belongs to user. When user starts using application after installation then device pairing is performed. After pairing device with some unique id, unique device installation id is generated and this installation is assigned to user. It is possible to have one active installation on specific device for specific user.
Session Token	Token which defines User. It is an authorization way of the User. This entity is created after paring device and this is needed to perform any actions in the application. When session is expired then user authentication needs to be performed. Session is valid 10 minute s, however it is configurable parameter.
Sender	Verestro Wallet user which triggers transaction to the Receiver (check User description).
Receiver	Receiver can be identified in Wallet Server (Internal) or may be an entity that does not exist in Wallet Server (External) ◦ Internal – this type of Receiver has his own unique identifier just like sender. It can also act as a Sender in the transaction process, ◦ External – this type of Receiver does not exist in Wallet Server. Transfers that are made to this type of Receiver require the entering of his card data by Sender.
Mid	Merchant identifier. This entity is representing Merchant in Acquirer's system. Customer have to provide the mid information to enable mid configuration in the Verestro system. Required to process 3DS authentication via Verestro System.

Acquirer	External institution responsible for processing transaction and 3ds requests ordered by the Verestro Money Transfer Hub Application. Acquirer connects with banks / card issuers and returns information whether the ordered action on a given card is possible.
PAN	It is 7-15 digits of credit card number. These digits contain the Permanent Account Number (PAN) assigned by the bank to uniquely identify the account holder.
Wallet Server	Provides the backend services to support Mobile Payment Application via Verestro Wallet SDK and is responsible for managing users, devices, cards , device tokens, storing transactions history and communication with Acquirers.
PCI DSS	PCI DSS (Payment Card Industry Data Security Standard) is a security standard used in environments where the data of payment cardholders is processed. The standard covers meticulous data processing control and protection of users against violations.
QR	QR code is a type of barcode or scannable pattern that contains various forms of data like website links, account information, phone numbers, or even entire object of the transaction.

QR Transactions

QR Transaction is a technology which enables to perform transaction by QR code. This functionality allows to initialize payment by showing QR by Merchant/Receiver to the Sender, which scans it, and pays using Verestro application. The QR code is generated in accordance with the global standard, however, depending on the customer's needs, it can be extended as long as the standard is maintained. This solution requires the 3DS Authentication method, which is described later in the document and it is an individual requirement depending on the Settlement Agent.
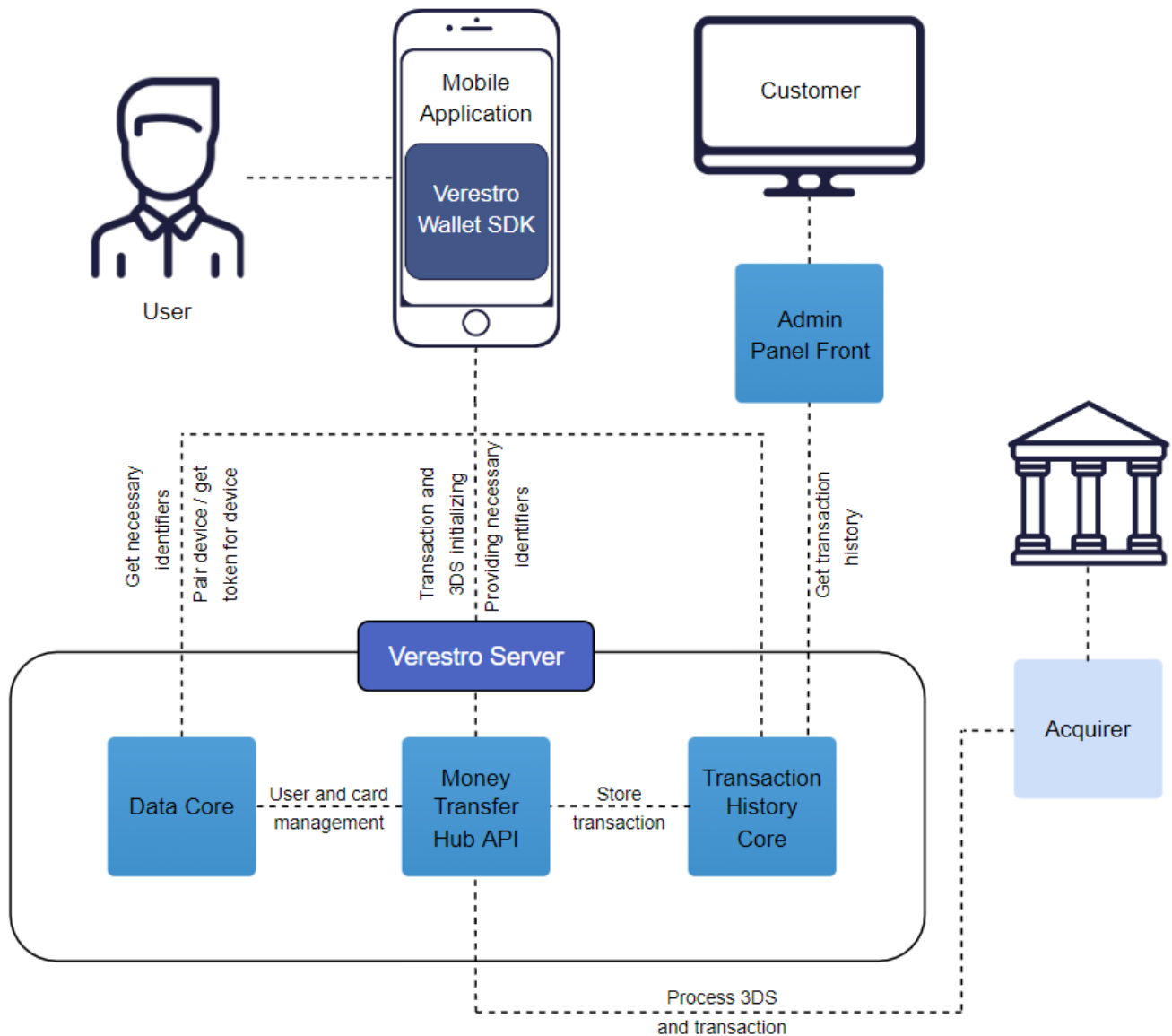
The MC Send 2.0 solution also provides the ability to perform transactions using a QR code - Mastercard QR. However the customer must be integrated with Verestro Money Transfer and Mastercard Send 2.0 to be able to use the MCQR solution.

The MCQR solution offers some features such as:

- crossboarded transfers,
- the possibility of issuing a QR for the customer's clients,
- compliance with the global QR standard.

QR Transactions high level overview

This diagram shows high level components which are involved in whole solution:

QR Transaction Key Components

Component	Description
Verestro Wallet Server	Backend services of Money Transfer solution. In the described product, they are responsible handling data provided from QR code. On the basis of such data, the backend enables the execution of transactions and 3ds authentication by connecting to various Acquirers.
Verestro QR Wallet SDK	Provides all functionalities needed for Money Transfer Hub Solution with QR transfers. It is responsible for generate QR, parsing data embedded in it and deliver necessary data to pass all Verestro Wallet Server functionalities.
Notification Service	Delivers all necessary information about transaction statuses and other actions which was performed between individual Verestro backend components and/or external.

Admin Panel	Frontend component that allows Customer to check transaction statuses and transaction history of his clients.

Verestro QR Money Transfer Hub

Verestro Payment QR Hub is a solution that was created to provide possibility of QR generating and scanning via application. Verestro QR Money Transfer Hub provides functionalities for making transfers based on the data contained in QR codes.

Solution consists of:

- Server components:
  - Wallet Server – backend component,
  - Wallet Admin Panel – frontend component,
- Mobile components:
  - Wallet SDK – Android / iOS libs.

Wallet Types

Money Transfer Hub Solution supports one type of wallet which can be used in the implementation:

- OPEN - user registers itself in the application and provides data like PAN etc.

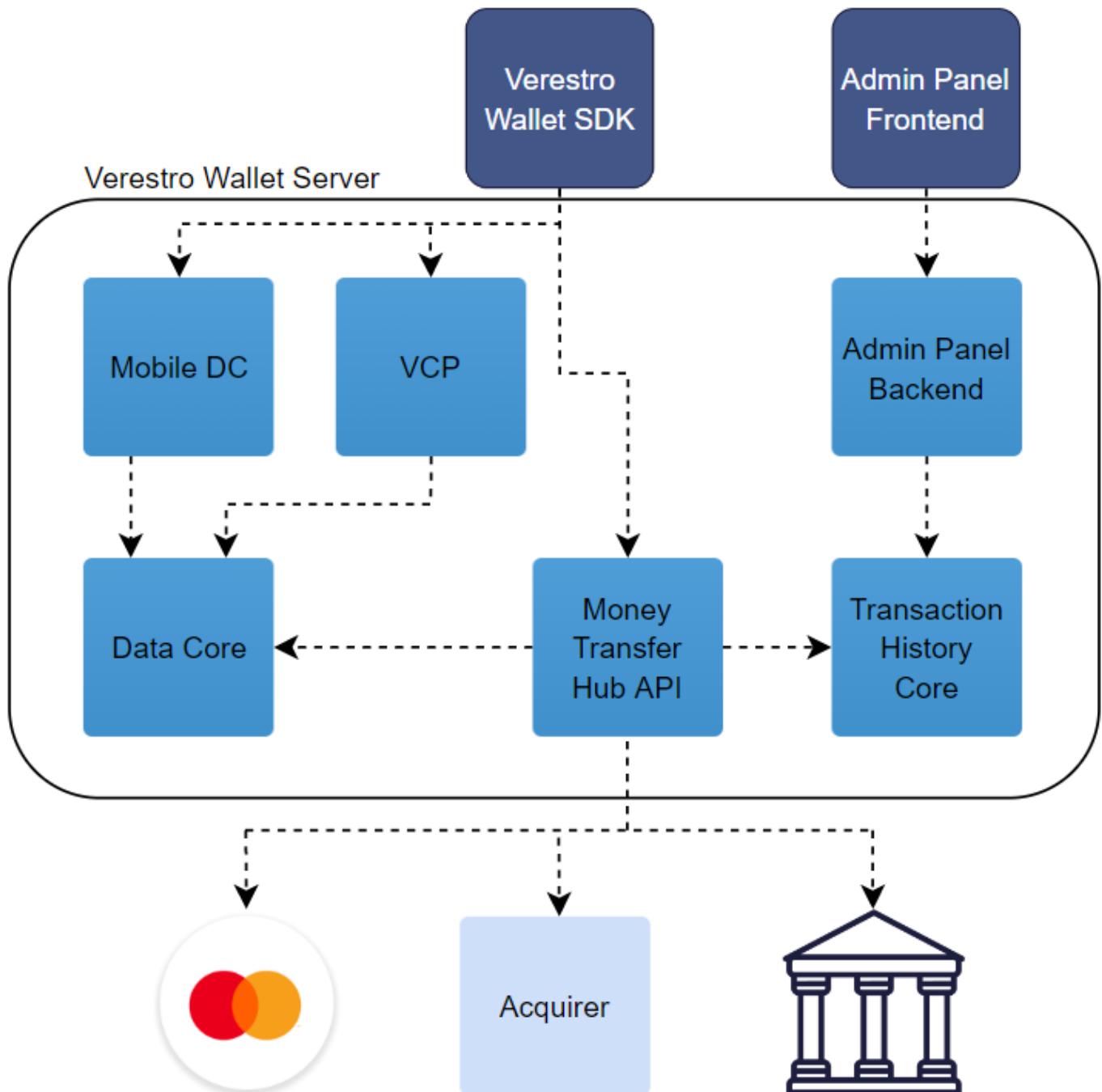*Note that the User can provide generated QR to external entites.*

Implementation models

Verestro provides QR in Software-and-a-Service model hosting the solution for customer. Verestro provides Wallet SDK and Wallet Server. Customer is responsible for integration of provided SDK with his own application and user authentication management (based on MDC SDK).

Verestro can also provide QR payments inside its White Label Application. In this model full implementation process is on Verestro side and project can be launch quickly.

Architecture

This diagram shows big picture of Verestro Money Transfer Hub architecture:

Server Components

Server components are backend services which are designed to process requests from the mobile part, provide the necessary information such as user ID and communicate with Acquirers.

Deployment Models

In QR Money Transfer Hub Solution Server components are deployed and configured on Verestro side. Verestro is responsible for maintaining infrastructure, deploying applications and monitoring.

Wallet Server

Wallet Server is the backend component which consists of few internal services which are responsible for managing users, cards, security tokens, QR payments and transaction history. This component is also responsible for connection with Acquirers. Services included in the Wallet Server. *For more detailed information about Wallet Server component please see [Money Transfer documentation](#).*

Wallet Server operates with domain objects like:

- User (Sender) - User which is using QR Money Transfer Hub,
- Session Token – Token which defines User. It is an authorization way of the User,
- Device – This entity is created after user registration and is required to login the User,
- Card - User Card which can be charged or recharged,
- Receiver - Verestro Wallet user or external entity which receives funds from the Sender.

Mobile Components

Mobile components are dedicated to handle the QR solution on the Android and iOS.

Wallet SDK

QR SDK is responsible for creating the appropriate QR code, parse it and for transferring the data contained in this code. Based on such data, a transaction will be initiated.

 Below is a detailed list of SDKs included in Mobile Components:

- P2P Transfers SDK - supports the process of generating and reporting transactions. The share of this module in the application takes its payment functions to a higher level, enabling the initiation of transfers to a card, telephone number or QR code *(for more technical information please check "P2P Transfer SDK documentation")*.
- QR SDK - The QR module was designed to work with the applicable MPQR (Merchant Presented QR) standard developed by EMV. Thanks to the integration of this module with P2P and meeting the requirements of Mastercard, the user will be able to pay with sellers using QR codes. An additional functionality is that the user can use the code generated for his card and thus receive funds from other people within one implementation *(for more technical information please check "QR SDK documentation")*.

Access

 The account at Verestro Artifactory is required to get access to Verestro repository.

Versioning

SDK version contains three numbers. For example: 1.0.0.:
(For more information check what is "Semantic Versioning" standrand)

- First version digit tracks compatibility-breaking changes in SDK public APIs. It is mandatory to update application code, to use SDK, when this is incremented.
- Second version digit tracks new, not compatibility-breaking changes in public API of SDK. It is optional to update application code, when this digit is incremented.
- Third version digit tracks internal changes in SDK. No updates in application code are necessary to update to version, which has this number incremented.

Changes not breaking compatibility:

- Adding new optional interface to SDK setup,
- Adding new method to any domain,
- Adding new enum value to input or output,
- Adding new field in input or output model.

Communication with Wallet Server

Wallet SDK at the very beginning performs authentication of application and device to Wallet Server.

Security

MDC SDK is responsible for most of the security issues. However, in the Money Transfer Hub solution, sensitive data such as PAN or CVC are processed. They are taken as an array of characters. This data is not held, but immediately wiped from RAM.

Security Checks and Data Clearing

There are performed security checks on Wallet SDK side. Security checks consists of:

- root access detection,
- hooking protection,
- debugging protection,
- custom ROM protection,
- data tampering protection.

Requirements

Wallet SDK has some mandatory requirements to make it work:

- device cannot be rooted,
- Android OS should be in version 6.0 or above,
- iOS OS should be in version 13.0 or above,
- devices cannot have enabled debugging,
- MDC SDK integration.

Configuration

The entire solution requires configuration data necessary for the product to operate in line with the Customer's expectations. *For more detailed information about customer account configuration requirements please see* [*Money Transfer documentation*](.).